



HAL
open science

Méthode multigrilles pour les grandes déformations et simulation numérique du procédé de roulage

Benoît Rey

► **To cite this version:**

Benoît Rey. Méthode multigrilles pour les grandes déformations et simulation numérique du procédé de roulage. Mécanique [physics.med-ph]. École Nationale Supérieure des Mines de Paris, 2007. Français. NNT: . tel-00173519

HAL Id: tel-00173519

<https://pastel.hal.science/tel-00173519>

Submitted on 20 Sep 2007

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

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

Remerciements

Je souhaiterais en premier lieu remercier M. LEGAIT, directeur de l'ENSMP, et M. CHENOT, directeur du CEMEF, de m'avoir permis d'effectuer cette thèse dans leur laboratoire, ainsi que M. COUPEZ, responsable du cycle doctoral de mécanique numérique, sans qui je n'aurais peut être jamais connu le CEMEF.

J'aimerais également exprimer toute ma gratitude aux différents membres du jury qui ont bien voulu juger mon travail. Merci à M. ROUX d'avoir accepté présider le jury, à MM. BOYER et GUILLARD d'avoir accepté d'être rapporteur, ainsi qu'à M. DUBAR.

Cette thèse a été réalisée sous la direction de Katia MOCELLIN et Lionel Fourment au sein des groupes de recherche M3P (Modélisation Mécanique et MultiPhysique), anciennement IEN (Informatique Et Numérique) et CIM (Calcul Intensif en Mise en forme des matériaux). J'aimerais leur témoigner toute ma sympathie et mon respect pour ces trois ans et demi de collaboration fructueuse et enrichissante, tant d'un point de vue personnel que professionnel. Je tiens par la même occasion à remercier Philippe CROISSANT ainsi que toutes les autres personnes de la société SNR qui ont permis le bon déroulement de ce travail. Enfin, je remercie très chaleureusement Christine BERAUDO et Etienne PERCHAT de la société Transvalor qui m'ont énormément apporté durant cette thèse, par leurs conseils et leur connaissance du logiciel, ainsi que par leur très grande disponibilité.

Un grand merci également à mon employeur actuel, les Forges de Courcelles, de m'avoir permis de soutenir cette thèse dans de bonnes conditions.

J'aimerais avoir une pensée amicale envers mes collègues du CEMEF pour tous les moments que l'on a pu partager, que ce soit pour comprendre des sources non commentés, pour prendre un café, jouer au tennis, faire la fête ou simplement discuter. La liste serait trop vaste et forcément non exhaustive, alors simplement merci à tous ceux avec qui j'ai eu l'opportunité de partager un morceau de vie.

Il me faut également remercier les personnes qui sont chères à mon cœur.

Je pense tout d'abord à ma famille. Papa, Maman, Emilie, Papy et Mamy, merci d'avoir toujours été là pour moi. Merci de votre présence, de votre soutien, de votre amour, et merci d'avoir fait de moi ce que je suis. Moi aussi je vous aime.

Merci à mon parrain, sa femme et ma marraine d'avoir assisté à la soutenance et participé à ce grand jour.

Je pense également aux amis que j'ai eu la chance de côtoyer durant ces années passées dans le sud. Merci Tom d'être ce que tu es, même si je sais que ce n'est pas toujours évident de vivre avec soi même. Merci Isa et Matt d'être là et merci pour tous ces moments que l'on a partagé, avec également Véro et Nico. Merci Antoine « papa ours » de ne jamais compter les kilomètres pour se voir et de tenir la distance pour me tenir compagnie en fin de soirée...

Merci au Cinessi, j'ai nommé David, Erven, Sam, Tom (encore toi) et Yann (et lui-même), pour ces folles années d'Ecole. Et merci Oliver (qui n'a pas peur) pour tous ces tennis et ces soirées. Il y a aussi ces amis géographiquement plus lointains. Merci Max et Ben pour votre amitié et votre simplicité. Et merci Alain, d'être venu pour la soutenance mais aussi d'être là après toutes ces d'années, de comprendre mes délires de câbles, de projecteurs et d'enceintes.

Je voudrais finir en te remerciant. Toi qui m'a supporté durant cette épreuve et qui remplie ma vie de bonheur. Anne-Laure, je t'aime. Cette thèse t'est dédiée...

*Les anges aux dents cariées n'attendent rien d'un sourire.
Les anges aux poches percées n'ont que leur grappin pour pas mourir.
Tu verras qu'on s'en sortira sans devenir durs comme du bois.
Nous porterons nos montagnes, on se rencontrera.*

N'aies pas peur, viendra le jour et l'heure, en douceur la vie soufflera ses vraies couleurs.

*Sens-tu que l'amour est en moi, que mon coeur te montre du doigt.
Sens-tu cette liberté nouvelle débarrassée d'un fantôme cruel.
Vois-tu ta beauté dans mes yeux quand ils s'éclairent de ta lumière.
Vois-tu la paix que tu leur confères quand mon poing s'ouvre enfin.
Vois-tu la joie dans les larmes d'un combattant que tu désarmes, fourbu de fourbir chaque
soir ses rasoirs pour trancher dans le noir.
Sais-tu que tes bras seront pour moi tout un monde où réchauffer mes guerres.
Sais-tu que tes bras seront pour moi tout un monde où peut-être enfin je pourrais me taire.*

« Sens-tu » de Mano Solo

Table des matières

Introduction générale	9
1. La simulation du forgeage en 3D	11
1.1. Le problème mécanique	11
1.1.1. Formulation du problème continu	11
1.1.2. Formulation du problème discrétisé	17
1.1.3. Résolution du système	22
1.2. Le problème thermique	28
1.2.1. Equation de la chaleur	28
1.2.2. Conditions aux limites	28
1.2.3. Formulation variationnelle	29
1.2.4. Discrétisation spatiale	30
1.2.5. Discrétisation temporelle	31
1.2.6. Thermique asynchrone	32
1.3. Couplage thermo-mécanique	33
1.4. Modélisation des outils flottants rigides	34
1.4.1. Principe des puissances virtuelles dans l'outil	34
1.4.2. Système à résoudre	36
1.4.3. Contributions liées au contact	37
1.4.4. Contributions liées au frottement	38
1.5. Conclusion	40
2. Le roulage	41
2.1. Présentation du procédé	42
2.1.1. Avant propos : le procédé de laminage circulaire	42
2.1.2. Principe de base du roulage	42
2.1.3. Pilotage de la molette	43
2.1.4. Outillage et pièce	44
2.1.5. Description d'un cycle	45
2.1.6. Motivations de la simulation	46
2.2. Cinématique de la molette	48
2.2.1. Modélisation d'un outil flottant en rotation et en excentration	48
2.2.2. Formalisation du couplage pièce / outil flottant	49
2.2.3. Système à résoudre	50
2.2.4. Contributions liées au contact	50
2.2.5. Contributions liées au frottement	51
2.3. Instrumentation de la machine	53
2.3.1. Partie moteur	53
2.3.2. Matrice	53
2.4. Simulations du procédé de roulage sur le symbole CUV40918	54
2.4.1. Molette, bague finale	54
2.4.2. Données expérimentales	54
2.4.3. Données numériques	56
2.4.4. Première approche : Vitesse d'excentration constante et molette folle en rotation	57
2.4.5. Deuxième approche : pilotage en effort maximal et molette folle en rotation	64
2.5. Etude thermique des outils pour le symbole CUV40918	68
2.5.1. Trajectoire point par point	68

2.5.2.	Recalage thermique	68
2.5.3.	Maillages et temps de calcul	72
2.5.4.	Contraintes équivalentes de Von Mises	73
2.5.5.	Etape de planage	74
2.5.6.	Etapes de laminage et de retour	74
2.5.7.	Etapes d'extraction de la bague et d'insertion du lopin	74
2.5.8.	Evolution de la température du lopin	75
2.5.9.	Remplissage	77
2.5.10.	Bilan de la simulation	77
2.6.	Validation de la simulation sur le symbole BEGB40899	78
2.6.1.	Molette, bague	78
2.6.2.	Données expérimentales	78
2.6.3.	Données numériques	79
2.6.4.	Maillages et temps de calcul	80
2.6.5.	Evolution de la température du lopin au cours d'un cycle	81
2.6.6.	Cycle thermique de la matrice	84
2.6.7.	Remplissage	85
2.7.	Conclusions	86
2.8.	Perspectives	87
3.	Les méthodes multigrilles	89
3.1.	Présentation	89
3.1.1.	Principe	90
3.1.2.	Multigrilles algébriques (AMG)	93
3.1.3.	Multigrilles géométriques	96
3.1.4.	Avantages et inconvénients des deux approches	101
3.2.	La méthode développée	102
3.2.1.	La librairie PETSc	102
3.2.2.	Déraffinement automatique de maillage	104
3.2.3.	Le lissage et la correction	108
3.3.	Influence de la qualité de déraffinement	112
3.3.1.	Influence du taux de déraffinement	112
3.3.2.	Influence de la taille de la grille grossière	113
3.3.3.	Nombre de niveaux	113
3.3.4.	Comparaison entre des méthodes deux et trois grilles	114
3.3.5.	Remaillage	114
3.3.6.	Limitation	115
3.4.	Performances	117
3.4.1.	Convergence asymptotique	117
3.4.2.	Etude sur trois incréments de temps de calcul	118
3.4.3.	Simulation du forgeage d'une bielle	120
3.4.4.	Simulation entière de forgeage d'une bague CUV40918	121
3.4.5.	Simulation entière du forgeage d'une bague BEG40899	124
3.5.	Conclusion	127
3.6.	Perspectives	128
	Conclusion générale	129
	Perspectives	131
	Liste des figures	133
	Liste des tableaux	135
	Bibliographie	137

Introduction générale

L'utilisation de la simulation numérique, encore balbutiante il y a une vingtaine d'années, est en pleine expansion dans le milieu industriel et tend à devenir incontournable. Son utilité est multiple. Tout d'abord elle permet de réduire les temps de développement de nouveaux produits en facilitant les différents choix à effectuer (processus de fabrication, type de matériaux, type de machine, etc.) et en remplaçant les études de faisabilité. Elle permet également de réduire les coûts en optimisant le procédé (gain matière, augmentation de la durée des outils, réduction des temps de cycle, etc.). Afin d'être autant que possible en adéquation avec la réalité, les simulations se doivent d'être précises, tout en conservant des temps de calcul compétitifs. C'est en ce sens qu'est développé au centre de mise en forme des matériaux (CEMEF) le logiciel Forge2005®, logiciel commercialisé par la société Transvalor. Dédié à la simulation de procédé de forgeage (déformation de solide massif par déformation plastique à l'état solide) en trois dimensions, il permet de réaliser des simulations multi-corps afin de connaître les sollicitations thermo-mécaniques imposées aux outils durant la mise en forme du lopin. L'étude de ces sollicitations aide à comprendre la fatigue et les casses constatées sur les outils lors du forgeage réel. On peut ainsi accroître la durée de vie des outils en optimisant la mise en données du procédé grâce au logiciel.

Le procédé de roulage est un procédé original, développé et breveté par SNR. On peut le ranger parmi les procédés de forgeage, et donc le simuler avec un logiciel comme Forge2005®. La simulation de ce procédé, comme souvent celle de nouveau procédé, nécessite un premier travail de mise au point. Il faut tout d'abord adapter le logiciel aux spécificités du procédé, à savoir la cinématique très particulière de l'outillage dans le cadre du roulage. Cette cinématique nécessite également de fournir un travail conséquent sur l'accélération des temps de calcul, afin de rester dans des grandeurs acceptables pour une utilisation à caractère industriel. Enfin une phase d'identification des coefficients thermo-mécaniques régissant le procédé s'avère souvent primordiale, afin d'être le plus prédictif possible. Ce travail, réalisé en partenariat avec les sociétés SNR et Transvalor, est ainsi dédié à la simulation de la mise en forme de bague extérieure de roulement à bille par le procédé de roulage.

Dans un premier chapitre, on présente les équations générales régissant le problème thermo-mécanique dans le cadre du forgeage, ainsi que les discrétisations et les méthodes de résolution utilisés dans le logiciel de calcul implicite Forge2005®. On présente également la modélisation des outils flottants utilisée dans le logiciel.

Dans un deuxième chapitre, on se concentre plus précisément sur le procédé de roulage. On présente un nouveau formalisme d'outils flottants afin de rendre compte de la cinématique particulière de l'outillage. S'en suit une large étude du procédé, ponctuée de confrontations entre résultats numériques et résultats issus d'une campagne expérimentale menée chez SNR. On s'intéresse tout particulièrement au recalage des paramètres thermiques, afin de tendre vers la prédictibilité de la simulation.

Dans un troisième chapitre, on traite du développement d'un nouveau type de résolution de système linéaire, basé sur une méthode multigrilles, permettant de réduire les temps de calcul de manière significative.

1. La simulation du forgeage en 3D

Dans ce chapitre, nous présentons la modélisation numérique de la mise en forme de matériaux par forgeage à chaud, dans le cadre du logiciel Forge2005® et de sa version 3D, que l'on appelle également Forge3®. Dans un premier temps, la résolution du problème mécanique sera abordée. Elle permettra de détailler les équations qui devront être résolus par les nouvelles méthodes présentées dans le dernier chapitre. On s'intéressera ensuite à la résolution du problème thermique puisque ces aspects sont importants pour comprendre les résultats obtenus. Rappelons que les contraintes thermiques sont centrales dans l'analyse de la durée de vie de l'outillage. On détaillera ensuite la gestion du couplage thermo mécanique. Comme nous le verrons dans le deuxième chapitre, les outils de roulage possèdent une cinématique complexe ayant la particularité d'être libre en rotation. Il nous faut donc présenter aussi l'état actuel de la modélisation des outils flottants dans le code. Rappelons tout d'abord que l'on se place à l'échelle macroscopique, la matière ainsi que les outils sont assimilés à des milieux continus et homogènes. On se limite ici à la présentation du problème à résoudre pour un corps déformable. Pour plus de détails sur la résolution du problème multi-corps, le lecteur peut se référer à [Terzolo04] et [Barboza04].

1.1. Le problème mécanique

1.1.1. Formulation du problème continu

Le métal à forger est assimilé à un domaine fini Ω de frontière $\partial\Omega$. Son comportement est régi par les principes fondamentaux de la mécanique des milieux continus que sont la conservation de la masse ainsi que la conservation de la quantité de mouvement. Ses équations sont complétées par des conditions aux limites et par des lois rhéologique et tribologique décrivant respectivement le comportement des matériaux et le comportement de ces domaines aux interfaces. On obtient ainsi le système décrivant le problème mécanique. On utilise une formulation à deux champs, que sont la vitesse v et la pression p . Les équations sont définies en tout point de Ω en fonction du champ (v, p) .

1.1.1.1. Equations de conservation

L'équation de conservation de la masse peut s'écrire sous la forme :

$$\frac{d\rho}{dt} + \text{div}(\rho v) = 0 \quad (1-1)$$

où ρ représente la masse volumique.

La dérivée par rapport au temps de la masse volumique s'annule en considérant l'hypothèse de stationnarité. De plus, le matériau est supposé parfaitement incompressible, on néglige la déformation plastique et la dilatation thermique au cours du procédé, ces

déformations étant faibles devant celles plastiques. Ainsi, la masse volumique est constante dans le temps. L'équation (1-1) se résume donc à :

$$\operatorname{div} v = 0 \quad (1-2)$$

L'équation de conservation de la quantité de mouvement s'écrit :

$$\operatorname{div} \sigma + \rho f = \rho \frac{dv}{dt} \quad (1-3)$$

avec σ le tenseur des contraintes et f les forces volumiques gravitationnelles.

Dans le cadre de la mise en forme des matériaux, il est d'usage d'effectuer les deux hypothèses suivantes :

- les forces de la gravité sont négligeables
- les forces d'inertie sont également négligeables

Par ces deux hypothèses on sous-entend que la matière se déforme seulement sous l'action des outils. Ainsi, l'équation d'équilibre (1-3) du problème se ramène à la forme :

$$\operatorname{div} \sigma = 0 \quad (1-4)$$

On cherche donc à résoudre le système :

$$\begin{cases} \operatorname{div} v = 0 \\ \operatorname{div} \sigma = 0 \end{cases} \quad (1-5)$$

Définissons maintenant les hypothèses de fermeture mécanique de ce système.

1.1.1.2. Conditions aux limites

La surface $\partial\Omega$ de la frontière du domaine est décomposée en plusieurs parties disjointes :

$$\partial\Omega = \partial\Omega_l \cup \partial\Omega_T \cup \partial\Omega_v \cup \partial\Omega_c \quad (1-6)$$

- sur la frontière libre $\partial\Omega_l$, la pièce ne subit aucune contrainte :

$$\sigma n = 0 \quad (1-7)$$

avec n la normale sortante au domaine, visible sur la figure 1-1.

- sur la frontière à chargement imposé $\partial\Omega_T$, la contrainte normale est égale au vecteur contrainte appliquée T :

$$\sigma n = T \quad (1-8)$$

- sur la frontière à vitesse imposée $\partial\Omega_v$, on a simplement égalité des vitesses :

$$v = v_0 \quad (1-9)$$

avec v_0 la vitesse imposée sur $\partial\Omega_v$.

- sur la frontière en contact avec les outils $\partial\Omega_c$, on distingue deux types de conditions imposées:
 - une condition de contact de type unilatéral entre la pièce et les outils décrivant la non pénétration de la matière dans les outils, décrite par les *conditions de Signorini* :

$$\begin{cases} (v - v_{outil}) \cdot n \leq 0 \\ \sigma_n \leq 0 \\ [(v - v_{outil}) \cdot n] \sigma_n = 0 \end{cases} \quad (1-10)$$

où v_{outil} est la vitesse de l'outil, $\sigma_n = \sigma n \cdot n$ est la pression de contact.

- une condition de frottement entre la pièce et les outils. Cette condition tangentielle permet d'exprimer le taux de cisson de frottement τ :

$$\tau = \sigma n - (\sigma n \cdot n) n \quad (1-11)$$

Cette cisson est associée à une loi tribologique, loi qui sera introduite dans le paragraphe suivant.

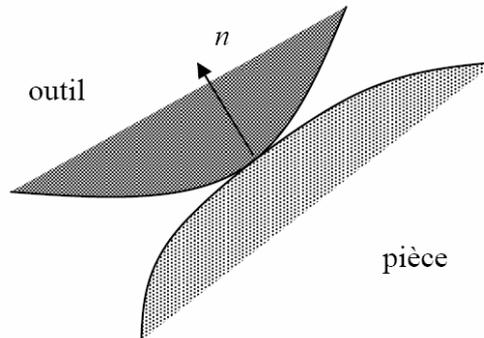


figure 1-1: définition de la normale sortante n

1.1.1.3. Lois de comportement

- Loi rhéologique

Une loi rhéologique caractérise le comportement de la matière au cours de sa mise en forme. Plusieurs types de comportements sont actuellement disponibles dans Forge2005®, on peut citer les comportements de type élastique, élasto-plastique, élasto-viscoplastique ou viscoplastique. Nous nous limitons ici à la présentation de la loi viscoplastique isotrope qui est habituellement utilisé pour le comportement de la pièce forgée pour le forgeage à chaud. La composante élastique de la déformation est encore considérée comme négligeable devant la composante visqueuse. Le lecteur peut se référer à [Terzolo04] pour un passage en revue détaillée des différents types de comportements.

Le comportement viscoplastique est modélisé par une loi de type Norton-Hoff. Cette loi relie le déviateur des contraintes s au tenseur des vitesses de déformation $\dot{\epsilon}$ par une expression non linéaire fonction de la vitesse de déformation généralisée $\dot{\bar{\epsilon}}$, de la consistance du matériau K et de la sensibilité à la vitesse de déformation m :

$$s = 2K \left(\sqrt{3} \dot{\bar{\epsilon}}(v) \right)^{m-1} \dot{\epsilon}(v) \quad (1-12)$$

Le déviateur des contraintes s correspond donc à la partie déviatorique du tenseur des contraintes σ , pI étant la partie sphérique:

$$\sigma = s - pI \quad (1-13)$$

où p est la pression hydrostatique définie par $p = -\frac{1}{3}Trace(\sigma)$ et I le tenseur identité.

Le tenseur des vitesses de déformation s'écrit :

$$\dot{\epsilon}(v) = \frac{1}{2}(\text{grad}v + (\text{grad}v)^T) \quad (1-14)$$

La vitesse de déformation généralisée est définie de la manière suivante :

$$\dot{\bar{\epsilon}}(v) = \sqrt{\frac{2}{3}\dot{\epsilon}(v) : \dot{\epsilon}(v)} \quad (1-15)$$

On définit également la contrainte équivalente de Von Mises $\bar{\sigma}$:

$$\bar{\sigma} = \sqrt{\frac{3}{2}s : s} \quad (1-16)$$

Ainsi que la déformation équivalente $\bar{\epsilon}$:

$$\bar{\epsilon}(v) = \int_0^t \dot{\bar{\epsilon}}(v) dt \quad (1-17)$$

Pour un matériau thermo-écrouissable, l'écrouissage est décrit par la consistance K qui peut également dépendre de la température. Un modèle de matériau thermo-écrouissable est défini, dans le logiciel Forge2005®, par le biais d'une loi puissance reliant la consistance du matériau à la déformation équivalente et à la température :

$$K(T, \bar{\epsilon}) = K_0 (\bar{\epsilon} + \epsilon_0)^n \exp\left(\frac{\beta}{T}\right) \quad (1-18)$$

où K_0 est une constante, n est le coefficient de sensibilité à l'écrouissage, β est le coefficient de sensibilité à la température et ϵ_0 est le terme de régulation de l'écrouissage.

- Loi tribologique

Le frottement se caractérise par l'existence d'une force à l'interface de contact entre deux corps lorsqu'un mouvement de glissement relatif apparaît entre eux. De cette force résulte une contrainte tangentielle, correspondant au taux de cisson τ . Nous utilisons une loi de frottement de type Norton, qui correspond bien au cadre du forgeage à chaud. Pour de plus amples informations sur d'autres types de lois, le lecteur peut se reporter à [Bousseta05].

La loi de Norton relie le taux de cisson à la vitesse de glissement de l'outil Δv_g :

$$\tau = -\alpha_f K \|\Delta v_g\|^{p_f - 1} \Delta v_g \quad (1-19)$$

avec :

$$\Delta v_g = v - v_{outil} - [(v - v_{outil}) \cdot n]n \quad (1-20)$$

où α_f est un coefficient de frottement et p_f un coefficient de sensibilité à la vitesse de glissement.

1.1.1.4. Système à résoudre

La formulation forte de notre problème est donc définie par le système suivant :

$$\begin{cases} \operatorname{div} \sigma = 0 & \text{sur } \Omega \\ \operatorname{div} v = 0 & \text{sur } \Omega \\ \tau = -\alpha_f K \|\Delta v_g\|^{p_f-1} \Delta v_g & \text{sur } \partial\Omega_c \\ (v - v_{outil}) \cdot n \leq 0 & \text{sur } \partial\Omega_c \\ \sigma_n \leq 0 & \text{sur } \partial\Omega_c \\ [(v - v_{outil}) \cdot n] \sigma_n = 0 & \text{sur } \partial\Omega_c \end{cases} \quad (1-21)$$

En décomposant σ en sa partie déviatorique et sa partie sphérique, il vient :

$$\begin{cases} \operatorname{div} s - \operatorname{grad} p = 0 & \text{sur } \Omega \\ \operatorname{div} v = 0 & \text{sur } \Omega \\ \tau = -\alpha_f K \|\Delta v_g\|^{p_f-1} \Delta v_g & \text{sur } \partial\Omega_c \\ (v - v_{outil}) \cdot n \leq 0 & \text{sur } \partial\Omega_c \\ \sigma_n \leq 0 & \text{sur } \partial\Omega_c \\ [(v - v_{outil}) \cdot n] \sigma_n = 0 & \text{sur } \partial\Omega_c \end{cases} \quad (1-22)$$

1.1.1.5. Formulation faible du problème continu

La formulation faible du problème est obtenue par multiplication des équations de conservation par des fonctions tests, puis par intégration sur le domaine. On obtient l'expression faible du problème (1-21), composé du « Principe des Puissances Virtuelles » (1-23a) et de la forme faible de l'équation d'incompressibilité :

$$\begin{aligned} & \text{Trouver } (v, p) \in (V^{ca}, P) \text{ telle que} \\ & \left\{ \begin{aligned} \forall v^* \in V_0^{ca} \quad \int_{\Omega} s : \dot{\epsilon}(v^*) d\omega - \int_{\Omega} p \operatorname{div} v^* d\omega - \int_{\partial\Omega_c} \tau \cdot v^* ds = 0 & (1-23a) \\ \forall p^* \in P \quad \int_{\Omega} p^* \operatorname{div} v d\omega = 0 & (1-23a) \end{aligned} \right. \end{aligned}$$

Les ensembles V^{ca} (espace des vitesses cinématiquement admissibles), V_0^{ca} (espace des vitesses cinématiquement admissibles à zéro) et P sont définis par :

$$\begin{aligned}
 V^{ca} &= \left\{ v \in \left(H^1(\Omega)^3 \mid (v - v_{outil}) \cdot n \leq 0 \text{ sur } \partial\Omega_c \right) \right\} \\
 V_0^{ca} &= \left\{ v \in \left(H^1(\Omega)^3 \mid v \cdot n \leq 0 \text{ sur } \partial\Omega_c \right) \right\} \\
 P &= L^2(\Omega)
 \end{aligned} \tag{1-24}$$

Il est à noter que la condition de contact unilatéral apparaît pour le moment dans la définition des champs cinématiquement admissibles. Cette condition est gérée de manière incrémentale par une méthode de pénalisation, qui sera explicitée par la suite.

La démonstration de l'existence et l'unicité de la solution (v^*, p) dans l'espace (V_0^{ca}, P) n'est pas abordée ici. La curiosité du lecteur peut être guidée vers [Traoré01]. Remarquons simplement que l'existence et l'unicité de la solution v^* dans V_0^{ca} sont assurées par l'équivalence de la première équation de système (1-23a) avec la condition de stationnarité de la fonctionnelle viscoplastique strictement convexe Ψ [Germain85] :

$$\Psi(v) = \int_{\Omega} \frac{K}{m+1} \left(\sqrt{3} \dot{\varepsilon}(v) \right)^{m+1} d\Omega + \int_{\partial\Omega_c} \frac{\alpha_f K}{m+1} \| \Delta v_g \|^m dS \tag{1-25}$$

Le système (1-23a) est donc équivalent à la minimisation de la fonctionnelle viscoplastique, sous la contrainte d'incompressibilité :

$$\begin{cases} \text{Min}_{v \in V^{ca}} \Psi(v) \\ \forall p^* \in P \quad \int_{\Omega} p^* \text{div} v d\omega = 0 \end{cases} \tag{1-26}$$

Le lagrangien L associé à ce problème (1-26) s'écrit ainsi :

$$L : (v, p) \in (V^{ca}, P) \mapsto \Psi(v) - \langle \text{div} v, p \rangle \tag{1-27}$$

La pression p joue le rôle de multiplicateur de Lagrange associé à la contrainte d'incompressibilité.

Le problème (1-23a) à résoudre équivaut alors à l'application des conditions suffisantes de minimalité, compte tenu du fait que Ψ est strictement convexe, donc qu'il y a existence d'un minimum local :

$$\begin{cases} \langle D_v L(v, p), v^* \rangle = 0 \\ \langle D_p L(v, p), p^* \rangle = 0 \end{cases} \tag{1-28}$$

1.1.2. Formulation du problème discrétisé

1.1.2.1. Discrétisation du problème faible

On approxime les espaces d'admissibilités V^{ca} et P par des ensembles discrets de dimensions finies V_h^{ca} et P_h , où h représente la taille de maille moyenne. Ces ensembles V_h^{ca} et P_h sont basés sur une triangulation Γ_h , en éléments ω_h , de l'espace Ω_h . Cet espace discrétise le domaine continu Ω .

$$\Omega_h = \bigcup_{\omega_h \in \Gamma_h} \omega_h \quad (1-29)$$

On cherche donc la solution (v_h, p_h) dans l'espace (V_h^{ca}, P_h) , en souhaitant que (v_h, p_h) converge vers (v, p) lorsque h tend vers 0. Cette convergence est assurée par la consistance du problème, consistance avérée puisque les espaces d'approximation dans lesquels nous travaillons sont des espaces de polynômes.

La formulation couplée du problème en vitesse et pression nous impose également de respecter la condition de Brezzi-Babuska [Babuska73] qui correspond à une condition de stabilité spatiale. Cette restriction va influencer directement sur le choix de l'élément que l'on va utiliser.

1.1.2.2. Discrétisation temporelle

Le code Forge2005® est basé sur une formulation Lagrangienne, réactualisé au cours du temps. Soit t_0 et t_{fin} les temps initial et final de la simulation du forgeage. L'intervalle de temps $[t_0, t_{fin}]$ est discrétisé en plusieurs intervalles $[t_n, t_{n+1}]$. La configuration à l'instant t_{n+1} est calculée à partir de la configuration à l'instant précédent t_n , et non par rapport à l'instant initial t_0 comme en Lagrangien total. Un schéma d'intégration temporelle est donc nécessaire. Notre choix se porte sur un schéma de type Euler explicite :

$$X^{t+\Delta t} = X^t + \Delta t V^t \quad (1-30)$$

On considère la vitesse comme étant constante au cours d'un incrément de temps $[t_n, t_n + \Delta t]$. Certains procédés de forgeage particuliers, comme le laminage circulaire ou le roulage, nécessitent un schéma d'intégration d'ordre supérieur afin de prévenir des variations de volume non négligeables [Traoré01], induites par l'utilisation d'un schéma d'intégration trop simple combinée avec des cinématiques basées sur une ou plusieurs rotations. Le développement d'un schéma d'ordre deux de type Runge-Kutta par [Boyère99] dans le code a permis de résoudre ces problèmes et pourra être utilisé lors de la simulation du procédé de roulage.

1.1.2.3. Discrétisation spatiale

L'élément tétraédrique P1+ / P1 (figure 1-2), dit « mini-élément » est choisi ici pour discrétiser le domaine. Il a été introduit par [Arnold et al.84] et implémenté dans Forge2005® par [Coupez95] pour une formulation viscoplastique incompressible. La vitesse est ainsi approximée par une partie linéaire sur le tétraèdre et est enrichi par une partie bulle linéaire

définie sur les quatre sous tétraèdres formés par les sommets et le centre de gravité du tétraèdre. La pression quand à elle est discrétisée par un tétraèdre linéaire.

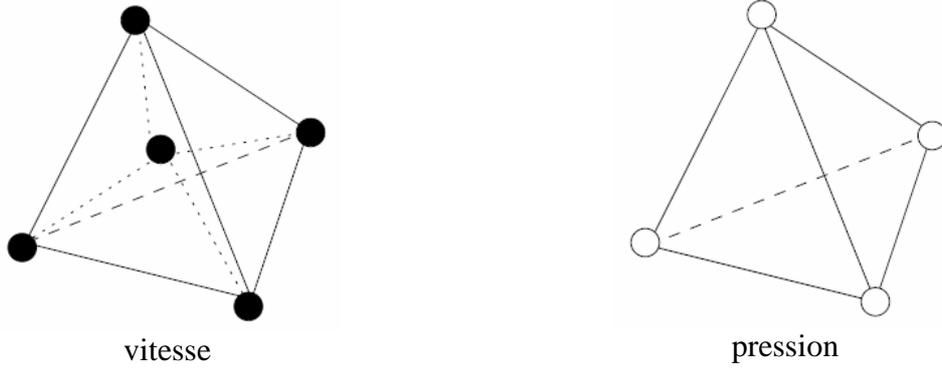


figure 1-2: le mini-élément P1+ / P1

Soit $(\omega_h^k)_{k=1,2,3,4}$ la famille des 4 sous tétraèdres (visible sur la gauche de la figure 1-2) dont le sommet commun est le centre de gravité du tétraèdre.

On peut ainsi définir les ensembles discrets V_h^{ca} et P_h :

$$\begin{aligned}
 P_u &= \left\{ p_h \in C^0(\Omega_h) \quad / \quad \forall \omega_h \in \Gamma_h, \quad p_h|_{\omega_h} \in P^1(\omega_h) \right\} \\
 V_h &= L_h \oplus B_h \\
 L_h &= \left\{ v_h \in (C^0(\Omega_h))^3 \quad / \quad \forall \omega_h \in \Gamma_h, \quad v_h|_{\omega_h} \in (P^1(\omega_h))^3 \right\} \\
 B_h &= \left\{ \begin{array}{l} b_h \in (C^0(\Omega_h))^3 \quad / \quad \forall \omega_h \in \Gamma_h, \quad b_h|_{\omega_h^k} \in (P^1(\omega_h^k))^3, k=1,\dots,4 \\ \text{et } b_h = 0 \quad \text{sur } \partial\Omega_h \end{array} \right\}
 \end{aligned} \tag{1-31}$$

Ce choix d'élément est motivé par la stabilité qu'il confère. Stabilité tout d'abord spatiale, puisqu'il remplit la condition de Brezzi-Babuska, mais stabilité également numérique, puisque l'on verra par suite que la condensation de la bulle améliore le conditionnement de la matrice du système linéaire à résoudre. De plus, les éléments tétraédriques permettent de décrire correctement des volumes complexes, tout en restant facilement maniables pour le maillage et le remaillage [Coupez94]. La méthode de remaillage automatique utilisée dans le code n'est pas détaillée ici, le lecteur peut se référer à [Coupez91]. Notons simplement qu'elle est basée sur une approche locale et procède par amélioration topologique des éléments du maillage.

Notons N^l et N^b les fonctions de base linéaires et bulle. Les champs de vitesse et de pression discrets résultants s'écrivent :

$$\begin{aligned}
 v_h &= v_h^l + v_h^b = \sum_{k=1}^{nbnoe} V_k N_k^l + \sum_{j=1}^{nbel} V_j^b N_j^b \\
 p_h &= \sum_{k=1}^{nbnoe} P_k N_k^l
 \end{aligned} \tag{1-32}$$

où $nbnoe$ (respectivement $nbel$) représente le nombre de noeuds (respectivement le nombre d'éléments) du maillage.

La formulation discrétisée du problème s'obtient en remplaçant dans l'équation (1-23a) les fonctions tests par les fonctions de base ainsi que v par v_h et p par p_h :

$$\left\{ \begin{array}{l} \forall k = 1, \dots, nbnoe; \forall i = 1, \dots, 3 \\ \int_{\Omega_h} 2K \left(\sqrt{3} \dot{\bar{\epsilon}} (v_h^l + v_h^b) \right)^{m-1} \dot{\bar{\epsilon}} (v_h^l + v_h^b) : \dot{\bar{\epsilon}} (N_k^l e_i) d\Omega_h \\ + \int_{\partial\Omega_{ch}} \alpha_f K \left\| \Delta v_{g_h} \right\|^{m-1} (\Delta v_{g_h} \cdot e_i) N_k dS - \int_{\Omega_h} p_h \operatorname{div} (N_k^l e_i) d\Omega_h = 0 \\ \forall e = 1, \dots, nbelt; \forall i = 1, \dots, 3 \\ \int_{\Omega_h^e} 2K \left(\sqrt{3} \dot{\bar{\epsilon}} (v_h^l + v_h^b) \right)^{m-1} \dot{\bar{\epsilon}} (v_h^l + v_h^b) : \dot{\bar{\epsilon}} (N_k^b e_i) d\Omega_h - \int_{\Omega_h^e} p_h \operatorname{div} (N_k^b e_i) d\Omega_h = 0 \\ \forall k = 1, \dots, nbnoe \\ \int_{\Omega_h} N_k^l \operatorname{div} (v_h^l + v_h^b) d\Omega_h = 0 \end{array} \right. \quad (1-33)$$

où e_i est le vecteur unitaire dans la direction i de l'espace.

Le terme de frottement n'intervient pas dans la seconde équation du système (1-33) concernant le champ de vitesse bulle, car celui-ci est nul aux bords des tétraèdres du maillage, donc nul au bord du domaine.

Cette formulation est valable, à cet instant, pour un champ cinématiquement admissible qui vérifie les équations de contact. Cette restriction est levée dans le paragraphe suivant.

1.1.2.4. Gestion incrémentale du contact

N'est présentée ici qu'une description très générale de la gestion incrémentale du contact telle qu'elle a été développée dans Forge2005®. Une présentation complète et détaillée est disponible dans [Mocellin99].

Définissons la notion de distance signée δ à l'outil. En tout point x du domaine Ω définissant la géométrie de la pièce, $\delta(x)$ représente la distance de ce point à la surface de l'outil le plus proche. La convention suivante est utilisée, décrite par la figure 1-3 :

- $\delta(x) > 0$ si le point x est à l'extérieur de l'outil
- $\delta(x) = 0$ si le point x appartient à la surface de l'outil
- $\delta(x) < 0$ sinon (le point x a pénétré dans l'outil)

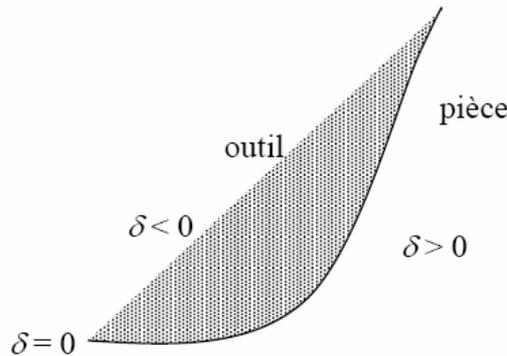


figure 1-3: définition de la distance de pénétration δ

On utilise ici une approche par contact nodal, c'est-à-dire que l'on impose exactement la condition de contact aux nœuds de la surface. La condition de non pénétration (1^{ère} équation de (1-10)) revient à écrire, de manière discrète aux nœuds :

$$\forall k \in \partial\Omega_h \quad \delta_k \geq 0 \quad (1-34)$$

avec $\delta_k = \delta(X_k)$.

Durant un incrément de temps, aucun nœud du domaine ne doit pénétrer les outils. Ces points sont seulement autorisés soit à rester en contact avec l'outil ($\delta_k = 0$), soit à s'en éloigner ($\delta_k > 0$). On impose donc la condition :

$$\forall k \in \partial\Omega_h \quad \delta_k^{t+\Delta t} \geq 0 \quad (1-35)$$

On utilise un développement limité à l'ordre 1 en temps :

$$\forall k \in \partial\Omega_h \quad \delta_k^{t+\Delta t} = \delta_k^t + (V_k - v_{outil}) \cdot n_k^t \Delta t + O(\Delta t^2) \quad (1-36)$$

où n_k^t est la normale sortante à l'outil, au nœud frontière X_k à l'instant t .

On obtient une nouvelle expression de la condition de contact unilatéral, exprimée à l'instant t :

$$\forall k = 1, \dots, nbnoe \quad h(V_k, v_{outil}) = (V_k - v_{outil}) \cdot n_k - \frac{\delta_k}{\Delta t} \leq 0 \quad (1-37)$$

A partir de cette condition de contact unilatéral, on introduit un potentiel de contact (exprimé ici sous sa forme discrète) :

$$\phi_{contact}(v_h, v_{out}) = \frac{1}{2} \rho_c \sum_{k=1}^{nbnoe} S_k [h(V_k, v_{outil})]^2 \quad (1-38)$$

où ρ_c est un coefficient de pénalisation pris suffisamment élevé. $f : x \mapsto [x]^+$ est la fonction partie positive, qui prend la valeur de x si elle est positive et zéro sinon. S_k est la surface associée au nœud k , jouant le rôle de facteur de pondération. Elle est utilisée pour affecter un poids plus fort aux nœuds liés à des faces frontières de plus grande surface et est définie par :

$$S_k = \int_{\partial\Omega_{ch}} N_k ds \quad (1-39)$$

On définit un nouveau Lagrangien à partir de (1-27) qui prend en compte la condition de contact unilatéral pénalisé par l'intermédiaire du potentiel de contact (1-38) :

$$L_\rho(v_h, p_h) = L(v_h, p_h) + \phi_{contact}(v_h, v_{out}) \quad (1-40)$$

En injectant son expression (1-38) dans le Lagrangien (1-40), il vient :

$$L_\rho(v_h, p_h) = L(v_h, p_h) + \frac{\rho_c}{2} \sum_{k=1}^{nbnoe} S_k \left[(v_h - v_{outil}) \cdot n_k - \frac{\delta_k}{\Delta t} \right]^2 \quad (1-41)$$

Soit $1_{\partial\Omega_{ch}}$ la fonction indicative du contact :

$$\forall k \in \partial\Omega_h, \begin{cases} 1_{\partial\Omega_{c_h}}(k) = 1 & \text{ssi } (v_k - v_{outil}) \cdot n_k - \frac{\delta_k}{\Delta t} > 0 \\ 1_{\partial\Omega_{c_h}}(k) = 0 & \text{sinon} \end{cases} \quad (1-42)$$

On peut réécrire le Lagrangien (1-41) du problème de la manière suivante :

$$L_\rho(v_h, p_h) = L(v_h, p_h) + \frac{\rho_c}{2} \sum_{k=1}^{nbnoe} 1_{\partial\Omega_c}(k) S_k \left[(v_h - v_{outil}) \cdot n_k - \frac{\delta_k}{\Delta t} \right]^{+2} \quad (1-43)$$

Définissons maintenant la fonction indicatrice de frottement $1_{\partial\Omega_{f_h}}$ ainsi que la surface de frottement $\partial\Omega_{f_h}$:

$$\begin{aligned} \partial\Omega_{f_h} &= \{x \in \partial\Omega_h \mid \delta(x) \leq 0\} \\ \forall k \in \partial\Omega_h, \begin{cases} 1_{\partial\Omega_{f_h}}(k) = 1 & \text{ssi } \delta_k \leq 0 \\ 1_{\partial\Omega_{f_h}}(k) = 0 & \text{sinon} \end{cases} \end{aligned} \quad (1-44)$$

Pour obtenir la formulation faible finale discrétisée « éléments finis » de notre problème de forgeage, il suffit de réécrire les conditions suffisantes de minimalité (1-28) en utilisant le Lagrangien L_ρ (1-43) qui tient compte de la formulation pénalisée du contact :

$$\left\{ \begin{array}{l} \forall k = 1, \dots, nbnoe; \forall i = 1, \dots, 3 \\ \int_{\Omega_h} 2K \left(\sqrt{3} \dot{\epsilon}(v_h^l + v_h^b) \right)^{m-1} \dot{\epsilon}(v_h^l + v_h^b) : \dot{\epsilon}(N_k^l e_i) d\Omega_h \\ + \int_{\partial\Omega_h} 1_{\partial\Omega_{f_h}}(k) \alpha_f K \left\| \Delta v_{gh} \right\|^{m-1} (\Delta v_{gh} \cdot e_i) N_k^l dS - \int_{\Omega_h} p_h \operatorname{div}(N_k^l e_i) d\Omega_h \\ + \rho_c 1_{\partial\Omega_{c_h}}(k) S_k \left((v_k - v_{outil}) \cdot n_k - \frac{\delta_k}{\Delta t} \right) n_k^i = 0 \\ \forall e = 1, \dots, nbelt; \forall i = 1, \dots, 3 \\ \int_{\Omega_h^e} 2K \left(\sqrt{3} \dot{\epsilon}(v_h^l + v_h^b) \right)^{m-1} \dot{\epsilon}(v_h^l + v_h^b) : \dot{\epsilon}(N_k^b e_i) d\Omega_h - \int_{\Omega_h^e} p_h \operatorname{div}(N_k^b e_i) d\Omega_h = 0 \\ \forall k = 1, \dots, nbnoe \\ \int_{\Omega_h} N_k^l \operatorname{div}(v_h^l + v_h^b) d\Omega_h = 0 \end{array} \right. \quad (1-45)$$

Les inconnues de cette formulation (1-45) que sont la vitesse et la pression sont maintenant exprimées dans des espaces d'admissibilités classiques. Nous allons ainsi pouvoir nous concentrer dans la prochaine partie à la résolution numérique de ce système.

1.1.3. Résolution du système

La résolution du système vectoriel non linéaire (1-45) nous permet de calculer V^l pour passer d'une configuration à un instant t à la configuration suivante à l'instant $t + \Delta t$. Il faut donc résoudre ce système à chaque incrément temporel afin d'accomplir la simulation dans son ensemble. Dans un souci de simplification des notations, réécrivons ce système sous forme matricielle :

$$R(v^l, v^b, p) = \begin{pmatrix} R_l(v^l, v^b, p) \\ R_b(v^l, v^b, p) \\ R_p(v^l, v^b, p) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \quad (1-46)$$

Avec :

$$\begin{aligned} R_l(v^l, v^b, p) &= R_l^l + R_l^p + R_c^l \\ R_b(v^l, v^b, p) &= R_b^b + R_b^p \\ R_p(v^l, v^b, p) &= R_p^l + R_p^b \end{aligned} \quad (1-47)$$

où :

$$\begin{aligned} (R_l^l)_{ik} &= \int_{\Omega_h} 2K(\sqrt{3}\dot{\epsilon}(v_h^l))^{m-1} \dot{\epsilon}(v_h^l) : \dot{\epsilon}(N_k^l e_i) d\Omega_h \\ (R_l^p)_{ik} &= - \int_{\Omega_h} p_h \text{div}(N_k^l e_i) d\Omega_h \\ (R_c^l)_{ik} &= \int_{\partial\Omega_{fh}} (k)\alpha_f K \|\Delta v_{gh}^l\|^{m-1} (\Delta v_{gh}^l \cdot e_i) N_k^l dS + \rho_c 1_{\partial\Omega_{ch}} (k) S_k \left((v_h^l - v_{outil}) n_k - \frac{\delta_k}{\Delta t} \right) n_k^i \end{aligned} \quad (1-48)$$

$$\begin{aligned} (R_b^b)_{ik} &= \int_{\Omega_h} 2K(\sqrt{3}\dot{\epsilon}(v_h^b))^{m-1} \dot{\epsilon}(v_h^b) : \dot{\epsilon}(N_k^b e_i) d\Omega_h & (R_p^l)_{ik} &= \int_{\Omega_h} N_k^l \text{div}(v_h^l) d\Omega_h \\ (R_b^p)_{ik} &= - \int_{\Omega_h} p_h \text{div}(N_k^b e_i) d\Omega_h & (R_p^b)_{ik} &= \int_{\Omega_h} N_k^b \text{div}(v_h^b) d\Omega_h \end{aligned} ,$$

1.1.3.1. Linéarisation du système

L'algorithme de Newton-Raphson est employé afin de résoudre ce système. A partir d'un triplet initial $(v_{init}^l, v_{init}^b, p_{init})$, on cherche une correction $(\Delta v^l, \Delta v^b, \Delta p)$ à apporter telle que :

$$R(v_{init}^l + \Delta v^l, v_{init}^b + \Delta v^b, p_{init} + \Delta p) = 0 \quad (1-49)$$

On approche le résidu R à l'ordre un à l'aide d'un développement de Taylor :

$$R(v_{init}^l + \Delta v^l, v_{init}^b + \Delta v^b, p_{init} + \Delta p) \approx R(v_{init}^l, v_{init}^b, p_{init}) + \frac{\partial R}{\partial v^l} \Delta v^l + \frac{\partial R}{\partial v^b} \Delta v^b + \frac{\partial R}{\partial p} \Delta p \quad (1-50)$$

Le système à résoudre devient donc :

$$\begin{pmatrix} \frac{\partial R_l}{\partial v^l} & \frac{\partial R_l}{\partial v^b} & \frac{\partial R_l}{\partial p} \\ \frac{\partial R_b}{\partial v^l} & \frac{\partial R_b}{\partial v^b} & \frac{\partial R_b}{\partial p} \\ \frac{\partial R_p}{\partial v^l} & \frac{\partial R_p}{\partial v^b} & \frac{\partial R_p}{\partial p} \end{pmatrix} \begin{pmatrix} \Delta v^l \\ \Delta v^b \\ \Delta p \end{pmatrix} = - \begin{pmatrix} R_l \\ R_b \\ R_p \end{pmatrix} \quad (1-51)$$

1.1.3.2. Condensation de la bulle

On peut tout d'abord remarquer que d'après le système (1-47), le terme $\frac{\partial R_p}{\partial p}$ est nul. De plus, le champ bulle possède la propriété que son gradient est orthogonal à tout tenseur constant C sur l'élément e du maillage [Perchat00] :

$$\int_{\Omega_e} C : \dot{\varepsilon}_h(v^b) d\Omega_h = 0 \quad (1-52)$$

Nous pouvons en déduire que :

$$\int_{\Omega_e} \dot{\varepsilon}_h(v^l) : \dot{\varepsilon}_h(v^b) d\Omega_h = 0 \quad (1-53)$$

Donc que :

$$\frac{\partial R_l}{\partial v^b} = \frac{\partial R_b}{\partial v^l} = 0 \quad (1-54)$$

Le système (1-46) se simplifie en :

$$\begin{pmatrix} \frac{\partial R_l}{\partial v^l} & 0 & \frac{\partial R_l}{\partial p} \\ 0 & \frac{\partial R_b}{\partial v^b} & \frac{\partial R_b}{\partial p} \\ \frac{\partial R_p}{\partial v^l} & \frac{\partial R_p}{\partial v^b} & 0 \end{pmatrix} \begin{pmatrix} \Delta v^l \\ \Delta v^b \\ \Delta p \end{pmatrix} = - \begin{pmatrix} R_l \\ R_b \\ R_p \end{pmatrix} \quad (1-55)$$

En extrayant la deuxième équation du système (1-55), il vient :

$$\Delta v^b = - \left(\frac{\partial R_b}{\partial v^b} \right)^{-1} \left(\frac{\partial R_b}{\partial p} \Delta p + R_b \right) \quad (1-56)$$

En injectant dans la troisième équation du système (1-55) cette expression de Δv^b , on obtient :

$$\frac{\partial R_p}{\partial v^l} \Delta v^l - \frac{\partial R_p}{\partial v^b} \left(\frac{\partial R_b}{\partial v^b} \right)^{-1} \frac{\partial R_b}{\partial p} \Delta p = - \left(R_p - \frac{\partial R_p}{\partial v^b} \left(\frac{\partial R_b}{\partial v^b} \right)^{-1} R_b \right) \quad (1-57)$$

On pose $\Delta v = \Delta v^l$ et on définit les matrices hessiennes (ou de raideur) et les gradients :

$$\left\{ \begin{array}{l} K^{vv} = \frac{\partial R_l}{\partial v^l} \\ K^{vp} = \frac{\partial R_l}{\partial p} \\ K^{pv} = \frac{\partial R_p}{\partial v^l} \\ K^{pp} = \frac{\partial R_p}{\partial v^b} \left(\frac{\partial R_b}{\partial v^b} \right)^{-1} \frac{\partial R_b}{\partial p} \end{array} \right. \quad \text{et} \quad \left\{ \begin{array}{l} R_v = R_l \\ R_p \leftarrow R_p - \frac{\partial R_p}{\partial v^b} \left(\frac{\partial R_b}{\partial v^b} \right)^{-1} R_b \end{array} \right. \quad (1-58)$$

Avec ces nouvelles notations, on peut réécrire le système (1-55) qui devient indépendant des termes liés à la vitesse de la bulle :

$$\begin{pmatrix} K^{vv} & K^{vp} \\ K^{pv} & -K^{pp} \end{pmatrix} \begin{pmatrix} \Delta v \\ \Delta p \end{pmatrix} = - \begin{pmatrix} R_v \\ R_p \end{pmatrix} \quad (1-59)$$

On obtient ainsi un système à 4 inconnues par nœuds, 3 composantes pour le champ de vitesse et 1 composante pour le champ de pression par élément. Cette condensation du terme bulle permet d'accroître la stabilité du système, en aboutissant à une matrice mieux conditionnée, puisqu'elle élimine tous les termes nuls anciennement présents sur la diagonale. On se ramène ainsi à une résolution classique avec des champs de vitesse et pression linéaires, mais qui vérifie la condition de compatibilité des espaces de Brezzi-Babuska.

On peut noter que la matrice K^{pp} est symétrique définie positive, ce qui implique que la matrice du système global (1-59) que l'on cherche à résoudre est symétrique, mais n'est pas définie positive.

En pratique, comme la formulation obtenue est équivalente à une formulation stabilisée, on introduit un paramètre de stabilisation β^2 , que l'on ajuste de manière à contrôler la précision et l'efficacité de la formulation. Un cheminement similaire est mené dans [Waren et al.93] dans le cadre de la formulation P1P1 globalement stabilisée, où les auteurs recherchent la valeur optimale du paramètre β^2 minimisant le nombre d'itérations de leur méthode itérative de résolution sans détériorer la qualité de la solution. On résout ainsi le système suivant :

$$\begin{pmatrix} K^{vv} & K^{vp} \\ K^{pv} & -\beta^2 K^{pp} \end{pmatrix} \begin{pmatrix} \Delta v \\ \Delta p \end{pmatrix} = - \begin{pmatrix} R_v \\ R_p \end{pmatrix} \quad (1-60)$$

1.1.3.3. Méthode de résolution

Le système (1-60) est actuellement résolu dans le logiciel Forge2005® par une méthode itérative de type résidu conjugué préconditionné [Marie97]. Il est fondamental de posséder un solveur linéaire le plus compétitif possible. Si l'on étudie le temps de calcul total de la simulation, on constate que près de 80% de ce temps est dédié à la résolution des systèmes linéaires. Une méthode avec une rapidité d'exécution plus grande aura donc un impact direct sur la durée totale de la simulation. Nous présenterons dans le chapitre 3 un solveur plus

compétitif que le solveur classique, en terme de robustesse et de temps de calcul, basé sur une méthode multigrille.

Notons maintenant $Ax = b$ le système que l'on cherche à résoudre. On rappelle que A n'est pas définie positive. On définit la fonctionnelle J par :

$$J(x) = \frac{1}{2}(Ax, x) - (b, x) \quad (1-61)$$

Cette fonctionnelle n'est pas strictement convexe donc l'existence et l'unicité de la solution du problème de minimisation n'est pas assurée. On s'intéresse donc à une méthode de résidu conjugué, où l'on cherche à minimiser la norme du vecteur résidu :

$$E(r) = (r, r) \quad (1-62)$$

où r est le résidu du système défini par :

$$r = b - Ax \quad (1-63)$$

On cherche ainsi à minimiser la fonctionnelle :

$$E(r) = (b - Ax, b - Ax) = (A^2x, x) - 2(Ab, x) + (b, b) \quad (1-64)$$

On se ramène à un problème où l'on résout $A^2x = Ab$, où A^2 est une matrice symétrique définie positive. Ceci assure l'existence et l'unicité du minimum de la fonctionnelle, donc de la solution du système que l'on cherche à résoudre.

Pour accélérer la convergence, on introduit une matrice de préconditionnement M , inversible, symétrique et définie positive. On cherche alors à minimiser la fonctionnelle :

$$E(r) = (M^{-1}r, r) \quad (1-65)$$

On souhaite que M^{-1} soit proche de A^{-1} , pour que le conditionnement de $M^{-1}A$ soit proche de 1.

L'algorithme générique de la méthode de résidu conjugué préconditionné est donné dans le tableau 1-1. Le critère de convergence de la méthode est l'obtention de la réduction voulue du résidu relatif.

tableau 1-1 : Algorithme de Minimisation du résidu

Initialisations :

$$k = 0$$

Choix du nombre de pente $(s + 1)$

$$r_0 = b$$

$$p_0 = M^{-1}r_0$$

$$z_0 = Ap_0$$

Tant que (non convergence)

$$\alpha_k = \frac{(r_k, M^{-1}z_k)}{(M^{-1}r_k, z_k)}$$

$$x_{k+1} = x_k + \alpha_k p_k$$

$$r_{k+1} = r_k - \alpha_k z_k$$

$$j = k - s_k \text{ avec } s_k = \text{Min}(k, s)$$

Tant que ($j \leq k$)

$$\beta_k^j = -\frac{(AM^{-1}r_{k+1}, M^{-1}z_j)}{(M^{-1}z_j, z_j)} \text{ pour } k - s_k \leq j \leq k \text{ avec } s_k = \text{Min}(k, s)$$

$$j \leftarrow j + 1$$

Fin tant que

$$p_{k+1} = M^{-1}r_{k+1} + \sum_{j=k-s_k}^k \beta_k^j p_j$$

$$z_{k+1} = AM^{-1}r_{k+1} + \sum_{j=k-s_k}^k \beta_k^j z_j$$

$$k \leftarrow k + 1$$

Fin tant que

L'introduction du vecteur z_k permet d'effectuer un unique produit matrice vecteur par itération. Il faut également effectué 2 calculs de préconditionnement à chaque itération.

La convergence asymptotique de ce solveur résidu conjugué préconditionné est de $O\left(N^{\frac{3}{2}}\right)$ pour un problème de type Stokes en 3D, avec N le nombre de degrés de liberté du système, égal à quatre fois le nombre de nœuds du maillage dans notre cas.

1.1.3.4. Préconditionnement

Plusieurs types de préconditionneurs sont disponibles dans Forge2005®. Pour une revue en détails ainsi qu'une analyse des performances de chacun, le lecteur peut se référer à [Mocellin99] et [Perchat00]. Nous ne présentons ici que le préconditionnement par factorisation incomplète [Arany96], [Papagarakakis et al.93] utilisé par défaut dans le code. Considérons tout d'abord une décomposition de Crout :

$$A = \bar{L}\bar{D}\bar{L}^T \quad (1-66)$$

où \bar{L} est une matrice triangulaire inférieure de diagonale 1 et \bar{D} est une matrice diagonale. L'existence de cette décomposition est assurée par le fait que A soit symétrique. Cependant, il n'y a pas unicité de cette décomposition. Par abus de notation, elle est généralement noté IC(0), où le (0) désigne le degré de remplissage.

On peut, à partir de cette décomposition, définir des décompositions incomplètes de différents ordres avec des matrices creuses L et D . Notre choix s'arrête sur l'ordre 0, ce qui veut dire que le stockage compact de la matrice L est identique en terme d'espace à celui de la matrice A . Ainsi, on ne calcule que les termes L_{ij} tels que $A_{ij} \neq 0$:

$$\begin{cases} \forall i = 1, \dots, 4nbnoe \\ \forall j = 1, \dots, 4nbnoe \end{cases} \quad \text{pour } i < j \text{ et } A_{ij} \neq 0 \quad L_{ij} = \frac{1}{D_j} \left(A_{ij} - \sum_{k=1, A_{ik} \neq 0 \text{ et } A_{kj} \neq 0}^{j-1} L_{ik} L_{kj} D_k \right) \quad (1-67)$$

$$\forall i = 1, \dots, 4nbnoe \quad D_i = A_{ii} - \sum_{k=1, A_{ik} \neq 0}^{i-1} L_{ik}^2 D_k$$

La matrice de préconditionnement utilisée est donc la suivante :

$$M = LDL^T \quad (1-68)$$

1.2. Le problème thermique

On utilise ici une formulation à un champ, qui est la température T , pour caractériser le système décrivant le problème thermique.

1.2.1. Equation de la chaleur

Les échanges thermiques entre la pièce, les outils et le milieu extérieur sont régis par l'équation de la chaleur :

$$\rho c \frac{dT}{dt} + \text{div}q = f_p \sigma : \dot{\epsilon} \quad \text{sur } \Omega \quad (1-69)$$

avec ρ la densité, c la chaleur massique, T le champ de température et f_p la fraction de puissance de déformation transformée en chaleur (comprise généralement entre 0.9 et 1). Le flux q est défini par une loi de Fourier en fonction du gradient de température et de la conductivité thermique k du matériau. En supposant que le matériau obéit à une loi de conduction isotrope :

$$q = -k \text{grad}T \quad (1-70)$$

L'équation de la chaleur (1-69) devient :

$$\rho c \frac{dT}{dt} - \text{div}(k \text{grad}T) = f_p \sigma : \dot{\epsilon} \quad \text{sur } \Omega \quad (1-71)$$

1.2.2. Conditions aux limites

Les conditions aux limites de ce problème thermique sur la frontière $\partial\Omega$ peuvent être de quatre types différents. Elles résultent des phénomènes physiques de transfert de chaleur. On décompose la frontière du domaine en plusieurs parties disjointes :

$$\partial\Omega = \partial\Omega_{cr} \cup \partial\Omega_{cond} \cup \partial\Omega_{flux} \cup \partial\Omega_{temp} \quad (1-72)$$

- Convection et rayonnement sur $\partial\Omega_{cr}$

De la chaleur est échangée sur les surfaces libres $\partial\Omega_{cr}$ à cause des phénomènes de convection et de rayonnement. L'échange dû à la convection est quantifié par la relation :

$$(-k \text{grad}T).n = h_{conv} (T - T_{ext}) \quad \text{sur } \partial\Omega_{cr} \quad (1-73)$$

où h_{conv} est le coefficient de transfert de convection et T_{ext} la température extérieure.

L'échange dû au rayonnement est quantifié par la relation :

$$(-k \text{grad}T).n = \epsilon \sigma (T^4 - T_{ext}^4) \quad \text{sur } \partial\Omega_{cr} \quad (1-74)$$

où ϵ est l'émissivité et σ la constante de Stephan-Bolzman.

Regroupons les coefficients de transfert de convection et de rayonnement pour n'en avoir qu'un, que l'on nomme h_{cr} . On a :

$$h_{cr} = h_{conv} + \varepsilon\sigma(T^2 + T_{ext}^2)(T + T_{ext}) \quad (1-75)$$

- Conduction sur $\partial\Omega_{cond}$

Sur la surface de contact $\partial\Omega_{cond}$ entre deux corps, il s'opère un transfert de chaleur par conduction. Cet échange n'est pas parfait, d'où l'utilisation d'un coefficient de transfert h_{cond} traduisant la résistance thermique de contact :

$$(-k\text{grad}T).n = h_{cond}(T - T_{face}) \quad \text{sur } \partial\Omega_{cond} \quad (1-76)$$

avec T_{face} la température du corps au point en face du point considéré. h_{cond} peut dépendre de la température et de la pression de contact.

- Flux imposé sur $\partial\Omega_{flux}$

On peut imposer un flux de chaleur Φ_{imp} sur une partie $\partial\Omega_{flux}$ du bord du domaine :

$$\Phi_{imp} = (k\text{grad}T).n \quad \text{sur } \partial\Omega_{flux} \quad (1-77)$$

Dans le cas particulier d'un flux produit par frottement, le flux de chaleur à imposé s'écrit :

$$\Phi_{frott} = \tau \|\Delta v_g\| \quad \text{sur } \partial\Omega_{flux} \quad (1-78)$$

- Température imposée sur $\partial\Omega_{temp}$

On peut enfin tout simplement imposer une température T_{imp} sur une partie $\partial\Omega_{temp}$ du bord du domaine :

$$T = T_{imp} \quad \text{sur } \partial\Omega_{temp} \quad (1-79)$$

1.2.3. Formulation variationnelle

L'équation de la chaleur (1-71) enrichie des conditions aux limites précédemment définies s'écrit sous forme variationnelle :

$$\begin{aligned}
 & \text{Trouver } T \in \Theta / \\
 \forall T \in \Theta_0, & \left\{ \begin{aligned}
 & \int_{\Omega} \rho c \frac{dT}{dt} T^* d\Omega + \int_{\Omega} k \text{grad}(T) \text{grad}(T^*) d\Omega + \int_{\partial\Omega_{cr}} h_{cr} (T - T_{ext}) T^* dS \\
 & + \int_{\partial\Omega_{cond}} h_{cond} (T - T_{face}) T^* dS + \int_{\partial\Omega_{flux}} \Phi_{frott} T^* dS - \int_{\partial\Omega_{flux}} \Phi_{imp} T^* dS \\
 & - \int_{\Omega} f\sigma : \dot{\epsilon} T^* d\Omega = 0
 \end{aligned} \right. \quad (1-80)
 \end{aligned}$$

Avec $\Theta = \{T \in \Omega / T = T_{imp} \text{ sur } \partial\Omega_{temp}\}$ et $\Theta_0 = \{T \in \Omega / T = 0 \text{ sur } \partial\Omega_{temp}\}$.

1.2.4. Discrétisation spatiale

A partir de cette formulation faible, une méthode éléments finis P1 est construite. Elle s'appuie sur le même maillage que celui qui discrétise le problème mécanique. On utilise le même type d'élément tétraédrique, linéaire aux nœuds. On emploie les mêmes fonctions de bases N définies pour le problème mécanique dans le paragraphe 1.1.2.3 :

$$T_h = \sum_{k=1}^{nbnoe} T_k N_k \quad (1-81)$$

En remplaçant les fonctions tests par les fonctions de bases, la formulation variationnelle (1-80) peut s'écrire :

$$C \frac{dT_h}{dt} + K T_h = Q \quad (1-82)$$

avec C la matrice de capacité, K la matrice de conductivité et Q le terme source, appelé également vecteur de chargement thermique :

$$\begin{aligned}
 & \forall (i, j) \in [1, \dots, nbnoe] \times [1, \dots, nbnoe] \\
 & C_{ij} = \int_{\Omega_h} \rho c N_i N_j d\Omega \\
 & K_{ij} = \int_{\Omega_h} k \text{grad}(N_i) \text{grad}(N_j) d\Omega + \int_{\partial\Omega_{hcr}} h_{cr} N_i N_j dS + \int_{\partial\Omega_{hc}} h_c N_i N_j dS \\
 & Q_i = \int_{\partial\Omega_{hcr}} h_{cr} N_i T_{ext} dS + \int_{\partial\Omega_{hcond}} h_{cond} N_i T_{face} dS - \int_{\partial\Omega_{hflux}} N_i \Phi_{frott} dS \\
 & \quad + \int_{\partial\Omega_{hflux}} N_i \Phi_{imp} dS + \int_{\Omega_h} N_i f\sigma : \dot{\epsilon} d\Omega
 \end{aligned} \quad (1-83)$$

Les paramètres thermiques étant supposés dépendre de la température, les matrices de capacité, de conductivité et le terme source dépendent de T . Le problème (1-82) n'est pas linéaire.

1.2.5. Discrétisation temporelle

La discrétisation temporelle est basée sur un schéma de type différences finies à trois niveaux, développée dans [Soyris90]. Lors de la résolution du problème faible (1-82) à l'incrément temporel $[t_n, t_{n+1}]$, au lieu de se placer à l'instant t_n pour effectuer la discrétisation, on se place à un instant t^* appartenant à l'intervalle $[t_n, t_{n+1}]$ dans le but de rendre le schéma plus consistant :

$$t^* = \alpha_1 t_{n-1} + \alpha_2 t_n + \alpha_3 t_{n+1} \quad (1-84)$$

avec t_{n-1} , t_n et t_{n+1} trois pas de temps successifs séparés par des durées $\Delta t_0 = t_n - t_{n-1}$ et $\Delta t = t_{n+1} - t_n$. Les coefficients α_1 , α_2 et α_3 sont choisis de manière à ce que leur somme soit égale à 1.

Le champ de température T^* au temps t^* s'écrit :

$$T^* = \alpha_1 T_{n-1} + \alpha_2 T_n + \alpha_3 T_{n+1} \quad (1-85)$$

et sa dérivée par rapport au temps à l'instant t^* s'écrit :

$$\frac{dT^*}{dt} = \frac{\beta_1 T_{n-1} + \beta_2 T_n}{dt_0} + \frac{\gamma_1 T_n + \gamma_2 T_{n+1}}{dt} \quad (1-86)$$

D'après [Soyris90], le schéma est consistant pour :

$$\begin{cases} \alpha_2 = 1 - \alpha_1 + \left(\frac{1}{2} - \alpha_1\right) \frac{\Delta t_0}{\Delta t} - \frac{1}{2} \gamma_2 \left(1 + \frac{\Delta t_0}{\Delta t}\right) \\ \alpha_3 = \left(\alpha_1 - \frac{1}{2}\right) \frac{\Delta t_0}{\Delta t} + \frac{1}{2} \gamma_2 \left(1 + \frac{\Delta t_0}{\Delta t}\right) \end{cases} \quad (1-87)$$

et inconditionnellement stable pour :

$$\begin{cases} \gamma_2 \geq \frac{1}{2} \\ \alpha_3 > \frac{1}{2} \gamma_2 \end{cases} \quad (1-88)$$

On réécrit le problème (1-82) à résoudre en utilisant les relations (1-85) et (1-86) :

$$C \left(\frac{\beta_1 T_{n-1} + \beta_2 T_n}{\Delta t_0} + \frac{\gamma_1 T_n + \gamma_2 T_{n+1}}{\Delta t} \right) + K.(\alpha_1 T_{n-1} + \alpha_2 T_n + \alpha_3 T_{n+1}) = Q \quad (1-89)$$

$$\Leftrightarrow \left(C \frac{\gamma_2}{\Delta t} + K.\alpha_3 \right) T_{n+1} = Q - K.(\alpha_1 T_{n-1} + \alpha_2 T_n) - C \left(\frac{\beta_1 T_{n-1} + \beta_2 T_n}{\Delta t_0} + \frac{\gamma_1 T_n}{\Delta t} \right)$$

$$\Leftrightarrow GAT_{n+1} = \frac{1}{\alpha_3} \left(Q - K.(\alpha_1 T_{n-1} + \alpha_2 T_n) - C \left(\frac{\beta_1 T_{n-1} + \beta_2 T_n}{\Delta t_0} + \frac{\gamma_1 T_n}{\Delta t} \right) \right)$$

avec $GA = C \frac{\gamma_2}{\alpha_3 \Delta t} + K$

$$GA T_{n+1} = \frac{1}{\alpha_3} [Q + (C \frac{\gamma_2}{\alpha_3 \Delta t} - GA) (\alpha_1 T_{n-1} + \alpha_2 T_n) - C (\frac{\beta_1 T_{n-1} + \beta_2 T_n}{\Delta t_0} + \frac{\gamma_1 T_n}{\Delta t})]$$

$$\Leftrightarrow GA \frac{T^*}{\alpha_3} = \frac{1}{\alpha_3} \left(Q + C \left(\left(\frac{\alpha_1 \gamma_2}{\alpha_3 \Delta t} - \frac{\beta_1}{\Delta t_0} \right) T_{n-1} + \left(\frac{\alpha_2 \gamma_2}{\alpha_3 \Delta t} - \frac{\beta_2}{\Delta t_0} - \frac{\gamma_1}{\Delta t} \right) T_n \right) \right) \quad (1-90)$$

On résout ce système (1-90) par la méthode du gradient conjugué, préconditionné par la diagonale. Une fois le champ T^* calculé, on détermine le champ T_{n+1} comme suit :

$$T_{n+1} = \frac{T^*}{\alpha_3} - (\alpha_1 T_{n-1} + \alpha_2 T_n) \quad (1-91)$$

1.2.6. Thermique asynchrone

Il arrive souvent que le gradient thermique à l'interface entre la pièce et les outils soit très important. Pour minimiser les problèmes numériques qui peuvent résulter de ce choc thermique, on pourrait utiliser un maillage très fin aux interfaces. Cependant, la discrétisation spatiale étant la même pour les problèmes mécanique et thermique, un fort accroissement du nombre de nœuds se traduirait par une forte augmentation du temps de calcul de la solution mécanique. Ainsi, on utilise généralement des maillages calibrés pour le problème mécanique, même si la précision de la résolution du problème thermique s'en ressent.

Pour prendre en compte correctement ce phénomène thermique, une technique de pas de temps asynchrone est utilisée. Elle est basée sur le fait que les chocs thermiques, qui entraînent des oscillations spatio-temporelles du champ de température, sont liés à un temps de pénétration de la chaleur t_p . Ce temps est défini comme celui nécessaire au flux thermique pour parcourir une taille de maille. Avec des éléments de taille de maille caractéristique h , le choc thermique sera pris en compte seulement lorsque le gradient thermique aura traversé l'élément frontière, c'est-à-dire au bout d'un temps caractéristique Δt_{asyn} :

$$\Delta t_{asyn} = \frac{\rho c}{k} h \quad (1-92)$$

On obtient ainsi une relation entre les pas de discrétisation temporelle et spatiale, permettant de minimiser cette instabilité numérique. Le problème thermique doit donc être résolu avec un pas de temps supérieur à Δt_{asyn} . Ce pas pouvant être supérieur au pas de temps utilisé pour résoudre le problème mécanique, on parle de thermique asynchrone. La température correspondant au pas de temps mécanique est ensuite obtenue par simple régression linéaire. Pour une présentation plus en détail, le lecteur peut notamment se référer à [Aliaga00] pour ses travaux sur la thermique.

1.3. Couplage thermo-mécanique

A chaque incrément de temps, les inconnues du problème sont d'une part le couple vitesse / pression et d'autre part la température. Le choix adopté ici est d'utiliser un couplage faible. On ne cherche pas à résoudre le problème couplé par exemple avec un algorithme de point fixe ou une résolution globale (V,P,T) . Les motivations et les justifications de ce choix sont présentées dans [Soyris90]. On résout indépendamment et de façon séquentielle le problème mécanique (1-5) et le problème thermique (1-71). Si l'on se place à l'instant t , le problème mécanique est résolu avec la configuration mécanique et thermique à l'instant t , alors que le problème thermique sera ensuite résolu avec la configuration mécanique (en particulier les termes de dissipation visqueuse et de frottement) à l'instant $t + \Delta t$ calculé précédemment et la thermique à l'instant t . La figure 1-4 résume ce couplage.

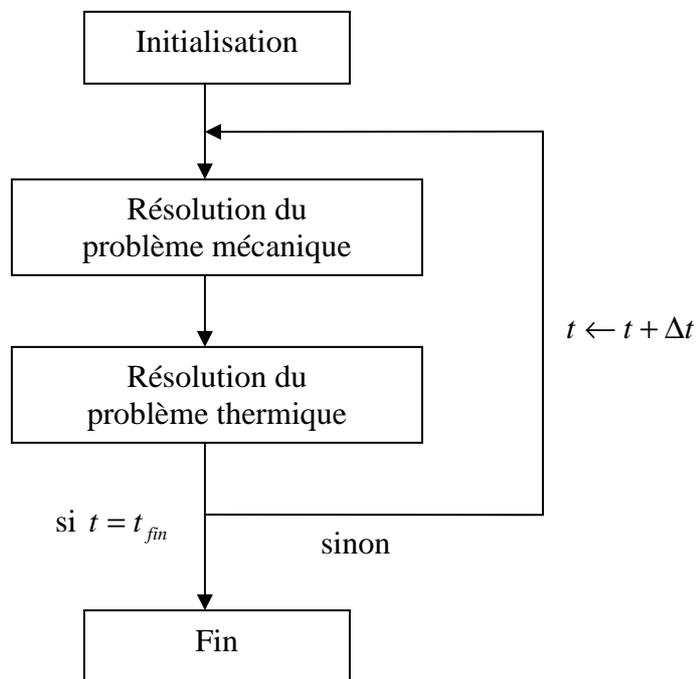


figure 1-4: couplage thermo-mécanique dans Forge2005®

1.4. Modélisation des outils flottants rigides

En règle générale, le déplacement des outils de forgeage est fonction de la nature de la presse utilisée. Leur cinématique est ainsi connue a priori. Cependant dans certains procédés particuliers, le déplacement des outils suivant certaines directions dépend des forces extérieures, de l'écoulement de la matière et des forces de contacts résultantes. Leur mouvement est dit « flottant », et est en interaction avec celui de la matière. Le problème mécanique inclut la détermination du mouvement des outils flottants. Cette problématique a été étudiée et implémentée dans Forge2005® par [Boyère99] et [Traoré01].

Le problème mécanique (1-45) reste quasiment inchangé, à la différence près que v_{outil} n'est plus une donnée, mais une inconnue du système. Intéressons nous alors aux équations du problème des outils flottants, en les définissant comme étant rigides. Le mouvement d'un outil est alors entièrement déterminé par six degrés de liberté, les trois premiers correspondant à un vecteur vitesse de translation et les trois autres à un vecteur vitesse de rotation. Si le procédé simulé contient n outils flottants, le système à résoudre aura $6n$ degrés de liberté supplémentaires. Pour ne pas surcharger l'exposé, on se restreint au cas où seulement un outil est considéré comme flottant.

Soit O le centre d'inertie d'un outil flottant rigide (figure 1-5), v_o sa vitesse de translation et ω sa vitesse de rotation. La vitesse v_{outil} de tout point M de l'outil flottant peut s'écrire :

$$v_{outil}(M) = v_o + \omega \wedge OM \quad (1-93)$$

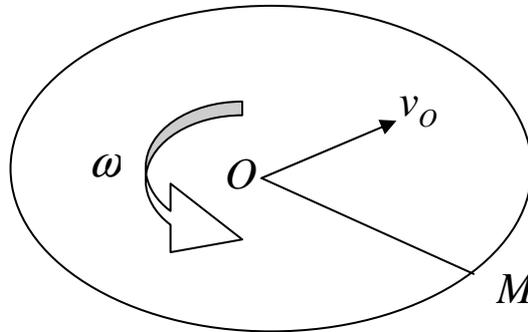


figure 1-5: modélisation d'un outil flottant rigide

On suppose également que l'outil flottant peut être contrôlé par des actionneurs, ce qui se traduit par l'application d'une force F et d'un couple Γ extérieures sur l'outil. Ainsi, il peut soit être libre de tout mouvement ($F = 0$ et $\Gamma = 0$), où alors une force ou un couple lui sont imposés.

1.4.1. Principe des puissances virtuelles dans l'outil

De la même manière que précédemment (1-23a) pour le matériau à forger, on applique le principe des puissances virtuelles à l'outil flottant, en tenant compte de la gestion pénalisée du contact en utilisant le potentiel de contact (1-38), sous sa forme continue :

$$\forall (v_o^*, \omega^*) \in (O^*, O^*) \quad - \int_{\partial\Omega_f} \tau \cdot v_{outil}^* ds - \frac{\partial \phi_{contact}}{\partial v_{outil}} = F \cdot v_o^* + \Gamma \cdot \omega^* \quad (1-94)$$

Avec $O^* = \left\{ \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \right\}$. v_o^* et ω^* représentent respectivement la vitesse de translation virtuelle et la vitesse de rotation virtuelle de l'outil.

La disparition de l'intégrale sur le volume traduisant le comportement rhéologique est due au fait que l'outil est rigide. L'apparition des forces extérieures appliquées à l'outil dans le second membre traduit les asservissements éventuels de l'outil.

En décomposant la vitesse de l'outil flottant par l'intermédiaire de (1-93) et en passant en mode discret, il vient :

$$\begin{cases} \forall v_o^* \in O^* & - \int_{\partial\Omega_{fh}} \tau \cdot v_o^* ds - \frac{\partial \phi_{contact}}{\partial v_o} = F \cdot v_o^* \\ \forall \omega^* \in O^* & - \int_{\partial\Omega_{fh}} \tau \cdot (\omega^* \wedge OM) ds - \frac{\partial \phi_{contact}}{\partial \omega} = \Gamma \cdot \omega^* \end{cases} \quad (1-95)$$

Exprimons les dérivées discrètes du potentiel de contact par rapport aux vitesses virtuelles de translation et de rotation de l'outil :

$$\begin{aligned} \forall i = 1, \dots, 3 \quad \frac{\partial \phi_{contact}}{\partial v_{O_i}} &= -\rho \sum_{k \in \partial\Omega_{ch}} S_k [h_k(v_k, v_{outil})]^+ n_k^i \\ \forall i = 1, \dots, 3 \quad \frac{\partial \phi_{contact}}{\partial \omega_i} &= -\rho \sum_{k \in \partial\Omega_{ch}} S_k [h_k(v_k, v_{outil})]^+ (e_i \wedge OM) \cdot n_k \end{aligned} \quad (1-96)$$

où n_k^i est la $i^{\text{ème}}$ composante de n_k .

On peut alors expliciter les équations discrètes finales du système (1-95) à résoudre, en utilisant ici une loi de frottement viscoplastique :

$$\left\{ \begin{aligned} &\forall i = 1, \dots, 3 \\ &- \int_{\partial\Omega_{fh}} \alpha_f K \|\Delta v_{g_h}\|^{m-1} (\Delta v_{g_h} \cdot e_i) dS + \rho \sum_{k \in \partial\Omega_{ch}} 1_{\partial\Omega_{ch}}(k) S_k \left((v_k - v_{outil}) \cdot n_k - \frac{\delta_k}{\Delta t} \right) n_k^i = F_i \\ &\forall i = 1, \dots, 3 \\ &- \int_{\partial\Omega_{fh}} \alpha_f K \|\Delta v_{g_h}\|^{m-1} (\Delta v_{g_h} \cdot (e_i \wedge OM)) dS \\ &+ \rho \sum_{k \in \partial\Omega_{ch}} 1_{\partial\Omega_{ch}}(k) S_k \left((v_h - v_{outil}) \cdot n_k - \frac{\delta_k}{\Delta t} \right) (e_i \wedge OM) \cdot n_k = \Gamma_i \end{aligned} \right. \quad (1-97)$$

Qui s'écrit sous la forme :

$$R(v, p, v_{outil}) = \begin{pmatrix} R_v(v, p, v_{outil}) \\ R_p(v, p, v_{outil}) \\ R_{v_{outil}}(v, p, v_{outil}) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \quad (1-98)$$

1.4.2. Système à résoudre

Le système à résoudre est défini par la réunion des problèmes mécaniques du matériau (1-45) et de l'outil flottant (1-97). L'ajout de degrés de liberté supplémentaires ne modifie pas les équations présentées en 1.1.3.1 quant à la linéarisation pour la résolution par l'algorithme de Newton-Raphson, et en 1.1.3.2 quant à la condensation de la bulle. Ainsi, le système que l'on résout reprend les équations (1-59) enrichie des degrés de liberté de l'outil flottant :

$$\begin{pmatrix} K^{vv} & K^{vp} & K^{vv_{outil}} \\ K^{pv} & -K^{pp} & 0 \\ K^{v_{outil}v} & 0 & K^{v_{outil}v_{outil}} \end{pmatrix} \begin{pmatrix} \Delta v \\ \Delta p \\ \Delta v_{outil} \end{pmatrix} = - \begin{pmatrix} R_v \\ R_p \\ R_{v_{outil}} \end{pmatrix} \quad (1-99)$$

avec :

$$K^{vv_{outil}} = \frac{\partial R_{v_{outil}}}{\partial v} \quad K^{v_{outil}v} = \frac{\partial R_v}{\partial v_{outil}} \quad K^{v_{outil}v_{outil}} = \frac{\partial R_{v_{outil}}}{\partial v_{outil}} \quad (1-100)$$

La sous matrice $K^{v_{outil}v_{outil}}$ est symétrique et $K^{vv_{outil}} = (K^{v_{outil}v})^T$, ce qui permet à la matrice globale du système globale de rester symétrique.

En décomposant les inconnues liées à l'outil flottant en leurs parties vitesse de translation et vitesse de rotation, le système (1-99) devient :

$$\begin{pmatrix} K^{vv} & K^{vp} & K^{vv_o} & K^{v\omega} \\ K^{vp^T} & -K^{pp} & 0 & 0 \\ K^{vv_o^T} & 0 & K^{v_o v_o} & K^{v_o \omega} \\ K^{v\omega^T} & 0 & K^{v_o \omega^T} & K^{\omega\omega} \end{pmatrix} \begin{pmatrix} \Delta v \\ \Delta p \\ \Delta v_o \\ \Delta \omega \end{pmatrix} = - \begin{pmatrix} R_v \\ R_p \\ R_{v_o} \\ R_\omega \end{pmatrix} \quad (1-101)$$

avec :

$$\begin{aligned} K^{vv_o} &= \frac{\partial R_v}{\partial v_o} & , & & K^{v_o v_o} &= \frac{\partial R_{v_o}}{\partial v_o} & , & & K^{\omega\omega} &= \frac{\partial R_\omega}{\partial \omega} \\ K^{v\omega} &= \frac{\partial R_v}{\partial \omega} & , & & K^{v_o \omega} &= \frac{\partial R_{v_o}}{\partial \omega} & , & & & \end{aligned} \quad (1-102)$$

On explicite les nouvelles contributions à la matrice et au second membre du système en fonction de leur nature (termes de contact ou termes de frottements) sachant que ces contributions s'ajoutent indépendamment les unes aux autres.

1.4.3. Contributions liées au contact

Pour simplifier les notations, on introduit le vecteur m défini tel que $(\omega \wedge OM).n = \omega.m$. On a $(\omega \wedge OM).n = (OM \wedge n).\omega$, d'où $m = (OM \wedge n)$. On peut expliciter m de la manière suivante :

$$(\omega \wedge OM).n_k = \omega.m_k$$

$$m_k = \begin{pmatrix} (M_2 - O_2)n_k^3 - (M_3 - O_3)n_k^2 \\ (M_3 - O_3)n_k^1 - (M_1 - O_1)n_k^3 \\ (M_1 - O_1)n_k^2 - (M_2 - O_2)n_k^1 \end{pmatrix} \quad (1-103)$$

On peut ainsi expliciter les contributions au gradient du contact :

$$\forall k = 1, \dots, nbnoe; \quad \forall i = 1, 2, 3$$

$$(R_v)_{ik}^{contact} = \rho 1_{\partial\Omega_{ch}}(k) S_k \left(V_k \cdot n_k - v_O \cdot n_k - \omega.m_k - \frac{\delta_k}{\Delta t} \right) n_k^i N_k$$

$$(R_{v_O})_i^{contact} = -\rho \sum_{k \in \partial\Omega_h} 1_{\partial\Omega_{ch}}(k) S_k \left(V_k \cdot n_k - v_O \cdot n_k - \omega.m_k - \frac{\delta_k}{\Delta t} \right) n_k^i \quad (1-104)$$

$$(R_\omega)_i^{contact} = -\rho \sum_{k \in \partial\Omega_h} 1_{\partial\Omega_{ch}}(k) S_k \left(V_k \cdot n_k - v_O \cdot n_k - \omega.m_k - \frac{\delta_k}{\Delta t} \right) m_k^i$$

De même, on obtient les contributions à la matrice hessienne des termes de contact:

$$\forall k = 1, \dots, nbnoe; \quad \forall \lambda = 1, \dots, nbnoe; \quad \forall i = 1, 2, 3$$

$$(K^{vv})_{\lambda ik}^{contact} = \frac{\partial}{\partial v_{O\lambda}} (R_v)_{ik}^{contact} = -\rho 1_{\partial\Omega_{ch}}(k) S_k n_\lambda^i n_k^i N_k$$

$$(K^{v_O v_O})_{\lambda i}^{contact} = \frac{\partial}{\partial v_{O\lambda}} (R_{v_O})_i^{contact} = \rho \sum_{k \in \partial\Omega_h} 1_{\partial\Omega_{ch}}(k) S_k n_\lambda^i n_k^i$$

$$(K^{v\omega})_{\lambda ik}^{contact} = \frac{\partial}{\partial \omega_\lambda} (R_v)_{ik}^{contact} = -\rho 1_{\partial\Omega_{ch}}(k) S_k m_k^\lambda n_k^i N_k \quad (1-105)$$

$$(K^{v_O \omega})_{\lambda i}^{contact} = \frac{\partial}{\partial \omega_\lambda} (R_{v_O})_i^{contact} = \rho \sum_{k \in \partial\Omega_h} 1_{\partial\Omega_{ch}}(k) S_k m_k^\lambda n_k^i$$

$$(K^{\omega\omega})_{\lambda i}^{contact} = \frac{\partial}{\partial \omega_\lambda} (R_\omega)_i^{contact} = -\rho \sum_{k \in \partial\Omega_h} 1_{\partial\Omega_{ch}}(k) S_k m_k^\lambda m_k^i$$

1.4.4. Contributions liées au frottement

On remarque que :

$$\Delta v_{g_h} \cdot (e_i \wedge OM) = (OM \wedge \Delta v_{g_h})_i \quad (1-106)$$

D'où les termes de gradients traduisant le frottement :

$$\begin{aligned} \forall k = 1, \dots, nbnoe; \quad \forall i = 1, 2, 3 \\ (R_v)_{ik}^{frott} &= \int_{\partial\Omega_{f_h}} \alpha_f K \|\Delta v_{g_h}\|^{m-1} \Delta v_{g_h}^i N_k dS \\ (R_{v_o})_i^{frott} &= - \int_{\partial\Omega_{f_h}} \alpha_f K \|\Delta v_{g_h}\|^{m-1} \Delta v_{g_h}^i dS \\ (R_{\omega})_i^{frott} &= - \int_{\partial\Omega_{f_h}} \alpha_f K \|\Delta v_{g_h}\|^{m-1} (OM \wedge \Delta v_{g_h})_i dS \end{aligned} \quad (1-107)$$

Il faut maintenant expliciter la valeur des dérivées de la vitesse de glissement par rapport aux vitesses de translation et de rotation de l'outil. On note δ_{ij} le symbole de Kronecker, δ_{ijk} le symbole Levi-Civita et on utilise la convention de sommation d'Einstein afin d'alléger autant que possible les formules :

$$\begin{aligned} \forall i = 1, 2, 3 \\ \frac{\partial}{\partial v_{O_i}} \|\Delta v_{g_h}\|^{m-1} &= -(m-1) \|\Delta v_{g_h}\|^{m-3} \Delta v_{g_h}^i \\ \frac{\partial}{\partial v_{O_i}} \Delta v_{g_h}^j &= -\delta_{ij} + n_k^i n_k^j \\ \frac{\partial}{\partial \omega_i} \|\Delta v_{g_h}\|^{m-1} &= (m-1) \|\Delta v_{g_h}\|^{m-3} (\Delta v_{g_h} \wedge OM)_i \\ \frac{\partial}{\partial \omega_i} \Delta v_{g_h}^j &= -OM_\alpha \delta_{ij\alpha} + m_k^i n_k^j \end{aligned} \quad (1-108)$$

On peut ainsi expliciter les contributions à la matrice de raideur liées au frottement :

$$\forall k = 1, \dots, nbnoe; \quad \forall \lambda = 1, \dots, nbnoe; \quad \forall i = 1, 2, 3$$

$$\begin{aligned} (K^{vv_o})_{\lambda ik}^{frott} &= \frac{\partial}{\partial v_{o\lambda}} (R_v)_{ik}^{frott} \\ &= - \int_{\partial\Omega_{f_h}} \alpha_f K \left((m-1) \|\Delta v_{g_h}\|^{m-3} \Delta v_{g_h}^\lambda \Delta v_{g_h}^i + \|\Delta v_{g_h}\|^{m-1} (-\delta_{ij} + n_k^\lambda n_k^i) \right) N_k dS \end{aligned}$$

$$\begin{aligned} (K^{v_o v_o})_{\lambda i}^{frott} &= \frac{\partial}{\partial v_{o\lambda}} (R_{v_o})_i^{frott} \\ &= \int_{\partial\Omega_{f_h}} \alpha_f K \left((m-1) \|\Delta v_{g_h}\|^{m-3} \Delta v_{g_h}^\lambda \Delta v_{g_h}^i + \|\Delta v_{g_h}\|^{m-1} (-\delta_{ij} + n_k^\lambda n_k^i) \right) dS \end{aligned}$$

$$\begin{aligned} (K^{v\omega})_{\lambda ik}^{frott} &= \frac{\partial}{\partial \omega_\lambda} (R_v)_{ik}^{frott} \\ &= \int_{\partial\Omega_{f_h}} \alpha_f K \left((m-1) \|\Delta v_{g_h}\|^{m-3} (\Delta v_{g_h} \wedge OM)_\lambda \Delta v_{g_h}^i \right. \\ &\quad \left. + \|\Delta v_{g_h}\|^{m-1} (-OM_\alpha \delta_{\lambda i \alpha} + m_k^\lambda n_k^i) \right) N_k dS \end{aligned} \tag{1-109}$$

$$\begin{aligned} (K^{v_o \omega})_{\lambda i}^{frott} &= \frac{\partial}{\partial \omega_\lambda} (R_{v_o})_i^{frott} \\ &= \int_{\partial\Omega_{f_h}} \alpha_f K \left((m-1) \|\Delta v_{g_h}\|^{m-3} (\Delta v_{g_h} \wedge OM)_\lambda \Delta v_{g_h}^i \right. \\ &\quad \left. + \|\Delta v_{g_h}\|^{m-1} (-OM_\alpha \delta_{\lambda i \alpha} + m_k^\lambda n_k^i) \right) dS \end{aligned}$$

$$\begin{aligned} (K^{\omega\omega})_{\lambda i}^{frott} &= \frac{\partial}{\partial \omega_\lambda} (R_\omega)_i^{frott} \\ &= - \int_{\partial\Omega_{f_h}} \alpha_f K \left((m-1) \|\Delta v_{g_h}\|^{m-3} (\Delta v_{g_h} \wedge OM)_\lambda (OM \wedge \Delta v_{g_h})_i \right. \\ &\quad \left. + \|\Delta v_{g_h}\|^{m-1} (-OM_\alpha \delta_{\lambda \alpha \beta} + m_k^\lambda n_k^\beta) OM_\gamma \varepsilon^{i\beta\gamma} \right) dS \end{aligned}$$

1.5. Conclusion

En l'état actuel de la modélisation des outils flottants dans le logiciel, il est impossible de retranscrire la cinématique réelle de l'outillage dans le cadre du roulage. La première partie du chapitre suivant présente donc l'introduction d'un nouveau formalisme plus générique des outils flottants afin de pouvoir notamment simuler au plus juste le roulage.

Concernant les temps de calcul d'une simulation, on estime entre 60 et 70% le temps passé à résoudre les systèmes linéaires (1-60). Le troisième chapitre traite donc du développement d'une nouvelle méthode de résolution de ces systèmes, basée sur une approche multigrilles, permettant de réduire très significativement les temps de calcul.

2. Le roulage

Dans ce chapitre, on présente le procédé de mise en forme de roulage d'anneaux à mi-chaud. C'est un procédé développé par SNR qui permet de forger des bagues extérieures de roulement à billes. La figure 1-3 présente une vue éclatée d'un roulement standard, sur laquelle apparaît la bague extérieure qui nous intéresse ici.

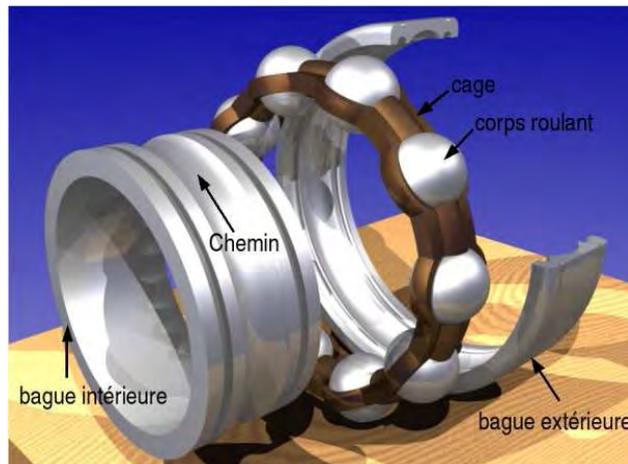


figure 2-1: vue éclatée d'un roulement à bille

Est abordé dans un premier temps une description détaillée du procédé de roulage et son analogie avec le procédé de laminage circulaire [Traoré01]. Vient ensuite un passage en revue de l'ensemble des développements apportés au code Forge2005® pour simuler au mieux ce procédé. Ils portent sur la cinématique de la molette et son caractère libre en rotation. On décrira ensuite le contexte expérimental dans lequel les essais de roulage ont été réalisés. Les données expérimentales mesurées sont confrontées à diverses simulations numériques, donnant lieu au recalage de plusieurs paramètres caractéristiques de la modélisation. Enfin, afin de valider ce jeu de paramètres et montrer sa généralité pour toutes les opérations de roulage, on l'applique à la simulation d'un symbole différent, i.e. une autre forme de bague extérieure. La figure 2-2 présente un lopin initial, avant sa mise en forme, ainsi que sa forme finale, une fois le processus de roulage terminé.



figure 2-2: lopin initial (gauche) et bague finale (droite)

2.1. Présentation du procédé

2.1.1. Avant propos : le procédé de laminage circulaire

Le laminage circulaire permet de produire des anneaux à partir d'une ébauche forgée. Le schéma de gauche de la figure 2-3 offre une vision simplifiée du procédé. Un rouleau piloté en rotation lamine la matière contre un autre rouleau libre en rotation et piloté en translation. Des rouleaux latéraux, non représentés sur ce schéma, assurent le centrage de l'anneau et permettent le contrôle de l'écoulement de matière suivant la hauteur de l'anneau. Le procédé se passe généralement à froid, pour un meilleur contrôle géométrique et de l'état de surface.

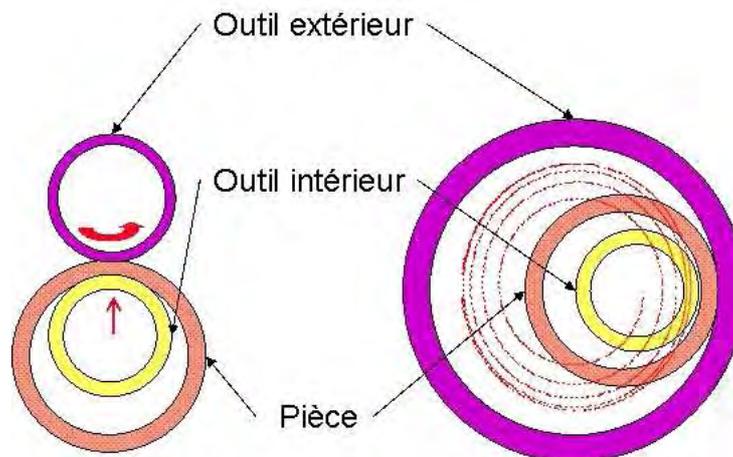


figure 2-3: Schéma des procédés de laminage circulaire (gauche) et de roulage (droite)

2.1.2. Principe de base du roulage

Le procédé de roulage (schéma de droite de la figure 2-3) permet d'éviter la fabrication d'ébauches forgées. L'anneau de départ est directement obtenu par cisailage et chanfreinage d'un tube. Il est ensuite chauffé par induction à environ 700°C. Le principe du roulage consiste en une expansion de l'anneau dans une matrice fixe par un cylindre cannelé. Cette molette intérieure, libre en rotation autour de son axe, forme le détail des gorges du roulement et plaque le tube contre les parois de la matrice extérieure, suivant un mouvement en spirale. L'écoulement de matière est d'avantage contraint qu'en laminage circulaire. Il permet ainsi d'obtenir, à mi-chaud et donc pour des efforts moindres, une précision géométrique satisfaisante, habituellement obtenue seulement à froid. Les cinétiques des outillages ainsi que les formes des outils extérieurs sont différentes entre le laminage et le roulage, mais du point de vue du matériau, les sollicitations subies sont de même nature. Toutefois, l'écoulement de roulage est beaucoup plus contraint avec peu de surfaces libres. La figure 2-4 présente une machine de roulage dans son ensemble. Elle permet de se rendre compte de l'encombrement de la machine, ainsi que de son caractère « masqué » : rien n'est visible durant le procédé de mise en forme, le lopin restant confiné à l'intérieur de la machine.



figure 2-4 : vue latérale d'une machine de roulage

2.1.3. Pilotage de la molette

Le mouvement de la molette intérieure est imposé par un ensemble complexe composé d'une poulie Redex commandée par une crémaillère, de différents fourreaux servant à l'excentration et d'un moteur hydraulique permettant de donner une vitesse de rotation initiale à la molette. Le pilotage de la molette est ainsi décrit par un mouvement combiné de rotation autour de l'axe de la matrice fixe suivant une trajectoire en spirale et d'une rotation libre sur son axe. La figure 2-5 montre une trajectoire de molette à partir du moment où à lieu l'accostage de la matière. On définit le plan (y, z) comme étant le plan dans lequel la molette décrit la spirale. L'axe des x correspond ainsi à l'axe suivant lequel est monté la molette.

La crémaillère de la poulie Redex est actionnée par un vérin hydraulique non asservi. Sa position initiale est le centre de la matrice et sa position finale est déterminée par une butée mécanique définie pour obtenir l'épaisseur visée de la bague. Du fait du caractère hydraulique, la durée du roulage est dépendante de la puissance absorbée (puissance dissipée par déformation et par frottement). Cette durée, ainsi que la trajectoire de la molette, peut donc varier d'un type de bague à l'autre et n'est pas connue a priori.

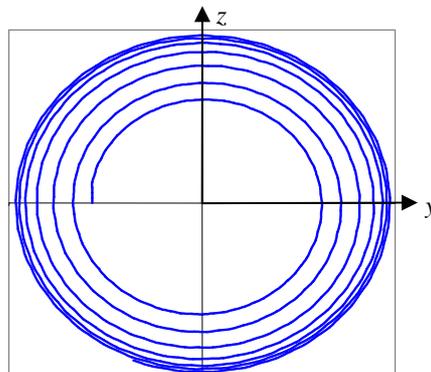


figure 2-5: trajectoire de la molette

2.1.4. Outillage et pièce

2.1.4.1. Molette

La molette est l'outil qui imprime la gravure interne à la pièce. Elle est composée d'une partie encastrée dans le nez de la machine et de la partie qui va mettre en forme l'anneau (figure 2-6). La molette est réalisée en acier HS 6-5-2. Sa température de peau en régime thermique établi se situe entre 200°C et 300°C. Elle est libre en rotation sur elle-même. Cependant pour faciliter l'accostage de la molette dans la matière, un moteur hydraulique lui transmet une vitesse de rotation initiale de l'ordre de 90 tr/min dans le sens opposé à celui du mouvement.

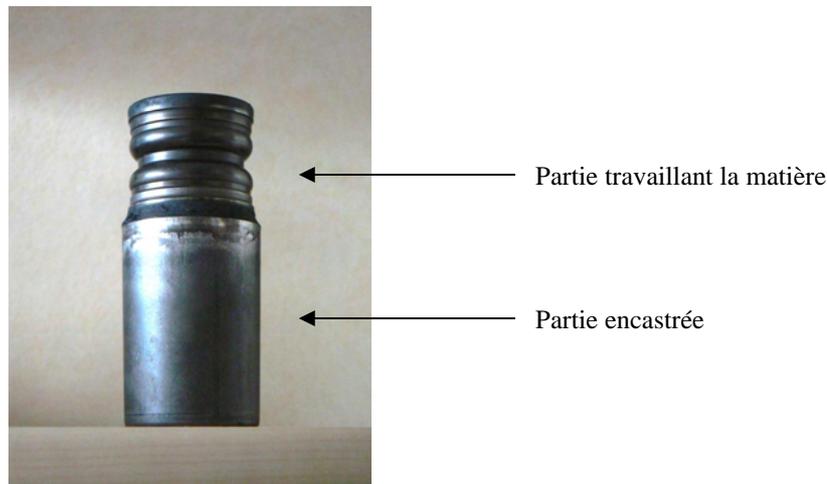


figure 2-6: vue de côté de la molette

2.1.4.2. Matrice

La matrice est l'outil externe du procédé de roulage. Elle est composée de deux parties, l'une fixe et l'autre mobile, ce qui permet d'extraire la bague une fois le cycle de mise en forme terminé. Cependant au cours d'un cycle, la matrice peut être considérée comme une pièce unique et fixe. Elle est également réalisée en acier HS 6-5-2. L'aspect extérieur de la bague finale étant lisse, il n'y a pas de forme particulière à usiner sur l'intérieur de la matrice. Il suffit simplement de bien dimensionner le diamètre intérieur de cette matrice pour qu'il corresponde au diamètre extérieur souhaité de la bague.

2.1.4.3. Ebauche

Le procédé de roulage permet l'utilisation de lopins initiaux simples obtenus par cisailage et chanfreinage d'un tube d'acier rapide 100Cr6. Une telle ébauche est visible sur la gauche de la figure 2-2.

2.1.5. Description d'un cycle

Un cycle débute lorsque la molette s'écarte de son emplacement initial, au centre de gravité de la matrice, que l'on définit comme étant le centre du repère. Il se termine lorsque la molette revient dans sa position initiale. On visualise ce cycle en traçant l'évolution de la course crémaillère au cours du temps, qui est directement liée à l'excentration de la molette par rapport au centre du repère. On décompose un cycle en plusieurs étapes présentées sur la figure 2-7 et définies par rapport à l'avancée de la course crémaillère dc , où dc_{max} représente la valeur de la course correspondant à l'atteinte de la butée mécanique.

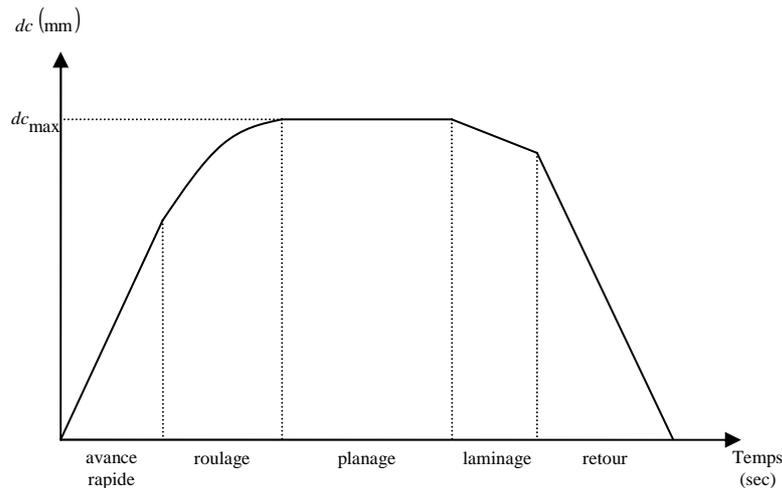


figure 2-7: courbe de déplacement de la crémaillère décrivant un cycle

Détaillons ici les différentes phases d'un cycle :

- L'avance rapide

La molette part de sa position initiale pour arriver en contact avec la matière. Elle ne subit aucune contrainte puisqu'il n'y a pas de contact. La course de la crémaillère est linéaire.

- Le roulage

Cette phase correspond à la trajectoire en spirale visible sur la figure 2-5. La molette s'excentre jusqu'à atteindre la butée mécanique de fin de course. On constate que le déplacement de la crémaillère est linéaire au début du roulage. Une fois le tonnage maximum atteint la vitesse d'excentration diminue, jusqu'à l'atteinte de la butée de fin de course. Cette phase nécessite la plus grande attention, notamment au niveau de la modélisation de la cinématique de la molette, ce qui sera développé dans la suite du chapitre. Le nombre de tours moyen décrit par la molette durant cette phase est généralement compris entre cinq et dix.

- Le planage

Durant cette étape, l'excentration de la crémaillère est constante et égale à l'excentration maximale. La molette décrit ainsi un cercle autour du centre du repère. L'effort appliqué sur la crémaillère reste donc constant. Cette phase permet d'aplanir la bague et d'imprimer correctement le dessin de la molette sur l'intérieur de la bague tout en s'assurant qu'il soit aux

bonnes cotes. La modélisation de la cinématique de la molette ne présente ici aucune difficulté particulière. Le nombre de tours moyen décrit durant cette phase est également de l'ordre de cinq à dix.

- Le laminage

Le laminage est une phase de retour élastique correspondant au relâchement de l'effort sur la crémaillère. La molette recule simplement vers le centre du repère en cessant le contact avec la matière. A ce stade, l'ébauche a atteint sa configuration finale et est devenue une bague.

- Le retour

Cette étape correspond au retour de la molette au centre du repère. Une fois cette position atteinte, les deux parties de la matrice se désolidarisent afin de laisser sortir la bague roulée.

Pour obtenir un temps de cycle global, il faut bien évidemment ajouter à ces étapes le temps d'insertion de l'ébauche ainsi que le temps d'extraction de la bague. Il faut également prendre en compte la lubrification des outils, qui a lieu après l'extraction de la bague roulée lorsque la machine est en position ouverte, et avant l'insertion d'un nouveau lopin. Ce temps de cycle global peut aller de 12 à 18 secondes en fonction du type et de la dimension des bagues roulées.

2.1.6. Motivations de la simulation

Les multiples objectifs de l'industriel sont d'augmenter ses capacités de roulage, d'améliorer ses techniques de conceptions et de réduire ses coûts de fabrication par des techniques de conception avancées.

L'augmentation des capacités passe par un meilleur contrôle du procédé et donc de la qualité des pièces produites, ainsi que l'extension du procédé à des roulements de plus gros diamètres. Ces roulements sont actuellement réalisés par décolletage, qui est un procédé plus coûteux et aux cadences de fabrication inférieures.

L'amélioration des techniques de conception a pour objectifs de répondre à de nouvelles exigences du marché, en terme de nouveaux produits et de qualité, et de réduire les coûts de fabrication ainsi que les temps de conception et de mise au point. Les clients de l'industrie automobile ont des exigences de plus en plus poussées sur la qualité des pièces et même sur leur aspect visuel. Un léger défaut localisé dans une zone non fonctionnelle et sans conséquence sur la qualité du roulement mais visible n'est plus accepté. Il faut viser le zéro défaut absolu. Cela nécessite une meilleure maîtrise de l'écoulement du métal pendant le roulage, et de prendre en compte d'éventuels défauts issus de la fabrication des ébauches.

La motivation majeure de l'emploi de la simulation numérique est d'essayer de mieux comprendre pourquoi certaines molettes cassent après assez peu de cycles et également d'essayer de prédire leur durée de vie.

Les coûts de fabrication par roulage comprennent le temps de conception et de mise au point d'une nouvelle gamme de fabrication, la durée du cycle de roulage proprement dit et la

durée de vie des outillages. Réduire le temps de conception permet d'améliorer la réactivité du bureau recherche et développement, ce qui est recherché. Toutefois, des gains plus importants peuvent être directement obtenus en réduisant la durée et le nombre d'essais de la phase de mise au point. De même, réduire le temps de cycle de roulage permettrait un accroissement de productivité, ainsi que de limiter les casses d'outils qui entraînent des arrêts de production conséquents. Cette casse est également coûteuse en outillage, les molettes ayant un coût important à l'échelle du procédé.

2.2. Cinématique de la molette

2.2.1. Modélisation d'un outil flottant en rotation et en excentration

Comme on a pu le voir précédemment, la molette est libre en rotation sur elle-même tout en décrivant une spirale au cours de la phase de roulage. Lorsque le tonnage maximal de la presse est atteint, son excentration n'est plus déterminée a priori et devient une inconnue du problème. Cette cinématique est impossible à décrire avec le modèle des outils flottants disponible dans Forge2005® et décrit dans la partie 1.4. Une nouvelle formulation d'outils flottants permettant de modéliser le roulage est donc développée.

Soit M un point quelconque de la surface de la molette (figure 2-8). Sa vitesse est définie par v_1 la vitesse d'excentration de la molette, ω_1 la vitesse de rotation primaire de la molette et ω_2 la vitesse de rotation de la molette sur elle-même. Soit O le centre du plan (y, z) . La vitesse de rotation ω_1 autour de l'axe (O, x) imprime une trajectoire circulaire à la molette et la vitesse d'excentration radiale v_1 permet d'obtenir une spirale. La vitesse de rotation ω_2 , toujours autour de l'axe (O'', x) , traduit la rotation de la molette sur elle-même.

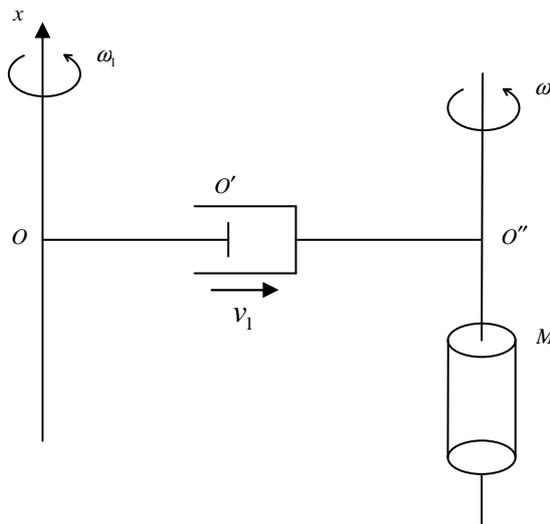


figure 2-8: modélisation de la cinématique de la molette

La vitesse de la molette en un point quelconque M s'écrit :

$$v_{outil}(M) = \omega_1 e_1 \wedge OO' + v_1 e_r + \omega_2 e_1 \wedge O''M \quad (2-1)$$

avec :

$$e_r = \frac{OO'}{\|OO'\|} \quad (2-2)$$

La valeur de v_1 varie au cours du temps. Lors du roulage et tant que le tonnage maximal de la machine n'est pas atteint, sa valeur est constante et égale à une consigne. Mais dès lors que la charge maximale est atteinte, v_1 diminue et n'est plus connue a priori, cette grandeur devient un degré de liberté supplémentaire du système. La valeur de ω_1 est connue et fixe, égale à 327 tr/min. Le caractère flottant en rotation de la molette sur elle-même impose

également l'ajout d'un degré de liberté supplémentaire qui est ω_2 . Il convient donc d'ajouter deux degrés de liberté supplémentaires au système que l'on cherche à résoudre.

2.2.2. Formalisation du couplage pièce / outil flottant

Comme on a pu déjà le faire dans la partie 1.4.1., on applique le principe des puissances virtuelles à l'outil (1-91) avec cette fois ci la nouvelle expression de sa vitesse (2-1). Il vient :

$$\forall v_{outil}^* \in \partial\Omega_{outil} \quad - \int_{\partial\Omega_f} \tau \cdot v_{outil}^* ds - \left(\frac{\partial\phi_{contact}}{\partial v}, v_{outil}^* \right) = F \cdot v_1^* e_r \quad (2-3)$$

Puisque

$$v_{outil}^*(M) = \omega_1 e_1 \wedge OO' + v_1^* e_r + \omega_2^* e_1 \wedge O''M \quad (2-4)$$

On obtient

$$\begin{cases} - \int_{\partial\Omega_{fh}} \tau \cdot e_r ds - \frac{\partial\phi_{contact}}{\partial v_1} = F \cdot e_r \\ - \int_{\partial\Omega_{fh}} \tau \cdot (e_1 \wedge O''M) ds - \frac{\partial\phi_{contact}}{\partial \omega_2} = 0 \end{cases} \quad (2-5)$$

avec :

$$\begin{aligned} \frac{\partial\phi_{contact}}{\partial v_1} &= -\rho \sum_{k \in \partial\Omega_{ch}} S_k [h_k(v_h, v_{outil})]^+ e_r \cdot n_k \\ \frac{\partial\phi_{contact}}{\partial \omega_2} &= -\rho \sum_{k \in \partial\Omega_{ch}} S_k [h_k(v_h, v_{outil})]^+ (e_1 \wedge O''M) \cdot n_k \end{aligned} \quad (2-6)$$

$\Gamma = 0$ traduit le caractère libre en rotation de la molette sur elle-même ($\Gamma \neq 0$ équivaut à imposer un couple).

La formulation discrète finale du problème des outils flottants en roulage, si l'on considère une loi de frottement de type Norton, prend ainsi la forme suivante :

$$\begin{cases} - \int_{\partial\Omega_{fh}} \alpha_f K \|\Delta v_{g_h}\|^{m-1} (\Delta v_{g_h} \cdot e_r) dS + \rho \sum_{k \in \partial\Omega_h} 1_{\partial\Omega_{ch}}(k) S_k \left((v_h - v_{outil}) \cdot n_k - \frac{\delta_k}{\Delta t} \right) e_r \cdot n_k = F \\ - \int_{\partial\Omega_{fh}} \alpha_f K \|\Delta v_{g_h}\|^{m-1} (\Delta v_{g_h} \cdot (e_1 \wedge O''M)) dS \\ + \rho \sum_{k \in \partial\Omega_h} 1_{\partial\Omega_{ch}}(k) S_k \left((v_h - v_{outil}) \cdot n_k - \frac{\delta_k}{\Delta t} \right) (e_1 \wedge O''M) \cdot n_k = 0 \end{cases} \quad (2-7)$$

On résume le système (2-7) à :

$$\begin{cases} R_{v_1} = 0 \\ R_{\omega_2} = 0 \end{cases} \quad (2-8)$$

2.2.3. Système à résoudre

Le système à résoudre est identique au système (1-95), à la différence près que la définition de la vitesse de l'outil n'est pas la même. Si on décompose la vitesse de la molette en les vitesses d'excentration et de rotation sur elle-même, le système à résoudre s'écrit :

$$\begin{pmatrix} K^{vv} & K^{vp} & K^{vv_1} & K^{v\omega_2} \\ K^{vp^T} & -K^{pp} & 0 & 0 \\ K^{vv_1^T} & 0 & K^{v_1v_1} & K^{v_1\omega_2} \\ K^{v\omega_2^T} & 0 & K^{v_1\omega_2^T} & K^{\omega_2\omega_2} \end{pmatrix} \begin{pmatrix} \Delta v \\ \Delta p \\ \Delta v_1 \\ \Delta \omega_2 \end{pmatrix} = - \begin{pmatrix} R_v \\ R_p \\ R_{v_1} \\ R_{\omega_2} \end{pmatrix} \quad (2-9)$$

avec :

$$\begin{aligned} K^{vv_1} &= \frac{\partial R_v}{\partial v_1}, & K^{v_1v_1} &= \frac{\partial R_{v_1}}{\partial v_1}, & K^{\omega_2\omega_2} &= \frac{\partial R_{\omega_2}}{\partial \omega_2} \\ K^{v\omega_2} &= \frac{\partial R_v}{\partial \omega_2}, & K^{v_1\omega_2} &= \frac{\partial R_{v_1}}{\partial \omega_2} \end{aligned}, \quad (2-10)$$

On explicite une nouvelle fois les contributions de la matrice et du second membre du système en fonction de leur nature (termes de contact ou termes de frottement). Ces contributions s'ajoutent indépendamment les unes aux autres.

2.2.4. Contributions liées au contact

Avec cette nouvelle cinématique des outils flottants, les termes de résidu liés au contact s'écrivent :

$$\begin{aligned} (R_v)_{ik}^{contact} &= \rho 1_{\partial\Omega_{ch}}(k) S_k \begin{pmatrix} v_h \cdot n_k - \omega_1 (x \wedge OO') \cdot n_k - v_1 e_r \cdot n_k \\ -\omega_2 (e_1 \wedge O''M) \cdot n_k - \frac{\delta_k}{\Delta t} \end{pmatrix} n_k^i N_k \\ (R_{v_1})^{contact} &= -\rho \sum_{k \in \partial\Omega_h} 1_{\partial\Omega_{ch}}(k) S_k \begin{pmatrix} v_h \cdot n_k - \omega_1 (e_1 \wedge OO') \cdot n_k - v_1 e_r \cdot n_k \\ -\omega_2 (e_1 \wedge O''M) \cdot n_k - \frac{\delta_k}{\Delta t} \end{pmatrix} e_r \cdot n_k \\ (R_{\omega_2})^{contact} &= -\rho \sum_{k \in \partial\Omega_h} 1_{\partial\Omega_{ch}}(k) S_k \begin{pmatrix} v_h \cdot n_k - \omega_1 (e_1 \wedge OO') \cdot n_k - v_1 e_r \cdot n_k \\ -\omega_2 (e_1 \wedge O''M) \cdot n_k - \frac{\delta_k}{\Delta t} \end{pmatrix} (e_1 \wedge O''M) \cdot n_k \end{aligned} \quad (2-11)$$

En dérivant ces termes de résidu, on obtient les contributions à la matrice hessienne liées au contact :

$$\begin{aligned}
 (K^{vv_1})_{ik}^{contact} &= \frac{\partial}{\partial v_1} (R_v)_{ik}^{contact} = -\rho 1_{\partial\Omega_{ch}}(k) S_k n_k^i N_k e_r \cdot n_k \\
 (K^{v_1v_1})^{contact} &= \frac{\partial}{\partial v_1} (R_{v_1})^{contact} = \rho \sum_{k \in \partial\Omega_h} 1_{\partial\Omega_{ch}}(k) S_k (e_r \cdot n_k)^2 \\
 (K^{v\omega_2})_{ik}^{contact} &= \frac{\partial}{\partial \omega_2} (R_v)_{ik}^{contact} = -\rho 1_{\partial\Omega_{ch}}(k) S_k n_k^i N_k (e_1 \wedge O^*M) \cdot n_k \\
 (K^{v_1\omega_2})^{contact} &= \frac{\partial}{\partial \omega_2} (R_{v_1})^{contact} = \rho \sum_{k \in \partial\Omega_h} 1_{\partial\Omega_{ch}}(k) S_k ((e_1 \wedge O^*M) \cdot n_k) e_r \cdot n_k \\
 (K^{\omega_2\omega_2})^{contact} &= \frac{\partial}{\partial \omega_2} (R_{\omega_2})^{contact} = \rho \sum_{k \in \partial\Omega_h} 1_{\partial\Omega_{ch}}(k) S_k ((e_1 \wedge O^*M) \cdot n_k)^2
 \end{aligned} \tag{2-12}$$

2.2.5. Contributions liées au frottement

Les termes de résidu liés au frottement s'écrivent :

$$\begin{aligned}
 (R_v)_{ik}^{frott} &= \int_{\partial\Omega_{fh}} \alpha_f K \|\Delta v_{gh}\|^{m-1} \Delta v_{gh}^i N_k dS \\
 (R_{v_1})^{frott} &= - \int_{\partial\Omega_{fh}} \alpha_f K \|\Delta v_{gh}\|^{m-1} (\Delta v_{gh} \cdot e_r) dS \\
 (R_{\omega_2})^{frott} &= - \int_{\partial\Omega_{fh}} \alpha_f K \|\Delta v_{gh}\|^{m-1} (O^*M \wedge \Delta v_{gh})_1 dS
 \end{aligned} \tag{2-13}$$

Explicitons comme précédemment les valeurs des dérivées de la norme de la vitesse de glissement à la puissance $m-1$ et d'expressions faisant intervenir cette vitesse de glissement par rapport aux vitesses v_1 et ω_2 de l'outil :

$$\begin{aligned}
 \frac{\partial}{\partial v_1} \|\Delta v_{gh}\|^{m-1} &= -(m-1) \|\Delta v_{gh}\|^{m-3} \Delta v_{gh} \cdot e_r \\
 \frac{\partial}{\partial v_1} (\Delta v_{gh} \cdot e_r) &= -\|e_r\|^2 - \|e_r \cdot n\|^2 \\
 \frac{\partial}{\partial \omega_2} (\Delta v_{gh} \cdot e_r) &= -(e_1 \wedge O^*M) \cdot e_r - (n \cdot e_r) (e_1 \wedge O^*M) \cdot n \\
 \frac{\partial}{\partial \omega_2} \|\Delta v_{gh}\|^{m-1} &= (m-1) \|\Delta v_{gh}\|^{m-3} (\Delta v_{gh} \wedge O^*M)_1 \\
 \frac{\partial}{\partial \omega_2} (O^*M \wedge \Delta v_{gh})_1 &= \varepsilon^{1ij} O^*M_i (- (e_1 \wedge O^*M)_j + ((e_1 \wedge O^*M) \cdot n_k) n_k^j)
 \end{aligned} \tag{2-14}$$

En dérivant les termes de résidu, grâce aux expressions précédentes, on obtient les contributions à la matrice hessienne liées au frottement :

$$\begin{aligned}
 (K^{vv_1})_{ik}^{frott} &= \frac{\partial}{\partial v_1} (R_v)_{ik}^{frott} \\
 &= - \int_{\partial\Omega_{f_h}} \alpha_f K \left((m-1) \|\Delta v_{g_h}\|^{m-3} (\Delta v_{g_h} \cdot e_r) \Delta v_{g_h}^i + \|\Delta v_{g_h}\|^{m-1} (-e_r^i + (e_r \cdot n_k) n_k^i) \right) N_k dS \\
 \\
 (K^{v_1 v_1})_{ik}^{frott} &= \frac{\partial}{\partial v_1} (R_{v_1})_{ik}^{frott} \\
 &= \int_{\partial\Omega_{f_h}} \alpha_f K \left((m-1) \|\Delta v_{g_h}\|^{m-3} (\Delta v_{g_h} \cdot e_r)^2 + \|\Delta v_{g_h}\|^{m-1} (-e_r^i + (e_r \cdot n_k) n_k^i) \right) dS \\
 \\
 (K^{v\omega_2})_{ik}^{frott} &= \frac{\partial}{\partial \omega_2} (R_v)_{ik}^{frott} \\
 &= \int_{\partial\Omega_{f_h}} \alpha_f K \left((m-1) \|\Delta v_{g_h}\|^{m-3} (\Delta v_{g_h} \wedge O^*M)_i \Delta v_{g_h}^i + \|\Delta v_{g_h}\|^{m-1} (-(e_1 \wedge O^*M)_i + ((e_1 \wedge O^*M) \cdot n_k) n_k^i) \right) N_k dS \quad (2-15) \\
 \\
 (K^{v_1 \omega_2})_{ik}^{frott} &= \frac{\partial}{\partial \omega_2} (R_{v_1})_{ik}^{frott} \\
 &= \int_{\partial\Omega_{f_h}} \alpha_f K \left((m-1) \|\Delta v_{g_h}\|^{m-3} (\Delta v_{g_h} \wedge O^*M)_i (\Delta v_{g_h} \cdot e_r) + \|\Delta v_{g_h}\|^{m-1} (-(e_1 \wedge O^*M)_i + ((e_1 \wedge O^*M) \cdot n_k) n_k^i) \right) dS \\
 \\
 (K^{\omega_2 \omega_2})_{ik}^{frott} &= \frac{\partial}{\partial \omega_2} (R_{\omega_2})_{ik}^{frott} \\
 &= - \int_{\partial\Omega_{f_h}} \alpha_f K \left((m-1) \|\Delta v_{g_h}\|^{m-3} ((\Delta v_{g_h} \wedge O^*M)_i)^2 + \|\Delta v_{g_h}\|^{m-1} \varepsilon^{1ij} O^*M_i (-(e_1 \wedge O^*M)_j + ((e_1 \wedge O^*M) \cdot n_k) n_k^j) \right) dS
 \end{aligned}$$

Dorénavant, il est possible de simuler le caractère « fou » de la rotation de la molette sur elle-même, combiné ou non avec une excentration de la molette en force maximale.

Dans le paragraphe suivant, on décrira les différentes hypothèses possibles pour simuler les déplacements de la molette. Le paragraphe 2.3.3 apporte plus particulièrement la validation de ces développements par une confrontation aux relevés expérimentaux.

2.3. Instrumentation de la machine

2.3.1. Partie moteur

Les différentes parties motrices de la machine sont instrumentées de manière à enregistrer la course de la crémaillère, et de reconstruire la trajectoire de la molette. Un capteur est également placé dans le moteur qui lance initialement la molette en rotation sur elle-même. Il permet de connaître la vitesse de rotation une fois que la molette est laissée libre en rotation. L'automate de la machine permet enfin de connaître dans quelle étape du cycle on se situe.

2.3.2. Matrice

Une matrice spécifique est réalisée pour deux symboles de bagues différents. Ces matrices contiennent des thermocouples, placés à 3 mm et 20 mm de la surface suivant l'axe (O, y) et au milieu de la matrice suivant l'axe (O, x) . Ils permettent de visualiser les variations de température au cours d'un cycle. La figure 2-9 présente une matrice démontée sur sa gauche. On devine les thermocouples sur la partie inférieure gauche de la matrice, protégés par une pâte les empêchant de fondre. Le schéma sur la droite de la figure 2-9 représente leur implantation dans la matrice.

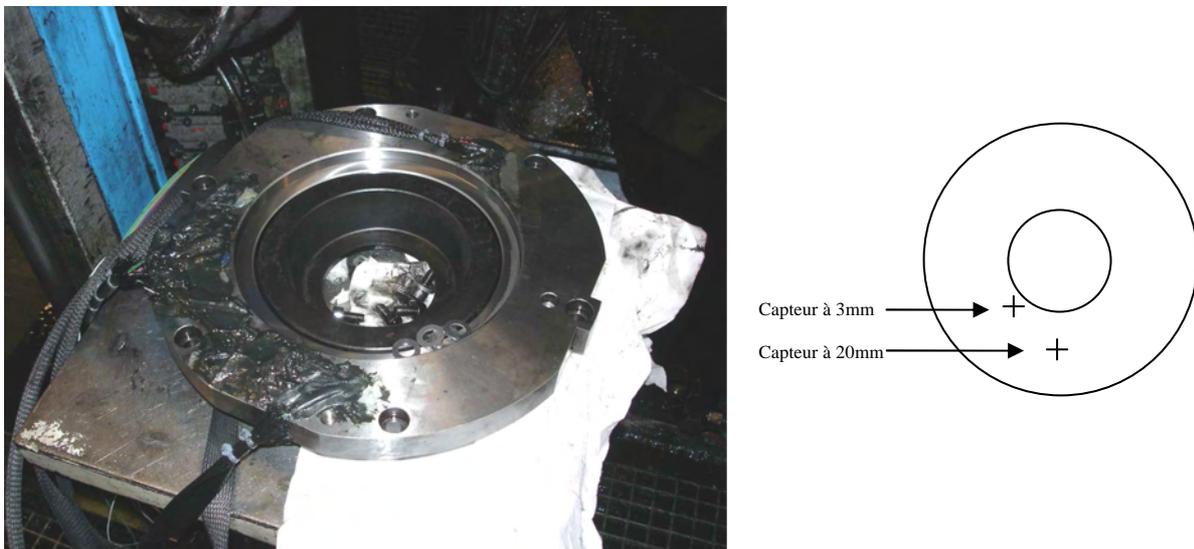


figure 2-9: matrice instrumentée et schéma d'implantation des thermocouples

2.4. Simulations du procédé de roulage sur le symbole CUV40918

2.4.1. Molette, bague finale

On représente sur la gauche de la figure 2-10 une CAO de la molette utilisée pour rouler la bague référencée CUV40918, une CAO de cette bague dans son état final étant visible sur la droite de la figure.

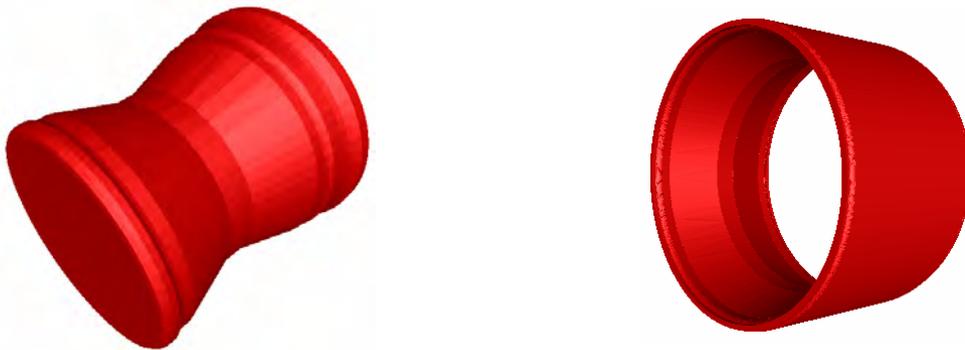


figure 2-10: molette (à gauche) et bague (à droite) du symbole CUV40918

2.4.2. Données expérimentales

Les mesures expérimentales présentées ici correspondent aux moyennes des valeurs enregistrées au cours d'une vingtaine de cycles de mise en forme, une fois le régime thermique établi. En effet, la matrice monte en température lors des premiers cycles de mise en forme et l'on attend l'atteinte d'un état thermique stationnaire.

2.4.2.1. Relevé moyen des thermocouples

La figure 2-11 montre l'évolution des températures dans la matrice à 3 mm et 20 mm de profondeur. Les traits verticaux en pointillés délimitent les différentes étapes d'un cycle :

- le roulage dure 1.24 s.
- le planage dure 2.3 s.
- le laminage associé au retour dure 3.5 s.
- l'extraction de la bague, l'insertion du lopin et l'avance rapide, dure 11 s.

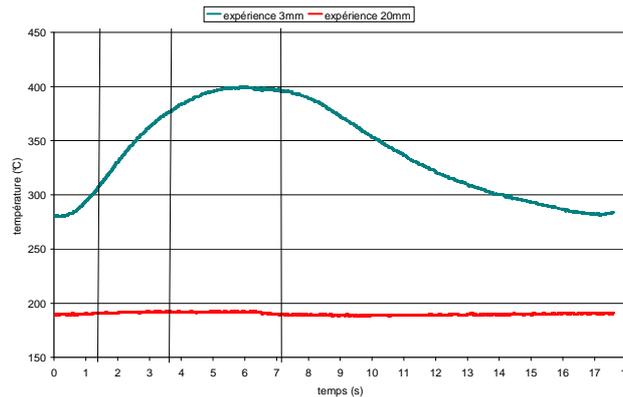


figure 2-11: évolution de la température à 3 mm et à 20 mm sur CUV40918

Le choix de débuter le cycle par le roulage est naturel puisqu'il correspond au début du processus de mise en forme, la molette rentrant en contact avec la matière à former. On peut grouper les phases de retour, d'extraction de la bague, d'insertion du lopin et d'avance rapide en une seule et même étape et simulation, puisque la molette n'est pas en contact avec la matière durant toutes ces étapes.

2.4.2.2. Vitesse de Rotation de la molette sur elle-même

Un capteur est placé dans la partie du moteur qui entraîne la molette en rotation sur elle-même à l'initialisation. Il permet de connaître cette vitesse durant la phase de roulage et de planage. Sur l'ensemble des mesures effectuées on constate le même phénomène, les valeurs relevées ne paraissent pas totalement fiables. Un exemple de relevé lors de la phase de roulage est présenté sur la figure 2-12. On constate une décroissance de la vitesse de rotation de la molette à l'accostage de la bague qui paraît étrange. De plus, les variations brusques constatées au cours du roulage n'ont aucune cause physique. Cependant, la tendance de cette courbe reste tout à fait crédible, avec un accroissement progressif de la vitesse de rotation allant de 75 tr/min après l'accostage à 120 tr/min à l'atteinte de la butée de fin de roulage. Cette rotation est exprimée en valeur absolue, cependant il est à noter que les deux vitesses de rotation ω_1 et ω_2 sont opposées. En effet, la molette tourne sur elle-même dans le sens contraire à sa rotation primaire qui lui permet de décrire un cercle dans le plan (y, z) .

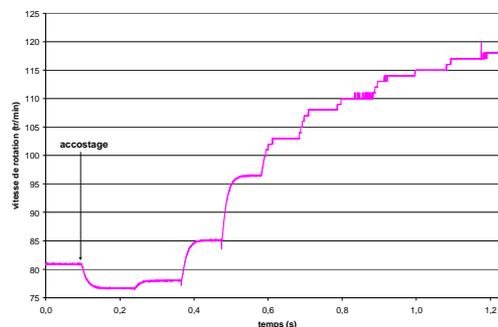


figure 2-12: évolution de la vitesse de rotation de la molette sur elle-même au cours du roulage

Afin de mieux pouvoir juger les valeurs obtenues par la simulation, on calcule la vitesse que devrait avoir la molette pour rouler parfaitement sans glisser sur la bague au cours du temps. Cette vitesse de roulement sans glissement dépend de l'excentration de la molette et

également du plus petit rayon de la molette en contact avec la matière. Elle correspond à la plus petite distance des points de la surface de la molette en contact avec de la matière à l'axe central de la molette. La distance initiale entre le centre de la molette et le point le plus proche de ce centre, en contact avec la bague est de 26,56 mm. En fin de roulage, cette distance chute à 22,64 mm. On suppose que cette distance diminue linéairement au cours de la phase de roulage. L'évolution de cette vitesse théorique est présentée sur la figure 2-13. Sa valeur à l'accostage de la molette est de 84 tr/min et sa valeur finale, lorsque l'excentration maximale est atteinte, est de 170 tr/min. En toute logique, il est impossible de constater une vitesse de rotation supérieure à cette vitesse de roulement sans glissement. Cela reviendrait à dire que la matière se déplace plus vite que l'outil à son contact. On devrait constater une vitesse simulée inférieure à cette vitesse de roulement sans glissement, traduisant la présence d'un frottement entre la molette et la matière qui induit un glissement.

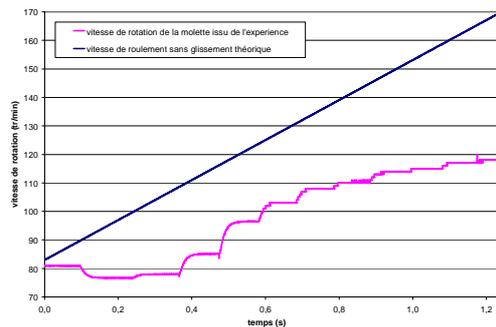


figure 2-13: évolution de la vitesse de roulement sans glissement théorique de la molette

2.4.2.3. Course de la crémaillère

Sur l'ensemble des enregistrements effectués durant les essais, l'évolution de la course de la crémaillère est quasiment toujours identique, au variation de mesure près. On choisit comme référence pour les simulations un relevé de trajectoire quelconque. C'est la trajectoire en spirale visible la figure 2-5.

2.4.3. Données numériques

2.4.3.1. Conditions initiales thermiques du lopin

Avant d'être roulé, le lopin est chauffé uniformément dans un four à induction. Une sonde thermique placée à la sortie de ce four indique que sa température moyenne en sortie est de 735°C. On sait également que la phase de transfert entre la sortie du four et le début du roulage dure 4 secondes. On calcule donc la température initiale du lopin avant roulage en effectuant la simulation du refroidissement d'un anneau à 735°C laissé durant 4 secondes à l'air ambiant. La figure 2-14 présente l'isovaleur de température sur une vue coupée de ce lopin après refroidissement. La température varie de 730°C au cœur du lopin à 719°C sur ces arêtes extérieures.

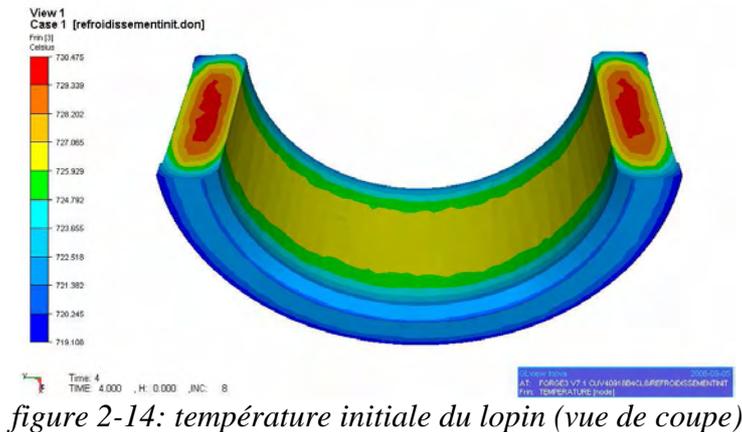


figure 2-14: température initiale du lopin (vue de coupe)

2.4.3.2. Données thermiques du lopin, des outils

Le lopin a une densité de 7596 kg/m^3 , une capacité calorifique de 800 J/kg/C , une conductivité de 25.2 W/m/C et une émissivité de 0.7 . L'effusivité de la matrice et de la molette vaut $10063 \text{ W.s}^{1/2}/(\text{m}^2.\text{C})$. Ces grandeurs ne sont pas sujettes à modification puisqu'elles sont très bien identifiées et intrinsèques aux matériaux et ne fluctuent pas en fonction des procédés de mise en forme étudiés.

2.4.3.3. Loi de comportement du lopin et loi de frottement

Le lopin est en acier 100Cr6. C'est un matériau répandu et caractérisé de manière fiable par une loi de comportement de type Norton-Hoff (équation 1-12), avec $K = 2707.108 \text{ MPa}$ et $m = 0.1529$.

La lubrification est assurée par de l'huile graphitée qui est projetée sur les outils à chaque fin de cycle, lorsque la bague est éjectée hors de la machine. Il est donc complexe de caractériser le frottement. On utilise des valeurs standard de la forge à mi chaud avec ce type de lubrification, à savoir une loi de frottement de Coulomb avec $\mu = 0.25$ saturée par une loi de Tresca avec $\bar{m} = 0.3$.

2.4.4. Première approche : Vitesse d'excentration constante et molette folle en rotation

On ne s'intéresse ici qu'à la simulation de la phase de roulage. S'agissant d'une simulation mono domaine, on définit la température de la molette comme constante et égale à 250°C et la température de la matrice constante et égale à 350°C .

2.4.4.1. Trajectoire

La vitesse d'excentration constante est la cinématique la plus simple, mais elle n'est jamais rencontrée lors des essais. En effet, les machines sont dimensionnées en fonction des efforts mis en jeu au cours de la mise en forme, le tonnage maximal est donc toujours atteint. Cette cinématique est cependant une première approche peu coûteuse. Elle permet notamment de valider les développements concernant le caractère flottant de la molette sur elle-même.

La figure 2-15 présente sur sa gauche la spirale décrite par la molette en vitesse d'excentration constante en comparaison avec une spirale rencontrée lors d'un essai expérimental. On constate que la spirale issue de la simulation est quasiment confondue avec la spirale expérimentale durant les trois premiers tours, qui correspondent à la première phase du roulage lorsque le tonnage maximal de la machine n'est pas encore atteint. Ensuite, il suffit d'un peu plus d'un tour à la spirale issue de la simulation pour atteindre l'excentration maximale tandis qu'il en faut quasiment trois de plus à la spirale expérimentale pour atteindre cette même excentration.

Sur la droite de la figure 2-15 est représentée l'évolution de l'excentration calculée et expérimentale du centre de la molette par rapport au centre du repère en fonction du temps. Ces excentrations sont quasiment confondues durant une demi seconde, ce qui correspond au trois premiers tours de la spirale. Ensuite, l'excentration issue de la simulation continue à être linéaire tandis que la vitesse d'excentration expérimentale ne cesse de diminuer, jusqu'à atteindre la butée. Le temps de roulage est ici de 0.76 s. Il est plus court que le temps de roulage rencontré lors des essais (1.24 s) puisque l'excentration est constante, la butée est atteinte plus rapidement.

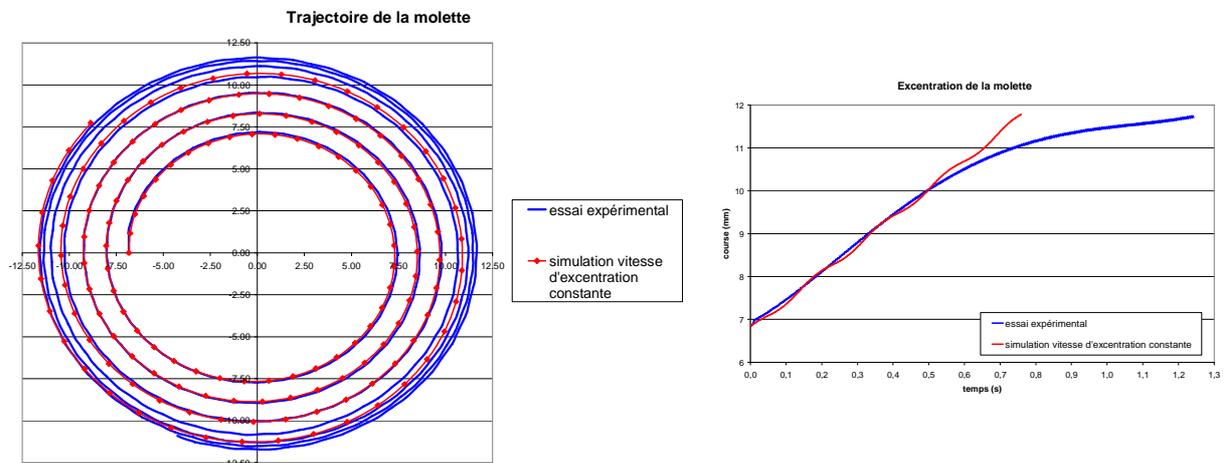
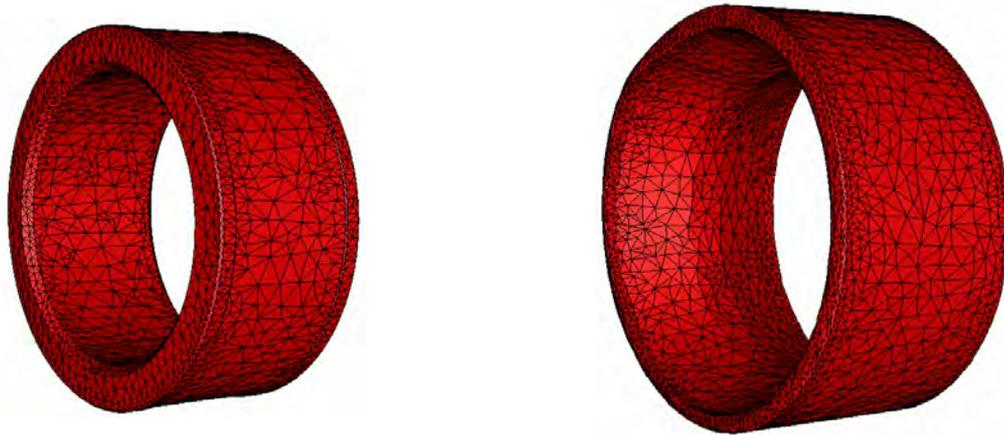


figure 2-15: comparaison de trajectoires avec une vitesse d'excentration constante

2.4.4.2. Maillage et temps de calcul

La figure 2-16 monte les maillages en début et en fin de simulation de la phase de roulage. Ils comportent respectivement 5291 et 7384 nœuds. Le temps de calcul de cette simulation de roulage est de 17 h 40 min.



Maillage initial :
5291 nœuds

Maillage en fin de roulage :
7384 noeuds

figure 2-16: maillage en début et fin de simulation avec excentration constante

2.4.4.3. Evolution du contact durant le roulage

La figure 2-17 présente l'évolution du contact au cours de la simulation. A l'instant initial, le contact est ponctuel sur les deux côtés de la face interne du lopin avec la molette ainsi que sur une partie de sa face externe avec la matrice. Après un tour, la molette s'est excentrée et pénètre plus largement à l'intérieur du lopin. Sa surface externe est également plus largement en contact avec la matrice. On constate la même évolution entre le premier et le second tour. Après trois tours, on observe visuellement un accroissement des rayons interne et externe du lopin. Le contact s'étend sur toute la section du lopin. Le contact avec la matrice s'étend sur quasiment la moitié de la surface externe du lopin. Après quatre tours, l'extérieur du lopin est totalement en contact avec la matrice, la bague a atteint son diamètre extérieur final. A l'intérieur, le contact est largement établi sur toute sa section. L'excentration maximum, signifiant la fin du roulage, est atteinte 0,06 s après la fin du quatrième tour, soit environ $1/6^{\circ}$ de tour de molette.

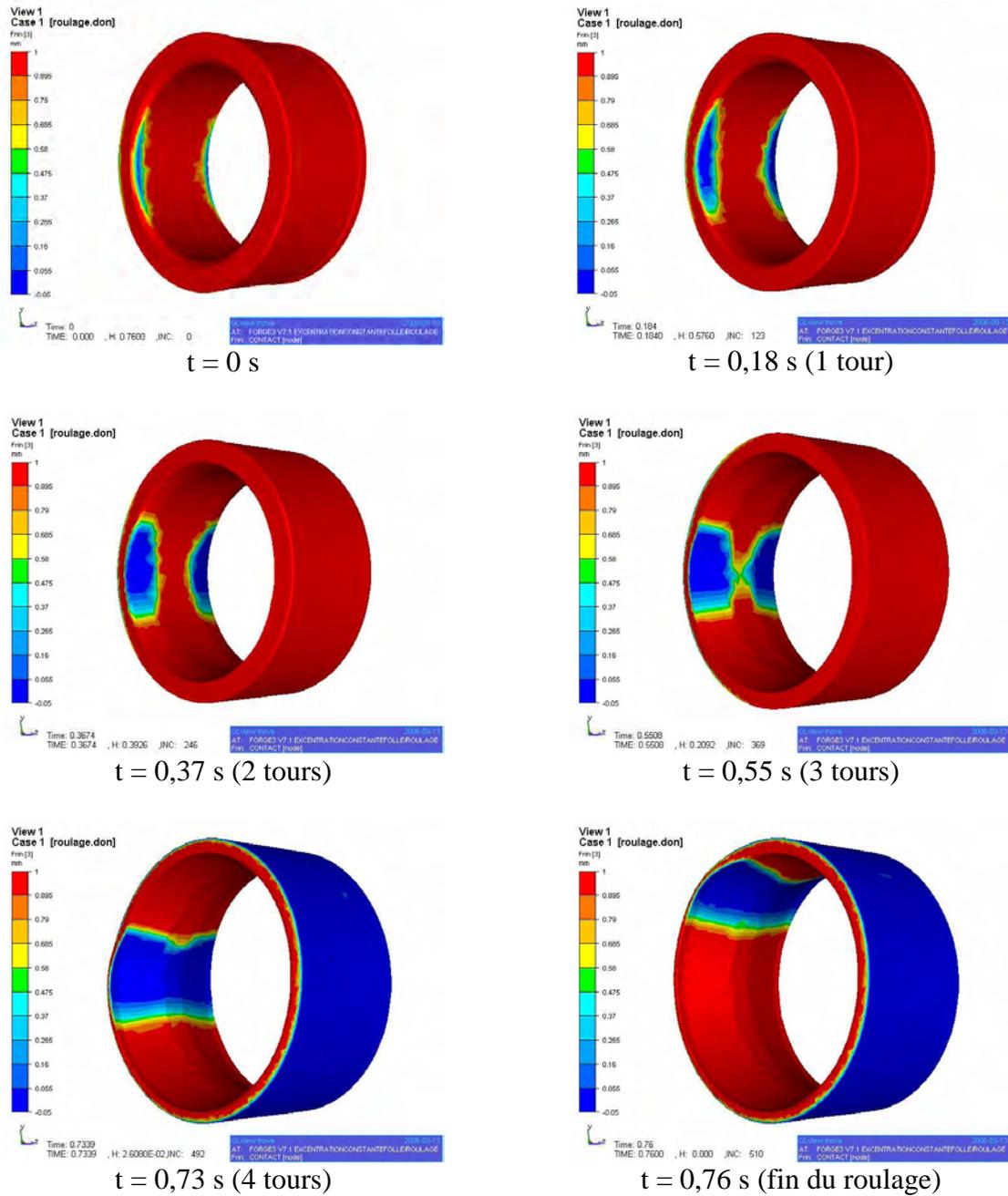


figure 2-17: évolution du contact durant la phase de roulage

2.4.4.4. Vitesse de rotation de la molette sur elle-même

On trace l'évolution de la vitesse de rotation de la molette sur elle-même. Sur la figure 2-18 elle atteint une valeur de 75 tr/min en début de roulage et croît quasiment linéairement jusqu'à $t=0,6$ s. On constate alors une augmentation brusque d'une dizaine de tr/min. On verra par la suite que cet instant correspond également à une forte augmentation de l'effort imposé sur la molette. Ce moment, situé entre le 3^e et le 4^e tour de molette, correspond à l'atteinte de l'expansion totale de la bague dans la matrice, c'est-à-dire qu'à partir de là, toute la surface extérieure de la bague est en contact avec la matrice et va le rester. Ensuite, la vitesse de rotation de la molette sur elle-même se stabilise aux alentours de 130 tr/min.

Cette hausse soudaine de la vitesse de rotation peut s'expliquer par le placage de la bague contre la matrice, ce qui entraîne une augmentation de la contrainte normale de contact et par conséquent du frottement. La vitesse se rapproche de celle de roulement sans glissement. Si l'on compare ces deux vitesses, on en déduit qu'il y a glissement puisque la vitesse calculée est inférieure à la vitesse théorique de roulement sans glissement.

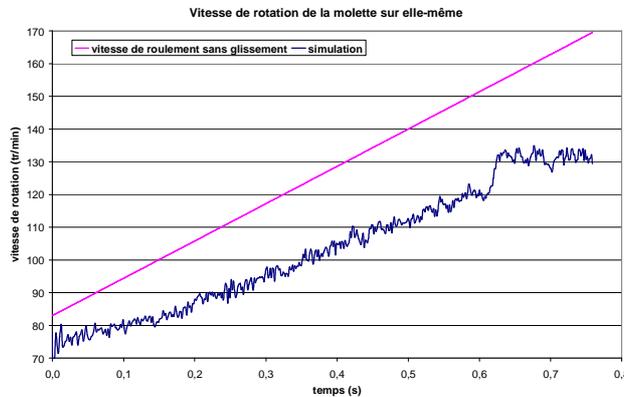


figure 2-18: vitesse de rotation de la molette comparée à la vitesse de roulement sans glissement

Sur la figure 2-19 cette vitesse est comparée avec celle obtenue lors de l'expérience. On constate tout d'abord que les temps de roulage sont très sensiblement différents, la butée est atteinte plus rapidement. Cependant, en négligeant les variations brusques des relevés expérimentaux, on constate que les pentes des deux courbes sont sensiblement les mêmes entre 0,3 et 0,7 s. En revanche, on ne retrouve pas l'augmentation brusque de la vitesse en fin de roulage sur la courbe expérimentale. Il y a un décalage inexplicable, probablement dû à l'inexactitude des relevés expérimentaux. La valeur finale de la vitesse simulée est ainsi plus importante d'une dizaine de tr/min par rapport à la vitesse expérimentale.

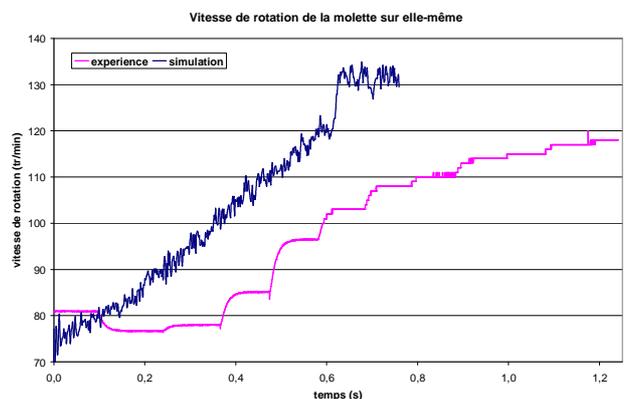


figure 2-19: vitesse de rotation de la molette comparée à l'expérience

2.4.4.5. Effort sur la molette

La figure 2-20 présente l'évolution du calcul de l'effort relevé sur la molette durant la phase de roulage. Il croît quasiment linéairement entre l'instant initial et l'instant $t=0,6$ s pour atteindre une valeur de 20 t. On retrouve cet instant particulier, qui a été déjà mis en exergue sur la figure 2-19 pour la vitesse de rotation de la molette. Cette hausse soudaine de l'effort est logique puisqu'il traduit l'expansion totale de la bague à l'intérieur de la matrice. Il n'y a

plus de surface libre, l'écoulement de la matière devient beaucoup plus contraint, la seule possibilité offerte étant un échappement radial suivant l'axe (O, x) . L'effort exercé sur la molette va dès lors croître très rapidement pour atteindre une valeur finale de 55t.

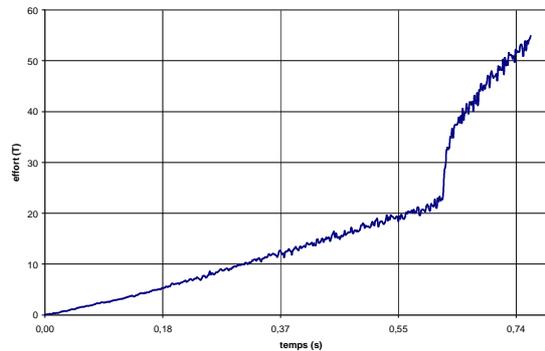


figure 2-20: effort imposé sur la molette durant le roulage

2.4.4.6. Evolution de la température durant le roulage

La figure montre l'évolution de la température du lopin durant la phase de roulage. A l'instant initial, la température varie entre 720°C et 725°C en surface. Après un tour, un léger échauffement se produit sur la partie interne du lopin en contact avec la molette. Après deux puis trois tours, ce phénomène s'accroît. A la fin de la phase de roulage, on voit clairement deux fronts chauds symétriques à l'intérieur du lopin qui suivent l'évolution du contact avec des températures atteignant quasiment 800°C. Sur le centre de la face extérieure du lopin, on constate un léger refroidissement dû au contact avec la matrice, avec des températures de l'ordre de 715°C.

2.4.4.7. Bilan de la simulation

Cette première simulation permet de valider les développements effectués sur le caractère flottant en rotation sur elle-même de la molette. Elle permet également de mieux appréhender ce procédé de roulage, notamment grâce à l'analyse de l'évolution du contact et de la température. Cependant, cette cinématique est éloignée de la réalité tant en terme de trajectoire que d'effort ou de durée. Il est nécessaire d'utiliser un autre type de cinématique. On s'intéresse ainsi dans la partie suivante à un pilotage en effort maximal.

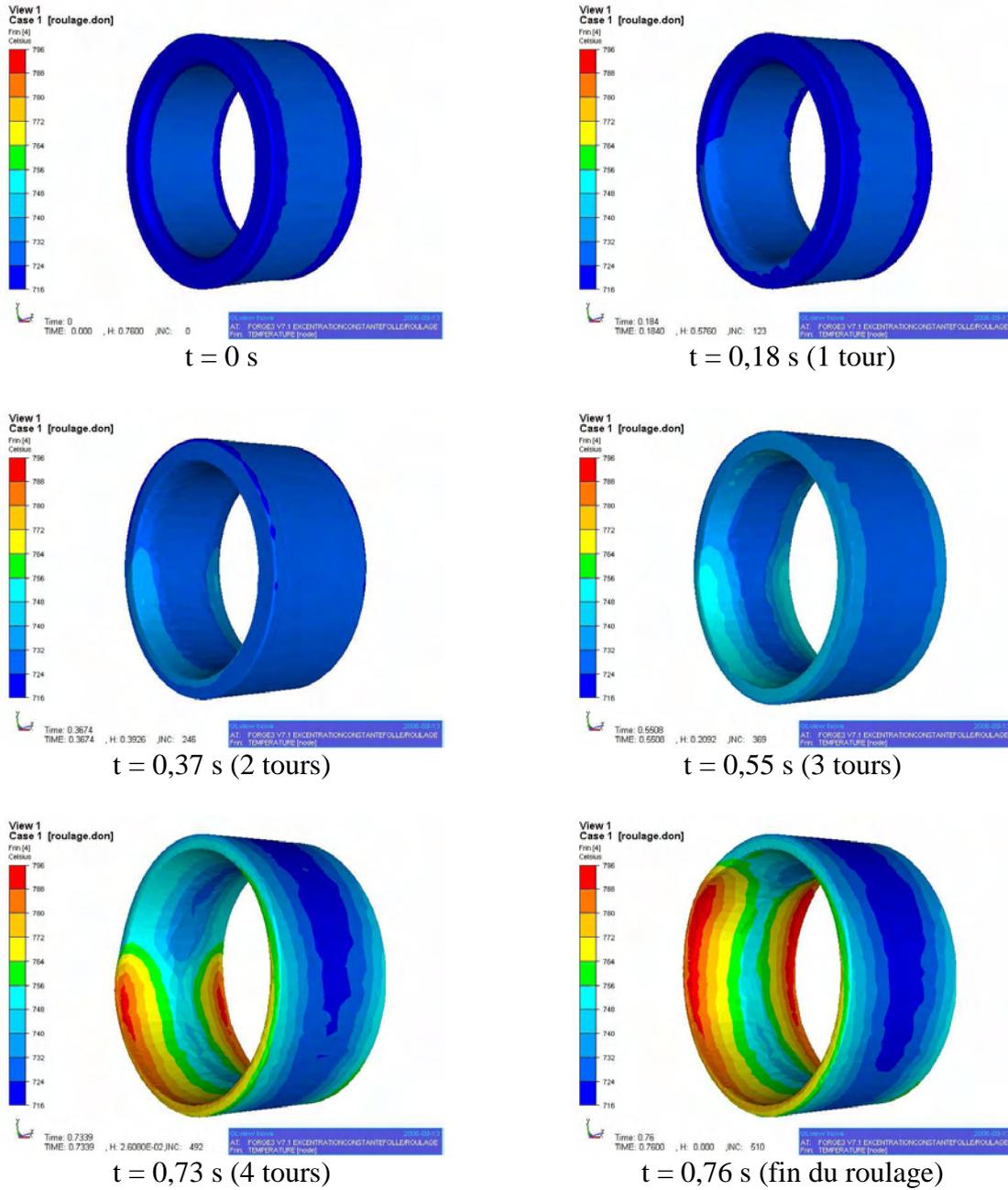


figure 2-21: évolution de la température durant la phase de roulage

2.4.5. Deuxième approche : pilotage en effort maximal et molette folle en rotation

2.4.5.1. Trajectoire

Le tonnage maximum de la presse est défini égal à 30T, d'après les valeurs expérimentales. Une fois cette limite obtenue, on ajoute un degré de liberté au système, correspondant à la vitesse d'excentration de la molette. Ce degré de liberté supplémentaire est déterminé en fonction de l'écoulement de la matière et des forces extérieures mises en jeu, par l'intermédiaire des développements effectués et explicités dans la partie 2.2. On passe dans un mode que l'on appelle « fluage ».

On observe sur la gauche de la figure 2-22 que la cinématique de la molette issue de la simulation est quasiment identique à celle expérimentale. Cette observation est confirmée par la comparaison de l'évolution des excentrations au cours du temps tracée sur la droite de la figure 2-22. On constate quelques oscillations de l'excentration au cours du temps, mais l'erreur commise est minime. On observe un décalage de la prédiction du passage en mode de fluage d'environ $1/10^{\text{e}}$ de seconde entre la simulation et les relevés expérimentaux. On constate que le temps de roulage de la simulation est légèrement plus court que le temps réel, d'un $1/10^{\text{e}}$ de seconde. Cet écart temporel est cependant tout à fait acceptable. L'excentration finale calculée est également déterminée par la valeur de butée.

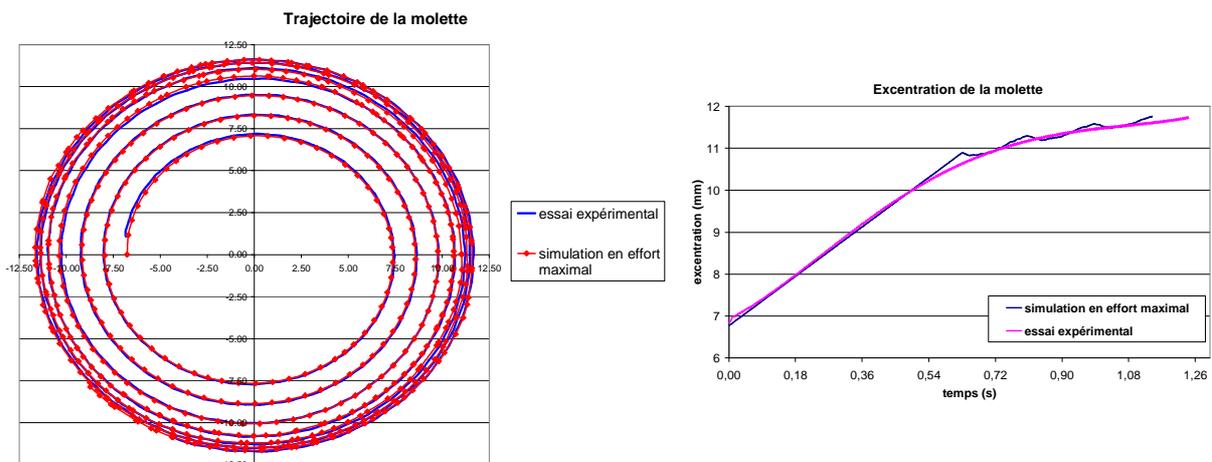


figure 2-22: comparaison de trajectoires avec une formulation d'effort maximal

C'est un premier résultat très important qui permet de valider toute la chaîne de la simulation. La similitude des courses de la crémaillère entre l'expérience et la simulation traduit l'équivalence des efforts imposés par la molette et de la résistance du matériau. On valide ainsi la modélisation de la cinématique de la molette, de même que les caractérisations du matériau et du frottement. La simulation se montre prédictive sur la trajectoire de la molette ainsi que sur le temps de roulage. C'est un résultat important pour la mise au point du procédé puisque le temps de roulage joue sur la température de la bague et donc sur ses caractéristiques, ainsi que pour ce qui concerne la productivité.

2.4.5.2. Maillage et temps de calcul

La figure 2-23 montre les maillages issus de cette simulation en début de procédé, à la fin de la phase de roulage et à l'issue de l'étape de planage. Le temps de calcul de la simulation de roulage est de 21 h 8 min tandis que la simulation de planage dure 10 h 47 min.

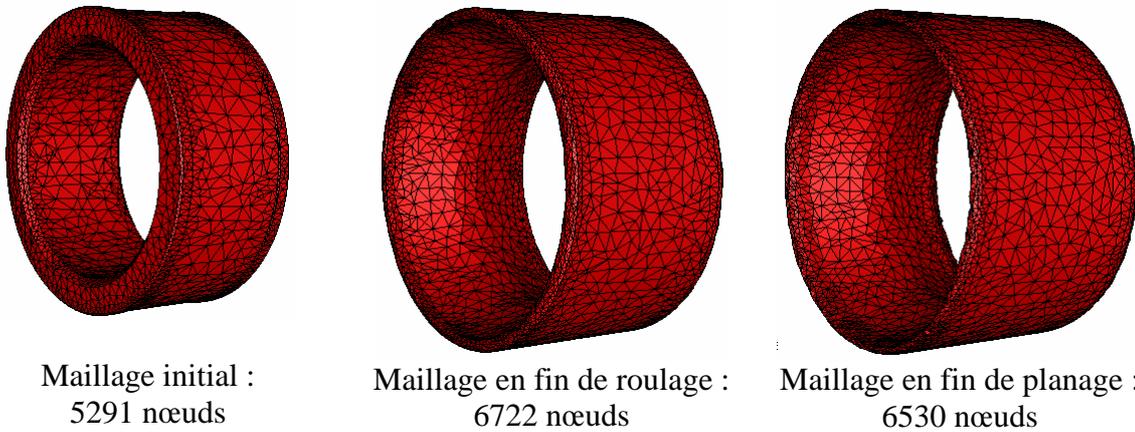


figure 2-23: maillages en début, fin de roulage et fin de planage en effort maximal

2.4.5.3. Etape de planage

Lorsque la simulation de la phase de roulage est terminée, la simulation de l'étape de planage est effectuée. Elle ne présente pas de difficulté du point de vue de la cinématique, la molette décrivant un cercle de rayon égal à l'excentration maximale. Le caractère flottant en rotation de la molette sur elle-même est conservé. La figure 2-24 présente l'évolution de cette vitesse de rotation au cours du planage, comparée au relevé d'un essai expérimental et à la vitesse de roulement sans glissement théorique. La vitesse de rotation simulée est constante au cours du planage, à quelques oscillations près, et égale à environ 145 tr/min, soit la valeur de cette même vitesse en fin de roulage. L'excentration ne variant plus, ce résultat est logique. Cette valeur est supérieure de 25 tr/min à l'expérience. Il est cependant difficile d'interpréter cet écart, la modification des paramètres de la loi de frottement de la simulation n'ayant quasiment aucune influence sur la vitesse de rotation de la molette. L'écart entre la vitesse de roulement sans glissement et la vitesse simulée est également constant.

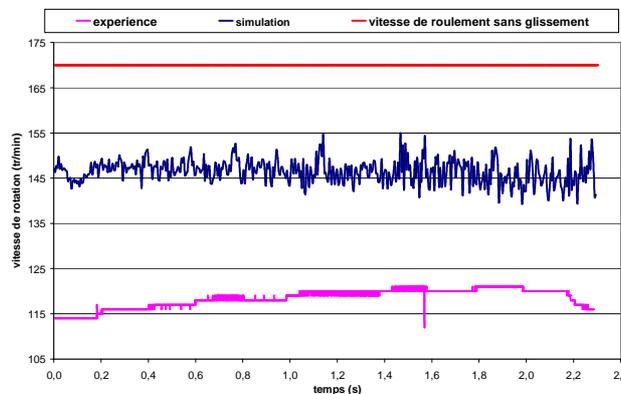


figure 2-24: vitesse de rotation de la molette sur elle-même durant le planage

2.4.5.4. Température du lopin durant le planage

Il est intéressant d'observer l'évolution de la température du lopin au cours de la phase de planage et notamment le refroidissement de sa face extérieure au contact de la matrice. La figure 2-25 montre l'évolution de cette température, avec dans la colonne du milieu une vue entière du lopin et dans la colonne de droite un plan de coupe du lopin. Après le premier tour de planage, le lopin a quasiment atteint sa configuration géométrique finale, il n'y a donc plus d'échauffement dû à la déformation de la matière. La température décroît plus rapidement sur la partie externe du lopin que sur sa partie interne, à cause du contact avec la matrice qui a une température uniforme de 350°C. Le plan de coupe permet de voir l'évolution du gradient de température dans l'épaisseur du lopin. En fin de planage, la température de la partie externe de la bague est quasiment uniforme et de l'ordre de 650°C. La partie interne de la bague a une température allant de 730°C au centre à 650°C sur les bords.

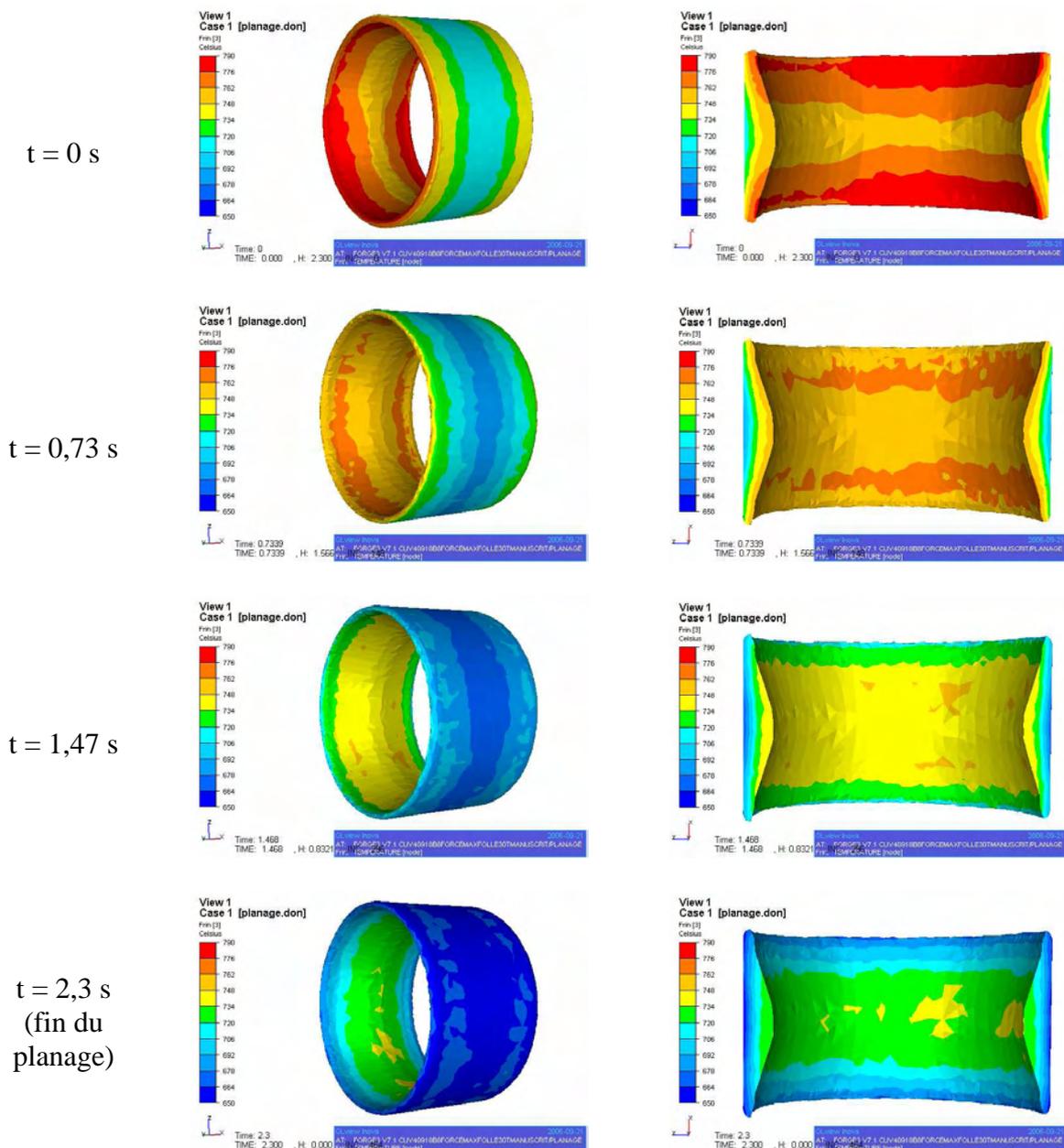


figure 2-25: évolution de la température du lopin durant le planage

2.4.5.5. Déformation équivalente en fin de planage

En observant l'iso-valeur de la déformation équivalente (figure 2-26) en fin de planage, on constate que les parties les plus déformées sont les deux extrémités de la bague, avec une déformation équivalente de l'ordre de 2,3. A contrario, la partie la moins déformée est la partie centrale de la bague, notamment sa face externe, avec une valeur de déformation équivalente de l'ordre de 0,5.

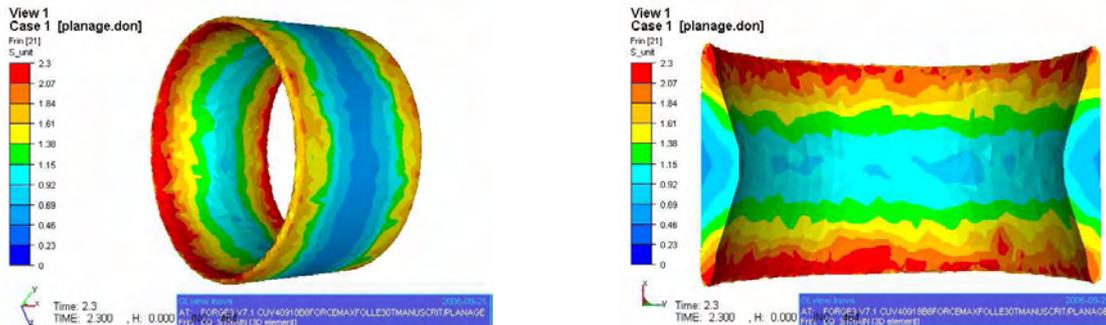


figure 2-26: déformation équivalente en fin de planage

2.4.5.6. Bilan de la simulation

Les résultats de ces simulations apportent un grand nombre d'informations jusqu'alors inaccessibles, que ce soit au niveau des efforts mis en jeux, de la vitesse de rotation de la molette sur elle-même, ou au niveau de l'évolution de la température et du mouvement de matière. Ces informations peuvent être utilisées pour la mise au point de nouvelle gamme de roulage car elles permettent de comprendre et maîtriser l'écoulement de la matière. On peut ainsi valider les remplissages de gravure pour les bagues et avoir une comparaison des efforts nécessaires pour différentes gammes.

Les développements effectués permettent dorénavant de simuler la cinématique réelle en effort maximal combiné avec le caractère flottant de la molette. La confrontation à l'expérience des résultats a mis en valeur la prédictibilité de la simulation sur la trajectoire de la molette ainsi que sur le temps de roulage, ce qui constitue une avancée très significative pour la simulation.

2.5. Etude thermique des outils pour le symbole CUV40918

On s'intéresse ici au comportement thermique des outils au cours d'un cycle. La prise en compte de la température de la matrice et de la molette, confrontée avec des relevés thermiques expérimentaux, est un autre moyen de valider les calculs en plus d'être une information importante pour l'endommagement de la molette.

2.5.1. Trajectoire point par point

Une simulation thermo-mécanique multi-corps du procédé serait trop coûteuse en temps de calcul. De plus, la déformation élastique des outils ne présente pas d'intérêt pour cette étude. On effectue seulement un calcul thermique dans les outils, en les considérant comme indéformables. Cependant, si l'on ne considère que la thermique dans les outils, il est impossible de les définir comme étant flottant. La cinématique en effort maximal est donc inutilisable ici. Il est également impossible de définir la molette folle en rotation sur elle-même. Un nouveau type de cinématique a ainsi été développé afin de rester le plus fidèle possible à la trajectoire originale de la molette lors de la phase de roulage, c'est la trajectoire point par point. Elle permet de définir exactement les différentes positions de l'outil dans l'espace au cours du temps. On peut donc simuler parfaitement la trajectoire en spirale de la molette lors de la phase de roulage. On constate sur la figure 2-28, grâce aux comparaisons des trajectoires (à gauche) et des excentrations (à droite) de la molette qu'elles sont identiques entre l'expérience et la simulation. Les données sont correctement prises en compte. Cette option permet donc de simuler le roulage d'une bague en connaissant a priori le relevé de la trajectoire de l'outil.

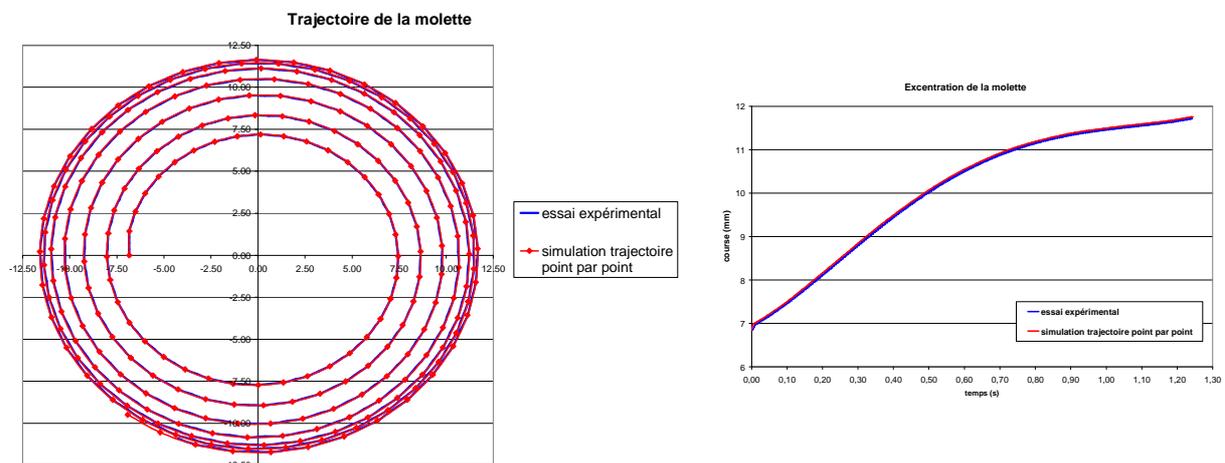


figure 2-27: comparaison de trajectoires avec une discrétisation point par point

2.5.2. Recalage thermique

Afin d'exploiter au mieux les informations disponibles sur la thermique de la matrice au cours d'un cycle (figure 2-11), on effectue un travail d'identification des paramètres

d'échange thermique ainsi qu'une optimisation des conditions thermiques initiales de la matrice.

2.5.2.1. Conditions initiales et conditions aux limites

La température initiale de la matrice est définie d'après les données expérimentales. La température à 20 mm de profondeur est quasiment constante au cours de tout le cycle et égale à 190°C. Pour refléter ces conditions, on réalise un maillage réduit de la matrice d'une épaisseur de 20 mm seulement auquel on imposera durant tout le cycle une température sur la surface extérieure de 190°C. Au commencement du roulage, la température de la matrice à 3 mm est de 280°C. On définit donc un gradient thermique non linéaire dans l'épaisseur de la matrice afin que la température à 3 mm soit de l'ordre de 280°C. On obtient ainsi la configuration thermique initiale, visible sur le plan de coupe sur la gauche de la figure 2-28.

La température initiale de la molette en surface (droite de la figure 2-28) est comprise entre 250 et 270°C. Cette configuration est obtenue en simulant entièrement 3 cycles consécutifs afin d'atteindre un pseudo régime thermique établi.

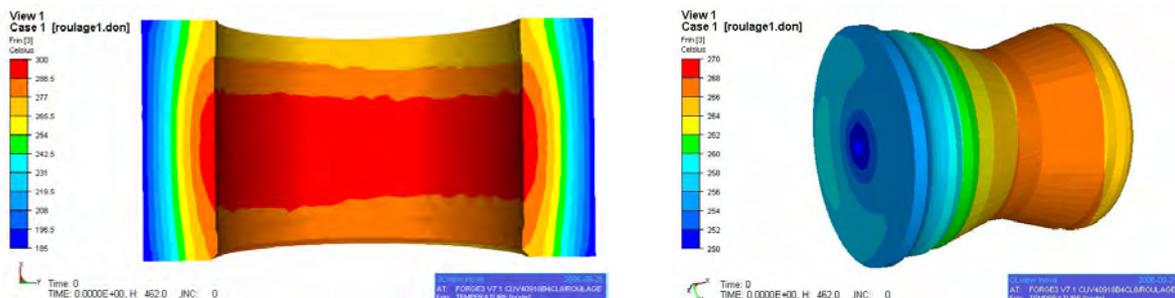


figure 2-28: température initiale de la matrice (vue de coupe) et de la molette

2.5.2.2. Coefficients d'échange thermique et température ambiante

On s'intéresse ici à l'évolution de la température dans l'épaisseur de la matrice aux profondeurs de 3 mm et 20 mm durant un cycle entier. Les premières simulations réalisées utilisent les valeurs usuelles des coefficients d'échange thermique et de température ambiante. La figure 2-29 présente la comparaison des températures relevées lors des essais avec les températures simulées avec des données standard. Les traits noirs verticaux délimitent les différentes simulations nécessaires à la modélisation d'un cycle : apparaît tout d'abord l'étape de roulage (de 0 à 1,3 s), puis l'étape de planage (de 1,3 à 3,6 s) et enfin les étapes de laminage et de retour (de 3,6 à 7,1 s). Enfin le cycle se termine par la simulation regroupant les étapes d'extraction de la bague et d'insertion du lopin (de 7,1 à 17,6 s). Malgré un respect des températures initiales à 3mm et 20mm, on observe un comportement très éloigné de la réalité. La courbe de température à 20mm croît de manière continue tout au long du cycle pour atteindre une valeur finale de l'ordre de 210°C, alors qu'elle devrait stagner autour de 190°C. Concernant l'évolution à 3mm, on observe une température constante durant le roulage, puis un échauffement durant le planage. Les pentes entre l'expérience et la simulation sont cependant très éloignées, la matrice ne chauffant que très peu d'après la simulation. On constate le même phénomène durant la phase de laminage et de retour, avec une augmentation trop restreinte de la température. A la fin de l'étape de retour (t = 7,1 s), la température simulée est de 325°C alors que l'expérience donne une température de 395°C. Durant la phase de refroidissement, la simulation ne présente toujours pas un comportement

en adéquation avec l'expérience, la perte de chaleur étant très modérée. La simulation reste cependant cohérente avec elle-même, la matrice ne pouvant pas restituer durant le refroidissement la chaleur qu'elle n'a pas emmagasinée durant les étapes précédentes.

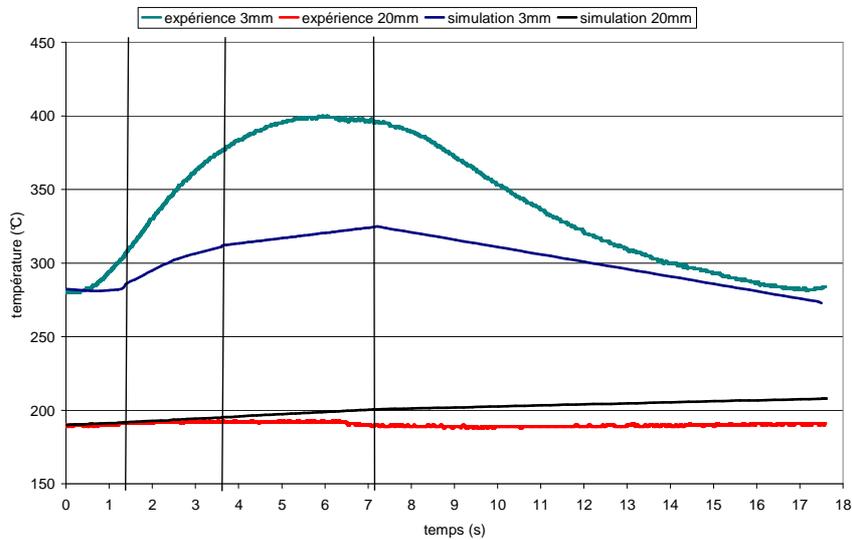


figure 2-29: comparaison de la température de la matrice durant un cycle avant recalage

On s'attache donc à adapter les coefficients d'échange thermique et la température ambiante afin de faire correspondre au mieux les courbes expérimentales à celles issues de la simulation. Après plusieurs simulations de cycles entiers en changeant progressivement les différents coefficients influents, on arrive à des résultats tout à fait convaincants, visibles sur la figure 2-30.

La température ambiante est ainsi fixée à 150°C durant les phases de roulage, planage, laminage et retour. Cette valeur de température reflète le caractère fermé et clos du procédé, sachant que le logiciel ne prend pas en compte le rayonnement thermique. Lors des phases d'extraction et d'insertion du lopin, la température de l'air est considérée égale à 50°C, puisque la machine est ouverte durant ces périodes. Le coefficient d'échange thermique avec l'air est fixé égal à 500 W/m²K et le coefficient d'échange thermique des outils fixé à 7000 W/m²K.

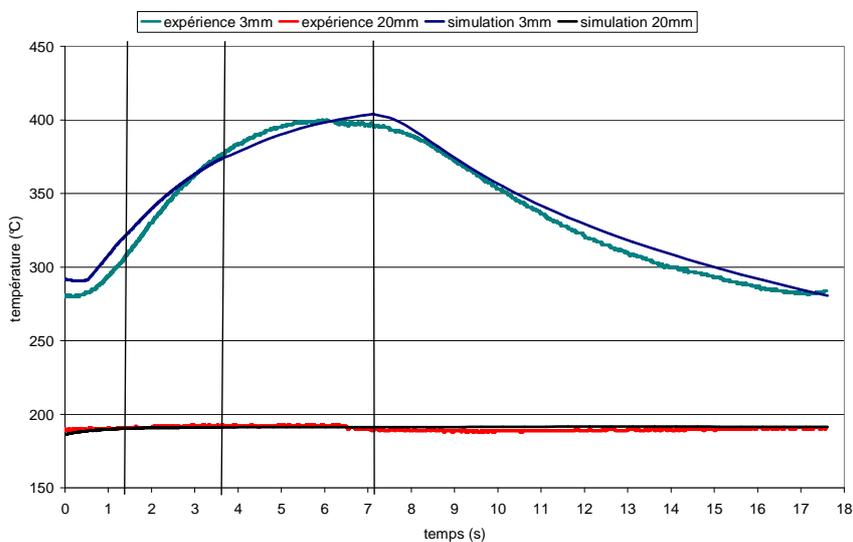


figure 2-30: comparaison de la température de la matrice durant un cycle après recalage

Concernant la température à 20 mm, on constate que les conditions aux limites imposées à la matrice lors des différentes simulations d'un cycle sont bien respectées. La température moyenne reste constante égale à 190°C, conformément à l'expérience. Concernant la température à 3 mm, les valeurs issues des simulations sont également très proches de l'expérience. On constate durant la phase de roulage une sur-évaluation d'une petite dizaine de degrés, due à une valeur initiale légèrement supérieure. La phase de planage est parfaitement reproduite, avec des pentes quasi équivalentes. Ces résultats reposent sur la bonne identification des coefficients de transfert thermique des outils. Les étapes de laminage et de retour sont également fidèlement simulées, avec des pentes similaires. Enfin, les étapes d'extraction et d'insertion sont également correctement modélisées, avec des pentes de refroidissement quasiment confondues. Cette très bonne acuité des courbes numériques valide les valeurs choisies pour les coefficients d'échange thermique et pour la température de l'air ambiant.

Le tableau 2-1 récapitule les différents paramètres numériques utilisés. Apparaissent en gras les paramètres thermiques spécifiques au roulage qui ont été identifiés grâce à la confrontation de la simulation à l'expérience. Les autres paramètres, définissant le lopin et les outils, n'ont pas été sujets à recalage, eu égard à leurs bonnes identifications ou caractérisations.

Température ambiante (roulage, planage, laminage, retour)	150°C
Température ambiante (extraction et insertion)	50°C
Coefficient d'échange thermique du lopin avec l'air	500 W/m².K
Coefficient d'échange thermique de la matrice	7000 W/m².K
Coefficient d'échange thermique de la molette	7000 W/m².K
Densité du lopin	7596 kg/m ³
capacité calorifique du lopin	800 J/kg/C
Conductivité thermique du lopin	25.2 W/m/C
Emissivité	0.7
Effusivité de la matrice	10063 W.s ^{1/2} /(m ² .C)
Effusivité de la molette	10063 W.s ^{1/2} /(m ² .C)
Comportement du lopin (Acier 100Cr6)	Loi de Norton-Hoff $K = 2707.108 \text{ MPa}$ et $m = 0.1529$
Frottement	Loi de Coulomb $\bar{m} = 0.3$ et $\mu = 0.25$

tableau 2-1 : tableau récapitulatif des différents paramètres numériques

2.5.3. Maillages et temps de calcul

On présente ici des résultats obtenus avec un maillage plus fin que précédemment concernant le lopin (figure 2-31), afin d'obtenir les résultats les plus précis possibles. Le maillage initial du lopin contient 13819 nœuds. Il atteint 18779 nœuds en fin de roulage et retombe à 15934 nœuds en fin de planage. Les temps de calcul sont ici beaucoup plus conséquents que lors des deux précédentes et nécessite l'utilisation d'un cluster de calcul. La phase de roulage dure 13 h 9 min sur 9 processeurs, tandis que la phase de planage dure 15 h 43 min sur 8 processeurs. Les autres étapes du cycle ne sont quand à elles peu coûteuses en temps de calcul, puisqu'il ne s'agit que d'un calcul thermique. La simulation dédiée au laminage et au retour dure 51 min sur un processeur tandis que la simulation dédiée au refroidissement (durant l'extraction et l'insertion d'une bague) dure 2 min.

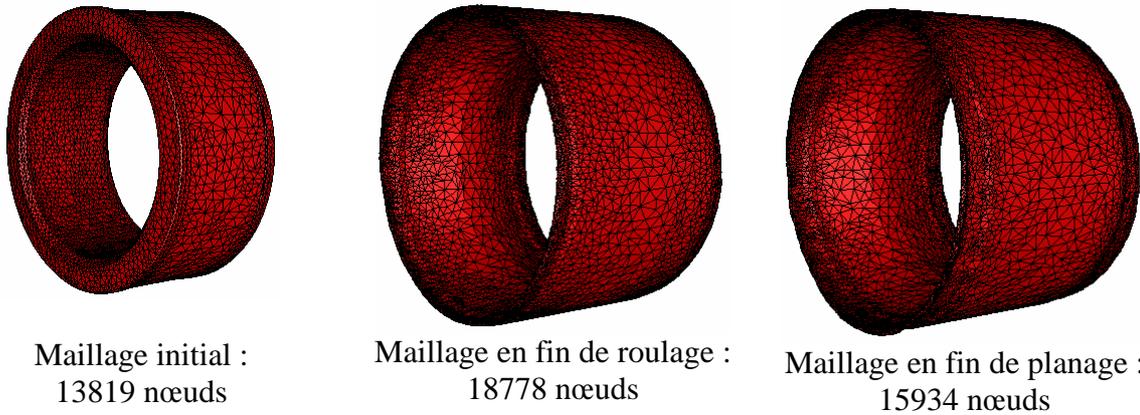


figure 2-31: maillages en début, fin de roulage et fin de planage en multi-corps

La matrice et la molette ont un comportement supposé rigide, leur thermique seule étant prise en compte. Les maillages les définissant restent donc les mêmes tout au long des différentes simulations. Ils sont visibles sur la figure 2-32.

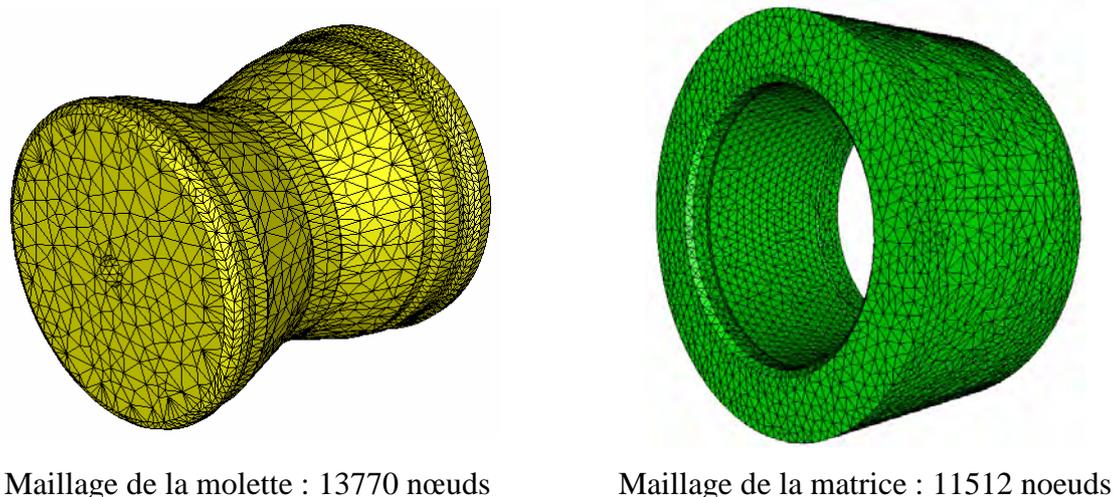


figure 2-32: maillages de la molette et de la matrice pour le symbole CUV40918

2.5.4. Contraintes équivalentes de Von Mises

La figure 2-33 présente l'évolution des contraintes équivalentes de Von Mises sur le lopin au cours de la phase de roulage. Dans la colonne du milieu, on visualise la partie interne du lopin, tandis que la colonne de droite montre la face extérieure du lopin. Après un tour, ces valeurs sont assez faibles, le lopin ayant une assez grande liberté de mouvement dans la matrice. Après trois tours, la bague s'est largement expansée et les contraintes les plus importantes se situent sur les bords interne et externe du lopin, avec des valeurs de l'ordre de 450 MPa. Après cinq tours, la bague a quasiment atteint son diamètre extérieur final et les contraintes les plus importantes dessinent une frange sur la partie externe du lopin. A la fin du roulage, les contraintes équivalentes à l'intérieur de la bague ont nettement diminué, les fortes valeurs se concentrant sur sa partie externe. On constate durant tout le roulage une concentration des contraintes équivalentes en amont du contact.

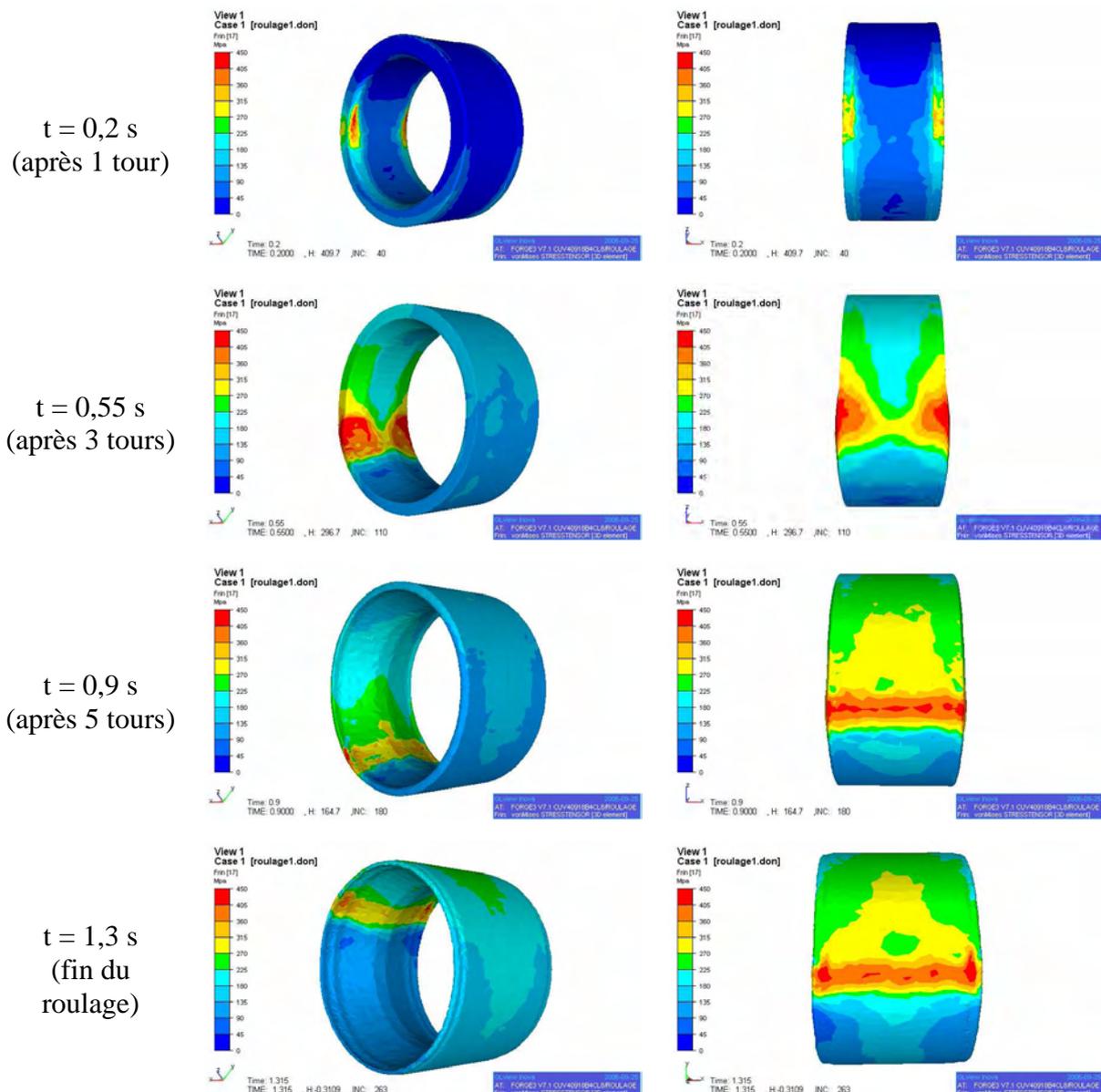


figure 2-33: évolution des contraintes de Von Mises au cours du roulage

2.5.5. Etape de planage

Au contact de ce lopin, la matrice emmagasine de la chaleur. La figure 2-34 montre une vue de coupe de la matrice permettant de visualiser l'évolution du gradient de température dans son épaisseur au cours du temps. La température en peau passe d'une moyenne de 410°C en début de planage à une moyenne de 470°C à la fin. Le gradient thermique dans l'épaisseur de la matrice évolue en conséquence, puisque la température à 20 mm de profondeur reste constante et égale à 190°C.

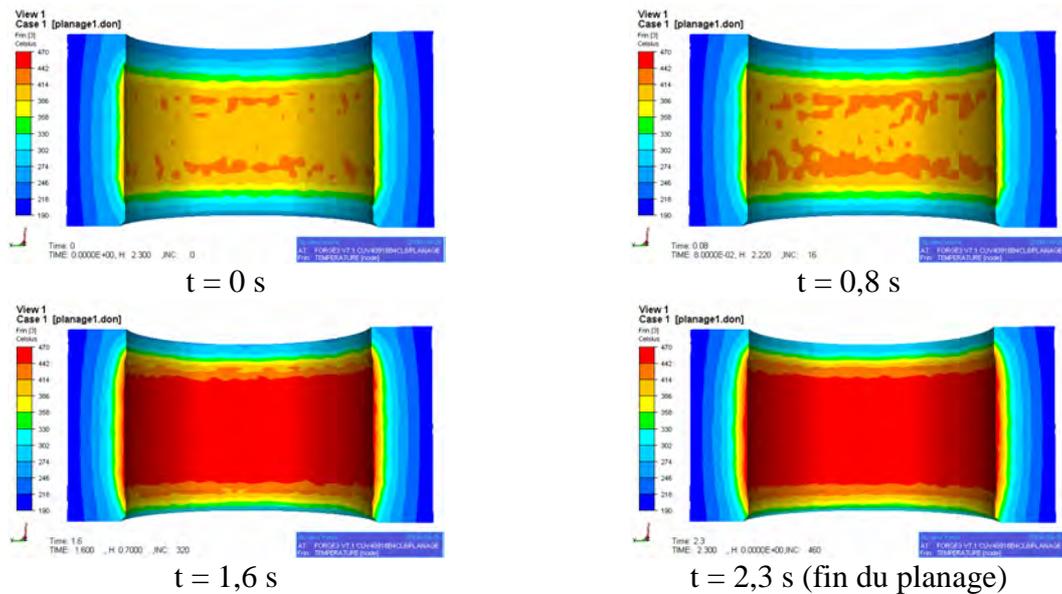


figure 2-34: évolution de la température de la matrice durant le planage (vue de coupe)

2.5.6. Etapes de laminage et de retour

On regroupe ces deux étapes dans la même simulation. Mécaniquement parlant, plus rien ne se passe, mais la thermique continue d'évoluer. Durant ces deux phases, la molette n'est plus en contact avec la bague, il n'y a donc plus de déformation de la matière. Cependant, la matrice et la bague restent en contact et échangent de la chaleur. On néglige les échanges par radiation et le refroidissement par lubrification. On simule cet échange thermique, durant 3.5 s, temps nécessaire pour réaliser ces deux étapes expérimentalement. Le coefficient d'échange thermique du lopin avec l'air est toujours égal à 500 W/m².K et le coefficient d'échange thermique des outils vaut toujours 7000 W/m².K. La température extérieure garde également la même valeur que durant les phases de roulage et de planage, à savoir 150°C.

2.5.7. Etapes d'extraction de la bague et d'insertion du lopin

De même que précédemment, on regroupe ces deux étapes dans la même simulation. Durant ces deux phases, le lopin n'est plus en contact avec la matrice. On simule donc simplement le refroidissement de la matrice à l'air libre, tout en gardant à l'esprit que la température de la surface externe de la matrice est maintenue à 190°C pour se conformer aux relevés expérimentaux. Ces deux étapes cumulées durent 10,5 s. Les coefficients d'échange

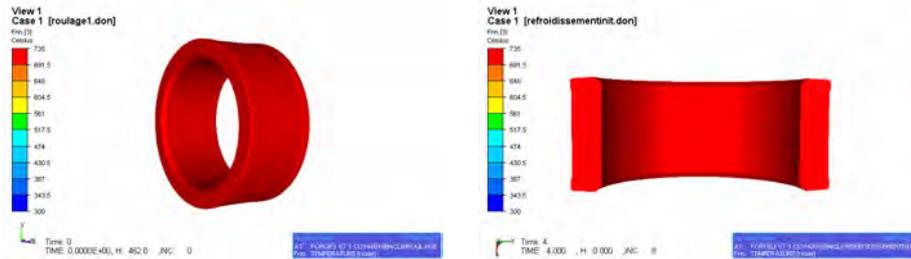
thermique du lopin et des outils gardent toujours leurs valeurs précédentes. Cependant, la température extérieure est maintenant égale à 50°C, ce qui caractérise l'état ouvert de la machine.

2.5.8. Evolution de la température du lopin

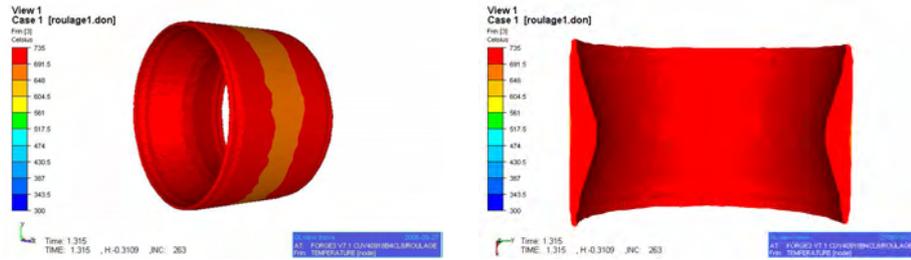
La figure 2-35 présente l'évolution de la température du lopin à différents moments clefs du cycle. La colonne du centre montre le lopin dans son intégralité, tandis que la colonne de droite fait apparaître un plan de coupe du lopin permettant d'apprécier les variations de température dans l'épaisseur. Il est à noter que l'échelle de temps n'est pas la même que lors des relevés de température sur la matrice, puisque l'instant de départ est ici la sortie du four, tandis que précédemment, l'instant initial correspondait au début de la phase de roulage.

La température du lopin à la sortie du four à induction est uniforme et égale à 735°C. Au début du roulage, le lopin a légèrement refroidi et sa température est comprise entre 730 et 720°C. Au départ de l'étape de planage, on ne perçoit toujours pas de gradient de température dans l'épaisseur ni dans la partie interne du lopin. Cependant, une frange de température inférieure à la moyenne fait son apparition au centre sur la surface externe du lopin. Au commencement du laminage, on constate une franche diminution de la température sur la face externe du lopin, due au contact avec la matrice. Sur le plan de coupe, on distingue un gradient de température dans l'épaisseur avec deux parties distinctes : la partie interne du lopin encore à une température de l'ordre de 700°C et la partie externe du lopin ainsi que les bords qui refroidissent plus rapidement, avec une température comprise entre 520 et 650°C. Le gradient de température devient de plus en plus prononcé à l'extraction de la bague, avec des bandes de températures concentriques. Après extraction de la bague et 11 s de refroidissement, la température s'homogénéise et atteint des valeurs de l'ordre de 490°C au cœur de l'épaisseur et de 250°C dans les bavures.

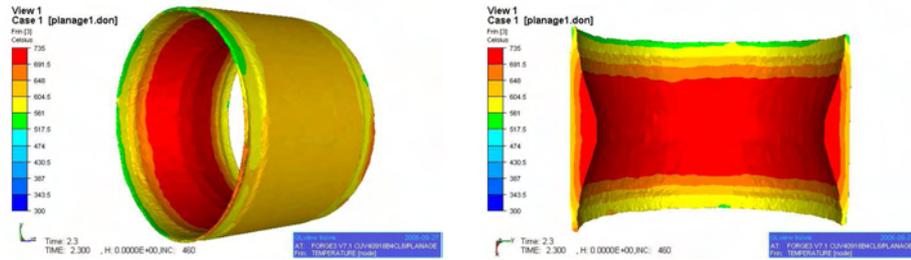
Début du roulage
t = 4 s



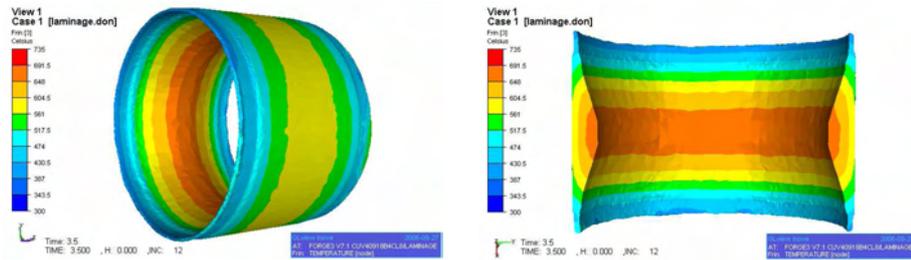
Début du planage
t = 5,3 s



Début du laminage
t = 7,6 s



Extraction de la bague
t = 11,1 s



Après 11 s de refroidissement
t = 22,1 s

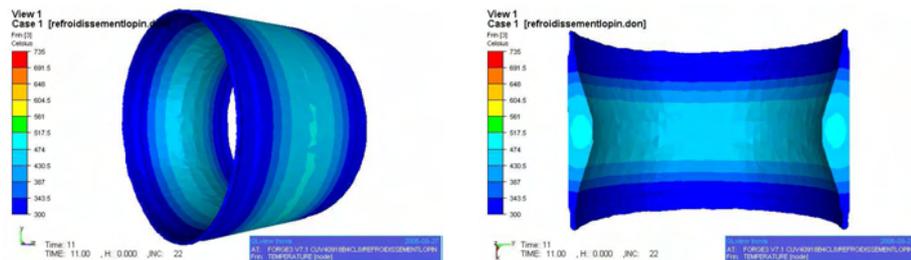


figure 2-35: évolution de la température du lopin durant le cycle

La figure 2-36 représente un plan de coupe de la bague après 11 secondes de refroidissement, mais avec une échelle de température différente de la figure 2-35. Elle permet de mieux visualiser le gradient de température dans l'épaisseur. On observe une zone au centre du lopin où la chaleur est la moins dissipée, tandis qu'aux opposés les bavures sont les parties les plus froides de la bague.

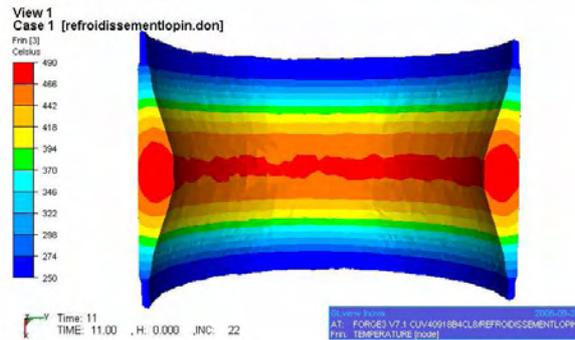


figure 2-36: température de la bague après 11 secondes de refroidissement (plan de coupe)

2.5.9. Remplissage

On visualise sur la figure 2-37 la distance entre la CAO d'une bague roulée et le maillage d'une bague issue de la simulation en fin de cycle. On constate une distance positive au niveau des deux bavures, ce qui est normal puisque la CAO est celle d'une bague ébavurée. En ce qui concerne les distances entre la CAO et la simulation sur les parties externes et internes de la bague, on constate une erreur de remplissage inférieure à 1/10 de mm.

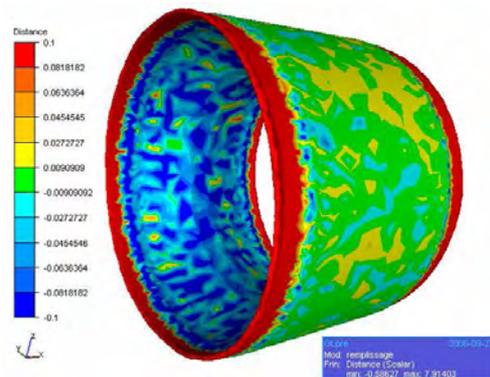


figure 2-37: remplissage sur le symbole CUV40918

2.5.10. Bilan de la simulation

Cette étude présente les premiers résultats de simulations prenant en compte la thermique des outils. Le travail de recalage thermique rendu possible par la campagne expérimentale menée chez SNR a permis d'identifier les paramètres d'échange thermique ainsi que de travailler sur les conditions initiales thermiques des outils. Ce recalage, validée par la bonne adéquation des simulations avec l'expérience permet l'obtention d'informations précieuses sur l'évolution de la température de la matrice et de la molette une fois le régime thermique presque établi. C'est un premier pas primordial pour la compréhension et vers la prédiction de l'usure des outils.

2.6. Validation de la simulation sur le symbole BEGB40899

Afin de valider complètement le travail de paramétrisation effectué, on s'intéresse maintenant à l'étude d'un symbole de bague différent, avec une géométrie plus complexe. Le but est d'utiliser les paramètres identifiés sur le symbole précédent afin de s'assurer s'ils sont valables sur un autre symbole. On s'intéresse donc de nouveau à la comparaison des relevés expérimentaux de température dans la matrice avec les résultats numériques.

2.6.1. Molette, bague

On présente sur la gauche de la figure 2-38 une CAO de la molette correspondante au symbole référencé BEGB40899. Sur la droite est visible une CAO de la bague roulée.

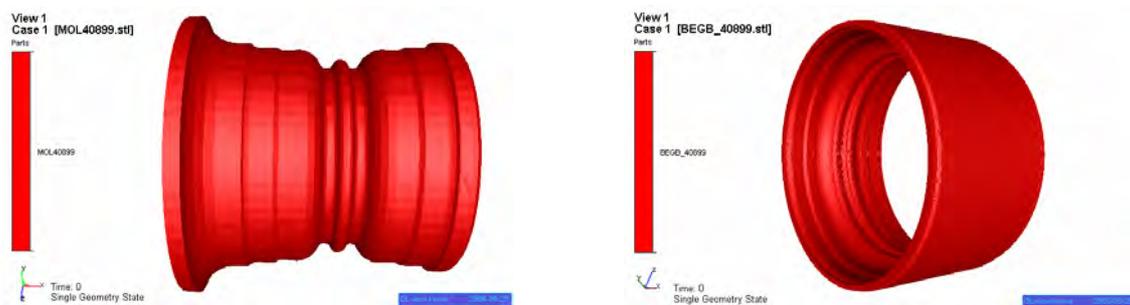


figure 2-38: molette (gauche) et bague roulée (droite) du symbole BEGB40899

2.6.2. Données expérimentales

2.6.2.1. Relevé moyen des thermocouples

On dispose des mêmes informations thermiques expérimentales que précédemment sur la matrice à 3 mm et 20 mm durant un cycle, concernant le symbole BEGB40899. La figure 2-39 monte l'évolution de ces températures.

Les traits verticaux délimitent les différentes étapes au cours d'un cycle. On peut remarquer que la durée des différentes étapes n'est pas la même que celle du premier symbole. La phase de roulage dure ici 1,37 s, tandis que la phase de planage dure 0,68 s. L'étape de laminage associée à l'étape de retour dure 4 s et l'étape d'extraction de la bague cumulée avec l'insertion du lopin et l'avance rapide dure 6,5 s.

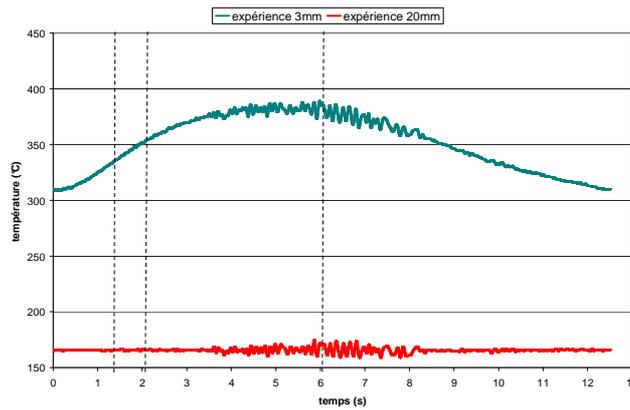


figure 2-39: évolution de la température à 3 mm et 20 mm sur BEGB40899

On constate un certain bruit sur les courbes de température durant les phases de retour et d'extraction de la bague. Il est difficilement explicable, les thermocouples étant « incrustés » dans la matrice. Les mesures relevées restent cependant totalement acceptables et exploitables. Il apparaît à 20 mm de profondeur une température moyenne constante au cours du cycle et égale à 165°C. Pour la température à 3 mm de profondeur, elle varie de 310°C juste avant le roulage à 390°C au départ de l'étape de retour.

2.6.2.2. Trajectoire de la molette

On utilise ici la trajectoire point par point basée sur un relevé expérimental. Cette trajectoire en spirale est visible sur la gauche de la figure 2-40, tandis que l'évolution de l'excentration de la molette est visible sur la droite.

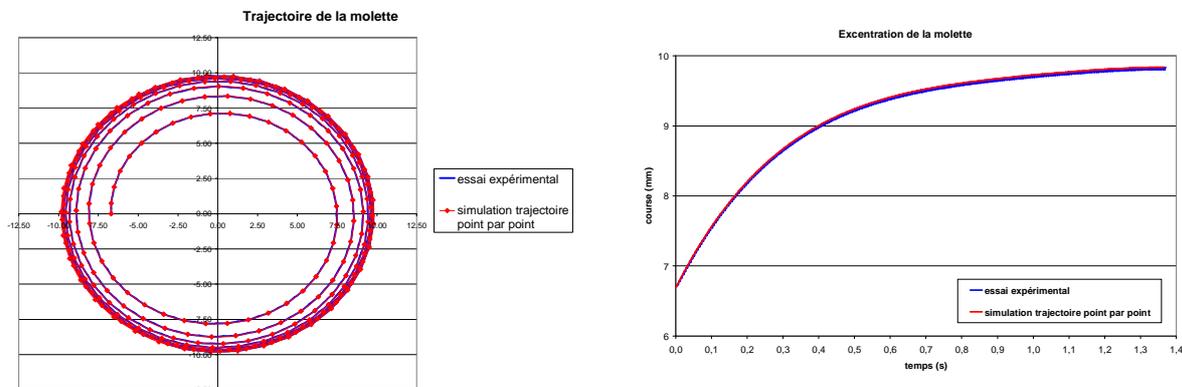


figure 2-40: comparaison de trajectoires avec une discrétisation point par point sur le symbole BEGB40899

2.6.3. Données numériques

On utilise ici les mêmes conditions initiales thermiques pour le lopin, à savoir une température uniforme en sortie de four égale à 735°C et un temps de transfert avant le roulage de 4 s.

Les paramètres numériques thermiques du lopin, des outils et du milieu ambiant restent également inchangés par rapport au symbole CUV40918. Le but est ici de réussir à reproduire les courbes de température de la matrice sur un cycle par la simulation sur ce symbole, afin de

valider les données numériques recalées sur le symbole précédent. Les lois de comportement du lopin et de frottement restent donc également inchangées.

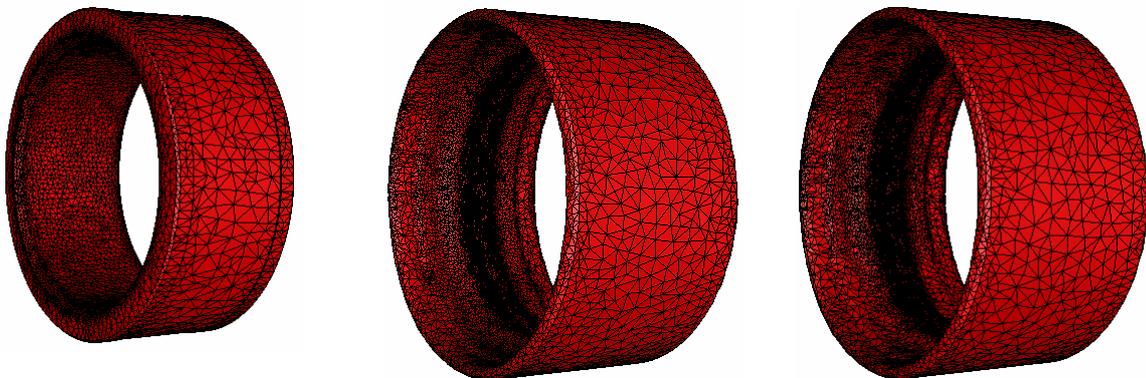
L'état initial thermique de la matrice est obtenu en lui imposant un gradient de température avec une valeur de 165°C à 20 mm de profondeur et de 310°C à 3 mm. En ce qui concerne la molette, on simule entièrement 3 cycles consécutifs afin d'obtenir un régime thermique établi.

On impose comme condition aux limites de la matrice (i.e. à 20 mm de profondeur) une température constante égale à 165°C durant toutes les simulations nécessaires à la modélisation d'un cycle.

2.6.4. Maillages et temps de calcul

La complexité géométrique du symbole nécessite l'emploi d'un maillage comportant un nombre conséquent de nœuds afin d'obtenir une bonne précision dans les résultats. Le maillage initial comporte 19960 nœuds. Ce nombre atteint une valeur de 22610 en fin de roulage, puis de 34354 en fin de planage. Ces maillages sont visibles sur la figure 2-41. La figure 2-45 montre une coupe du maillage en fin de planage permettant de mieux visualiser la finesse du maillage, notamment au centre de la molette où se situe le chemin des billes.

La simulation du roulage nécessite 15 h 26 min de temps de calcul sur 9 processeurs, tandis que la simulation du planage dure 14 h 36 min sur 9 processeurs également. La simulation des étapes de laminage et de retour dure 32 min en séquentiel et la simulation du refroidissement demande 2 min.



Maillage initial :
19960 nœuds

Maillage en fin de roulage :
22610 nœuds

Maillage en fin de planage :
34354 nœuds

figure 2-41: maillages en début, fin de roulage et fin de planage en multi-corps

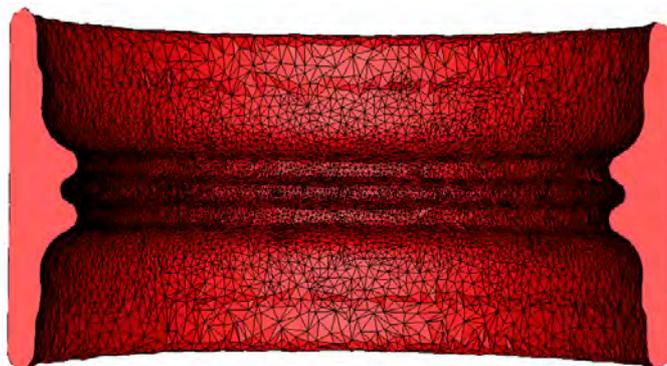
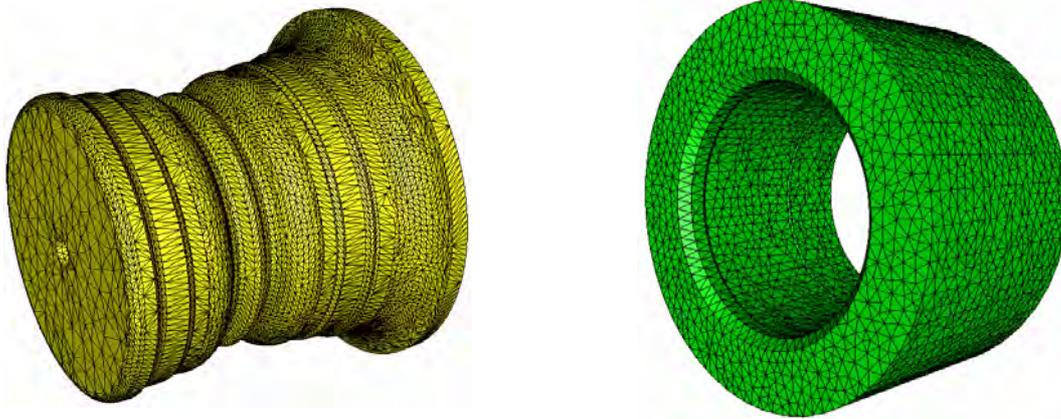


figure 2-42: maillage en fin de planage en multi-corps (plan de coupe)

De même que dans l'étude précédente, les maillages de la molette et de la matrice sont inchangés au cours du temps, puisque seul leur thermique est prise en compte. Ces maillages sont visibles sur la figure 2-43.



Maillage de la molette : 43786 nœuds

Maillage de la matrice : 15424 noeuds

figure 2-43: maillages de la molette et de la matrice pour le symbole BEGB40899

2.6.5. Evolution de la température du lopin au cours d'un cycle

La figure 2-44 présente l'évolution de la température du lopin au cours de la phase de roulage. Est visible au milieu une vue entière du lopin, tandis que sur la droite une coupe du lopin permet de visualiser la température dans l'épaisseur. En surface, la température initiale est de l'ordre de 720°C. Sur la paroi extérieure, elle décroît durant tout le roulage de manière quasi homogène pour atteindre une température de 650°C à la fin de cette étape. Sur la partie interne du lopin, on constate un échauffement de la matière dû au frottement sur les zones en contact avec la molette. A la fin du roulage apparaît un gradient de température, avec une zone à 770°C au centre correspondant aux deux couloirs délimitant le chemin des billes, Les bordures externes du lopin atteignent une température de 710°C.

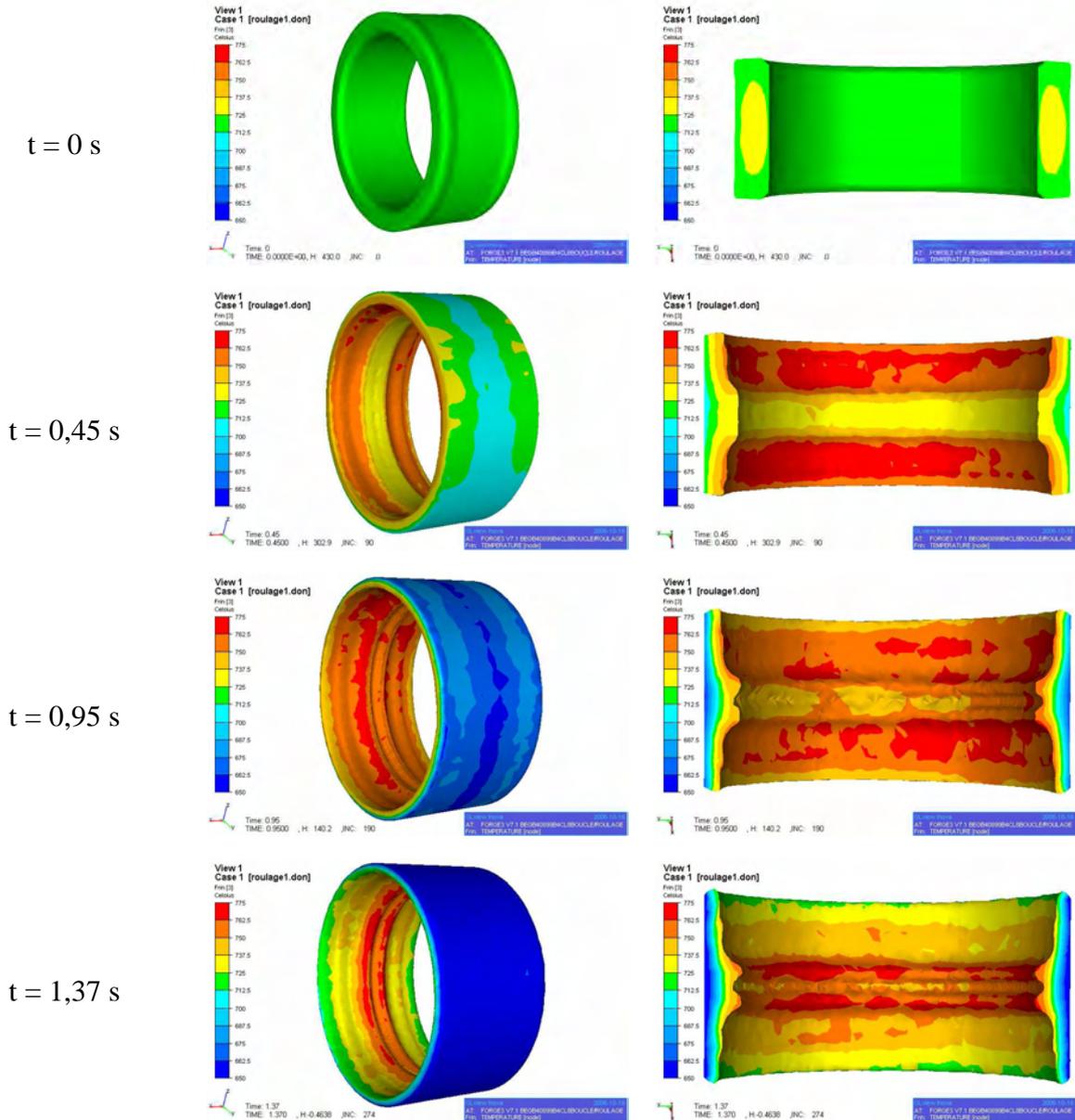


figure 2-44: température du lopin durant le roulage sur BEGB40899

Durant la phase de planage, il n'y a quasiment plus de contraintes appliquées sur ce dernier a pratiquement atteint sa configuration géométrique finale. On constate ainsi un refroidissement de la matière lié aux échanges thermiques et aux contacts permanents avec la matrice et ponctuel avec la molette (figure 2-45). A la fin du planage, la température de la face extérieure de la bague est de 610°C tandis que sur l'intérieur, les deux couloirs du chemin ont une température surfacique de l'ordre de 730°C et le couloir de 715°C. Un gradient de température homogène est visible dans l'épaisseur du lopin.

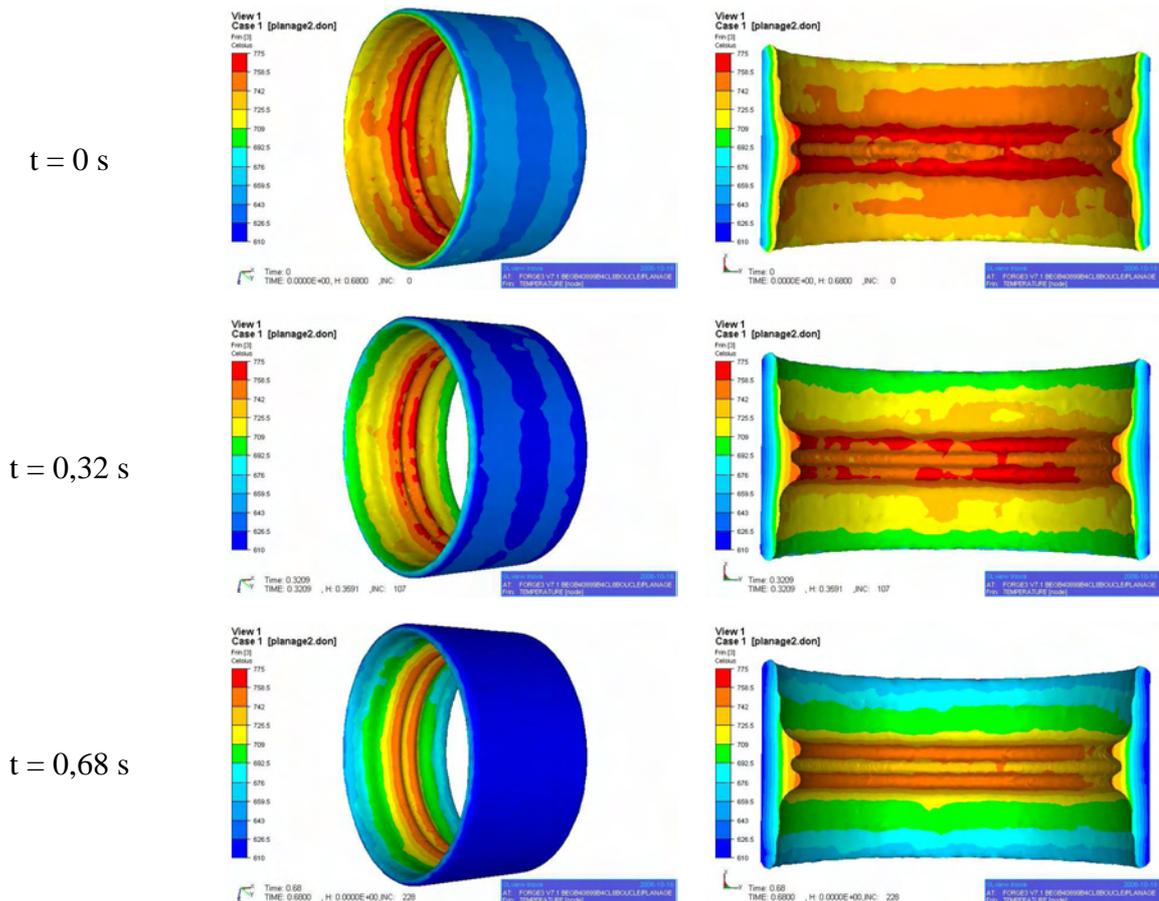


figure 2-45: température du lopin durant le planage sur BEGB40899

Durant les phases de laminage et de retour (figure 2-46), un refroidissement homogène s'opère sur la face extérieure de la bague, dû au contact permanent de la matrice et de la matière. Sur la face interne, la bague refroidit au contact de l'air. Le gradient de température à la fin de la phase de retour part des deux couloirs du chemin, avec une zone à 625°C, et décroît constamment pour atteindre une température de 460°C sur les bords de la bague.

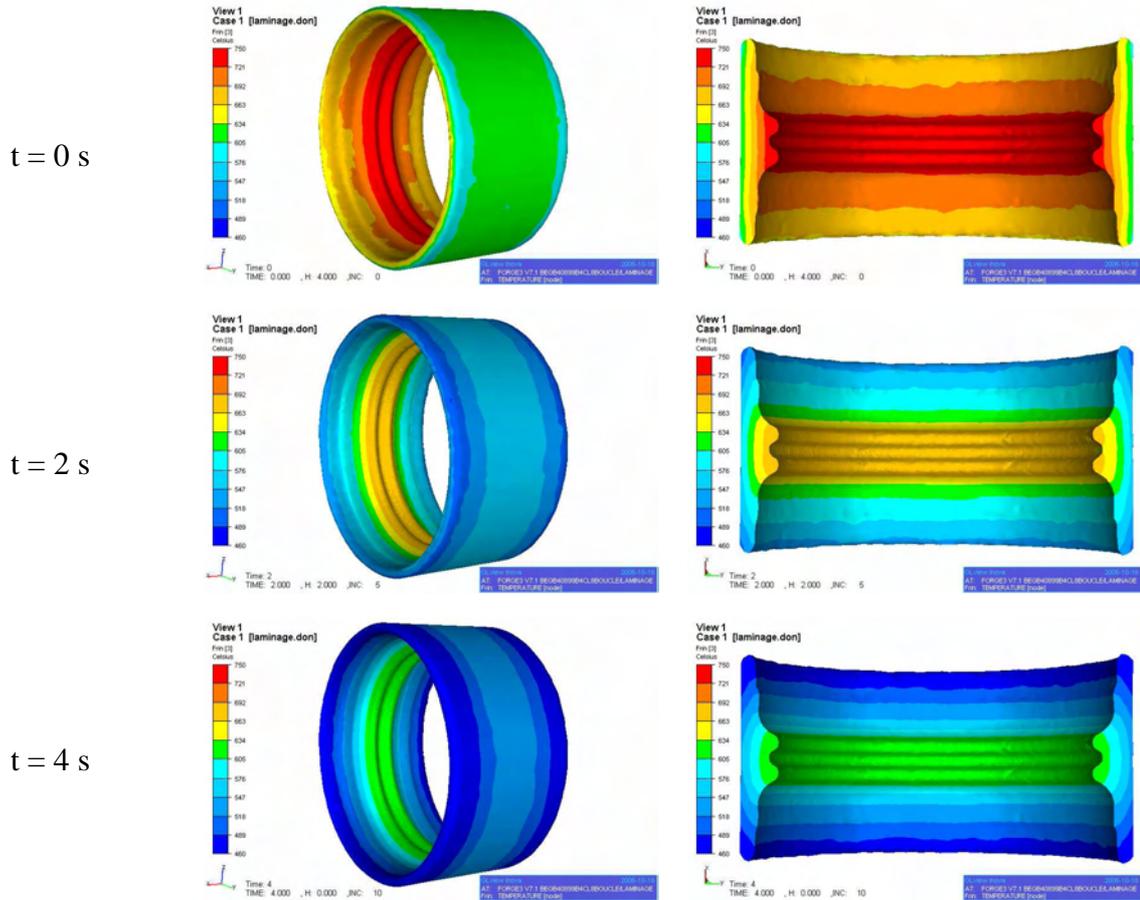


figure 2-46: température du lopin durant le laminage et le retour sur BEGB40899

La figure 2-47 présente la configuration thermique de la bague 10 s après son extraction de la rouleuse. Le centre de la bague est à une température de l'ordre de 545°C, cette dernière diminuant constamment jusqu'aux extrémités.

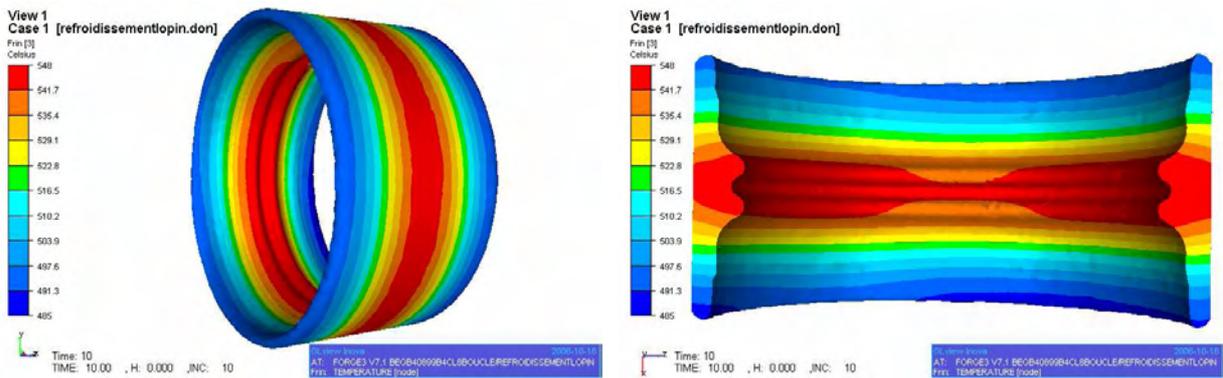


figure 2-47: température du lopin après 10 secondes de refroidissement sur BEGB40899

2.6.6. Cycle thermique de la matrice

Comme on peut le constater sur la figure 2-48, la température simulée à 20 mm reste constante égale à 165°C, ce qui traduit le respect des conditions aux limites. En ce qui concerne la température à 3 mm de profondeur, on constate une parfaite reproduction des

valeurs expérimentales durant la simulation sur l'ensemble des phases d'un cycle. Ce résultat très satisfaisant permet de valider le recalage effectué sur le symbole précédent et de dire que les paramètres identifiés sont valables quel que soit le type de bague à rouler. La très bonne adéquation des courbes expérimentales et numériques traduit la prédictibilité de la simulation sur le comportement thermique de la matrice tout au long d'un cycle.

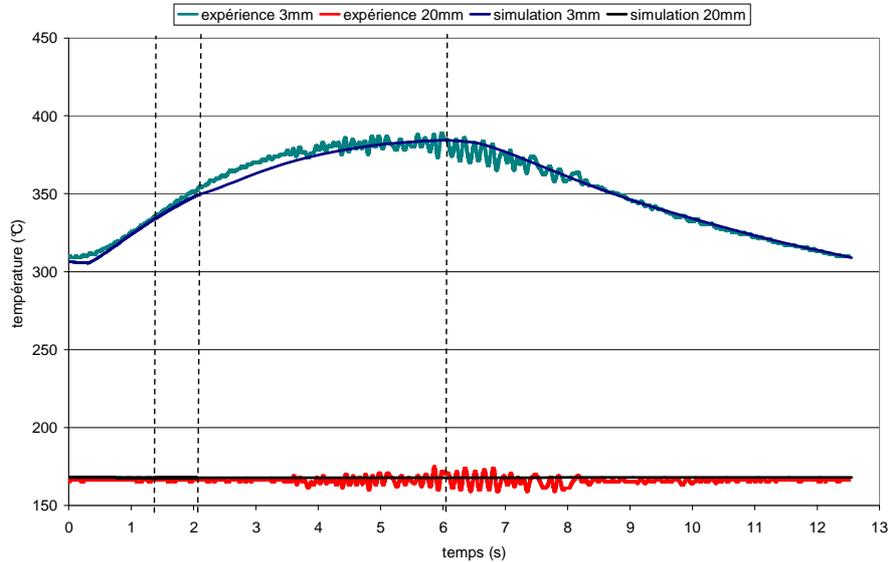


figure 2-48: comparaison de la température de la matrice durant un cycle sur BEGB40899

2.6.7. Remplissage

La figure 2-49 montre le remplissage du symbole BEGB40899. On visualise la distance entre la CAO d'une bague roulée et le maillage d'une bague en fin de simulation. L'échelle est exprimée en millimètre. Sur la partie externe de la bague, on constate une variation de la distance de l'ordre de 0,06 mm qui est dû à une discrétisation légèrement trop grossière de la matrice. Ce choix s'explique par la nécessité de réduire au maximum le nombre de degrés de liberté dans les outils, afin de ne pas trop pénaliser le temps de calcul de la simulation, s'agissant d'un calcul multi-corps. Cette variation du remplissage sur la partie externe du lopin n'a rien de pénalisant. Sur la partie interne et spécialement sur les couloirs et le chemin, le remplissage est tout à fait satisfaisant, avec une précision numérique de l'ordre de 0,02 mm. Sur le reste de la face interne de la bague, la précision du remplissage atteint 0,02 mm. Ces valeurs permettent de valider la modélisation de la cinématique de la rouleuse.

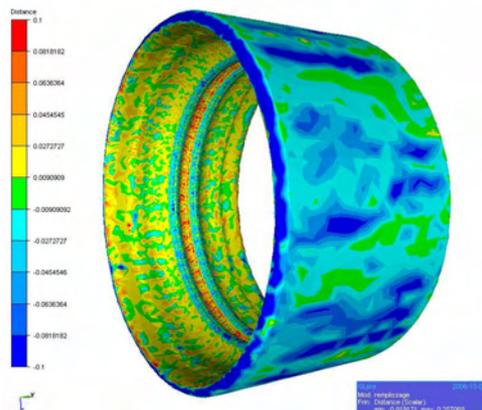


figure 2-49: remplissage sur le symbole BEGB40899

2.7. Conclusions

Par l'intermédiaire de nouveaux développements relatifs à la cinématique pour les outils flottants et à une campagne expérimentale très riche réalisée chez SNR, une grande avancée résulte de ce travail sur la simulation du procédé de roulage.

Le développement d'un nouveau formalisme d'outil dans le logiciel permet de simuler la cinématique exacte de la rouleuse. Par l'intermédiaire de l'ajout de degrés de liberté supplémentaires dans le système à résoudre, on modélise d'une part le pilotage en force maximale de la presse et le caractère flottant de la molette d'autre part. Ces développements ont été validés par confrontation des simulations à l'expérience, avec des résultats très satisfaisants. En effet, la simulation se montre prédictive sur la trajectoire de la molette durant la phase de roulage ainsi que sur la durée de cette étape, ce qui est d'une grande importance au niveau industriel à la vue les cadences de fabrication.

Le deuxième volet de cette étude s'articule autour du travail de recalage effectué afin que la simulation reflète au mieux l'expérience. Il est rendu possible par la campagne expérimentale menée chez SNR. A partir d'un type de bague, un certain nombre de coefficients thermiques ont été recalés afin de faire correspondre l'évolution de la température dans la matrice durant un cycle entier. Ce nouveau jeu de paramètres a été ensuite testé sur un autre type de bague, de géométrie et de taille significativement différentes, avec des résultats excellents permettant la validation de ces valeurs. La simulation se montre prédictive sur l'évolution de la température de la matrice tout au long d'un cycle en régime thermique établi, ce qui valide l'approche et indirectement la valeur des coefficients retenus.

Ces améliorations de la simulation permettent dorénavant de disposer de résultats fiables concernant tout d'abord l'évolution mécanique et thermique du lopin au cours d'un cycle de mise en forme, mais également concernant l'évolution de l'état thermique des outils.

2.8. Perspectives

Le travail effectué ici présente des premiers résultats significatifs permettant de mieux appréhender le roulage. Cependant il ne s'agit que d'une première étape, indispensable pour un autre objectif essentiel, la prédiction de la durée de vie moyenne de la molette, et les motifs de casses éventuelles. Disposant dorénavant de simulations prédictives, il faudra les utiliser pour étudier l'endommagement de la molette, et enfin pour optimiser le procédé.

De plus, la connaissance des données mécaniques de la molette rend possible la réduction des temps de cycle, en optimisant le pilotage de la machine et le temps de planage. Il faudra prendre soin de ne pas trop augmenter les contraintes sur l'outillage en réduisant ce temps, d'où la nécessité de calculer les contraintes appliquées à la molette, ainsi que sa réponse.

Enfin, en analysant les différentes simulations effectuées, il paraît indispensable d'essayer de réduire les temps de calcul. En effet, bien que les temps de calcul des simulations mono-corps présentés restent acceptables, grâce à l'utilisation de maillage relativement pauvre en terme de nombre de nœuds, on constate que la simulation multi-corps, comportant un plus grand nombre de nœud, est extrêmement coûteuse à réaliser. Elle le sera d'ailleurs encore plus si l'on prend en compte le comportement mécanique de la molette. Le chapitre suivant présente les travaux effectués dans ce sens, avec l'introduction d'une nouvelle méthode de résolution de systèmes linéaires dans le code, basée sur des algorithmes multigrilles.

3. Les méthodes multigrilles

Dans ce chapitre, on présente une nouvelle méthode de résolution de systèmes linéaires basée sur des algorithmes multigrilles. Cette méthode itérative est introduite dans le logiciel Forge3® en lieu et place de la méthode de résolution présentée en 1.1.3.3 dans le but de réduire les temps de calcul. En effet, la caractéristique principale des méthodes multigrilles est une convergence asymptotique optimale.

On présente dans un premier temps les méthodes multigrilles, leur principe, les différentes manières de les mettre en œuvre. S'en suit une partie sur la méthode développée dans cette étude, avec la présentation du déraffinement automatique de maillage sur laquelle elle repose et qui constitue comme une étape déterminante. On regarde ensuite l'influence de la qualité du déraffinement sur les performances du solveur. Enfin, nous exposons une étude paramétrique permettant d'optimiser la méthode, ainsi que ses performances sur différents cas industriels de mise en forme, dont le roulage.

Ce travail a été présenté à la conférence EMG2005 [Rey et al.05] dédiée aux méthodes multigrilles et fut l'objet d'une publication [Rey et al.07] dont l'ensemble des résultats est détaillé dans la suite de ce manuscrit.

3.1. Présentation

Les méthodes multigrilles sont des méthodes de résolution de systèmes linéaires relativement récentes, apparues dans les années 1960 [Fedorenko62]. Historiquement introduites pour la résolution de l'équation de Poisson, leur emploi se démocratise dans le cadre de la mécanique des fluides, où leur utilisation se conjugue assez facilement avec des discrétisations volumes finis ou différences finis [Wesseling92]. On trouve ainsi des travaux traitant de la résolution des équations d'Euler [South et al.76], [Jameson et al.81], [Ollivier-Gooch95], du problème de Stokes [Vincent97], de Navier-Stokes [Brandt et al.79], [Wesseling et al.80] et de problèmes de turbulence [Sheng et al.95], [Carre95]. Leur utilisation dans le cadre d'une discrétisation éléments finis est plus récente. On peut noter dans le domaine de la mécanique des solides des travaux sur la visco-plasticité [El Hadj87], [Mocellin99], sur l'élasticité [LeDain91], [Lee98] mais aussi sur l'élasto-plasticité [Feng et al.98].

La caractéristique la plus intéressante de ces méthodes multigrilles est une convergence asymptotique en $O(N)$ où N est le nombre d'inconnues du système résolu. Cette complexité théorique a été démontrée dans divers cas de discrétisation d'équations elliptiques [Wesseling82], [Bachvalov66], [Hackbusch78]. Leurs performances sont ainsi bien meilleures que celles des méthodes itératives classiques utilisées jusqu'alors et qui présentent des convergences asymptotiques de l'ordre de $O(N^\alpha)$, avec $1 < \alpha \leq 2$.

La particularité de la méthode vient de l'emploi, comme son nom l'indique, de plusieurs niveaux de calcul dans le but d'accélérer la convergence des calculs. En fonction de la méthode d'obtention de ces différents niveaux de calcul, on divise les multigrilles en deux familles, les approches géométrique et algébrique. Cette distinction est détaillée dans la suite de ce chapitre.

3.1.1. Principe

L'utilisation de plusieurs niveaux de calcul est motivée par l'observation que les méthodes itératives classiques (gradient conjugué, Jacobi, Gauss-Seidel) présentent un très bon lissage des hautes fréquences de la solution, ce qui se traduit par une décroissance rapide de la norme du résidu durant les toutes premières itérations de résolution. Le principe des méthodes multigrilles est donc de faire quelques itérations d'une méthode itérative classique sur le niveau de calcul courant afin d'approximer au mieux les hautes fréquences, puis on restreint le problème à un système grossier, plus pauvre en terme de nombre de degrés de liberté mais sur lequel les composantes basses fréquences sont représentées, afin de les évaluer beaucoup plus rapidement.

On présente dans un premier temps la méthode bigrille en ne considérant que le niveau de calcul usuel et un niveau de calcul grossier. On généralise ensuite au cadre des méthodes multigrilles, où le nombre de niveaux de calcul grossiers peut être quelconque.

3.1.1.1. La méthode bigrille

On considère d'une part le niveau de calcul traditionnel, représenté par la matrice A_h et le second membre b_h du système discrétisé. On considère d'autre part un niveau de calcul grossier défini par la matrice grossière A_H . L'obtention de cet élément grossier sera discutée par la suite.

Le problème discret à résoudre est :

$$A_h X_h = b_h \quad (3-1)$$

Avec X_h le vecteur solution.

Les différentes étapes composant un cycle multigrilles sont :

- Pré-lissage

On effectue ν itérations de lissage à l'aide d'une méthode de résolution itérative L . On obtient une solution approchée de X_h : \tilde{X}_h . Dans la terminologie des multigrilles, le lissage équivaut donc à des itérations d'une méthode itérative classique.

$$\tilde{X}_h \leftarrow L(A_h, b_h, \nu) \quad (3-2)$$

- Calcul du résidu

On calcule le résidu r_h du système.

$$r_h = b_h - A_h \tilde{X}_h \quad (3-3)$$

- Restriction du résidu

Par l'intermédiaire de l'opérateur de restriction p_h^H , on restreint le résidu au niveau grossier.

$$r_H = p_h^H r_h \quad (3-4)$$

- Calcul de la correction

On calcule la correction de la grille grossière δX_H avec une méthode de résolution S .

$$\delta X_H = S(A_H, r_H) \quad (3-5)$$

- Prolongement de la correction

On prolonge cette correction δX_H sur le niveau de calcul fin pour obtenir δX_h par l'intermédiaire de l'opérateur de prolongement p_H^h .

$$\delta X_h = p_H^h \delta X_H \quad (3-6)$$

- Correction

On actualise la valeur de \tilde{X}_h en lui ajoutant la correction de la grille grossière δX_h .

$$\tilde{X}_h \leftarrow \tilde{X}_h + \delta X_h \quad (3-7)$$

- post-lissage

On effectue ν' itérations de lissage à l'aide d'une méthode de résolution itérative L' (en général, la même que la méthode de pré-lissage).

$$\tilde{X}_h \leftarrow L'(A_h, b_h, \nu') \quad (3-8)$$

\tilde{X}_h tend vers X_h .

La figure 1-1 représente le cycle bigrille qui vient d'être décrit.

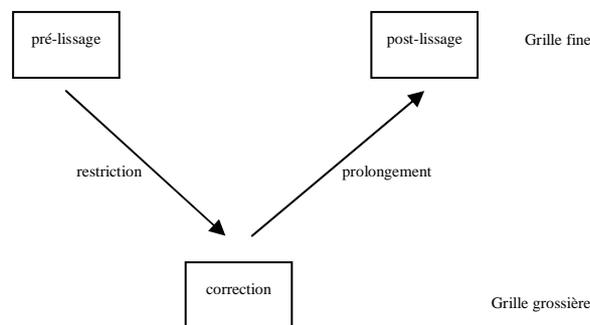


figure 3-1: un cycle bigrille

Ce cycle est l'élément itératif des méthodes multigrilles. Les cycles se succèdent donc jusqu'à obtention de la précision de convergence souhaitée.

Les différents paramètres de la méthode à définir sont :

- le pré-lisseur L et son nombre d'itérations ν
- le solveur sur la grille grossière S ainsi que la précision de résolution
- le post-lisseur L' et son nombre d'itérations ν'

Les différents éléments de la méthode à construire sont :

- l'opérateur de restriction p_h^H
- l'opérateur de prolongement p_H^h
- la matrice du système grossier A_H

Le choix des différents paramètres ainsi que les méthodes de construction des différents éléments nécessaires est détaillé par la suite.

3.1.1.2. Les méthodes multigrilles

Le passage d'une méthode bigrille à une méthode multigrilles s'effectue en modifiant l'étape (3-5) du cycle bigrille. On remplace la méthode de résolution S du problème grossier par une nouvelle méthode bigrille utilisant un système encore plus grossier. Il suffit pour cela de définir de nouveaux opérateurs de restriction et de prolongement ainsi que les éléments du système de ce nouveau niveau. En itérant le processus, on obtient une méthode basée sur n grilles.

Dès lors que l'on dispose de trois niveaux de grille ou plus, il existe de nombreuses manières d'enchaîner les résolutions aux différents niveaux. Dans le cadre des méthodes sur trois niveaux, on peut noter le V-cycle, le N-cycle et le W-cycle, décrits sur la figure 3-2. Les variations viennent de l'omission de certaines étapes de post-lissage ou de résolution sur certains niveaux.

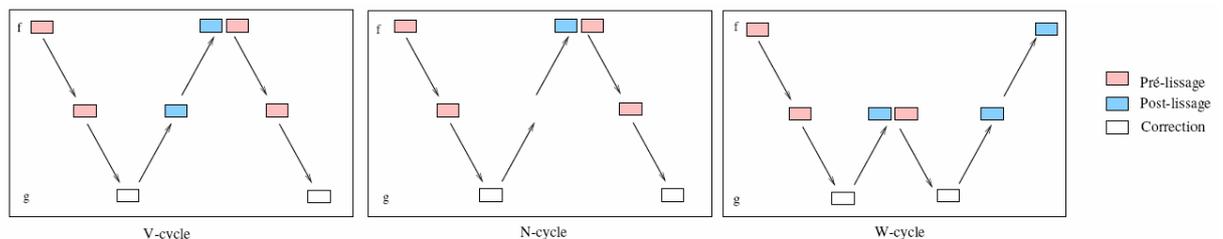


figure 3-2: différents cycles 3 grilles

Dans la pratique, la programmation d'un V-cycle sur n niveaux de grilles s'effectue de manière récursive. Soit $A_0 X_0 = b_0$ le système à résoudre. L'algorithme présenté sur le tableau 3-1 résume la méthode de résolution (on appelle $MG(A_0, b_0)$). L'opérateur R_i restreint le résidu du niveau fin i au niveau grossier $i+1$ et l'opérateur P_i prolonge le résidu du niveau grossier $i+1$ au niveau fin i .

La généralisation d'une méthode bigrille à une méthode multigrilles ne posant pas de défi particulier et dans un souci de compréhension, on se limitera dans la suite de cette partie d'introduction aux méthodes multigrilles à la considération de deux niveaux de grilles. Le niveau de calcul s'apparente au niveau fin et est dénoté par la lettre h tandis que le niveau grossier est représenté par la lettre H .

tableau 3-1 : algorithme d'un V-cycle multigrilles (n niveaux de grilles)

```

fonction  $MG(A_i, b_i)$  {
  si  $i < n$  alors {
     $X_i \leftarrow L(A_i, b_i, v)$  // pré-lissage
     $r_i = b_i - A_i X_i$  // calcul du résidu
     $b_{i+1} = R_i r_i$  // restriction du résidu
     $\delta X_{i+1} = MG(A_{i+1}, b_{i+1})$  // calcul de la correction
     $\delta X_i = P_i \delta X_{i+1}$  // prolongation de la correction
     $X_i \leftarrow X_i + \delta X_i$  // correction
     $X_i \leftarrow L'(A_i, b_i, v')$  // post-lissage
  } sinon {
     $X_i \leftarrow S(A_i, b_i)$  // résolution du système le plus grossier
  }
  retourne  $X_i$ 
}

```

L'architecture des méthodes multigrilles étant maintenant définie, on s'intéresse à la construction des opérateurs de restriction et de prolongement ainsi que de la matrice du système grossier. Dans les parties suivantes sont détaillées les deux approches existantes leur construction, à savoir les multigrilles algébriques et les multigrilles géométriques.

3.1.2. Multigrilles algébriques (AMG)

3.1.2.1. Principe

Cette méthode est apparue dans le début des années 1980 [Alcouffe et al.81], [Brandt et al.82] pour la résolution de problèmes elliptiques. Elle permet de facilement automatiser la construction des entités nécessaires [Dendy82]. En effet, cette approche se base uniquement sur l'étude de la matrice de calcul pour construire le niveau grossier et les opérateurs.

3.1.2.2. Construction d'un niveau grossier par « analyse matricielle »

Les opérateurs de prolongement et de restriction sont définis à partir des termes a_{ij} de la matrice A_h , en mettant en évidence les forts couplages entre degrés de liberté.

Soit N_i l'ensemble des voisins du degré de liberté i :

$$\forall i \in \{1, \dots, N_{ddl}\}, N_i = \{j \in \{1, \dots, N_{ddl}\} / a_{ij} \neq 0\} \quad (3-9)$$

avec N_{ddl} le nombre de degrés de liberté du système.

Soit S_i l'ensemble des points auxquels le degré de liberté i est fortement connecté :

$$\forall i \in N_i, \quad S_i = \left\{ j \in \{1, \dots, N_{ddl}\} \mid |a_{ij}| \geq \varepsilon \max_{k \neq i} |a_{ik}| \text{ avec } \varepsilon \in]0,1[\right\} \quad (3-10)$$

A partir de ces deux ensembles N_i et S_i , on applique une procédure itérative pour construire l'ensemble C des degrés de liberté fortement couplés. C représente l'ensemble des degrés de liberté du niveau grossier. Des exemples de méthodes de décomposition de domaine utilisées pour déterminer cet ensemble sont présentés dans [Lallemand92] et [Iwamura et al.03]. Une fois ce sous-ensemble C déterminé, on définit l'opérateur de prolongement P de la manière suivante :

$$(u_h)_i = (Pu_H)_i = \begin{cases} (u_H)_i & \text{si } i \in C \\ \sum_{j \in C} \frac{a_{ij}}{a_{ii}} (u_H)_j & \text{si } i \notin C \end{cases} \quad (3-11)$$

L'opérateur de restriction R est en général défini par la relation de Ritz-Galerkin [Janka02] qui définit un opérateur comme la transposée de l'autre. On prend ici :

$$R = P^T \quad (3-12)$$

La matrice grossière est généralement définie par la relation de Galerkin :

$$A_H = RA_hP \quad (3-13)$$

On a ainsi construit tous les éléments nécessaires à la méthode multigrilles. Pour plus de détails sur les méthodes algébriques, le lecteur peut se reporter aux ouvrages de référence que sont [Ruge et al.87] et [Stüben01]. Un exemple d'application de cette méthode dans le cadre des éléments finis pour la résolution des équations d'Euler et de Navier Stokes est présenté dans [Okusanya02].

3.1.2.3. Construction d'un niveau grossier par agglomération

La méthode présentée ci-dessous est tirée de [Janka02]. Elle fait partie de la famille des multigrilles algébriques puisque seul le maillage initial est nécessaire, les grilles grossières étant déterminées par agglomération de volumes de contrôle (pour une formulation volumes finis) ou par agrégation des degrés de liberté (pour une formulation éléments finis). Cette méthode a été historiquement introduite pour la résolution de l'équation d'Euler dans le cadre des volumes finis, puis appliquée à des problèmes de convection et diffusion. Elle est actuellement utilisée pour la résolution des équations de Navier-Stokes pour des écoulements turbulents compressibles et visqueux, avec des discrétisations volumes finis. Pour appliquer cette stratégie dans le cadre des éléments finis, il suffit de considérer des agrégats nodaux du maillage en lieu et place des volumes de contrôle agglomérés. Un agrégat est un ensemble de degrés de liberté correspondants à des nœuds (au sens éléments finis) ou des volumes (au sens volume finis) connexes du maillage. Chaque agrégat correspond à un « nœud » (ensemble de degrés de liberté) du niveau grossier.

On se place dans le contexte d'un algorithme multigrilles à plus de deux niveaux, mais les explications se limitent au passage d'une grille fine à une grille grossière. Soit C_h l'ensemble de tous les nœuds éléments finis du maillage fin : $C_h = \{1, \dots, n_h\}$. Le but est de définir un algorithme de type *glouton* pour partitionner l'ensemble des nœuds C_h en agrégats

nodaux $\{C_h(p)\}, p = 1, \dots, n_H$. Pour chaque nœud grossier p , $C_h(p)$ définit un agglomérat de nœud fin. La figure 3-3 montre un exemple d'agrégats nodaux, vus comme des sous-graphes du maillage. On peut voir le maillage initial, puis la construction des agrégats nodaux pour les 5 autres niveaux de grilles du calcul.

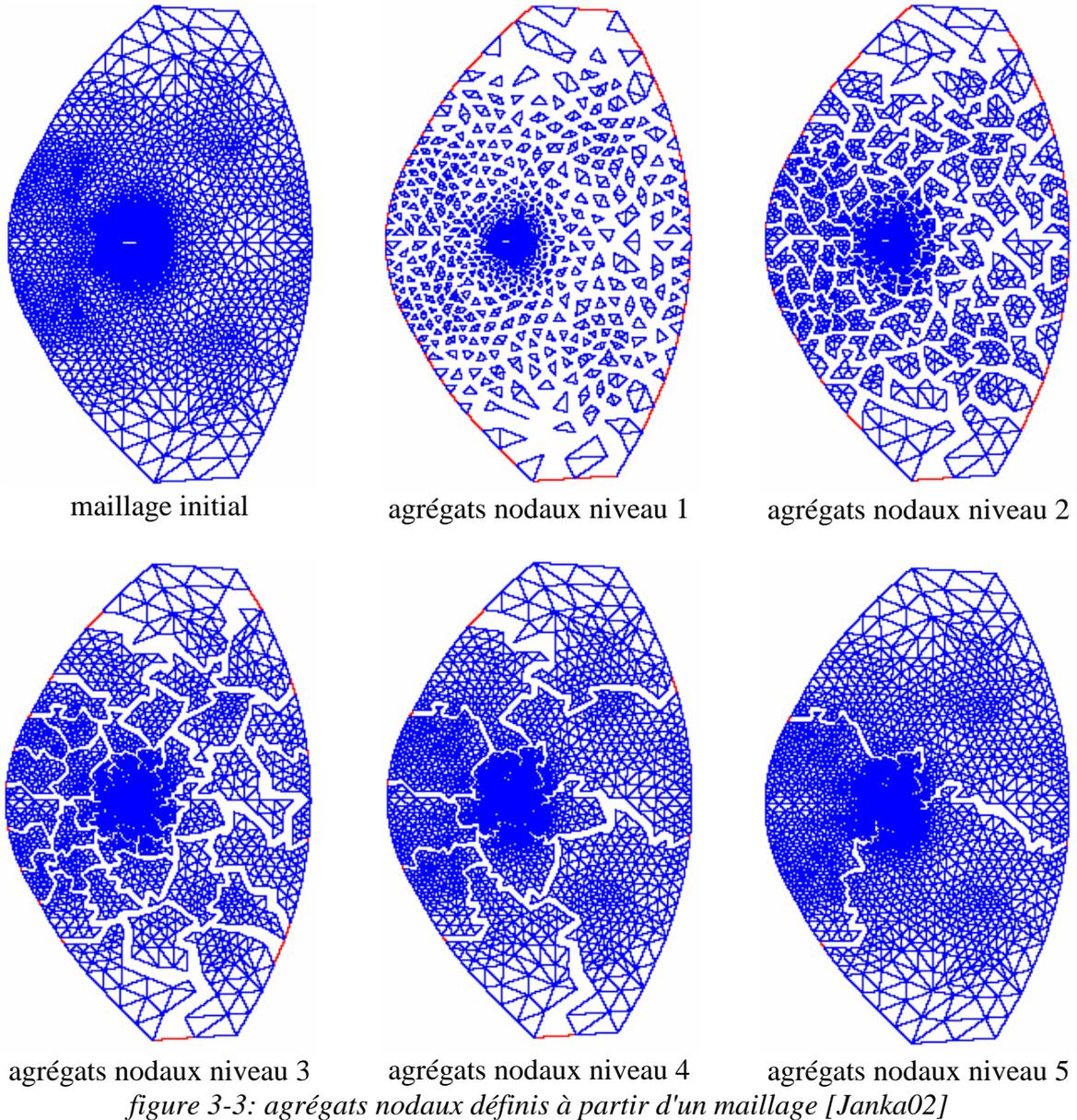


figure 3-3: agrégats nodaux définis à partir d'un maillage [Janka02]

Une fois la construction des agrégats effectuée, on définit l'opérateur de prolongement non-lissé \tilde{I}_H^h comme une injection canonique :

$$\forall (i, p) \in (\{1, \dots, n_h\}, \{1, \dots, n_H\}) \quad (\tilde{I}_H^h)_{i,p} = \begin{cases} 1 & \text{si } i \in A_h(p) \\ 0 & \text{si } i \notin A_h(p) \end{cases} \quad (3-14)$$

Pour compléter cette opération, on définit un lisseur de prolongement (positif semi défini) $S_h = s_r(A_h)$ où s_r est un polynôme de degré r sur le spectre de A_h . Dans [Janka02], l'auteur

prend par exemple : $S_h = I - \frac{1}{\rho(A_h)} A_h$ dans le cas du calcul de la partie diffusive de l'équation de Poisson ou pour un problème de Laplacien, $\rho(A_h)$ étant une estimation du rayon spectral de A_h . Le rôle du lisseur est d'imposer une plus grande régularité aux fonctions de l'espace grossier. On définit ainsi la matrice de l'espace grossier par :

$$A_H = (S_h \tilde{I}_H^h)^T A_h S_h \tilde{I}_H^h \quad (3-15)$$

On constate que l'on a construit le problème sur l'espace grossier en ne se servant que du maillage de calcul, d'où la classification en multigrilles algébriques. Les niveaux grossiers ne sont pas des maillages mais des agglomérats, construits de manière géométrique (on peut également les construire d'une manière purement algébrique).

3.1.3. Multigrilles géométriques

3.1.3.1. Principe

Dans l'approche géométrique, les différents niveaux de grilles sont apparentés à des maillages, le but étant de construire différentes grilles de taille de plus en plus grossière. La principale difficulté liée à ce type de méthode est la construction des différents maillages grossiers. Cette difficulté est d'autant plus grande si les maillages sont non structurés. Une fois ces maillages grossiers réalisés, on utilise la plupart du temps les relations de Ritz-Galerkin (3-12) et Galerkin (3-13) pour définir une matrice grossière [Adams98], [Mocellin99], [Feng et al.98].

Pour de plus amples informations sur les multigrilles géométriques, le lecteur peut se reporter à [Wesseling et al.01]. On rencontre plusieurs techniques de construction des différents maillages dans la littérature dont voici une liste non exhaustive mais représentative. D'autres approches et références peuvent être trouvées dans la partie 4.3 de [Mavripilis97].

3.1.3.2. Grilles indépendantes

La méthode consiste à construire différents maillages de manière totalement indépendante, le but étant d'avoir un nombre de nœuds décroissant [Mavripilis95], [Zang et al.95]. On construit ensuite les opérateurs de restriction et de prolongement par l'intermédiaire d'une méthode d'interpolation des nœuds du maillage fin dans les éléments du maillage grossier (restriction), puis des nœuds grossiers dans le maillage fin (prolongement) (figure 3-4). De plus, les différents maillages se doivent de décrire la même frontière. Cette technique est adoptée notamment dans [Feng et al.98] (figure 3-5) dans le cadre de calcul en élasto-plasticité.

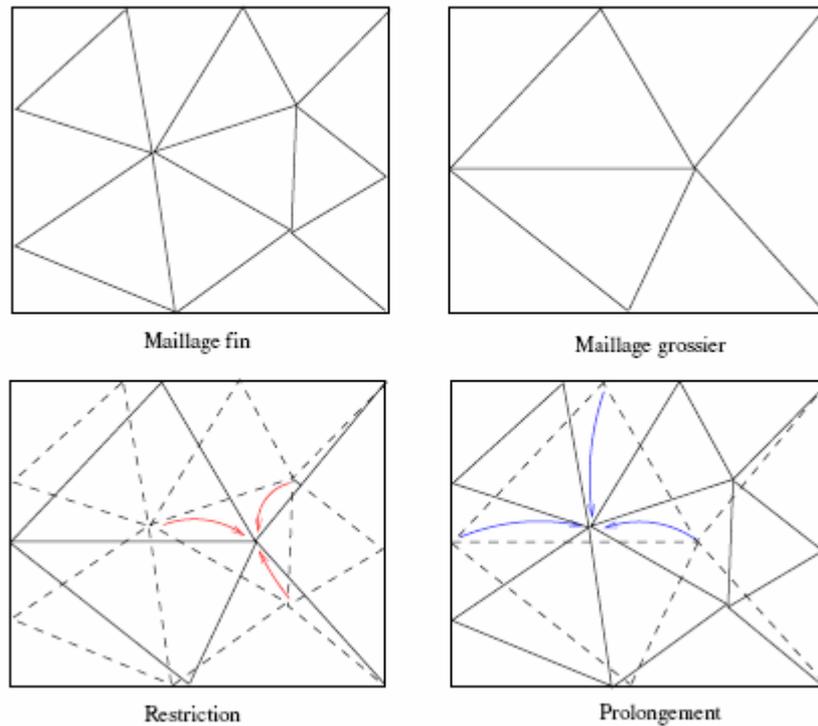


figure 3-4: opérateurs de prolongement et restriction sur deux maillages indépendants

Le principal inconvénient de cette approche est la complexité des opérateurs du au fait que deux grilles successives n'ont aucune corrélation entre elles. Cette hypothèse induit un temps de calcul non négligeable durant les communications entre différents niveaux. Dans le cadre du forgeage, cet inconvénient est encore plus pénalisant à cause des remaillages fréquents. En effet il faudrait, à chacune de ces étapes, recalculer de nouveaux maillages grossiers puis effectuer les interpolations, ce qui alourdirait d'autant plus les temps de calcul.

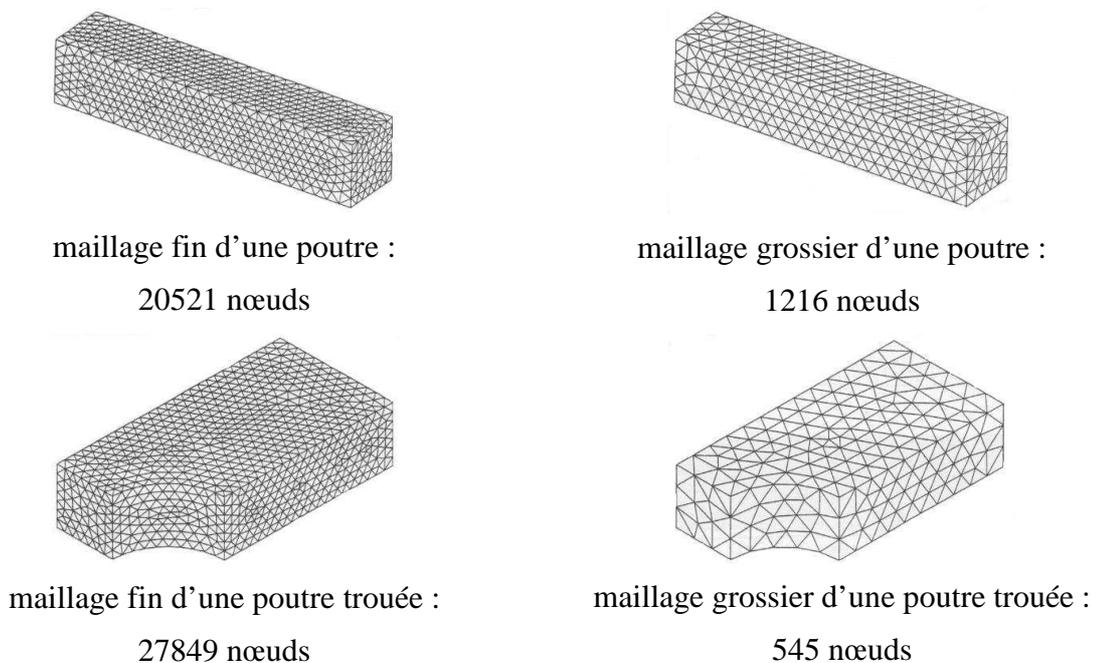


figure 3-5: déraffinement de maillage ; maillages indépendants [Feng et al.98]

3.1.3.3. Grilles emboîtées par éléments

Le principe ici est de partir du maillage grossier et de l'enrichir pour obtenir le maillage fin [Mocellin99], [El Hadj87]. Une approche consiste à raffiner chaque élément du maillage pour créer un maillage plus fin, chaque tétraèdre donnant 8 tétraèdres (figure 3-6) par exemple. L'avantage d'une telle méthode est l'obtention de maillages qui sont emboîtés par élément, ce qui réduit grandement le temps de calcul des opérateurs de prolongement et de restriction. Ces opérateurs sont très simples et peu coûteux à l'utilisation, en plus d'être précis. Cependant, le principal désavantage de cette méthode est que la procédure de construction des différents niveaux de grille part du maillage grossier qui n'est pas le maillage de calcul. Ceci est incompatible avec une utilisation industrielle où l'on demande à l'utilisateur de fournir le maillage de calcul (le maillage fin). De même, avec cette méthode de déraffinement, on ne peut empêcher la génération d'éléments de qualité moindre en raffinant [Muttin et al.95], ce qui peut entraîner la dégénérescence du maillage de la grille fine et qui limite le nombre de grille utilisable.

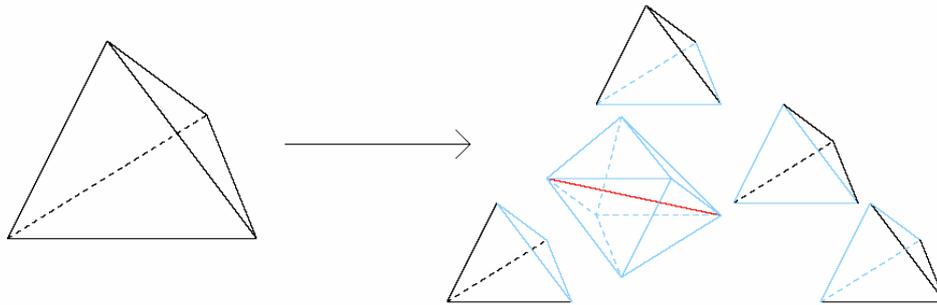


figure 3-6: raffinement d'un tétraèdre [Mocellin99]

Enfin, la description géométrique de la pièce sur un maillage fin n'est pas correcte, sauf cas très particulier de géométrie non courbe. On doit donc mettre en place une méthode de projection des nœuds de la surface pour palier ce problème, comme on le voit sur la figure 3-7, mais celle-ci introduit encore une perte de qualité des éléments. Il faut ensuite régulariser le maillage en volume. Cette méthode de projection ne marche pas dans les cas de géométries vraiment complexes, comme on en rencontre en forgeage.

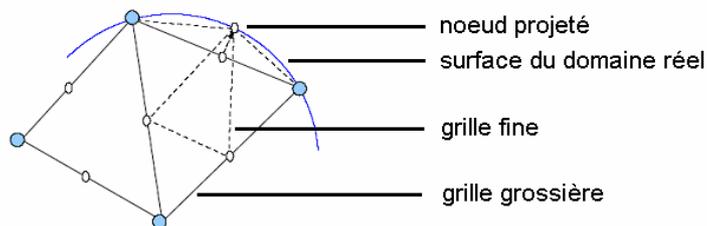


figure 3-7: projection de noeuds fins en frontière pour avoir un maillage représentatif [Mocellin99]

3.1.3.4. Grilles emboîtées par nœuds

On présente une approche de déraffinement de maillage automatique permettant d'obtenir des maillages emboîtés par nœuds, ce qui signifie que tous les nœuds du maillage grossier sont également des nœuds du maillage fin.

L'approche en dimension 2 du déraffinement par un algorithme de type MIS (Maximal Independent Set) a été développée indépendamment dans [Guillard93] et [Chan et al.94]. L'idée sous-jacente à cette méthode est basée sur l'utilisation d'un algorithme de type Delaunay-Voronoi de triangulation de nuages de points. Elle s'applique dans le cadre des éléments finis, sur des maillages triangulaires. La méthode consiste à définir un sous-ensemble des nœuds fins qui comporte moins de nœuds et que l'on triangularise ensuite par la méthode de Delaunay-Voronoi, en respectant le plus fidèlement possible la géométrie du maillage initial.

La liste des nœuds fins retenus est construite par un algorithme de type MIS (Maximal Independent Set), que l'on traduit par un *ensemble maximal indépendant* (de points ou de nœuds dans notre cas). Soit $G = (V, E)$ le graphe associé à notre maillage, avec V la liste des nœuds et E la table de connectivité. Un ensemble indépendant de points est un ensemble $I \subset V$ dans lequel aucun point n'est adjacent (voisin) avec un autre : $\forall (v, w) \in I^2 \quad (v, w) \notin E$. Un ensemble maximal indépendant (MIS) est un ensemble indépendant qui ne peut être contenu dans aucun autre ensemble indépendant. On peut noter qu'il n'y a généralement pas unicité des MIS.

L'algorithme de construction de l'ensemble des nœuds grossiers dépend évidemment de l'ordonnement des nœuds au début. On définit une liste ordonnée contenant tous les nœuds fins que l'on parcourt pour sélectionner les nœuds grossiers. On place tout d'abord les nœuds frontières en tête de liste de manière à en supprimer le moins possible ce qui permet de conserver la géométrie du maillage fin. On les ordonne entre eux par la valeur absolue de l'angle qu'ils forment avec leurs nœuds voisins frontières, afin de traiter en premier les nœuds traduisant un changement local brusque de géométrie et pour conserver les coins s'il y en a. Enfin on applique une méthode de type glouton afin de supprimer les nœuds voisins du nœud traité. Cet algorithme est présenté dans le tableau 3-2.

tableau 3-2 : algorithme MIS de partition de l'ensemble des nœuds fins en 2D

1. définition d'une liste L contenant l'ensemble des nœuds fins ordonnée de telle sorte que les nœuds appartenant à la frontière du domaine apparaissent en premier. Ces nœuds frontières sont eux-mêmes rangés dans l'ordre inverse de la valeur absolue de l'angle formé avec leurs deux voisins frontières.
 2. pour chaque nœud de la liste L faire :
 - si le nœud a déjà été supprimé, aller au nœud suivant de la liste L
 - sinon, ajouter ce nœud à la liste des nœuds grossiers et supprimer tous ses voisins de la liste L
-

Cet algorithme permet de créer un sous-ensemble de nœuds du maillage fin (constituant les nœuds du maillage grossier), que l'on maille par un algorithme de Delaunay-Voronoi qui n'est pas développé ici. On obtient ainsi un maillage grossier emboîté par nœuds vis-à-vis du maillage dont il est issu.

L'extension de l'algorithme précédent pour des maillages tridimensionnels a été développée dans [Adams et al.00]. La seule modification vis-à-vis de l'algorithme 2D présentée ci-dessus est la manière d'ordonner la liste de nœuds. Une fois cette liste classée, on applique l'algorithme du tableau 3-2 .

L'auteur considère trois types différents de nœuds frontières : les nœuds en surface, les nœuds sommets et les coins, visibles sur la figure 3-8.

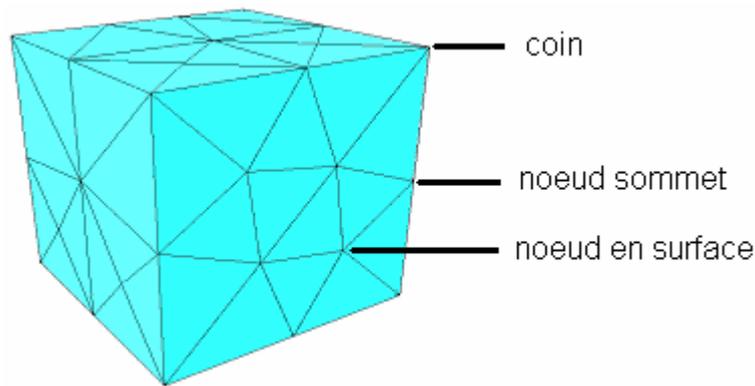


figure 3-8: classification des nœuds dans l'algorithme MIS

Pour identifier ces différents nœuds, il faut déterminer la liste des surfaces frontières de la pièce. On considère donc les faces frontières de la pièce (et non des éléments du maillage).

- si un nœud appartient à une seule surface frontière, c'est un nœud en surface ;
- si un nœud appartient à deux surfaces frontières, c'est un nœud sommet ;
- si un nœud appartient à plus de deux surfaces frontières, c'est un coin.

Une fois ces distinctions topologiques effectuées, on classe les nœuds dans la liste : tout d'abord les coins, puis les nœuds sommets, puis les nœuds en surface et enfin les nœuds internes. On peut également ordonner les nœuds à l'intérieur des différentes classes topologiques et choisir de modifier quelque peu la définition de la table de connectivité du graphe, afin d'essayer de conserver au mieux la géométrie globale du maillage fin. Le maillage de la grille grossière est comme précédemment effectué par un algorithme de type Delaunay-Voronoi à partir de la liste des nœuds grossiers générée par la méthode précédente.

La figure 3-9 montre un exemple de déraffinement de maillage obtenu par cette méthode. On peut voir que le maillage de départ est fait à partir de quadrilatères tandis que les maillages grossiers sont construits avec des tétraèdres.

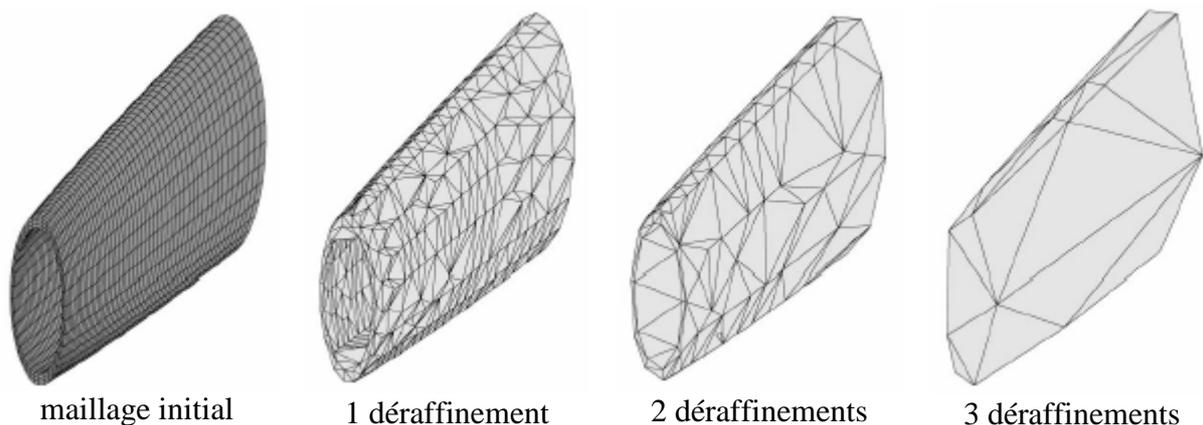


figure 3-9: déraffinement de maillage 3D par un algorithme MIS [Adams et al.00]

L'intérêt de cette méthode est qu'à partir d'un maillage non structuré donné, on construit de manière totalement automatique les niveaux de grilles grossières. Les maillages obtenus sont emboîtés par nœud, ce qui simplifie le calcul des opérateurs de transfert correspondant et qui minimise la diffusion d'information lors des différentes opérations de restriction et prolongement. Cependant, la principale limitation est qu'elle ne s'applique qu'à des

géométries relativement simples, dont on peut définir les faces frontières et qui peuvent être correctement maillées à partir d'une liste de nœuds par un algorithme de Delaunay-Voronoi.

3.1.4. Avantages et inconvénients des deux approches

Les deux approches présentent chacune des caractéristiques intéressantes. Les multigrilles algébriques ont l'avantage d'être totalement indépendantes du maillage, donc de sa complexité. Elles sont donc très facilement automatisables pour pouvoir se comporter en boîte noire. Il est donc plus facile de les incorporer dans un code industriel. Cependant, l'expérience montre qu'elles sont généralement moins performantes que des multigrilles géométriques correctement mises en œuvre [Lallemand92].

Les méthodes algébriques sont basées sur la comparaison de termes matriciels. Or on utilise dans le logiciel Forge3® une méthode de pénalisation pour la gestion du contact qui implique des différences d'ordre de grandeur dans ces termes. Il faudrait analyser si c'est un point positif ou non pour les multigrilles algébriques.

Dans une approche géométrique, la principale difficulté réside dans l'obtention des maillages grossiers. Dans le cadre d'un projet industriel, le maillage de calcul doit être le maillage fourni par l'utilisateur, ainsi l'option de raffinement de maillage grossier n'est pas utilisable ici. On a également pu constater que l'utilisation de maillages indépendants n'était également pas souhaitable dans notre étude du fait des remaillages fréquents. On se concentre donc sur les méthodes de déraffinement de maillage permettant d'obtenir des maillages emboîtés par nœuds.

Le but de ce travail est de concevoir une méthode multigrilles tirant partie des avantages de chaque méthode, à savoir la rapidité et la robustesse de l'approche géométrique avec le côté transparent et automatique de l'approche algébrique. On développe donc par la suite une méthode de déraffinement automatique de maillage basée sur une approche de remaillage locale. Cette méthode s'inscrit dans le cadre d'un solveur multigrilles géométrique autonome, qui utilise les relations de Ritz-Galerkin (3-12) et Galerkin (3-13) pour la définition d'une matrice grossière.

Ces développements sont intégrés dans le logiciel en utilisant la bibliothèque de calcul PETSc. Nous détaillons dans la partie suivante notre approche, avec dans un premier temps une présentation de la librairie PETSc, puis de la méthode de déraffinement automatique et enfin des différents choix effectués pour le lissage et la résolution sur la grille grossière.

3.2. La méthode développée

3.2.1. La librairie PETSc

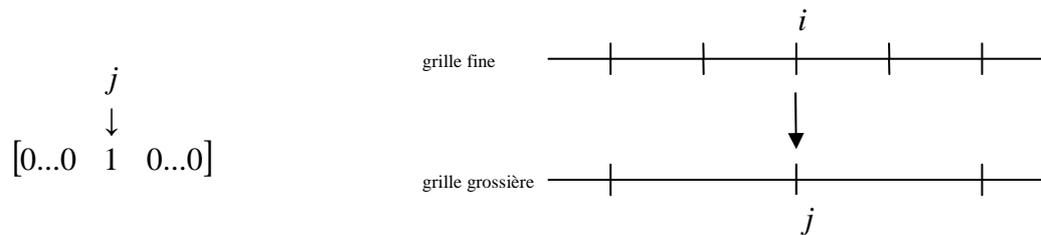
3.2.1.1. Présentation

PETSc (Portable, Extensible Toolkit for Scientific Computation) [Balay et al.97] est une librairie de calcul scientifique, basée sur les librairies BLAS et LAPACK. Développée en C++, elle est entièrement parallélisée et optimisée pour le calcul vectoriel et matriciel, ainsi que la résolution de systèmes linéaires. Elle est totalement gratuite et ses sources sont disponibles et modifiables.

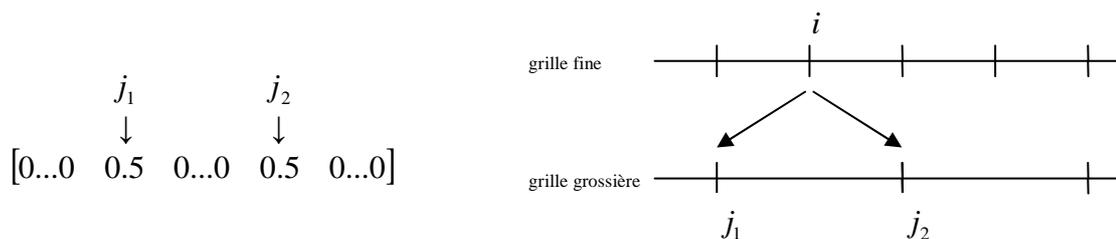
Les raisons de son emploi dans notre contexte sont multiples. Elle présente tout d'abord des performances très intéressantes en ce qui concerne le calcul matriciel, ce qui est primordial dans ce travail eu égard à la relation de Galerkin (3-13) utilisé qui est un double produit matriciel. Ces performances sont d'ailleurs explicitement testées dans le paragraphe suivant. Un autre avantage de cette bibliothèque est la présence d'un préconditionneur multigrilles, ce qui facilite l'intégration des développements dans le logiciel Forge3. Enfin, l'aspect hautement parallèle de la librairie est appréciable dans l'optique d'une parallélisation de ce travail.

3.2.1.2. Performances

On compare ici les temps de calcul de la matrice grossière par la relation de Galerkin (3-13) sur différents cas avec d'une part l'utilisation de la librairie PETSc et d'autre part une méthode implémentée dans Forge3. Cette méthode de produit matriciel a été développée dans [Mocellin99] spécialement pour la relation de Galerkin, dans le cadre d'une méthode d'enrichissement de maillage. Avec des maillages emboîtés par élément, l'opérateur de restriction est très simple puisque composé uniquement de valeur 1 ou 1/2. Si le noeud i du maillage fin est également un noeud grossier, alors la ligne i de l'opérateur de restriction est :



Si le noeud fin i est situé au milieu de l'arrête composée des noeuds grossiers j_1 et j_2 , alors la ligne i de l'opérateur de restriction est :



L'opérateur de prolongement est défini comme étant la transposée de l'opérateur de restriction. Ces matrices sont donc d'une creusité extrême, avec seulement une ou deux valeurs non nulles par ligne (pour l'opérateur de restriction) ou par colonne (pour l'opérateur de prolongement). L'algorithme implémenté dans Forge3 prend en compte la spécificité de ces matrices et est optimisé en conséquence.

Le tableau 3-3 présente les temps de calcul des deux méthodes sur trois différents exemples. Les résultats montrent que la librairie PETSc présente des temps de calcul de l'ordre de dix fois plus petits.

	Taille de la matrice fine	Temps CPU avec Forge3	Temps CPU avec PETSc
Système 1	22466 noeuds	6,2	0,7
Système 2	47525 noeuds	12,7	1,5
Système 3	48755 noeuds	20,1	2,0

tableau 3-3 : comparaison des temps de calcul (en seconde) de $A_H = RA_h P$

3.2.1.3. Préconditionnement par multigrilles

Historiquement, les méthodes multigrilles ont été développées en tant que solveur linéaire itératif. Cependant l'expérience montre qu'elles sont encore plus efficaces en tant que préconditionneur d'une méthode de Krylov [Vuik et al.95] et sont la plupart du temps employées comme telles ([Iwamura et al.03], [Adams02], [Pereira et al.06]). On utilise ainsi les multigrilles comme préconditionneur d'une méthode de résidu conjugué, en choisissant de définir une étape de preconditionnement par uniquement un cycle de la méthode multigrilles. L'algorithme du résidu conjugué preconditionné tel qu'il est implémenté dans PETSc (et donc utilisé ici) est décrit dans le tableau 3-4, où M est la matrice de preconditionnement.

Dans toute la suite de l'exposé, on nomme « solveur multigrilles » ce solveur résidu conjugué preconditionné par un V-cycle multigrilles.

tableau 3-4 : algorithme de Résidu conjugué préconditionné implémenté dans PETSc

Initialisation:

$$r_0 = b$$

$$p_0 = M^{-1}r_0$$

$$z_0 = Ap_0$$

$$k = 0$$

Tant que ($k < nbIteMax$ & non convergence) {

$$\alpha_k = \frac{(M^{-1}r_k, AM^{-1}r_k)}{(z_k, M^{-1}z_k)}$$

$$x_{k+1} = x_k + \alpha_k p_k$$

$$M^{-1}r_{k+1} = M^{-1}r_k - \alpha_k M^{-1}z_k$$

$$r_{k+1} = r_k - \alpha_k z_k$$

$$\beta_k = \frac{(M^{-1}r_{k+1}, AM^{-1}r_{k+1})}{(M^{-1}r_k, AM^{-1}r_k)}$$

$$p_{k+1} = M^{-1}r_{k+1} + \beta_k p_k$$

$$z_{k+1} = AM^{-1}r_{k+1} + \beta_k z_k$$

$$k = k + 1$$

} fin tant que

3.2.2. Déraffinement automatique de maillage

3.2.2.1. Méthode

Le but est de construire, à partir d'un maillage donné, un maillage grossier définissant la même géométrie, de manière automatique et sans aucune intervention extérieure. On utilise pour cela une technique de déraffinement basée sur un algorithme itératif de suppression de nœuds combinée avec une procédure de remaillage local [Carte et al.00]. L'ensemble de la méthode est construit à partir du mailleur automatique MTC® développé au CEMEF [Coupez00].

A partir d'un maillage tridimensionnel M_h et d'un coefficient de déraffinement C , on souhaite construire un maillage grossier qui respecte la géométrie initiale et qui, localement, a une taille de maille C fois plus grande. M_h est défini par l'ensemble de ses nœuds I_h et de ses tétraèdres T_h . La procédure de déraffinement est la suivante : pour tous les nœuds $i \in I_h$, on calcule la longueur moyenne $l_h(i)$ des arêtes qui lui sont connectées. On cherche ainsi à obtenir un maillage grossier M_H tel que pour tous les nœuds $i \in I_H$, $l_H(i) \approx Cl_h(i)$. On définit C_R le facteur de déraffinement :

$$C_R = \frac{\text{nombre de noeuds du maillage fin}}{\text{nombre de noeuds du maillage grossier}} \quad (3-16)$$

Dans le cadre d'un déraffinement isotrope, on obtient la relation $C_R = C^d$, avec d la dimension du problème. Pour des raisons de robustesse et d'efficacité de la méthode multigrilles, le facteur C doit être proche de 2 [Guillard93]. On cherche donc à obtenir un rapport de 8 entre le nombre de nœuds du maillage initial et celui du maillage grossier résultant.

Concernant M_H , on définit $T_H(i)$ l'ensemble des tétraèdres contenant le nœud i , $\mathcal{E}_H(i)$ l'ensemble des nœuds de $T_H(i)$ (i exclu) et $F_H(i)$ l'ensemble des faces extérieures de $T_H(i)$ ne contenant pas i . On utilise l'algorithme de suppression de nœuds décrit dans le tableau 3-5.

tableau 3-5 : algorithme de déraffinement automatique de maillage

1. **Initialisation** :

- pour chaque noeud $i \in I_h$, calcul de $l_c(i) = Cl_h(i)$
- $M_H = M_h$; $T_H = T_h$ et $I_H = I_h$

2. pour chaque noeud i successivement dans I_H , calcul de $l_H(i)$ la longueur moyenne des arêtes connectées à i

3. si $l_H(i) \geq l_c(i)$ alors

$i \leftarrow i + 1$ et aller à l'étape 3

sinon

- Remailler le volume défini par $F_H(i)$ en utilisant seulement les noeuds de $\mathcal{E}_H(i)$.
Soit $T_{new}(i)$ l'ensemble des tétraèdres de cette nouvelle triangulation (qui ne contient pas i)
 - si $T_{new}(i)$ est acceptable alors
 - $I_H = I_H - \{i\}$
 - $T_H = T_H - T_H(i) + T_{new}(i)$
 - $i \leftarrow i + 1$ et aller à l'étape 3
 - sinon
 - $i \leftarrow i + 1$ et aller à l'étape 3
-

A la fin de la procédure, on obtient une nouvelle triangulation dont les nœuds forment un sous ensemble de l'ensemble des nœuds du maillage fin et où la longueur moyenne des arêtes est approximativement $l_H(i)$. L'étape 4 de l'algorithme concernant le remaillage de la cavité définie par $F_H(i)$ après suppression de i est décrite sur la figure 3-10. Afin de ne pas trop perdre en régularité, il faut choisir une bonne triangulation de la cavité. Cependant, on ne peut se permettre de tester toutes les triangulations possibles. On se contente donc des triangulations obtenues en reliant un point à toutes les faces du polyèdre, et on garde celle dont les tétraèdres sont les plus proches du tétraèdre équilatéral [Janka et al.03].

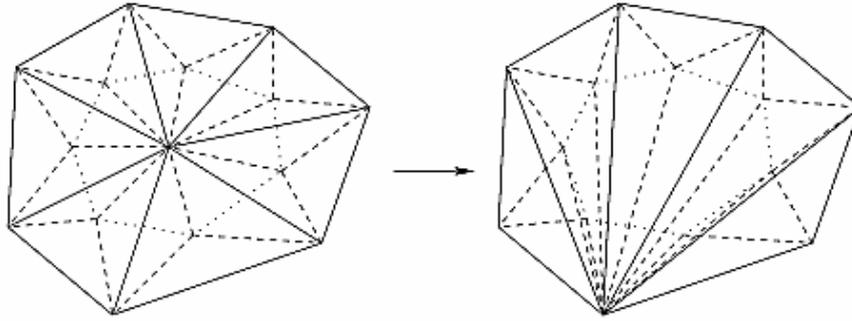


figure 3-10: remaillage d'une cavité

Lorsque le nœud à supprimer appartient à la surface du maillage, cela entraîne un changement de la géométrie. Pour décider d'effectuer cette suppression ou non, on définit un critère de tolérance sur le changement de volume entre le maillage fin et le maillage grossier. Si la suppression du nœud en surface n'engendre pas une variation de volume supérieure à la tolérance, elle est acceptée. Ce critère sera explicité par la suite.

Comme on se contente de supprimer des nœuds, le maillage grossier obtenu est emboîté par nœuds dans le maillage fin, le calcul de l'opérateur de restriction reliant ces deux maillages sera plus simple, donc a priori moins coûteux et plus précis.

La figure 3-11 illustre les résultats de cet algorithme sur le cas de forgeage d'une bielle en fin de mise en forme. On constate que le facteur de déraffinement ($C_R = 8,4$) est proche de sa valeur optimale (8), même si la géométrie est complexe avec notamment une bavure sur tout le tour de la pièce. Le maillage de calcul étant très fin, on ne constate pas problème pour le déraffiner dans le volume. La figure 3-12 présente un autre exemple de déraffinement sur une bague en fin de roulage, avec également un facteur de déraffinement satisfaisant ($C_R = 7,1$). Il est cependant plus faible que la valeur optimale, car le maillage déraffiné ne contient plus qu'un ou deux éléments dans l'épaisseur. La suppression de d'avantage de nœuds entraînerait une trop grande perte de volume de la géométrie. La figure 3-13 montre le maillage initial et le maillage déraffiné d'un quart de pièce forgée par pilonnage ($C_R = 7,6$).

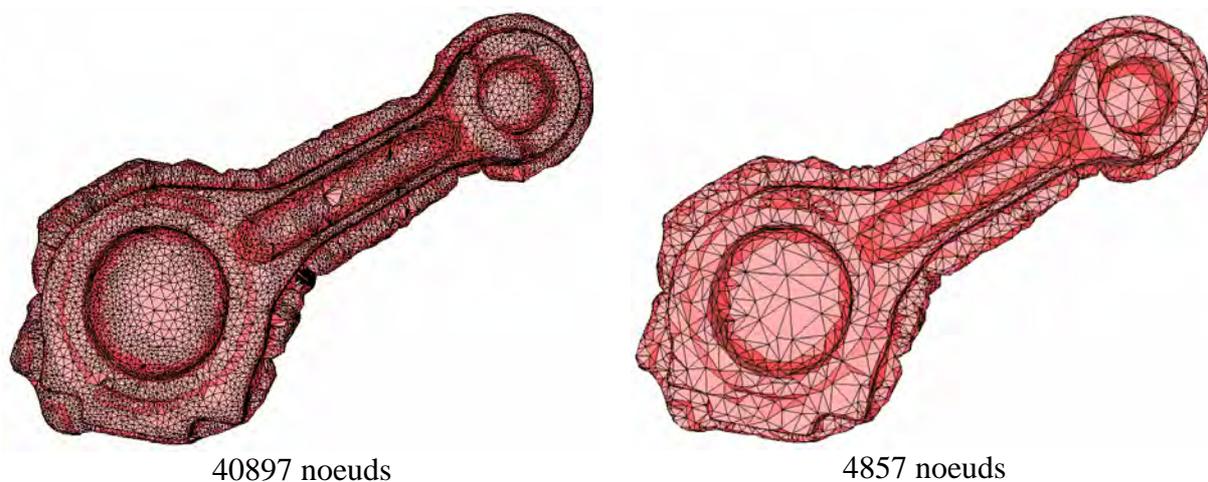


figure 3-11: déraffinement du maillage d'une bielle en fin de forgeage ($C_R = 8,4$)

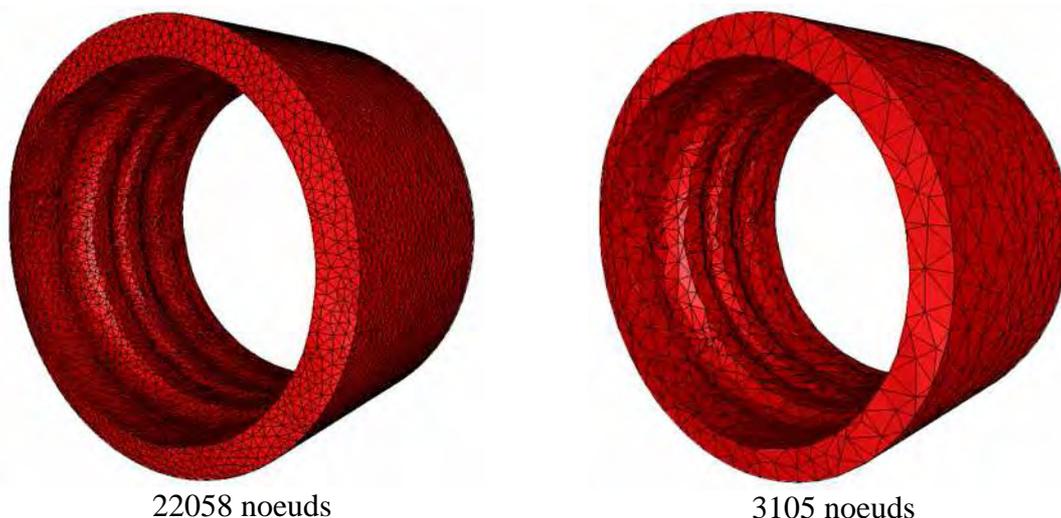


figure 3-12: déraffinement du maillage d'une bague en fin de roulage ($C_R = 7,1$)

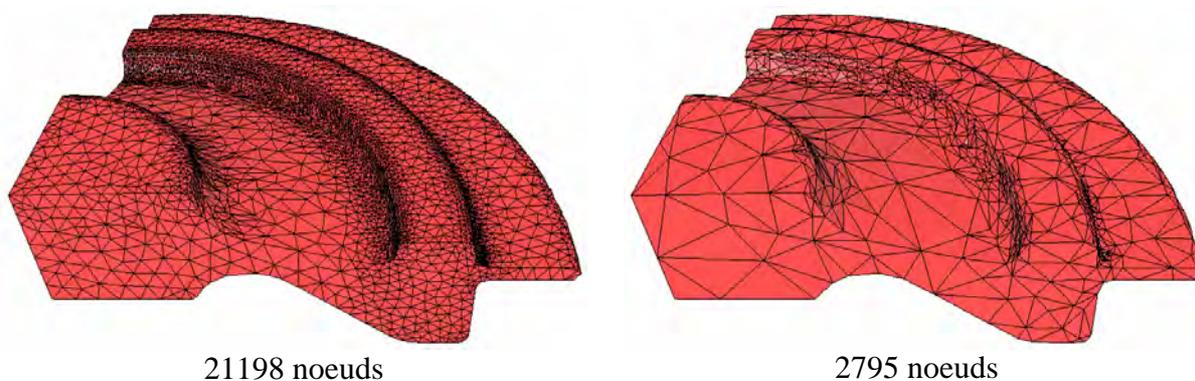


figure 3-13: déraffinement du maillage d'un quart de pièce ($C_R = 7,6$)

3.2.2.2. Construction d'un opérateur de restriction

On rappelle que l'on utilise la relation de Galerkin (3-13) pour construire la matrice grossière. On utilise la relation de Ritz-Galerkin (3-12) pour définir l'opérateur de prolongement comme étant la transposée de l'opérateur de restriction que l'on va évaluer ($P = R^T$).

La seule entité à construire pour le passage d'un niveau fin à un niveau grossier est ainsi l'opérateur de restriction défini par la matrice R . Il est calculé par interpolation linéaire des nœuds du maillage fin dans les éléments du maillage grossier. La figure 3-14 décrit cette interpolation en 2D.

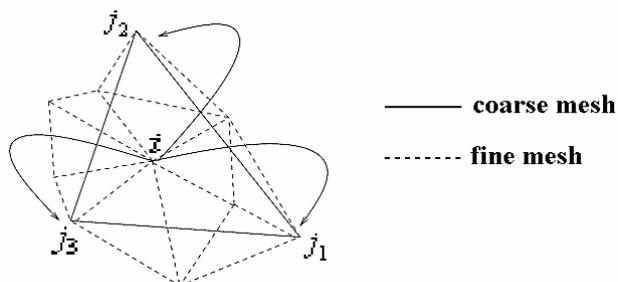


figure 3-14: interpolation d'un nœud fin dans un élément grossier (en 2D)

Pour chaque nœud i du maillage fin, on cherche l'élément grossier (j_1, j_2, j_3, j_4) qui contient i puis on calcule les coordonnées barycentriques $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$ de i dans cet élément. On définit alors la $i^{\text{ème}}$ ligne de la matrice de restriction :

$$\begin{cases} \forall k \in \{1, 2, 3, 4\} & R(i, j_k) = \alpha_k \\ \forall k \notin \{j_1, j_2, j_3, j_4\} & R(i, k) = 0 \end{cases} \quad (3-17)$$

Si le nœud grossier i à interpoler est également un nœud j du maillage fin, on a dans la $i^{\text{ème}}$ ligne de la matrice de restriction :

$$\begin{cases} R(i, j) = 1 \\ \forall k \neq j & R(i, k) = 0 \end{cases} \quad (3-18)$$

La matrice de restriction est donc d'une extrême creusité avec seulement une à quatre valeurs non nulles par ligne. Le double produit matrice - matrice nécessaire à la définition de la matrice grossière n'est cependant pas coûteux en temps de calcul, grâce à l'emploi de la librairie PETSc. Ce dernier point a été vérifié numériquement dans la partie précédente.

3.2.3. Le lissage et la correction

La construction des opérateurs ainsi que de la matrice grossière étant présentées, il ne reste plus qu'à choisir les différents paramètres de la méthode afin que celle-ci soit complète et prête à être utilisée. On détaille dans la suite le choix des lisseurs et du nombre de pas de lissage ainsi que le choix de la méthode de résolution sur la grille la plus grossière. La résolution de tous les systèmes linéaires étudiés par la suite est réalisée à une précision relative de 10^{-6} , qui correspond à la précision que l'on demande lors des résolutions effectuées dans le logiciel Forge3®.

3.2.3.1. Lissage

Le lisseur le plus utilisé dans la littérature ([Adams00], [Heys et al.05]) est une méthode itérative de Jacobi. Elle est extrêmement simple en plus d'être rapide, tout en présentant une bonne propriété de lissage des hautes fréquences. Après plusieurs tests de différentes méthodes itératives disponibles dans PETSc, il est effectivement apparu qu'un lisseur Jacobi par bloc de 4 (correspondant aux 4 degrés de liberté par nœuds) était la méthode la plus performante et la plus rapide, en faisant apparaître un très bon comportement avec le solveur principal en résidu conjugué.

Pour déterminer le nombre optimal d'itérations de pré et post lissage, on se place dans le cadre d'une méthode bigrille idéale, ce qui se traduit par l'emploi de deux niveaux de grilles et d'une résolution exacte sur la grille grossière. On ne s'intéresse ainsi qu'à l'influence du nombre de pas de pré et post lissages sur le nombre cumulé de lissages, le temps de calcul n'étant pas considéré ici. On utilise une méthode directe (factorisation LU) pour résoudre exactement le problème grossier.

Afin d'obtenir des résultats offrant au solveur multigrilles des propriétés de convergence optimales mais également de la robustesse, les tests sont effectués sur un système linéaire issu du forgeage d'une bielle en milieu de procédé (figure 3-15). Le lopin a un comportement

viscoplastique de type Norton-Hoff ($K = 1809$ MPa et $m = 0,14$) et le frottement est caractérisé par une loi de Coulomb ($\mu = 0,15$) limité Tresca ($\bar{m} = 0,3$). Le maillage fin contient 26659 nœuds, tandis que la grille grossière contient 2427 nœuds.

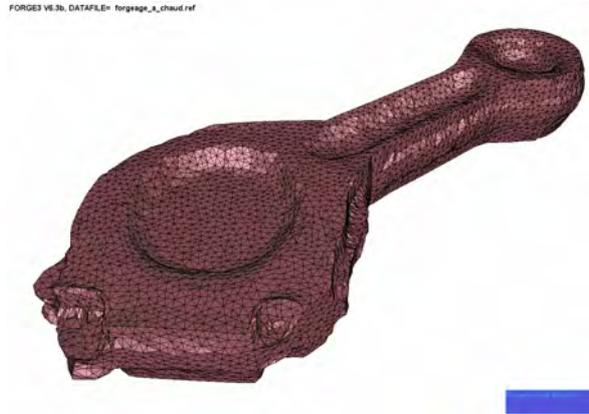


figure 3-15: bielle en milieu de forgeage

Le tableau 3-6 présente les résultats de ce test pour l'ensemble des combinaisons du nombre de pré et post lissage, avec un maximum de 3 itérations. Le facteur pertinent est le nombre total d'itérations de lissage, que l'on cherche à minimiser. On constate tout d'abord la présence de combinaison entraînant la non convergence, voir la divergence du solveur. Ceci se produit lorsque nous avons 1 ou 2 itérations de pré lissage sans post lissage, ou alors 1 de post lissage sans pré lissage. Cependant, la méthode converge avec 3 itérations de pré lissage sans post lissage, ou alors avec 2 ou 3 itérations de post lissage sans pré lissage, ce qui traduit une grande robustesse du solveur multigrilles.

On observe le minimum du nombre total d'itérations de lissage pour 1 itération de pré et de post lissage. En plus de minimiser le nombre total de pas de lissage, on constate également que cette combinaison présente le meilleur temps CPU. On fixe donc le nombre d'itération de pré et post lissage à 1. Dans la littérature, ce nombre varie en général entre 1 et 3 et il est effectivement souvent égal à 1.

Nombre d'itérations de pré lissage	Nombre d'itérations de post lissage	Nombre d'itérations du solveur global	Nombre total d'itérations de lissage	CPU
1	0	Non convergence	X	X
2	0	Non convergence	X	X
3	0	53	159	39.367
0	1	Divergence	X	X
1	1	30	60	25.727
2	1	48	144	36.933
3	1	43	172	37.784
0	2	131	262	67.978
1	2	38	114	32.006
2	2	27	108	28.651
3	2	39	195	38.505
0	3	39	117	32.487
1	3	52	208	42.861
2	3	41	205	39.827
3	3	25	150	31.445

tableau 3-6 : test de bigrilles idéales; influence du nombre de pré et post lissage

Afin de vérifier les propriétés de lissage des hautes fréquences de la méthode itérative de Jacobi par bloc, on l'applique en tant que méthode de résolution au système linéaire utilisé ci-dessus. On trace l'évolution de la norme du résidu en fonction du nombre d'itérations (figure 3-16). On constate une très forte décroissance du résidu sur la première itération puis une quasi stagnation de son évolution. Ce résultat confirme celui obtenu durant le test de bigrilles idéales, à savoir qu'une itération de pré et post lissage suffit à lisser l'erreur, les itérations suivantes n'apportant pas une réduction significative du résidu vis-à-vis de leur temps de calcul.

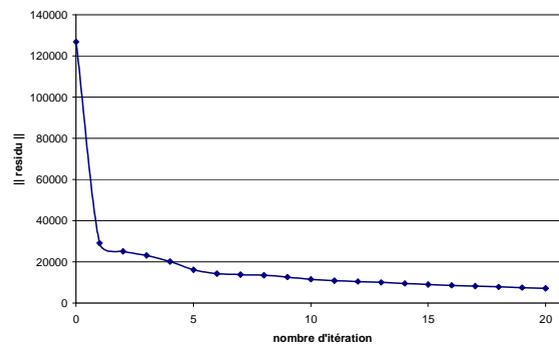


figure 3-16: propriété de lissage de la méthode Jacobi par bloc

3.2.3.2. Résolution du système grossier

Le calcul de la correction sur la grille la plus grossière doit être peu coûteux en temps tout en étant le plus précis possible. En règle générale, on utilise une méthode de résolution directe car le maillage grossier contient suffisamment peu de degrés de liberté pour que cette méthode soit compétitive par rapport à des méthodes itératives inexactes. On choisit ici une résolution directe par factorisation LU, en supposant que la méthode de déraffinement automatique est assez performante et robuste pour toujours produire des maillages suffisamment pauvres en nœuds. En effet, les méthodes directes sont les plus rapides sur un système contenant très peu de degrés de liberté, mais leur convergence asymptotique en $O(N^{7/3})$ pénalisent les temps de calcul énormément dès lors que l'on dépasse une limite de nombre de degrés de liberté. Cette hypothèse est discutée dans la partie suivante, avec notamment une étude sur cette limite.

3.3. Influence de la qualité de déraffinement

On s'intéresse ici à l'influence de l'algorithme de déraffinement de maillage sur les performances du solveur multigrilles. On étudie tout d'abord la valeur optimale du taux de déraffinement C_R , puis l'influence de la taille de la grille grossière sur le temps de résolution globale du solveur, ce qui nous permet de déterminer le nombre de niveaux de la méthode.

3.3.1. Influence du taux de déraffinement

Les calculs sont réalisés sur un système linéaire issu d'une simulation de roulage en cours de mise en forme, avec un comportement viscoplastique ($K = 2707$ MPa et $m = 0,1529$) et un frottement parfaitement glissant. Le maillage de calcul compte 12130 nœuds. On compare six méthodes deux grilles utilisant des taux de déraffinement différents ($C_R = 3,9$, $C_R = 6,0$, $C_R = 8,1$, $C_R = 17,2$, $C_R = 13,2$ et $C_R = 26,6$), afin de tester l'influence de ce taux sur les temps de résolution. Le tableau 2-1 résume le nombre d'itérations du solveur global résidu conjugué préconditionné par multigrilles, ainsi que le temps de calcul pour ces 4 cas.

Taux de déraffinement	Nombre de nœuds du maillage grossier	Nombre d'itération du solveur	CPU
3,9	3088	9	23,4
6,0	2007	13	17,345
8,1	1497	14	14,1
13,2	921	19	15,924
17,2	705	24	18,4
26,6	456	41	28,7

tableau 3-7 : influence du taux de déraffinement sur le temps de calcul

On constate tout d'abord que le nombre d'itérations croît en même temps que le taux de déraffinement. Ce comportement est tout à fait naturel, puisque plus la grille grossière comporte de nœuds, plus la correction calculée par la résolution exacte approche le solveur de la solution exacte. En ce qui concerne les temps de calcul, on peut supposer que le temps de calcul de la méthode avec un $C_R = 3,9$ est fortement pénalisé par la taille de la grille grossière (3088 nœuds). En effet, on inverse dans ce cas un système de 12352 degrés de liberté par une méthode directe. On constate tout de même que la méthode la plus performante est celle utilisant un taux de déraffinement de 8,1, ce qui confirme les observations précédentes. Le solveur multigrilles perd en robustesse et en rapidité si l'on cherche à trop déraffiner entre deux niveaux successifs. On cherchera donc à toujours obtenir un taux de déraffinement de l'ordre de 8 entre deux grilles consécutives, ce qui rejoint les résultats de la littérature [Guillard93].

3.3.2. Influence de la taille de la grille grossière

On cherche ici à montrer que la taille de la grille grossière peut être pénalisante en terme de temps de calcul si celle-ci contient trop de nœuds. En effet, la résolution exacte par factorisation LU devient très coûteuse si le maillage grossier est trop important. On étudie un système également issu de la simulation du roulage en cours de mise en forme avec un maillage de 26025 nœuds et les mêmes lois de comportement que précédemment. Trois solveurs deux grilles sont testés, avec des grilles grossières comportant respectivement 3040, 1446 et 942 nœuds. Le tableau 3-8 montre que le solveur le plus performant est celui avec le maillage grossier de 1446 nœuds, ce qui correspond à $C_R = 18$. Le solveur avec $C_R = 8,5$ est trop pénalisé par la taille de sa grille grossière (3040 nœuds) ce qui l'empêche d'être compétitif.

Taux de déraffinement	Nombre de nœuds du maillage grossier	Nombre d'itération du solveur	CPU
8,5	3040	14	42,716
18	1446	17	38,222
27,6	942	28	55,609

tableau 3-8 : influence de la taille de la grille grossière sur le temps de calcul

On peut supposer que le temps de calcul de la résolution exacte par factorisation complète LU devient pénalisant lorsque le maillage grossier dépasse les 1500 nœuds. On essaiera donc de ne jamais dépasser cette limite et on fixe le nombre de grilles en fonction de cette valeur.

3.3.3. Nombre de niveaux

En supposant que la méthode de déraffinement automatique soit suffisamment efficace pour toujours fournir un maillage contenant moins de 1500 nœuds, il apparaît que trois niveaux de grilles sont suffisants dans les applications qui nous concernent actuellement. En effet, en considérant cette limite de 1500 nœuds combinée avec un facteur de déraffinement de l'ordre de 8, on peut traiter de façon optimale des cas comportant jusqu'à 96000 nœuds, ce qui correspond à l'extrême majorité des cas rencontrés dans les simulations de forgeage. On cherche donc à toujours obtenir entre la grille fine et la grille intermédiaire un facteur de déraffinement de l'ordre de 8 et à avoir une grille grossière contenant moins de 1500 nœuds. On nomme τ_1 le facteur de déraffinement entre la grille fine et la grille intermédiaire et τ_2 le facteur de déraffinement entre la grille intermédiaire et la grille grossière. De la même manière, on nomme R_1 la matrice de restriction permettant de passer du niveau fin au niveau intermédiaire et R_2 la matrice de restriction permettant de passer du niveau intermédiaire au niveau grossier. Le solveur ainsi développé est nommé MG3 dans la suite. Il définit un solveur résidu conjugué préconditionné par multigrilles, avec un V-cycle comme étape de préconditionnement.

3.3.4. Comparaison entre des méthodes deux et trois grilles

Pour se convaincre des choix effectués et de l'apport d'une troisième grille, on compare ici deux méthodes deux grilles ($C_R = 23,8$ et $C_R = 59,5$) avec une méthode trois grilles présentant des paramètres optimaux ($\tau_1 = 8,3$, $\tau_2 = 8,2$ et grille grossière de 504 nœuds). Le système linéaire utilisé est issu d'une simulation de roulage avec un maillage de 34593 nœuds. Les méthodes deux grilles présentent deux taux de déraffinement élevés avec des grilles grossières de respectivement 1456 nœuds et 581 nœuds. Le tableau 3-9 détaille les temps de calcul et les nombres d'itérations de chaque solveur multigrilles. Le solveur trois grilles est le plus rapide, loin devant les deux solveurs deux grilles. On constate cependant que malgré des taux de déraffinement très importants, ces solveurs deux grilles convergent, traduisant ainsi la bonne robustesse de la méthode malgré un déraffinement sous optimal.

Nombre de nœuds du maillage fin	Nombre de nœuds du maillage intermédiaire	Nombre de nœuds du maillage grossier	Nombre d'itération du solveur	CPU
34593	=>	1456 ($C_R = 23,8$)	15	44,6
34593	=>	581 ($C_R = 59,5$)	42	109,4
34593	4 151 ($\tau_1 = 8,3$)	504 ($\tau_2 = 8,2$)	25	28,0

tableau 3-9 : temps de calcul entre deux méthodes deux grilles et une méthode trois grilles

3.3.5. Remaillage

L'utilisation d'une formulation Lagrangienne dans le code Forge3® implique une grande déformation du maillage due aux sollicitations subies par le domaine. Lorsque ce maillage est trop déformé pour garantir une bonne précision des résultats, une procédure de remaillage automatique est effectuée. Une fois cette étape finalisée, on calcule un maillage grossier (qui sera le niveau intermédiaire) à partir du nouveau maillage par l'algorithme de déraffinement automatique. Puis l'on applique une nouvelle fois cet algorithme à la grille intermédiaire pour obtenir le niveau le plus grossier. On dispose alors automatiquement des trois maillages nécessaires au nouveau solveur. On construit les opérateurs R_1 et R_2 par la procédure d'interpolation détaillée en 3.3.2.

Entre deux étapes de remaillage, on suppose ces opérateurs constants. Bien que les nœuds des trois maillages se déplacent à chaque incrément temporel, on fait donc le choix de ne pas réévaluer ces opérateurs. Cette hypothèse se base sur les résultats obtenus dans [Mocellin99] dans lequel il apparaît qu'elle observe qu'elle n'engendre pas de chute des performances du solveur à travers le temps.

On dispose ainsi d'un solveur multigrilles entièrement automatisé et totalement transparent pour l'utilisateur. Il ne nécessite aucune intervention extérieure pour la réalisation de simulation entière.

3.3.6. Limitation

Le point clef de la méthode est la capacité à obtenir un déraffinement automatique satisfaisant, avec un taux de déraffinement de l'ordre de 8 entre la grille fine et la grille intermédiaire, ainsi qu'avec une grille grossière contenant 1500 nœuds ou moins. Malgré la qualité de l'algorithme de déraffinement par remaillage local, il est quelquefois difficile de réunir ces conditions sur des géométries très complexes, comme celles parfois rencontrées en fin de forgeage. Afin de satisfaire le plus souvent possible à toutes les conditions, une procédure modifiant les critères de déraffinement est mise en place.

Les deux facteurs régissant un déraffinement sont la tolérance *tol* sur la variation de volume entre le maillage initial et le maillage résultant ainsi que le coefficient de déraffinement *C*. Les valeurs par défaut choisies sont *tol* = 0,25 ce qui correspond à une variation possible de 25% du volume et *C* = 2 pour obtenir idéalement un taux de déraffinement de 8 si la géométrie le permet. Si ces valeurs ne suffisent pas pour atteindre un taux de déraffinement de 5,5, on effectue un nouveau déraffinement en étant plus exigeant sur le coefficient de déraffinement (on le multiplie par 1,5) et ainsi de suite. On cherche à déraffiner d'avantage là où c'est possible, sans modifier le volume de la pièce. L'algorithme implémenté est détaillé dans le tableau 3-10.

tableau 3-10 : algorithme pilotant la procédure de déraffinement de maillage

```

C = 2 // valeur par défaut
tol = 0.25 // valeur par défaut
CR = 0 // initialisation

Tant que ( CR ≤ 5,5 et C ≤ 6 ) {
    maillageGrossier = déraffinement(maillageFin, C , tol )
    CR =  $\frac{nbnoeudsMaillageFin}{nbnoeudsMaillageGrossier}$ 
    C ← 1.5C
} fin tant que
retourne maillageGrossier

```

Cette procédure permet d'obtenir des maillages qui sont suffisamment grossiers sur des géométries complexes. Cependant, il arrive encore dans quelques cas particuliers que cela ne suffise pas, et que l'on n'arrive pas à obtenir un maillage grossier de moins de 1500 nœuds. Ceci représente à l'heure actuelle le facteur limitatif de notre approche. La convergence du solveur multigrilles n'est pas remise en cause, mais son efficacité n'est plus optimale. Pour pallier à ce souci, on pourra toujours réutiliser le solveur linéaire traditionnel du logiciel dès lors que le déraffinement n'est plus efficace, afin de conserver un solveur compétitif. Une autre solution est de résoudre le problème sur la grille grossière par une méthode itérative, ce qui nécessite de définir son critère de convergence comme dans [Mocellin99]. Ces situations ne se rencontrant que sur des types de procédé particulier et généralement en fin de forgeage, les multigrilles restent beaucoup plus performantes sur la plus grande partie de la simulation.

La mise en place de la procédure pilotant le déraffinement de maillage rend la génération de maillage grossier plus adaptée à notre solveur, ce qui lui confère une plus grande robustesse et une meilleure rapidité. Ces performances sont discutées dans la partie suivante.

3.4. Performances

3.4.1. Convergence asymptotique

On s'intéresse ici au taux de convergence asymptotique de notre solveur multigrilles. Pour cela, on utilise six maillages avec respectivement 5944, 11245, 31200, 47083, 61987 et 91127 nœuds sur lesquels on isole un système linéaire issu d'une simulation de roulage en début de mise en forme. On s'intéresse alors au temps de résolution de ce système. Les données numériques de la simulation sont celles caractéristiques du roulage, déjà détaillées dans 3.5.1. La figure 3-17 présente un exemple de maillage fin (31200 nœuds) avec les maillages intermédiaire (3233 nœuds) et grossier (509 nœuds) correspondants.

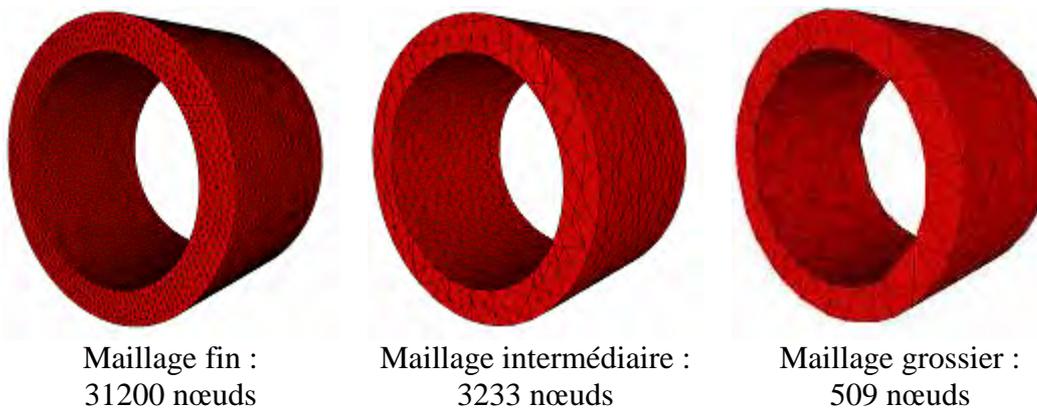


figure 3-17: maillages fin, intermédiaire et grossier d'une bague en début de roulage

On trace tout d'abord, sur la figure 3-18, l'évolution du temps de résolution en fonction du nombre de nœuds du système pour la méthode multigrilles et pour le solveur habituel de Forge3®. On constate le comportement traditionnel en $O(N^{3/2})$ du solveur résidu conjugué préconditionné par factorisation incomplète, tandis que le solveur multigrilles présente un comportement qui semble linéaire. Cette première comparaison montre l'apport fondamental des solveurs de type multigrilles, avec une réduction des temps de calcul de plus en plus conséquente à mesure que la taille du système croît. On constate également que le solveur multigrilles toujours plus performant que le solveur traditionnel, même sur la résolution d'un système de 5944 nœuds.

La figure 3-19 présente l'évolution du comportement du seul solveur multigrilles, approchée par une asymptote linéaire. On vérifie ainsi numériquement la propriété de convergence asymptotique linéaire dans le cas de l'utilisation des multigrilles en tant que préconditionneur.

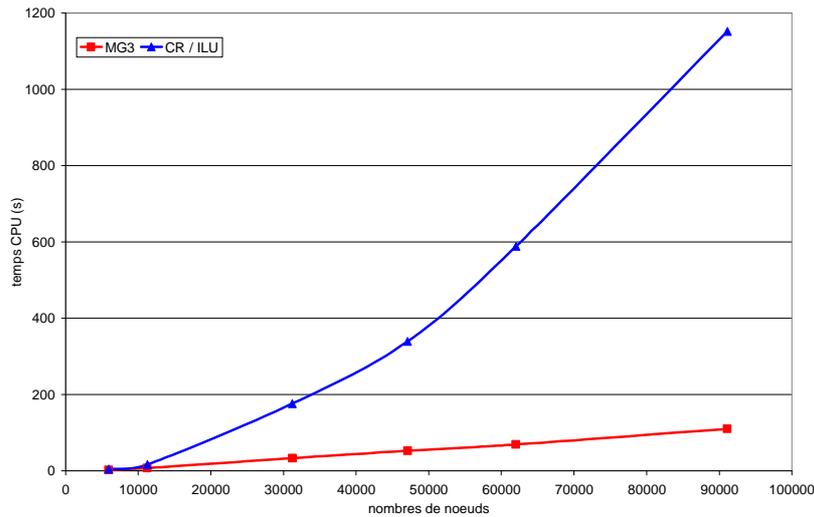


figure 3-18: convergence asymptotique des solveurs traditionnel et multigrilles

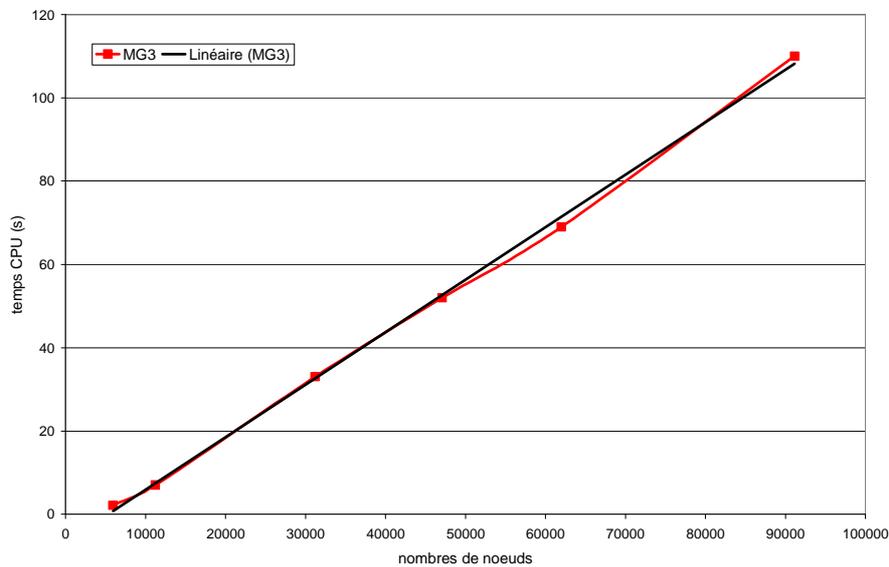


figure 3-19: convergence asymptotique linéaire du solveur multigrilles

3.4.2. Etude sur trois incréments de temps de calcul

On se place dans le cadre de la simulation du roulage en début de mise en forme, en s'intéressant à trois simulations différentes (11245, 31200 et 61987 nœuds). On s'intéresse tout d'abord à quantifier le temps d'initialisation du solveur multigrilles par rapport au temps de résolution, puis on l'étudie aux performances réelles du solveur.

3.4.2.1. Temps de création d'un niveau grossier

Le but ici est de comparer le temps d'initialisation du solveur multigrilles, nécessaire en début de simulation et après chaque remaillage, avec le temps de calcul de trois incréments temporels. La phase d'initialisation comprend le déraffinement automatique des maillages fin et intermédiaire ainsi que le calcul des deux matrices de restriction par la procédure d'interpolation. Le tableau 3-11 résume les nombres de nœuds des trois différents maillages

utilisés ainsi que des maillages intermédiaires et grossiers résultants, avec les taux des déraffinements. Ces taux sont compris entre 6 et 10, avec des tailles de grille grossière allant de 189 à 790 nœuds. On peut donc déjà conclure que le déraffinement a été efficace pour les trois cas.

Nombre de nœuds du maillage de calcul	11245	31200	61987
Nombre de nœuds du maillage intermédiaire	1452	3233	6061
Nombre de nœuds du maillage grossier	189	509	790
τ_1	7,74	9,65	10,23
τ_2	7,68	6,35	7,67

tableau 3-11 : Nombre de nœuds et taux de déraffinement des trois cas

Le tableau 3-12 détaille les temps de construction du niveau intermédiaire et du niveau grossier, ainsi que le pourcentage de temps passé à initialiser le solveur vis-à-vis du temps de calcul total de trois incréments de temps. La première conclusion est que cette étape d'initialisation est négligeable par rapport à la résolution des incréments temporels. On constate que pour une simulation à 31200 nœuds, l'initialisation du solveur multigrilles représente 1 % du temps de calcul de trois incréments temporels. Sachant que d'une manière générale, la fréquence des remaillages est comprise entre 5 et 20 incréments, avec des remaillages de plus en plus fréquents en fin de simulation, on peut affirmer que l'initialisation du solveur multigrilles représente un coût négligeable.

Nombre de nœuds	11245	31200	61987
Temps de construction du niveau intermédiaire (s)	7,1	25,0	49,3
Temps de construction du niveau grossier (s)	1,3	2,6	5,6
Temps de construction des deux opérateurs (s)	8,4	27,6	54,9
% de temps CPU de l'initialisation du solveur MG3 par rapport au temps CPU de trois incréments	4 %	1 %	0,8 %

tableau 3-12 : temps de construction des niveaux intermédiaire et grossier

3.4.2.2. Accélération

On compare ici (tableau 3-13) les temps de calcul des trois incréments temporels, avec d'une part le logiciel Forge3® dans sa version commerciale et d'autre part le même logiciel enrichie du solveur multigrilles. Comme prévu, les accélérations de temps de calcul constatées augmentent avec la taille du maillage considéré. Pour une simulation avec 11245

nœuds, le temps de calcul est divisé par 2,1 tandis que pour une simulation à 61987 nœuds, il est divisé par 7,6.

Nombre de nœuds	11245	31200	61987
Temps de calcul avec Forge3®	6min 53sec	3h 39min 51sec	14h 23min 2sec
Temps de calcul avec Forge3® + solveur MG3	3min 15sec	43min 49sec	1h 53min 20sec
Accélération	2,1	5,0	7,6

tableau 3-13 : accélération de temps de calcul sur trois incréments de temps

3.4.3. Simulation du forgeage d'une bielle

On s'intéresse à la simulation du forgeage d'une bielle en début de calcul. Les données numériques sont les mêmes que lors de la simulation étudiée dans la 3.1.3 traitant également du forgeage d'une bielle. Les différents niveaux de la méthode multigrilles contiennent 35290, 5973 et 1085 nœuds, visible sur la figure 3-20.



figure 3-20: maillages fin, intermédiaire et grossier d'une bielle en début de forgeage

On compare les temps de calcul des itérations de Newton-Raphson lors des deux premiers incréments temporels. Il est à noter que le temps de calcul d'une itération de Newton-Raphson comprend la phase de sous incrémentation (calcul de plusieurs résidus) en plus de la résolution elle-même. On constate une accélération moyenne de l'ordre de 2,4 par itération de Newton-Raphson, ce qui correspond à une accélération moyenne d'environ 3,5 par résolution de système linéaire.

Incrément	Numéro de l'itération	CPU Forge3 (s)	CPU Forge3 MG3 (s)	Accélération
1	1	26	20	1,3
	4	97	36	2,7
	7	57	29	2,0
	10	49	26	1,9
2	1	68	30	2,3
	5	74	33	2,2
	10	83	36	2,3
	15	93	35	2,7
	20	80	37	2,2
	25	72	28	2,6
	30	81	22	3,7

tableau 3-14 : accélération du temps de calcul durant les itérations de Newton-Raphson

3.4.4. Simulation entière de forgeage d'une bague CUV40918

On simule ici entièrement la phase de roulage sur le symbole CUV40918 avec une cinématique en excentration constante et en faisant varier la richesse des maillages de calcul. Les données numériques utilisées sont toujours celles caractéristiques du roulage définies dans le chapitre précédent.

3.4.4.1. Simulation avec un maillage de 5000 nœuds

Le maillage de calcul comporte, durant l'ensemble de la simulation, un nombre de nœuds de l'ordre de 5000. La figure 3-21 trace l'évolution du nombre de nœuds de chaque grille, les variations étant le résultat des remaillages. Il y a eu 13 étapes de remaillage, avec une fréquence plus grande en fin de simulation. On constate également que les maillages intermédiaires comportent toujours moins de 1000 nœuds, et les maillages grossiers moins de 600. Aucune difficulté de déraffinement n'est apparue. Les maillages fin, intermédiaire et grossier en fin de calcul sont visibles sur la figure 3-22.

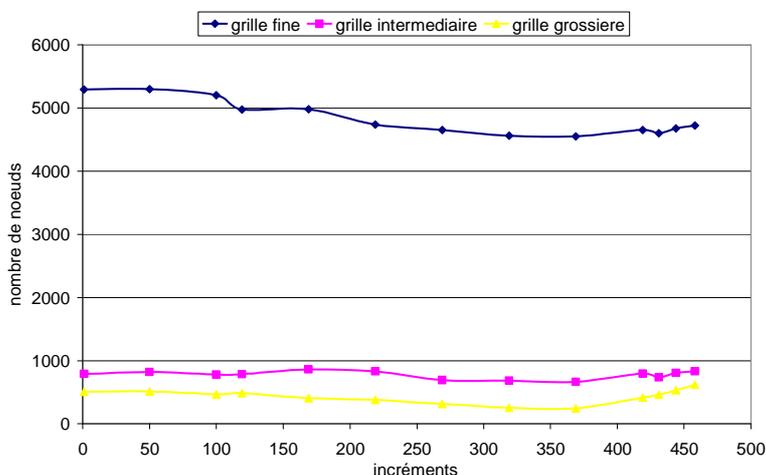


figure 3-21: évolution du nombre de nœuds des maillages fin, intermédiaire et grossier

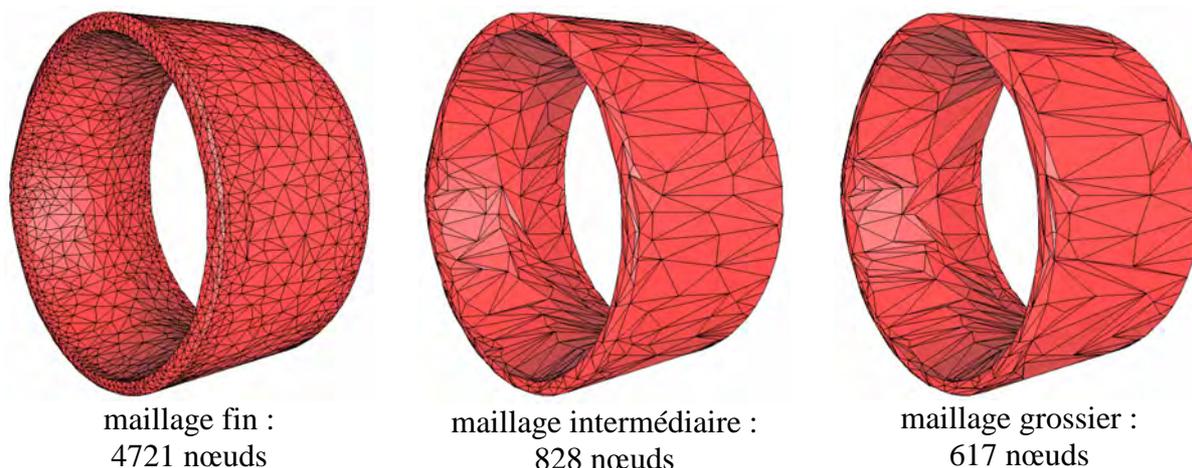
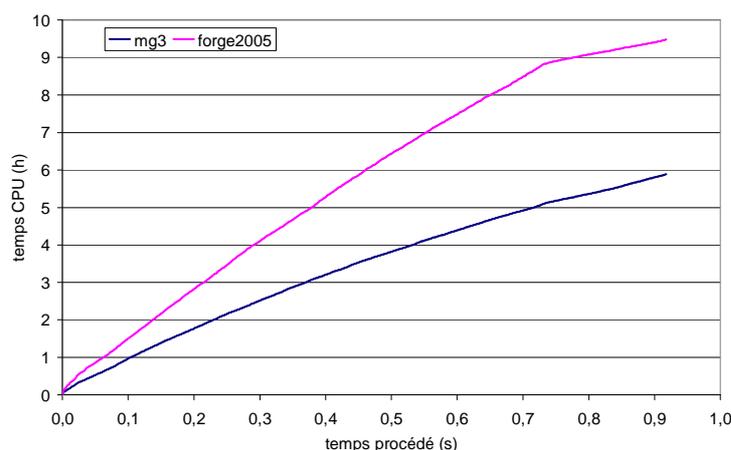


figure 3-22: maillages fin, intermédiaire et grossier en fin de calcul

Le temps de résolution (figure 3-23) est quasiment constant avec le solveur multigrilles. Avec le solveur traditionnel, on constate une amélioration en fin de simulation qui se manifeste lorsque la bague est totalement expansée dans la matrice. On constate ce phénomène sur l'ensemble des simulations de roulage et ce quelque soit le symbole étudié. On peut légitimement penser que ceci est dû au fait qu'il n'y a quasiment plus de mouvement de corps rigide en fin de simulation, le lopin étant entièrement en contact avec la matrice. On ne constate pas ce phénomène avec le solveur multigrilles, celui-ci étant quasiment insensible au mouvement de corps rigide. En effet, un mouvement de corps rigide concerne les basses fréquences puisqu'il est question de déplacement à travers tout le domaine. Ceci demande à un solveur classique un grand nombre d'itérations pour converger. C'est précisément pour contrer ce phénomène que les méthodes multigrilles ont été conçues. Ainsi l'évolution du temps de calcul au cours de la simulation est constant.

On constate une réduction de temps de calcul d'un facteur 1,6 avec un gain de plus de 3h30min sur l'ensemble de la simulation. Ce résultat est très important et justifie pleinement l'emploi d'un solveur multigrilles, même sur des simulations peu riches en terme de nombre de nœuds (5000 nœuds ici).



	Temps CPU
Forge2005®	9h 28min
Forge2005 MG3	5h 53min
Accélération	1,6

figure 3-23: évolution des temps de calcul sur CUV40918 et accélération

3.4.4.2. Simulation avec un maillage de 60000 nœuds

On effectue ici la même simulation que précédemment, avec un maillage initial beaucoup plus riche (60725 nœuds). La figure 3-24 montre sur sa gauche l'évolution de la taille des

maillages de calcul, intermédiaire et grossier, durant toute la simulation comprenant 28 étapes de remaillage. Sur la droite, on regarde de plus près l'évolution des grilles intermédiaire et grossière. Alors que le nombre de nœuds du maillage de calcul varie entre 60000 et 70000, celui du maillage intermédiaire possède entre 4000 et 8000. En ce qui concerne le maillage grossier, sa taille est d'environ 1000 nœuds tout au long du calcul, avec une augmentation jusqu'à 1800 nœuds en fin de simulation. La figure 3-25 montre les différents niveaux après le dernier remaillage.

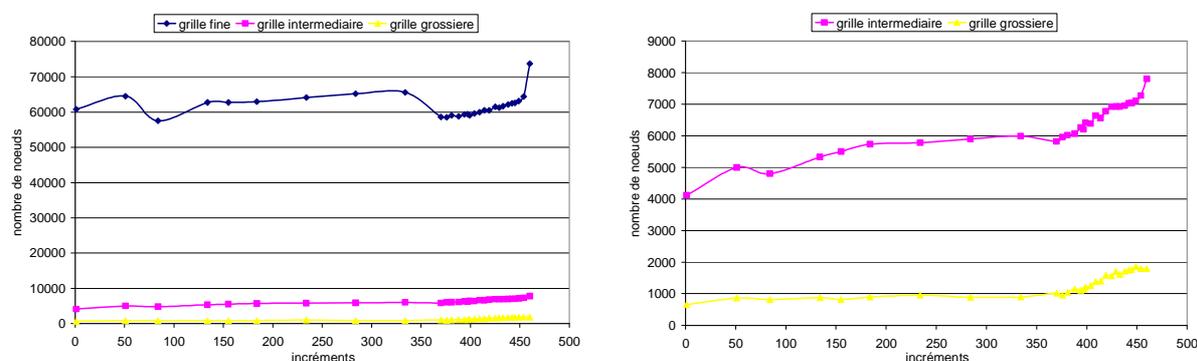
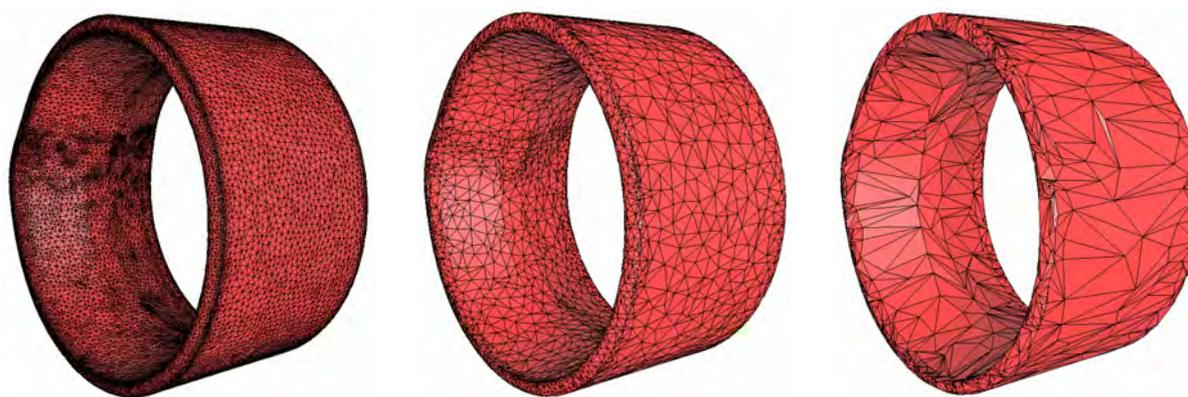


figure 3-24: évolution du nombre de nœuds des maillages fin, intermédiaire et grossier



maillage fin :
73772 nœuds

maillage intermédiaire :
7803 nœuds

maillage grossier :
1795 nœuds

figure 3-25: maillages fin, intermédiaire et grossier en fin de calcul

On s'intéresse tout d'abord au temps de construction des niveaux intermédiaire et grossier tout au long de la simulation (figure 3-26). On constate que ces temps sont quasi constants et de l'ordre de la minute, ce qui est négligeable vis-à-vis des temps de résolution.

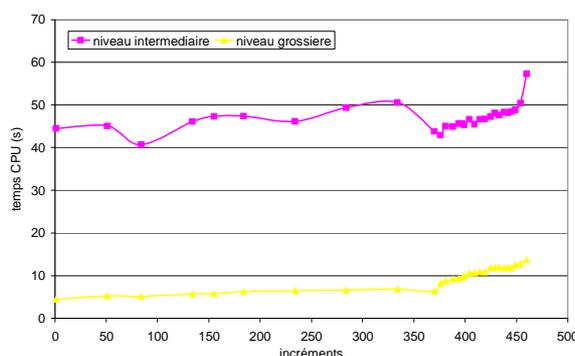
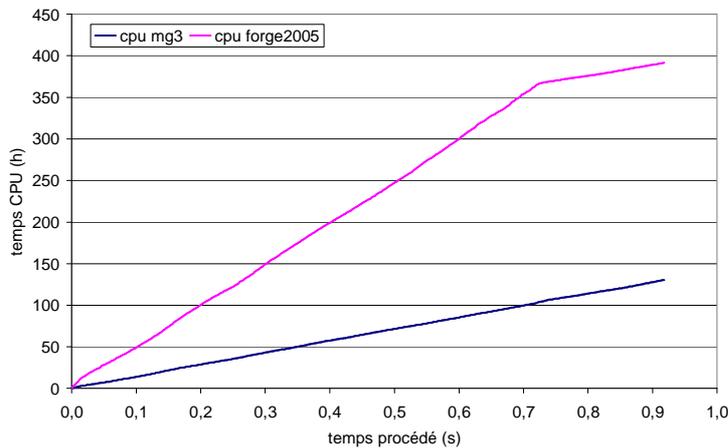


figure 3-26: évolution du temps de construction des grilles intermédiaire et grossière

En ce qui concerne l'évolution des temps de calcul (figure 3-27), les tendances sont très proches de celles relevées sur la simulation précédente. Le comportement du solveur multigrilles est constant dans le temps. On ne constate pas d'affaiblissement des performances en fin de simulation qui aurait pu être induit par la taille de la grille grossière. La méthode reste donc compétitive avec une grille grossière de 1800 nœuds. Si l'on compare les temps de calcul avant le changement de pente de la méthode classique (temps procédé de 0,7s), on constate une accélération par un facteur 3,5. Sur l'ensemble de la simulation, l'accélération due au solveur multigrilles est d'environ 3, avec un gain de l'ordre de 261h.

Du point de vue de l'utilisateur, cette accélération peut paraître relativement faible au regard du nombre de nœuds de la simulation. Cependant, il faut bien garder à l'esprit que l'on compare les temps de calcul de l'ensemble de la simulation. Si l'on compare les temps de résolution d'un système linéaire isolé, on observe une accélération de l'ordre de 6. Le gain acquis grâce à l'emploi du solveur multigrilles est donc extrêmement appréciable. Mais outre la résolution de système linéaire, une simulation comporte de nombreuses autres étapes dont le coût reste inchangé, comme la sous incrémentation de Newton-Raphson, l'analyse de contact, le remaillage ou encore le transport des variables d'état. Ceci explique pourquoi l'accélération globale sur une simulation est inférieure à l'accélération constatée sur la résolution d'un système linéaire.



	Temps CPU
Forge2005@	391h 34min
Forge2005 MG3	130h 27min
Accélération	3

figure 3-27: évolution des temps de calcul sur CUV40918 et accélération

3.4.5. Simulation entière du forgeage d'une bague BEG40899

On s'intéresse maintenant au symbole BEGB40899 avec un maillage de l'ordre de 15000 nœuds. On simule entièrement la phase de roulage avec une cinématique en excentration constante, et en faisant varier la richesse des maillages de calcul. Le matériau et le frottement sont les mêmes que précédemment.

La figure 3-28 présente l'évolution du nombre de nœuds des différentes grilles au cours des 35 étapes de remaillage effectuées. La figure 3-29 montre quant à elle les trois différentes grilles à la suite du dernier remaillage.

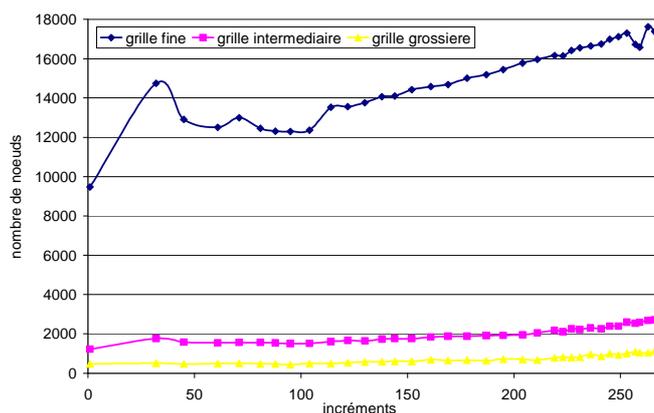


figure 3-28: évolution du nombre de nœuds des maillages fin, intermédiaire et grossier

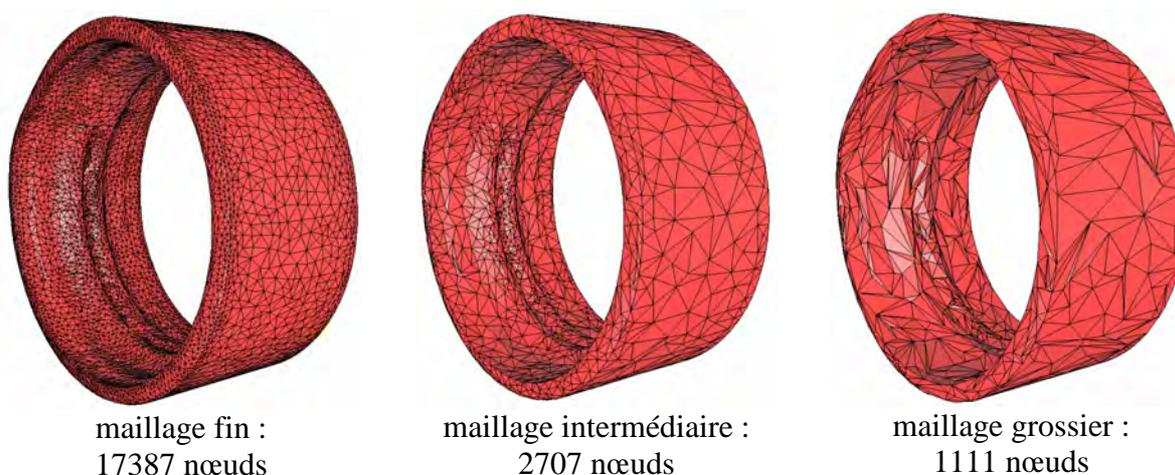


figure 3-29: maillages fin, intermédiaire et grossier en fin de calcul

Si l'on s'intéresse à l'évolution des taux de déraffinement (figure 3-30), on constate un très bon comportement de l'algorithme de déraffinement automatique pour passer du niveau fin au niveau intermédiaire, avec un taux quasi constant égal à 8. En ce qui concerne le passage du niveau intermédiaire au niveau grossier, on observe un taux de déraffinement de l'ordre de 3 avec une taille de grille grossière comprise entre 500 et 1100 nœuds, ce qui est également parfaitement acceptable.

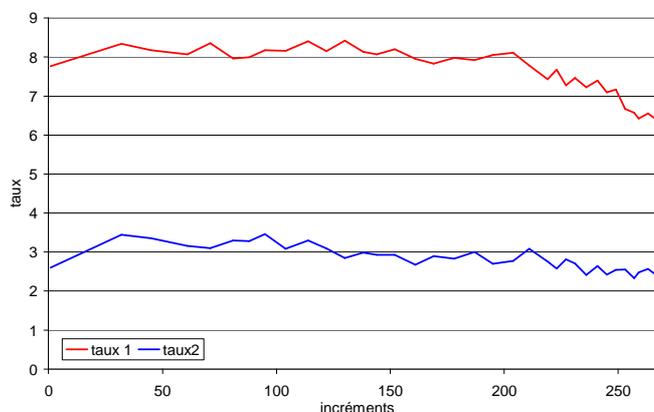
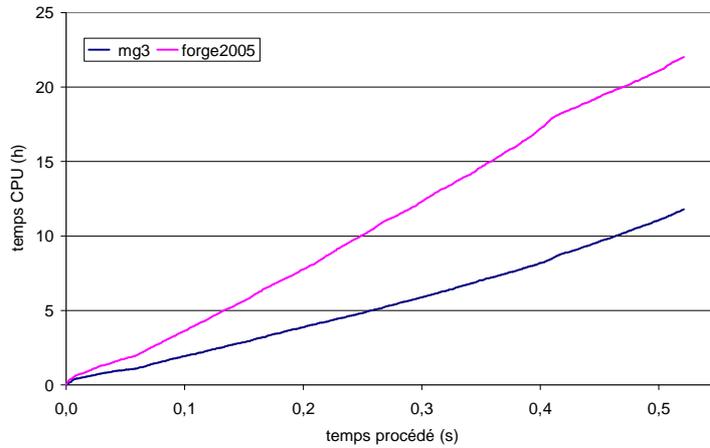


figure 3-30: évolution des taux de déraffinement

En terme de temps de calcul (figure 3-31), les évolutions sont assez identiques à celles relevées sur le symbole CUV40918, avec cependant une accélération moins marquée de la

méthode traditionnelle en fin de calcul. Sur cette simulation en moyenne de 15000 nœuds, on constate une accélération d'un facteur 1,9, ce qui se traduit par un gain de plus de 10h.



	Temps CPU
Forge2005®	22h 00min
Forge2005 MG3	11h 47min
Accélération	1,9

figure 3-31: évolution des temps de calcul sur BEGB40899 et accélération

3.5. Conclusion

Le temps de calcul nécessaire pour une simulation est une préoccupation primordiale pour l'utilisation industrielle d'un logiciel. La diminution de ce temps était l'enjeu de ce travail. La mise en place d'une nouvelle méthode de résolution de systèmes linéaires basé sur la méthode multigrilles permet donc de réduire considérablement les temps de calcul. La taille des simulations étant vouée à continuer à s'accroître afin d'obtenir une précision de plus en plus grande et simuler des procédés de plus en plus complexes, les facteurs de réduction engendrés par les multigrilles vont également devenir de plus en plus intéressants, du fait de sa convergence asymptotique linéaire, convergence observée numériquement pour la résolution d'un système linéaire. L'utilisation des multigrilles est d'ores et déjà un succès, avec une réduction effective allant d'un facteur 1,6 à 3 pour des simulations de roulage allant de 5000 à 60000 nœuds. C'est un résultat important qui permet une grande économie de temps. On constate également que le temps de remaillage, incompressible, finit par devenir une limitation aux accélérations obtenues.

Ce travail prouve également que l'emploi d'un solveur multigrilles est bien adapté à notre domaine d'application, à savoir la simulation du forgeage en 3D, avec une formulation couplée vitesse pression, sur des matériaux incompressibles.

Les choix effectués ici ont permis la mise en place d'une méthode autonome totalement transparente pour l'utilisateur et ne nécessitant aucune intervention extérieure. Ainsi, la méthode a été bâtie autour d'un algorithme de déraffinement automatique de maillage afin de construire les différentes grilles. L'emploi de la librairie PETSc a permis de bénéficier d'une implémentation ultra compétitive du produit matriciel pour des matrices creuses. Cette librairie a également mis à notre disposition la structure de la méthode multigrilles ainsi qu'un solveur résidu conjugué optimisé. Les choix en terme de lisseur et de solveur sur la grille grossière ont été assez naturels, avec une méthode itérative de Jacobi par bloc pour le lissage et une résolution directe par factorisation LU pour résoudre le système grossier. L'étude paramétrique effectuée a permis de construire un solveur robuste et performant sur différents cas de forgeage. Il converge dans tous les cas présentés ici et ce de manière automatique. La seule limitation actuelle de la méthode réside dans la difficulté à déraffiner suffisamment les géométries trop tortueuses. Si tel n'est pas le cas, le solveur trois grilles est toujours plus performant que le solveur de référence.

On peut noter enfin le très bon comportement du solveur multigrilles sur le procédé de roulage d'anneaux, procédé ayant motivé cette étude. Les réductions de temps de calcul obtenus sur les différents symboles étudiés sont très encourageantes et sont une grande avancée pour l'utilisation encore plus massive de la simulation dans l'industrie.

3.6. Perspectives

En terme d'optimisation des performances, deux pistes peuvent être explorées dans un futur immédiat. La première concerne l'actualisation des opérateurs de transfert. L'hypothèse a été faite dans cette étude que les opérateurs sont constants au cours du temps, entre deux remaillages. Bien que ce choix n'ait pas entraîné de perte de performance visible sur les résultats obtenus, on peut légitimement penser que l'actualisation de ces opérateurs au cours du temps aura un impact bénéfique sur les temps de résolution, et que cet apport sera supérieur au temps de calcul additionnel pour l'actualisation. En effet, les opérateurs étant plus exactes, la résolution devrait n'en être que plus rapide et le solveur plus performant. L'autre axe d'amélioration du solveur est le pilotage du déraffinement présenté dans le tableau 3-10. Bien que le choix des coefficients (facteur de déraffinement minimum, tolérance sur la variation du volume, critère de déraffinement) régissant cet algorithme soit le résultat d'une longue étude sur diverses géométries rencontrées en forgeage, rien ne permet d'annoncer que ces paramètres sont optimaux. Il devrait être possible de les affiner afin de repousser la limitation actuelle de la méthode sur les géométries très complexes. La mise en place d'une méthode itérative pour la résolution du système grossier peut également être une autre alternative.

Les perspectives liées à ce travail dans le but de généraliser son utilisation sont diverses. On peut tout d'abord penser à la compatibilité du solveur multigrilles avec l'utilisation d'outils flottants. Il faudra prendre en compte l'ajout de degrés de liberté supplémentaires dans la construction des opérateurs afin de les rendre compatibles avec le système à résoudre. De la même manière, l'extension du solveur multigrilles au calcul multi-corps ne devrait pas être source de points bloquants. Il faudra, dans ce cas, déraffiner séparément chaque corps pour construire un opérateur de restriction local spécifique. A partir de ces différents opérateurs locaux, on définira un opérateur de restriction global, compatible avec la matrice du système à résoudre, afin de construire la matrice du système grossier.

Ce travail a été réalisé dans le but de permettre son extension au calcul parallèle. C'est une des multiples raisons qui nous a poussé à utiliser PETSc. Ainsi, le solveur multigrilles est d'ores et déjà parallélisé. Cependant, un travail conséquent reste à mener sur le déraffinement de maillage en parallèle, ainsi que sur la construction des opérateurs de transfert, pour faire fonctionner ces opérations de manière parallèle sans coût de communication prohibitif. Une réflexion devra aussi être menée sur la manière de résoudre en parallèle le système grossier. L'émergence de méthode de résolution directe en parallèle ([Amestoy et al.98], [Li et al.03]) semble être un début de réponse.

Conclusion générale

La première partie de ce travail a pour objet l'amélioration de la simulation numérique tridimensionnelle du procédé de roulage d'anneaux, développé par la société SNR.

Un premier volet a été le développement d'une nouvelle formalisation des outils dans le logiciel Forge2005® afin de rendre compte de la cinématique réelle de la machine, à savoir un pilotage en force maximale et une molette folle en rotation sur elle-même. Les résultats de cette étape sont fort satisfaisants, la simulation se montrant prédictive sur la trajectoire de la molette durant la phase de roulage et sur le temps de cycle, ce qui traduit entre autre la bonne évaluation des efforts mises en jeu. C'est un premier résultat fondamental pour l'industriel, permettant l'optimisation du procédé.

Un deuxième volet de ce travail a été le recalage thermique effectué un cycle entier de production en régime thermique établi. Cette étude fût rendue possible grâce à la campagne expérimentale menée chez SNR qui nous a permis de disposer d'informations précieuses et jusqu'alors inaccessibles. On a tout d'abord recalé les paramètres thermiques (température ambiante durant les différentes étapes, coefficients d'échange thermique entre le lopin et l'air et des outils) de la simulation sur un premier symbole en s'attachant à reproduire l'évolution de la température de la matrice à 3 mm et 20 mm de profondeur. Puis le recalage a été validé en utilisant ce nouveau jeu de paramètres sur un deuxième symbole sensiblement différent en taille et en gravure interne. Les résultats sont extrêmement satisfaisants, la simulation se montrant prédictive sur l'évolution de la température de la matrice durant tout le cycle une fois le régime thermique établi. On constate également une bonne modélisation des conditions aux limites pour le procédé. Ceci va permettre l'étude de la durée de vie de l'outillage.

Les efforts consentis nous ont permis d'effectuer un bond un avant significatif en matière de simulation numérique du procédé de roulage, mais ils nécessitent également des temps de calcul importants et extrêmement contraignants. On s'est donc également intéressé en parallèle à la réduction de ces temps de calcul, ce qui nous amène à la seconde partie de cette contribution.

La deuxième partie de ce travail traite du développement et de l'intégration d'une nouvelle méthode de résolution de systèmes linéaires, plus compétitive que celle actuellement employée, afin de réduire les temps de calcul des simulations sous Forge2005® dans sa version 3D. Notre choix s'est porté vers un solveur multigrilles, ces méthodes étant optimales en terme de convergence asymptotique, avec un taux de convergence en $O(N)$, en comparaison au $O(N^{3/2})$ du solveur par résidu minimal préconditionné original. De plus, de précédents travaux [Mocellin99], [El Hadj87] avaient prouvé que les multigrilles étaient bien adaptées à notre cadre d'étude. Une des principales exigences requise pour ce solveur, hormis ses performances, fut son caractère autonome et automatique, de sorte qu'il soit totalement transparent pour l'utilisateur et compatible avec les nombreux remaillages effectués au cours des simulations. Une large partie de cette étude a donc été dédiée à la paramétrisation et à l'optimisation de cette méthode multigrille, fondé sur une méthode de déraffinement de maillage automatique. A cet effet, l'utilisation du mailleur MTC® permet d'obtenir des maillages grossiers suffisamment déraffiné et exploitable par le solveur multigrilles. Les résultats en terme de réduction de temps de calcul sont très encourageants. La convergence

asymptotique linéaire de la méthode multigrilles est obtenue numériquement. On constate également une meilleure robustesse de ce solveur, vis-à-vis de la méthode traditionnelle.

L'apport du solveur est visible dès 5000 nœuds, avec une division globale du temps de temps de calcul de la simulation par un facteur 1,6. Pour une simulation utilisant un maillage de l'ordre de 60000 nœuds, on constate des divisions de temps de résolution d'un système linéaire par un facteur 6, pour une division du temps total de simulation par un facteur 3, ce qui représente un gain de plus de 260h.

Perspectives

Les perspectives liées à cette étude sont diverses, tant du point de vue de la simulation du roulage que du point de vue de l'utilisation de méthodes multigrilles.

A l'origine de ce travail, le but de l'industriel était de connaître les raisons de la casse de la molette. Cette contribution a permis de faire une avancée significative vers cet objectif, mais celui-ci reste à atteindre. Le modèle semble prédictif en effort et en température, il reproduit fidèlement le procédé autant que l'on peut en juger, ce qui semble être une bonne base pour introduire un modèle d'endommagement, encore faut-il utiliser un modèle adapté au roulage. Ce choix passera probablement par la mise en place de nouvelles campagnes expérimentales.

Le travail effectué va d'ores et déjà permettre d'améliorer le paramétrage actuel de la rouleuse. En utilisant le modèle développé, l'optimisation du procédé devrait permettre de minimiser les efforts exercés sur la molette et de réduire les temps de cycle.

Concernant l'utilisation du solveur multigrilles, les perspectives semblent assez naturelles. Du point de l'optimisation, de futurs travaux permettront sans doute d'optimiser le déraffinement de maillage automatique.

D'autre part, des travaux sont actuellement en cours afin de l'utiliser en calcul parallèle. La principale difficulté sera d'essayer d'ajouter les accélérations des deux approches.

Enfin, les performances et la robustesse du solveur multigrilles devront être mises à l'épreuve en l'utilisant sur d'autres cas industriels que ceux traités dans cet étude, afin de jauger tout son potentiel.

Liste des figures

figure 1-1: définition de la normale sortante n	13
figure 1-2: le mini-élément $P1+ / P1$	18
figure 1-3: définition de la distance de pénétration δ	19
figure 1-4: couplage thermo-mécanique dans Forge2005®	33
figure 1-5: modélisation d'un outil flottant rigide	34
figure 2-1: vue éclatée d'un roulement à bille	41
figure 2-2: lopin initial (gauche) et bague finale (droite)	41
figure 2-3: Schéma des procédés de laminage circulaire (gauche) et de roulage (droite)	42
figure 2-4 : vue latérale d'une machine de roulage	43
figure 2-5: trajectoire de la molette	43
figure 2-6: vue de coté de la molette	44
figure 2-7: courbe de déplacement de la crémaillère décrivant un cycle	45
figure 2-8: modélisation de la cinématique de la molette	48
figure 2-9: matrice instrumentée et schéma d'implantation des thermocouples	53
figure 2-10: molette (à gauche) et bague (à droite) du symbole CUV40918	54
figure 2-11: évolution de la température à 3 mm et à 20 mm sur CUV40918	55
figure 2-12: évolution de la vitesse de rotation de la molette sur elle-même au cours du roulage	55
figure 2-13: évolution de la vitesse de roulement sans glissement théorique de la molette	56
figure 2-14: température initiale du lopin (vue de coupe)	57
figure 2-15: comparaison de trajectoires avec une vitesse d'excentration constante	58
figure 2-16: maillage en début et fin de simulation avec excentration constante	59
figure 2-17: évolution du contact durant la phase de roulage	60
figure 2-18: vitesse de rotation de la molette comparée à la vitesse de roulement sans glissement	61
figure 2-19: vitesse de rotation de la molette comparée à l'expérience	61
figure 2-20: effort imposé sur la molette durant le roulage	62
figure 2-21: évolution de la température durant la phase de roulage	63
figure 2-22: comparaison de trajectoires avec une formulation d'effort maximal	64
figure 2-23: maillages en début, fin de roulage et fin de planage en effort maximal	65
figure 2-24: vitesse de rotation de la molette sur elle-même durant le planage	65
figure 2-25: évolution de la température du lopin durant le planage	66
figure 2-26: déformation équivalente en fin de planage	67
figure 2-27: comparaison de trajectoires avec une discrétisation point par point	68
figure 2-28: température initiale de la matrice (vue de coupe) et de la molette	69
figure 2-29: comparaison de la température de la matrice durant un cycle avant recalage	70
figure 2-30: comparaison de la température de la matrice durant un cycle après recalage	70
figure 2-31: maillages en début, fin de roulage et fin de planage en multi-corps	72
figure 2-32: maillages de la molette et de la matrice pour le symbole CUV40918	72
figure 2-33: évolution des contraintes de Von Mises au cours du roulage	73
figure 2-34: évolution de la température de la matrice durant le planage (vue de coupe)	74
figure 2-35: évolution de la température du lopin durant le cycle	76
figure 2-36: température de la bague après 11 secondes de refroidissement (plan de coupe)	77
figure 2-37: remplissage sur le symbole CUV40918	77
figure 2-38: molette (gauche) et bague roulée (droite) du symbole BEGB40899	78
figure 2-39: évolution de la température à 3 mm et 20 mm sur BEGB40899	79
figure 2-40: comparaison de trajectoires avec une discrétisation point par point sur le symbole BEGB40899	79
figure 2-41: maillages en début, fin de roulage et fin de planage en multi-corps	80
figure 2-42: maillage en fin de planage en multi-corps (plan de coupe)	80
figure 2-43: maillages de la molette et de la matrice pour le symbole BEGB40899	81
figure 2-44: température du lopin durant le roulage sur BEGB40899	82
figure 2-45: température du lopin durant le planage sur BEGB40899	83
figure 2-46: température du lopin durant le laminage et le retour sur BEGB40899	84
figure 2-47: température du lopin après 10 secondes de refroidissement sur BEGB40899	84

figure 2-48: comparaison de la température de la matrice durant un cycle sur BEGB40899	85
figure 2-49: remplissage sur le symbole BEG40899	85
figure 3-1: un cycle bigrille	91
figure 3-2: différents cycles 3 grilles	92
figure 3-3: agrégats nodaux définis à partir d'un maillage [Janka02]	95
figure 3-4: opérateurs de prolongement et restriction sur deux maillages indépendants	97
figure 3-5: déraffinement de maillage ; maillages indépendants [Feng et al.98]	97
figure 3-6: raffinement d'un tétraèdre [Mocellin99]	98
figure 3-7: projection de noeuds fins en frontière pour avoir un maillage représentatif [Mocellin99]	98
figure 3-8: classification des nœuds dans l'algorithme MIS	100
figure 3-9: déraffinement de maillage 3D par un algorithme MIS [Adams et al.00]	100
figure 3-10: remaillage d'une cavité	106
figure 3-11: déraffinement du maillage d'une bielle en fin de forgeage ($C_R = 8,4$)	106
figure 3-12: déraffinement du maillage d'une bague en fin de roulage ($C_R = 7,1$)	107
figure 3-13: déraffinement du maillage d'un quart de pièce ($C_R = 7,6$)	107
figure 3-14: interpolation d'un nœud fin dans un élément grossier (en 2D)	107
figure 3-15: bielle en milieu de forgeage	109
figure 3-16: propriété de lissage de la méthode Jacobi par bloc	110
figure 3-17: maillages fin, intermédiaire et grossier d'une bague en début de roulage	117
figure 3-18: convergence asymptotique des solveurs traditionnel et multigrilles	118
figure 3-19: convergence asymptotique linéaire du solveur multigrilles	118
figure 3-20: maillages fin, intermédiaire et grossier d'une bielle en début de forgeage	120
figure 3-21: évolution du nombre de nœuds des maillages fin, intermédiaire et grossier	121
figure 3-22: maillages fin, intermédiaire et grossier en fin de calcul	122
figure 3-23: évolution des temps de calcul sur CUV40918 et accélération	122
figure 3-24: évolution du nombre de nœuds des maillages fin, intermédiaire et grossier	123
figure 3-25: maillages fin, intermédiaire et grossier en fin de calcul	123
figure 3-26: évolution du temps de construction des grilles intermédiaire et grossière	123
figure 3-27: évolution des temps de calcul sur CUV40918 et accélération	124
figure 3-28: évolution du nombre de nœuds des maillages fin, intermédiaire et grossier	125
figure 3-29: maillages fin, intermédiaire et grossier en fin de calcul	125
figure 3-30: évolution des taux de déraffinement	125
figure 3-31: évolution des temps de calcul sur BEGB40899 et accélération	126

Liste des tableaux

<i>tableau 1-1 : Algorithme de Minimisation du résidu</i>	26
<i>tableau 2-1 : tableau récapitulatif des différents paramètres numériques</i>	71
<i>tableau 3-1 : algorithme d'un V-cycle multigrilles (n niveaux de grilles)</i>	93
<i>tableau 3-2 : algorithme MIS de partition de l'ensemble des nœuds fins en 2D</i>	99
<i>tableau 3-3 : comparaison des temps de calcul (en seconde) de $A_H = RA_h P$</i>	103
<i>tableau 3-4 : algorithme de Résidu conjugué préconditionné implémenté dans PETSc</i>	104
<i>tableau 3-5 : algorithme de déraffinement automatique de maillage</i>	105
<i>tableau 3-6 : test de bigrilles idéales; influence du nombre de pré et post lissage</i>	110
<i>tableau 3-7 : influence du taux de déraffinement sur le temps de calcul</i>	112
<i>tableau 3-8 : influence de la taille de la grille grossière sur le temps de calcul</i>	113
<i>tableau 3-9 : temps de calcul entre deux méthodes deux grilles et une méthode trois grilles</i>	114
<i>tableau 3-10 : algorithme pilotant la procédure de déraffinement de maillage</i>	115
<i>tableau 3-11 : Nombre de nœuds et taux de déraffinement des trois cas</i>	119
<i>tableau 3-12 : temps de construction des niveaux intermédiaire et grossier</i>	119
<i>tableau 3-13 : accélération de temps de calcul sur trois incréments de temps</i>	120
<i>tableau 3-14 : accélération du temps de calcul durant les itérations de Newton-Raphson</i>	121

Bibliographie

- [Adams98] **M. F. Adams**, « Multigrid Equation Solvers for Large Scale Nonlinear Finite Element Simulations », *Ph.D. thesis*, EECS Department, University of California, Berkeley, 1998
- [Adams et al.00] **M. F. Adams et R. L. Taylor**, « Parallel multigrid solvers for 3D unstructured finite element problems in large deformation elasticity and plasticity », *International Journal for Numerical Methods in Engineering*, vol.48, pp.1241-1262, 2000
- [Alcouffe et al.81] **R. E. Alcouffe, A. Brandt, J.E. Dendy, J. W. Painter**, « The multi-grid method for diffusion equation with strongly discontinuous coefficients ». *SIAM J. Sci. Statis. Comput.*, vol.2, pp.430-454, 1981
- [Aliaga00] **C. Aliaga**, « Simulation numérique par éléments finis 3D du comportement thermomécanique au cours du traitement thermique d'aciers : application à la trempe de pièces forgées ou coulées ». *Thèse de doctorat*, Ecole Nationale Supérieure des Mines de Paris, Sophia Antipolis, 2000
- [Amestoy et al.98] **P.R. Amestoy, I.S. Duff et J.-Y. L'Excellent**, « Parallélisation de la factorisation LU de matrices creuses non-symétriques pour des architectures à mémoire distribuée », *Calculateurs Parallèles Réseaux et systèmes répartis*, vol.10, n°5, pp.509-520, 1998
- [Arany96] **I. Arany**, « The preconditioned conjugate gradient method with incomplete factorization preconditioners ». *Computers Math. Applic*, vol. 31 pp.1-5, 1996
- [Arnold et al.84] **D.N. Arnold, F. Brezzi, M. Fortin**. « A stable finite element for the Stokes equations ». *Calcolo*, Vol. 21, pp ; 337-344, 1984.
- [Babuska73] **I. Babuška**, « The finite element method with penalty ». *Math. Comp.*, vol. 27, pp. 221-228, 1973
- [Bachvalov66] **N.S. Bachvalov**, « On the convergence of a relaxation method with natural constraints on the elliptic operator ». *USSR Comput. Math. Math. Phys.*, Vol. 6, pp.101-135, 1966
- [Balay et al.97] **Satish Balay, William D. Gropp, Lois Curfman McInnes and Barry F. Smith**, « Efficient Management of Parallelism in Object Oriented Numerical Software Libraries ». *Modern Software Tools in Scientific Computing*, Birkhauser Press, pp.163-202, 1997

-
- [Barboza04] **J. A. P. Barboza**, « Traitement du contact entre corps déformables et calcul parallèle pour la simulation 3D du forgeage multicorps ». *Thèse de doctorat*, École Nationale Supérieure des Mines de Paris, Sophia Antipolis, 2004
- [Bousseta05] **Ramzy Bousseta**, « Estimateurs d'erreur et remaillage adaptatif : application à la simulation 3D des procédés de mise en forme des matériaux ». *Thèse de doctorat*, Ecole Nationale Supérieure des Mines de Paris, Sophia Antipolis, 2005
- [Boyère99] **E. Boyère**, « Contribution à la modélisation numérique thermomécanique tridimensionnelle du forgeage ». *Thèse de doctorat*, Ecole Nationale Supérieure des Mines de Paris, Sophia Antipolis, 1999
- [Brandt et al.79] **A. Brandt and N. Dinar**, « Multigrid solutions to flow problems ». *Numerical Methods for Partial Differential Equations*, Academic Press, New York, pp.53–147, 1979
- [Brandt et al.82] **A. Brandt, S. F. McCormick, J. Ruge**, « Algebraic multigrid (AMG) for automatic multigrid solution with application to geodic computations ». *Institute for Computational Studies*, POB 1852, Fort Collins, Colorado, 1982
- [Carre95] **G. Carre**, « Simulation numérique d'écoulements turbulents compressibles et stationnaires par méthodes multigrille ». *Thèse de doctorat*, Université de Nice-Sophia Antipolis, 1995
- [Carte et al.00] **G. Carte, T. Coupez, H. Guillard, S. Lanteri**, « Coarsening technique in multigrid applications on unstructured meshes ». *European Congress on Computational Methods in Applied Sciences and Engineering*, ECCOMAS, Barcelona, 2000
- [Chan et al.94] **T. F. Chan et B. F. Smith**, « Domain decomposition and multigrid algorithm for elliptic problems on unstructured meshes », *Electronic Transactions on Numerical Analysis*, vol.2, pp.171-182, 1994
- [Coupez91] **T. Coupez**, « Grandes déformations et remaillage automatique ». *Thèse de doctorat*, Ecole Nationale Supérieure des Mines de Paris, Sophia Antipolis, 1991
- [Coupez95] **T. Coupez**, « A mesh improvement method for 3D automatic remeshing ». *4th International Conference of Numerical Grid Generation in Computational Fluid Dynamics and Related Fields*. Swansea, Wales 6th – 8th April 1994
- [Coupez00] **T. Coupez**, « Génération de maillage et adaptation de maillage par optimisation locale », *Revue européenne des éléments finis*, vol.9, pp.403-423, 2000

-
- [Dendy82] **J. E. Dendy Jr.**, « Black box multigrid ». *J. Comput. Phys.*, vol.48, pp.366-386, 1982
- [El Hadj87] **M. El Hadj**, « Application des méthodes multi-grilles aux calculs des structures élastiques et visco-plastiques ». *Thèse de doctorat*, Ecole Nationale Supérieure des Mines de Paris, Sophia Antipolis, 1987
- [Fedorenko62] **R. P. Fedorenko**, « A relaxation method for solving elliptic difference equation ». *U.S.S.R. Comput. Math. And Math. Phys.*, 1962
- [Feng al. 98] **Y. T. Feng, D. Peric, D.R.J. Owen**, « A multi-grid enhanced gmres algorithm for elasto-plastic problems ». *Int. J. Numer. Methods Eng.*, vol.42, n°8, pp.1441-1462, 1998
- [Germain85] **Y. Germain**, « Modélisation par éléments finis d'écoulement viscoplastique avec frottement. Application au forgeage à chaud ». *Thèse de doctorat*, Ecole Nationale Supérieure des Mines de Paris, Sophia Antipolis, 1985
- [Guillard93] **H. Guillard**, « *Node-nested multigrid with Delaunay coarsening* », Rapport de recherche INRIA n°1898, 1993
- [Hackbusch78] **W. Hackbusch**, « On the multi-grid method applied to difference equations ». *Computing*, vol.20, pp.291–306, 1978
- [Heys et al.05] **J.J. Heys, T.A. Manteuffel, S.F. McCormick, L.N. Olson**, « Algebraic multigrid for higher-order finite elements », *Journal of computational Physics*, vol. 204 , pp.520-532, 2005
- [Iwamura et al.03] **C. Iwamura, F.S. Costa, I. Sbarski, A. Easton et N. Li**, « An efficient algebraic multigrid preconditioned conjugate gradient solver », *Computer Methods in Applied Mechanics and Engineering*, vol.192, pp.2299-2318, 2003
- [Jameson et al.81] **A. Jameson, W. Schmidt, E. Turkel**, « Numerical solution of the Euler equations by finite volume methods using Runge Kutta time stepping schemes ». *AIAA, Fluid and Plasma Dynamics Conference*, 14th, Palo Alto, CA, June 23-25, 1981. 15 p
- [Janka et al.03] **A. Janka et H. Guillard**, « Développement et tests de méthodes multigrille pour l'accélération de la convergence des méthodes de résolution des équations de Navier-Stokes sur maillages non-structurés », *Final report, Dassault contract no. 42G01441, ref. AW/MIT*, 2003
- [Lallemand92] **M. H. Lallemand**, « Algebraic multigrid », *Multigrid method for computational Fluid Dynamics*, ERCIM.-INRIA, Sophia Antipolis, France, 1992

-
- [Le Dain91] **M. A. Le Dain**, « Méthodes multigrille et H-adaptation : application au calcul de structures élastiques bidimensionnelles ». *Thèse de doctorat*, Ecole Nationale Supérieure des Mines de Paris, Sophia Antipolis, 1991
- [Lee98] **C. O. Lee**, « Multigrid methods for the pure traction problem of linear elasticity: mixed formulation ». *SIAM J. Numer. Anal.*, vol.35, n°1, pp.121-145, 1998
- [Li et al.98] **X. S. Li and J. W. Demmel**, « A Scalable Distributed-Memory Sparse Direct Solver for Unsymmetric Linear Systems », *ACM Trans. Mathematical Software*, vol.29, n°2, pp. 110-140, 2003
- [Marie97] **S. Marie**, « Un modèle de parallélisation S.P.M.D. pour la simulation numérique de procédés de mise en forme de matériaux ». *Thèse de doctorat*, École Nationale Supérieure des Mines de Paris, Sophia Antipolis, 1997
- [Mavripilis92] **D. J. Mavripilis**, « Multigrid acceleration of an upwind Euler solver on unstructured meshes ». *American Institute of Aeronautics and Astronautics journal*, vol.33, n°10, pp.1822-1827, 1995
- [Mavripilis95] **D. J. Mavripilis**, « Three dimensional multigrid Reynolds averaged Navier Stokes solver for unstructured meshes », *American Institute of Aeronautics and Astronautics journal*, vol.33, n°3, pp.445-453, 1995
- [Mavripilis97] **D. J. Mavripilis**, « Unstructured grid techniques », *Annu. Rev. Fluid. Mech.*, vol.29, pp.473-514, 1997
- [Mocellin99] **K. Mocellin**, « Contribution à la simulation numérique tridimensionnelle du forgeage à chaud. Etude du contact et calcul multigrille ». *Thèse de doctorat*, Ecole Nationale supérieure des Mines de Paris, Sophia Antipolis, 1999
- [Muttin et al.95] **F. Muttin et J. L. Chenot**, « On a conjugate two-grid method for three dimensional elasticity », *Engineering Computation*, vol.12, pp.3-20, 1995
- [Okusanya02] **T. O. Okusanya**, « Algebraic multigrid for stabilized finite element discretizations of the Navier Stokes equations », *Doctoral Thesis*, Massachusetts Institute of Technology, 2002
- [Olliver-Gooch95] **C. F. Olliver-Gooch**, « Multigrid acceleration of an upwind Euler solver on unstructured meshes ». *American Institute of Aeronautics and Astronautics journal*, vol.33, n°10, pp.1822-1827, 1995

-
- [Paparakakis et al.93] **M. Paparakakis & N. Bitoulas**, « Accuracy and effectiveness of preconditioned conjugate gradient algorithms for large and ill-conditioned problems », *Computers Methods in Applied Mechanics and engineering*, vol.109, pp.219-232, 1993
- [Perchat00] **E. Perchat**, « MINI-Élément et factorisations incomplètes pour la parallélisation d'un solveur de Stokes 2D. Application au forgeage ». *Thèse de doctorat*, Ecole Nationale Supérieure des Mines de Paris, Sophia Antipolis, 2000
- [Pereira et al.06] **F. H. Pereira, S. L. L. Verardi et S. I. Nabeta**, « A fast algebraic multigrid preconditioned conjugate gradient solver », *Applied Math. And Comp.*, vol.179, pp.344-351, 2006
- [Rey et al.05] **B. Rey, K. Mocellin, L. Fourment**, « A node-nested Galerkin multigrid method for metal forging simulation », *8th European Multigrid Conference EMG2005*, Scheveningen The Hague, The Netherlands, September 27-30, 2005
- [Rey et al.07] **B. Rey, K. Mocellin, L. Fourment**, « A node-nested Galerkin multigrid method for metal forging simulation », *Computing and Visualization in Science*, Springer Berlin / Heidelberg, DOI 10.1007/s00791-006-0054-5, Published online: 6 January 2007
- [Ruge et al.87] **J. Ruge, K. Stüben**, « Algebraic Multigrid (AMG) ». S.F. McCormick, Editor, *Multigrid Methods Frontiers in Applied Mathematics* Vol.5, SIAM, Philadelphia, PA, pp.73-130, 1987
- [Sheng et al.95] **C. Sheng, L. K. Taylor, D. L. Whitfield**, « Multigrid algorithm for three dimensional incompressible high Reynolds number turbulent flows ». *AIAA*, vol.33, pp.2073-2079, 1995
- [South et al.76] **J. C. South, A. Brandt**, « Application of a multi-level grid method to transonic flow calculations ». *NASA Langley Research Center*, ICASE Report 76-8, Hampton, Virginia, 1976
- [Soyris90] **N. Soyris**, « Modélisation tridimensionnelle du couplage thermique en forgeage à chaud ». *Thèse de doctorat*, Ecole Nationale Supérieure des Mines de Paris, Sophia Antipolis, 1990
- [Stüben01] **K. Stüben**, « A review of algebraic multigrid ». *Journal of computation and applied mathematics*, vol.128, pp.281-309, 2001
- [Terzolo04] **L. Terzolo**, « Vers une prédiction de la durée de vie des outils de forge à chaud par la modélisation numérique du régime thermique et de l'usure abrasive ». *Thèse de doctorat*, Ecole Nationale Supérieure des Mines de Paris, Sophia Antipolis, 2004

-
- [Traoré01] **K. Traoré**, « Simulation thermomécanique du laminage circulaire: développement d'une formulation quasi-eulérienne tridimensionnelle sur une architecture parallèle ». *Thèse de doctorat*, Ecole Nationale Supérieure des Mines de Paris, Sophia Antipolis, 2001
- [Vincent97] **C. Vincent**, « Méthodes de gradient conjugué préconditionné et techniques multigrille pour la résolution du problème de Stokes par éléments finis mixtes Q1-P0 stabilisés. Applications ». *Thèse de doctorat*, Université de provence, Centre Saint Charles, 1997
- [Vuik et al.95] **C. Vuik, P. Wesseling and S. Zeng**, « Krylov subspace and multigrid methods applied to the incompressible Navier-Stokes equations ». *Proceedings of the Seventh Copper Mountain Conference on Multigrid Methods*, Copper Mountain, Col., April 2-7, pp.737-753, 1995
- [Waren et al.93] **A. Waren & D. Silvester**, « Fast iterative solution of stabilised Stokes systems Part1 : Using simple diagonal preconditioners ». *SIAM J. Numer. Anal.*, vol.30(3), pp.630-649, 1993
- [Wesseling et al.80] **P. Wesseling and P. Sonneveld**, « Numerical experiments with a multiple grid and a preconditioned Lanczos type method ». *Approximation Methods for Navier–Stokes Problems, Lecture Notes in Mathematics Vol.771*, pp.543–562, 1980
- [Wesseling82] **P. Wesseling**, « Theoretical and practical aspects of a multigrid method ». *SIAM J. Sci. Comput.*, vol.3, pp.387–407, 1982
- [Wesseling92] **P. Wesseling**, « An introduction to multigrid method ». Wiley, Chichester, 1992
- [Wesseling et al.01] **P. Wesseling et C.W. Oosterlee**, « Geometric Multigrid with Applications to Computational Fluid Dynamics », *Journal of Computational and Applied Mathematics*, vol.128, pp.311-334, 2001
- [Zang et al.95] **Y. Zang et R. L. Street**, « A composite multigrid method for calculating unsteady incompressible flows in geometrically complex domains », *Int. J. of Num. meth. in Fluids*, vol.20, pp.341-361, 1995

Résumé

Ce travail s'inscrit dans le cadre de la simulation numérique de procédé de forgeage des métaux. Il s'articule autour de deux thèmes principaux que sont l'adaptation du logiciel Forge3® au procédé de roulage d'anneaux à mi-chaud d'une part, et la réduction des temps de calcul d'autre part.

Après un premier chapitre décrivant le cadre d'étude, est présentée dans une deuxième partie l'étude du procédé de roulage. Grâce à l'introduction d'un nouveau formalisme des outils flottants dans le logiciel, on arrive à reproduire fidèlement la cinématique particulière de l'outillage. Ces développements sont validés par une confrontation directe avec des résultats expérimentaux. Vient ensuite un travail d'identification des paramètres thermiques de la simulation. Ces paramètres sont validés par l'étude de deux types de bagues sensiblement différentes. Les résultats sont très encourageants, la simulation se montrant prédictive sur la trajectoire de l'outillage, ainsi que sur l'évolution de la température de la matrice tout au long d'un cycle, et ce sur les deux symboles différents.

On présente enfin dans une troisième partie une nouvelle méthode de résolution de systèmes linéaires basée sur un algorithme multigrilles trois niveaux. Le solveur multigrille, construit à partir de la librairie PETSc, est fondé sur une technique de déaffinement automatique de maillage qui permet d'obtenir des maillages grossiers emboîtés par nœuds. Après une étude paramétrique permettant la bonne configuration du solveur, on vérifie numériquement sa convergence asymptotique linéaire. Les performances obtenues sur différents cas de forgeage sont édifiantes, avec une division du temps de résolution d'un système linéaire par un facteur 6, pour une division du temps de calcul total de la simulation par un facteur 3 pour des maillages d'environ 60000 nœuds.

Mots clefs : forgeage, éléments finis, roulage, outils flottants, multigrilles

Abstract

This work deals with the numerical simulation of the metal forging processes. It is connected to two principal themes: adapting the software Forge3® to the warm ring rolling forging process and reducing the time computation of the simulation.

The first chapter describes the framework and the second part is dedicated to the study of the ring rolling process. Thanks to the introduction of a new definition of the floating dies within the software, it is then possible to compute precisely the particular tool kinematics. These improvements are confirmed by a direct comparison with experimental results. Then an identification work of the thermal parameters takes place. These parameters are validated by the study of two slightly different rings. The results are very encouraging; the simulation seems to predict well the inner die's trajectory and also the temperature evolution of the outer die during a whole cycle production. These results are true for both of the different types of ring.

The last part presents a new method for resolving linear system based on three-level multigrid algorithm. The multigrid solver is based on the PETSc library and an automatic mesh coarsening technique that helps obtaining node nested coarser meshes. A parametrical study is carried out in order to reach an optimized configuration of the solver. The linear rate of convergence of the method is numerically proved. We obtain substantial results on different forging cases with reduction of the linear system computation by a factor 6 for a 60000 nodes system. This corresponds to a division of the global time simulation by a factor 3.

Keywords : forging, finite elements, ring rolling, floating dies, multigrid
