



HAL
open science

Modélisation électro-thermomécanique et lois d'écoulement neuronales pour la simulation du comportement des alliages métalliques lors d'essais Gleeble

Pierre Tize Mha

► **To cite this version:**

Pierre Tize Mha. Modélisation électro-thermomécanique et lois d'écoulement neuronales pour la simulation du comportement des alliages métalliques lors d'essais Gleeble. Mécanique des matériaux [physics.class-ph]. Institut National Polytechnique de Toulouse - INPT, 2023. Français. NNT : 2023INPT0097 . tel-04352955

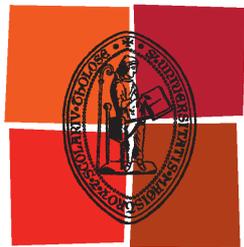
HAL Id: tel-04352955

<https://theses.hal.science/tel-04352955>

Submitted on 19 Dec 2023

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.



Université
de Toulouse

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Institut National Polytechnique de Toulouse (Toulouse INP)

Discipline ou spécialité :

Génie Mécanique, Mécanique des Matériaux

Présentée et soutenue par :

M. TIZE MHA PIERRE

le mardi 7 novembre 2023

Titre :

Modélisation électro-thermomécanique et lois d'écoulement neuronales
pour la simulation du comportement des alliages métalliques lors d'essais
Gleeble

École doctorale :

Mécanique, Energétique, Génie civil, Procédés (MEGeP)

Unité de recherche :

Laboratoire Génie de Production de l'ENIT (E.N.I.T-L.G.P.)

Directeur(s) de Thèse :

M. OLIVIER PANTALÉ

M. AMÈVI TONGNE

Rapporteurs :

M. CHRISTOPHE DESRAYAUD, ENSM ST ETIENNE

M. NICOLAS RANC, ENSAM - ARTS ET METIERS PARISTECH

Membre(s) du jury :

MME SALIMA BOUVIER, UNIVERSITE DE TECHNOLOGIE DE COMPIEGNE, Présidente

M. AMÈVI TONGNE, ECOLE NATIONALE D'INGENIEURS DE TARBES, Membre

M. MOHAMMAD JHAZI, ECOLE DE TECHNOLOGIE SUPERIEURE QUEBEC, Membre

M. OLIVIER PANTALÉ, ECOLE NATIONALE D'INGENIEURS DE TARBES, Membre

Table des matières

Table des matières	i
Table des figures	v
Abréviations et notations	ix
Remerciements	xi
Résumé	xv
Introduction et positionnement de l'étude	1
1 Caractérisation thermomécanique des matériaux	5
1.1 Introduction et état de l'art	6
1.2 Mise en forme des métaux à haute température	8
1.2.1 Traitements thermomécaniques lors d'un procédé de mise en forme	8
1.2.2 Types de comportement	9
1.3 Déformation à chaud et recristallisation	10
1.3.1 Aspects pilotant l'écrouissage d'un matériau	11
1.3.2 Restauration dynamique (DRV)	13
1.3.3 Recristallisation dynamique (DRX)	14
1.3.4 Restauration statique et recristallisation statique	18
1.3.5 Recristallisation métadynamique	20
1.4 Campagne expérimentale sur Gleeble	20
1.4.1 Présentation du dispositif Gleeble	20
1.4.2 Matériau d'étude (3Cr2Mo)	21
1.4.3 Résultats des essais de compression sur le 3Cr2Mo	22
1.5 Conclusions du chapitre	25

2	Modélisation du comportement non-linéaire des matériaux	27
2.1	Forme générale d'une loi de comportement	28
2.2	Modèles de comportement	29
2.2.1	Modèles quasi-statiques	29
2.2.2	Modèles découplés	31
2.2.3	Modèles couplés	33
2.2.4	Modèles semi-empiriques	37
2.3	Modèle phénoménologique (PTM)	42
2.3.1	Formulation du modèle PTM	43
2.3.2	Procédure d'identification	43
2.4	Identification du comportement du 3Cr2Mo	44
2.4.1	Identification du modèle de Johnson–Cook	44
2.4.2	Identification du modèle modifié de Zerilli–Armstrong	47
2.4.3	Identification du modèle de Hensel–Spittel	49
2.4.4	Identification du modèle d'Arrhenius	49
2.4.5	Identification du modèle PTM	54
2.5	Comparaison des modèles identifiés	55
2.6	Conclusions du chapitre	57
3	Loi d'écoulement / Réseau de Neurones Artificiels (RNA)	59
3.1	Les Réseaux de Neurones Artificiels	60
3.1.1	Bref historique de la méthode	60
3.1.2	Émergence des réseaux de neurones dans la technique	61
3.1.3	Perceptron simple	62
3.1.4	Le perceptron multicouches	66
3.2	Loi de comportement basée sur un RNA	70
3.2.1	Architecture du réseau de neurones	70
3.2.2	Pré et post traitement des données	70
3.3	Validation du modèle RNA sur un 42CrMo4	73
3.3.1	Génération des données d'apprentissage	73
3.3.2	Apprentissage et analyse de la performance du RNA	74
3.4	Identification de la loi d'écoulement du 3Cr2Mo	76
3.4.1	Apprentissage et prédiction du réseau	76
3.4.2	Comparaison des modèles analytiques avec le RNA	77
3.4.3	Capacité d'interpolation/extrapolation des modèles	79
3.5	Conclusions du chapitre	84

4	Simulations numériques thermomécaniques	85
4.1	Implémentation d'une loi d'écoulement via une subroutine dans Abaqus	86
4.1.1	Implémentation numérique du modèle de Johnson–Cook	88
4.1.2	Implémentation du modèle de Zerilli–Armstrong Modifié (MZA)	90
4.1.3	Implémentation du modèle PTM	90
4.1.4	Implémentation du modèle RNA	91
4.2	Validation numérique du modèle RNA	92
4.2.1	Essai de rétrécissement d'une barre circulaire	92
4.2.2	Essai d'impact de Taylor	95
4.3	Simulation d'un essai de compression	97
4.3.1	Simulation basée sur le modèle de Johnson–Cook	97
4.3.2	Simulation basée sur le modèle MZA	98
4.3.3	Simulation basée sur le modèle PTM	99
4.3.4	Simulation basée sur le modèle RNA	99
4.3.5	Comparaison des modèles de comportement	100
4.4	Recristallisation dynamique	101
4.4.1	Modèle RNA pour le filtrage des données	102
4.4.2	Conditions critiques de la DRX	104
4.4.3	Cinétique de la DRX	113
4.5	Simulation numérique de la DRX	117
4.5.1	Données expérimentales	117
4.5.2	Modèle EF et implémentation de la DRX	119
4.5.3	Effet de la température et de la vitesse de déformation sur la DRX	120
4.5.4	Effet de la température et de la vitesse de déformation sur la taille de grain	122
4.6	Conclusions du chapitre	124
5	Modélisation électro-thermomécanique d'un essai Gleeble	125
5.1	Gradient de température lors d'un essai Gleeble	126
5.2	Modélisation électro-thermique	128
5.2.1	Régulation thermique sur Gleeble	128
5.2.2	Équations électro-thermiques	130
5.2.3	Conditions aux limites du modèle	130
5.2.4	Algorithme de contrôle de température	132
5.2.5	Modélisation électro-thermique	134
5.3	Modélisation thermomécanique	135
5.4	Modélisation électro-thermomécanique	140
5.5	Conclusions du chapitre	142

Conclusion générale et perspectives	145
A VUHARD du modèle de Johnson–Cook	149
B VUHARD du modèle PTM	155
C VUHARD du modèle RNA	163
D UAMP pour le contrôle du courant	169
E Code Python pour l'identification du modèle PTM	175
Bibliographie	179
Index	191

Table des figures

1.1	Évolution des procédés de mise en forme à chaud	9
1.2	Position du TTM dans la chaîne de production	10
1.3	Courbes contrainte-déformation caractéristiques pour deux types de SFE	11
1.4	Courbes contrainte-déformation de trois types de réponses d'un matériau	12
1.5	Étapes de la restauration	13
1.6	Changements de microstructure pendant la restauration dynamique	14
1.7	Évolution de la microstructure pendant la recristallisation	15
1.8	Évolution de la microstructure pendant une déformation à chaud d'un matériau	15
1.9	Microstructure de la CDRX	17
1.10	Représentation schématique de la microstructure au cours d'une DDRX	19
1.11	DDRX en collier dans un superalliage In718	19
1.12	Simulateur thermomécanique Gleeble-3800	20
1.13	Cycles des essais effectués sur Gleeble-3800	22
1.14	Courbes contrainte-déformation du 3Cr2Mo	23
1.15	Micrographies de l'acier 3Cr2Mo	24
2.1	Modules de base pour le comportement rhéologique des matériaux	29
2.2	Représentation schématique du modèle de Bingham	30
2.3	Évolution de quelques modèles quasi-statiques	31
2.4	Évolution de la loi de Lurdos / Voce	36
2.5	Évolution de I_1 et S_1 en fonction de la déformation	42
2.6	Prédictions du modèle de Johnson-Cook	46
2.7	Prédictions du modèle modifié de Zirilli-Armstrong	48
2.8	Prédictions du modèle de Hensel-Spittel	50
2.9	Fonctions pour identification du modèle AR	51
2.10	Paramètres du modèle d'Arrhenius en fonction de la déformation.	52

2.11	Prédictions du modèle d'Arrhenius	53
2.12	Détermination des C_k^l paramètres du modèle PTM [Tize Mha <i>et al.</i> , 2022].	55
2.13	Prédictions du modèle PTM	56
3.1	Perceptron simple de Rosenblatt	63
3.2	Perceptron à une couche cachée, 2 entrées et 1 sortie, de type 2-2-1.	66
3.3	Perceptron multicouches à (k)-couches cachées avec 3 entrées et 1 sortie.	67
3.4	Architecture générale d'un RNA modélisant une loi de comportement.	70
3.5	Comparaison de la précision de six modèles de RNA.	75
3.6	Convergence des valeurs prédites par le modèle 3-15-7-1-sig	75
3.7	Convergence du RNA 3-15-7-1-sig lors de l'apprentissage.	77
3.8	Comparaison entre expériences et prédictions du modèle RNA	78
3.9	Capacité d'interpolation des modèles identifiés	80
3.10	Capacité d'extrapolation des modèles identifiés	82
4.1	Organigramme général d'un code EF	87
4.2	Organigramme général de l'algorithme de Retour-Radial	88
4.3	Modèle numérique pour le rétrécissement d'une barre	93
4.4	Contrainte de von Mises $\bar{\sigma}$ pour le rétrécissement	93
4.5	Déformation équivalente $\bar{\epsilon}$ et contrainte de von Mises $\bar{\sigma}$ vs. déplacement axial	94
4.6	Modèle numérique d'impact de Taylor	95
4.7	Déformation équivalente $\bar{\epsilon}$ pour l'essai d'impact de Taylor	96
4.8	Modélisation numérique de l'essai de compression du P20	97
4.9	Déformation et contrainte de von Mises – Modèle de Johnson–Cook	98
4.10	Déformation équivalente et contrainte de von Mises – Modèle MZA	98
4.11	Déformation équivalente et contrainte de von Mises – Modèle PTM	99
4.12	Déformation équivalente et contrainte von Mises – Modèle RNA	100
4.13	Comparaison des modèles identifiés	100
4.14	Architecture du modèle RNA utilisé pour le filtrage des données.	103
4.15	Filtrage des données à l'aide du RNA pour 10^{-3} s^{-1} et 1050°C	103
4.16	Exemple de d'une courbe de durcissement.	104
4.17	Conditions critiques de la recristallisation dynamique	105
4.18	Prédiction d'écroûissage basée sur l'approche de Najafizadeh et Jonas	106
4.19	Prédiction des courbes d'écroûissage basée sur un polynôme de degré 5.	112
4.20	Diagramme des trois étapes de la DRX	113
4.21	Prédiction de la DRX-version modifiée de Najafizadeh	116
4.22	Prédiction de la DRX basée sur un polynôme de degré 5	116

4.23	Effet de la température sur la DRX du 3Cr2Mo	117
4.24	Effet de la vitesse de déformation sur la DRX du 3Cr2Mo	118
4.25	X_{drx} prédites pour $T_0 = 1150^\circ\text{C}$ pour différentes réduction et vitesses de déformation	120
4.26	Évolution de la X_{drx} et de la taille de grain pour $\dot{\epsilon} = 10^{-2}\text{s}^{-1}$	121
4.27	X_{drx} prédites pour $\dot{\epsilon} = 10^{-2}\text{s}^{-1}$ pour différentes réduction et températures initiales	121
4.28	Évolution de la X_{drx} et de la taille de grain pour $T_0 = 1150^\circ\text{C}$	122
4.29	Comparaison entre valeurs expérimentales et prédites de X_{drx}	123
5.1	Utilisation de trois thermocouples fixés sur un échantillon	127
5.2	Températures enregistrées par les trois thermocouples fixés sur l'échantillon.	127
5.3	Modélisation d'un essai de compression sur Gleeble	129
5.4	Modélisation d'un essai de traction sur Gleeble	129
5.5	Propriétés physiques du matériau étudié	131
5.6	Modèle numérique axisymétrique avec 3 thermocouples	133
5.7	Cycle thermique d'essai sur Gleeble	133
5.8	Organigramme de la routine UAMP	134
5.9	Comparaison entre prédiction numérique et valeurs expérimentales	134
5.10	Comparaison entre prédiction et valeurs expérimentales de la température	135
5.11	Distribution de la température	136
5.12	Contrainte de von Mises et déformation plastique pour une température fixe	137
5.13	Effet de réduction et de distribution de la température sur la déformation plastique	138
5.14	Contrainte de von Mises et déformation plastique pour une vitesse fixe	139
5.15	Modélisation 3D du couplage électro-thermomécanique	140
5.16	Distribution de la température	141
5.17	Déformation plastique lors des simulation 3D TM et ETM	143
5.18	Contrainte de von Mises lors des simulation 3D TM et ETM	144
D.0.1	Cône de révolution	170

Abréviations et notations

Abréviations

ADAM	Adaptive Moment Estimation (optimiseur)
AR	Arrhenius (loi d'écoulement)
Cd	Cadnium
CPU	Central processing unit
CDRX	Continuous Dynamic Recrystallization
EBSD	Diffraction Électronique Rétrodiffusée
DRX	Dynamic Recrystallization Recristallistion Dynamique
JMAK	Johnson-Mehl-Avrami-Kohnogorov
DDRX	Discontinuous Dynamic Recrystallization
SRX	Static Recrystallization
DRC	Dynamic Reccovery
DSI	Dynamic Systems Inc
FE	Finite Element
Ga	Gallium
GDRX	Geometric Dynamic Recrystallization
GAN	Générateurs Adverses
HS	Hensel-Spittel (loi d'écoulement)
JC	Johnson-Cook (loi d'écoulement)
LVDT	Linear Variable Differential Transformer
MZA	Modified Zerilli-Armstrong
Mo	Molybdène
Nb	Niobium
Pb	Plomb
PTM	Pierre Tize Mha (loi d'écoulement)
RNA	Réseaux de Neurones Artificiels
RNC	Réseaux de Neurones Convolutif
RNR	Réseaux de Neurones Récurrent
RX	Recrystallization
SFE	Stacking Fault Energy (en anglais)
SIBM	Strain Induced grain Boundaries Migration
Sn	Étain
Ta	Tantale
TTM	Traitement thermomécanique
ZA	Zerilli-Armstrong (loi d'écoulement)
W	Tungstène

WH Work Hardening durcissement en français

Notations

$\chi_{drx}^{e,p}$	DRX's Volume fraction : e, p pour expérimental et prédiction respectivement
$\varepsilon, \dot{\varepsilon}$	Déformation, Vitesse de déformation
σ, T	Contrainte, Température
T_m	Température de fusion
E_{MAR}	Erreur Relative Absolue Moyenne
E_{RMS}	Erreur Quadratique Moyenne de la Racine
E, ν	Module de Young, Coefficient de poisson
ρ	Densité volumique
C_p, h_c	Chaleur spécifique, coefficient de conductivité électrique
I, J	Intensité, Densité de courant électrique
R, r	Résistance électrique, Résistivité électrique
U	Tension électrique
P	Puissance

Remerciements

À *Dieu*, source infinie de grâce et de sagesse, en ce jour de réussite et de triomphe, je tiens à m'incliner humblement devant toi, reconnaissant que sans ton amour, ta guidance et tes bénédictions, cette thèse n'aurait pas été possible. Les réussites individuelles sont souvent le résultat d'efforts collectifs. Comme le disait Helen Keller, '*Alone we can do so little; together we can do so much.*' Les pages suivantes sont dédiées à exprimer ma profonde gratitude envers celles et ceux qui ont partagé leur expertise, leur soutien et leur inspiration tout au long de ce voyage académique.

Je tiens tout d'abord à exprimer ma profonde gratitude envers Monsieur Olivier Pantalé, directeur de cette thèse dont la guidance experte et le dévouement ont été essentiels à la réalisation de cette thèse. Son engagement constant à l'excellence académique, sa patience et ses conseils éclairés ont été une source d'inspiration. Ses compétences et rigueurs scientifiques, qualités humaines ont sans doute forgé en moi l'esprit d'un scientifique en devenir. Je ne pourrais oublier les bases de la programmation dans divers langages et l'esprit de curiosité qu'il a incités en moi. Merci pour toutes ces compétences acquises. Mes remerciements vont également envers Monsieur Amévi Tongne mon co-directeur, dont l'expertise et les conseils avisés ont enrichi ce travail de manière significative. Sa disponibilité, sa perspicacité et son soutien inestimable ont grandement contribué à l'aboutissement de ce projet. La collaboration avec Olivier Pantalé et Amévi Tongne a été une expérience enrichissante, marquée par l'apprentissage constant et le partage de connaissances. Leur impact sur mon parcours académique restera gravé dans ma mémoire, et je suis reconnaissant d'avoir eu l'opportunité de bénéficier de leur encadrement bienveillant.

C'est avec une profonde gratitude que je tiens à remercier Monsieur Mohammad Jahazi pour avoir ouvert les portes de son équipe de recherche et m'avoir accueilli chaleureusement pendant près de 14 mois à l'École de Technologie Supérieure (ÉTS) de Montréal, au Canada. Cette période a été non seulement fructueuse sur le plan professionnel, mais elle a également été marquée par des moments d'apprentissage enrichissants et la création de liens précieux. Monsieur Jahazi a joué un rôle essentiel dans mon parcours académique en me permettant de faire partie de son équipe CM2P. Sa vision éclairée, son soutien constant et son expertise ont été une source d'inspiration inestimable. Un merci sincère s'adresse également à tous les membres de l'équipe de recherche CM2P. Ensemble, nous avons formé une communauté où l'apprentissage et la découverte ont été au cœur de notre expérience collective. Ces 14 mois resteront gravés dans ma mémoire comme une période de croissance personnelle et professionnelle, et je suis reconnaissant envers le Professeur Jahazi et toute son équipe de recherche pour cette opportunité exceptionnelle.

La réalisation de cette soutenance a été grandement rendue possible grâce à l'engagement exceptionnel de Madame Salima Bouvier, qui a accepté de présider le jury de ma thèse avec une grande expertise. Les échanges que nous avons eus se sont distingués par la pertinence des questions posées, et les réponses ont permis d'éclairer les zones d'ombre identifiées par les rapporteurs (Christophe Desrayaud et Nicolas Ranc). Je tiens à exprimer ma gratitude envers ces derniers pour avoir examiné

le manuscrit avec attention et pour leurs commentaires constructifs qui ont contribué à améliorer la version finale du document. Les conseils précieux dispensés après la soutenance, jetant les bases de mon parcours professionnel, resteront gravés dans ma mémoire.

Je remercie l'ensemble du personnel de l'ENIT-LGP et mes collègues du laboratoire dont leurs contributions ont été particulièrement significatives tout au long de ces années de recherche. Leur soutien et leur collaboration ont véritablement marqué cette période, et je tiens à les remercier chaleureusement pour leur amitié.

Ma reconnaissance profonde envers les membres de ma famille au **Cameroun** qui ont eu la patience de me voir loin d'eux physiquement pendant trois années mais proche d'eux spirituellement car leurs prières ont été à mes côtés tous les jours. Leur compréhension, leur encouragement et leur soutien ont été des piliers essentiels qui ont contribué à rendre possible cette période de recherche.

Ma chère maman **KOUVOU Esther**, c'est avec une immense joie et une profonde gratitude que je prends le temps de t'écrire ces quelques lignes. Aujourd'hui, ma thèse de doctorat a été soutenue avec succès, et je voulais prendre un moment pour exprimer tout mon amour et ma reconnaissance envers toi. Ta présence constante dans ma vie a été une source inépuisable de soutien et d'inspiration. Ton amour inconditionnel a été le moteur qui a alimenté toutes mes ambitions, et c'est grâce à toi que j'ai pu atteindre ce moment significatif de ma carrière académique. Les longues heures passées à travailler sur ma thèse ont parfois été épuisantes, mais ton encouragement indéfectible m'a donné la force nécessaire pour persévérer. Ton soutien moral et émotionnel a été une boussole qui m'a guidé à travers les moments difficiles. Aujourd'hui, en soutenant ma thèse, je sais que c'est aussi une victoire pour toi, une récompense pour toutes les fois où tu as sacrifié ton temps et ton énergie pour me permettre de poursuivre mes rêves. Ta foi en moi a été le catalyseur qui a transformé mes ambitions en réalisations concrètes. Je veux que tu saches à quel point ton rôle dans ce succès est fondamental. Ta gentillesse, ta sagesse et ton amour infini sont des cadeaux qui dépassent de loin tout ce que j'aurais pu accomplir seul. Cette thèse est autant la tienne que la mienne. Alors, merci, du fond du cœur. Merci pour ta patience, ton dévouement et ton amour incommensurable. Cette réussite est le fruit de notre partenariat, de notre voyage ensemble à travers les défis et les triomphes. Je t'aime plus que les mots ne peuvent l'exprimer, et je suis honoré de pouvoir partager ce moment de bonheur avec toi.

Mon feu père **ZRA Joseph**, aujourd'hui, alors que ma thèse de doctorat atteint son apogée, je ressens à la fois la joie du succès et la douleur de ton absence. Dans ce moment de triomphe, je voudrais dédier ces accomplissements à la personne qui aurait été la plus fière de moi, à toi, mon père regretté. Ton soutien indéfectible a toujours été ma force motrice. Les valeurs que tu as inculquées, ton amour pour l'apprentissage et ta sagesse infinie ont été les fondations sur lesquelles j'ai construit cette thèse. Chaque étape de ce parcours académique a été guidée par les leçons que tu m'as enseignées. Je m'imagine partager avec toi les détails de cette recherche, discuter des découvertes passionnantes et recevoir tes conseils précieux. Bien que tu ne sois pas physiquement présent, je ressens toujours ton influence bienveillante qui plane sur chaque page de cette thèse. Ce moment de succès est aigre-doux, car j'aurais tant aimé le célébrer avec toi. Néanmoins, je choisis de dédier ce triomphe à ta mémoire, reconnaissant que ton héritage continue de vivre à travers moi. J'espère que, d'où tu es, tu vois le fruit de nos conversations, de nos rêves partagés et de ton amour éternel. Papa, ce succès est le tien autant que le mien. Merci pour ton inspiration infinie, pour avoir allumé en moi la passion du savoir et pour avoir été le modèle d'un parent aimant et encourageant. Ta fierté aurait été ma plus grande récompense, et je garde cette pensée précieuse dans mon cœur aujourd'hui.

Enfin j'exprime ma sincère gratitude à ma communauté chrétienne de l'EEV de Tarbes pour le soutien inestimable qu'elle m'a offert tout au long de ces années de recherche. Votre encouragement, vos prières et votre solidarité ont été des sources d'inspiration, de motivation et de réconfort. Merci du fond du cœur pour le soutien précieux de cette communauté qui a contribué à rendre cette période de recherche plus significative et enrichissante.

Résumé

Résumé

Le recours à des blocs massifs fabriqués à base des matériaux faiblement carbonés, en raison de leurs propriétés mécaniques robustes, de leur résistance à la corrosion et de leur aptitude à supporter des contraintes dynamiques, est répandu dans divers secteurs industriels tels que l'automobile, le ferroviaire, l'aéronautique et l'aérospatiale. La complexité du processus de mise en forme de ces blocs requiert une analyse approfondie à chaque étape pour une optimisation tant en termes de coûts que de temps. Cependant, les expériences physiques nécessaires à cette étude sont coûteuses et les entreprises cherchent à minimiser les erreurs dans un contexte de concurrence accrue. Pour relever ces défis, la modélisation numérique non linéaire est devenue prépondérante pour simuler ces processus industriels complexes. Cette transition vers la simulation numérique nécessite une représentation à l'échelle du laboratoire, réalisée à l'aide de dispositifs expérimentaux comme le simulateur thermomécanique Gleeble. Cet appareil est prisé pour sa capacité à effectuer des essais électro-thermomécaniques entièrement couplés, statiques et dynamiques, reproduisant ainsi les phénomènes industriels. Cependant, la simulation numérique de ces essais demeure un défi, exigeant la création de modèles numériques électro-thermomécaniques entièrement couplés. Cette thèse vise précisément à relever ce défi en proposant une démarche en plusieurs étapes. Le premier chapitre établit le contexte en caractérisant le matériau d'étude et en présentant le simulateur thermomécanique Gleeble. Le second chapitre est consacré plus particulièrement à une comparaison des lois de comportement analytiques et propose une nouvelle formulation : le modèle PTM. Le troisième chapitre explore l'utilisation de réseaux de neurones artificiels pour modéliser le comportement, contournant ainsi l'étape d'identification des paramètres dans les modèles analytiques. Une fois les modèles établis, dans le quatrième chapitre, une simulation est réalisée pour reproduire un essai de compression thermomécanique effectué sur le Gleeble. L'originalité de cette simulation réside dans sa capacité à prédire la recristallisation dynamique, avec les réseaux de neurones montrant une meilleure prédiction par rapport aux modèles analytiques lors de l'étape d'identification. Enfin, le dernier chapitre traite de la simulation électro-thermomécanique entièrement couplée et propose un algorithme pour prédire la densité de courant à chaque étape de calcul. Cette simulation permet de prévoir le gradient thermique de manière fidèle par rapport au dispositif Gleeble, en suivant l'évolution thermique et la déformation du matériau. Ce chapitre présente une analyse détaillée des phénomènes en jeu sur le dispositif, évitant ainsi la nécessité de mener plusieurs simulations pour des essais complexes.

Mots clés : Gleeble, modélisation électro-thermomécanique, loi de comportement, réseau de neurones, simulation, recristallisation dynamique.

Abstract

The use of massive blocks made from low-carbon steel is widespread in various industrial sectors such as automotive, railways, aerospace, and aviation due to their robust mechanical properties, corrosion resistance, and ability to withstand dynamic stresses. The complexity of shaping these blocks requires a thorough analysis at each stage to optimize both costs and time. However, the physical experiments required for this study are costly, and companies seek to minimize errors in a highly competitive environment. To address these challenges, nonlinear numerical modeling has become crucial for simulating these complex industrial processes. This transition to numerical simulation requires a laboratory-scale representation, achieved using experimental devices such as the Gleeble thermomechanical simulator. This device is valued for its ability to perform fully coupled electrothermomechanical tests, both static and dynamic, thus reproducing industrial phenomena. However, the numerical simulation of these tests remains a challenge, requiring the creation of fully coupled electrothermomechanical numerical models. This thesis aims to tackle this challenge by proposing a multi-step approach. The first chapter sets the context by characterizing the material under study and introducing the Gleeble thermomechanical simulator. The second chapter is dedicated to a comparison of analytical behavior laws and proposes a new formulation : the PTM model. The third chapter explores the use of artificial neural networks to model behavior, bypassing the parameter identification step in analytical models. Once the models are established, in the fourth chapter, a simulation is carried out to replicate a thermomechanical compression test performed on the Gleeble. The novelty of this simulation lies in its ability to predict dynamic recrystallization, with neural networks showing better prediction compared to analytical models during the identification step. Finally, the last chapter deals with fully coupled electrothermomechanical simulation and proposes an algorithm to predict current density at each calculation step. This simulation allows for an accurate prediction of thermal gradient compared to the Gleeble device, tracking the material's thermal evolution and deformation. This chapter provides a detailed analysis of the phenomena at play on the device, avoiding the need for multiple simulations for complex tests.

Key words : Gleeble, electro-thermomechanical modelling, constitutive flow law, neural network, simulation, dynamic recrystallization.

Introduction et positionnement de l'étude

Dans le paysage industriel actuel, où la compétition, l'innovation et la performance sont primordiales, la bataille autour de la fabrication de matériaux à haute valeur ajoutée occupe une place centrale. Les industries du monde entier sont engagées dans une quête incessante de matériaux offrant une résistance exceptionnelle, une durabilité et une polyvalence accrue pour obtenir un avantage compétitif vis à vis de la concurrence [Wang *et al.*, 2023]. Les aciers à teneur moyenne en carbone comme le 3Cr2Mo (souvent définis comme contenant environ 0.3% à 0.6% de carbone en masse) figurent dans cette liste de matériaux et occupent une place importante dans le domaine de l'ingénierie des matériaux. Cette classe d'aciers, avec sa composition soigneusement équilibrée, parvient à combiner harmonieusement la ductilité de l'acier faiblement carboné et la dureté de l'acier hautement carboné. Par conséquent, il trouve ses applications dans une multitude d'industries, de par sa large gamme de propriétés mécaniques adaptées à divers besoins en ingénierie [Murugesan *et al.*, 2023]. Le recours à des blocs massifs d'alliages moyennement carbonés, en raison de leurs propriétés mécaniques robustes, de leur résistance à la corrosion et de leur aptitude à supporter des contraintes dynamiques, est répandu dans divers secteurs industriels tels que le ferroviaire, l'aéronautique, l'aérospatiale et l'automobile où ils sont utilisés comme matrices servant à la production de pare-chocs et de tableaux de bord dans des processus d'injection plastique [He *et al.*, 2023].

Le processus de fabrication de ces blocs massifs commence par la coulée de lingots, suivie par le forgeage à matrice ouverte, et un traitement thermique de trempe et de revenu pour obtenir les propriétés mécaniques souhaitées [Chadha *et al.*, 2017, Chadha *et al.*, 2018a, Murugesan et Jung, 2019]. Au cours des dernières années, afin de répondre à la demande du marché, il a été nécessaire de produire des pièces forgées de plus grande taille. Parallèlement, des conditions plus strictes liées à l'homogénéité chimique, à la dureté, à la taille des grains et aux propriétés mécaniques de la surface au cœur du bloc forgé ont été exigées. Or, parmi les trois étapes de fabrication (coulée, forgeage et traitement thermique), le forgeage est l'étape où les changements microstructuraux les plus importants ont lieu, et qui influencent grandement les propriétés finales pouvant être obtenues [Murugesan *et al.*, 2019, Chadha *et al.*, 2020, Sripada *et al.*, 2022]. Le processus de forgeage à matrice ouverte est fondamentalement un processus de compression à chaud au cours duquel des effets de renforcement du matériau, tels que le durcissement et la précipitation, se produisent simultanément avec des phénomènes d'adoucissement tels que la restauration et la recristallisation sous des charges thermomécaniques statiques et/ou dynamiques [Tavakoli *et al.*, 2019, Tian *et al.*, 2023]. Il a également été montré que des transformations de phase peuvent se produire pendant la déformation et que celles-ci impactent les propriétés du produit forgé.

Un autre défi lié au forgeage des blocs massifs est la présence de porosités au sein du produit forgé. En effet, la fabrication des pièces forgées de qualité et de grande taille est extrêmement difficile en raison des défauts survenant lors de la coulée du lingot, tels que les inhomogénéités dans la microstructure, la macroségrégation et la microségrégation, les porosités et les inclusions, qui

détériorent la qualité des produits. Bien que de nombreuses investigations aient été menées [Loucif *et al.*, 2015, Loucif *et al.*, 2017, Loucif *et al.*, 2018, Loucif *et al.*, 2020], la production de pièces forgées reste un défi, en particulier pour les pièces de grande taille, et la survenue de ces défauts est encore inévitable. Plus la taille du lingot est importante, plus l'inhomogénéité de la microstructure et la ségrégation augmentent, de sorte que leur élimination est difficile même avec des traitements thermomécaniques ultérieurs [Pickering, 2013]. Ainsi, le choix et la conception d'un programme de forgeage approprié sont nécessaires pour réduire ces défauts et améliorer par la suite les propriétés mécaniques et la microstructure.

L'étendue et l'intensité des phénomènes induits pendant les opérations de forgeage dépendent fortement de trois grandeurs macroscopiques, à savoir la déformation ε , la vitesse de déformation $\dot{\varepsilon}$ et la température T [Ebrahimi *et al.*, 2017, Rudra *et al.*, 2019, Shi *et al.*, 2022, Zeng *et al.*, 2022]. Au regard de la taille des blocs, la complexité du processus de mise en forme requiert une analyse approfondie à chaque étape pour une optimisation tant en termes de coûts que de temps. Cependant, les expériences physiques nécessaires à cette étude sont coûteuses et les entreprises cherchent à minimiser les erreurs dans un contexte de concurrence accrue. Pour relever ces défis, la modélisation numérique non linéaire est devenue prépondérante pour simuler ces processus industriels complexes.

Cette transition vers la simulation numérique nécessite une représentation du comportement à l'échelle du laboratoire, réalisée à l'aide de dispositifs expérimentaux comme le simulateur thermomécanique Gleeble. Ce dernier est considéré de nos jours, comme le meilleur simulateur des procédés thermomécaniques industriels à l'échelle du laboratoire. Cet appareil est prisé pour sa capacité à effectuer des essais électro-thermomécaniques couplés, dans des configurations statiques et dynamiques, reproduisant ainsi les phénomènes industriels complexes. Ils combinent trois physiques : l'électrique, la thermique et la mécanique. Dans ces essais, un courant électrique peut être appliqué au matériau pendant qu'il est soumis à un chargement mécanique et/ou thermique. Le dispositif Gleeble permet de contrôler précisément la température du matériau, en un point, pendant l'essai, ce qui permet de simuler des conditions thermiques réalistes. Les variations de température peuvent être rapides ou lentes, en fonction des besoins de l'expérience. Les essais mécaniques incluent l'application de contraintes, de déformations et de vitesses de déformation contrôlées au cours de l'essai.

La simulation numérique des processus physiques au sein de la machine Gleeble, à l'échelle du laboratoire requiert le développement de modèles numériques électro-thermomécaniques couplés. Lors d'essais réalisés sur le dispositif Gleeble, l'un des défis majeurs réside dans la répartition non homogène de la température à l'intérieur de l'éprouvette, tant selon l'axe longitudinal que radial. Les travaux de [Kim *et al.*, 1996] et de [Zhang *et al.*, 2010], ont parmi d'autres, abordé cette problématique. Cette variation de température induit inévitablement une déformation non homogène, incitant ainsi de nombreux chercheurs à s'atteler à la caractérisation et la réduction de ce gradient thermique. Une solution souvent explorée consiste à modifier la géométrie des mors de manière à influencer la répartition du courant électrique à travers l'éprouvette, et par conséquent, à obtenir un gradient de température relativement uniforme. Bien que d'importantes avancées aient été réalisées dans ce domaine en proposant divers dispositifs expérimentaux, l'éradication complète de ce gradient de température demeure un défi. C'est pourquoi la simulation numérique basée sur la méthode des éléments finis est de plus en plus utilisée pour caractériser plus finement les conditions réelles des essais Gleeble. Ces simulations permettent d'estimer la température localement, facilitant ainsi l'identification des facteurs qui influencent l'inhomogénéité de la température et proposant des solutions, comme le démontrent les travaux de plusieurs chercheurs, notamment [Brown *et al.*, 1997] ou [Norris et Wilson, 1999].

Cependant, l'une des limitations de ces simulations numériques des essais Gleeble réside dans les conditions initiales, notamment la température initiale de l'éprouvette. En effet, la plupart de ces travaux considèrent la température initiale de l'éprouvette comme uniforme au début de la simulation,

ce qui ne reflète pas la réalité expérimentale des essais Gleeble. Pour remédier à cette limitation, des essais plus représentatifs prenant en compte des facteurs tels que l'effet de la friction et la température initiale ont été effectués, comme le montrent les travaux de Brennett *et al.* [Bennett *et al.*, 2010] lors d'un essai de compression axisymétrique. Zhang *et al.* ont développé un modèle électro-thermomécanique couplé d'un essai de traction pour déterminer des paramètres d'une loi de comportement en se basant sur la méthode inverse [Zhang *et al.*, 2010]. L'objectif de ce modèle est de prédire la formation des déchirures à chaud à la fin de la solidification et de fissures à des températures plus basses dans l'état solide de l'acier à haute résistance. Cette étude a mis en évidence l'importance de la prise en compte des gradients thermiques, inhérents aux essais Gleeble, pour une caractérisation rhéologique précise des matériaux. Liu *et al.* ont également examiné le problème des gradients thermiques sur Gleeble et les ont incorporés dans des études numériques afin d'examiner l'état de contrainte qui en résulte [Liu *et al.*, 2006]. La limite de leur étude réside dans le fait que des données expérimentales sur les gradients thermiques sont nécessaires pour effectuer l'analyse numérique de l'essai de traction à chaud, impliquant le besoin de plusieurs essais thermiques pour enregistrer les gradients thermiques axiaux. De plus, les gradients thermiques doivent être saisis manuellement à l'aide de partitions dans un modèle éléments finis, ce qui peut affecter la qualité de la simulation.

Certains auteurs se sont attachés à simuler numériquement la partie électro-thermique de Gleeble : Kardoulaki *et al.* [Kardoulaki *et al.*, 2014], Kumar *et al.* [Kumar, 2016], Xiao *et al.* [Xiao *et al.*, 2021]. L'objectif de ce type de simulation est de reproduire de manière numérique la composante électro-thermique des essais Gleeble, ce qui permet d'intégrer l'évolution du matériau pendant la phase de chauffage dans l'analyse thermomécanique. Cette approche de simulation repose sur l'utilisation de deux logiciels ou solveurs distincts, selon la physique considérée. La première étape vise à prédire la répartition de la température à l'intérieur de l'éprouvette en prenant en compte l'interaction entre les effets électriques et thermiques (effet Joule). La seconde étape consiste à extraire les données de température aux nœuds, puis à les transférer vers un autre logiciel ou un autre solveur afin d'effectuer la simulation thermomécanique. Il est important de noter que ce modèle dépasse les approches qui supposent une température homogène au début de la simulation, offrant ainsi une meilleure représentation de la réalité. Cependant, ce type de simulation présente des défis liés à la prise en compte de l'évolution réelle du matériau pendant la phase de chauffage, au cours de laquelle diverses phases du matériau entrent en jeu. Bien entendu, cela ne nécessite pas nécessairement la création d'un modèle qui suit la microstructure du matériau pendant le chauffage, mais plutôt de tenir compte des changements de microstructure qui surviennent pendant le chauffage, tels que les changements de phases, qui modifient les propriétés physiques du matériau et ont une incidence sur la simulation thermomécanique.

Les points énumérés ci-dessus conduisent alors à la nécessité de développer un modèle numérique unifié, capable de reproduire plus fidèlement un essai sur le dispositif Gleeble, permettant ainsi de suivre l'intégralité de l'historique du matériau depuis la phase de chauffage jusqu'à la transformation thermomécanique au cours de l'application des efforts. Ainsi, l'originalité principale de l'approche proposée dans ce travail de thèse, consiste à unifier les différentes physiques (électrique, thermique et mécanique) dans une seule simulation numérique du comportement électro-thermo-mécanique d'un matériau au cours d'un essai sur le simulateur thermomécanique Gleeble.

La structure de cette thèse est articulée en cinq chapitres, chacun jouant son rôle dans la compréhension et la modélisation du comportement des matériaux dans des conditions de déformation et de température spécifiques.

- Le premier chapitre est dédié à un état de l'art et une présentation du dispositif thermomécanique Gleeble. Nous détaillons les phénomènes activés pendant une déformation à chaud, introduisons le matériau d'étude, et présentons le dispositif d'essai Gleeble et les résultats d'une campagne d'essais.

- Le Chapitre 2 se penche sur l'importance de définir une loi de comportement du matériau pour une simulation numérique efficace lors du forgeage ou toute autre procédé de mise en forme des métaux. Nous regroupons et présentons les lois de comportement par catégorie, en identifiant notamment les lois les plus couramment utilisées pour le matériau étudié. Nous proposons, vers la fin de ce chapitre, une nouvelle formulation de loi de comportement qui se distingue par sa flexibilité, lui permettant de s'adapter à divers types d'essais.
- Le Chapitre 3 présente le développement et la mise en œuvre d'un modèle de comportement basé sur un réseau de neurones artificiels (RNA) pour décrire le comportement du matériau, en complément des modèles analytiques. Nous examinons comment cette nouvelle approche, tirant parti des avancées en intelligence artificielle (IA), peut améliorer la précision des simulations numériques en comparaison avec les modèles analytiques traditionnels.
- Une fois les modèles de comportement identifiés, le Chapitre 4 présente la phase d'implémentation numérique de ces modèles dans le code de calcul éléments finis Abaqus pour valider leur performance dans une simulation thermomécanique. Cette étape est réalisée en utilisant des routines utilisateur VUHARD dans lesquelles nous prenons également en considération la recristallisation dynamique et l'évolution de la taille des grains au cours de la simulation.
- Enfin, le Chapitre 5 se consacre à la simulation électro-thermomécanique complètement couplée de l'essai Gleeble. Nous présentons un algorithme spécialement développé pour contrôler la température de l'échantillon tout au long de l'essai, et nous détaillons comment il a été intégré dans le logiciel Abaqus grâce à une routine utilisateur UAMP. Cet algorithme a permis donc d'obtenir un gradient de température (validé expérimentalement) et l'influence sur l'état de contrainte pu être modélisée grâce à l'utilisation du modèle de comportement RNA implémenté dans une routine utilisateur UHARD.
- Ce mémoire se termine par une conclusion et perspectives où une brève rétrospective de cette étude est faite. Nous mettons en lumière les contributions majeures effectuées tout en examinant les limites et les défis non résolus puis nous identifions plusieurs perspectives de recherche qui pourraient susciter un intérêt.

Chapitre 1

Caractérisation thermomécanique des matériaux

1.1	Introduction et état de l'art	6
1.2	Mise en forme des métaux à haute température	8
1.2.1	Traitements thermomécaniques lors d'un procédé de mise en forme	8
1.2.2	Types de comportement	9
1.3	Déformation à chaud et recristallisation	10
1.3.1	Aspects pilotant l'écroissage d'un matériau	11
1.3.2	Restauration dynamique (DRV)	13
1.3.3	Recristallisation dynamique (DRX)	14
1.3.4	Restauration statique et recristallisation statique	18
1.3.5	Recristallisation métadynamique	20
1.4	Campagne expérimentale sur Gleeble	20
1.4.1	Présentation du dispositif Gleeble	20
1.4.2	Matériau d'étude (3Cr2Mo)	21
1.4.3	Résultats des essais de compression sur le 3Cr2Mo	22
1.5	Conclusions du chapitre	25

Dans ce premier chapitre, nous présentons une analyse des aspects clés liés à la transformation thermomécanique des métaux lors de procédés de mise en forme à chaud. Nous commencerons par présenter brièvement quelques procédés de mise en forme des métaux, les méthodes qui permettent leur transformation et les conséquences sur le comportement du matériau.

Ensuite, nous explorons plus profondément ces processus pour offrir une vision d'ensemble des mécanismes de transformation activés lors de la mise en forme thermomécanique des métaux. L'étude se concentre sur les réactions complexes que subissent les métaux face à des conditions thermomécaniques spécifiques, mettant en évidence l'influence de la température T , de la vitesse de déformation $\dot{\epsilon}$ et de la déformation ϵ sur les propriétés mécaniques des matériaux. Les notions de restauration et recristallisation sont également abordées dans cette approche. Pour étudier le comportement de l'acier 3Cr2Mo sous différentes conditions d'essai, nous avons utilisé un simulateur thermomécanique Gleeble-3800. Les essais mécaniques réalisés à l'aide de ce dispositif ont permis d'examiner le comportement du matériau face aux variations des paramètres d'essai.

1.1 Introduction et état de l'art

Les alliages moyennement carbonés, également connus sous le nom d'aciers alliés au carbone moyen, se caractérisent par leur teneur en carbone située généralement dans la plage de 0.3% à 0.6% en masse. Leur composition chimique confère à ces alliages des propriétés mécaniques améliorées par rapport aux aciers à faible teneur en carbone, tout en conservant une certaine ductilité et soudabilité. Ces propriétés font de ces alliages moyennement carbonés des matériaux polyvalents, adaptés à une variété d'applications nécessitant une combinaison de résistance et de ténacité. Par exemple, des blocs forgés de grande taille fabriqués à partir d'aciers alliés moyennement carbonés à haute résistance sont largement utilisés dans l'industrie automobile en tant que matrices pour la production de pare-chocs et de tableaux de bord par le processus d'injection plastique. Le processus de fabrication de ces grands blocs commence par la coulée en lingot, suivi du forgeage à matrice ouverte, puis d'un traitement thermique de trempe et de revenu pour obtenir les propriétés mécaniques souhaitées [Chadha *et al.*, 2017, Chadha *et al.*, 2018a].

Ces dernières années, afin de répondre à la demande du marché, il a été nécessaire de produire des pièces forgées de plus grande taille. Parallèlement, des conditions plus strictes en ce qui concerne l'homogénéité chimique, la dureté, la taille des grains et les propriétés mécaniques de la surface au cœur du bloc forgé ont été exigées. Parmi les trois étapes de fabrication (coulée, forgeage et traitement thermique), le forgeage est celle où les changements de microstructure les plus importants ont lieu, ce qui influence grandement les propriétés finales pouvant être obtenues [Murugesan *et al.*, 2019, Chadha *et al.*, 2020]. Le processus de forgeage à matrice ouverte est fondamentalement un processus de compression à chaud au cours duquel des effets de renforcement, tels que le durcissement (WH) et la précipitation, se produisent simultanément avec des phénomènes d'adoucissement tels que la restauration et la recristallisation sous des charges thermomécaniques statiques et/ou dynamiques [Tian *et al.*, 2023, Tavakoli *et al.*, 2019].

Il a également été démontré que des transformations de phase peuvent survenir pendant la déformation ce qui rend l'étude du matériau encore plus complexe. L'étendue et l'intensité des phénomènes susmentionnés dépendent fortement de trois grandeurs macroscopiques, à savoir la déformation ϵ , la vitesse de déformation $\dot{\epsilon}$ et la température T et la réussite de la fabrication des pièces exige au préalable une caractérisation approfondie de l'alliage [Ebrahimi *et al.*, 2017, Nassr *et al.*, 2018, Shi *et al.*, 2022].

La caractérisation thermomécanique d'un matériau est une démarche expérimentale et analytique visant à étudier le comportement du matériau sous l'effet conjugué de la température et des sollicitations mécaniques. Cette approche est essentielle pour comprendre et prédire le comportement d'un matériau dans des conditions réelles d'utilisation, par exemple, lorsqu'il est soumis à des charges mécaniques à différentes températures. Les principaux points abordés lors d'une caractérisation d'un matériau sont :

- La déformation mécanique : Cela inclut l'étude de la déformation élastique et anélastique du matériau soumis à des chargements mécaniques. Des tests tels que les essais de traction, de compression ou de flexion peuvent être réalisés à différentes températures pour évaluer la réponse mécanique du matériau dans des conditions variées.
- Les contraintes mécaniques : Les contraintes subies par le matériau lors des essais mécaniques à différentes températures sont également évaluées. Les contraintes peuvent varier en fonction de la température en raison des propriétés thermiques du matériau.
- Le coefficient de dilatation thermique : Le coefficient de dilatation thermique du matériau est un paramètre important qui caractérise la variation dimensionnelle du matériau en réponse aux changements de température. Il permet de comprendre comment le matériau se dilate ou se contracte en fonction de la température.

- La conductivité thermique : La conductivité thermique est la capacité d'un matériau à conduire la chaleur. Il est important de connaître cette caractéristique pour évaluer la manière dont le matériau réagit aux variations de température et aux gradients.
- La capacité thermique spécifique : La capacité thermique spécifique mesure la quantité de chaleur nécessaire pour élever la température d'une unité de masse du matériau d'une certaine quantité. Cette caractéristique est cruciale pour comprendre la capacité du matériau à stocker et à libérer de l'énergie thermique.
- Le comportement viscoélastique : Certains matériaux présentent un comportement viscoélastique, ce qui signifie qu'ils présentent à la fois des propriétés élastiques et visqueuses en réponse à une sollicitation mécanique et/ou thermique.
- La température de transition vitreuse : Pour certains polymères et matériaux amorphes, il existe une température de transition vitreuse qui influe sur leur comportement mécanique à différentes températures.

Pour réaliser ces études, des équipements spécifiques tels que des machines d'essais mécaniques associées à des systèmes de chauffage/refroidissement sont utilisés. Les résultats obtenus permettent de modéliser et de simuler le comportement du matériau dans diverses applications industrielles, telles que la conception de structures, les matériaux de construction, l'aérospatiale, l'automobile, etc. Cela permet également de garantir la sécurité et la performance des matériaux dans des environnements thermomécaniques extrêmes.

Lorsqu'un matériau est soumis à des sollicitations mécaniques à des températures élevées (dans la plage de température où la déformation à chaud est étudiée), plusieurs mécanismes peuvent être activés comme mentionné précédemment et servent à accommoder cette déformation. Ces mécanismes dépendent du type de matériau, de sa microstructure et de la température à laquelle il est soumis. Il est essentiel de comprendre ces mécanismes pour adapter les traitements thermomécaniques et concevoir des procédés de fabrication appropriés pour les matériaux destinés à des applications à haute température. La déformation à chaud peut être utilisée pour améliorer la formabilité des matériaux, leur ductilité, leur résistance et d'autres propriétés mécaniques en raison des mécanismes spécifiques activés lors de la déformation à des températures élevées [Sripada *et al.*, 2022, Murugesan et Jung, 2019].

La déformation à chaud fait référence à la déformation plastique d'un matériau à des températures élevées, généralement supérieures à une certaine fraction de sa température de fusion. Cette technique est utilisée pour façonner les matériaux à haute température, souvent dans une plage de température où la déformation plastique est favorisée et où les propriétés mécaniques du matériau sont affaiblies. La déformation à chaud peut être réalisée dans différentes configurations comme le forgeage à chaud, l'extrusion à chaud, le filage à chaud, l'étirage à chaud, etc. Les avantages de la déformation à chaud incluent une meilleure ductilité du matériau à haute température, une réduction de la résistance à la déformation. Cela permet également de réduire les forces de déformation requises, ce qui peut être particulièrement avantageux pour les matériaux à haute résistance ou difficiles à façonner à froid. Il est important de noter que la déformation à chaud doit être effectuée dans une plage de température spécifique, car des températures trop élevées peuvent entraîner une fusion partielle du matériau, tandis que des températures trop basses peuvent entraîner des déformations insuffisantes ou une fragilité accrue du matériau. La déformation à chaud est couramment utilisée dans divers secteurs industriels pour produire des pièces de haute qualité, durables et adaptées à des applications spécifiques. Cela permet également de réduire les étapes de fabrication, d'optimiser les processus de production et d'améliorer l'efficacité globale de la fabrication de pièces métalliques et d'autres matériaux à haute température [Versino *et al.*, 2017, Rudra *et al.*, 2019, Bomarito *et al.*, 2021].

1.2 Mise en forme des métaux à haute température

La déformation à chaud joue un rôle primordial au sein des processus de fabrication industriels, touchant quasiment tous les composants métalliques avec une ou plusieurs modifications réalisées à température élevée. Certains éléments sont utilisés dans leur état brut (comme les plaques épaisses, les rails, les profilés et les pièces matricées), faisant ainsi de la déformation à chaud le déterminant de leur structure et de leurs caractéristiques fonctionnelles. La majorité de ces pièces subissent ensuite des déformations à froid. Dans ce cas, les changements structurels (ou de texture) issus de la déformation à chaud peuvent altérer les conditions de cette étape ultérieure. Dans certains matériaux avec une mémoire de forme, tels que les alliages de titane ou de zirconium, ces modifications peuvent persister dans une certaine mesure jusqu'au stade final du produit. Enfin, les procédés de mise en forme à chaud offrent la possibilité d'influencer l'architecture du matériau. Dans de tels cas, ils sont désignés comme des traitements thermomécaniques.

Les investigations en métallurgie liées à la mise en forme à chaud se heurtent aux mêmes enjeux que ceux rencontrés dans le contexte de la mise en forme à froid, en ce qui concerne la reproduction précise, dans des conditions bien maîtrisées, des vitesses et des trajectoires de déformation caractéristiques des procédés [Montheillet, 2004, Montheillet, 2008]. De surcroît, alors que les gammes de températures employées pour le façonnage des métaux sont relativement reproductibles en laboratoire, la nature des opérations industrielles et les dimensions des pièces entraînent généralement des gradients thermiques significatifs, ainsi que des variations rapides au fil du temps (refroidissement résultant des pertes de chaleur dans les outils ou à travers les surfaces exposées, ou inversement, auto-échauffement associé à la déformation plastique). Ces évolutions s'avèrent complexes à reproduire fidèlement à l'aide d'essais instrumentés, qui eux-mêmes sont soumis aux mêmes phénomènes [Montheillet et Desrayaud, 2009]. En outre, l'étude des microstructures associées à la mise en forme à chaud se révèle délicate en raison de la difficulté à distinguer les évolutions dynamiques (en cours de déformation) des évolutions post-dynamiques (après déformation). En effet, même dans un contexte de laboratoire, les systèmes de refroidissement les plus avancés ne parviennent pas à figer de manière certaine l'état du matériau résultant de la déformation.

Il convient de rappeler que la plage de déformation qualifiée de *à chaud* est établie de manière empirique avec $T > 0.5 \times T_m$, où T représente la température absolue du matériau et T_m sa température absolue de fusion (également désignée comme la température de solidus). Ce domaine peut débuter, pour certains matériaux à faible point de fusion (Ga, Cd, Sn, Pb), à la température ambiante ou même en deçà de celle-ci. A l'inverse, les plages de déformation qualifiées de *tiède* et de *froide* s'étendent jusqu'à des températures substantielles pour les métaux dotés de points de fusion extrêmement élevés (Nb, Mo, Ta, W).

1.2.1 Traitements thermomécaniques lors d'un procédé de mise en forme

Jusqu'à la période médiane du 19^{ème} siècle, le matériau était considéré comme une boîte noire, dépourvue de structure interne (voir Figure 1.1(a)) et une mise en forme avait pour objectif exclusif d'atteindre les dimensions souhaitées pour le produit final (épaisseur de la tôle lors du laminage, par exemple). A partir des années 1960, les acteurs industriels ont entrepris d'utiliser la mise en forme à chaud pour optimiser la microstructure du produit final. Le choix de conditions appropriées de laminage à chaud a permis d'affiner la taille des grains des tôles, améliorant ainsi leur ténacité (voir Figure 1.1 (b)). La combinaison de passages de laminage à froid et de recuits a permis d'optimiser la texture des feuilles d'aluminium, ce qui a amélioré leur aptitude à l'emboutissage. Progressivement, notamment pour les pièces à haute valeur ajoutée, telles que les disques de turbine forgés, l'idée d'un contrôle total de l'architecture du matériau à une échelle de plus en plus fine a pris forme :

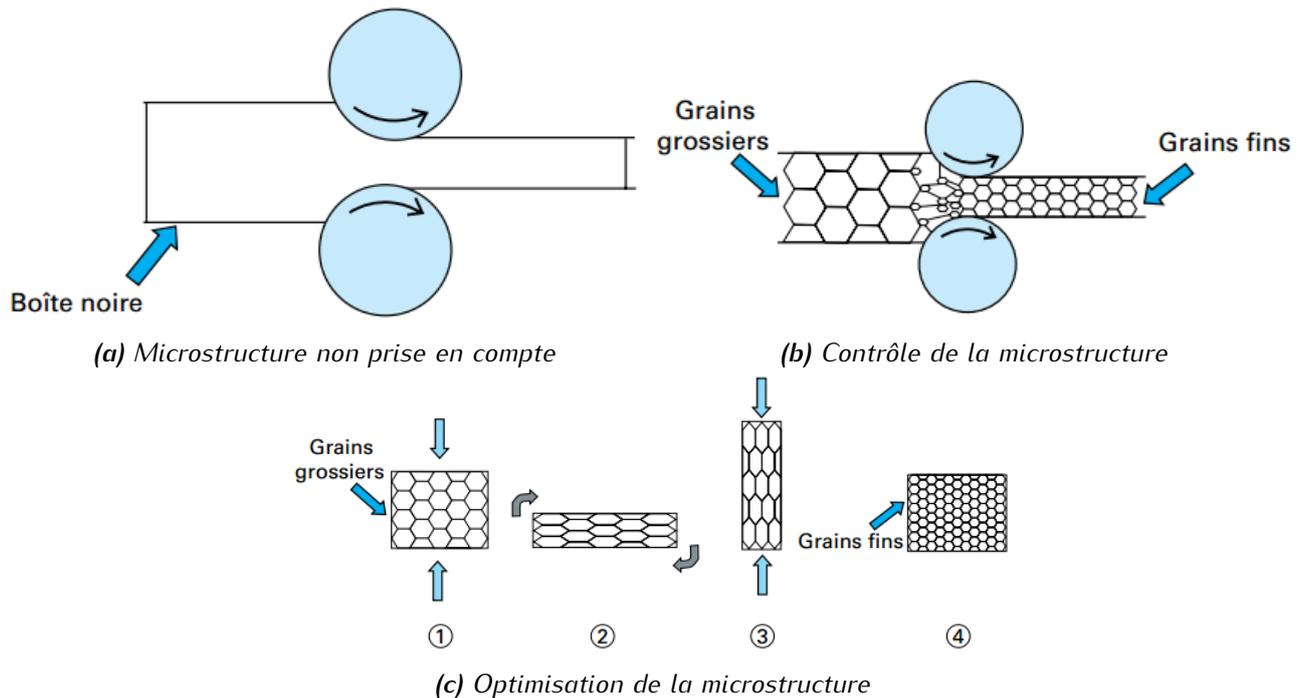


FIGURE 1.1 – Évolution des procédés de mise en forme à chaud d'après [Montheillet, 1996].

caractéristiques des limites de grains, orientation cristallographique des grains en relation avec leur position spatiale (désignée comme *texture locale*), distribution des précipités, etc.

Dans cette perspective, le but premier de la mise en forme a évolué vers l'obtention d'une microstructure optimale, et donc de propriétés optimales, comme en témoigne l'exemple du forgeage croisé tridimensionnel illustré sur la Figure 1.1 (c), où l'affinement des grains du matériau précède l'opération de mise en forme proprement dite. La Figure 1.2 présente de manière schématique la position du domaine des traitements thermomécaniques, qui connaissent actuellement une expansion significative, au sein du processus de fabrication des pièces métalliques.

1.2.2 Types de comportement

L'analyse des corrélations entre la contrainte d'écoulement et la déformation (résultant des lois de comportement) pour une vitesse de déformation $\dot{\epsilon}$ et une température T constantes, révèle la présence de deux catégories distinctes de métaux ou d'alliages métalliques :

- Dans les matériaux présentant une énergie significative de défaut d'empilement, généralement de l'ordre de quelques 10^{-1} J/m^2 [Humphreys et Hatherly, 2004] (comme l'aluminium, le fer α , le titane β et les aciers ferritiques), la courbe de contrainte-déformation montre initialement une phase d'écrouissage, suivie d'un pic de résistance relativement modéré, suivi ensuite d'un adoucissement progressif qui conduit à une plage stationnaire où la contrainte d'écoulement est maintenue constante (voir Figure 1.3). Atteindre ce stade peut parfois nécessiter une déformation très élevée [Perdrix *et al.*, 1981, Laasraoui et Jonas, 1991] et un tel comportement est associé aux mécanismes de recristallisation dynamique continue (CDRX) et de recristallisation dynamique géométrique (GDRX).
- Dans les matériaux présentant une énergie de défaut d'empilement limitée (tels que le cuivre, les laitons, le fer γ et les aciers austénitiques), la phase d'écrouissage est généralement suivie d'un pic de résistance unique, plus marqué que dans le scénario précédent. Le régime stationnaire est atteint pour des niveaux de déformation beaucoup plus faibles, après une série

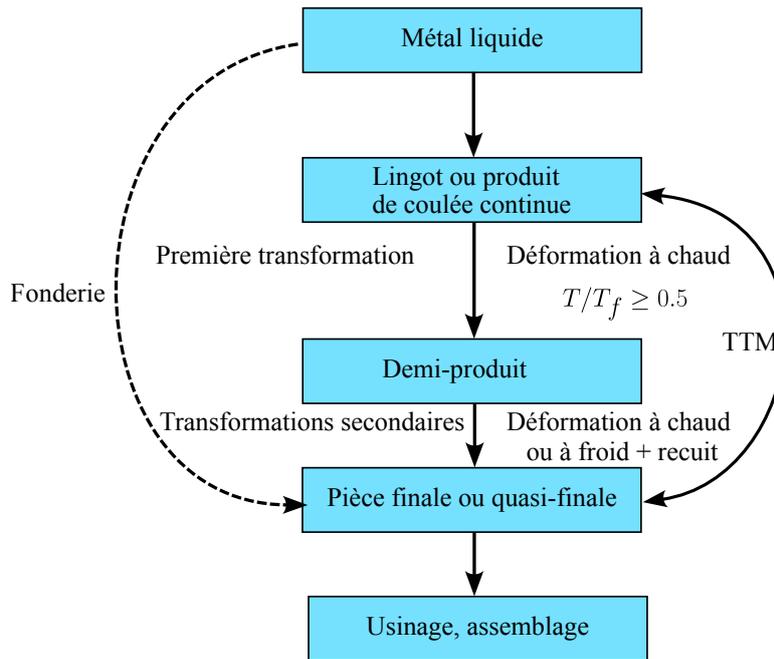


FIGURE 1.2 – Position du TTM dans la chaîne de production d'après Montheillet [Montheillet, 2009].

d'oscillations, dont l'amplitude diminue graduellement. Ce comportement est lié au processus de recristallisation dynamique discontinue (DDRX). Contrairement à la déformation à froid, la contrainte d'écoulement dépend également de la vitesse de déformation et de la température.

La recristallisation dynamique (DRX), est devenue un outil industriel d'une importance cruciale dans la création de traitements thermomécaniques car il permet de fournir à la pièce fabriquée des propriétés visées en fonction de l'application [Jonas, 1994].

Par ailleurs, il a longtemps été pensé que la recristallisation dynamique ne se produisait que dans les métaux présentant une faible énergie de défaut d'empilement, comme dans le cas de la recristallisation dynamique discontinue (DDRX). Le comportement de l'autre catégorie de matériaux était supposé être gouverné par la restauration dynamique. Cependant, cette dernière n'implique que les dislocations et ne peut pas expliquer l'apparition de nouveaux grains équiaxes lors de déformations significatives, ni même la mise en place d'un régime stationnaire. L'étude de microstructures fortement déformées, notamment grâce à des techniques telles que la diffraction des électrons rétrodiffusés (EBSD), a permis de mettre en évidence les processus de recristallisation dynamique continue (CDRX) et géométrique (GDRX).

1.3 Déformation à chaud et recristallisation

La déformation à chaud est un processus de modification des propriétés et de la forme des matériaux lorsqu'ils sont portés à des températures élevées, généralement proches ou supérieures à la moitié de leur température de fusion ($T \geq 0.5 \times T_m$). A ces températures, les atomes ou les ions qui composent ces matériaux gagnent en énergie cinétique, ce qui augmente leur mobilité et permet aux dislocations responsables de la plasticité des matériaux de se déplacer plus facilement à travers la structure cristalline.

La déformation à chaud donne aux ingénieurs et aux fabricants la possibilité de façonner les matériaux de manière plus complexe et précise qu'il ne serait possible à température ambiante et peut améliorer les propriétés mécaniques des matériaux en favorisant la recristallisation (RX) [Doherty

et al., 1997, Brown *et al.*, 1997, Brown *et al.*, 1997, Norris et Wilson, 1999, Ferguson et Lindeman, 2006], un processus au cours duquel de nouvelles structures cristallines se forment, réduisant ainsi les imperfections et les déformations accumulées. On a alors création d'une nouvelle structure dans un matériau déformé par formation et migration de joints de grains de grande désorientation qui se déplacent sous l'effet de l'énergie stockée pendant la déformation.

En fonction des conditions thermomécaniques, il existe plusieurs types de recristallisation : dont les plus connues sont la recristallisation dynamique (DRX) et la recristallisation statique (SRX) [Yoshie *et al.*, 1987, Ferguson et Lindeman, 2006, Karanjgaokar *et al.*, 2011]. La DRX agit pendant la déformation, par opposition à la SRX qui se produit lors d'un recuit, après déformation par exemple. Durant la DRX, la nucléation et la migration des joints de grains interviennent au cours de la déformation. Les courbes contrainte-déformation présentent alors : soit un caractère oscillant (Figure 1.3 (a)), soit un seul pic de contrainte (Figure 1.3 (a) et Figure 1.3 (b)) [Luton et Sellars, 1969, Lindeman, 2008].

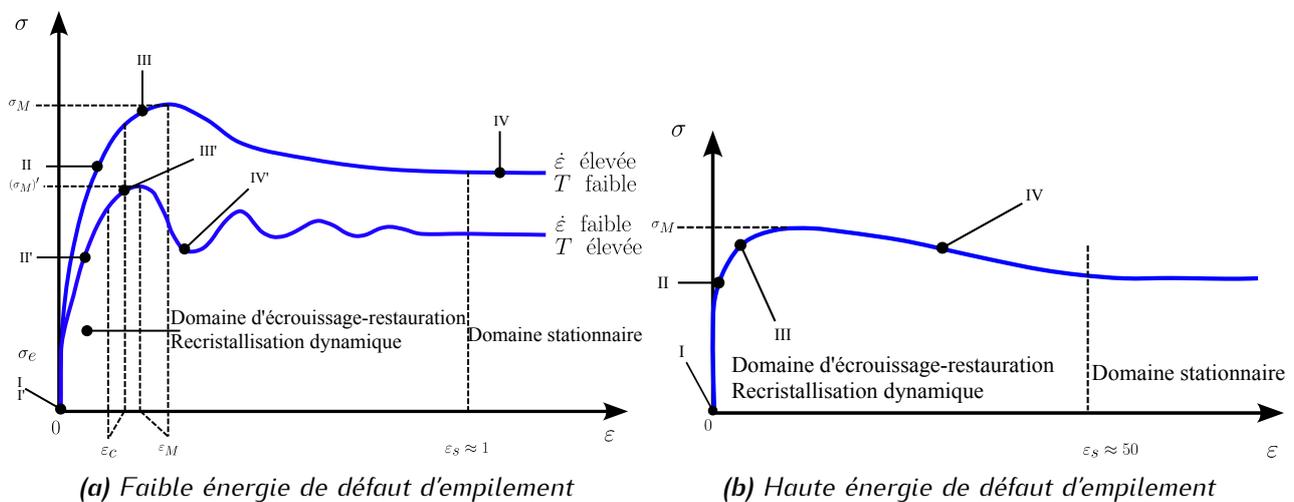


FIGURE 1.3 – Courbes contrainte-déformation caractéristiques pour deux type d'énergie de défaut d'empilement d'après Montheillet [Montheillet, 1996].

Un des paramètres très important pour la caractérisation des matériaux est l'énergie de défaut d'empilement (*SFE*, pour *Stacking Fault Energy* en anglais) qui est un état dans lequel les atomes d'un matériau cristallin ne parviennent pas à adopter leur position la plus stable dans une structure empilée pour diverses raisons, telles que la présence d'impuretés, de dislocations, ou d'autres facteurs perturbateurs. Ce défaut d'empilement se produit lorsque les atomes prennent des positions légèrement décalées ou incorrectes, ce qui peut entraîner une augmentation de l'énergie totale par rapport à la configuration idéale. Ce phénomène peut jouer un rôle important dans les propriétés et le comportement des matériaux cristallins dans diverses applications.

1.3.1 Aspects pilotant l'écroutissage d'un matériau

Au début du processus de déformation, il se produit une augmentation de la quantité de dislocations qui deviennent mobiles [Humphreys et Hatherly, 2012] et engendre une rapide élévation de la contrainte exercée, marquant ainsi le stade d'écroutissage. Cet écroutissage, aussi appelé durcissement par déformation, est un procédé métallurgique où un matériau, généralement un métal, gagne en résistance et en dureté. On crée des dislocations dans la structure cristalline du matériau, ce qui entrave la déformation ultérieure et rend le matériau moins déformable. Ceci crée des contraintes internes, rendant le matériau moins déformable.

Cet écrouissage implique de déformer mécaniquement le matériau sans l'exposer à des températures hautement élevées ce qui le rend plus fragile et moins ductile et signifie qu'il aura tendance à se casser plutôt qu'à se déformer. Il est largement utilisé dans l'industrie pour fabriquer des composants qui nécessitent une résistance accrue, tels que des pièces automobiles, des structures métalliques et des outils. Il est important de trouver un équilibre entre le durcissement souhaité et la ductilité nécessaire pour éviter la fragilisation excessive du matériau.

Sur une courbe contrainte-déformation, l'écrouissage est clairement identifiable. Au début de la déformation, la courbe présente une augmentation rapide de la contrainte avec une augmentation correspondante de la déformation qui témoigne de l'accumulation croissante de dislocations dans la structure du matériau, provoquant une résistance significative à la déformation. Cet écrouissage peut s'accompagner d'une restauration dynamique et d'une recristallisation dynamique.

La restauration dynamique est un mécanisme qui intervient pour réduire l'effet de l'écrouissage à des températures élevées, car les dislocations peuvent se déplacer plus facilement ce qui entraîne une réduction partielle de l'écrouissage initial. Ce processus permet au matériau de retrouver partiellement ses propriétés mécaniques d'origine.

La recristallisation dynamique est un phénomène où de nouveaux grains cristallins se forment à l'intérieur du matériau déformé lorsque la déformation plastique à chaud atteint un certain seuil de température et de déformation. Elle est bénéfique car elle restaure complètement les propriétés du matériau et élimine les effets de l'écrouissage. Les nouveaux grains cristallins sont exempts de dislocations et offrent au matériau une meilleure ductilité et des propriétés mécaniques améliorées. La Figure 1.4 donne un aperçu de la représentation de ces trois phénomènes sur une courbe contrainte-déformation.

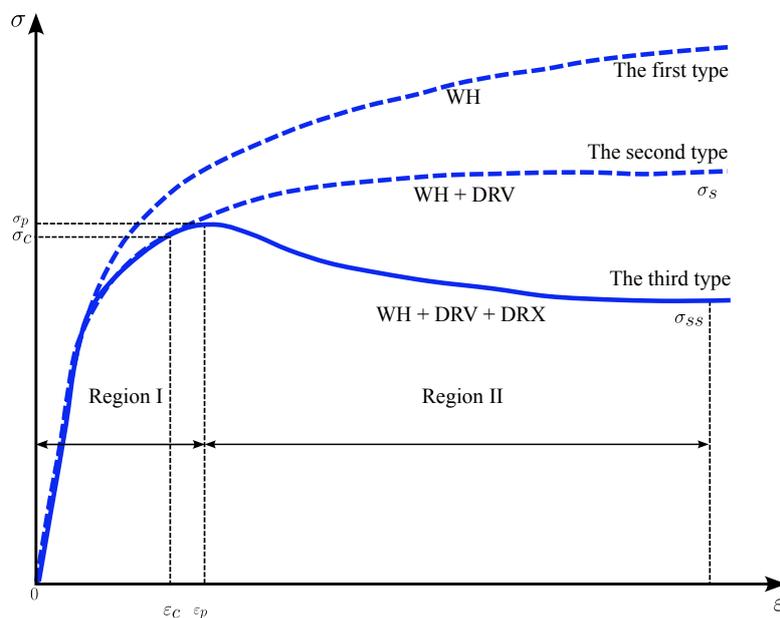


FIGURE 1.4 – Courbes contrainte-déformation de trois types de réponses d'un matériau pendant une déformation [Jonas et al., 2009].

1.3.2 Restauration dynamique (DRV)

La restauration dynamique se produit lorsqu'un matériau subit une déformation plastique et que les dislocations se déplacent à l'intérieur de sa structure cristalline [Souaï, 2011]. La déformation plastique provoque une augmentation de la densité de dislocations entraînant un durcissement du matériau [Montheillet, 2009]. Au cours du procédé, la restauration dynamique atténue cet effet de durcissement.

Au début du processus de déformation, l'apparition et la multiplication de dislocations entraînent une augmentation rapide de la contrainte d'écoulement et une fois que le nombre de dislocations atteint un seuil critique, cela déclenche le début du processus d'annihilation des dislocations et de réorganisation du matériau. Ceci se traduit par l'apparition d'un plateau sur la courbe contrainte-déformation. Ce mécanisme de réorganisation de dislocations est le principal moteur du phénomène de restauration selon [Humphreys et Hatherly, 2004].

Ce processus de restauration est illustré sur la Figure 1.5. Sur la Figure 1.5 (a), le processus de

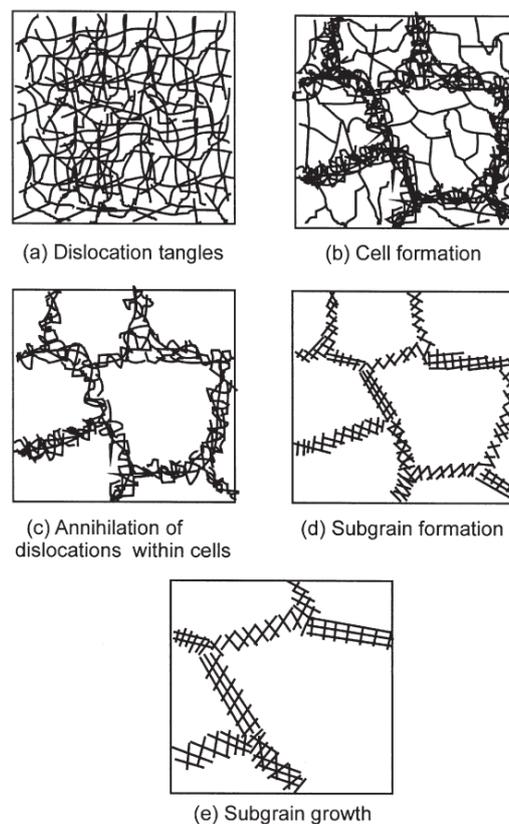


FIGURE 1.5 – Étapes de la restauration d'après Humphreys [Humphreys et Hatherly, 2004]

déformation produit des enchevêtrements de dislocations au sein de la structure qui se transforment en agencements cellulaires sur la Figure 1.5 (b). Pour les matériaux avec une énergie de défaut d'empilement élevée, cette transformation peut se produire simultanément avec la déformation. Par la suite, les Figures 1.5(c) et 1.5 (d) montrent le renforcement des limites cellulaires dans les limites de sous-grains développés, provoqué par le réarrangement des dislocations et leur transition vers des configurations à faible énergie. Comme illustré sur la Figure 1.5 (e), un recuit continu déclenche la croissance des sous-grains, entraînant une réduction globale de l'énergie interne [Humphreys et Hatherly, 2004].

Pendant la phase de restauration, la courbe contrainte-déformation suit un schéma de croissance initiale suivie de la mise en place d'un niveau constant de contrainte d'écoulement. Lorsque la den-

sité de dislocations augmente, la vitesse de restauration s'accélère, conduisant à la formation de microstructures contenant des sous-grains. A ce stade, un état stable de contrainte d'écoulement est atteint, où la densité de dislocations reste constante et l'interaction entre le durcissement par déformation et les taux de restauration atteint un équilibre dynamique [Humphreys et Hatherly, 2004]. Les modifications de microstructures résultant de la restauration dynamique sont illustrées sur la Figure 1.6.

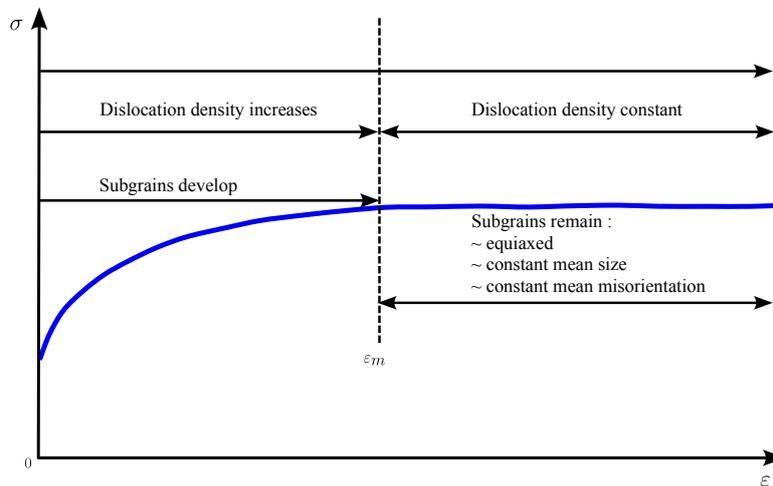


FIGURE 1.6 – Changements de microstructure pendant la restauration dynamique d'après Humphreys [Humphreys et Hatherly, 2004].

1.3.3 Recristallisation dynamique (DRX)

La recristallisation dynamique (DRX) est un processus métallurgique qui se produit lors de la déformation à chaud des matériaux, en particulier dans les métaux et les alliages. C'est un phénomène important qui permet de restaurer la microstructure d'un matériau déformé à haute température, en formant de nouveaux grains cristallins exempts de déformations et de contraintes.

Ce processus se déroule généralement dans les régions du matériau où les déformations sont plus élevées, comme les zones proches des joints de grain ou les endroits présentant des contraintes locales. Les dislocations et la déformation créent de nouveaux grains qui croissent et forment de nouveaux grains recristallisés.

La DRX commence plus aux frontières de grains à angles élevés, qui peuvent exister dans la structure ou se former pendant la déformation. Le bombement des frontières des grains est la première étape de la DRX et correspond à la migration des frontières de grains induite par la déformation (SIBM) [Ebrahimi *et al.*, 2013]. Dans de nombreux métaux, la DRX est un phénomène courant pendant le processus réel de mise en forme à chaud.

Près de la contrainte maximale, la SIBM se produit à une déformation critique et les frontières localement bombées éliminent alors les dislocations, ce qui ralentit l'écroutissage [Ebrahimi *et al.*, 2011b]. Poliak et Jonas [Poliak et Jonas, 1996] ont identifié cette déformation critique comme étant l'intersection de la dérivée première du durcissement avec la contrainte d'écoulement.

Au delà de cette valeur critique de la déformation, le taux de DRX et d'accumulation de dislocations atteint un état d'équilibre résultant en un plateau sur la courbe d'écoulement. La Figure 1.7 montre le développement de la microstructure pendant la recristallisation dynamique, démontrant que l'évolution de la microstructure dépend du rapport (D_0/D_R) des tailles de grains recristallisés

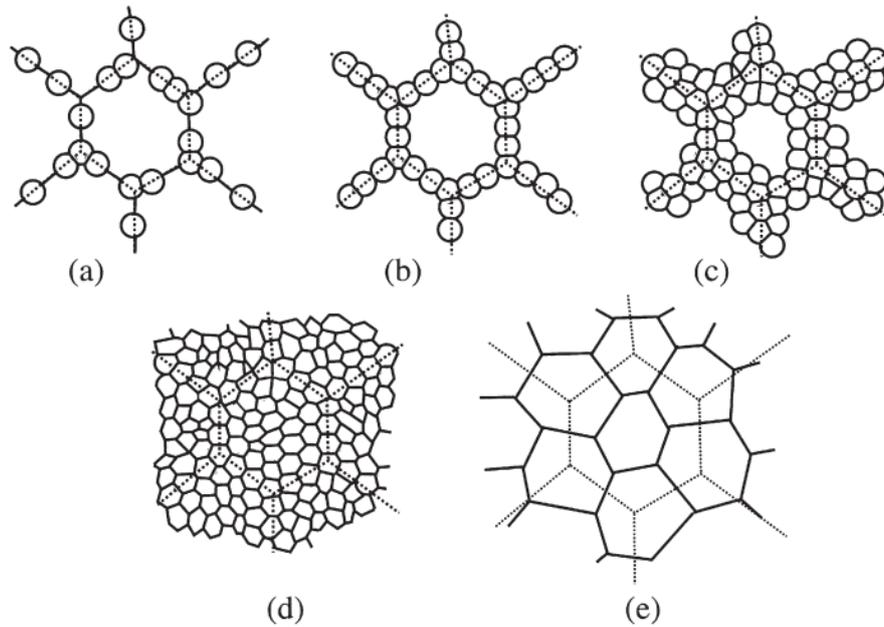


FIGURE 1.7 – Évolution de la microstructure pendant la recristallisation : (a)-(d) large grains initiaux (e) petits grains initiaux d'après Humphreys [Humphreys et Hatherly, 2004].

D_R et initiaux D_R . Si $(D_0/D_R) > 2$, alors la microstructure se développe comme indiqué sur la Figure 1.7 (a-d). Si $(D_0/D_R) < 2$, alors les nouveaux grains se développent tous environ en même temps car il y a suffisamment de sites (c'est-à-dire d'anciennes frontières) pour que la recristallisation soit complète en un cycle, comme indiqué sur la Figure 1.7 (e). Poliak and Jonas [Poliak et Jonas, 1996] ont étudié la relation entre la taille initiale des grains et la forme des courbes contrainte-déformation.

Dans le cas de l'évolution de la microstructure telle qu'illustrée sur la Figure 1.7 (a-d), les courbes présentent un pic initial suivi d'un plateau. Cependant, dans le cas de la Figure 1.7 (e), cela se traduit par une courbe de contrainte-déformation qui oscille de manière significative. L'évolution de la microstructure pendant la restauration dynamique (DRV) et la recristallisation dynamique (DRX) est schématiquement représentée sur la Figure 1.8.

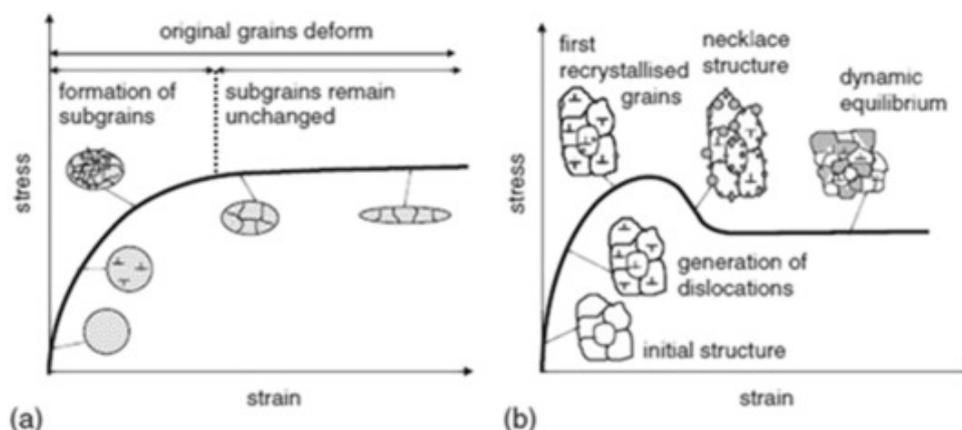


FIGURE 1.8 – Évolution de la microstructure pendant une déformation à chaud d'un matériau : (a) restauration dynamique et (b) recristallisation dynamique [Alaneme et Okotete, 2019].

La recristallisation dynamique peut être catégorisée en fonction de la température à laquelle elle se produit et des mécanismes qui la sous-tendent [Ebrahimi et al., 2018] :

- **Recristallisation dynamique à haute température** $T > 0.5 \times T_m$: A ces températures, les mouvements des dislocations sont plus actifs, ce qui favorise la restructuration des grains cristallins et la formation de nouveaux grains. Cela peut se produire lors de processus de déformation à chaud tels que le laminage, le forgeage ou l'extrusion.
- **Recristallisation dynamique à basse température** $T < 0.5 \times T_m$: Ce type de recristallisation se produit généralement après une déformation à température moyenne, telle que le laminage ou l'étirage.

Dans les deux cas, le mécanisme est essentiellement le même dans le sens où les dislocations introduites par la déformation plastique servent de noyaux pour la formation de nouveaux grains cristallins. Ces nouveaux grains ont une orientation cristallographique plus favorable, ce qui réduit les défauts et améliore les propriétés mécaniques du matériau.

Dans le cas des alliages à forte énergie de défaut d'empilement et d'autres conditions de déformation (par exemple, $T < 0.5 \times T_m$), un autre mécanisme de relaxation de contrainte peut apparaître. Ce mécanisme appelé recristallisation dynamique continue, consiste en la formation de grains issus des sous-grains de la restauration dynamique. En effet sous l'effet de la déformation, la désorientation des sous-grains, formés lors de la restauration, augmente pour former des grains à part entière [Humphreys et Hatherly, 2004]. Le Tableau 1.1, résume les caractéristiques de ces deux types de recristallisation dynamique.

Dans le cas des alliages présentant une forte densité de défaut d'empilement ($T < 0.5 \times T_m$), un mécanisme alternatif de relâchement des contraintes peut se manifester. Ce mécanisme, connu sous le nom de recristallisation dynamique continue, implique la création de grains à partir des sous-grains issus du processus de restauration dynamique. En effet, lors de la déformation, la désorientation au sein de ces sous-grains, formés lors de la restauration, augmente progressivement jusqu'à ce qu'ils se transforment en grains distincts, comme l'a décrit Humphreys en 2004 [Humphreys et Hatherly, 2004].

TABLE 1.1 – Comparaison entre les deux types de recristallisation dynamique

Recristallisation continue	Recristallisation discontinue
Les sous grains formés lors de la restauration statique se transforment en grains indépendants	Cycles rapides de durcissement → nucléation → Croissance de grains
Homogénéité des densités de dislocations Restauration dynamique forte Taux de migration de dislocation faible	Hétérogénéité de dislocations Restauration dynamique faible Taux de dislocations élevé

1.3.3.1 Recristallisation dynamique continue (CDRX)

La recristallisation dynamique continue est un processus qui se produit lors de la déformation plastique d'un matériau à hautes températures. Elle implique la formation continue et progressive de nouveaux grains cristallins dans le matériau déformé. Les nouveaux grains se forment au fur et à mesure que les anciens grains subissent une déformation plastique et que les dislocations s'accumulent. Contrairement à la recristallisation dynamique discontinue, où les premiers germes de grains recristallisés se forment à un point spécifique de la courbe contrainte-déformation, la CDRX se produit de manière graduelle tout au long de la déformation. La recristallisation dynamique continue a lieu principalement dans les métaux à faible énergie de défaut d'empilement. Ce processus de CDRX

contribue à améliorer les propriétés mécaniques et la ductilité du matériau en réduisant les effets de l'écroutissage.

A des niveaux de déformation très élevés, le processus de recristallisation dynamique continue conduit à la formation d'une microstructure qui est illustrée sur la Figure 1.9. Sur cette microstructure, les limites entre les grains sont représentées par des lignes noires, tandis que les limites entre les sous-grains sont marquées par des lignes grises. On considère qu'une interface entre deux cristaux est un sous-joint lorsque la désorientation entre eux est inférieure à 15 degrés ($\theta < 15^\circ$), et elle est considérée comme un joint lorsque la désorientation dépasse 15 degrés ($\theta > 15^\circ$) [Montheillet et Desrayaud, 2009].

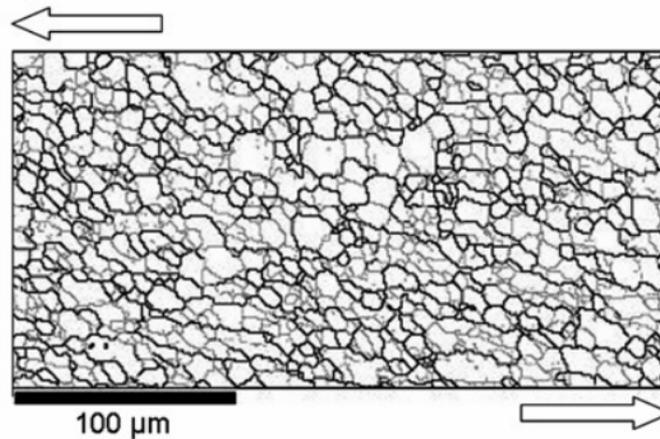


FIGURE 1.9 – Microstructure de la CDRX d'après Montheillet [Montheillet et Desrayaud, 2009].

1.3.3.2 Recristallisation dynamique discontinue (DDRX)

La recristallisation dynamique discontinue est un phénomène qui se produit lors de la déformation plastique d'un matériau à des températures élevées. Contrairement à la recristallisation dynamique continue qui se produit de manière graduelle tout au long de la déformation, la DDRX se caractérise par une phase distincte au cours de laquelle de nouveaux grains cristallins se forment de manière soudaine et rapide dans le matériau déformé. Pendant la DDRX, des germes de grains recristallisés émergent à un point spécifique de la courbe contrainte-déformation, généralement près du pic de la courbe. Ces germes se développent rapidement en nouveaux grains exempts de déformations et de défauts. La DDRX est généralement observée dans les matériaux à faible énergie de défaut d'empilement. La recristallisation dynamique discontinue est importante car elle permet de régénérer la microstructure du matériau, améliorant ainsi ses propriétés mécaniques, sa ductilité et sa résistance. Elle contribue également à réduire les effets négatifs de l'écroutissage qui peut résulter de la déformation plastique continue.

Dans les matériaux présentant une faible énergie de défaut d'empilement, les dislocations subissent une dissociation significative, ce qui entrave leur capacité à se déplacer et complique leur réarrangement en sous-joints, réduisant ainsi l'efficacité du processus de restauration dynamique. Par conséquent, l'écroutissage joue un rôle prépondérant, conduisant à l'accumulation locale de fortes densités de dislocations au sein du matériau.

Cette accumulation d'énergie crée une force suffisante pour déclencher la formation de nouveaux grains, surtout à des températures élevées. Lorsque la déformation atteint une valeur critique, légèrement en deçà de la déformation correspondant au pic de la courbe contrainte-déformation, des germes de nouveaux grains commencent à émerger au sein de la matrice déjà renforcée, comme illustré sur la Figure 1.10.

Un germe désigne une région de petite taille, typiquement de l'ordre du micromètre, qui présente une faible densité de dislocations et qui est séparée de la matrice environnante par des interfaces caractérisées par une forte désorientation. Plusieurs mécanismes semblent être en mesure de créer ces germes, notamment la rotation rapide d'un sous-grain particulier, la fusion de sous-grains adjacents, ou encore le déplacement local d'une limite de grain à l'intérieur d'un grain voisin. Ces germes ont tendance à se former à proximité des joints de grains initiaux. La migration des joints de grains induit ainsi une réduction de l'énergie du système. La contrainte d'écoulement atteint alors un pic et la structure déformée initiale est rapidement remplacée par une structure recristallisée. Ce processus empêche le développement des mécanismes plus lents de recristallisation dynamique continue [Sakai et Jonas, 1984].

La continuation du processus est conditionnée par la relation entre la taille initiale des grains D_0 du matériau et la taille moyenne des grains D_s atteinte lors de l'état stationnaire. Cette taille moyenne dépend uniquement de la température et de la vitesse de déformation imposée pour un matériau donné [Sakai et Jonas, 1984]. On peut donc distinguer deux scénarios distincts :

- Si la recristallisation conduit à un affinement des grains (lorsque $D_0 > 2D_s$), le processus de recristallisation dynamique se produit de manière continue le long des joints des grains initiaux déformés, comme illustré sur les Figures 1.10 (a) et 1.11. Les nouveaux grains recristallisés progressent graduellement depuis les bords extérieurs des anciens grains, ce qui se traduit par une diminution de la contrainte d'écoulement du matériau. Cependant, les grains recristallisés subissent des déformations en réponse à la sollicitation appliquée. Au fur et à mesure que les grains initiaux sont entièrement remplacés, la structure d'écrouissage du matériau devient fortement hétérogène. Par conséquent, la recristallisation dynamique se déroule de manière asynchrone, avec certaines régions du matériau en phase de germination de nouveaux grains, tandis que d'autres connaissent une phase de croissance simultanée à l'écrouissage des grains existants. Cela se traduit par l'apparition d'une contrainte d'écoulement stationnaire après l'atteinte d'un unique pic dans la courbe contrainte-déformation.
- Si le processus conduit à un grossissement des grains (lorsque $D_0 < 2D_s$), la croissance initiale des nouveaux grains recristallisés le long des joints des grains initiaux déformés est rapidement limitée par leur interaction mutuelle (comme illustré sur la Figure 1.10 (b)). En conséquence, la première vague de recristallisation, qui initie une décroissance initiale de la courbe contrainte-déformation, ne permet pas aux grains recristallisés d'atteindre leur taille stationnaire D_s . L'ensemble du matériau subit alors une nouvelle phase d'écrouissage relativement homogène, jusqu'à ce qu'une deuxième vague de recristallisation se déclenche, accompagnée d'une nouvelle décroissance de la courbe contrainte-déformation et ce cycle se répète par la suite. La recristallisation dynamique se produit donc de manière synchronisée, ce qui engendre des oscillations successives sur la courbe de contrainte d'écoulement. Au fur et à mesure que le processus se poursuit, le développement progressif d'hétérogénéités lors de l'écrouissage du matériau entraîne l'amortissement des oscillations, conduisant à l'établissement d'une contrainte d'écoulement stationnaire.

1.3.4 Restauration statique et recristallisation statique

Après la phase de déformation à chaud d'un matériau, d'autres mécanismes peuvent être activés lors du refroidissement et des traitements thermiques. Ces mécanismes se manifestent en l'absence de déformation et ne sont pas soumis à une vitesse de déformation imposée. L'énergie emmagasinée dans le matériau pendant la déformation à une température élevée entraîne l'amorçage de trois mécanismes microstructuraux à savoir la restauration statique, la recristallisation statique et la recristallisation métadynamique [Montheillet et Desrayaud, 2009]. La restauration statique ainsi que

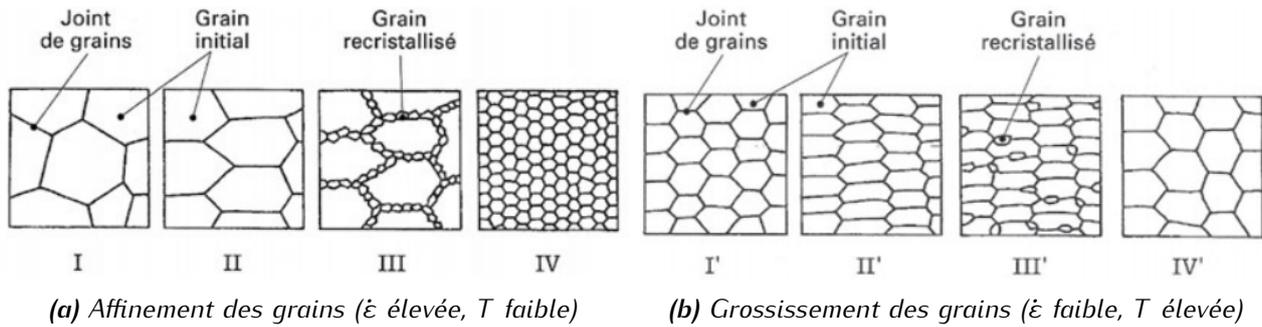


FIGURE 1.10 – Représentation schématique de la microstructure au cours d'une DDRX d'après Montheillet [Montheillet et Desrayaud, 2009].

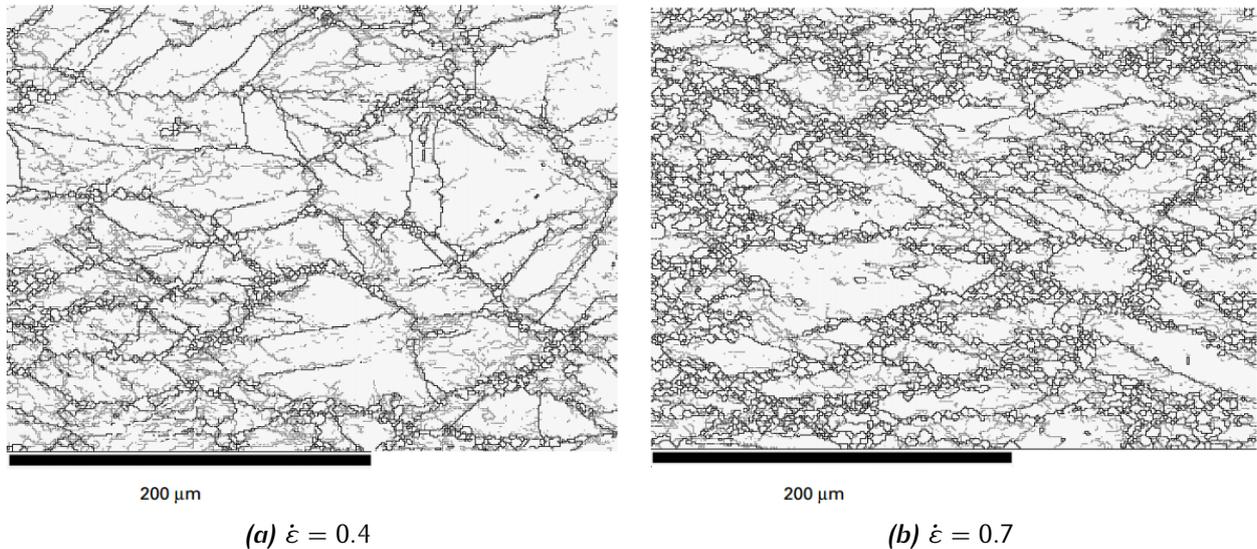


FIGURE 1.11 – DDRX en collier dans un superalliage à base de nickel (In-718) déformée à chaud d'après Montheillet [Montheillet et Desrayaud, 2009].

la recristallisation statique sont gouvernées par les mêmes mécanismes microstructuraux que lorsqu'il y a présence de déformation [Montheillet et Desrayaud, 2009]. Sur la Figure 1.5, différentes phases de la restauration statique sont mises en évidence, telles que l'enchevêtrement des dislocations (a), la formation de cellules (b), l'annihilation de certaines dislocations (c), la création de sous-grains (d) et l'élargissement de ces sous-grains (e). Cependant, étant donné que la recristallisation dynamique a été enclenchée, la restauration statique s'avère très limitée [Humphreys et Hatherly, 2004]. Les différentes étapes de la recristallisation statique sont similaires à celles décrites sur la Figure 1.7. Cependant, en raison de l'absence de déformation durant ce processus, les grains recristallisés qui se forment sont caractérisés par une faible densité de dislocations, contrairement à ceux qui se forment lors de la recristallisation dynamique.

1.3.5 Recristallisation métadynamique

Une fois que le seuil d'amorçage de la recristallisation dynamique a été franchi, des amorfes se forment dans le matériau. Si la déformation cesse mais que le matériau demeure à une température élevée (lors d'un recuit par exemple), ces amorfes vont se développer davantage en tirant parti de l'effet de la déformation antérieure ainsi que de la température [Humphreys et Hatherly, 2004]. Ce processus est connu sous le nom de recristallisation métadynamique. Il entraîne la formation d'une microstructure inhomogène à l'intérieur du matériau, constituée de petites amorfes ayant peu de dislocations internes, de grands grains recristallisés avec une densité de dislocations moyenne, ainsi que de grains non recristallisés présentant une densité de dislocations élevée [Humphreys et Hatherly, 2004]. En raison de cette hétérogénéité microstructurale, la modélisation du comportement du matériau devient complexe. Il est crucial d'étudier et de maîtriser ces mécanismes d'adoucissement, à savoir la restauration statique, la recristallisation statique et la recristallisation métadynamique. Ces mécanismes revêtent une importance similaire à ceux déclenchés pendant la déformation, car ils engendrent des modifications significatives dans la microstructure et les propriétés finales du matériau [Montheillet et Desrayaud, 2009].

Après une exposition approfondie des mécanismes sous-jacents à la déformation à chaud, il est maintenant important de mettre ces connaissances en pratique en se penchant sur l'étude d'un cas concret, qui constituera le sujet central de la section suivante : une campagne d'essais de compression sur Gleeble.

1.4 Campagne expérimentale sur Gleeble

1.4.1 Présentation du dispositif Gleeble

Le dispositif expérimental utilisé dans le cadre de cette étude est le simulateur thermomécanique Gleeble-3800 disponible au sein de l'équipe CM2SP de l'École de Technologie Supérieure (ÉTS) de Montréal au Canada, présenté sur la Figure 1.12.



FIGURE 1.12 – Simulateur thermomécanique Gleeble-3800 – ÉTS CM2SP, Montréal.

Il peut effectuer une variété d'essais mécaniques, tant dynamiques que statiques pour des conditions thermomécaniques variées et à haute température. Un essai sur Gleeble se divise en trois phases

principales : le chauffage, le maintien et la déformation. On amène l'échantillon à une température prédéterminée, on le maintient à cette température pendant une durée spécifiée, puis on le soumet à une déformation à une vitesse constante.

Ce système peut intégrer deux méthodes de chauffage : le chauffage résistif (par effet Joule) et le chauffage par induction. Le système de chauffage résistif peut chauffer les échantillons à des vitesses allant jusqu'à 20 000 °C/s. Des mors à conductivité thermique élevée maintiennent l'échantillon, ce qui permet d'atteindre des vitesses de refroidissement élevées. Un système de trempe peut permettre de dépasser 20 000 °C/s à la surface de l'échantillon. Plusieurs thermocouples peuvent être utilisés pour mesurer la température au cours de l'essai et le choix du thermocouple est guidé par la nature du matériau.

Le Gleeble-3800 intègre un système mécanique alimenté par un système hydraulique, capable d'appliquer une force pouvant atteindre 20 t en traction ou en compression. Il permet d'atteindre des vitesses de déplacement allant jusqu'à 1000 mm/s. Ce système est équipé de transducteurs LVDT, de cellules de pesée et d'extensomètres laser sans contact, qui fournissent des retours d'information essentiels pour assurer une exécution précise et reproductible des programmes de tests mécaniques.

1.4.2 Matériau d'étude (3Cr2Mo)

Le matériau utilisé dans cette étude est un acier moyennement carboné, de type P20 à bas carbone (dénomination Canadienne car les essais ont été réalisés à Montréal). Selon la norme Européenne, cet acier correspond au 3Cr2Mo, c'est la dénomination qui sera retenue dans toute la suite du document. Cet acier est utilisé pour la fabrication de moules, sa composition chimique est donnée dans le Tableau 1.2.

TABLE 1.2 – Composition chimique du 3Cr2Mo, Fe = équilibre.

Element	C	Mn	Mo	Si	Ni	Cr	Cu
Wt %	0.30	0.89	0.52	0.34	0.68	1.86	0.17

Des échantillons cylindriques ont été usinés avec un diamètre initial $d = 10$ mm et une hauteur $h_0 = 15$ mm.

Un ensemble de 30 essais de compression à haute température a été effectué sur le simulateur Gleeble-3800 (voir Figure 1.12) avec 5 températures (1050 °C, 1100 °C, 1150 °C, 1200 °C et 1250 °C) et 6 vitesses de déformation (10^{-3} s^{-1} , 10^{-2} s^{-1} , 10^{-1} s^{-1} , 1 s^{-1} , 2 s^{-1} et 5 s^{-1}). Il faut noter que le choix des plages de températures et des vitesses de déformation sont liées aux conditions de forgeage à hautes températures imposées par l'entreprise Finkl-Steel (Montréal, Canada) qui a fourni les échantillons. Des feuilles de tantale ont été utilisées comme matériau lubrifiant entre les enclumes et l'échantillon, comme montré sur la Figure 1.13. Les échantillons ont été chauffés à une température de $T = 1260$ °C avec une vitesse de chauffage de 2 °C/s, et ils ont été maintenus à cette température pendant une durée de 5 minutes afin d'éliminer les gradients de température. Ensuite, les échantillons ont été refroidis à une vitesse de refroidissement de 1 °C/s jusqu'à atteindre la température d'essai. Ils ont ensuite été maintenus à cette température constante pendant une période de 1 minute avant toute déformation dont la durée dépend de la vitesse de déformation.

Après la déformation, l'échantillon est soumis à un refroidissement rapide pour figer sa microstructure en prévision d'une analyse ultérieure. La Figure 1.13 montre également la géométrie des échantillons avant et après la compression : h_0 et r_0 représentent respectivement la hauteur et le rayon de

l'échantillon avant la compression, tandis que h , r_m et r_t désignent respectivement la hauteur, le grand rayon et le petit rayon de l'échantillon après la compression.

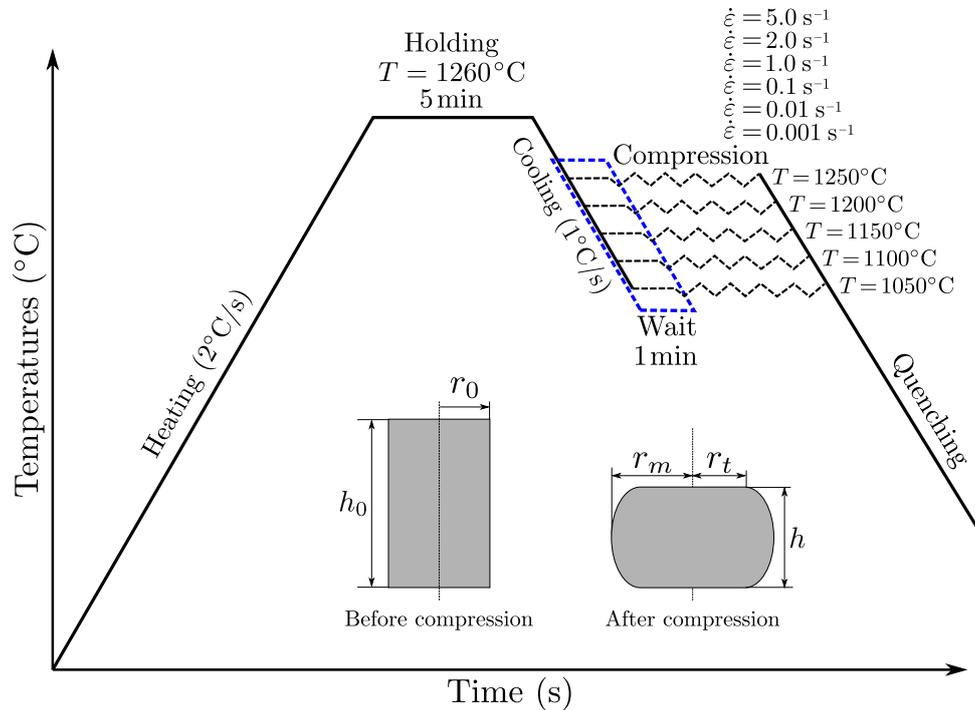


FIGURE 1.13 – Cycles des essais effectués sur Gleeble-3800.

Les courbes contrainte-déformation sont automatiquement extraites du dispositif Gleeble sous forme de contrainte et de déformation réelles, en utilisant l'extensomètre longitudinal. L'extensomètre longitudinal L-gauge permet de calculer la contrainte (σ) avec la formule $\sigma = F/A$, où F représente la force mesurée par la cellule de charge du Gleeble, et la déformation (ε) avec la formule $\varepsilon = \ln(h_0 + \Delta h/h_0)$, où h_0 est la hauteur initiale de l'échantillon et Δh est le changement de hauteur. Le simulateur Gleeble intègre également l'extensomètre radial C-gauge qui n'a pas été utilisé dans le cas de notre étude. Il faut noter que les valeurs des contraintes issues de Gleeble sont des contraintes et déformations vraies. Par conséquent, pour une implémentation dans un code de calcul élément fini dans la suite du travail, la déformation élastique est retirée et pour une question de simplification on a adopté tout simplement la notation de ε pour déformation plastique.

Étant donné que les données brutes contiennent des bruits, la méthode de lissage *savgol_filter* de la bibliothèque Python *scipy* a été utilisée [Luo *et al.*, 2005]. Cela nous permet d'éliminer les variations bruitées des données et d'obtenir des données plus régulières et moins sujettes aux fluctuations. Pour les identifications des lois de comportement qui sont présentées dans la suite du travail, la soustraction de la partie élastique de la déformation a été effectuée.

1.4.3 Résultats des essais de compression sur le 3Cr2Mo

L'ensemble des courbes de contrainte d'écoulement σ en fonction de la déformation ε obtenues à partir des essais de compression réalisés sur le simulateur Gleeble-3800 pour chaque condition d'essai (6 vitesses de déformation et 5 températures) est présenté sur la Figure 1.14. Toutes les courbes de données contiennent 700 valeurs de déformation équidistantes jusqu'à $\varepsilon = 0.7$. La base de données comprend donc 21 000 points de mesure. Le comportement général de ces courbes montre que la contrainte d'écoulement σ augmente avec la vitesse de déformation $\dot{\varepsilon}$, mais diminue

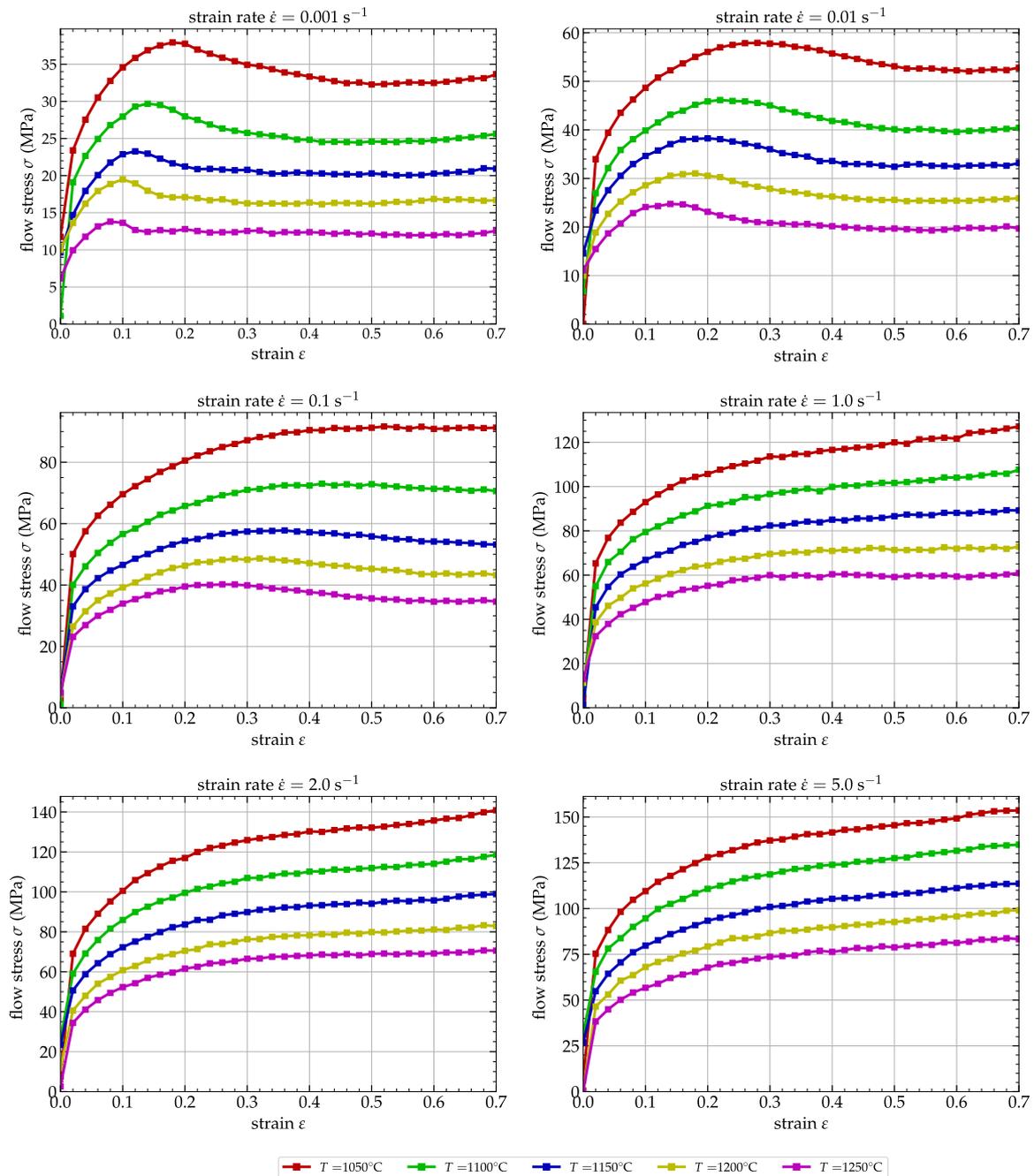


FIGURE 1.14 – Courbes contrainte-déformation du 3Cr2Mo extraites du simulateur Gleeble-3800 pour 6 températures (T) et 5 vitesses de déformation ($\dot{\epsilon}$).

avec l'augmentation de la température T . Il convient de noter que la déformation influence également la contrainte d'écoulement. En effet, pour les vitesses de déformation $\dot{\epsilon}$ les plus faibles, la contrainte d'écoulement σ augmente avec la déformation ϵ jusqu'à une valeur d'environ $\epsilon = 0.2$ à $\epsilon = 0.3$, puis diminue pour maintenir une valeur plus ou moins constante jusqu'à la fin de l'essai.

Pour les vitesses de déformation les plus élevées (1 s^{-1} , 2 s^{-1} , et 5 s^{-1}), la contrainte d'écoulement augmente tout au long de l'essai. La légère augmentation de la contrainte à des vitesses de déformation faibles, lorsque la déformation est importante, serait due au frottement entre l'échantillon et l'enclume pendant l'essai [Galos *et al.*, 2022]. L'effet de frottement est également visible lors des essais à faible vitesse de déformation, car l'effet de la lubrification diminue avec le temps et le frotte-

ment augmente donc. L'augmentation de la contrainte observée au début de la déformation et jusqu'à $\varepsilon = 0.1$ est due à l'écroutissage (WH). Après $\varepsilon = 0.1$ et jusqu'à $\varepsilon = 0.2$, la contrainte d'écoulement montre une réduction continue avec l'augmentation de la déformation jusqu'à un pic ou une inflexion du taux d'écroutissage. Ce comportement indique que l'adoucissement thermique devient de plus en plus prédominant jusqu'à ce qu'il dépasse l'écroutissage.

A ce stade, la courbe de contrainte/déformation présente trois profils différents avec l'augmentation de la déformation : (i) une diminution progressive jusqu'à un état stable avec un adoucissement décrivant le phénomène de restauration/recristallisation dynamique (DRV/DRX). C'est le cas pour toutes les températures, déformations et toutes les vitesses de déformation entre $\dot{\varepsilon} = 0.001 \text{ s}^{-1}$ et $\dot{\varepsilon} = 0.1 \text{ s}^{-1}$, à l'exception des températures $T = 1050^\circ\text{C}$ et $T = 1100^\circ\text{C}$; (ii) des niveaux de contrainte plus élevés sans adoucissement ni écroutissage significatifs à $T = 1050^\circ\text{C}$ et $T = 1100^\circ\text{C}$ et une vitesse de déformation de $\dot{\varepsilon} = 0.1 \text{ s}^{-1}$; et (iii) une augmentation continue avec un écroutissage significatif (toutes les températures et la vitesse de déformation de $\dot{\varepsilon} = 1 \text{ s}^{-1}$). On peut donc conclure que l'adoucissement dû à la DRX, caractérisé par une courbe d'écoulement avec un seul pic suivi d'un écoulement stable, a lieu à des températures élevées et à des vitesses de déformation faibles. En revanche, à des vitesses de déformation plus élevées et à des températures plus basses, le taux d'écroutissage plus élevé ralentit le taux d'adoucissement dû au DRX et, par conséquent, le pic de contrainte et le début de l'écoulement en régime permanent sont déplacés vers des niveaux de déformation plus élevés. En fait, la chute observée sur la contrainte est due à l'apparition de DRX à toutes les températures et à des vitesses de déformation de $\dot{\varepsilon} = 0.001 \text{ s}^{-1}$ à $\dot{\varepsilon} = 5 \text{ s}^{-1}$. A titre d'exemple, la Figure 1.15 montre les microstructures obtenues après un temps de maintien de 5 min à $T = 1260^\circ\text{C}$ avant et après la compression.

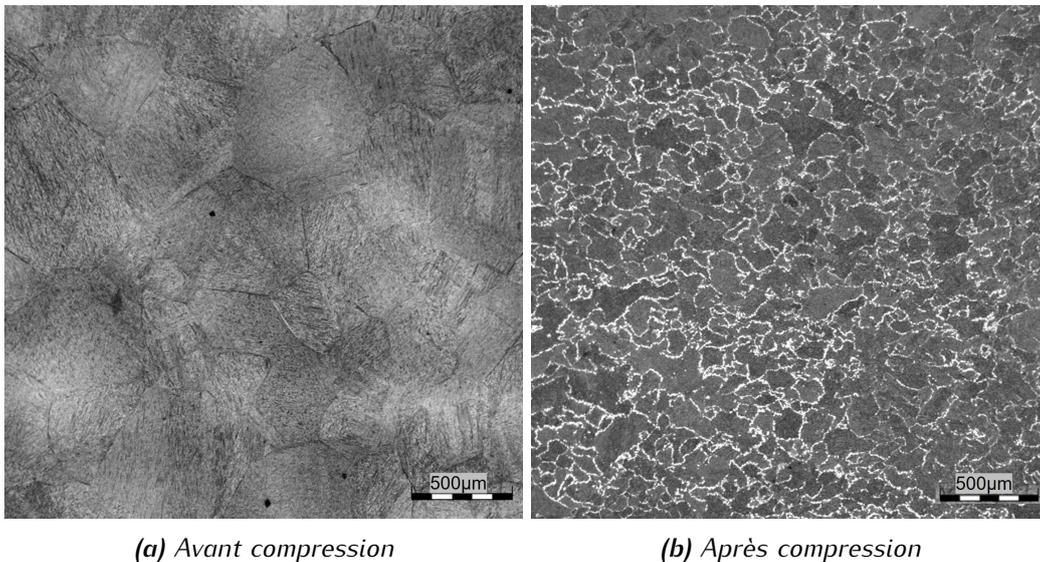


FIGURE 1.15 – Micrographies de l'acier 3Cr2Mo.

Comme on remarque bien sur les courbes d'écoulement rapportées sur la Figure 1.14, on peut voir en comparant les deux micrographies de la Figure 1.15 que la recristallisation dynamique est presque complète pour cette température et cette vitesse de déformation dans l'échantillon maintenu pendant 5 minutes à $T = 1260^\circ\text{C}$ et rapidement refroidi pour obtenir un grain austénitisé (car la température reste au delà de 800°C avant déformation) d'une part, et d'autre part, maintenu pendant 1 minute à $T = 1150^\circ\text{C}$ avant la compression avec une vitesse de déformation de $\dot{\varepsilon} = 0.001 \text{ s}^{-1}$. Les données présentées sur la Figure 1.14 seront utilisées dans la suite pour déterminer les paramètres des lois d'écoulement choisies pour le 3Cr2Mo et pour déterminer le modèle qui correspond le mieux aux résultats expérimentaux.

1.5 Conclusions du chapitre

Ce chapitre a couvert d'une part et de façon simplifiée le processus de déformation à chaud d'un matériau en commençant par la présentation des procédés de mise en forme des métaux suivi des mécanismes activés lors de la déformation à chaud. Il s'agit principalement de la restauration et de la recristallisation dynamique entre autres. La mise en place de ces phénomènes lors de la déformation dépend à la fois des conditions expérimentales mais également de la composition chimique du matériau (type de matériau). La recristallisation est devenue un outil important pour la caractérisation des matériaux dans le sens où elle permet le contrôle de la microstructure qui influence sur les propriétés mécaniques du produit final.

La deuxième partie de ce chapitre a concerné les essais effectués sur le dispositif Gleeble-3800 en utilisant le matériau le 3Cr2Mo. D'une part les données issues de cette campagne expérimentale, traduites sous forme de courbes contrainte-déformation, seront utilisées pour l'identification des lois de comportement dans les chapitres suivants et d'autre part, quelques analyses microscopiques ont été effectuées pour construire une base des données nécessaires au développement du modèle lié à la recristallisation dynamique.

Chapitre 2

Modélisation du comportement non-linéaire des matériaux

2.1	Forme générale d'une loi de comportement	28
2.2	Modèles de comportement	29
2.2.1	Modèles quasi-statiques	29
2.2.2	Modèles découplés	31
2.2.3	Modèles couplés	33
2.2.4	Modèles semi-empiriques	37
2.3	Modèle phénoménologique (PTM)	42
2.3.1	Formulation du modèle PTM	43
2.3.2	Procédure d'identification	43
2.4	Identification du comportement du 3Cr2Mo	44
2.4.1	Identification du modèle de Johnson–Cook	44
2.4.2	Identification du modèle modifié de Zerilli–Armstrong	47
2.4.3	Identification du modèle de Hensel–Spittel	49
2.4.4	Identification du modèle d'Arrhenius	49
2.4.5	Identification du modèle PTM	54
2.5	Comparaison des modèles identifiés	55
2.6	Conclusions du chapitre	57

La formulation et l'identification des lois de comportement, aussi appelée caractérisation des matériaux, est essentielle en science des matériaux et ingénierie. Elle révèle les relations entre contraintes appliquées et déformations d'un matériau sous diverses charges. Le comportement mécanique, influencé par des facteurs tels que la température, la déformation et le taux de déformation, varie en fonction du matériau et par conséquent la loi qui décrit ce comportement en dépend. Généralement pour caractériser un matériau, on le sollicite mécaniquement, on mesure les déformations puis on analyse les données pour identifier les lois de comportement adéquates. Loi linéaire, non linéaire, élastique, plastique, viscoélastique, viscoplastique, etc. sont des appellations courantes selon le matériau et l'identification des paramètres de ces lois s'effectue par diverses méthodes, allant des essais simples (traction/compression) à de l'optimisation mathématique complexe avec essais multi-axiaux thermomécaniques à vitesse variable. Ces lois de comportement alimentent les simulations numériques largement utilisées de nos jours pour la modélisation de procédés industriels dans divers secteurs (aérospatiale, automobile, construction). Par conséquent la précision de telles simulations dépend fortement d'une bonne identification de ces lois de comportement.

Dans la première partie de ce chapitre, plusieurs lois de comportement largement utilisées sont exposées et définies. Après une brève introduction de la forme générale des lois de comportement analytiques, diverses lois de comportement sont présentées incluant des modèles empiriques, semi-empiriques et physiques. Ensuite, une sélection de certaines lois sera effectuée pour leur utilisation dans le cadre de cette étude, accompagnée des techniques d'identification de leurs paramètres. Bien que l'utilisation de ces lois analytiques soit répandue depuis longtemps dans la simulation numériques de divers processus, se pose la question de la flexibilité de ces lois de comportement. Pour répondre à cette problématique, nous définirons une nouvelle loi d'écoulement, le modèle PTM, dérivant d'une forme de Zerilli–Armstrong modifiée. Finalement, ce chapitre se terminera par l'identification des paramètres d'une sélection de lois d'écoulement en fonction des données des essais réalisés au paragraphe 1.4.3, page 22.

2.1 Forme générale d'une loi de comportement

Une loi de comportement est une relation mathématique entre la réponse du matériau et les paramètres du chargement qui lui est appliqué. Elle est utilisée en mécanique des matériaux pour modéliser le comportement élastique, plastique ou viscoélastique d'un matériau soumis à des sollicitations mécaniques. Une loi de comportement permet de modéliser le comportement mécanique du matériau dans une gamme de conditions spécifiques et l'hypothèse de comportement *isotrope* est très souvent utilisée dans la formulation d'une telle loi pour simplifier l'analyse des matériaux si elle est licite pour ce matériau (cas qui nous concerne pour cette étude sur les matériaux métalliques). Il est important de noter que l'hypothèse d'isotropie peut ne pas être valide pour certains matériaux qui présentent une anisotropie intrinsèque ou une structure directionnelle significative, tels que les matériaux composites, les matériaux fibreux ou les matériaux monocristallins. Dans ce cas, des lois de comportement anisotropes spécifiques doivent être utilisées pour tenir compte des caractéristiques directionnelles du matériau. Dans la suite, le matériau sera supposé isotrope dans la formulation de la loi de comportement par conséquent on utilisera les contraintes et vitesses de déformation équivalentes au sens de von Mises [Park et Cho, 2021].

Plusieurs paramètres aussi bien à l'échelle macroscopique (déformation ε , vitesse de déformation $\dot{\varepsilon}$ et température T) qu'à l'échelle microscopique (densité de dislocation, taille de grains, etc.) influencent la déformation d'un matériau et la prise en compte de ces paramètres dans la formulation de la loi de comportement permet une bonne caractérisation d'un tel matériau. Or, l'évolution de la microstructure d'un matériau au cours de la déformation dépend aussi des grandeurs macroscopiques; ce qui rend la loi de comportement plus complexe quant à la prise en compte de l'interdépendance des variables physiques. La forme générale d'une loi de comportement d'un matériau est généralement formulée par l'équation suivante :

$$\sigma = \sigma(\varepsilon, \dot{\varepsilon}, T, \alpha_i) \quad \text{et} \quad \alpha_i = \alpha_i(\varepsilon, \dot{\varepsilon}, T), \quad (2.1)$$

où α_i est un ensemble de variables microscopiques. Selon la prise en compte des paramètres α_i dans la formulation d'une loi de comportement on peut distinguer trois types de modèles, à savoir :

- Les modèles phénoménologiques ou empiriques qui sont généralement formulées de manière simple, sans l'utilisation de variables internes, ce qui permet d'identifier leurs coefficients par de simples essais. Leur forme mathématique est choisie de manière à faciliter leur mise en œuvre dans les logiciels de calcul numérique, ce qui permet des calculs avec peu de ressources numériques et expérimentales. Ces deux avantages expliquent pourquoi les modèles empiriques sont largement utilisés.
- Les modèles physiques, plus avancés proposent des solutions souvent plus complexes à utiliser, car ils sont basés sur la physique de la déformation. Ces modèles intègrent des variables

internes qui représentent l'histoire de la déformation et/ou les caractéristiques physiques du matériau (comme la taille des grains, la densité de dislocations, etc.) comme introduit au chapitre 1.

- Les modèles semi-empiriques ont une formulation simple tout en intégrant un minimum de caractéristiques physiques déduites de la physique de la déformation.

2.2 Modèles de comportement

Lorsqu'un matériau est soumis à une sollicitation et en fonction de sa nature, de sa composition et des conditions de chargement, deux phénomènes majeurs sont observés : un comportement élastique et un comportement anélastique. Par conséquent, il est important qu'une loi de comportement intègre ces deux phénomènes dans sa formulation. Ces comportements élastique et anélastique d'un matériau sont pris en compte dans une loi de comportement en utilisant des modèles mathématiques qui décrivent comment le matériau réagit en fonction des charges appliquées et des déformations subies. Dans le cas du comportement élastique, la loi se base généralement sur la loi de Hooke généralisée, qui énonce que les contraintes sont proportionnelles aux déformations dans la plage élastique linéaire ($\sigma = E\varepsilon$), avec E le module d'élasticité (ou module de Young) du matériau.

Le comportement anélastique est plus complexe à représenter de manière empirique, car il peut dépendre de facteurs tels que le temps t , la température T , la vitesse de déformation $\dot{\varepsilon}$, etc. Dans un modèle empirique, le comportement anélastique peut être représenté par des fonctions qui capturent les déformations permanentes et les non-linéarités observées dans les données expérimentales. Les divers comportements qu'un matériau peut présenter en réponse aux différents types de charge sont symbolisés par les modules présentés sur la Figure 2.1.

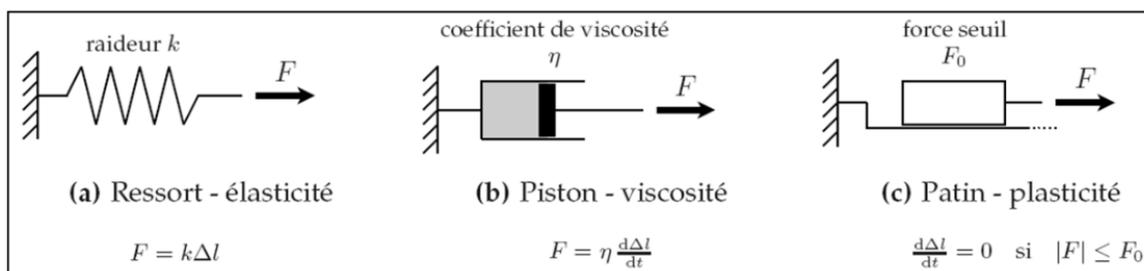


FIGURE 2.1 – Modules de base pour le comportement rhéologique des matériaux [Hor, 2011].

En combinant ces modules, on parvient à dépeindre de manière plus élaborée le comportement global du matériau. Parmi les modèles qui intègrent ces modules, le modèle de Bingham [Lurdos, 2008] se distingue comme un exemple illustratif du comportement élastoviscoplastique observé pour certains matériaux. La représentation et la modélisation de ce comportement sont exposées sur la Figure 2.2. Les modèles empiriques se subdivisent en modèles quasi-statiques, modèles découplés et en modèles couplés.

2.2.1 Modèles quasi-statiques

Un modèle de comportement quasi-statique est une représentation mathématique utilisée pour décrire la réaction d'un matériau ou d'une structure soumis à des charges appliquées à des taux de variation considérablement lents par rapport au temps caractéristique du matériau. Du fait de ce temps long, le matériau peut ajuster progressivement sa réponse en fonction des contraintes appliquées, ce qui permet d'atteindre un état d'équilibre à chaque étape du processus de chargement.

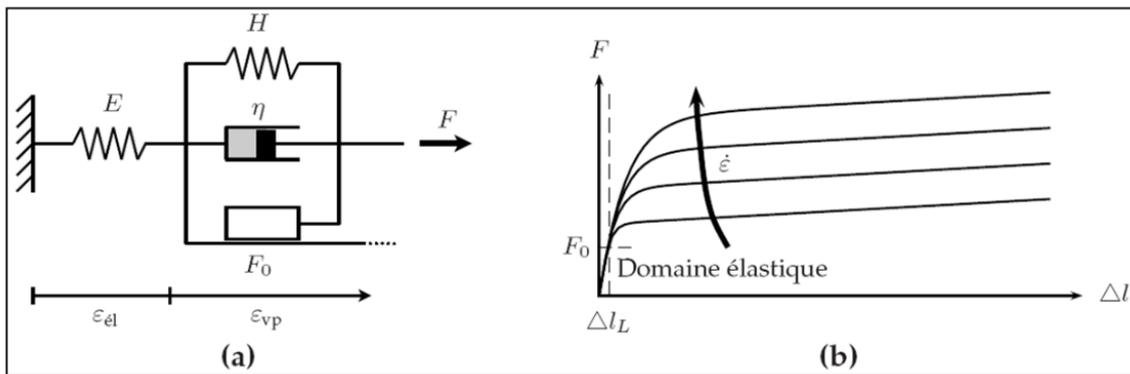


FIGURE 2.2 – Représentation schématique du modèle de Bingham (a) et réponse associée pour une sollicitation en traction (b) [Lurdos, 2008].

Les premiers modèles quasi-statiques conçus pour décrire la corrélation entre la contrainte et la déformation se concentraient exclusivement sur la variable de déformation ε , négligeant ainsi les deux autres variables, à savoir la vitesse de déformation $\dot{\varepsilon}$ et la température T considérées fixes. Quelques exemples de ces modèles sont donnés par les équations ci-dessous :

- Modèle de Ludwik [Ludwik, 1909] : Le modèle de Ludwik peut être conceptualisé comme la combinaison de deux composants distincts. Le premier composant traduit la limite élastique du matériau, tandis que le second rend compte de l'influence de la déformation, englobant ainsi la prise en considération de la plasticité du matériau :

$$\sigma = \sigma_0 + K\varepsilon^n, \quad (2.2)$$

avec σ_0 , K et n les paramètres du modèle.

- Modèle de Hollomon [Atkinson, 1982] : Le modèle de Hollomon est une expression qui combine la limite d'élasticité et la déformation sous forme d'un produit de deux facteurs, comme illustré dans l'équation suivante :

$$\sigma = K\varepsilon^n, \quad (2.3)$$

avec K et n les paramètres du modèle.

- Modèle de Prager [Prager, 1955] : Le modèle de Prager formulé à la manière du modèle de Ludwik, se divise en deux termes distincts. La partie élastique est identique à celle du modèle de Ludwik tandis que la partie plastique diffère de la fonction utilisée pour sa description :

$$\sigma = \sigma_0 + \tanh(E\varepsilon/\sigma_0). \quad (2.4)$$

- Modèle de Voce [Voce, 1948] : Le modèle de Voce est un modèle qui utilise l'inverse d'une fonction exponentielle pour représenter le comportement plastique d'un matériau. Cette approche permet de prendre en considération le comportement non linéaire à l'adoucissement, même en l'absence de la variable température. Par conséquent, on peut anticiper un comportement similaire à l'adoucissement provoqué par l'effet thermique, bien que la température ne soit pas explicitement présente dans le modèle :

$$\sigma = \sigma_s + (\sigma_0 - \sigma_s) \exp(-\varepsilon/\varepsilon_0) K\varepsilon^n, \quad (2.5)$$

avec σ_s une contrainte de saturation.

- Modèle de Swift [Swift, 1952] : Le modèle de Swift est quant à lui décrit par une fonction puissance ; ce qui permet au modèle d'avoir une évolution toujours croissante en toute circonstance si le paramètre n est positif :

$$\sigma = \sigma_0(\varepsilon + \varepsilon_0)^n. \quad (2.6)$$

Un aperçu global de ces modèles est présenté sur la Figure 2.3. On peut noter que ces modèles présentent divers comportements, chacun étant caractérisé par sa propre formulation. Les modèles de Hollomon et de Swift se distinguent particulièrement par leur évolution marquée. Cette caractéristique découle directement de leur formulation reposant sur la fonction puissance, qui connaît une croissance rapide après à la fonction exponentielle.

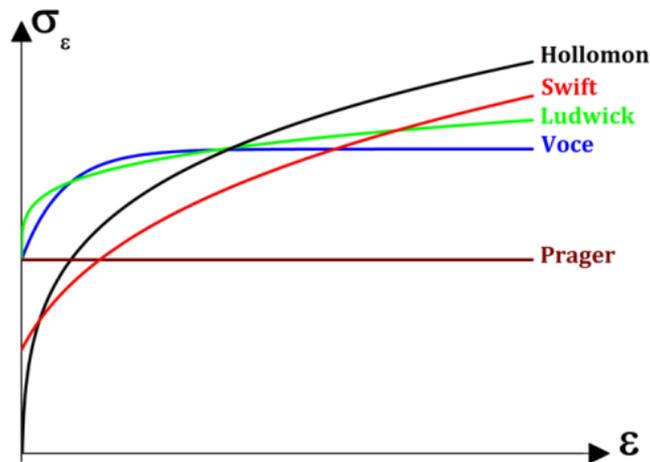


FIGURE 2.3 – Évolution de quelques modèles quasi-statiques [Hor, 2011].

2.2.2 Modèles découplés

Le concept de découplage, appliqué ici pour décrire les modèles de comportement, repose sur la notion que certains modèles prennent en compte de manière distincte les influences de la déformation ε , de la vitesse de déformation $\dot{\varepsilon}$ et de la température T . En général, les modèles quasi-statiques présentés dans la section précédente sont multipliés par des modules dans le but de décrire empiriquement l'influence de la température et de la vitesse de déformation. Cette approche permet ainsi de considérer trois facteurs distincts, chacun représentant l'effet de l'une de ces variables physiques comme illustrée par l'équation suivante :

$$\sigma = \sigma_{\varepsilon}(\varepsilon) \times \sigma_{\dot{\varepsilon}}(\dot{\varepsilon}) \times \sigma_T(T), \quad (2.7)$$

où $\sigma_{\varepsilon}(\varepsilon)$ est la fonction qui décrit l'effet de la déformation, $\sigma_{\dot{\varepsilon}}(\dot{\varepsilon})$ celle qui traduit la sensibilité à la vitesse de déformation et $\sigma_T(T)$ est la fonction qui prend en compte la sensibilité à la température.

Les Tableaux 2.1 et 2.2, regroupent certaines fonctions qui caractérisent respectivement la vitesse de déformation et la température pour divers modèles. La combinaison de ces fonctions permet de créer des modèles, qui peuvent être plus ou moins complexes, en prenant en compte les effets de ces variables physiques de manière séparée. Cependant, la véritable interconnexion de ces effets n'est pas encore atteinte. Une étude ultérieure des modèles couplés, comme explorée plus en détail dans ce travail, permettra d'observer les effets résultant de la combinaison de ces variables interdépendantes.

2.2.2.1 Modèle de Johnson–Cook

Le modèle de Johnson–Cook [Johnson et Cook, 1983] est aujourd'hui très répandu et couramment intégré dans les logiciels de simulation par éléments finis, en raison de sa simplicité et du faible nombre de paramètres qu'il nécessite. Il représente une extension du modèle de plasticité isotrope

TABLE 2.1 – Fonctions vitesse de déformation de quelques modèles.

Modèles	Fonctions	Paramètres
Loi de puissance	$\dot{\epsilon}^m$	m
Cowper et Symonds [Cowper et Symonds, 1957]	$1 + (\dot{\epsilon}/\dot{\epsilon}_0)^m$	$\dot{\epsilon}_0, m$
Ludwik [Ludwik, 1909]	$\log(\dot{\epsilon}/\dot{\epsilon}_0)$	$\dot{\epsilon}_0$
Sokolowsky	$\log(1 + (\dot{\epsilon}/\dot{\epsilon}_0))$	$\dot{\epsilon}_0$
Johnson et Cook [Johnson et Cook, 1983]	$1 + C \ln(\dot{\epsilon}/\dot{\epsilon}_0)$	$\dot{\epsilon}_0, C$

TABLE 2.2 – Fonctions température de quelques modèles.

Modèles	Fonctions	Paramètres
Loi de puissance	$T^{-\nu}$	ν
Johnson et Cook [Johnson et Cook, 1983]	$1 - \left(\frac{T-T_0}{T_f-T_0}\right)^m$	T_0, m
Litonsky	$1 - T/T_0$	T_0
Vinh et al. [Vinh et al., 1980]	$\exp(W/T)$	W
Zhao et Gary [Zhao et Gary, 1996]	$1 - \beta(T - T_0)$	T_0, β
Arrhénus	$\exp(-Q/RT)$	Q

classique, incorporant un terme d'évolution de la déformation qui prend en compte la sensibilité à la température et à la vitesse de déformation. Ce modèle est formulé à l'aide de l'équation suivante :

$$\sigma = (A + B\epsilon^n) \left[1 + C \ln \left(\frac{\dot{\epsilon}}{\dot{\epsilon}_0} \right) \right] \left[1 - \left(\frac{T - T_0}{T_f - T_0} \right)^m \right], \quad (2.8)$$

dans laquelle $\dot{\epsilon}_0$ est la vitesse de déformation de référence, T_f et T_0 sont respectivement les températures de fusion et de référence (ou initiale) du matériau. Dans le cas de cette étude, les valeurs des ces températures sont $T_f = 1420^\circ\text{C}$ et $T_0 = 1050^\circ\text{C}$. Les cinq paramètres du modèle sont notés A, B, C, n et m . Divers auteurs comme Holmquist et Johnson [Holmquist et Johnson, 1991, Duc-Toan et al., 2012] et bien d'autres ont apporté des modifications à ce modèle afin d'intégrer la relation d'interdépendance des trois variables physiques. Ces formes modifiées seront examinées plus en détail dans la section consacrée aux modèles couplés.

Du fait de sa large utilisation en simulation numérique et de sa présence dans le code de calcul Abaqus, ce modèle de comportement, bien que souffrant de certaines lacunes, sera utilisé dans la suite de cette étude.

2.2.2.2 Lois puissance et exponentielle

Les lois puissance et exponentielle sont des formulations couramment utilisées pour décrire le comportement du matériau comme c'est le cas dans [Atkinson, 1982]. Parmi ces modèles, il existe deux

lois de comportement découplées qui utilisent trois paramètres pour décrire l'effet de la déformation, de la vitesse de déformation et la température respectivement. Ces modèles sont donnés par :

$$\sigma = \sigma_0^n \dot{\varepsilon}^m T^\nu, \quad (2.9)$$

et

$$\sigma = \sigma_0 \varepsilon^n \dot{\varepsilon}^m \exp\left(\frac{Q}{RT}\right), \quad (2.10)$$

avec n la sensibilité à la déformation, m la sensibilité à la vitesse de déformation, ν l'effet de la température, Q (J mol⁻¹) l'énergie d'activation du matériau et $R = 8.314$ J mol⁻¹ K⁻¹ la constante des gaz parfaits. Ces deux lois permettent de mettre en évidence les paramètres rhéologiques qui permettent de décrire l'écoulement du matériau.

Comme l'élévation de la température d'un matériau équivaut à une augmentation de l'énergie thermique, il est souvent préférable d'utiliser l'énergie d'activation pour décrire comment cette élévation de température influence le matériau. Cette approche est souvent préférée à la sensibilité directe à la température. Dans ce contexte, on recourt généralement à une forme exponentielle plutôt qu'à une loi de puissance. De plus, cette forme exponentielle intègre implicitement le paramètre de Zener-Hollomon défini par la relation :

$$Z = \dot{\varepsilon} \exp\left(\frac{Q}{RT}\right), \quad (2.11)$$

qui établit une relation entre la température et la vitesse de déformation d'un matériau [Zener et Hollomon, 1944]. Le paramètre Z permet de quantifier l'effet de la température et de la vitesse de déformation sur la déformation du matériau et par conséquent une valeur élevée de Z indique une plus grande sensibilité à la température et/ou à la vitesse de déformation, ce qui signifie que la déformation du matériau sera plus significative à des températures élevées et/ou à des vitesses de déformation plus rapides.

2.2.2.3 Modèle de Sellars

Le modèle de Sellars, également connu sous le nom de modèle de Sellars-McTegart [Sellars, 1985, Sellars, 1995], est un modèle utilisé pour décrire le comportement viscoplastique des matériaux métalliques à chaud. Ce modèle prend en compte l'effet de la température, de la vitesse de déformation et de la déformation accumulée sur le comportement du matériau. Le modèle de Sellars est principalement utilisé pour modéliser le comportement des métaux lors du formage à chaud et des procédés de forgeage. Il est basé sur l'hypothèse que le comportement viscoplastique des matériaux à chaud est influencé par la déformation accumulée, la vitesse de déformation et la température. Dans la formulation de Sellars, la relation entre la contrainte et la déformation est généralement exprimée par l'équation suivante :

$$\sigma = \sigma_0 \varepsilon^n (1 + (\dot{\varepsilon}/\dot{\varepsilon}_0)^m) \exp\left(\frac{Q}{RT}\right), \quad (2.12)$$

où ε est la déformation accumulée, σ_0 et $\dot{\varepsilon}_0$ déterminent la résistance initiale du matériau et sa vitesse de déformation de référence, respectivement et n et m quantifient l'effet de la déformation et de la vitesse de déformation accumulée sur le comportement viscoplastique du matériau.

2.2.3 Modèles couplés

La formulation couplée d'une loi de comportement implique l'intégration de multiples phénomènes ou variables interdépendantes pour décrire le comportement d'un matériau. On a recours à cette

approche lorsque le comportement d'un matériau est influencé par plusieurs phénomènes simultanément, et qu'il n'est pas possible de les traiter de manière indépendante. Par exemple, dans le cas d'un matériau viscoélastique, la formulation couplée tient compte à la fois de la déformation mécanique et de la réponse visqueuse du matériau, en prenant en considération des paramètres tels que la vitesse de déformation, le temps, la température, et d'autres facteurs. De même, pour un matériau thermomécanique, la formulation couplée prend en compte les interactions entre les champs de température et de déformation mécanique. Cette approche vise à mieux représenter le comportement réel du matériau.

2.2.3.1 Modèle de Khan et de Liang

Le modèle proposé par Khan et Liang [Lurdos, 2008] est donné par l'équation suivante :

$$\sigma = \left[A + B \left(1 - \frac{\ln \dot{\varepsilon}}{\ln \dot{\varepsilon}_0} \right) \varepsilon^n \right] \dot{\varepsilon}^C \left[1 - \left(\frac{T - T_0}{T_f - T_0} \right)^m \right]. \quad (2.13)$$

Cette loi de comportement est une forme modifiée du modèle Johnson–Cook dans laquelle le premier facteur intègre à la fois la sensibilité à la déformation et à la vitesse de déformation. Le deuxième facteur concerne exclusivement la sensibilité à la vitesse de déformation, tandis que le dernier facteur se rapporte à la sensibilité à la température. Son avantage par rapport au modèle de Johnson–Cook réside dans le seul fait qu'il peut prendre en compte l'interdépendance entre la vitesse de déformation et la température.

2.2.3.2 Modèle de Bäker

Contrairement au modèle de Johnson–Cook, le modèle de Bäker [Bäker, 2006] traduit l'adoucissement thermique par une fonction exponentielle et introduit un couplage entre la déformation et la vitesse de déformation, ce qui permet d'estimer simultanément les effets de l'écroutissage et les effets liés à la vitesse de déformation. On introduit ainsi un couplage des variables physiques. Ce modèle est établi dans le cadre d'une simulation de la formation de copeaux au cours de l'usinage. Ce modèle a été conçu de manière à ce que l'adoucissement dû à la température entre en compétition avec l'écroutissage provoqué par la déformation. Cette compétition entre adoucissement et écroutissage joue un rôle fondamental dans l'amorçage et l'amplification de la localisation de la déformation, un phénomène à l'origine de la formation des bandes de cisaillement adiabatique. Ce modèle présente ainsi un couplage entre les effets de l'écroutissage, de la vitesse de déformation et de la température. Le modèle de Bäker est donné comme suit :

$$\sigma = \left[A \varepsilon^{n_0 f(T)} \right] \left[1 + C \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right) \right] f(T) \quad \text{et} \quad f(T) = \exp \left[- \left(\frac{T}{T_0} \right)^m \right], \quad (2.14)$$

avec A , n_0 , C , T_0 et m les paramètres du modèle.

2.2.3.3 Modèle de Coupard (TANH)

Le modèle de Coupard [Coupard, 1993] constitue une autre forme d'extension du modèle Johnson–Cook, visant à incorporer le phénomène d'adoucissement, qui n'est pas pris en compte dans la formulation initiale de ce dernier. son expression est définie par l'expression suivante :

$$\sigma = [A + B \varepsilon^n] \left[1 + C \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right) \right] \left[1 - \left(\frac{T - T_0}{T_f - T_0} \right)^m \right] \left[D(\varepsilon, T) + (1 - D(\varepsilon, T)) \tanh \left(\frac{1}{\varepsilon + \varepsilon_0} \right) \right], \quad (2.15)$$

avec :

$$D(\varepsilon, T) = 1 - \left(\frac{p\varepsilon^r}{1 + p\varepsilon^r} \right) \tanh \left(\frac{T - T_0}{T_{rec} - T_0} \right)^q. \quad (2.16)$$

Dans cette formulation, $D(\varepsilon, T)$ est une fonction qui lie l'adoucissement à la température et la déformation. Dans l'équation (2.16), p contrôle la pente de la décroissance de la contrainte après le pic, T_{rec} contrôle la température d'apparition de l'adoucissement et q contrôle l'intervalle de température sur lequel apparaît le phénomène de *strain softening*. Selon l'auteur, ce modèle a l'avantage de conserver la validité des paramètres déjà identifiés pour le modèle de base de Johnson–Cook. Il suffit juste d'identifier les paramètres (ε_0 , p , q , r et T_{rec}) relatifs à la nouvelle formulation.

2.2.3.4 Modèle de Lurdos et Montheillet

Lurdos et Montheillet [Lurdos, 2008] ont développé un modèle dont la formulation est basée sur la loi de Voce présentée dans la Section 2.2.1. Cette loi, initialement définie en fonction de la déformation, tend à converger vers une contrainte de saturation σ_s . Lurdos et Montheillet [Lurdos, 2008] ont ajouté un terme additionnel ($A\varepsilon^n$) pour représenter de manière plus précise un pic de contrainte observé dans certains cas et pour certains types de matériaux. L'expression de ce modèle est donnée comme suit :

$$\sigma = \sigma_s(Z) + \left(\sigma_0(Z) - \sigma_s(Z) + A(Z)\varepsilon^{n(Z)} \right) \exp(-r(Z)\varepsilon), \quad (2.17)$$

où chaque coefficient de l'équation (2.17) est donné par :

$$\begin{cases} \sigma_s(Z) = \sigma_s^0 Z^{m_s} \\ \sigma_0(Z) = \sigma_0^0 Z^{m_0} \\ r(Z) = r_0 Z^{m_r} \\ A(Z) = A_0 Z^{m_A} \\ n(Z) = n_0 Z^{m_n} \end{cases}, \quad (2.18)$$

avec m_0 , m_s , m_A , m_r et m_n les constantes qui mettent en exergue les sensibilités à la vitesse de déformation, σ_s^0 , σ_0^0 , r_0 , n_0 et A_0 les constantes adimensionnelles. Comme introduit précédemment, Z est le paramètre Zener–Hollomon. Il est important de noter qu'à partir de ce modèle on constate que toutes les constantes ont la même fonction d'évolution; ceci n'est qu'une hypothèse simplificatrice pour l'identification du modèle qui pourrait limiter la performance de ce modèle. Une comparaison entre la loi de Voce conventionnelle et la version modifiée par Lurdos est illustrée sur la Figure 2.4.

2.2.3.5 Modèle de Mulyadi

Lors du forgeage d'un alliage de Titane, Mulyadi et al. [Mulyadi et al., 2006] ont proposé un modèle analytique qui permet de prendre en compte la compétition écrouissage–adoucissement. La fonction décrivant l'écrouissage est multipliée par un facteur λ évoluant de façon exponentielle avec la déformation vers un état d'équilibre $\lambda_s = \lambda_0 Z^q$. Le paramètre λ est donné par :

$$\lambda = \lambda_s + (1 - \lambda_s) \exp(-\alpha\varepsilon). \quad (2.19)$$

La relation qui constitue le modèle de Mulyadi est donnée par :

$$\sigma = K\lambda Z^m (1 - \exp(-\beta\varepsilon)), \quad (2.20)$$

avec K , λ_0 , m , q , α et β les paramètres du modèle. Ce modèle est utilisé dans le cas du forgeage à chaud ou déformation à chaud au sens large.

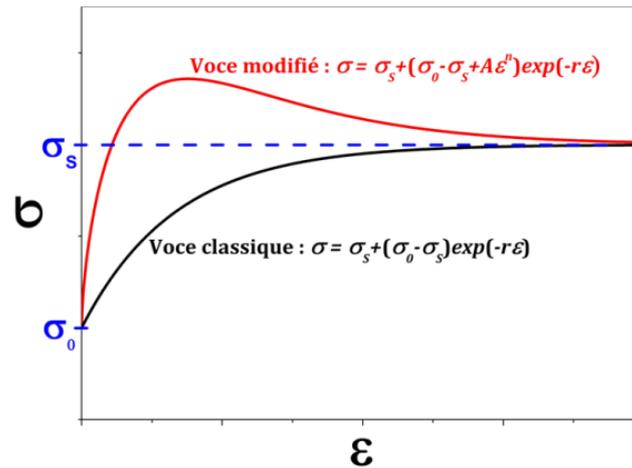


FIGURE 2.4 – Évolution de la loi de Voce et de sa forme modifiée par Lurdos [Lurdos, 2008].

2.2.3.6 Modèle de Yong-Cheng Lin

Le modèle de Yong-Cheng [Lin *et al.*, 2010] est une des formes modifiées du modèle de Johnson-Cook qui prend en compte des effets couplés de la vitesse de déformation et de la température. Ce modèle est donné par l'équation ci-dessous :

$$\sigma = \left(A_1 + B_1 \varepsilon + B_2 \varepsilon^2 \right) \left[1 + C_1 \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right) \right] \exp \left[\left(\lambda_1 + \lambda_2 \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right) \right) (T - T_0) \right], \quad (2.21)$$

où A_1 , B_1 , B_2 , C_1 , λ_1 et λ_2 sont les paramètres du modèle et les autres coefficients sont identiques au modèle original de Johnson-Cook à la seule différence que ce modèle ne prend pas en compte la température de fusion T_f du matériau. Pour le développement de ce modèle, un alliage d'acier a été utilisé comme matériau d'étude lors d'un essai de traction. Cette étude est menée pour une plage de vitesse de déformation ($\dot{\varepsilon} = 10^{-4} \text{ s}^{-1} - 10^{-2} \text{ s}^{-1}$) et une plage de température ($T = 850 \text{ °C} - 1100 \text{ °C}$). En raison des conditions d'essai qui ont servi de fondement à l'élaboration de ce modèle, il est possible de considérer ce modèle comme quasi-statique. Le même auteur a également proposé un autre modèle, qui est essentiellement une variante modifiée du modèle de Johnson-Cook. Ce modèle permet de réaliser un couplage entre les trois paramètres de déformation à chaud. L'équation qui définit ce modèle est présentée ci-après :

$$\sigma = \left(A + B(T) \varepsilon^{n(T)} \right) \left[1 + C(T, \varepsilon) \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right) \right], \quad (2.22)$$

où B et n sont des fonctions de la température tandis que C est une fonction de la déformation et de la température qui n'a pas une forme explicite mais dépend de l'allure des différents points la représentant graphiquement. Les autres paramètres sont similaires à ceux de la forme originale du modèle de Johnson-Cook. Ce modèle est formulé pour un essai de compression réalisé sur un alliage d'Al-Cu-Mg pour une plage de température de ($T = 380 \text{ °C} - 470 \text{ °C}$) et une plage de vitesse de déformation de ($\dot{\varepsilon} = 10^{-2} \text{ s}^{-1} - 1 \text{ s}^{-1}$). Plus de détails sur le modèle se trouvent dans [Lin *et al.*, 2012].

2.2.3.7 Modèle de Hong

Hong *et al.* [Li *et al.*, 2013] ont élaboré une variante du modèle de Yong-Cheng [Lin *et al.*, 2010], où les paramètres λ_1 et λ_2 sont traités en tant que fonctions. Plus précisément, ils définissent λ_1 comme

une fonction de la déformation, tandis que λ_2 dépend à la fois de la vitesse de déformation et de la température. L'équation qui définit le modèle est donnée ci-dessous :

$$\sigma = \left(A_1 + B_1 \varepsilon + B_2 \varepsilon^2 \right) \left[1 + C_1 \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right) \right] \exp \left[\left(\lambda_1(\varepsilon) + \lambda_2(\dot{\varepsilon}, T) \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right) \right) (T - T_0) \right]. \quad (2.23)$$

Il est évident que cette formulation permet de générer plusieurs modèles en ajustant simplement les paramètres de la fonction λ_1 .

2.2.3.8 Modèle d'Anis Hor

Le modèle de Hor [Hor, 2011] est une forme améliorée du modèle de Coupard. L'équation du modèle est :

$$\begin{aligned} \sigma = & [A + B\varepsilon^n] \left[D(\varepsilon, T) + (1 - D(\varepsilon, T)) \tanh \left(\frac{1}{\varepsilon + \varepsilon_0} \right) \right] \\ & \times \left[1 - m_1 \left(\frac{T - T_0}{T_f - T_0} \right)^{m_2} \right] \left[1 + C(T) \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right) \right], \end{aligned} \quad (2.24)$$

avec :

$$\begin{cases} D(\varepsilon, T) = 1 - \left(\frac{p\varepsilon}{1 + p\varepsilon} \right) \tanh \left(\frac{T - T_0}{T_{rec} - T_0} \right) \\ C(T) = \frac{C_1 \exp(C_2 T / T_f)}{T / T_f} \end{cases}. \quad (2.25)$$

Ce modèle est une forme simplifiée du modèle de Coupard où deux paramètres (m_1 et m_2) modélisent la sensibilité à la température. Dans cette formulation, la sensibilité à la vitesse de déformation est considérée comme une fonction exponentielle dépendant de deux paramètres C_1 et C_2 .

Il faut noter que deux hypothèses soutiennent la formulation de ce modèle. Selon l'auteur, les modèles couplés tels que le modèle de Lurdos nécessitent d'atteindre une contrainte de saturation, ce qui n'est pas facile à réaliser en temps réel. De plus le modèle de Lurdos contient plusieurs paramètres à déterminer. Il est donc judicieux de formuler un modèle qui soit capable de prendre en compte un quelconque état de déformation et avec moins de paramètres à déterminer.

Pour élaborer ce modèle, l'auteur s'appuie sur des observations expérimentales et effectue plusieurs types de couplages. Tout d'abord, il réalise un couplage qui prend en compte la compétition entre l'érouissage et l'adoucissement, ce qui se manifeste à travers les effets d'érouissage et les effets de la température. Ensuite, il opère un autre couplage entre les effets de la vitesse de déformation et les effets de la température. La rivalité entre l'érouissage et l'adoucissement se reflète dans les deux premiers facteurs du modèle. Les sensibilités à la température et à la vitesse de déformation sont quant à elles représentées par les deux derniers facteurs de la même équation.

2.2.4 Modèles semi-empiriques

Tous les modèles précédemment présentés sont des représentations mathématiques simplifiées, adaptées en fonction des simulations numériques à effectuer. Ces formulations ne sont pas directement basées sur des phénomènes physiques observables lors de la déformation des matériaux. Cependant, il est important de souligner que l'incorporation des réalités physiques dans le développement de modèles analytiques permet de prédire avec précision les phénomènes expérimentaux qui sont observables à partir des simulations numériques.

Il est également pertinent de noter que la majorité de ces modèles a été conçue à une époque où les capacités informatiques étaient limitées. Ainsi, avec l'évolution et la puissance croissante des ordinateurs, il est devenu possible de concevoir des modèles qui intègrent de manière plus approfondie les phénomènes physiques observés lors des expériences, ou du moins qui s'en rapprochent davantage. En fin de compte, les modèles semi-empiriques se positionnent comme une approche intermédiaire entre les modèles empiriques et les formulations basées sur des principes physiques. Quelques exemples de ces modèles sont présentés dans cette section.

2.2.4.1 Modèle de Klepaczko

Klepaczko et *al.* [Klepaczko, 1975] sont parmi les pionniers dans l'élaboration de modèles qui intègrent certains phénomènes physiques observés au niveau de la microstructure. En effet, ils ont adopté une approche qui repose sur l'utilisation de la densité de dislocations en tant que variable interne où la contrainte est décomposée en deux composantes : l'une résultant de la création de dislocations immobiles, et l'autre associée aux phénomènes activés thermiquement. Ainsi la forme du modèle est donnée par :

$$\sigma = \frac{\mu(T)}{\mu_0} [\sigma_d(\varepsilon, \dot{\varepsilon}, T) + \sigma_{th}(\dot{\varepsilon}, T)], \quad (2.26)$$

avec $\mu(T)$ le module de cisaillement, σ_d la contrainte liée à la densité de dislocations du matériau donnée par $\sigma_d = \alpha\mu b\sqrt{\rho}$. Le second terme est déterminé par la physique de la déformation thermiquement activée. La version améliorée du modèle proposé par Klepaczko et *al.* [Klepaczko, 1987] donne directement les expressions de ces deux fonctions selon les équations suivantes :

$$\sigma_d = \sigma_0(\varepsilon + \varepsilon_0)^n, \quad (2.27)$$

et

$$\sigma_{th} = \sigma_{th}^0 \left[1 + C_1 \frac{T}{T_f} \log\left(\frac{\dot{\varepsilon}_m}{\dot{\varepsilon}}\right) \right]^m, \quad (2.28)$$

où ε_0 , σ_{th}^0 , C_1 et m sont des constantes du matériau et $\dot{\varepsilon}_m$ la vitesse de déformation maximale admissible ($\dot{\varepsilon}_m \approx 10^7 \text{ s}^{-1}$). De plus, les paramètres σ_0 et n dépendent de la déformation et de la température :

$$\sigma_0 = C_2 \left[\frac{T}{T_f} \log\left(\frac{\dot{\varepsilon}_m}{\dot{\varepsilon}}\right) \right]^{-\nu}, \quad (2.29)$$

et

$$n = n_0 \left[1 - C_3 \frac{T}{T_f} \log\left(\frac{\dot{\varepsilon}_0}{\dot{\varepsilon}}\right) \right], \quad (2.30)$$

avec $\dot{\varepsilon}_0$ la vitesse minimale admissible, C_2 , C_3 et ν les constantes du matériau. Cette loi de comportement a été développée dans le cadre de la simulation des tôles d'aluminium pour l'automobile.

2.2.4.2 Modèle de Zerilli–Armstrong

Zerilli et Armstrong [Zerilli et Armstrong, 1987] ont développé un modèle se basant sur la microstructure du matériau et dont la forme finale dépend de la structure cristalline du métal étudié, comme structure cubique à face centrée (C.F.C.), structure cubique centrée (C.C.) ou structure hexagonale centrée (H.C.). Dans le cas de la structure C.C., les vallées de Peierls (déformations périodiques ou des perturbations de la structure cristalline d'un matériau qui se forment à des défauts dans un cristal) constituent les barrières principales aux mouvements des dislocations alors que dans une structure C.C. les dislocations sont stoppées par la forêt de dislocations. Une forme de l'énergie des

obstacles aux dislocations est également proposée par les auteurs et l'équation est donnée comme suit :

$$\Delta G = \frac{k}{C_7} \ln \left(\frac{\sigma_{th0}}{\sigma_{th}} \right) - kT \frac{C_6}{C_7}, \quad (2.31)$$

avec C_6 et C_7 constantes du matériau, σ_{th0} la contrainte thermique qui varie en fonction de la structure cristalline du matériau. Pour les métaux de structure C.C., σ_{th0} est une constante égale à C_5 , tandis que pour les métaux de structure C.F.C., σ_{th0} est donnée par $\sigma_{th0} = C_3 \varepsilon^{1/2}$, où ε est une variable dépendant de la déformation.

La composante athermique du modèle dépend également de la structure du matériau. Pour les métaux de structure C.C., la contrainte athermique σ_{ath} est définie comme $\sigma_{ath} = C_1 + C_2/\sqrt{D} + C_4 \varepsilon^n$, où C_1 , C_2 et C_4 sont des constantes du matériau, D représente la taille moyenne des grains, et ε^n est une fonction de la déformation. En revanche, pour les métaux de structure C.F.C., la composante athermique s'exprime comme $\sigma_{ath} = C_1 + C_2/\sqrt{D}$, sans dépendance vis-à-vis de la déformation.

En combinant les termes thermiques et athermiques, le modèle final prend la forme suivante :

$$\text{C.F.C. : } \sigma = C_1 + \frac{C_2}{\sqrt{D}} + C_3 \sqrt{\varepsilon} \exp(-C_6 T + C_7 T \ln \dot{\varepsilon}). \quad (2.32)$$

et

$$\text{C.C. : } \sigma = C_1 + \frac{C_2}{\sqrt{D}} + C_4 \varepsilon^n + C_5 \exp(-C_6 T + C_7 T \ln \dot{\varepsilon}). \quad (2.33)$$

Zerilli et Armstrong [Zerilli et Armstrong, 1995] ont plus tard proposé un autre modèle de comportement qui s'applique aux métaux de structure cristalline hexagonale compactes (H.C.) et certains alliages de comportement intermédiaire entre les métaux de structure (C.C.) et (C.F.C.). Dans ce cas, les dislocations sont arrêtées par les vallées de Peierls et par la forêt de dislocations. L'équation de cette loi de comportement est donnée ci-après :

$$\sigma = C_1 + \frac{C_2}{\sqrt{D}} + C_3 \sqrt{\varepsilon} \exp(-C_6 T + C_7 T \ln \dot{\varepsilon}) + C_5 \exp(-C'_6 T + C'_7 T \ln \dot{\varepsilon}), \quad (2.34)$$

où les variables C_i , C'_6 , C'_7 sont les paramètres du modèle. Le développement de toutes les formulations des modèles proposés par Zerilli et Armstrong est basé sur le développement limité par rapport à la variable $x = kT \ln(\dot{\varepsilon}/\dot{\varepsilon}_0)/C_0$ qui n'est valable pour $x \ll 1$ avec C_0 l'énergie des obstacles à 0 K.

Toutes ces hypothèses limitent le domaine d'application du modèle dans le sens où il n'est applicable que pour de faibles températures et vitesses de déformation. Ainsi des formes modifiées ont été proposées pour améliorer le domaine d'applicabilité du modèle de Zerilli–Armstrong.

Voyiadjis et Abed [Voyiadjis et Abed, 2005] ont développé d'une part deux modèles en reprenant complètement les travaux de Zerilli et Armstrong sans utiliser leur approximations et d'autre part deux autres modèles ont été développés par Kocks et al. [Kocks, 1975] où la méthode de Voyiadjis et Abed est appliquée à l'énergie d'activation libre. Les deux formulations qui sont une extension du modèle de Zerilli–Armstrong sont données par :

$$\text{C.F.C. : } \sigma = C_1 + (C_2 + C_3 \varepsilon^n)(1 + C_4 T^m - C_5 T + C_6 T \ln \dot{\varepsilon}), \quad (2.35)$$

et

$$\text{C.C. : } \sigma = C_1 + C_2 \varepsilon^n + C_3(1 + C_4 T^m - C_5 T + C_6 T \ln \dot{\varepsilon}). \quad (2.36)$$

Les deux autres expressions développées par [Kocks, 1975] sont données par :

$$\text{C.F.C. : } \sigma = C_1 + C_2 \varepsilon^n \left[1 - (C_4 T - C_5 T + C_6 T \ln \dot{\varepsilon})^{1/q} \right]^{1/q}, \quad (2.37)$$

et

$$\text{C.C. : } \sigma = C_1 + C_2 \varepsilon^n + C_3 \left[1 - (C_4 T - C_5 T + C_6 T \ln \dot{\varepsilon})^{1/q} \right]^{1/q}. \quad (2.38)$$

où $C_{i \in \{1,6\}}$, n , m , p et q sont les paramètres du modèle. L'avantage de ces deux modèles réside dans le fait qu'ils sont applicables à haute température et vitesse de déformation et donc très utiles pour la simulation des procédés industriels où la microstructure est sensible à ces deux grandeurs macroscopiques. Étant donné que les propriétés des matériaux dépendent à la fois des conditions expérimentales et de leur composition chimique, Zerilli et Armstrong ont par la suite introduit une autre formulation qui ne fait pas appel à une structure matérielle spécifique :

$$\sigma = A_0 + A_1 \varepsilon^n \exp \left[-A_2 T + A_3 \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right) \right], \quad (2.39)$$

où A_0 représente la contrainte athermique, T désigne la température absolue du matériau, et A_1 , A_2 , A_3 ainsi que n sont les paramètres du modèle, il est important de noter que ce modèle se divise en deux composantes distinctes. D'une part, il y a la contrainte athermique qui représente une limite élastique constante du matériau. D'autre part, il y a une composante qui tient compte de l'adoucissement du matériau, lequel est influencé par la température et la vitesse de déformation.

2.2.4.3 Modèle Modifié de Zerilli–Armstrong (MZA)

Samantaray et al. [Samantaray et al., 2009b] ont proposé une version améliorée du modèle de Zerilli–Armstrong à partir des essais de compression effectués sur un alliage de titane D9 pour une large gamme de températures ($T = 800^\circ\text{C} - 1200^\circ\text{C}$), de déformations ($\varepsilon = 0.1 - 0.5$) et de vitesses de déformation ($\dot{\varepsilon} = 10^{-3} \text{ s}^{-1} - 1 \text{ s}^{-1}$). La formulation de ce modèle est donnée par l'équation :

$$\sigma = (C_1 + C_2 \varepsilon^n) \exp \left[-(C_3 + C_4 \varepsilon)(T - T_0) + (C_5 + C_6(T - T_0)) \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right) \right], \quad (2.40)$$

où C_i et n sont les paramètres du modèle ; les autres coefficients étant identiques au modèle original. En testant le modèle originel de Zerilli–Armstrong sur l'alliage D9, il s'est avéré incapable de décrire son comportement. C'est ainsi qu'une forme modifiée du modèle de Zerilli–Armstrong a été proposée en s'inspirant de la forme originale, l'objectif étant de rendre cette forme modifiée applicable au travail à hautes températures.

Le modèle de Zerilli–Armstrong divise la contrainte d'écoulement en deux composantes, thermique et athermique. La composante athermique représente la limite élastique du matériau et dépend de la température. Cependant, des études ont révélé que la limite élastique de l'alliage D9 diminue au-delà d'une certaine valeur de température ($T = 830^\circ\text{C}$), ce qui signifie que cette composante athermique ne peut plus être considérée dans la nouvelle formulation.

Selon le modèle de Zerilli–Armstrong, le terme $A_1 \varepsilon^n$ est calculé en utilisant l'intersection de la droite $\ln(\sigma - A_0)$ en fonction de T . Cela implique que l'intersection du tracé de $\ln(\sigma - A_0)$ en fonction de T représente la contrainte à 0 K. Ces intersections pour différentes déformations sont ensuite tracées par rapport à $\ln \varepsilon$ pour obtenir les valeurs de A_1 et n . Cependant, il se peut que les données expérimentales ne soient pas toujours disponibles à 0 K, ce qui peut poser un problème. Pour remédier à cela, une température de référence est prise en compte, et ainsi, $A_1 \varepsilon^n$ est remplacé par $C_1 + C_2 \varepsilon^n$.

En ce qui concerne le coefficient A_2 dans le modèle de Zerilli–Armstrong, il est calculé comme la pente de la relation $\ln \sigma - A_0$ en fonction de T en supprimant les effets de la vitesse de déformation. De plus, il est supposé que cette pente est indépendante de la déformation. Cependant, Samantaray et al. [Samantaray et al., 2009b] ont démontré que cette pente dépend effectivement de la déformation. Par conséquent, A_2 est remplacé par $(C_3 + C_4 \varepsilon)$. De manière similaire, la prise en compte de l'effet de la température sur le paramètre A_3 a conduit à son remplacement par la fonction $(C_5 + C_6(T - T_0))$.

Ce modèle ayant une forte importance pour la suite (le modèle PTM développé au cours de cette étude est basé sur le modèle MZA), il fait partie de la liste des modèles retenus pour la suite de nos travaux.

2.2.4.4 Modèle de Hensel & Spittel

Le modèle Hensel-Spittel (HS) [Hensel et Spittel, 1978] est un modèle peu connu, pourtant implémenté dans le logiciel FORGE et très utilisé dans la simulation des procédés industriels complexes tels que le forgeage, le matriçage, le martelage, le laminage, etc. A la différence des modèles de Johnson–Cook, MZA, le modèle HS est facile d'utilisation dans le sens où la détermination de ses paramètres est très simple. Une simple programmation d'un code Python permet d'obtenir simultanément tous les paramètres du modèle. Le modèle de Hensel-Spittel est donné par l'équation suivante :

$$\sigma = A e^{m_1 T} \varepsilon^{m_2} \dot{\varepsilon}^{m_3} e^{(m_4/\varepsilon)} (1 + \varepsilon)^{m_5 T} e^{m_6 \varepsilon} \dot{\varepsilon}^{m_7 T} T^{m_8}, \quad (2.41)$$

où les coefficients A et m_i sont les 9 paramètres du modèle à identifier. Ce modèle présente certaines limites dans le sens où la précision du modèle dépend du nombre de paramètres utilisés lors de son identification. Certains auteurs [Rudnytskyj *et al.*, 2020, Chadha *et al.*, 2018b, El Mehtedi *et al.*, 2015] limitent l'utilisation de ce modèle à 5 ou 6 paramètres en forçant les autres paramètres à prendre la valeur zéro.

Du fait de sa présence comme modèle classique dans le logiciel FORGE, ce modèle de comportement fait partie également de la liste de modèles retenus pour la suite de cette étude.

2.2.4.5 Modèle d'Arrhenius

Le modèle d'Arrhenius [Arrhenius, 1889], initialement développé en chimie, a été adapté à la mécanique des matériaux [Samantaray *et al.*, 2011a, Mirzadeh et Najafizadeh, 2010]. Dans ce contexte, il établit une corrélation entre la contrainte d'écoulement, la déformation, la température et la vitesse de déformation en prenant en compte le paramètre de Zener–Hollomon défini par l'équation (2.11). Les équations qui définissent ce modèle sont les suivantes :

$$\dot{\varepsilon} = \begin{cases} A_1 \sigma^{n_1} \exp\left(-\frac{Q}{RT}\right) & \alpha\sigma < 0.8 \\ A_2 \exp(\beta\sigma) \exp\left(-\frac{Q}{RT}\right) & \alpha\sigma > 1.2 \\ A_3 [\sinh(\alpha\sigma)]^{n_2} \exp\left(-\frac{Q}{RT}\right) & \text{pour tout } \sigma \end{cases} \quad (2.42)$$

En combinant ces équations, on obtient l'équation suivante qui définit le modèle :

$$\sigma = \frac{1}{\alpha} \ln \left\{ \left(\frac{Z}{A} \right)^{1/n} + \left[1 + \left(\frac{Z}{A} \right)^{2/n} \right]^{1/2} \right\}, \quad (2.43)$$

où T la température absolue, A_i , α , n_i , n , β sont les constantes du matériau définissant le modèle avec $\alpha = \beta/n$. Plusieurs travaux ont utilisé ce modèle ainsi que sa forme modifiée et une comparaison avec les modèles connus comme le modèle Johnson–Cook, le modèle de Zerilli–Armstrong etc. a montré que le modèle d'Arrhenius a une bonne prédiction.

Plusieurs études, notamment celles menées par [Lin *et al.*, 2008b, Lin *et al.*, 2009, Abspoel *et al.*, 2016] ainsi que par [Gao, 2007, Gao *et al.*, 2020, Neto *et al.*, 2020], ont employé ce modèle, y compris sa version modifiée. Des comparaisons ont été effectuées avec des modèles bien établis tels que le modèle de Johnson–Cook ou le modèle de Zerilli–Armstrong modifié [He *et al.*, 2014], la combinaison

entre Johnson–Cook et Zerilli–Armstrong [Lin et Chen, 2010, Kardoulaki *et al.*, 2014, Quan *et al.*, 2016], entre autres. Les résultats de ces comparaisons ont révélé que le modèle d'Arrhenius présente une capacité de prédiction relativement satisfaisante. C'est la raison pour laquelle, il fait également partie de la liste des modèles retenus pour la suite de cette étude.

2.3 Modèle phénoménologique (PTM)

Dans cette section, nous présentons une loi de comportement basée sur la formulation de [Samantaray *et al.*, 2009b] dans le but d'étendre l'applicabilité de leur modèle. Ce modèle est une formulation généralisée, qui surmonte certaines limitations du modèle MZA. En effet, dans le modèle MZA, les fonctions $I_1 = \ln(C_1 + C_2 \varepsilon^n)$ et $S_1 = -(C_3 + C_4 \varepsilon)$ (voir Figure 2.5 pour les illustrations) sont décrites comme des fonctions polynomiales. Cependant, il existe des cas où l'utilisation de ce modèle avec l'hypothèse de droite affine réduit la performance du modèle [Yu *et al.*, 2020, Gao *et al.*, 2020, Ganapathy *et al.*, 2015, Gurusamy et Rao, 2017, Samantaray *et al.*, 2009a, He *et al.*, 2014, Zhan *et al.*, 2014, Gupta *et al.*, 2013, Yu *et al.*, 2020, Liu *et al.*, 2020]. Dans le modèle MZA ces fonctions ne peuvent pas être correctement décrites par des fonctions affines, comme montré sur la Figure 2.5 (c). Par conséquent, l'utilisation de ce modèle avec l'hypothèse d'une relation linéaire réduit les performances du modèle.

Pour surmonter ces limites, il est nécessaire de trouver une autre relation qui permette de généraliser ces fonctions fondamentales. Cette approche permet alors au modèle de s'adapter aux types d'essais, de le rendre indépendant du matériau, car elle ne repose pas sur une forme fixe.

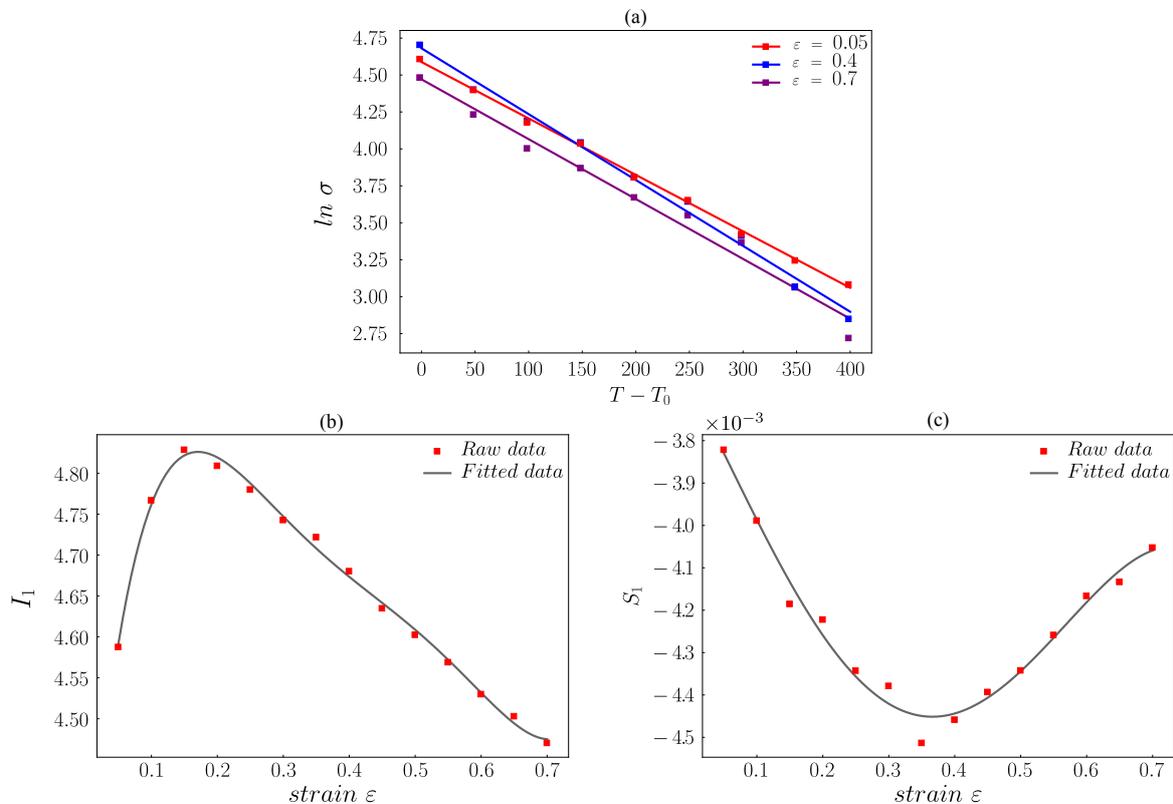


FIGURE 2.5 – Évolution de I_1 et S_1 en fonction de la déformation [Tize Mha *et al.*, 2022].

2.3.1 Formulation du modèle PTM

Comme évoqué en introduction, le modèle PTM est une généralisation du modèle MZA. Sa formulation a été décrite dans Tize Mha et *al.* [Tize Mha et al., 2022]. Les fonctions h_1 , S_1 et S_2 du modèle MZA sont remplacées par des fonctions polynomiales de degrés inconnus, à déterminer en fonction des données expérimentales. La dépendance de chaque coefficient de ces fonctions polynomiales est évaluée pour établir une complète interdépendance entre les trois grandeurs physiques à savoir la déformation ε , la vitesse de déformation $\dot{\varepsilon}$ et la température T . Dans le modèle MZA, ces coefficients sont considérés comme de simples constantes, alors que dans le modèle PTM, ce sont des polynômes. Ainsi, la forme du modèle PTM est donc donnée par l'équation suivante :

$$\sigma(\varepsilon, \dot{\varepsilon}, T) = \left(\sum_{i=0}^q A_i \varepsilon^i \right) \exp \left[\left(\sum_{j=0}^r B_j \varepsilon^j \right) (T - T_0) + \left(\sum_{k=0}^s \left(\sum_{l=0}^t C_k^l \varepsilon^l \right) (T - T_0)^k \right) \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right) \right], \quad (2.44)$$

où A_i , B_j , C_k^l sont les paramètres du modèle à identifier en utilisant la procédure d'identification proposée dans la section suivante, et les quantités q , r , s et t constituent les degrés des polynômes choisis pour décrire mieux le comportement du matériau. Si les degrés des polynômes sont trop élevés l'identification pourrait être mauvaise ; par conséquent il faut bien choisir les degrés des polynômes de telle sorte que l'aspect physique soit mis en avant concernant la précision du modèle.

2.3.2 Procédure d'identification

La détermination des paramètres du modèle PTM est effectuée en utilisant la bibliothèque *Lmfit* de Python comme présenté à l'annexe E (d'autres techniques peuvent être utilisées et la méthode peut être adaptée selon les besoins) suivant les étapes ci-après :

1. **Étape 1** : Déterminations des fonctions $h_1(\varepsilon)$ et $S_1(\varepsilon)$.

Lorsque la vitesse de déformation courante est égale à la vitesse de déformation de référence, *i.e.* quand $\dot{\varepsilon} = \dot{\varepsilon}_0$, l'équation (2.44) est écrite comme :

$$\sigma(\varepsilon, T) = \left(\sum_{i=0}^q A_i \varepsilon^i \right) \exp \left[\left(\sum_{j=0}^r B_j \varepsilon^j \right) (T - T_0) \right]. \quad (2.45)$$

En prenant le logarithme de chaque membre de cette équation on obtient :

$$\ln \sigma(\varepsilon, T) = h_1(\varepsilon) + S_1(\varepsilon) (T - T_0), \quad (2.46)$$

où :

$$h_1(\varepsilon) = \ln \left(\sum_{i=0}^q A_i \varepsilon^i \right) \quad \text{et} \quad S_1(\varepsilon) = \sum_{j=0}^r B_j \varepsilon^j \quad (2.47)$$

En traçant la fonction $\ln \sigma(\varepsilon, T)$ pour une valeur donnée de ε , on déduit $h_1(\varepsilon)$ et $S_1(\varepsilon)$ à partir de l'ordonnée à l'origine et de la pente respectivement. Ces fonctions sont évaluées pour toutes les valeurs de déformation ε disponibles dans la base de données expérimentale. Les tracés des fonctions $h_1(\varepsilon)$ et $S_1(\varepsilon)$ permettent d'obtenir q paramètres A_i et r paramètres B_j .

2. **Étape 2** : Détermination de la fonction $S_2(\varepsilon, T)$ comme fonction dépendante de la variable T . Une fois que l'étape 1 est effectuée, il est possible de prendre le logarithme de toute l'équation (2.44) pour obtenir la relation suivante :

$$\ln \sigma(\varepsilon, \dot{\varepsilon}, T) - h_1(\varepsilon) - S_1(\varepsilon) (T - T_0) = S_2(\varepsilon, T) \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right), \quad (2.48)$$

où :

$$S_2(\varepsilon, T) = \sum_{k=0}^s \left(\sum_{l=0}^t C_k^l \varepsilon^l \right) (T - T_0)^k. \quad (2.49)$$

Le tracé de la fonction $\ln \sigma - I_1(\varepsilon) - S_1(\varepsilon)(T - T_0)$ en fonction de $\ln(\dot{\varepsilon}/\dot{\varepsilon}_0)$ permet de déduire $S_2(\varepsilon, T)$ comme pente pour une certaine valeur de déformation ε . L'évolution de S_2 en fonction de $(T - T_0)$ permet de déterminer la même fonction pour toutes les valeurs de déformation qui sont décrites par le degré k du polynôme dans l'équation (2.44). Le travail ne s'arrête pas à cette étape ; il est nécessaire d'évaluer la dépendance de la déformation ε de chaque coefficient de $S_2(\varepsilon, T)$. Ceci n'a jamais été pris en compte dans les travaux précédents [Samantaray *et al.*, 2009a, Samantaray *et al.*, 2011c, Samantaray *et al.*, 2011b, Gao *et al.*, 2020, Gupta *et al.*, 2013, Zhan *et al.*, 2014, Gurusamy et Rao, 2017, Yu *et al.*, 2019, SHEN -hui *et al.*, 2020, Trimble *et al.*, 2020].

3. **Étape 3** : Détermination des coefficients de la fonction $S_2(\varepsilon, T)$ comme des fonctions polynomiales dépendantes de la déformation ε .

Cette étape est une conséquence directe de la précédente, qui consiste à déterminer chaque coefficient de la fonction $S_2(\varepsilon, T)$ sous forme de fonctions polynomiales. Une fois que l'on connaît l'ordre du polynôme qui décrit la fonction $S_2(\varepsilon, T)$ comme décrit dans l'étape précédente, pour chaque valeur de contrainte, les coefficients correspondants de $S_2(\varepsilon, T)$ doivent être enregistrés dans des listes dynamiques. Pour une valeur de contrainte donnée, il faut parcourir toutes les températures et en déduire les valeurs de chaque coefficient de la fonction $S_2(\varepsilon, T)$ pour cette contrainte qui seront enregistrées dans ces listes. On utilise un programme qui parcourt toutes les valeurs des déformations et ajoute à chaque liste de coefficients du polynôme $S_2(\varepsilon, T)$ la valeur déduite de la déformation correspondante. Une fois que toutes les données ont été explorées, on peut évaluer les paramètres du modèle PTM. La procédure d'identification de ce modèle est décrite dans Tize Mha *et al.* [Tize Mha *et al.*, 2022].

2.4 Identification du comportement du 3Cr2Mo

L'identification d'une loi de comportement implique la détermination des paramètres d'une formulation mathématique de cette loi en se basant sur les données expérimentales provenant d'essais effectués sous des conditions de chargement spécifiques. Dans cette section, nous allons procéder à l'identification de certains modèles utilisés dans le cadre de notre étude, en nous appuyant sur les données expérimentales présentées au paragraphe 1.4.3. Le choix de ces modèles est partiellement justifié par leur large disponibilité dans des logiciels de simulation par éléments finis commerciaux tels qu'Abaqus [Abaqus, 1989], Forge, etc.

2.4.1 Identification du modèle de Johnson–Cook

La procédure utilisée ici pour déterminer les paramètres du modèle de Johnson–Cook est en accord avec celle proposée par [Ganapathy *et al.*, 2018, Zeng *et al.*, 2022]. Cette méthode permet d'obtenir les paramètres de façon séquentielle dans l'ordre A , B , n , C et m .

Lorsque la température et la vitesse de déformation sont égales aux valeurs de référence T_0 et $\dot{\varepsilon}_0$ le modèle de Johnson–Cook donné par l'équation (2.8) se réduit à :

$$\sigma = A + B\varepsilon^n. \quad (2.50)$$

Compte tenu des données des essais Gleeble reportées dans la section 1.4.3, la valeur de A se déduit donc de la limite élastique du matériau qui est $A = 13.51$ MPa. En substituant cette valeur dans cette équation et en traçant la fonction $\ln(\sigma - A)$ en fonction de $\ln(\dot{\epsilon})$, les valeurs $n = 0.0746$ et $B = 21.81$ sont obtenues comme pente et ordonnée à l'origine respectivement. Une fois ces trois premiers paramètres obtenus, on peut réduire l'équation (2.8) en considérant cette fois-ci la température T égale à celle de référence T_0 mais sans condition sur la vitesse de déformation. Il vient alors :

$$\sigma = (A + B\epsilon^n) \left[1 + C \ln \left(\frac{\dot{\epsilon}}{\dot{\epsilon}_0} \right) \right]. \quad (2.51)$$

A partir de cette équation le paramètre C peut être calculé à partir de la pente de $\sigma/(A + B\epsilon^n)$ en fonction de $\ln(\dot{\epsilon}/\dot{\epsilon}_0)$. Ici $C = 0.34$.

De la même manière en considérant cette fois-ci la vitesse de déformation comme égale à la valeur de référence et sans condition sur la température, on peut calculer le paramètre m lorsque la vitesse de déformation est égale à la valeur de référence selon l'équation suivante :

$$\sigma = (A + B\epsilon^n) \left[1 - \left(\frac{T - T_0}{T_f - T_0} \right)^m \right]. \quad (2.52)$$

En utilisant les données expérimentales des contraintes à une valeur particulière de déformation ϵ et pour toutes les valeurs de la température T , on peut tracer la courbe $\ln[1 - \sigma/(A + B\epsilon^n)]$ en fonction de $\ln[T^*]$ avec $T^* = \left(\frac{T - T_0}{T_f - T_0} \right)^m$. Le paramètre $m = 0.71$ est alors obtenu comme étant la valeur de la pente à partir de cette courbe. Tous les paramètres du modèle de Johnson–Cook sont résumés dans le Tableau 2.3.

TABLE 2.3 – Paramètres du modèle de Johnson–Cook (JC) pour le 3Cr2Mo.

A(MPa)	B(MPa)	n	C	m
13.5143	21.816	0.0746	0.3404	0.7057

Il est également possible d'obtenir tous les paramètres du modèle de Johnson–Cook en utilisant une méthode d'optimisation multivariées comme la méthode *lmfit* [Newville *et al.*, 2016], mais cette approche n'a pas été retenue dans cette étude afin de rester dans le cadre d'identifications selon des méthodes classiques.

Les prédictions du modèles de Johnson–Cook et les valeurs expérimentales sont comparées sur la Figure 2.6. Il ressort de l'étude de cette figure que le modèle de Johnson–Cook n'est pas en mesure de décrire l'évolution de la contrainte d'écoulement moyenne pour tous les niveaux de déformation et de vitesse de déformation. Les contraintes d'écoulement expérimentales montrent une croissance puis une diminution avec la déformation, en particulier pour les faibles vitesses de déformation, alors que le modèle de Johnson–Cook, de par sa formulation, ne permet qu'une évolution croissante de la contrainte d'écoulement σ en fonction de la déformation, indépendamment de la valeur de la vitesse de déformation.

L'écart entre les valeurs prédites et les données expérimentales est important pour les faibles déformations et parfois acceptable pour les fortes déformations. Comme prévu, la formulation mathématique de la loi d'écoulement de Johnson–Cook est incapable de représenter la chute de contrainte à faibles vitesses de déformation, le modèle de Johnson–Cook n'étant que croissant que de façon monotone

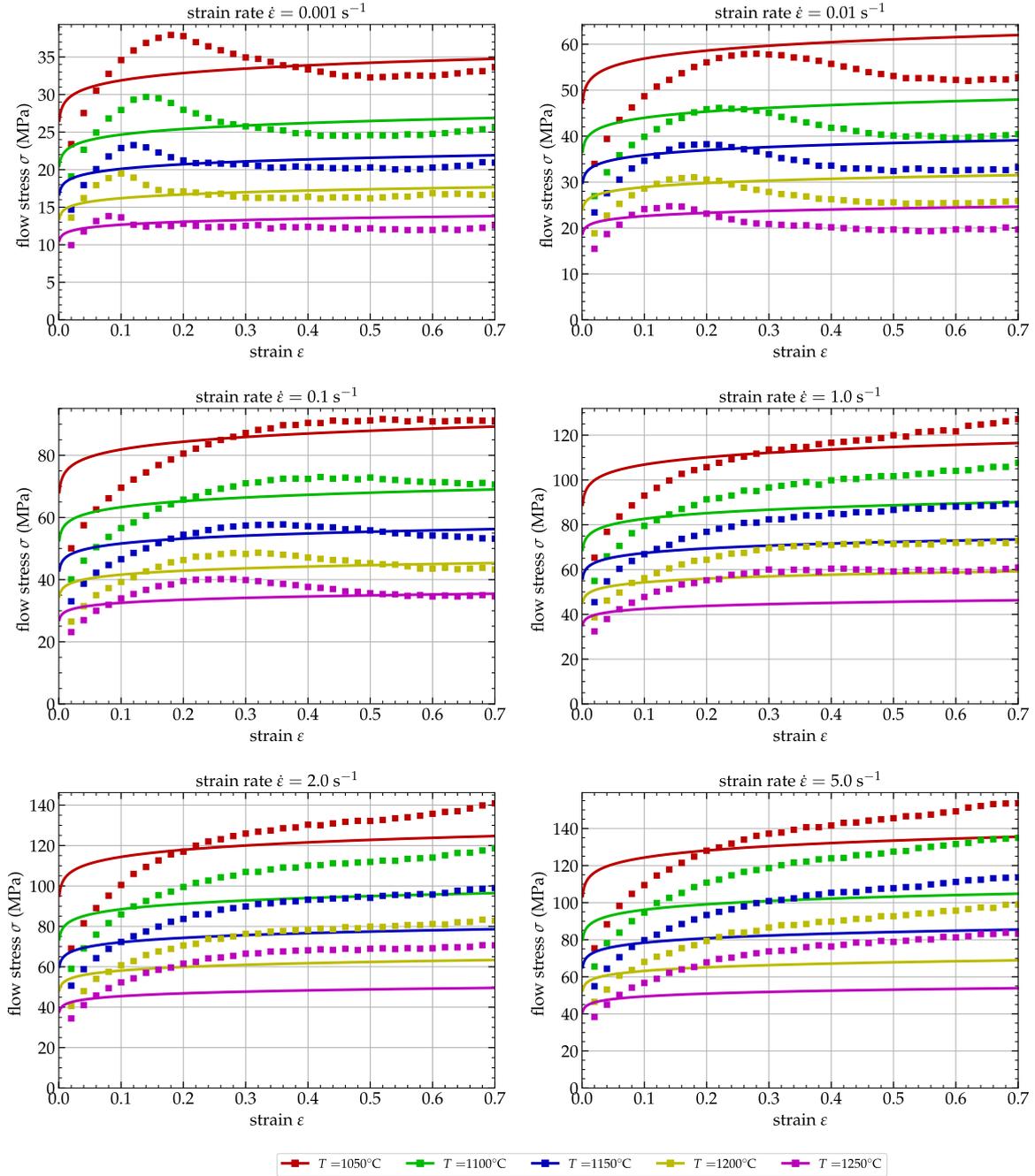


FIGURE 2.6 – Comparaison entre les contraintes d'écoulement expérimentales (points) et celles prédites (traits) par le modèle Johnson–Cook.

avec la déformation. Comme la plupart des paramètres (A , B et n) sont identifiés, par la méthode de [Ganapathy *et al.*, 2018, Zeng *et al.*, 2022] à la vitesse de déformation de référence, il en résulte une très mauvaise adéquation du modèle aux données expérimentales.

La précision et la capacité de prédiction des lois de comportement sont généralement évalués par certains coefficients tels que la valeur absolue de l'erreur relative (Mean Absolute Relative Error en anglais) (E_{MAR}) définie par l'équation (2.53) :

$$E_{MAR}(\%) = \frac{1}{N} \sum_{i=1}^N \left| \frac{\sigma_i^p - \sigma_i^e}{\sigma_i^e} \right| \times 100, \quad (2.53)$$

et la racine de la moyenne de l'erreur relative (Root Mean Square Error en anglais) (E_{RMS}) donnée par l'équation (2.54) :

$$E_{\text{RMS}} = \sqrt{\frac{1}{N} \sum_{i=1}^N (\sigma_i^p - \sigma_i^e)^2}, \quad (2.54)$$

où σ_i^e est la valeur expérimentale, σ_i^p la valeur prédite par un modèle donné et N est le nombre total des données utilisées pour calculer ces coefficients (21 000 dans le cas des identifications de modèles décrits dans ce chapitre).

Pour le modèle de Johnson-Cook, nous pouvons évaluer ces erreurs à $E_{\text{MAR}} = 14.05\%$ et $E_{\text{RMS}} = 12.00$ MPa. Un autre coefficient souvent utilisé pour décrire la performance d'un modèle est le coefficient de corrélation (R). Cependant, comme mentionné par Phaniraj et al. [Phaniraj et Lahiri, 2003] ce coefficient n'est pas toujours une bonne mesure pour estimer la qualité d'une loi de comportement spécialement dans le cas des fonctions hautement non linéaires car il n'indique que la corrélation du modèle par rapport aux données et non sa précision. Une haute valeur de R (proche de 1) ne signifie pas forcément une très bonne prédiction mais simplement établit une bonne corrélation entre l'expérience et la prédiction ; par conséquent dans cette étude ce coefficient est ne sera pas utilisé dans l'analyse comparative des résultats.

2.4.2 Identification du modèle modifié de Zerilli-Armstrong

L'identification des paramètres du modèle MZA est basée sur la méthode proposée par [Samantaray et al., 2009b] comme décrit ci-après. Lorsque la vitesse de déformation actuelle $\dot{\varepsilon}$ est identique à la vitesse de déformation de référence $\dot{\varepsilon}_0$, le modèle MZA décrit par l'équation (2.40) devient :

$$\sigma = (C_1 + C_2 \varepsilon^n) \exp[-(C_3 + C_4 \varepsilon)(T - T_0)]. \quad (2.55)$$

En prenant le logarithme de la relation précédente on a :

$$\ln \sigma = \ln(C_1 + C_2 \varepsilon^n) - (C_3 + C_4 \varepsilon)(T - T_0). \quad (2.56)$$

En traçant la droite $\ln(\sigma)$ en fonction de $(T - T_0)$ on peut en déduire respectivement l'ordonnée à l'origine et la pente de cette droite comme : $l_1 = \ln(C_1 + C_2 \varepsilon^n)$ et $S_1 = -(C_3 + C_4 \varepsilon)$. En réorganisant la première expression, on peut écrire :

$$\ln(\exp l_1 - C_1) = \ln C_2 + n \ln \varepsilon. \quad (2.57)$$

En traçant la fonction $\ln(\exp l_1 - C_1)$ en fonction de $\ln \varepsilon$ on peut déduire n et C_2 . De même à partir de $S_1 = -(C_3 + C_4 \varepsilon)$, et en traçant S_1 en fonction de ε on peut déduire C_3 et C_4 respectivement à partir de l'ordonnée à l'origine et de la pente. Une fois les quatre premiers termes du modèle évalués, on peut prendre maintenant le logarithme de tout le modèle décrit par l'équation (2.40) :

$$\ln \sigma = \ln(C_1 + C_2 \varepsilon^n) - (C_3 + C_4 \varepsilon)(T - T_0) + (C_5 + C_6(T - T_0)) \ln \left[\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right]. \quad (2.58)$$

Le graphe de $\ln(\sigma)$ en fonction de $\ln \left[\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right]$ permet de déduire C_5 et C_6 .

Les paramètres du modèle obtenu à partir des essais Gleeble décrits dans la section 1.4.3 sont résumés dans le Tableau 2.4.

TABLE 2.4 – Paramètres du modèle Modifié de Zerilli–Armstrong (MZA) pour le 3Cr2Mo.

C_1	C_2	C_3	C_4	C_5	C_6	n
13.5143	21.2591	4.7902×10^{-3}	1.4895×10^{-4}	0.1389	1.495×10^{-4}	0.0621

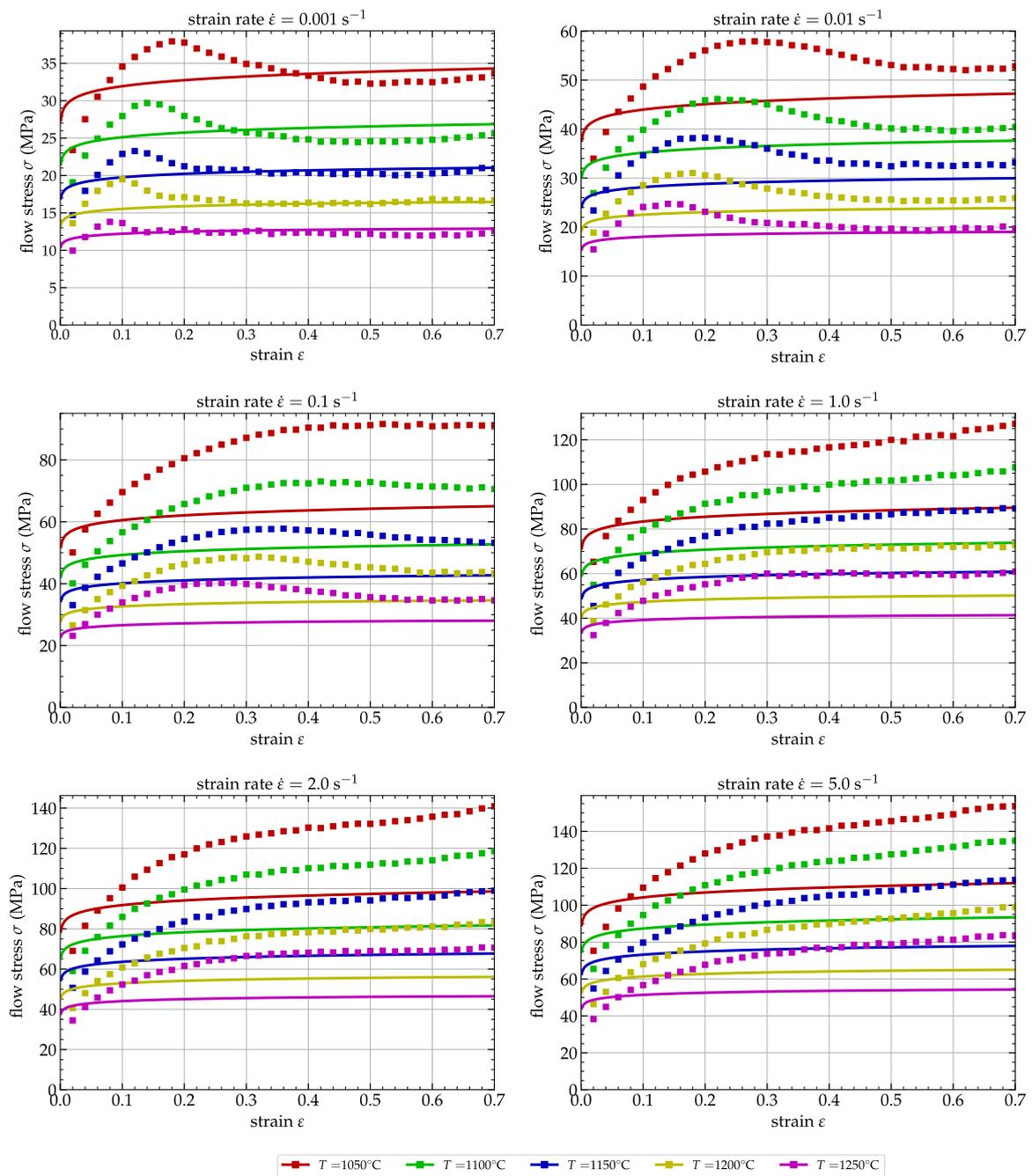


FIGURE 2.7 – Comparaison entre les contraintes d'écoulement expérimentales (points) et celles prédites (traits) par le modèle de Zerilli–Armstrong Modifié (MZA).

Sur la Figure 2.7, on peut constater que, tout comme le modèle de Johnson–Cook, le modèle MZA

ne parvient pas non plus à prédire de manière précise les valeurs expérimentales des contraintes d'écoulement.

On peut remarquer que l'écart entre les prédictions du modèle et les données expérimentales est significativement élevé, ce qui indique que le modèle a des difficultés à prédire l'adoucissement du matériau, malgré sa prise en compte du couplage entre la vitesse de déformation et la température. Pour ce modèle, $E_{MAR} = 21.20 \%$ et $E_{RMS} = 19.57 \text{ MPa}$, montrant ainsi une mauvaise performance du modèle comme rapporté par Tize Mha *et al.* [Tize Mha *et al.*, 2022, Tize Mha *et al.*, 2023]. De plus, il est également remarquable que, même pour des vitesses de déformation élevées où l'adoucissement du matériau n'est pas prononcé, le modèle MZA ne parvient pas à fournir des prédictions précises. Cette observation peut s'expliquer par le fait que le modèle a été initialement développé pour des essais en quasi-statique.

2.4.3 Identification du modèle de Hensel–Spittel

Comme abordé dans la section de présentation des modèles, la précision du modèle de Hensel–Spittel dépend du choix du nombre de paramètres du modèle. Dans le cadre de notre étude, compte tenu des données des essais Gleeble, le modèle donne de meilleurs résultats en considérant les 7 premiers paramètres m_i . À partir des données expérimentales du paragraphe 1.4.3 une technique basée sur l'optimiseur *Lmfit* [Newville *et al.*, 2016] permet d'obtenir les paramètres de ce modèle résumés dans le Tableau 2.5. La comparaison entre les contraintes prédites par le modèle de Hensel–Spittel et celles expérimentalement mesurées sont données sur la Figure 2.8.

TABLE 2.5 – Paramètres du modèle de Hensel–Spittel (HS) pour le 3Cr2Mo.

A	m_1	m_2	m_3	m_4
5.954×10^3	-3.3576×10^{-3}	0.2641	-0.0868	2.2688×10^{-4}
	m_5	m_6	m_7	m_8
	-4.2163×10^{-4}	-0.0561	2.264×10^{-4}	0

Pour ce modèle, $E_{MAR} = 7.75 \%$ et $E_{RMS} = 3.80 \text{ MPa}$. Comme pour les modèles précédents, le modèle de Hensel–Spittel ne convient pas à cet alliage, car ses prédictions ne sont pas précises pour les vitesses de déformation inférieures à $\dot{\epsilon} = 1 \text{ s}^{-1}$. Ce modèle n'est pas approprié pour caractériser ce type de matériau dans cette gamme de vitesses de déformations.

2.4.4 Identification du modèle d'Arrhenius

Comme évoqué plus haut dans le document, le modèle d'Arrhenius est l'un des modèles largement utilisés dans la simulation des procédés industriels de par sa précision. Pour la détermination des paramètres de ce modèle défini à partir des équations (2.11) et (2.42), on peut établir les relations suivantes :

$$\sigma = \frac{1}{\beta} \ln \dot{\epsilon} - \frac{1}{\beta} \ln A_2 + \frac{1}{\beta} \frac{Q}{RT}, \quad (2.59)$$

$$\ln \sigma = \frac{1}{n_1} \ln \dot{\epsilon} - \frac{1}{n_1} \ln A_1 + \frac{1}{n_1} \frac{Q}{RT}, \quad (2.60)$$

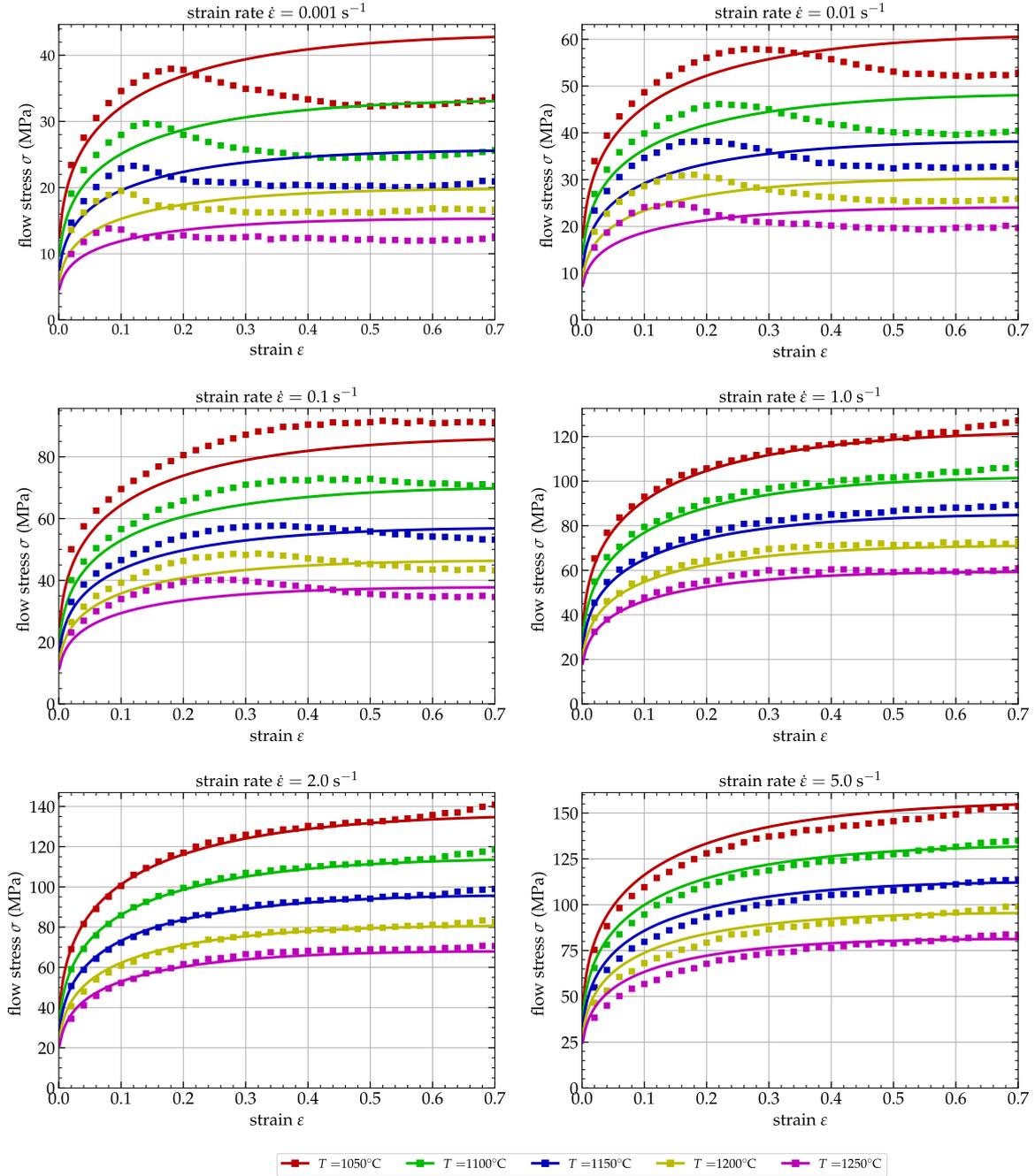


FIGURE 2.8 – Comparaison entre les contraintes d'écoulement expérimentales (points) et celles prédites (traits fins) par le modèle Hensel–Spittel (HS).

et

$$\ln \sinh(\alpha\sigma) = \frac{1}{n_2} \ln \dot{\epsilon} - \frac{1}{n_2} \ln A_3 + \frac{1}{n_2} \frac{Q}{RT}. \quad (2.61)$$

A partir de ces équations, et en fonction des données des essais reportées au paragraphe 1.4.3 page 22, on peut obtenir $n_1 = 4.91$, $A_1 = 4.45$, $\beta = 0.15$ et $A_2 = 68.77$. Puis, à partir des relations σ en fonction de $\ln \dot{\epsilon}$ (voir Figure 2.9 (a)) et $\ln \sigma$ en fonction de $\ln \dot{\epsilon}$ (voir Figure 2.9 (b)) respectivement et $\beta = \alpha n_1 = 0.022$ s'en déduit en conséquence. Quant aux paramètres A_3 et n_2 , ils sont obtenus en utilisant la relation $\ln \sinh(\alpha\sigma)$ en fonction de $\ln \dot{\epsilon}$ (Figure 2.9 (c)). En utilisant une dérivation par

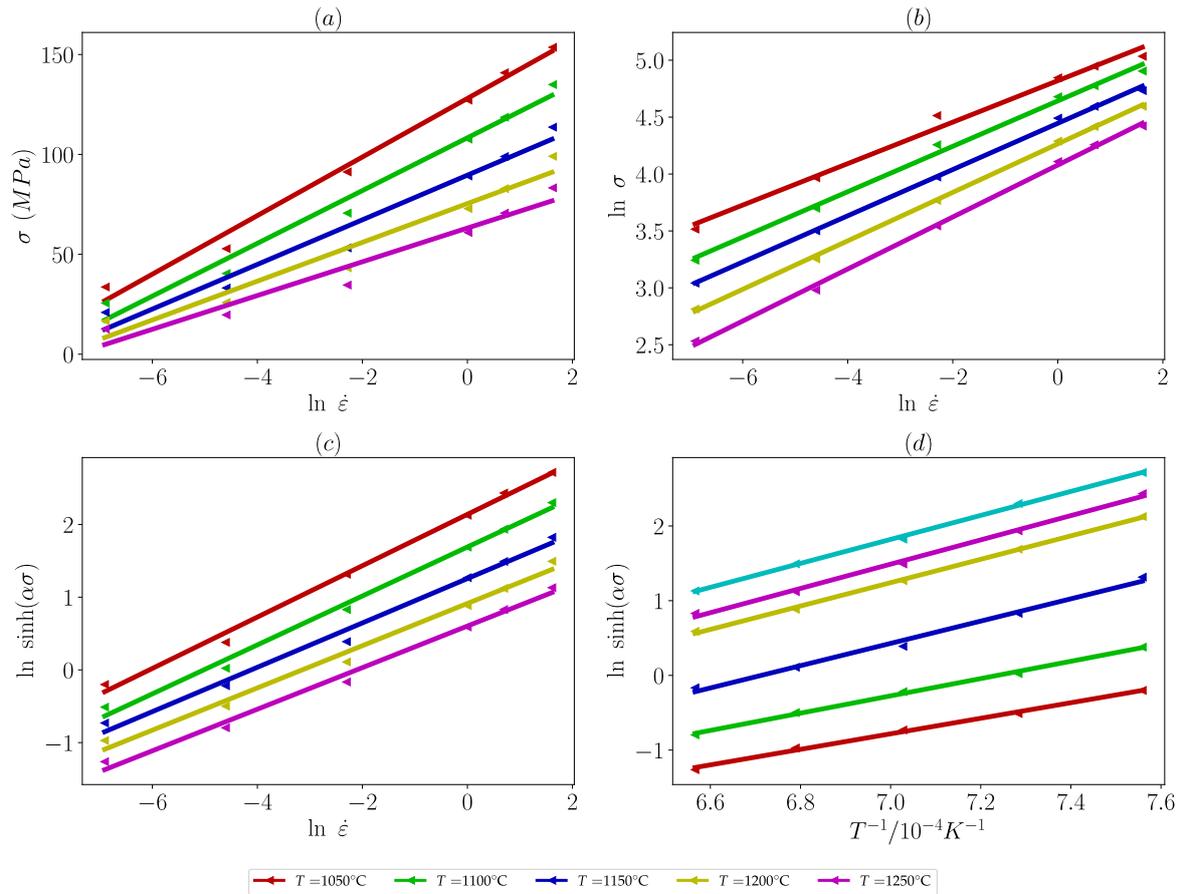


FIGURE 2.9 – Fonctions utiles pour l'identification des paramètres du modèle d'Arrhenius

chaîne de l'équation (2.61) il vient la relation suivante :

$$Q = R \cdot \left. \frac{\partial \dot{\epsilon}}{\partial \ln \sinh(\alpha \sigma)} \right|_T \left. \frac{\partial \ln \sinh(\alpha \sigma)}{\partial (1/T)} \right|_{\dot{\epsilon}}. \quad (2.62)$$

A partir de cette équation, le paramètre $Q = 437\,385$ se déduit en utilisant les tracés des relations $\ln \sinh(\alpha \sigma)$ en fonction de $\ln \dot{\epsilon}$ et $\ln \sinh(\alpha \sigma)$ en fonction de $1\,000/T$ (voir Figure 2.9 (d)).

Il faut noter que ces identifications sont faites à partir d'une valeur quelconque de la déformation et que la prise en compte de toutes déformations permet d'évaluer la dépendance à la déformation. Un programme Python permet d'effectuer ces calculs en parcourant toutes les valeurs de déformations disponibles. Une fonction du type polynôme de degré 9 permet de d'approximer correctement ces paramètres et en conséquence donne une bonne précision du modèle. Les équations suivantes donnent les expressions de chaque coefficient en fonction de la déformation :

$$\begin{cases} A(\epsilon) = \exp [\ln A_0 + \ln A_1 \epsilon + \ln A_2 \epsilon^2 + \ln A_3 \epsilon^3 + \dots + \ln A_m \epsilon^m] \\ Q(\epsilon) = Q_0 + Q_1 \epsilon + Q_2 \epsilon^2 + Q_3 \epsilon^3 + \dots + Q_m \epsilon^m \\ \alpha(\epsilon) = \alpha_0 + \alpha_1 \epsilon + \alpha_2 \epsilon^2 + \alpha_3 \epsilon^3 + \dots + \alpha_m \epsilon^m \\ n(\epsilon) = n_0 + n_1 \epsilon + n_2 \epsilon^2 + n_3 \epsilon^3 + \dots + n_m \epsilon^m \end{cases} \quad (2.63)$$

Le modèle d'Arrhenius peut donc finalement s'écrire comme suit en précisant que le n dans la nouvelle

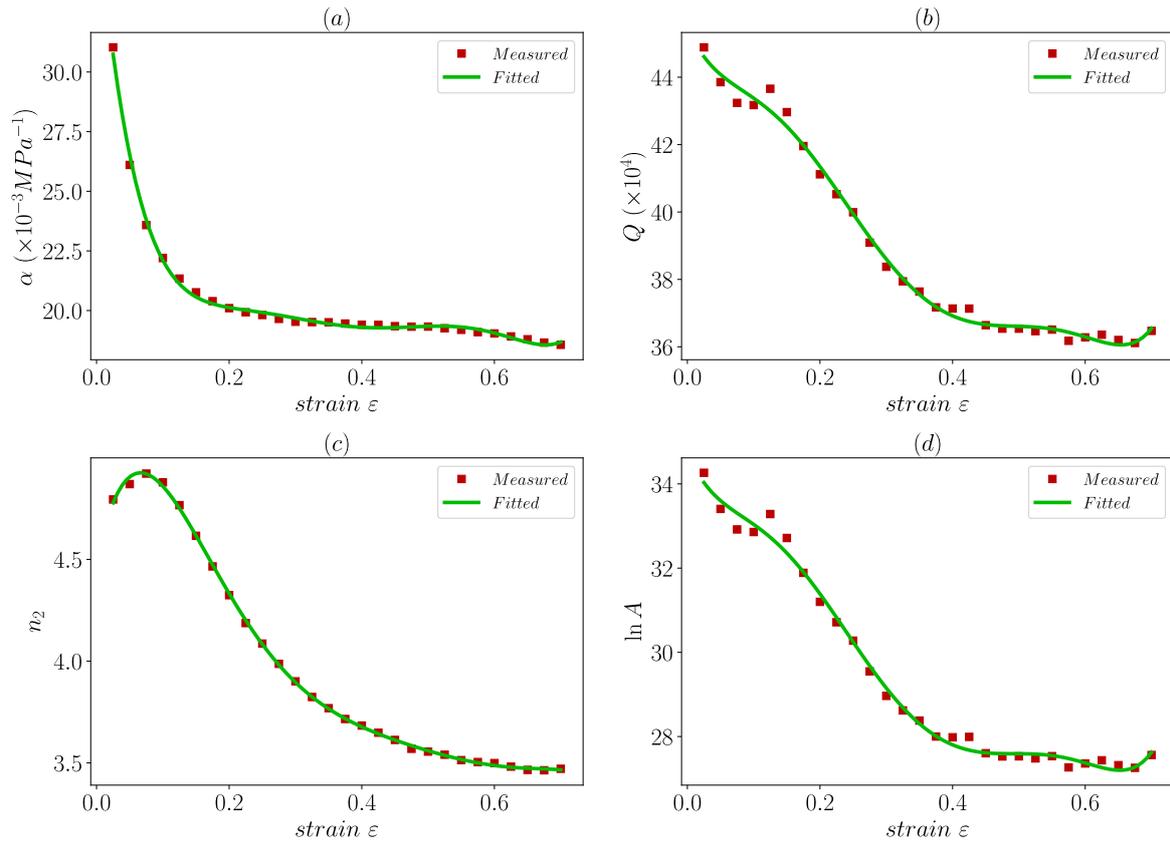


FIGURE 2.10 – Paramètres du modèle d'Arrhenius en fonction de la déformation.

TABLE 2.6 – Paramètres du modèle d'Arrhenius (AR) pour le 3Cr2Mo.

α_i	$Q_i (\times 10^{-6})$	$\ln A_i$	n_i
$\alpha_0 = 0.0407$	$Q_0 = 0.467$	$\ln A_0 = 35.8092$	$n_0 = 4.8217$
$\alpha_1 = -0.5167$	$Q_1 = -0.6517$	$\ln A_1 = -58.822$	$n_1 = 3.2814$
$\alpha_2 = 6.3912$	$Q_2 = 7.6084$	$\ln A_2 = 740.3303$	$n_2 = 71.5963$
$\alpha_3 = -47.3364$	$Q_3 = -48.016$	$\ln A_3 = -5.0493 \times 10^3$	$n_3 = -1.9562 \times 10^3$
$\alpha_4 = 220.0014$	$Q_4 = 66.795$	$\ln A_4 = 1.1305 \times 10^4$	$n_4 = 1.4461 \times 10^4$
$\alpha_5 = -654.4553$	$Q_5 = 468.8898$	$\ln A_5 = 2.022 \times 10^4$	$n_5 = -5.431 \times 10^4$
$\alpha_6 = 1.2421 \times 10^3$	$Q_6 = -2.3032 \times 10^3$	$\ln A_6 = -1.5387 \times 10^5$	$n_6 = 1.1761 \times 10^5$
$\alpha_7 = -1.4523 \times 10^3$	$Q_7 = 4.3707 \times 10^3$	$\ln A_7 = 3.1798 \times 10^5$	$n_7 = -1.4882 \times 10^5$
$\alpha_8 = 952.0619$	$Q_8 = -3.9394 \times 10^3$	$\ln A_8 = -2.9725 \times 10^5$	$n_8 = 1.0239 \times 10^5$
$\alpha_9 = -267.4994$	$Q_9 = 1.397 \times 10^3$	$\ln A_9 = 1.0759 \times 10^5$	$n_9 = -2.9621 \times 10^4$

équation est l'équivalent de n_2 dans l'équation initiale :

$$\sigma = \frac{1}{\alpha(\varepsilon)} \ln \left\{ \left(\frac{\dot{\varepsilon} \exp\left(\frac{Q(\varepsilon)}{RT}\right)}{A(\varepsilon)} \right)^{\frac{1}{n(\varepsilon)}} + \left[1 + \left(\frac{\dot{\varepsilon} \exp\left(\frac{Q(\varepsilon)}{RT}\right)}{A(\varepsilon)} \right)^{\frac{2}{n(\varepsilon)}} \right]^{\frac{1}{2}} \right\}. \quad (2.64)$$

La Figure 2.11 illustre la comparaison entre les prédictions du modèle d'Arrhenius et les données

expérimentales. Ce modèle parvient à prédire assez précisément les résultats expérimentaux. Cependant, il montre des limitations lorsque la vitesse de déformation est faible. Pour ce modèle, les coefficients $E_{MAR} = 3.56 \%$ et $E_{RMS} = 2.18 \text{ MPa}$.

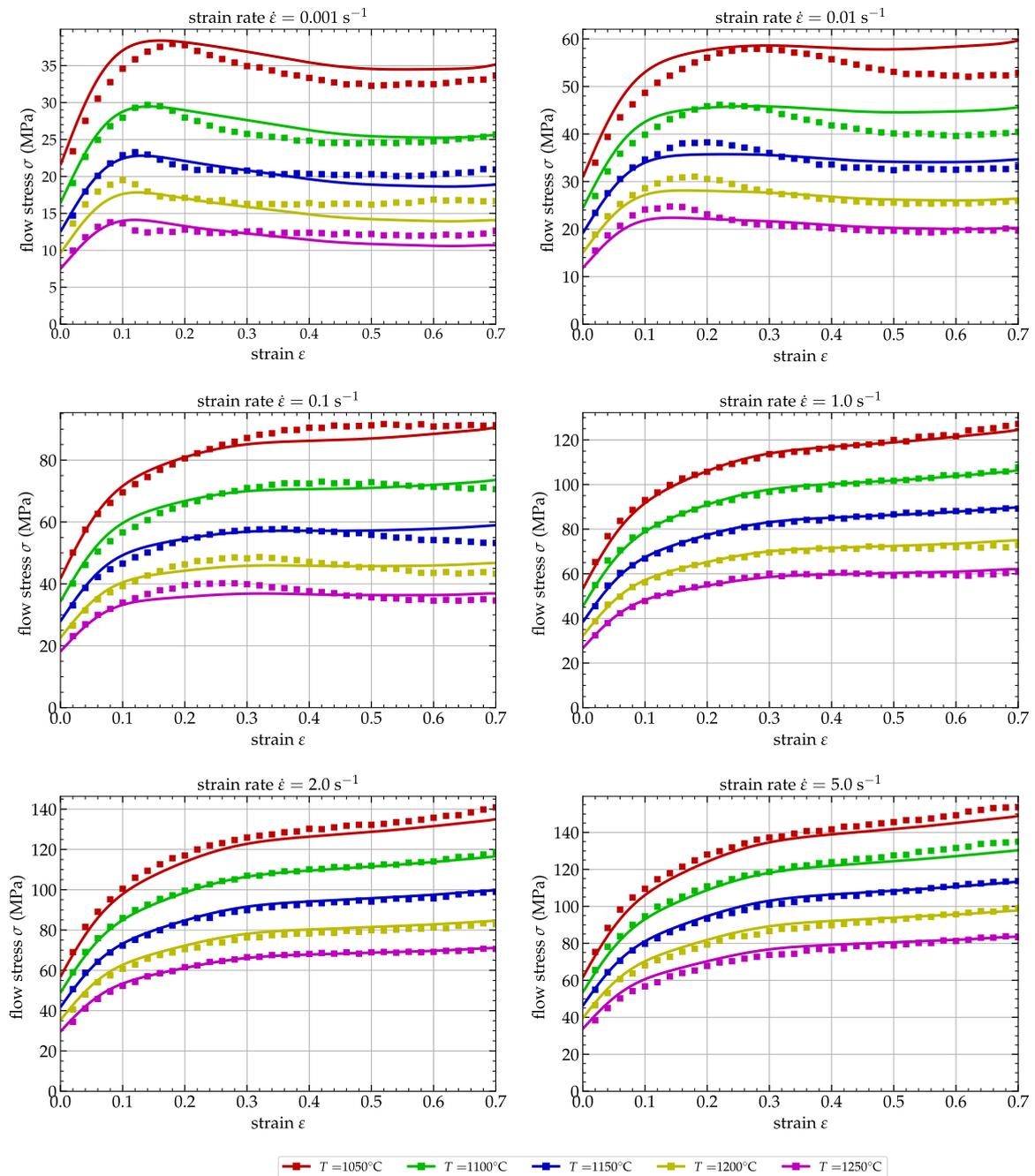


FIGURE 2.11 – Comparaison entre les contraintes d'écoulement expérimentales (points) et celles prédites (traits fins) par le modèle Arrhenius (AR).

2.4.5 Identification du modèle PTM

Comme pour les autres lois d'écoulement, les données des essais Gleeble du matériau étudié reportées dans le paragraphe 1.4.3 sont utilisées pour identifier les paramètres du modèle PTM. La même procédure a été appliquée sur un autre matériau dans Tize Mha et al. [Tize Mha et al., 2022].

Pour l'identification nous avons 840 données (en contraintes) avec 28 déformations plastiques (ε), 6 vitesses de déformations ($\dot{\varepsilon}$) et 5 températures (T). La première étape de cette identification concerne la détermination des fonctions $I_1(\varepsilon)$ et $S_1(\varepsilon)$ qui apparaissent dans l'équation (2.46) en considérant la vitesse de déformation actuelle égale la vitesse de déformation de référence ($\dot{\varepsilon} = \dot{\varepsilon}_0 = 10^{-3} \text{ s}^{-1}$ dans le notre cas). Ainsi, nous traçons en premier 28 courbes $\sigma(T)$ (i.e. une courbe pour chaque déformation ε) et extrayons l'ordonnée à l'origine et la pente de ces 28 courbes comme $I_1(\varepsilon)$ et $S_1(\varepsilon)$ respectivement. Ce processus est illustré sur la Figure 2.5 (a) tirée de [Tize Mha et al., 2022] pour le matériau 42CrMo4 avec trois valeurs de la déformation ε . A l'issue de ce premier calcul, on a 28 valeurs de I_1 de même que S_1 pour lesquelles il faut trouver le polynôme qui décrit bien la dépendance de la déformation (Voir Figures 2.5 (b) et Figures 2.5 (c) pour un exemple obtenu à partir du matériau 42CrMo4 [Tize Mha et al., 2022]). A partir de ces deux fonctions on peut donc en déduire les paramètres A_i et B_j .

En utilisant la partie gauche de l'équation (2.47), et après avoir tracé $\exp[I_1(\varepsilon)]$ pour une valeur appropriée du paramètre q (ici $q = 5$) définissant l'ordre de la fonction polynomiale utilisée pour approximer $I_1(\varepsilon)$, les cinq paramètres A_i sont identifiés. De la même façon, la partie droite de l'équation (2.47) permet de calculer les paramètres B_j de la fonction $S_1(\varepsilon)$ avec un ordre adéquat r de la fonction polynomiale (ici aussi $r = 5$).

Une fois les paramètres de ces deux fonctions ($I_1(\varepsilon)$ et $S_1(\varepsilon)$) identifiés, la dernière étape consiste à identifier les paramètres de la fonction $S_2(\varepsilon, T)$. A partir de l'équation (2.48), on trace la relation $\ln(\sigma) - I_1(\varepsilon) - S_1(\varepsilon) (T - T_0)$ en fonction de $\ln(\dot{\varepsilon}/\dot{\varepsilon}_0)$ pour toutes les déformations plastiques ε et toutes les températures T et on rassemble les $140 = 28 \times 5$ pentes définissant l'ensemble des paramètres S_2 dépendant respectivement de ε et de T . Ces 140 données sont maintenant regroupées en fonction de leurs valeurs de déformation et sont utilisées pour identifier 28 courbes $S_2(T)$. Pour chaque déformation on recherche le polynôme qui décrit bien $S_2(T)$ qui est $s = 1$ dans notre cas. Il faut noter que le principe consiste à utiliser la même fonction pour les 28 courbes $S_2(T)$ qui est une droite dans le cas de ce matériau. Un exemple de la fonction $S_2(T)$ est illustré sur la Figure 2.12(a) tirée de [Tize Mha et al., 2022] pour le matériau 42CrMo4 avec 3 valeurs de la déformation ε .

La dernière étape consiste à évaluer la dépendance de la déformation des coefficients de $S_2(T)$ en utilisant la fonction polynomiale convenable qui est de degré 5 dans notre cas. Comme $S_2(T)$ est une droite, alors il a deux coefficients qui dépendent de la déformation ; soit $12 = 6 \times 2$ coefficients C_k^l avec $k \in \{0, 1\}$ et $l \in \{0, 5\}$. Un exemple de courbe C_k^l est illustré sur la Figure 2.12(a) tirée de [Tize Mha et al., 2022] pour le matériau 42CrMo4. Le lecteur qui souhaite avoir plus des détails sur cette méthode d'identification peut se référer à [Tize Mha et al., 2022].

Les paramètres de ce modèle pour le 3Cr2Mo sont résumés dans le Tableau 2.7. La Figure 2.13 montre la comparaison entre l'expérience et les prédictions du modèle PTM. On constate que le modèle PTM a du mal prédire les contraintes pour les vitesses de déformations $\dot{\varepsilon} = 10^{-2} \text{ s}^{-1}$ et $\dot{\varepsilon} = 10^{-1} \text{ s}^{-1}$. Cela peut s'expliquer par le fait que le degré du polynôme a été limité à 5. En augmentant le degré de ce polynôme, le modèle sera plus précis. Pour le modèle PTM, $E_{\text{MAR}} = 4.79 \%$ et $E_{\text{RMS}} = 4.59 \text{ MPa}$, ce qui représente une performance globalement inférieure à celle du modèle d'Arrhenius.

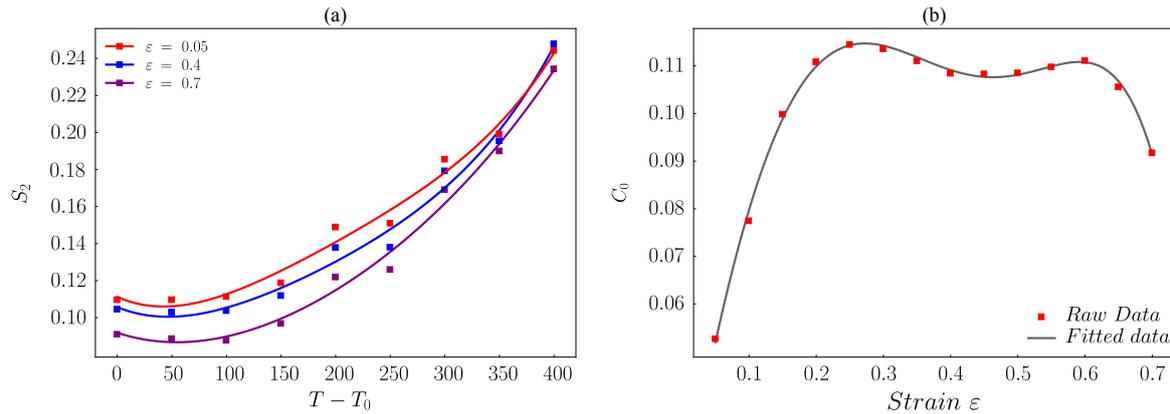


FIGURE 2.12 – Détermination des C_k^l paramètres du modèle PTM [Tize Mha et al., 2022].

TABLE 2.7 – Paramètres du modèle PTM pour le 3Cr2Mo.

A_i	B_i	C_0^i	C_1^i
$A_0 = 16.8529$	$B_0 = -3.5418 \times 10^{-3}$	$C_0^0 = 0.1608$	$C_1^0 = -1.9037 \times 10^{-5}$
$A_1 = 340.6451$	$B_1 = -0.0132$	$C_0^1 = -0.6202$	$C_1^1 = 2.67 \times 10^{-3}$
$A_2 = -1.9594 \times 10^3$	$B_2 = -4.4888 \times 10^{-3}$	$C_0^2 = 4.9516$	$C_1^2 = -3.5788 \times 10^{-3}$
$A_3 = 4.836 \times 10^3$	$B_3 = 0.2218$	$C_0^3 = -13.1694$	$C_1^3 = -0.0222$
$A_4 = -5.5176 \times 10^3$	$B_4 = -0.4988$	$C_0^4 = 15.25$	$C_1^4 = 0.0609$
$A_5 = 2.4058 \times 10^3$	$B_5 = 0.3211$	$C_0^5 = -6.587$	$C_1^5 = -0.0413$

2.5 Comparaison des modèles identifiés

A titre de comparatif, le Tableau 2.8 rassemble les valeurs E_{MAR} et E_{RMS} des cinq modèles analytiques identifiés dans la section précédente. Il ressort de ce tableau que le modèle d'Arrhenius (AR) affiche

TABLE 2.8 – Comparaison des 5 modèles de comportement identifiés.

Coefficients	JC	MZA	HS	AR	PTM
$E_{MAR}(\%)$	14.05	21.20	7.75	3.56	4.79
$E_{RMS}(\text{MPa})$	12.00	19.57	3.80	2.18	4.59

une meilleure aptitude à la prédiction par rapport aux quatre autres modèles, se distinguant par ses faibles valeurs de E_{MAR} et de E_{RMS} . Le modèle PTM présente également des valeurs relativement faibles des erreurs, mais sa précision demeure légèrement inférieure à celle du modèle d'Arrhenius. A l'opposé, les modèles de Johnson–Cook et de Zerilli–Armstrong Modifié (MZA) montrent une capacité de prédiction médiocre. Au niveau intermédiaire, le modèle de Hensel–Spittel (HS) peut être utilisé pour décrire le comportement de ce matériau en raison de ses faibles valeurs d'erreur. Dans l'ensemble, le modèle d'Arrhenius s'avère être le choix privilégié pour l'étude de ce matériau, ce qui confirme sa réputation de modèle de référence dans le contexte de la mise en forme à chaud des alliages, comme précisé dans les travaux de [Vannieuwenhuyze, 2019, Liang et al., 2022].

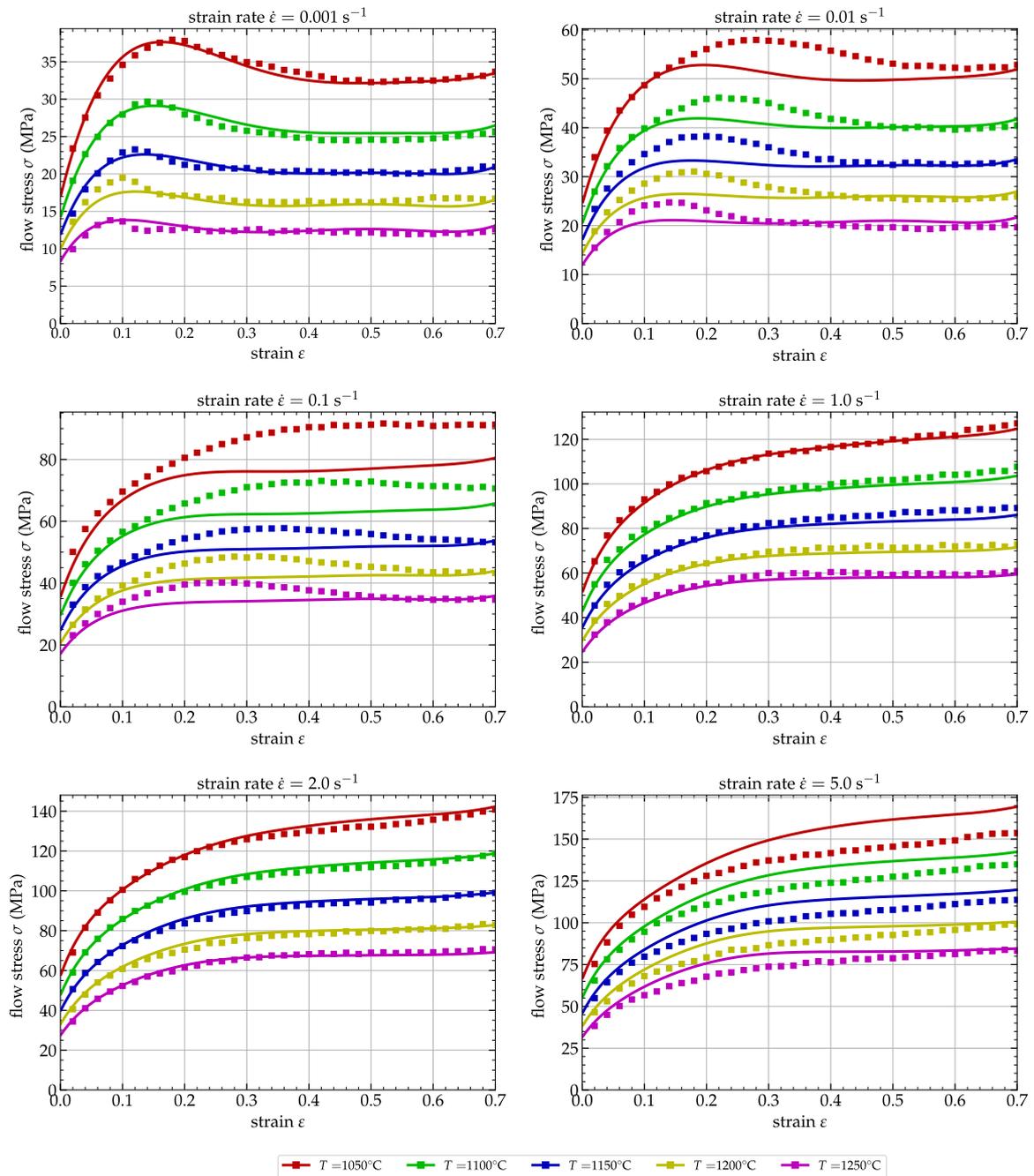


FIGURE 2.13 – Comparaison entre les contraintes d'écoulement expérimentales et celles prédites par le modèle PTM.

2.6 Conclusions du chapitre

Ce chapitre a couvert la présentation de modèles analytiques d'écoulement, dans un contexte général, et des procédures d'identification de cinq de ces modèles. Ce chapitre a également été l'objet de la proposition d'un nouveau modèle phénoménologique, le modèle PTM, qui s'est révélé avoir des performances supérieures à celles du modèle de Zerilli–Armstrong Modifié (MZA) dont il dérive. L'application pratique de ces modèles à un cas concret d'alliage de carbone (le 3Cr2Mo) a révélé que le modèle d'Arrhenius se démarque par sa capacité à fournir des prédictions précises, caractérisées par une marge d'erreur minimale. Cette étude a contribué à une meilleure compréhension des modèles de comportement, a permis d'identifier des pistes d'amélioration pour la prédiction du comportement des matériaux, et a souligné l'importance du nombre de paramètres dans la précision d'une loi de comportement.

Compte tenu de la limitation des modèles analytiques, du fait de leur formulation mathématique fixée, une solution alternative à ces modèles doit donc être trouvée avant leur identification et intégration dans un logiciel d'éléments finis. Le chapitre suivant se penchera sur la proposition d'une approche alternative, basée sur les Réseaux de Neurones Artificiels (RNA), pour la formulation de lois de comportement.

Chapitre 3

Loi d'écoulement / Réseau de Neurones Artificiels (RNA)

3.1	Les Réseaux de Neurones Artificiels	60
3.1.1	Bref historique de la méthode	60
3.1.2	Émergence des réseaux de neurones dans la technique	61
3.1.3	Perceptron simple	62
3.1.4	Le perceptron multicouches	66
3.2	Loi de comportement basée sur un RNA	70
3.2.1	Architecture du réseau de neurones	70
3.2.2	Pré et post traitement des données	70
3.3	Validation du modèle RNA sur un 42CrMo4	73
3.3.1	Génération des données d'apprentissage	73
3.3.2	Apprentissage et analyse de la performance du RNA	74
3.4	Identification de la loi d'écoulement du 3Cr2Mo	76
3.4.1	Apprentissage et prédiction du réseau	76
3.4.2	Comparaison des modèles analytiques avec le RNA	77
3.4.3	Capacité d'interpolation/extrapolation des modèles	79
3.5	Conclusions du chapitre	84

L'utilisation de l'Intelligence Artificielle (IA) à travers les Réseaux de Neurones Artificiels (RNA) dans la simulation numérique est une approche révolutionnaire pour améliorer les lois de comportement. Cette approche offre de nombreux avantages et ouvre de nouvelles perspectives dans divers domaines de la modélisation et de la simulation. En effet, contrairement aux lois de comportement classiques, les réseaux de neurones artificiels ne sont pas restreints par une forme analytique préalablement définie. Ils peuvent apprendre à partir des données expérimentales sans nécessiter une formulation mathématique précise, ce qui les rend extrêmement flexibles et adaptables à des situations complexes. De plus, ces réseaux peuvent capturer des comportements complexes et non linéaires, souvent difficiles à modéliser avec des équations mathématiques traditionnelles, en se basant sur les relations entre les données d'entrée et de sortie, sans exiger une connaissance préalable des lois sous-jacentes. La possibilité d'entraîner ces réseaux de neurones sur des données expérimentales permet de prendre en compte des variations qui peuvent survenir dans des environnements réels. De ce fait, ils peuvent offrir une précision accrue dans les prédictions en ajustant leurs paramètres pour s'adapter plus précisément aux données d'entrée, surpassant ainsi les modèles analytiques traditionnels.

Ce chapitre présente un nouveau modèle de comportement basé sur un réseau de neurones artificiels. Il commence par fournir un bref historique de cette méthode, suivi d'une brève revue de la littérature relative aux réseaux de neurones artificiels. Les étapes de développement de ce RNA en vue de son intégration dans un logiciel de simulation par éléments finis sont ensuite détaillées. Enfin, la dernière partie du chapitre se concentre sur la comparaison entre le modèle basé sur un réseau de neurones artificiels et les modèles analytiques identifiés dans la section 2.4. Les résultats montrent que le modèle basé sur un réseau de neurones artificiels offre une meilleure précision dans les prédictions.

3.1 Les Réseaux de Neurones Artificiels

Les réseaux de neurones artificiels constituent une avancée majeure dans le domaine de l'intelligence artificielle et de l'apprentissage automatique. Leur développement a été marqué par des étapes clés, allant des premières notions inspirées du cerveau humain jusqu'aux architectures sophistiquées d'aujourd'hui.

3.1.1 Bref historique de la méthode

L'histoire des réseaux de neurones artificiels remonte aux années 1940, lorsque Warren McCulloch et Walter Pitts ont publié *A Logical Calculus of Ideas Immanent in Nervous Activity* [McCulloch et Pitts, 1943], jetant ainsi les bases de la modélisation des neurones artificiels. Leurs travaux se sont inspirés des concepts biologiques du fonctionnement neuronal et ont posé les fondations de la logique binaire et des premiers modèles de neurones formels.

Frank Rosenblatt [Rosenblatt, 1957, Rosenblatt, 1958] a développé le perceptron dans les années 1950 et 1960 qui est considéré comme l'un des premiers modèles de réseau de neurones artificiels. Le perceptron était capable d'apprendre à reconnaître des motifs à partir de données d'entrée, mais ses limitations ont été mises en évidence dans le livre *Perceptrons* de Marvin Minsky et Seymour Papert [Minsky et Papert, 1969] en 1969. Après les critiques envers les capacités limitées des perceptrons, la recherche sur les réseaux de neurones artificiels a connu une période difficile appelée *l'IA hivernale*. Les progrès étaient rares, en grande partie en raison du manque de puissance de calcul et de données, ainsi que des défis conceptuels.

Les années 1980 ont vu un regain d'intérêt pour les réseaux de neurones artificiels grâce à l'introduction de nouvelles techniques d'apprentissage, telles que la rétro-propagation du gradient. Cette méthode a permis de résoudre le problème de l'apprentissage dans les réseaux à plusieurs couches, donnant ainsi naissance aux réseaux de neurones multicouches (ou perceptrons multicouches). Les travaux de chercheurs comme Geoffrey Hinton [Rumelhart *et al.*, 1986] ont été essentiels pour relancer l'intérêt pour les réseaux de neurones artificiels. Hornik *et al.* [Hornik *et al.*, 1989] ont rigoureusement établi en 1989 que les réseaux de neurones à action directe constituent une classe d'approximateurs universels, prolongeant ainsi les travaux proposés vingt ans auparavant par Minsky et Papert [Minsky et Papert, 1969] où ils démontraient que le simple perceptron à deux couches est incapable de représenter ou d'approximer utilement des fonctions en dehors d'une classe très étroite et spéciale.

Au cours des années 1990, les architectures des réseaux de neurones ont été explorées de manière plus approfondie. Les chercheurs ont développé des méthodes pour améliorer la mise en œuvre, la généralisation et la stabilité des réseaux. Cependant, malgré ces avancées, les limites de l'époque en matière de puissance de calcul ont entravé le déploiement pratique des réseaux de neurones à grande échelle.

Le début du 21^{ème} siècle a été marqué par des progrès significatifs dans les architectures des réseaux de neurones, notamment avec l'avènement des réseaux de neurones profonds. L'utilisation de GPU

(Graphical Processing Unit) pour accélérer les calculs a considérablement amélioré la formation et le déploiement de réseaux plus complexes. En 2012, le Réseau de Neurones Convolutif (RNC) d'AlexNet [Krizhevsky *et al.*, 2012] a remporté le concours ImageNet, attirant ainsi l'attention du grand public sur les capacités des réseaux de neurones artificiels.

La décennie actuelle a été témoin d'avancées spectaculaires dans les réseaux de neurones profonds. Les architectures telles que les réseaux de neurones récurrents (RNR) [Graves, 2013], les générateurs adverses (GAN) [Goodfellow *et al.*, 2014] et les transformers [Vaswani *et al.*, 2017] ont ouvert de nouvelles perspectives dans des domaines tels que la traduction automatique, la génération de contenu et la compréhension du langage naturel. Les architectures de pré-entraînement et de fine-tuning, comme le modèle GPT-3 puis GPT-4, ont atteint des niveaux de performance impressionnants dans diverses tâches linguistiques et cognitives.

3.1.2 Émergence des réseaux de neurones dans la technique

L'utilisation des réseaux de neurones artificiels en science et en physique est de plus en plus répandue, notamment en raison de la disponibilité actuelle d'outils efficaces permettant de programmer des réseaux de neurones artificiels grâce à des bibliothèques largement disponibles comme Tensorflow [Abadi *et al.*, 2015] ou Pytorch [Paszke *et al.*, 2019] par exemple. Les applications les plus médiatisées de l'apprentissage profond sont principalement liées au diagnostic médical, à la robotique, à la reconnaissance d'images et des langues, mais les applications vont bien au-delà et toutes les sciences peuvent désormais utiliser ces techniques, y compris la simulation numérique thermo-mécanique. Ainsi, les réseaux de neurones artificiels peuvent maintenant permettre de nouvelles approches pour la modélisation du comportement des matériaux et ont été appliqués avec succès dans la prédiction des relations constitutives de certains métaux et alliages ces dernières années.

L'applicabilité de ces techniques pour modéliser la dépendance plastique a été étudiée et une revue de la littérature peut être trouvée par exemple dans Gorji *et al.* [Gorji *et al.*, 2020] concernant l'utilisation des réseaux de neurones récurrents, dans Jamli *et al.* [Jamli *et al.*, 2019] concernant leur application dans l'analyse par éléments finis des processus de formage des métaux, ou dans Jiao *et al.* [Jiao *et al.*, 2020] concernant l'applicabilité aux méta-matériaux et leur caractérisation. Une distinction doit être faite entre les modèles de durcissement basés sur les réseaux de neurones et les modèles constitutifs basés sur les réseaux de neurones. Les deux approches ont été étudiées par de nombreux chercheurs au cours des trente dernières années. Ghaboussi *et al.* [Ghaboussi *et al.*, 1991] ont publié un article pionnier, dans lequel ils ont proposé un modèle constitutif basé sur les réseaux de neurones pour le béton plan sous chargement biaxial monotonique et chargement uniaxial cyclique dans lequel ils ont prédit avec succès plusieurs chemins de chargement dans la condition de chargement biaxial. Ils ont amélioré l'architecture du réseau en introduisant le réseau adaptatif et le réseau autoprogessif dans [Ghaboussi *et al.*, 1998, Ghaboussi *et al.*, 1998] où l'architecture du réseau évolue pendant la phase d'apprentissage pour mieux apprendre le comportement complexe contrainte-déformation des matériaux en utilisant une réponse globale charge-déflexion. L'approche adoptée pour cette étude est un modèle de durcissement basé sur les réseaux de neurones pour lequel, l'évaluation de la contrainte d'écoulement du matériau calculée par les réseaux de neurones est combinée avec un schéma d'intégration de type Retour-Radial.

Lin *et al.* [Lin *et al.*, 2008a] a développé un réseau pour prédire la contrainte d'écoulement de l'acier 42CrMo4 lors d'essais de compression à chaud sur un dispositif thermomécanique Gleeble et a montré une très bonne corrélation entre les résultats expérimentaux et prédits. Une extension à l'implémentation numérique de cette approche aurait été souhaitable. Javadi *et al.* [Javadi *et al.*, 2009] a utilisé un réseau pour capturer le comportement de matériaux complexes en utilisant un modèle EF incorporant les réseaux de neurones à rétro-propagation. Lu *et al.* [Lu *et al.*, 2011] a

présenté une étude comparative de la modélisation du comportement d'un alliage Al-Cu-Mg-Ag par une équation constitutive basée sur le paramètre de Zener-Hollomon et un modèle de réseaux de neurones. Il montre également que le modèle basé sur les réseaux de neurones propose une meilleure prédiction que l'équation constitutive.

Ashtiani *et al.* [Ashtiani et Shahsavari, 2016] a comparé les capacités de prédiction d'un modèle des réseaux de neurones contre une approche conventionnelle basée sur plusieurs lois de comportement telles que Johnson-Cook, Arrhenius et Arrhenius compensé par la déformation et a conclu à une meilleure efficacité et précision du réseaux de neurones dans la prédiction du comportement à chaud de l'alliage Al-Cu-Mg-Pb. Ashtiani *et al.* [Ashtiani et Shahsavari, 2016] ont montré qu'un modèle de réseau de neurones bien entraîné peut efficacement surmonter les lacunes de la physique provenant des comportements constitutifs analytiques tels que celui de Johnson-Cook, ou d'Arrhenius. Ali *et al.* [Ali *et al.*, 2019] a présenté un modèle de réseau de neurones couplé à une méthode éléments finis de plasticité cristalline dépendant de la vitesse pour simuler le comportement contrainte-déformation du matériau et l'évolution de la microstructure dans un AA6063-T6 sous cisaillement et traction simples. Stoffel *et al.* [Stoffel *et al.*, 2018, Stoffel *et al.*, 2019] a appliqué la méthode des réseaux de neurones à des déformations structurelles complexes de plaques chargées par ondes de choc impliquant des non-linéarités géométriques et physiques. Li *et al.* [Li *et al.*, 2019b] a mis en œuvre une routine VUMAT pour Abaqus où les paramètres ont été identifiés par une combinaison des formules analytiques et d'un algorithme de rétro-propagation. Récemment, Huang *et al.* [Huang *et al.*, 2021] a développé un modèle de réseau de neurones pour prédire la contrainte d'écoulement et l'évolution de la microstructure de l'alliage Ti-6Al-4V. Il a également montré la supériorité de cette approche par rapport à un modèle de comportement d'Arrhenius, notamment parce que le modèle RNA peut prédire la contrainte d'écoulement dans toute la gamme de déformation.

Les réseaux convolutifs temporels ont également été appliqués par Abueidda *et al.* [Abueidda *et al.*, 2021] pour prédire les réponses dépendant de l'histoire d'une classe de matériaux cellulaires. Des réseaux de neurones basés sur la thermodynamique ont également été proposés par Masi *et al.* [Masi *et al.*, 2021] pour réduire les incohérences physiques dans les prédictions du réseau. Ils démontrent la large applicabilité des réseaux convolutifs temporels pour la modélisation de matériaux élasto-plastiques, en utilisant des modèles d'hyper- et d'hypo-plasticité. Comme le présente Knight *et al.* [Knight et Nowotny, 2021], l'évolution des architectures de réseaux de neurones n'est pas la seule façon de progresser, l'évolution constante de l'architecture matérielle pour mettre en œuvre les réseaux de neurones devra également être prise en compte pour l'avenir.

Après une rapide présentation des fondamentaux des réseaux de neurones artificiels basée sur le perceptron simple et le perceptron multi-couches, nous allons voir en détail, les étapes de mise en place d'un modèle de réseau de neurones qui permet de définir une loi de comportement et son implémentation dans le code de calcul éléments finis Abaqus.

3.1.3 Perceptron simple

Le perceptron simple est l'unité de base des réseaux de neurones. C'est un modèle de classification binaire, capable de séparer linéairement deux classes de points. Il reçoit en entrée m valeurs x_i (avec $i \in \{1, m\}$) et renvoie une sortie y qui est une somme pondérée des entrées par des valeurs réelles w_i appelées poids synaptiques auxquelles on ajoute le biais b selon la relation suivante :

$$y = w_1x_1 + w_2x_2 + \dots + w_mx_m + b. \quad (3.1)$$

La notion de séparation est associée à une notion de probabilité introduite par une fonction d'activation liée à la sortie y de l'équation (3.1). Pour le modèle de Frank Rosenblatt la fonction d'activation

est binaire (fonction échelon à seuil fixe θ) qui prend la valeur 1 lorsque la somme pondérée est supérieure à θ et 0 lorsqu'elle est inférieure ou égale à θ .

$$\hat{y} = \begin{cases} +1 & \text{si } y > \theta \\ 0 & \text{si } y \leq \theta \end{cases} \quad (3.2)$$

Les deux principales étapes de calcul de la sortie d'un neurone formel sont résumées sur la Figure 3.1.

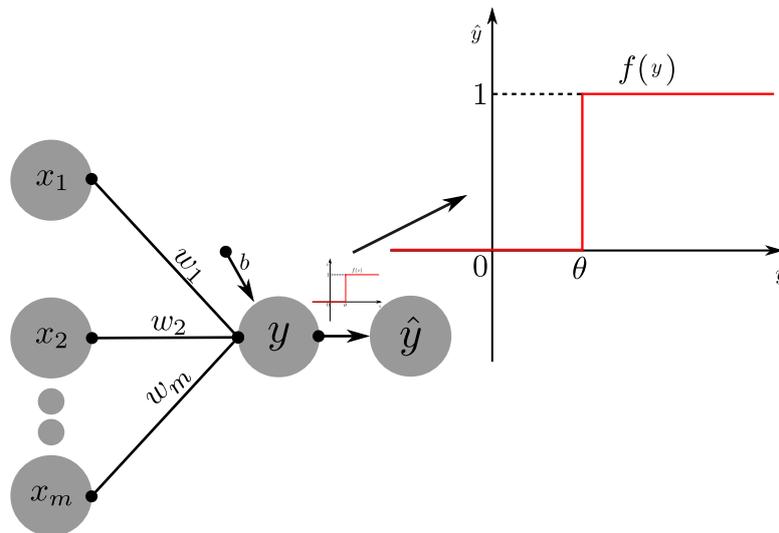


FIGURE 3.1 – Perceptron simple de Rosenblatt [Rosenblatt, 1958]

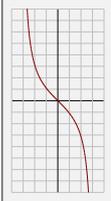
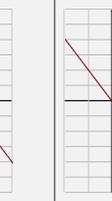
3.1.3.1 Fonctions d'activation

La fonction d'activation du perceptron simple donnée par l'équation (3.2) permet de prédire la classe associée à une entrée; soit 1 ou 0. Cette représentation relativement simpliste entraîne des insuffisances comme :

- La non flexibilité de la règle d'affectation sous-jacente à la fonction d'activation seuil.
- L'impossibilité de classer des données non linéairement séparables.

La fonction d'activation du perceptron simple peut facilement séparer deux données qui sont en réalité de la même classe et n'est efficace que pour des problèmes linéairement séparables, mais, cette fonction seuil brutale entraîne des problèmes de convergence et de traitement numérique du modèle. Plusieurs autres types de fonctions d'activation ont été proposées. Quelques unes parmi les plus utilisées sont regroupées dans le Tableau 3.1.

TABLE 3.1 – Fonctions d'activation classiquement utilisées dans les réseaux de neurones

Fonctions	Formules	Dérivées	Graphes	Étendue
Identité	$f(x) = x$	$f'(x) = 1$		\mathbb{R}
Heaviside	$f(x) = \begin{cases} 0 & \text{si } x < 0 \\ 1 & \text{si } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{si } x \neq 0 \\ ? & \text{si } x = 0 \end{cases}$		$[0; 1]$
Sigmoid	$f(x) = \frac{1}{1+e^{-x}}$	$f'(x) = f(x)(1 - f(x))$		$[0; 1]$
Arctan	$f(x) = \tan^{-1}(x)$	$f'(x) = \frac{1}{x^2+1}$		$[-\frac{\pi}{2}; \frac{\pi}{2}]$
tanh	$f(x) = \frac{2}{1+e^{-2x}} - 1$	$f'(x) = 1 - \tanh^2(x)$		$[-1; 1]$
ReLU	$f(x) = \begin{cases} 0 & \text{si } x < 0 \\ x & \text{si } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{si } x < 0 \\ 1 & \text{si } x \geq 0 \end{cases}$		\mathbb{R}_+
PReLU	$f(x) = \begin{cases} \alpha x & \text{si } x < 0 \\ x & \text{si } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} \alpha & \text{si } x < 0 \\ 1 & \text{si } x \geq 0 \end{cases}$		\mathbb{R}
ELU	$f(x) = \begin{cases} \alpha(e^x - 1) & \text{si } x < 0 \\ x & \text{si } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} f(x) + \alpha & \text{si } x < 0 \\ 1 & \text{si } x \geq 0 \end{cases}$		$[-\alpha; \infty[$

3.1.3.2 Apprentissage du modèle

L'apprentissage d'un réseau de neurones artificiels se décompose généralement en trois étapes successives : la propagation avant, le calcul de la fonction perte et la rétro-propagation.

1. Lors de la propagation avant (Forward Propagation), les données d'entrée \vec{x} sont transmises à travers le réseau de neurones pour générer des prédictions. Chaque neurone du réseau effectue une transformation linéaire de ces données d'entrée en utilisant les poids spécifiques du réseau (w_i et b), puis applique une fonction d'activation non linéaire comme définie précédemment pour produire une sortie \hat{y} .
2. Lorsque la sortie du réseau est atteinte, on doit évaluer la fonction perte (Loss Function). Après avoir généré des prédictions à l'aide de la propagation avant, il est nécessaire de mesurer à quel point ces prédictions sont éloignées des valeurs cibles attendues. Cela se fait en utilisant une fonction perte qui permet de quantifier les erreurs effectuées par un modèle par l'évaluation de l'écart entre les prédictions réalisées par le réseau et les valeurs réelles des observations au cours de l'apprentissage. Il existe plusieurs fonctions perte parmi lesquelles les plus largement utilisées sont la fonction Mean Root Squared Error (E_{RMS}) définie par l'équation (2.54) page 47 et la fonction Mean Absolute Error (E_{MAR}) définie par l'équation (2.53) page 46 respectivement.
3. La troisième étape est la rétro-propagation (Back Propagation). Une fois que la fonction perte a quantifié l'erreur, le réseau utilise cette information pour ajuster ses poids w_i et biais b afin de minimiser cette erreur. La rétro-propagation de l'erreur est une étape qui consiste à parcourir le réseau en sens inverse à la propagation (de la sortie vers l'entrée) dans le but d'estimer l'erreur effectuée par chaque neurone à partir de l'erreur globale. La rétro-propagation calcule donc les gradients de la fonction perte par rapport à chaque poids et biais dans le réseau. Ces gradients indiquent la direction et l'amplitude du changement requis pour minimiser la perte. En utilisant ces gradients, des algorithmes d'optimisation (tels que la descente de gradient par exemple) sont employés pour mettre à jour les paramètres du réseau.

Ces trois étapes se répètent itérativement pendant l'entraînement du réseau, ce qui lui permet d'apprendre à partir des données et de s'adapter à des tâches spécifiques, telles que la reconnaissance d'images, la traduction automatique, ou la prédiction de séquences. Plusieurs facteurs influencent l'apprentissage des réseaux de neurones artificiels parmi lesquels le choix de l'algorithme d'optimisation est fondamental. Plusieurs algorithmes d'optimisation existent dans le domaine du Machine Learning (ML) : la descente de gradient, Adam, Adamax, Adagrad, Nadam, RMSprop, etc. D'autres facteurs entrent en jeu, comme la nature des fonctions d'activation, le nombre de couches, le nombre de neurones par couche, et bien entendu les données d'entrée du modèle.

La littérature sur les réseaux de neurones artificiels est très fournie et de nombreuses sources d'apprentissage existent, parmi lesquelles le livre *Deep Learning* par Ian Goodfellow [Goodfellow *et al.*, 2016] qui est largement considéré comme l'un des ouvrages de référence pour comprendre les réseaux de neurones profonds et l'apprentissage profond en général. Il couvre en détail les concepts fondamentaux, y compris les architectures de réseaux, la régularisation, la rétro-propagation, et bien plus encore. Ce niveau d'apprentissage est largement suffisant pour comprendre la mise en œuvre des réseaux présentés dans ce travail de thèse. D'autres références se trouvent également dans [Heudin, 2016, Parizeau, 2004].

3.1.4 Le perceptron multicouches

Dans la précédente section, nous avons présenté le perceptron simple qui est capable de traiter des problèmes simples comme par exemple des problèmes linéaires. Or, dans la plupart des cas, les problèmes courants ne sont pas linéaires d'où la nécessité de trouver une solution alternative pour la résolution de ces problèmes. La méthode couramment utilisée consiste alors à associer plusieurs perceptrons simples (neurones) dans plusieurs couches. La notion de réseau de neurones artificiels, est utilisée lorsque l'on a plusieurs couches qui sont utilisées. Une couche est définie par un ensemble de neurones qui ne partagent pas de connexions entre eux. Un neurone d'une couche est connecté à tous les neurones de la couche précédente et tous les neurones de la couche suivante dans les cadre des architectures de réseau retenues pour cette étude.

Tous les neurones de la couche d'entrée constituent l'entrée du réseau et ceux de la couche de sortie forment la sortie du réseau. Entre la couche d'entrée et la couche de sortie, il est possible de mettre un certain nombre de couches intermédiaires communément appelées *couches cachées*. Le principe de fonctionnement des réseaux de neurones multicouches est donc basé sur deux mécanismes; la propagation en avant et la rétro-propagation entre les couches.

Pour illustrer le fonctionnement du perceptron multicouches, considérons un réseau à une couche cachée de deux neurones, 2 entrées et 1 sortie, de type 2-2-1, comme illustré sur la Figure 3.2.

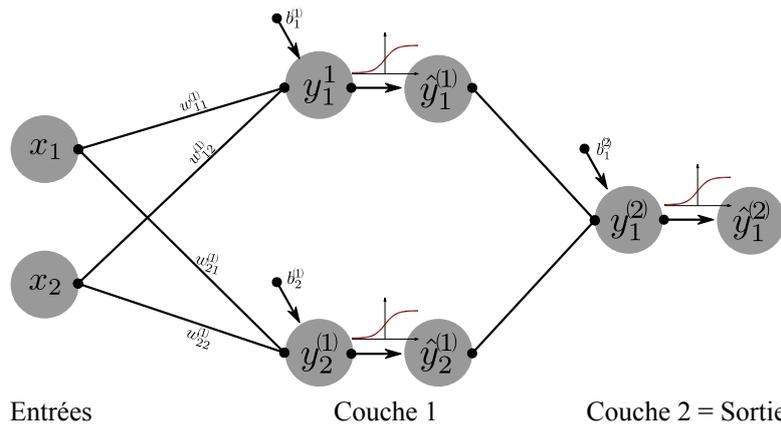


FIGURE 3.2 – Perceptron à une couche cachée, 2 entrées et 1 sortie, de type 2-2-1.

A partir de l'équation (3.1) et en utilisant la fonction sigmoïde pour définir les fonctions d'activation des neurones, on obtient le système d'équations suivant :

$$\left\{ \begin{array}{l} y_1^{(1)} = w_{11}^{(1)} x_1 + w_{12}^{(1)} x_2 + b_1^{(1)} \\ \hat{y}_1^{(1)} = \frac{1}{1 + e^{-y_1^{(1)}}} \\ y_2^{(1)} = w_{21}^{(1)} x_1 + w_{22}^{(1)} x_2 + b_2^{(1)} \\ \hat{y}_2^{(1)} = \frac{1}{1 + e^{-y_2^{(1)}}} \\ y_1^{(2)} = w_{11}^{(2)} x_1 + w_{12}^{(2)} x_2 + b_1^{(2)} \\ \hat{y}_1^{(2)} = \frac{1}{1 + e^{-y_1^{(2)}}} \end{array} \right. , \quad (3.3)$$

où $w_{ij}^{(k)}$ et $b_i^{(k)}$ sont les poids synaptiques et les biais, $y_i^{(k)}$ sont les valeurs intermédiaires et $\hat{y}_i^{(k)}$ sont les sorties du $i^{\text{ème}}$ neurone de la couche (k) ayant pour entrée la valeur x_j .

A partir de cet exemple simple, on peut étendre les réseaux de neurones à plusieurs couches avec autant de neurones que l'on souhaite par couche. Le principe de calcul de la valeur de sortie du réseau reste le même, c'est juste le nombre de paramètres internes qui augmente avec la complexité du réseau. Il faut noter que plus le réseau est profond (c'est-à-dire contenant plusieurs couches cachées et neurones par couche), plus il sera performant mais délicat à entraîner.

La Figure 3.3 donne un aperçu d'un réseau de neurones à (k) -couches cachées et contenant n -neurones par couche pour les $(k - 1)$ premières couches cachées, m -neurones pour la couche cachée (k) et 1 neurone pour la couche de sortie.

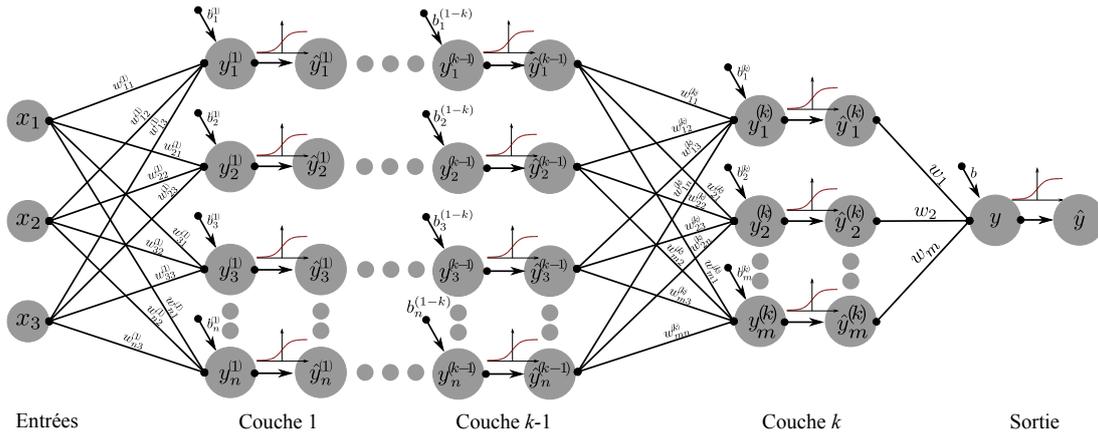


FIGURE 3.3 – Perceptron multicouches à (k) -couches cachées avec 3 entrées et 1 sortie.

3.1.4.1 Couche d'entrée

Selon la Figure 3.3, la couche d'entrée est définie par $\vec{x} = [x_1, x_2, x_3]^T$. Ces entrées représentent des quantités qui peuvent être de même nature ou de nature différente. Dans le cas d'une loi de comportement, \vec{x} a 3 composantes x_i qui sont la déformation ε , la vitesse de déformation $\dot{\varepsilon}$ et la température T . Comme nous le verrons par la suite, et compte tenu de l'utilisation de fonctions d'activation qui présentent un caractère de saturation, il sera nécessaire de normaliser les données d'entrée des réseaux de neurones. Ainsi, avant d'être introduites dans le réseau, une phase de pré-traitement de ces données d'entrée sera réalisée.

3.1.4.2 Couches cachées

Toute couche cachée (k) , contenant n neurones, prend une somme pondérée des sorties $\hat{y}^{(k-1)}$ de la couche précédente $(k - 1)$, contenant m neurones, donnée par l'équation suivante :

$$y_i^{(k)} = \sum_{j=1}^m w_{ij}^{(k)} \hat{y}_j^{(k-1)} + b_i^{(k)}, \quad (3.4)$$

où $y_i^{(k)}$ est la valeur du $i^{\text{ième}}$ neurone de la couche (k) , $w_{ij}^{(k)}$ est le paramètre poids entre le $i^{\text{ième}}$ neurone de la couche (k) et le $j^{\text{ième}}$ neurone de la couche $(k - 1)$, $b_i^{(k)}$ est le biais du $i^{\text{ième}}$ neurone de la couche (k) et $\hat{y}_j^{(k-1)}$ est la sortie du $j^{\text{ième}}$ neurone de la couche $(k - 1)$. Ces poids et biais sont les paramètres d'apprentissage du réseau qui seront ajustés pendant la procédure d'apprentissage. En utilisant la notation matricielle, l'équation (3.4) peut être réécrite sous la forme suivante :

$$\vec{y}^{(k)} = \mathbf{w}^{(k)} \cdot \vec{\hat{y}}^{(k-1)} + \vec{b}^{(k)}, \quad (3.5)$$

où $\vec{y}^{(k)} = [y_1^{(k)}, y_2^{(k)}, \dots, y_n^{(k)}]^T$ contient les valeurs internes des neurones issues de la sommation au niveau de la couche (k), $\vec{b}^{(k)} = [b_1^{(k)}, b_2^{(k)}, \dots, b_n^{(k)}]^T$ est la matrice biais de la couche (k) et $w^{(k)}$ est la matrice des paramètres poids [$n \times m$] de la couche (k) donnée ci-après :

$$w^{(k)} = \begin{bmatrix} w_{11}^{(k)} & w_{12}^{(k)} & \dots & w_{1m}^{(k)} \\ w_{21}^{(k)} & w_{22}^{(k)} & \dots & w_{2m}^{(k)} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n1}^{(k)} & w_{n2}^{(k)} & \dots & w_{nm}^{(k)} \end{bmatrix}. \quad (3.6)$$

Le nombre total des paramètres d'apprentissage N pour toute couche cachée (k) est la somme du nombre des poids et du nombre des biais de la couche (k), soit $N = n(m + 1)$. Après l'opération de sommation définie par l'équation (3.5), chaque neurone de la couche cachée (k) fournit une valeur de sortie $\hat{y}^{(k)}$ calculée à partir d'une fonction d'activation $f^{(k)}$ selon l'équation suivante :

$$\hat{y}_i^{(k)} = f^{(k)}(y_i^{(k)}) \quad \text{ou} \quad \vec{\hat{y}}^{(k)} = f^{(k)}(\vec{y}^{(k)}). \quad (3.7)$$

De nombreuses fonctions d'activation sont disponibles dans la littérature et leur choix dépend principalement du type d'application. Dans notre cas, il est très important que ces fonctions d'activation soient facilement dérivables afin de permettre par la suite le calcul de la dérivée de la contrainte σ en fonction de la déformation ε , de la vitesse de déformation $\dot{\varepsilon}$ et de la température T . Dans ce manuscrit, nous avons fait le choix d'en présenter deux parmi les plus utilisées qui sont les fonctions sigmoïde (sig) et tangente hyperbolique (tanh) (voir section 3.1.3.1).

3.1.4.3 Couche de sortie

La sortie s est calculée à partir des valeurs de la dernière couche cachée (l) du réseau, contenant m neurones, à l'aide de l'équation suivante :

$$s = \sum_{j=1}^m w_j \hat{y}_j^{(l)} + b, \quad (3.8)$$

où b est le biais associé au neurone de sortie et w_i sont les m paramètres poids entre la dernière couche cachée et le neurone de sortie s . En utilisant la notation matricielle, on peut réécrire cette relation comme ci-après :

$$s = \vec{w}^T \cdot \vec{\hat{y}}^{(l)} + b, \quad (3.9)$$

avec $\vec{w} = [w_1, w_2, \dots, w_m]^T$. Il n'y a pas de fonction d'activation pour le neurone de sortie (comme classiquement adopté pour le cas des problèmes de régression), donc s est directement la sortie du réseau. Le nombre total des paramètres pour la couche de sortie est $m + 1$.

Pour un réseau à deux couches cachées comportant m neurones sur la première couche cachée et n neurones sur la seconde, le nombre total des paramètres d'apprentissage est donc donné par : $N = 4m + n(m + 2) + 1$.

3.1.4.4 Calcul des dérivées pour N couches

Considérons un modèle de réseau de neurones à N -couches avec une entrée représentée par un vecteur \vec{x} et une sortie \vec{s} . Les paramètres de ce réseau sont les matrices contenant les poids synaptiques et biais de chaque couche. On utilisera les notations suivantes pour les étapes de calcul définies ci-après :

- On note $w^{(1)}$ la matrice des poids synaptiques de la couche 1 ; $w^{(2)}$ celle de la couche 2, $w^{(N)}$ celle de la couche N et w celle de la couche de sortie \vec{s} .
- On note $\vec{y}^{(1)}$ le vecteur des combinaisons linéaires s'effectuant sur chaque neurone de la couche 1 ; $\vec{y}^{(2)}$ celui de la couche 2 ; $\vec{y}^{(N)}$ celui de la couche N .
- On note $\vec{b}^{(1)}$ le vecteur biais associé à la couche 1 ; $\vec{b}^{(2)}$ celui associé à la couche 2, $\vec{b}^{(N)}$ celui associé à la couche N et \vec{b} celui de la couche de sortie.
- Finalement, $\vec{g}^{(1)} = f^{(1)}(\vec{y}^{(1)})$ est la sortie de la fonction d'activation appliquée à la couche 1 ; $\vec{g}^{(2)}$ celle de la couche 2 et $\vec{g}^{(N)}$ celle de la couche N .

A partir de ces définitions nous pouvons écrire les équations suivantes :

$$\left\{ \begin{array}{l} \vec{y}^{(1)} = w^{(1)} \cdot \vec{x} + \vec{b}^{(1)} \\ \vec{y}^{(2)} = w^{(2)} \cdot \vec{g}^{(1)} + \vec{b}^{(2)} \\ \vec{y}^{(3)} = w^{(3)} \cdot \vec{g}^{(2)} + \vec{b}^{(3)} \\ \vdots \\ \vec{y}^{(N)} = w^{(N)} \cdot \vec{g}^{(N-1)} + \vec{b}^{(N)} \end{array} \right. \implies \left\{ \begin{array}{l} \vec{g}^{(1)} = f^{(1)}(\vec{y}^{(1)}) \\ \vec{g}^{(2)} = f^{(2)}(\vec{y}^{(2)}) \\ \vec{g}^{(3)} = f^{(3)}(\vec{y}^{(3)}) \\ \vdots \\ \vec{g}^{(N)} = f^{(N)}(\vec{y}^{(N)}) \end{array} \right. , \quad (3.10)$$

et

$$\vec{s} = w^T \cdot \vec{g}^{(N)} + \vec{b}. \quad (3.11)$$

En utilisant la règle de dérivation en chaîne on a la relation suivante :

$$\vec{s}' = \frac{\partial \vec{s}}{\partial \vec{x}} = \frac{\partial \vec{s}}{\partial \vec{g}^{(N)}} \frac{\partial \vec{g}^{(N)}}{\partial \vec{y}^{(N-1)}} \cdots \frac{\partial \vec{g}^{(2)}}{\partial \vec{y}^{(1)}} \frac{\partial \vec{y}^{(1)}}{\partial \vec{x}}. \quad (3.12)$$

Si nous considérons la fonction tangente hyperbolique (\tanh) comme fonction d'activation pour les couches du réseau, on a :

$$\left\{ \begin{array}{l} \frac{\partial \vec{s}}{\partial \vec{g}^{(N)}} = w^T \\ \frac{\partial \vec{g}^{(N)}}{\partial \vec{y}^{(N-1)}} = \frac{\partial \vec{y}^{(N)}}{\partial \vec{y}^{(N-1)}} \frac{\partial f^{(N)}}{\partial \vec{y}^{(N-1)}}(\vec{y}^{(N)}) = w^{(N)} - w^{(N)} \circ \tanh^2(\vec{y}^{(N)}) \\ \frac{\partial \vec{g}^{(2)}}{\partial \vec{y}^{(1)}} = \frac{\partial \vec{y}^{(2)}}{\partial \vec{y}^{(1)}} \frac{\partial f^{(2)}}{\partial \vec{y}^{(1)}}(\vec{y}^{(2)}) = w^{(2)} - w^{(2)} \circ \tanh^2(\vec{y}^{(2)}) \\ \frac{\partial \vec{g}^{(1)}}{\partial \vec{x}} = \frac{\partial \vec{y}^{(1)}}{\partial \vec{x}} \frac{\partial f^{(1)}}{\partial \vec{x}}(\vec{y}^{(1)}) = w^{(1)} - w^{(1)} \circ \tanh^2(\vec{y}^{(1)}) \end{array} \right. , \quad (3.13)$$

où \circ est le produit élément par élément, connu sous le nom de produit de Hadamard, qui est une opération binaire qui prend deux matrices A et B de mêmes dimensions et produit une autre matrice C de même dimension que les opérands, où chaque élément $C_i = A_i B_i$.

L'équation (3.12) peut donc être réécrite sous la forme :

$$\vec{s}' = w^T \cdot \left[w^{(N)} - w^{(N)} \circ \tanh^2(\vec{y}^{(N)}) \right] \cdots \circ \left[w^{(2)} - w^{(2)} \circ \tanh^2(\vec{y}^{(2)}) \right] \circ \left[w^{(1)} - w^{(1)} \circ \tanh^2(\vec{y}^{(1)}) \right]. \quad (3.14)$$

Pour une fonction d'activation du type sigmoïde (sig) il suffit de remplacer la dérivée de la fonction $\tanh(x)$ dans l'équation (3.14) par celle de la fonction $\text{sig}(x)$ pour obtenir la relation suivante :

$$\vec{s}' = w^T \cdot \left(\frac{w^{(N)} \circ \exp(-\vec{y}^{(N)})}{(1 + \exp(-\vec{y}^{(N)}))^2} \right) \cdots \circ \left(\frac{w^{(2)} \circ \exp(-\vec{y}^{(2)})}{(1 + \exp(-\vec{y}^{(2)}))^2} \right) \circ \left(\frac{w^{(1)} \circ \exp(-\vec{y}^{(1)})}{(1 + \exp(-\vec{y}^{(1)}))^2} \right). \quad (3.15)$$

On rappelle que la fonction d'activation n'est pas utilisée sur la couche de sortie. Dans le cas où l'on souhaite utiliser une fonction d'activation spécifique à une couche, la généralisation de cette approche permet d'utiliser la dérivée de cette fonction d'activation en adaptant les deux équations précédentes.

3.2 Loi de comportement basée sur un RNA

3.2.1 Architecture du réseau de neurones

La capacité d'approximation universelle des réseaux de neurones, permet de prédire les valeurs de la contrainte d'écoulement σ en fonction des données d'entrée (ε , $\dot{\varepsilon}$ et T), à l'instar de ce qui est fait par les modèles analytiques identifiés dans le chapitre précédent. La principale différence est que cette approximation n'est pas liée à une formulation mathématique fixe (comme dans les modèles analytiques de Johnson–Cook, MZA, AR, HS et PTM), mais dépend uniquement des données utilisées pour l'apprentissage, et de l'architecture du réseau définie par le nombre de couches, le nombre de neurones par couche et la nature des fonctions d'activation associées aux neurones du réseau (également nommées hyperparamètres du réseau).

Après avoir testé différents types d'architectures du réseau, les travaux réalisés ont conduit à sélectionner pour la suite de l'étude une structure de réseau de neurones à deux couches cachées, comme illustré sur la Figure 3.4. La couche d'entrée est composée de 3 neurones représentant les

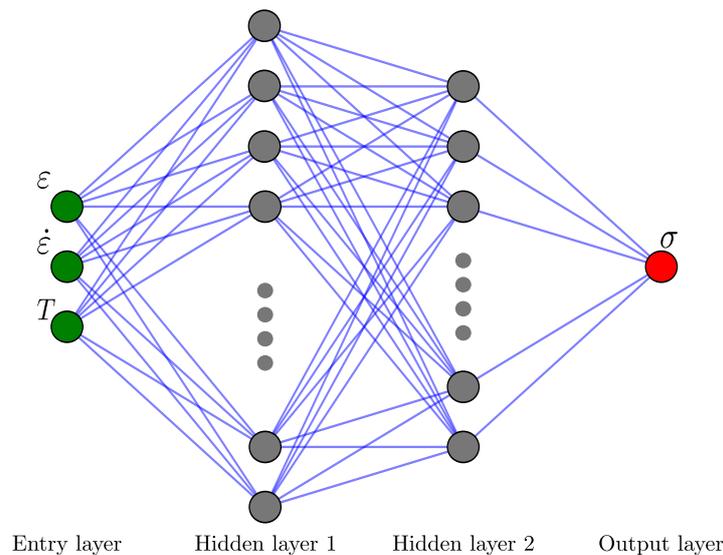


FIGURE 3.4 – Architecture générale d'un RNA modélisant une loi de comportement.

données physiques ε , $\dot{\varepsilon}$ et T , et la couche de sortie est composée d'un seul neurone correspondant à la contrainte d'écoulement σ .

3.2.2 Pré et post traitement des données

La présence de fonctions d'activation présentant une saturation en amplitude (toutes les valeurs sont contraintes dans le domaine $[-1, 1]$ ou $[0, 1]$ selon la fonction d'activation utilisée) entraîne la nécessité de modifier l'amplitude des valeurs d'entrée du réseau correspondant à la déformation ε , la vitesse de déformation $\dot{\varepsilon}$ et la température T dans la plage $[0, 1]$ [Lin *et al.*, 2008a, Lu *et al.*, 2011]. Par conséquent, les composantes de l'entrée \vec{x} du réseau sont calculées selon la procédure suivante :

- Puisque la contrainte dans les équations constitutives augmente généralement avec le logarithme de la vitesse de déformation ($\ln(\dot{\epsilon})$), il est opportun de pré-traiter cette vitesse de déformation en calculant le logarithme du rapport de la vitesse de déformation $\dot{\epsilon}$ sur la vitesse de déformation de référence $\dot{\epsilon}_0$ donnée par $\ln(\dot{\epsilon}/\dot{\epsilon}_0)$.
- Compte tenu de la remarque en début de paragraphe, on normalise ensuite les variables x_i dans l'intervalle $[0, 1]$.

Les trois composantes de l'entrée \vec{x} sont obtenues à partir de la déformation ϵ , de la vitesse de déformation $\dot{\epsilon}$ et de la température T en utilisant les expressions suivantes :

$$\vec{x} = \begin{cases} x_1 = \frac{\epsilon - [\epsilon]_{min}}{[\epsilon]_{max} - [\epsilon]_{min}} \\ x_2 = \frac{\ln(\dot{\epsilon}/\dot{\epsilon}_0) - [\ln(\dot{\epsilon}/\dot{\epsilon}_0)]_{min}}{[\ln(\dot{\epsilon}/\dot{\epsilon}_0)]_{max} - [\ln(\dot{\epsilon}/\dot{\epsilon}_0)]_{min}} \\ x_3 = \frac{T - [T]_{min}}{[T]_{max} - [T]_{min}} \end{cases}, \quad (3.16)$$

où $[]_{min}$ et $[]_{max}$ sont les limites de la plage du champ correspondant.

Pour la phase d'apprentissage du réseau, les contraintes σ sont également normalisées dans l'intervalle $[0, 1]$ à l'aide de l'expression suivante :

$$s = \frac{\sigma - [\sigma]_{min}}{[\sigma]_{max} - [\sigma]_{min}}. \quad (3.17)$$

3.2.2.1 Calcul de la contrainte d'écoulement

L'objectif du réseau étant de calculer la contrainte d'écoulement comme sortie du réseau, il est essentiel de montrer plus en détail le processus de calcul de l'équation (3.17). En utilisant la fonction d'activation du type sigmoïde (\vec{y} sig), la sortie de chacune des deux couches cachées ($\vec{y}^{(1)}$ pour la première couche cachée et $\vec{y}^{(2)}$ pour la seconde couche cachée) est donnée par les équations suivantes :

$$\begin{cases} \vec{y}^{(1)} = \frac{1}{1 + \exp(-w^{(1)} \cdot \vec{x} - \vec{b}^{(1)})} \\ \vec{y}^{(2)} = \frac{1}{1 + \exp(-w^{(1)} \cdot \vec{y}^{(1)} - \vec{b}^{(1)})} \end{cases}. \quad (3.18)$$

Ensuite la sortie s du réseau est obtenue à partir de la sortie $\vec{y}^{(2)}$ de la dernière couche cachée sans faire appel à la fonction d'activation selon la relation suivante :

$$s = w^T \cdot \vec{y}^{(2)} + b. \quad (3.19)$$

Comme aucune fonction d'activation n'est utilisée sur la sortie du réseau, la contrainte d'écoulement σ s'obtient alors comme suit :

$$\sigma = ([\sigma]_{max} - [\sigma]_{min}) s + [\sigma]_{min}. \quad (3.20)$$

Les valeurs $[]_{min}$ et $[]_{max}$ de la déformation ϵ , de la vitesse de déformation $\dot{\epsilon}$, de la température T , des contraintes σ ainsi que la valeur de la vitesse de déformation de référence $\dot{\epsilon}_0$ utilisées pendant la phase d'apprentissage doivent être stockées pour une utilisation ultérieure lors de l'implémentation des réseaux de neurones dans le code Abaqus/Explicit. Elles font partie de la solution finale avec les

poids $w^{(k)}$ et les valeurs de biais $\vec{b}^{(k)}$ des couches cachées et les poids \vec{w} et le biais b de la couche de sortie du réseau qui constituent les paramètres d'apprentissage du réseau.

La connaissance de ces quantités après la phase d'apprentissage permet ensuite d'extraire l'ensemble de l'architecture des réseaux de neurones de sa formulation en langage Python, sans le mécanisme d'apprentissage par rétro-propagation, pour son implémentation en Fortran 77 dans le code éléments finis Abaqus.

3.2.2.2 Calcul des dérivées pour la loi de comportement

L'implémentation des réseaux de neurones dans un code de calcul éléments finis à travers des algorithmes comme celui du Retour-Radial proposé par Ming et *al.* [Ming et Pantalé, 2018] par le biais d'une routine VUMAT ou directement par le biais d'une routine VUHARD pour Abaqus/Explicit, nécessite que le réseau renvoie non seulement la contrainte σ définie par l'équation (3.20), mais aussi les trois dérivées $\partial\sigma/\partial\varepsilon$, $\partial\sigma/\partial\dot{\varepsilon}$ et $\partial\sigma/\partial T$ sans avoir été entraîné à calculer ces dérivées. Ainsi conformément à l'architecture présentée sur la Figure 3.4, nous n'avons pas 4 sorties pour le réseau mais une seule et la capacité à calculer ces dérivées doit être intrinsèque.

Une solution simple, mais non recommandée, à ce problème consiste à calculer numériquement la dérivée de σ par rapport à ε , $\dot{\varepsilon}$ et T en utilisant la relation suivante :

$$\frac{\partial\sigma(x)}{\partial x} = \frac{\sigma(x + \delta x) - \sigma(x)}{\delta x}, \quad (3.21)$$

où δx est une petite augmentation ($\delta x = 10^{-6}$ par exemple) appliquée à l'une des 3 variables ε , $\dot{\varepsilon}$ et T successivement et indépendamment. Nous devons calculer quatre fois un résultat du réseau pour calculer la contrainte d'écoulement et approximer les trois dérivées, ce qui prend beaucoup de temps.

Une autre façon de calculer ces dérivées est de calculer la quantité $\vec{s}' = [s'_1, s'_2, s'_3]^T$ contenant les 3 dérivées de la sortie s définie par l'équation (3.9) du réseau par rapport à l'entrée \vec{x} . Ce calcul analytique dépend du nombre de couches cachées, et du type de fonctions d'activation utilisées. A partir de la généralisation faite dans la section 3.1.4.4, il est donc facile d'en déduire maintenant la dérivée d'un RNA à 1 et 2 couches cachées. De l'équation (3.14) et d'après la règle de dérivation par chaîne, avec une couche cachée et une fonction d'activation du type tangente hyperbolique (\tanh), \vec{s}' prend la forme suivante :

$$\vec{s}' = w^{(1)T} \cdot \left[\vec{w} - \vec{w} \circ \tanh^2 \left(\vec{y}^{(1)} \right) \right], \quad (3.22)$$

où $\vec{y}^{(1)}$ est donné par l'équation (3.5) avec $k = 1$. L'opérateur \tanh appliqué sur la quantité $\vec{y}^{(1)}$ est juste calculé pour chaque composante de $\vec{y}^{(1)}$, comme présenté sur la Figure 3.4, puisqu'il ne s'agit pas d'un réel mais seulement d'un tableau de réels (*i.e.* un tableau *numpy* dans notre cas, et l'opérateur et de type element-wise).

Avec une couche cachée et une fonction d'activation sigmoïde (sig), on obtient :

$$\vec{s}' = w^{(1)T} \cdot \left[\frac{\vec{w} \circ \exp(-\vec{y}^{(1)})}{[1 + \exp(-\vec{y}^{(1)})]^2} \right]. \quad (3.23)$$

Lorsque le nombre de couches cachées augmente, la complexité de la dérivée augmente, ainsi, pour deux couches cachées et une fonction d'activation \tanh pour les deux couches on obtient :

$$\vec{s}' = w^{(1)T} \cdot \left[w^{(2)T} \cdot \left(\vec{w} - \vec{w} \circ \tanh^2 \left(\vec{y}^{(2)} \right) \right) \circ \left(1 - \tanh^2 \left(\vec{y}^{(1)} \right) \right) \right]. \quad (3.24)$$

Enfin, pour deux couches cachées et une fonction d'activation sig on obtient :

$$\vec{s}' = w^{(1)T} \cdot \left[w^{(2)T} \cdot \left(\frac{\vec{w} \circ \exp(-\vec{y}^{(2)})}{[1 + \exp(-\vec{y}^{(2)})]^2} \right) \circ \left(\frac{\exp(-\vec{y}^{(1)})}{[1 + \exp(-\vec{y}^{(1)})]^2} \right) \right]. \quad (3.25)$$

En fonction du nombre de couches cachées et du type de fonctions d'activation utilisées, à partir des équations (3.22) à (3.25) et du fait du pré et post-traitement des quantités σ , ε , $\dot{\varepsilon}$ et T définies par les équations (3.16) et (3.17) on peut finalement obtenir les dérivées de la contrainte d'écoulement σ par rapport à ε , $\dot{\varepsilon}$ et T en utilisant l'expression suivante :

$$\begin{cases} \partial\sigma/\partial\varepsilon = s'_1 \frac{[\sigma]_{max} - [\sigma]_{min}}{[\varepsilon]_{max} - [\varepsilon]_{min}} \\ \partial\sigma/\partial\dot{\varepsilon} = \frac{s'_2}{\dot{\varepsilon}} \frac{[\sigma]_{max} - [\sigma]_{min}}{[\dot{\varepsilon}]_{max} - [\dot{\varepsilon}]_{min}} \\ \partial\sigma/\partial T = s'_3 \frac{[\sigma]_{max} - [\sigma]_{min}}{[T]_{max} - [T]_{min}} \end{cases}. \quad (3.26)$$

3.3 Validation du modèle RNA sur un 42CrMo4

Afin d'évaluer la performance de l'approche proposée pour l'implémentation des réseaux de neurones, nous avons décidé de reproduire le comportement de la loi d'écoulement de Johnson–Cook [Johnson et Cook, 1983] avec un réseau de neurones artificiels car il s'agit d'une des lois d'écoulement les plus utilisées pour la simulation des processus de déformation en dynamique. Reproduire le comportement d'une loi de Johnson–Cook par un modèle de réseau de neurones n'est bien sûr pas utile dans la pratique, mais c'est une phase nécessaire pour vérifier numériquement que les réseaux de neurones sont capables de prendre en compte un comportement non-linéaire et pour avoir un moyen de mesurer exactement les erreurs de prédiction du RNA tant dans le calcul de la contrainte σ que de ses trois dérivées $\partial\sigma/\partial\varepsilon$, $\partial\sigma/\partial\dot{\varepsilon}$ et $\partial\sigma/\partial T$.

Afin de valider le calcul des dérivées du RNA rapport aux grandeurs physiques, il est donc nécessaire de calculer analytiquement les dérivées de la loi d'écoulement de Johnson–Cook définie par l'équation (2.8) page 32. Il vient alors, sans difficulté majeure :

$$\begin{cases} \partial\sigma/\partial\varepsilon = nB\varepsilon^{n-1} \left[1 + C \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right) \right] \left[1 - \left(\frac{T - T_0}{T_m - T_0} \right)^m \right] \\ \partial\sigma/\partial\dot{\varepsilon} = \frac{C}{\dot{\varepsilon}} (A + B\varepsilon^n) \left[1 - \left(\frac{T - T_0}{T_m - T_0} \right)^m \right] \\ \partial\sigma/\partial T = \frac{-m(A + B\varepsilon^n)}{T - T_0} \left[1 + C \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right) \right] \left(\frac{T - T_0}{T_m - T_0} \right)^m \end{cases}. \quad (3.27)$$

3.3.1 Génération des données d'apprentissage

Un acier 42CrMo4, pour lequel les paramètres matériau proposés par Sattouf et *al.* [Sattouf et al., 2000, Dalverny et al., 2002] rapportés dans le Tableau 3.2 a été sélectionné comme matériau utilisé pour cette validation. Pour entraîner et valider le modèle RNA, nous avons utilisé un programme Python pour générer, grâce au modèle de Johnson–Cook et ses dérivées définies par l'équation (3.27) deux ensembles de données distincts :

TABLE 3.2 – Propriétés du matériau 42CrMo4 pour la validation du RNA.

E (GPa)	ν	A (MPa)	B (MPa)	C	n	m
206.9	0.29	806	614	0.0089	0.168	1.1
$\dot{\epsilon}_0$ (s^{-1})	T_0 °C	T_m °C	ρ kg/m^3	C_p $J/kg^\circ C$	α $10^{-6}/^\circ C$	η
1	20	1540	7830	460	12.3	0.9

- Le premier ensemble de données, celui de l'apprentissage, contient 2 520 points définis par 70 valeurs équidistantes de $\epsilon \in [0, 1]$, 6 taux de déformation $\dot{\epsilon} \in \{1, 10, 50, 500, 5\,000, 50\,000\}$ et 6 températures $\in \{20, 100, 200, 300, 400, 500\}$. Il contient donc les valeurs de la déformation ϵ , la vitesse de déformation $\dot{\epsilon}$, la température T , la contrainte d'écoulement σ calculée à partir de l'équation (2.8) page 32 du modèle de Johnson–Cook.
- Le deuxième ensemble de données pour tester le modèle, contient 5 000 points de données générés aléatoirement dans les plages $\epsilon \in [0, 1]$, $\dot{\epsilon} \in [1, 50\,000]$ et $T \in [20, 500]$. Il contient également les valeurs des trois dérivées calculées à partir de l'équation (3.27). Cet ensemble n'est pas utilisé lors de la phase d'apprentissage, mais uniquement lors de la validation.

Il est important de rappeler que le RNA doit permettre de remplacer la formulation analytique de la loi de comportement de Johnson–Cook afin de pouvoir réaliser une simulation numérique à partir de données expérimentales issues des tests thermomécaniques. Lors de ces essais, on acquiert la température T , la déformation ϵ , la vitesse de déformation $\dot{\epsilon}$ et les contraintes σ . Mais il est impossible d'acquérir les dérivées des contraintes par rapport aux quantités ϵ , $\dot{\epsilon}$, T . Par conséquent, dans une utilisation normale, il n'est pas possible d'inclure les dérivées dans la fonction objectif pour l'entraînement du RNA.

3.3.2 Apprentissage et analyse de la performance du RNA

Le programme Python utilisé pour l'apprentissage du réseau a été développé à l'aide de la bibliothèque Python spécialisée, Tensorflow [Abadi *et al.*, 2015, Mattmann, 2020]. L'optimiseur Adaptive Moment Estimation (ADAM) [Kingma et Lei, 2014] a été utilisé pour l'optimisation des paramètres internes du modèle.

Nous avons utilisé les deux fonctions d'activation présentées précédemment, les fonctions sigmoïde (sig) et tangente hyperbolique (tanh), et une ou deux couches cachées avec un nombre variable de neurones dans les couches cachées afin d'étudier l'influence de ces hyperparamètres sur la solution proposée. Tous les modèles portent le nom de leur constitution, où 3-x-1-tanh fait référence à un RNA à une couche cachée avec une fonction d'activation tanh et x neurones dans la couche cachée et 3-x-y-1-sig fait référence à un RNA à deux couches cachées avec une fonction d'activation sig dans les deux couches et x neurones dans la couche 1 et y neurones dans la couche 2. Tous les modèles ont été entraînés pour le même nombre d'itérations (50 000 itérations). Les temps d'entraînement pour tous les modèles sont plus ou moins les mêmes : environ 1 heure sur un ordinateur portable Dell XPS 13.

Dans le but d'illustrer l'efficacité du réseau, la Figure 3.5 et le Tableau 3.3 rapportent certains résultats obtenus après l'entraînement de six modèles différents (2 modèles à une couche cachée

et 4 modèles à deux couches cachées). De nombreuses configurations ont été étudiées, mais seuls

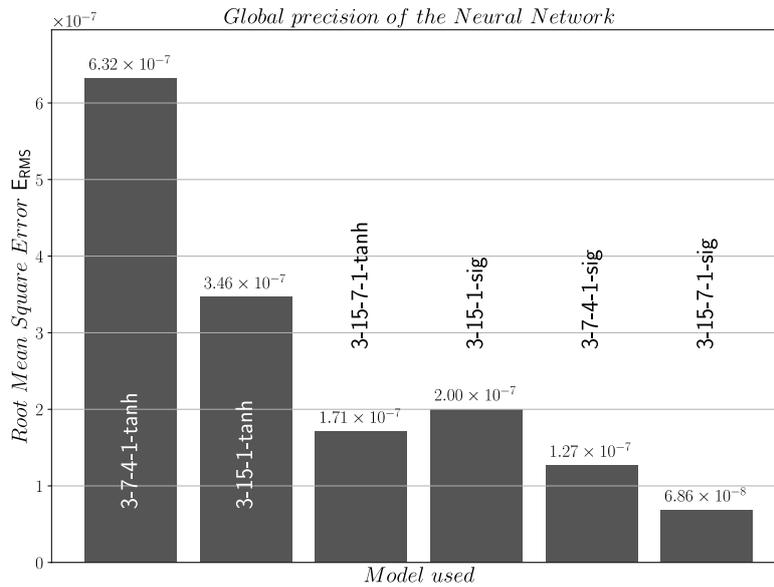


FIGURE 3.5 – Comparaison de la précision de six modèles de RNA.

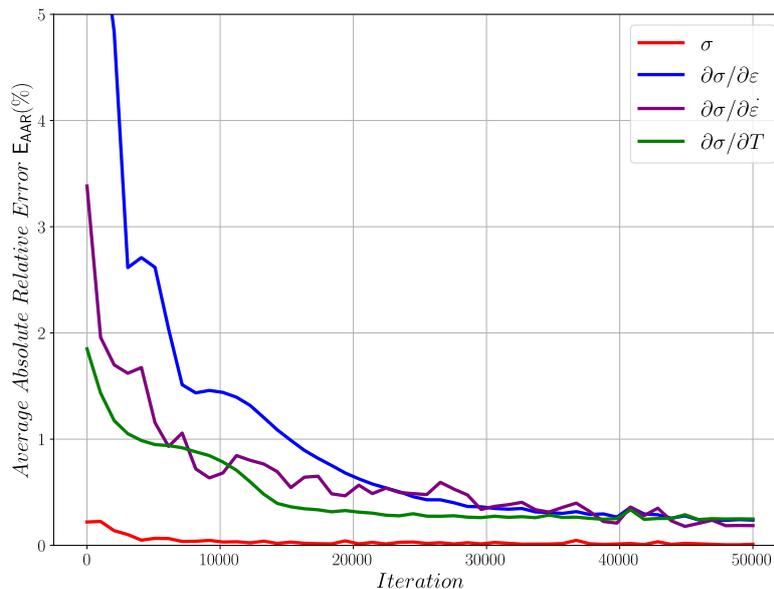


FIGURE 3.6 – Convergence des valeurs prédites par le modèle 3-15-7-1-sig

six modèles sont présentés ici pour illustrer les tendances. La Figure 3.5 montre un histogramme des valeurs moyennes des E_{RMS} évaluées au cours des derniers 5% du processus d'apprentissage donnant une idée de la convergence globale de l'algorithme ADAM. Dans le tableau 3.3, N est le nombre de paramètres internes du réseau. Les valeurs $\Delta \square$ se rapportent à l'erreur relative absolue moyenne (E_{AAR}) donnée par en utilisant l'équation (2.54) page 47.

A la lecture des données du Tableau 3.3, la performance globale des réseaux proposés est très bonne puisque l'erreur sur l'évaluation de la contrainte est d'environ 0.01% pour le réseau le plus performant alors qu'elle ne dépasse pas 0.04% pour tous les réseaux testés. L'évaluation des dérivées est également très bonne puisque l'erreur est de l'ordre de 0.2% pour le réseau le plus performant

TABLE 3.3 – Performance globale du modèle RNA pendant l'apprentissage

Modèles	N	E_{RMS} $\times 10^{-7}$	$\Delta\sigma$ %	$\Delta(\partial\sigma/\partial\varepsilon)$ %	$\Delta(\partial\sigma/\partial\dot{\varepsilon})$ %	$\Delta(\partial\sigma/\partial T)$ %
3-7-4-1-tanh	65	6.32	0.038	1.977	0.792	0.556
3-15-1-tanh	78	3.46	0.039	1.506	0.269	0.371
3-15-7-1-tanh	180	1.71	0.030	0.519	0.380	0.408
3-15-1-sig	78	2.00	0.030	0.686	0.521	0.675
3-7-4-1-sig	65	1.27	0.024	0.670	0.415	0.499
3-15-7-1-sig	180	0.68	0.011	0.247	0.199	0.256

(alors qu'elle reste inférieure à 2.0% sur l'ensemble des modèles) pour le terme $\partial\sigma/\partial\varepsilon$, 0.8% pour le terme $\partial\sigma/\partial\dot{\varepsilon}$ et 0.6% pour le terme $\partial\sigma/\partial T$.

Sans trop de surprise, on remarque que globalement, l'erreur sur l'évaluation de la contrainte d'écoulement est plus faible d'un facteur 10 que l'erreur sur l'évaluation des dérivées. Ceci s'explique facilement par le fait que le réseau a été entraîné pour minimiser l'erreur sur la contrainte d'écoulement uniquement. Néanmoins, sans jamais avoir été entraîné pour le calcul des dérivées, nous pouvons constater que les résultats obtenus sont également très bons concernant le calcul des dérivées.

La Figure 3.6 montre la convergence du modèle RNA 3-15-7-1-sig concernant l'évaluation de la contrainte d'écoulement et des 3 dérivées avec le nombre d'itérations. De ce dernier, on peut voir que la convergence du réseau sur l'évaluation de σ est plus rapide que la convergence des dérivées. Le facteur de 10 mentionné précédemment entre la contrainte et les dérivées est clairement visible sur ce graphique. Si l'erreur sur le calcul de la contrainte n'évolue pas après 10 000 itérations de l'algorithme d'apprentissage, les erreurs sur l'évaluation des dérivées nécessitent un nombre d'itérations plus important. Il est donc nécessaire de poursuivre l'apprentissage de ce type de modèle lorsque le critère de convergence de la contrainte est déjà satisfait afin de permettre la convergence des dérivées. On peut alors être confronté à un problème de sur-apprentissage, il est donc nécessaire de dimensionner la taille des réseaux de neurones au plus juste par rapport à l'application.

3.4 Identification de la loi d'écoulement du 3Cr2Mo

3.4.1 Apprentissage et prédiction du réseau

Nous reprenons dans cette partie à nouveau les résultats des essais de compression sur le dispositif Gleeble présentés dans la Section 1.4.3. Après avoir testé plusieurs architectures de réseaux, et conformément aux résultats présentés dans la précédente section, une structure de réseau à deux couches cachées, comme illustré sur la Figure 3.4, comprenant 15 neurones pour la première couche cachée et 7 neurones pour la seconde couche et des fonctions d'activation sigmoïde a donné le meilleur compromis entre prédiction, temps d'apprentissage et compacité du modèle. Par la suite, ce réseau sera noté 3-15-7-1-sig. Cette architecture conduit à un modèle global avec 180 paramètres à identifier (60 pour la première couche, 112 pour la deuxième couche et 8 pour la couche de sortie).

Le programme Python utilisé pour l'apprentissage du réseau a été à nouveau développé à l'aide de la bibliothèque Python Tensorflow et l'optimiseur ADAM a été utilisé pour l'apprentissage. Les données d'entraînement sont identiques à celles utilisées pour l'identification des modèles analytiques, et

sont composées de 21 000 quadruplets de valeurs (ε , $\dot{\varepsilon}$, T , σ). L'apprentissage est effectué sur la base de 5 000 époques de l'ensemble des données expérimentales. 40 minutes d'entraînement sur un ordinateur portable Dell XPS-13 7390 exécutant Ubuntu 22.04 LTS 64 bits avec 16 Go de Ram et un processeur Intel 4-core i7-10510U ont permis d'obtenir de la convergence du modèle. La Figure 3.7 montre l'évolution de l'erreur d'apprentissage définie par le \log_{10} du E_{RMS} pendant la phase d'apprentissage. Comme constaté sur cette figure, après 5 000 époques, on peut considérer que l'on

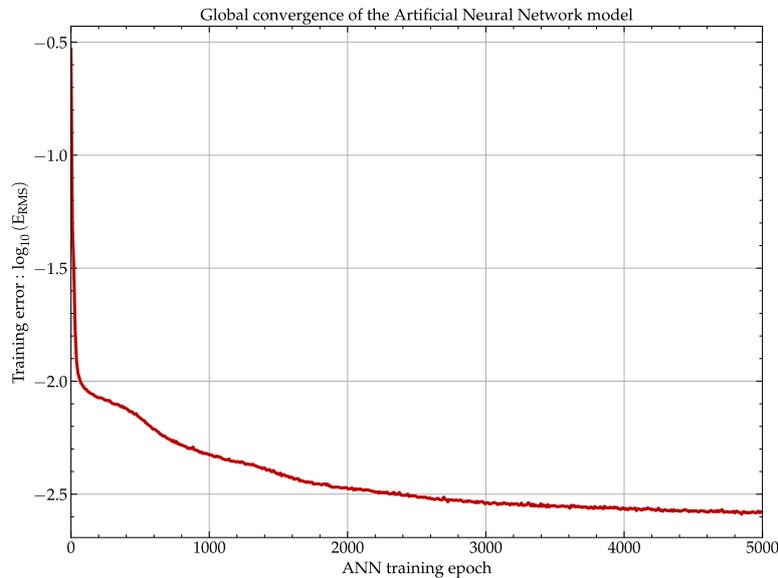


FIGURE 3.7 – Convergence du RNA 3-15-7-1-sig lors de l'apprentissage.

a atteint un état stationnaire de l'apprentissage du modèle, et qu'il est inutile de poursuivre la phase d'apprentissage.

Une fois la phase d'apprentissage terminée, le modèle peut être utilisé pour prédire le comportement du matériau d'étude (acier 3Cr2Mo dans notre cas) en fonction des données d'entrée, de la même manière que pour les modèles analytiques. On peut soit utiliser le modèle directement en lui fournissant de nouvelles données d'entrée, soit récupérer les 180 paramètres identifiés lors de l'apprentissage et les injecter dans un modèle mathématique basé sur les équations (3.5) et (3.7), qui peut être implémenté dans n'importe quel langage (*e.g.* en Fortran 77 pour une utilisation dans le code Abaqus/Explicit), comme proposé dans Pantalé *et al.* [Pantalé *et al.*, 2022, Pantalé, 2023].

La Figure 3.8 montre une comparaison entre les contraintes d'écoulement prédites par le modèle RNA et l'expérience. Les données expérimentales et la prédiction RNA sont en très bon accord sur toute la gamme de valeurs et les contraintes prédites peuvent suivre les régions de durcissement et d'adoucissement du matériau déformé à chaud. Les coefficients permettant d'évaluer la qualité de la solution sont $E_{MAR} = 0.62 \%$ et $E_{RMS} = 0.38 \text{ MPa}$ traduisant ainsi une très bonne prédiction du RNA.

Étant donné que les prédictions du modèle RNA reposent sur les données expérimentales utilisées pour identifier les modèles analytiques présentés dans le chapitre précédent, nous avons effectué une comparaison entre les coefficients E_{MAR} et E_{RMS} des modèles analytiques et du modèle RNA dans la section suivante.

3.4.2 Comparaison des modèles analytiques avec le RNA

Le Tableau 3.4 présente un résumé des coefficients permettant d'évaluer la capacité de prédiction de l'écoulement à haute température de l'alliage 3Cr2Mo pour tous les modèles présentés dans ce

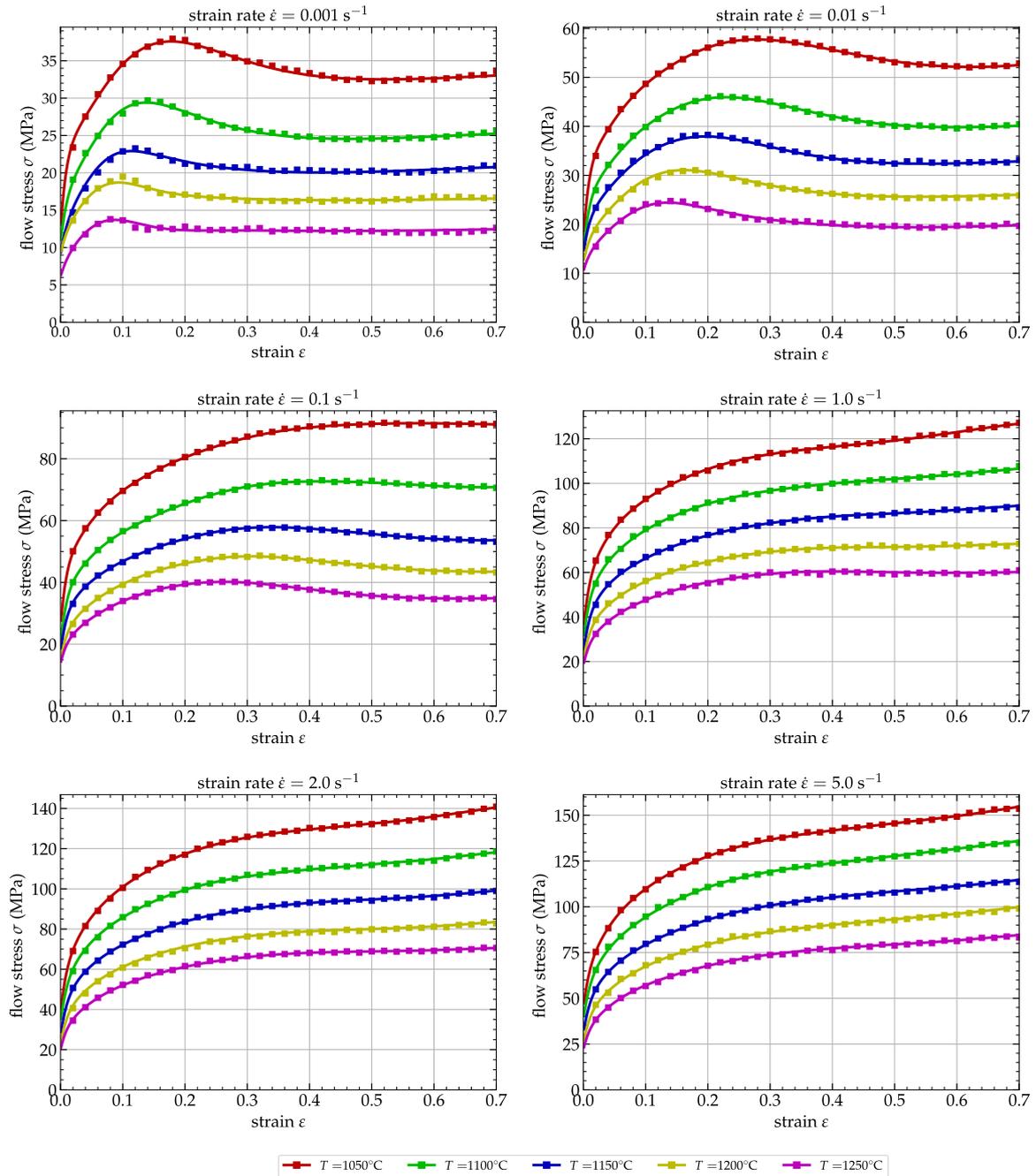


FIGURE 3.8 – Comparaison entre contraintes d'écoulement expérimentales et celles prédites par le modèle RNA 3-15-7-1-sig

travail. A partir de ce tableau, nous pouvons constater que le modèle RNA a une capacité prédictive

TABLE 3.4 – Comparaison des modèles de comportement identifiés sur l'acier 3Cr2Mo.

Coefficients	JC	MZA	HS	AR	PTM	RNA
$E_{MAR}(\%)$	14.05	21.20	7.75	3.56	4.79	0.62
$E_{RMS}(\text{MPa})$	12.00	19.57	3.80	2.18	4.59	0.38

nettement supérieure à tous les modèles analytiques présentés dans les chapitres précédents.

Globalement, les valeurs de E_{MAR} et de E_{RMS} du modèle RNA sont 6 fois inférieures à celles du meilleur des modèles analytiques, c'est-à-dire le modèle d'Arrhenius, cité comme référence dans le contexte du formage à chaud des alliages [Liang *et al.*, 2022, Zhu *et al.*, 2022].

Les modèles RNA, d'Arrhenius et PTM sont les seuls modèles qui prennent en compte l'adoucissement avec la déformation à faible vitesse de déformation, contrairement aux trois autres modèles, qui présentent uniquement une augmentation de la contrainte d'écoulement avec la déformation, indépendamment de la vitesse de déformation et de la température, d'où leur mauvaise performance dans la prédiction du comportement de ce matériau, notamment à faible vitesse de déformation.

Les paramètres présentés dans le Tableau 3.4 et les corrélations visibles sur les Figures 2.6, 2.7, 2.8, 2.11, 2.13 pages 46–53 et la Figure 3.8 permettent de conclure que le modèle RNA est le plus performant parmi tous les modèles présentés lorsqu'il s'agit de décrire le comportement de cet alliage pour des applications de déformation à haute température.

3.4.3 Capacité d'interpolation/extrapolation des modèles

Afin de mieux comparer les performances des modèles analysés au cours de cette étude (les cinq modèles analytiques du chapitre précédent et le modèle RNA de ce chapitre), nous analysons ci après la capacité de chacun de ces modèles à interpoler et extrapoler les résultats en fonction de la vitesse de déformation $\dot{\epsilon}$.

L'identification qui a été faite pour ces lois repose sur un ensemble de données expérimentales correspondant à six vitesses de déformation ($\dot{\epsilon} = 0.001 \text{ s}^{-1}$, $\dot{\epsilon} = 0.01 \text{ s}^{-1}$, $\dot{\epsilon} = 0.1 \text{ s}^{-1}$, $\dot{\epsilon} = 1 \text{ s}^{-1}$, $\dot{\epsilon} = 2 \text{ s}^{-1}$ et $\dot{\epsilon} = 5 \text{ s}^{-1}$), cinq températures (T) et des déformations allant de $\epsilon = 0$ à $\epsilon = 0.7$.

Pour tester la fiabilité de ces modèles, nous proposons ici de réaliser à nouveau ces identifications sur seulement cinq des six vitesses de déformation en omettant volontairement la vitesse de déformation de $\dot{\epsilon} = 1 \text{ s}^{-1}$ dans une première phase, puis $\dot{\epsilon} = 5 \text{ s}^{-1}$ dans une seconde phase.

Ainsi, comme la première vitesse de déformation omise ($\dot{\epsilon} = 1 \text{ s}^{-1}$) se situe dans la plage des vitesses de déformation pour l'identification du modèle ($\dot{\epsilon} = 0.001 \text{ s}^{-1} - 5 \text{ s}^{-1}$), nous pouvons tester la capacité des modèles à interpoler les résultats des cinq autres vitesses de déformation et quantifier toute déviation par rapport aux valeurs expérimentales.

En revanche, puisque la vitesse de déformation omise ($\dot{\epsilon} = 5 \text{ s}^{-1}$), dans la deuxième phase, se situe en dehors de la plage des vitesses de déformation pour l'identification du modèle ($\dot{\epsilon} = 0.001 \text{ s}^{-1} - 2 \text{ s}^{-1}$), nous testerons la capacité des modèles à extrapoler les résultats et quantifier les déviations par rapport aux valeurs expérimentales.

3.4.3.1 Validation de l'interpolation

Pour la validation par interpolation, la vitesse de déformation omise choisie est $\dot{\epsilon} = 1 \text{ s}^{-1}$, tandis que celles utilisées pour l'identification (ou l'entraînement pour le réseau de neurones) sont les cinq autres, c'est-à-dire $\dot{\epsilon} = \{0.001, 0.01, 0.1, 2, 5\} \text{ s}^{-1}$. Tous les modèles ont été ré-identifiés à partir des mêmes données expérimentales.

La Figure 3.9 montre une comparaison, pour la vitesse de déformation de $\dot{\epsilon} = 1 \text{ s}^{-1}$ (la vitesse de déformation n'ayant pas servie lors de l'identification), des contraintes à l'écoulement σ^y calculées par les modèles (sous forme de ligne) et les résultats expérimentaux (sous forme de points) pour les cinq modèles analytiques et le RNA. On peut constater que les deux premiers modèles (JC et MZA),

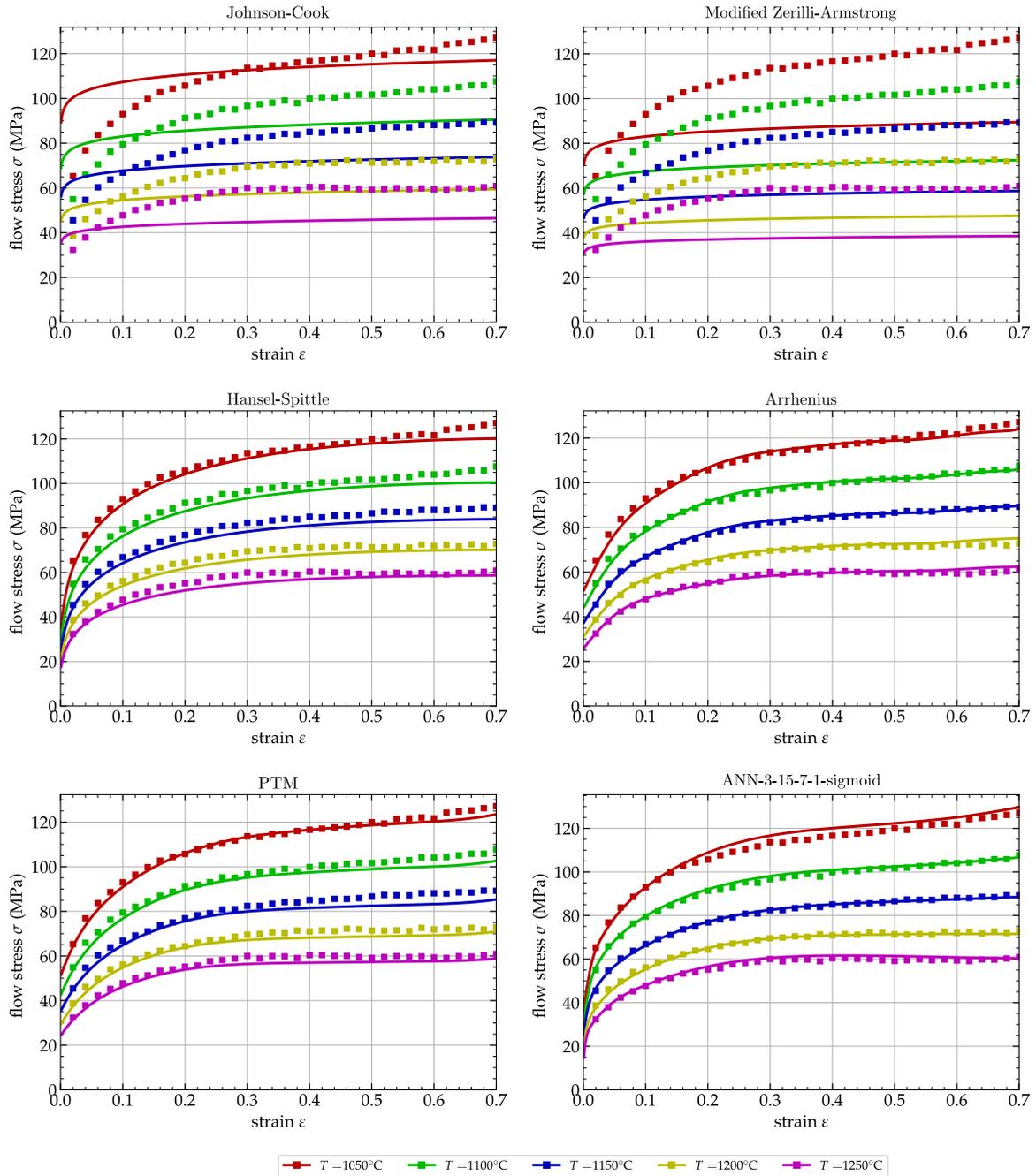


FIGURE 3.9 – Comparaison entre contraintes d'écoulement expérimentales et celles prédites par les six modèles pour $\dot{\epsilon} = 1 \text{ s}^{-1}$

présentés dans cette étude, n'ont pas la capacité de reproduire le comportement du matériau pour la vitesse de déformation de $\dot{\epsilon} = 1 \text{ s}^{-1}$.

Le Tableau 3.5 présente erreurs E_{MAR} et E_{RMS} entre les différents modèles et les données expérimentales, calculées soit pour les cinq vitesses de déformation identifiées (lignes référencées comme cinq- $\dot{\epsilon}$), soit pour les six vitesses de déformation (lignes référencées comme six- $\dot{\epsilon}$), ou uniquement pour la vitesse de déformation de $\dot{\epsilon} = 1 \text{ s}^{-1}$.

Dans l'ensemble, le modèle de Johnson-Cook donne de meilleurs résultats que le modèle MZA pour $\dot{\epsilon} = 1 \text{ s}^{-1}$, ce qui se reflète dans le Tableau 3.5 par une valeur plus faible de E_{MAR} et E_{RMS} pour JC que pour MZA. Néanmoins, ces valeurs sont supérieures à 10% pour le modèle JC et à 20% pour le modèle

TABLE 3.5 – Précision de l'interpolation pour tous les modèles avec $\dot{\epsilon} = 1 \text{ s}^{-1}$.

Vitesse $\dot{\epsilon}$	Coefficients	JC	MZA	HS	AR	PTM	RNA
cinq- $\dot{\epsilon}$	$E_{\text{MAR}}(\%)$	13.79	20.22	8.57	3.96	5.10	0.70
	$E_{\text{RMS}}(\text{MPa})$	11.83	19.00	3.97	2.29	4.73	0.38
$\dot{\epsilon} = 1 \text{ s}^{-1}$	$E_{\text{MAR}}(\%)$	14.87	27.79	3.74	1.46	3.16	2.47
	$E_{\text{RMS}}(\text{MPa})$	12.02	23.67	3.10	1.46	2.65	2.77
six- $\dot{\epsilon}$	$E_{\text{MAR}}(\%)$	14.42	28.90	7.55	3.45	4.90	0.96
	$E_{\text{RMS}}(\text{MPa})$	11.86	19.86	3.84	2.17	4.45	1.18

MZA, ce qui reflète la capacité médiocre de ces modèles à modéliser correctement le comportement de ce matériau. Cette constatation est en accord avec les résultats du chapitre précédent qui ont montré l'incapacité de ces modèles à prendre en compte l'adoucissement du comportement à faible vitesse de déformation et à basse température.

D'un point de vue général, les quatre autres modèles, HS, AR, PTM et RNA, donnent globalement des résultats similaires, avec une fiabilité supérieure pour les modèles AR et RNA par rapport aux deux autres modèles. Le Tableau 3.5 montre une comparaison quantitative en E_{MAR} et E_{RMS} pour trois cas différents : sur les cinq vitesses de déformation identifiées, uniquement pour la vitesse de déformation de $\dot{\epsilon} = 1 \text{ s}^{-1}$, et sur les six vitesses de déformation pour tous les modèles.

Il ressort des analyses du contenu de ce tableau que, tandis que les deux modèles, AR et RNA, donnent des résultats équivalents (et excellents) en ce qui concerne les valeurs de E_{MAR} et E_{RMS} pour la vitesse de déformation de $\dot{\epsilon} = 1 \text{ s}^{-1}$, le modèle RNA donne globalement de meilleurs résultats sur l'ensemble du spectre des vitesses de déformation, avec des valeurs de $E_{\text{MAR}} = 0.96\%$ et $E_{\text{RMS}} = 1.18 \text{ MPa}$ respectivement ; c'est-à-dire, des valeurs environ 2 à 3 fois inférieures pour le modèle RNA par rapport au modèle AR.

Cela montre la supériorité du modèle RNA par rapport aux cinq modèles analytiques présentés dans cette étude, tant en ce qui concerne la capacité d'interpolation du modèle que le comportement global.

3.4.3.2 Validation de l'extrapolation

Pour la validation de la capacité des modèles à extrapoler les données, la vitesse de déformation omise choisie est de $\dot{\epsilon} = 5 \text{ s}^{-1}$, et celles utilisées pour l'identification sont $\dot{\epsilon} = \{0.001, 0.01, 0.1, 1, 2\} \text{ s}^{-1}$. La vitesse de déformation omise dans cette analyse a donc la valeur la plus élevée, ce qui restreint le domaine d'identification. Dans cette nouvelle configuration d'identification, la Figure 3.10 montre une comparaison, pour la vitesse de déformation de $\dot{\epsilon} = 5 \text{ s}^{-1}$, des contraintes à l'écoulement σ calculées par les modèles (sous forme de ligne) et les résultats expérimentaux (sous forme de points) pour les cinq modèles analytiques et le RNA. Comme dans la section précédente concernant la capacité d'interpolation des modèles, sur la Figure 3.10, on peut constater que les deux premiers modèles, JC et MZA, dans cette étude n'ont toujours pas la capacité de reproduire correctement le comportement du matériau pour la vitesse de déformation de $\dot{\epsilon} = 5 \text{ s}^{-1}$.

Le Tableau 3.6 présente les déviations en terme de E_{MAR} et E_{RMS} entre les différents modèles et les données expérimentales, calculées soit pour les cinq vitesses de déformation identifiées (lignes référencées comme cinq- $\dot{\epsilon}$), les six vitesses de déformation (lignes référencées comme six- $\dot{\epsilon}$), ou uniquement pour la vitesse de déformation de $\dot{\epsilon} = 5 \text{ s}^{-1}$.

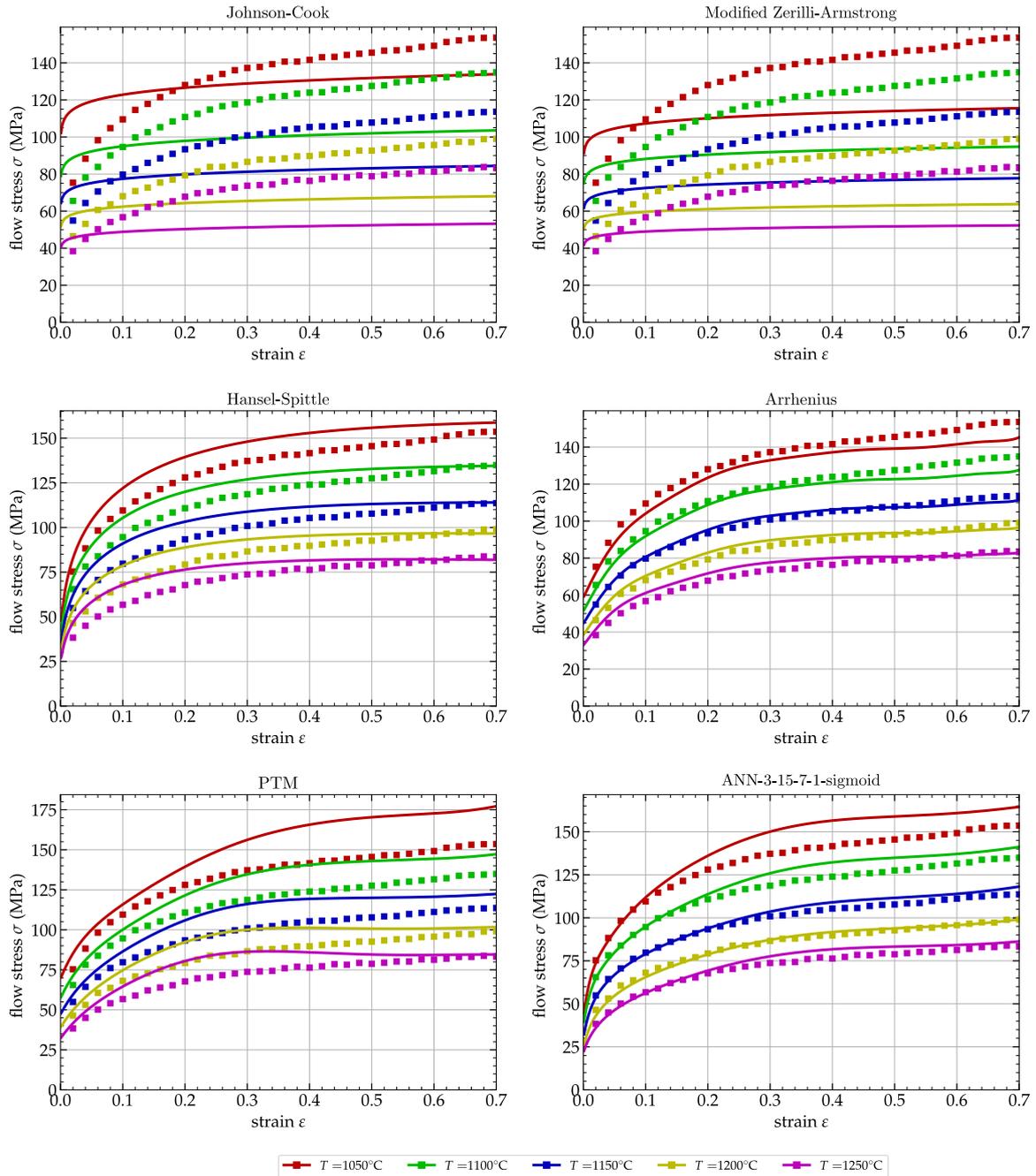


FIGURE 3.10 – Comparaison entre contraintes d'écoulement expérimentales et celles prédites par les six modèles pour $\dot{\epsilon} = 5 \text{ s}^{-1}$

Le modèle de Johnson–Cook donne de meilleurs résultats que le modèle MZA pour $\dot{\epsilon} = 5 \text{ s}^{-1}$, ce qui se reflète dans le Tableau 3.6 par une valeur plus faible de E_{MAR} et E_{RMS} pour Johnson–Cook que pour MZA. Néanmoins, ces valeurs sont trop élevées pour une utilisation adéquate. Une fois de plus, c'est l'incapacité de ces modèles à prendre en compte le ramollissement du comportement à faible vitesse de déformation et à basse température qui est à l'origine de ces valeurs.

Les quatre autres modèles, HS, AR, PTM et RNA, donnent de meilleurs résultats avec une fiabilité supérieure pour les modèles AR et RNA par rapport aux deux autres modèles. Le Tableau 3.6 montre une comparaison sur l'ensemble des vitesses de déformation et sur $\dot{\epsilon} = 5 \text{ s}^{-1}$ pour ces six modèles. Le modèle HS a une performance moindre en extrapolation que les modèles AR et RNA, tandis que

TABLE 3.6 – Précision de l'extrapolation des tous les modèles pour $\dot{\epsilon} = 5 \text{ s}^{-1}$.

Vitesse $\dot{\epsilon}$	Coefficients	JC	MZA	HS	AR	PTM	RNA
cinq- $\dot{\epsilon}$	$E_{\text{MAR}}(\%)$	13.09	19.09	7.30	4.03	4.34	0.61
	$E_{\text{RMS}}(\text{MPa})$	9.86	16.26	3.36	2.32	3.63	0.32
$\dot{\epsilon} = 5 \text{ s}^{-1}$	$E_{\text{MAR}}(\%)$	20.16	24.27	7.73	3.53	11.46	3.87
	$E_{\text{RMS}}(\text{MPa})$	20.73	25.95	7.83	4.02	12.91	5.84
six- $\dot{\epsilon}$	$E_{\text{MAR}}(\%)$	15.12	25.87	7.12	3.86	5.34	1.09
	$E_{\text{RMS}}(\text{MPa})$	12.36	18.24	4.43	2.68	6.23	2.40

le modèle PTM se classe en dernière position de ce classement avec des valeurs de E_{MAR} et E_{RMS} supérieures à 10% pour la vitesse de déformation de $\dot{\epsilon} = 5 \text{ s}^{-1}$. Les deux modèles, AR et RNA, donnent les meilleurs résultats en ce qui concerne les valeurs de E_{MAR} et E_{RMS} pour la vitesse de déformation de $\dot{\epsilon} = 5 \text{ s}^{-1}$.

Cette fois-ci, le modèle AR se comporte un peu mieux sur la vitesse de déformation de $\dot{\epsilon} = 5 \text{ s}^{-1}$ par rapport au modèle RNA, mais ce dernier donne globalement de meilleurs résultats sur l'ensemble du spectre des vitesses de déformation, avec des valeurs de $E_{\text{MAR}} = 1.09\%$ et $E_{\text{RMS}} = 2.40 \text{ MPa}$ respectivement.

Nous pouvons donc conclure de cette partie de l'étude que le modèle d'Arrhenius est le meilleur parmi les cinq modèles analytiques présentés, ce qui est en accord avec le fait qu'il est largement utilisé pour les simulations thermomécaniques de mise en forme. Cependant, l'approche du modèle RNA présente des avantages par rapport à l'approche du modèle d'Arrhenius, car dans l'ensemble, le modèle RNA est plus fidèle aux données expérimentales que le modèle d'Arrhenius, pour l'ensemble des vitesses de déformation. Plus de détails concernant ces analyses peuvent être trouvés dans [Tize Mha *et al.*, 2023].

3.5 Conclusions du chapitre

En conclusion, ce chapitre centré sur le développement d'un modèle de comportement à base de réseaux de neurones artificiels a donné des résultats prometteurs. Les résultats obtenus montrent que le remplacement des modèles classiques par des modèles basés sur les réseaux de neurones artificiels offre une capacité à simuler de manière fidèle et réaliste des phénomènes complexes.

La capacité des lois d'écoulement RNA à capturer des comportements complexes, non linéaires et dynamiques, sans nécessiter une formulation mathématique préalablement définie, représente une opportunité dans le domaine de la modélisation et de la simulation.

La finalité d'une loi de comportement étant son utilisation lors d'une simulation pour décrire le comportement du matériau, son implémentation dans le code élément finis fait l'objet du chapitre suivant.

Chapitre 4

Simulations numériques thermomécaniques

4.1	Implémentation d'une loi d'écoulement via une subroutine dans Abaqus	86
4.1.1	Implémentation numérique du modèle de Johnson–Cook	88
4.1.2	Implémentation du modèle de Zerilli–Armstrong Modifié (MZA)	90
4.1.3	Implémentation du modèle PTM	90
4.1.4	Implémentation du modèle RNA	91
4.2	Validation numérique du modèle RNA	92
4.2.1	Essai de rétrécissement d'une barre circulaire	92
4.2.2	Essai d'impact de Taylor	95
4.3	Simulation d'un essai de compression	97
4.3.1	Simulation basée sur le modèle de Johnson–Cook	97
4.3.2	Simulation basée sur le modèle MZA	98
4.3.3	Simulation basée sur le modèle PTM	99
4.3.4	Simulation basée sur le modèle RNA	99
4.3.5	Comparaison des modèles de comportement	100
4.4	Recristallisation dynamique	101
4.4.1	Modèle RNA pour le filtrage des données	102
4.4.2	Conditions critiques de la DRX	104
4.4.3	Cinétique de la DRX	113
4.5	Simulation numérique de la DRX	117
4.5.1	Données expérimentales	117
4.5.2	Modèle EF et implémentation de la DRX	119
4.5.3	Effet de la température et de la vitesse de déformation sur la DRX	120
4.5.4	Effet de la température et de la vitesse de déformation sur la taille de grain	122
4.6	Conclusions du chapitre	124

Dans les chapitres précédents, nous avons présenté et identifié les paramètres de cinq lois d'écoulement analytiques dans le chapitre 2 et d'une loi d'écoulement basée sur un réseau de neurones artificiel (RNA) dans le chapitre 3. Afin d'évaluer l'efficacité de ces modèles, nous allons maintenant mettre en œuvre des simulations numériques utilisant ces lois d'écoulement. Pour ce faire, nous allons implémenter ces lois d'écoulement en utilisant des routines VUHARD dans le code de calcul Abaqus.

La validation numérique de ces implémentations sera réalisée à travers des cas tests distincts. Tout

d'abord, nous allons effectuer deux simulations numériques sur le matériau 42CrMo4 dans le but de confirmer la validation de l'identification effectuée dans le chapitre 3. Ces simulations porteront sur un essai de rétrécissement d'une barre circulaire et un essai d'impact de Taylor, respectivement.

Ensuite, nous allons reproduire numériquement l'essai de compression du 3Cr2Mo réalisé au Chapitre 1 sur le simulateur thermomécanique Gleeble-3800. Nous allons également prendre en considération la recristallisation dynamique dans le processus, permettant ainsi d'observer l'évolution de la microstructure du matériau au cours de la simulation. Cette simulation sera basée sur un modèle de prédiction de la recristallisation développé en se basant sur un filtrage par un modèle RNA ; ce qui nous a permis de prédire les conditions critiques de cette recristallisation.

Ces simulations numériques sont réalisées pour évaluer et valider les modèles de comportement des matériaux que nous avons présenté dans les chapitres précédents. Elles permettent de confirmer l'efficacité de ces modèles dans la prédiction du comportement des matériaux dans diverses conditions d'essai, y compris la recristallisation dynamique.

4.1 Implémentation d'une loi d'écoulement via une sous-routine dans Abaqus

Les travaux réalisés dans le cadre de cette étude se basent sur l'utilisation du code de calcul Abaqus pour la simulation numérique. Seules certaines lois d'écoulement, celle de Johnson–Cook pour ce qui concerne cette étude, sont nativement implémentées dans Abaqus. Pour tout autre type de loi d'écoulement, il est possible pour l'utilisateur d'implémenter ses propres lois à travers des sous-routines en Fortran 77 comme VUMAT/VUHARD pour Abaqus/Explicit ou UMAT/UHARD pour Abaqus/Standard. Cette section présente, les étapes clés de l'implémentation d'une loi d'écoulement via une sous-routine VUHARD ou UHARD.

Si nous nous référons à l'organigramme général d'un code de calcul éléments finis (EF) tel qu'il est présenté sur la Figure 4.1, l'intégration de la loi d'écoulement concerne le calcul du tenseur des contraintes σ_1 à la fin d'un incrément, dans le rectangle jaune de cette même figure.

Dans le cadre d'une formulation EF en grandes déformations telle que celle utilisée dans la modélisation thermomécanique des procédés, ce calcul du tenseur des contraintes σ_1 est à effectuer sur tous les points d'intégration de chaque élément de la structure étudiée. Comme le modèle numérique peut comprendre des milliers d'éléments, eux-mêmes comprenant entre 1 et 8 points d'intégration selon les éléments utilisés, ce calcul des contraintes doit être le plus rapide possible afin de ne pas trop augmenter le temps CPU, mais suffisamment précis pour que les résultats soient conformes à la physique du procédé. Ceci est d'autant plus important si nous voulons intégrer cette loi d'écoulement dans un code EF explicite (Abaqus/Explicit) pour lequel une seconde de simulation physique correspond à plusieurs millions d'itérations de ces calculs de contraintes.

L'implémentation numérique d'une loi d'écoulement définie par l'utilisateur est réalisée ici dans un sous-programme VUHARD, similaire à celui proposé par van Rensburg et *al.* [Jansen van Rensburg et Kok, 2012], utilisé par le code Abaqus afin de permettre à un utilisateur de programmer le calcul de la contrainte d'écoulement σ et de ses 3 dérivées en fonction des données d'entrée du modèle : la déformation ε , la vitesse de déformation $\dot{\varepsilon}$ et la température T . Cette sous-routine est utilisée pour le calcul des contraintes σ_1 à la fin d'un incrément à partir du tenseur des contraintes au début de l'incrément σ_0 , les déformations, les paramètres du matériau et l'historique de la déformation à chaque point d'intégration de l'élément fini, selon l'algorithme d'intégration des contraintes basé sur la méthode de Retour–Radial tel que décrit dans Simo et *al.* [Simo et Hughes, 1998] pour les aspects généraux, Ming et *al.* [Ming et Pantalé, 2018] pour le code Abaqus/Explicit, ou Pantalé et *al.* [Pantalé et *al.*, 2004] pour le code de calcul DynELA.

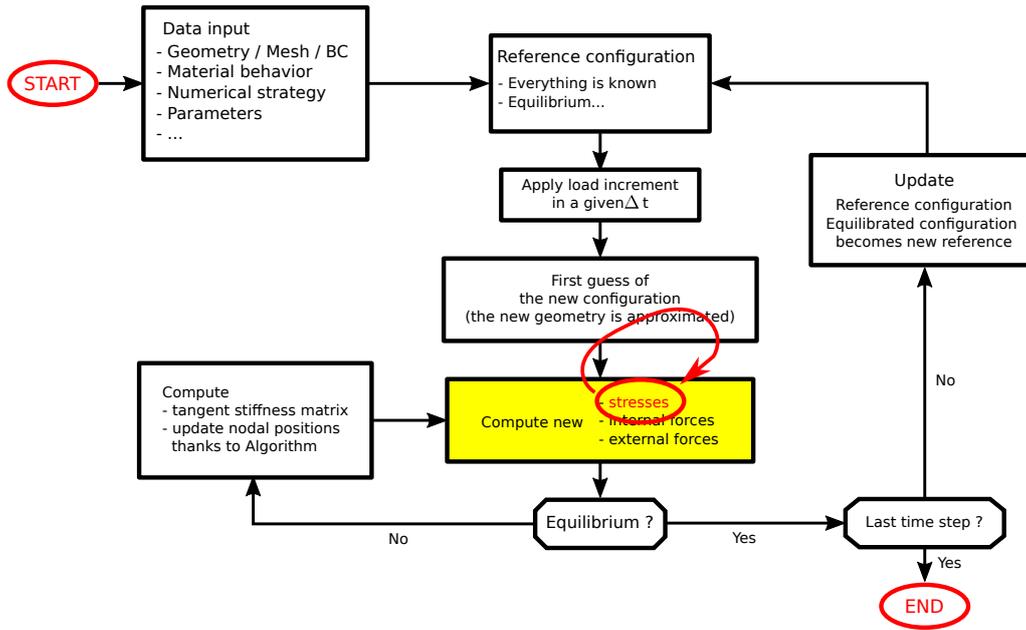


FIGURE 4.1 – Organigramme général d'un code EF (focus sur le calcul des contraintes à l'aide d'une procédure de résolution itérative).

Ainsi, sans entrer dans les détails du schéma d'intégration des contraintes utilisé dans les codes EF (le lecteur curieux peut se référer à [Ponthot, 2002, Pantalé *et al.*, 2004, Ming et Pantalé, 2018, Liang *et al.*, 2022] pour plus de détails sur cette méthode), la Figure 4.2 montre l'emplacement du sous-programme VUHARD utilisé pour calculer la contrainte d'écoulement σ et sa dérivée $\partial\sigma/\partial\Gamma$ utilisée dans l'écriture des deux quantités $\gamma(\Gamma)$ et $\dot{\gamma}'(\Gamma)$ utilisées dans la procédure de résolution de Newton-Raphson à partir de la relation suivante :

$$\frac{d\sigma}{d\Gamma} = \sqrt{\frac{2}{3}} \left(\frac{\partial\sigma}{\partial\varepsilon} + \frac{1}{\Delta t} \frac{\partial\sigma}{\partial\dot{\varepsilon}} + \frac{\eta\sigma}{\rho C_p} \frac{\partial\sigma}{\partial T} \right), \quad (4.1)$$

où Γ est le paramètre de consistance utilisé dans l'algorithme de Retour-Radial tel que défini par Simo *et al.*, Δt est l'incrément de temps, η est le coefficient de Taylor-Quinney définissant la quantité de travail plastique convertie en énergie thermique, C_p est le coefficient de chaleur spécifique, ρ est la densité du matériau et $\partial\sigma/\partial\varepsilon$, $\partial\sigma/\partial\dot{\varepsilon}$ et $\partial\sigma/\partial T$ sont les trois dérivées devant être implémentées dans la VUHARD. Ces dérivées partielles peuvent être calculées aussi bien analytiquement que numériquement selon la complexité de la formulation analytique de la loi d'écoulement :

- *Dérivation analytique* de la contrainte σ . Dans ce cas, il suffit de connaître la forme explicite de la contrainte $\sigma(\varepsilon, \dot{\varepsilon}, T)$ et en fonction de la complexité de la formulation analytique de la loi, ce calcul est plus ou moins simple.
- *Dérivation numérique* de la contrainte σ . La dérivée numérique se calcule en ajoutant respectivement des petits incréments $\Delta\varepsilon$ à la déformation ε , la vitesse de déformation $\dot{\varepsilon}$ et la température T comme illustré par l'équation suivante :

$$\begin{cases} \frac{\partial\sigma}{\partial\varepsilon} = \frac{\sigma(\varepsilon + \Delta\varepsilon, \dot{\varepsilon}, T) - \sigma(\varepsilon, \dot{\varepsilon}, T)}{\Delta\varepsilon} \\ \frac{\partial\sigma}{\partial\dot{\varepsilon}} = \frac{\sigma(\varepsilon, \dot{\varepsilon} + \Delta\dot{\varepsilon}, T) - \sigma(\varepsilon, \dot{\varepsilon}, T)}{\Delta\dot{\varepsilon}} \\ \frac{\partial\sigma}{\partial T} = \frac{\sigma(\varepsilon, \dot{\varepsilon}, T + \Delta T) - \sigma(\varepsilon, \dot{\varepsilon}, T)}{\Delta T} \end{cases} . \quad (4.2)$$

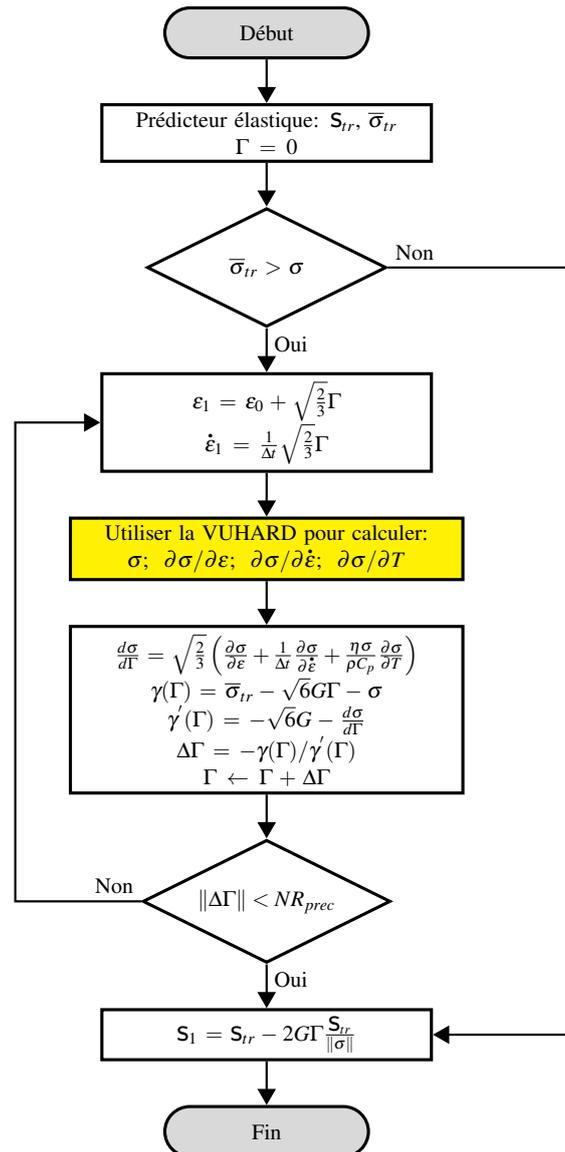


FIGURE 4.2 – Organigramme général de l’algorithme de Retour-Radial pour calculer le tenseur de contrainte final σ_1 .

La précision des résultats dépend fortement du choix des trois incréments $\Delta\varepsilon$, $\Delta\dot{\varepsilon}$ et ΔT . Selon les critères de convergence de l’algorithme de Newton Raphson, les valeurs préconisées de $\Delta\Gamma$ sont de 10^{-4} et 10^{-8} pour la simple précision et la double précision respectivement [Ming et Pantalé, 2018].

4.1.1 Implémentation numérique du modèle de Johnson–Cook

Le modèle de Johnson–Cook présenté dans la Section 2.2.2.1 page 31 est largement intégré dans des logiciels de simulation par éléments finis comme Abaqus. Il peut être utilisé nativement, après identification des paramètres du modèle comme proposé au paragraphe 2.4.1 page 44 et l’utilisateur n’a généralement pas besoin de réimplémenter ce modèle pour ses simulations. Néanmoins, dans ce paragraphe, nous présentons les équations permettant la ré-implémentation du modèle d’écoulement de Johnson–Cook afin de valider l’approche proposée pour l’implémentation numérique des lois d’écoulement.

La simplicité du modèle de Johnson–Cook découle de sa formulation découlée des effets de la déformation ε , de la vitesse de déformation $\dot{\varepsilon}$ et de la température T traduite par l'équation suivante :

$$\sigma = \sigma_\varepsilon(\varepsilon) \times \sigma_{\dot{\varepsilon}}(\dot{\varepsilon}) \times \sigma_T(T). \quad (4.3)$$

- La dépendance de l'écrouissage $\sigma_\varepsilon(\varepsilon)$ est donnée par :

$$\sigma_\varepsilon(\varepsilon) = A + B\varepsilon^n, \quad (4.4)$$

où les paramètres A , B et n décrivent la limite élastique, l'écrouissage du matériau et l'exposant respectivement.

- La dépendance à la vitesse de déformation $\sigma_{\dot{\varepsilon}}(\dot{\varepsilon})$ est formulée comme suit :

$$\sigma_{\dot{\varepsilon}}(\dot{\varepsilon}) = 1 + \ln C \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right), \quad (4.5)$$

avec $\dot{\varepsilon}$ la vitesse de déformation, $\dot{\varepsilon}_0$ la vitesse de déformation de référence, C le coefficient de sensibilité à la vitesse de déformation et $\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0}$ le rapport de vitesse de déformation qui est adimensionnel. Pour son implémentation via la routine VHUARD, cette dépendance de la vitesse de déformation sera modifiée comme c'est le cas dans les travaux de [Johnson et Cook, 1983, Zaera et Fernández-Sáez, 2006, Schwer, 2007] selon l'équation suivante :

$$\begin{cases} \sigma_{\dot{\varepsilon}}(\dot{\varepsilon}) = 1 + C \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right) & \text{si } \dot{\varepsilon} \geq \dot{\varepsilon}_0 \\ \sigma_{\dot{\varepsilon}}(\dot{\varepsilon}) = 1 & \text{si } \dot{\varepsilon} < \dot{\varepsilon}_0 \end{cases}. \quad (4.6)$$

En pratique, dans le logiciel Abaqus, le modèle tel qu'il est implémenté ne prend pas en compte cette modification de la formulation. Il peut se présenter des situations où la vitesse de déformation $\dot{\varepsilon}$ est inférieure à la vitesse de déformation de référence $\dot{\varepsilon}_0$. Dans ce contexte, cette reformulation s'avère nécessaire car $\ln(\dot{\varepsilon}/\dot{\varepsilon}_0) < 0$, ce qui entraîne une contrainte en deçà de la valeur admissible en conditions quasi-statiques.

- La dépendance à la température $\sigma_T(T)$ est définie de telle sorte que si $T < T_0$, il n'y a aucun effet de la température sur la contrainte et si $T \geq T_m$ le matériau est considéré dans un état liquide et la contrainte d'écoulement est nulle.

$$\begin{cases} \sigma_T(T) = 1 - \left(\frac{T - T_0}{T_m - T_0} \right)^m & \text{si } T_0 \leq T \leq T_m \\ \sigma_T(T) = 1 & \text{si } T < T_0 \\ \sigma_T(T) = 0 & \text{si } T > T_m \end{cases}, \quad (4.7)$$

avec T_0 et T_m les températures de référence et de fusion du matériau respectivement, m est l'exposant de l'effet de l'adoucissement et le terme $\left(\frac{T - T_0}{T_m - T_0} \right)^m$ est le rapport de température qui est un terme adimensionnel caractérisant l'adoucissement du matériau. Cette formulation particulière de la température est implémentée nativement dans le code de calcul Abaqus.

Compte tenu de ces définitions complémentaires, la dérivée partielle de la contrainte d'écoulement définie par le modèle de Johnson–Cook par rapport à la déformation $\frac{\partial \sigma}{\partial \varepsilon}$ est donnée par l'équation :

$$\frac{\partial \sigma}{\partial \varepsilon} = \left(nB\varepsilon^{n-1} \right) \times \sigma_{\dot{\varepsilon}}(\dot{\varepsilon}) \times \sigma_T(T). \quad (4.8)$$

Comme la dépendance à la vitesse de déformation $\dot{\varepsilon}$ et la dépendance à la température sont subdivisées en deux ou trois parties distinctes, il s'en suit que les dérivées par rapport à la vitesse de

déformation $\dot{\varepsilon}$ et à la température T sont également définies par parties selon les deux équations suivantes :

$$\begin{cases} \frac{\partial \sigma}{\partial \dot{\varepsilon}} = \frac{C(A + B\varepsilon^n)}{\dot{\varepsilon}} \sigma_T(T) & \text{si } \dot{\varepsilon} \geq \dot{\varepsilon}_0 \\ \frac{\partial \sigma}{\partial \dot{\varepsilon}} = 0 & \text{si } \dot{\varepsilon} < \dot{\varepsilon}_0 \end{cases}, \quad (4.9)$$

pour la dépendance à la vitesse de déformation $\dot{\varepsilon}$, et :

$$\begin{cases} \frac{\partial \sigma}{\partial T} = \frac{-m(A + B\varepsilon^n)}{T - T_0} \sigma_{\dot{\varepsilon}}(\dot{\varepsilon}) \left(\frac{T - T_0}{T_m - T_0} \right)^m & \text{si } T_0 \leq T \leq T_m \\ \frac{\partial \sigma}{\partial T} = 0 & \text{si } T \text{ ailleurs} \end{cases}, \quad (4.10)$$

pour la dépendance à la température T . L'ensemble de ces équations est alors implémenté numériquement sous forme d'une sous-routine Fortran VUHARD comme reporté dans l'Annexe A page 149.

4.1.2 Implémentation du modèle de Zerilli–Armstrong Modifié (MZA)

Comme mentionné au paragraphe 2.2.4.3, page 40, le modèle Zerilli–Armstrong Modifié (MZA) est une extension du modèle Zerilli–Armstrong. Cette forme modifiée du modèle de Zerilli–Armstrong a servi de base pour le développement du modèle PTM présenté au paragraphe 2.3. A partir de la définition du modèle MZA donnée par l'équation (2.40), les dérivées de la contrainte d'écoulement sont données comme suit :

$$\begin{cases} \frac{\partial \sigma}{\partial \varepsilon} = \left(nC_2\varepsilon^{n-1} - C_4(T - T_0)(C_1 + C_2\varepsilon^n) \right) \exp(C_r) \\ \frac{\partial \sigma}{\partial \dot{\varepsilon}} = (C_1 + C_2\varepsilon^n)(C_5 + C_6(T - T_0)) \frac{\exp(C_r)}{\dot{\varepsilon}} \\ \frac{\partial \sigma}{\partial T} = (C_1 + C_2\varepsilon^n) \left[-(C_3 + C_4\varepsilon) + C_6 \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right) \right] \exp(C_r) \end{cases}, \quad (4.11)$$

avec :

$$C_r = -(C_3 + C_4\varepsilon)(T - T_0) + (C_5 + C_6(T - T_0)) \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right).$$

L'ensemble de ces équations est alors implémenté numériquement sous forme d'une sous-routine Fortran VUHARD similaire à celle présentée dans l'Annexe A (seules les équations implémentées diffèrent ici par rapport à la loi d'écoulement de Johnson–Cook).

4.1.3 Implémentation du modèle PTM

Le modèle PTM, présenté dans la Section 2.3 page 42 est une forme généralisée du modèle de Zerilli–Armstrong Modifié offrant une couverture étendue de matériaux grâce à sa grande flexibilité. Ce qui caractérise ce modèle est sa capacité à être utilisé aussi bien dans des situations statiques que dynamiques. la mise en œuvre de l'implémentation numérique de ce modèle nécessite l'écriture des trois dérivées partielles de la contraintes par rapport à la déformation ε , la vitesse de déformation $\dot{\varepsilon}$ et la température T . Compte tenu de la formulation du modèle donnée par l'équation (2.44) page 43, ces dérivées sont alors données comme suit :

$$\left\{ \begin{array}{l} \frac{\partial \sigma}{\partial \varepsilon} = \left\{ \left(\sum_{i=1}^q i A_i \varepsilon^{i-1} \right) + \left(\sum_{i=0}^q A_i \varepsilon^i \right) \left[\left(\sum_{j=1}^r j B_j \varepsilon^{j-1} \right) (T - T_0) \right. \right. \\ \left. \left. + \left(\sum_{k=0}^s \left(\sum_{l=1}^t l C_k^l \varepsilon^{l-1} \right) (T - T_0)^k \right) \ln \left(\frac{\dot{\varepsilon}^p}{\dot{\varepsilon}_0} \right) \right] \right\} \exp(C_{mf}) \\ \frac{\partial \sigma}{\partial \dot{\varepsilon}} = \left(\sum_{i=0}^q A_i \varepsilon^i \right) \left[\sum_{k=0}^s \left(\sum_{l=0}^t C_k^l \varepsilon^l \right) (T - T_0)^k \right] \frac{\exp(C_{mf})}{\dot{\varepsilon}} \\ \frac{\partial \sigma}{\partial T} = \left(\sum_{i=0}^q A_i \varepsilon^i \right) \left\{ \left(\sum_{j=0}^r B_j \varepsilon^j \right) + \left[\sum_{k=1}^s k \left(\sum_{l=0}^t C_k^l \varepsilon^l \right) (T - T_0)^{k-1} \right] \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right) \right\} \exp(C_{mf}) \end{array} \right. , \quad (4.12)$$

avec :

$$C_{mf} = \left(\sum_{j=0}^r B_j \varepsilon^j \right) (T - T_0) + \left(\sum_{k=0}^s \left(\sum_{l=0}^t C_k^l \varepsilon^l \right) (T - T_0)^k \right) \ln \left(\frac{\dot{\varepsilon}}{\dot{\varepsilon}_0} \right). \quad (4.13)$$

L'ensemble des ces équations est alors implémenté numériquement sous forme d'une sous-routine Fortran VUHARD comme reporté dans l'Annexe B page 155.

4.1.4 Implémentation du modèle RNA

Comme présenté dans les sections précédentes, l'implémentation numérique d'un modèle RNA repose sur deux parties principales correspondant au calcul de la contrainte σ selon l'équation (3.20), page 71 et les 3 dérivées $\partial\sigma/\partial\varepsilon$, $\partial\sigma/\partial\dot{\varepsilon}$ et $\partial\sigma/\partial T$ selon l'équation (4.16), page 92.

Si nous voulons implémenter un réseau à 2 couches cachées avec une fonction d'activation sigmoïde (sig) contenant m neurones dans la couche 1 et n neurones dans la couche 2, nous devons utiliser l'équation (3.25) page 73 pour le calcul des dérivées. Afin d'optimiser l'implémentation numérique, et, comme le calcul de la contrainte d'écoulement et de ses dérivées partagent certains termes communs qui pourront être stockés pendant le calcul pour une utilisation ultérieure, le calcul est divisé en plusieurs sous-termes $\vec{z}^{(1)}$ à $\vec{z}^{(4)}$:

1. On calcule les termes internes du réseau pour en déduire la dérivée du réseau par rapport à l'entrée \vec{x} .

$$\left\{ \begin{array}{l} \vec{z}^{(1)} = \exp \left(-w^{(1)} \cdot \vec{x} - \vec{b}^{(1)} \right) \\ \vec{z}^{(2)} = \exp \left(w^{(2)} \cdot \frac{1}{1 + \vec{z}^{(1)}} + \vec{b}^{(2)} \right) \\ \vec{z}^{(3)} = \vec{w} \circ \frac{\vec{z}^{(2)}}{(1 + \vec{z}^{(2)})^2} \\ \vec{z}^{(4)} = \frac{\vec{z}^{(1)}}{(1 + \vec{z}^{(1)})^2} \end{array} \right. . \quad (4.14)$$

2. Ensuite, à partir des deux termes $\vec{z}^{(3)}$ et $\vec{z}^{(4)}$, on peut calculer les trois dérivées de la sortie s par rapport \vec{x} contenu dans le vecteur \vec{s}' selon l'équation suivante :

$$\vec{s}' = w^{(1)T} \cdot \left[\left(w^{(2)T} \cdot \vec{z}^{(3)} \right) \circ \vec{z}^{(4)} \right]. \quad (4.15)$$

3. Enfin, de l'équation (4.15) et conformément au principe de normalisation des entrées on obtient alors les trois dérivées de la contrainte d'écoulement σ en utilisant l'équation (4.16) :

$$\begin{cases} \partial\sigma/\partial\varepsilon = s'_1 \frac{[\sigma]_{max} - [\sigma]_{min}}{[\varepsilon]_{max} - [\varepsilon]_{min}} \\ \partial\sigma/\partial\dot{\varepsilon} = \frac{s'_2}{\dot{\varepsilon}} \frac{[\sigma]_{max} - [\sigma]_{min}}{[\dot{\varepsilon}]_{max} - [\dot{\varepsilon}]_{min}} \\ \partial\sigma/\partial T = s'_3 \frac{[\sigma]_{max} - [\sigma]_{min}}{[T]_{max} - [T]_{min}} \end{cases} \quad (4.16)$$

Un programme en Python permet l'écriture automatique de la sous-routine Fortran VUHARD à partir des paramètres des réseaux de neurones. Cette approche permet de supprimer les erreurs de transcription du modèle et de faciliter le lien entre l'identification du modèle et son utilisation dans un code de calcul EF. Pour plus de détails sur cette implémentation, et sa mise en œuvre, on pourra se référer à Pantalé et *al.* [Pantalé et *al.*, 2022, Pantalé, 2023, Pantalé, 2021].

L'ensemble de ces équations est alors implémenté numériquement sous forme d'une sous-routine Fortran VUHARD comme reporté dans l'Annexe C page 163. Maintenant que l'implémentation numérique de la loi d'écoulement RNA est effectuée, il est temps d'analyser les résultats des simulations numériques qui en découlent. Ceci fait l'objet de la prochaine Section de ce chapitre.

4.2 Validation numérique du modèle RNA

Conformément à la validation effectuée dans la Section 3.3 page 73, nous procédons dans cette section à la simulation numérique de deux essais basés sur l'utilisation du modèle RNA pour la description du comportement plastique du matériau 42CrMo4. Il s'agit essentiellement d'un essai de rétrécissement d'une barre circulaire et d'un essai d'impact de Taylor. Le RNA utilisé dans cette section ayant été développé afin de reproduire le comportement d'un acier 42CrMo4 via la loi d'écoulement de Johnson–Cook, il va donc être possible ici de comparer directement les résultats obtenus pour ces deux simulations numériques avec ceux obtenus nativement par le code de calcul Abaqus.

Deux architectures de RNA vont être utilisées pour ces comparaisons. Elles correspondent aux dénominations 3-15-7-1-sig et 3-7-4-1-sig, et diffèrent par le nombre de neurones contenus dans les couches cachées.

4.2.1 Essai de rétrécissement d'une barre circulaire

L'essai de rétrécissement d'une barre circulaire, tel que proposé par Ponthot et *al.* [Ponthot, 2002], est utile pour évaluer la performance des lois constitutives non linéaires. Un quart de modèle axisymétrique, tel que proposé par Ming et *al.* [Ming et Pantalé, 2018], est utilisé pour cette simulation. Les dimensions de l'éprouvette sont présentées sur la Figure 4.3. Nous avons imposé un déplacement total de 7 mm le long de l'axe \vec{z} sur le côté gauche de l'éprouvette tandis que le déplacement radial du même bord est nul. Du côté opposé, le déplacement axial est limité tandis que le déplacement radial est libre.

Le maillage est composé de 400 éléments CAX4RT (quadrilatère bilinéaire à 4 nœuds à intégration réduite et couplage température–déplacement) avec une zone raffinée de 200 éléments sur le côté droit sur 1/3 de la longueur totale. Le modèle EF est un modèle explicite couplé température–déplacement (il s'agit d'une analyse couplée température–contrainte où les solutions de transfert

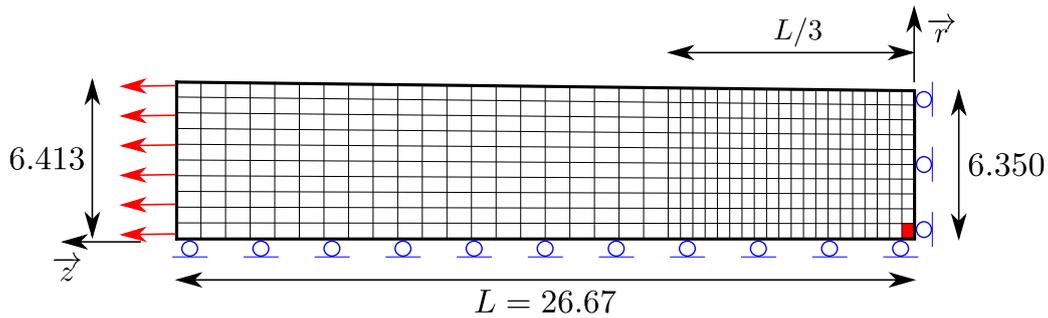


FIGURE 4.3 – Modèle numérique pour le rétrécissement d’une barre circulaire

thermique et mécanique sont obtenues simultanément par un couplage explicite), et le temps total de simulation est fixé à $t = 0.01$ s.

La Figure 4.4 montre la contrainte équivalente de von Mises $\bar{\sigma}$ de la barre déformée pour deux modèles différents : le modèle intégré dans le logiciel Abaqus (partie supérieure) et le modèle RNA 3-15-7-1-sig (partie inférieure). Il existe une très faible différence entre les résultats des deux modèles en termes de distribution spatiale de la contrainte et de valeur maximale.

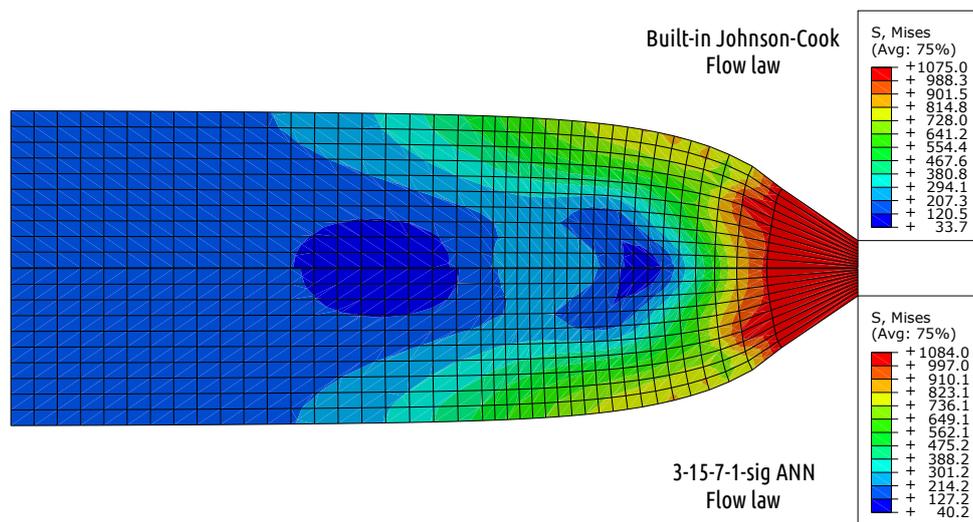


FIGURE 4.4 – Contrainte de von Mises $\bar{\sigma}$ pour le rétrécissement d’une barre circulaire pour une élongation de 7 mm, RNA 3-15-7-1-sig.

La contrainte maximale étant située au centre de la barre, nous avons choisi de représenter sur la Figure 4.5 l’évolution de la déformation équivalente $\bar{\epsilon}$ et de la contrainte équivalente de von Mises $\bar{\sigma}$ pour l’élément central de l’échantillon (l’élément rouge dans le coin inférieur droit de la Figure 4.3). Comme montré sur la Figure 4.5, le modèle de comportement de Johnson–Cook disponible dans Abaqus, son implémentation via la routine VUHARD basée sur sa forme analytique tel que présenté dans Ming et *al.* [Ming et Pantalé, 2018], et les deux versions du modèle RNA donnent presque les mêmes résultats, sauf dans la zone où l’allongement est supérieur à 6 mm, pour laquelle la contrainte équivalente de von Mises diffère entre les modèles RNA et les modèles Johnson–Cook intégré et analytique, comme nous le verrons plus loin.

Ceci est également confirmé dans le Tableau 4.1 rapportant une comparaison des quatre modèles pour deux valeurs du déplacement (3.5 mm et 7 mm, nommés respectivement *mid* et *end*). Par la suite, nous pouvons voir que la déformation équivalente $\bar{\epsilon}$, la contrainte équivalente de von Mises $\bar{\sigma}$ et la température T obtenues avec les deux modèles RNA sont très proches de celles obtenues par le

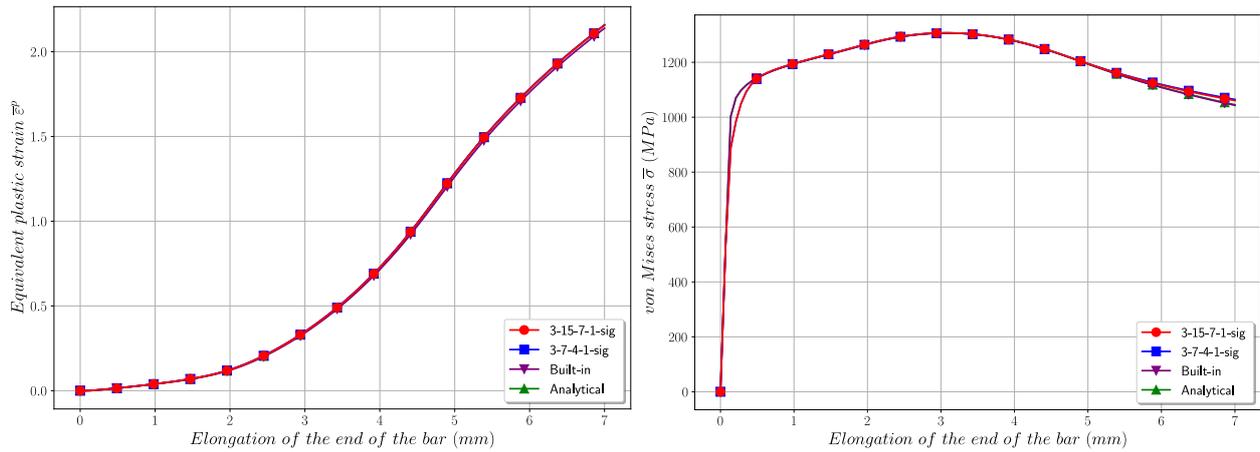


FIGURE 4.5 – Déformation équivalente $\bar{\epsilon}$ et contrainte de von Mises $\bar{\sigma}$ vs. déplacement axial

TABLE 4.1 – Comparaison des résultats de rétrécissement d'une barre circulaire pour un déplacement de 3.5 mm (*mid*) et 7 mm (*end*)

Modèle	Incr.	Time (s)	$\bar{\epsilon}_{mid}$	$\bar{\sigma}_{mid}$ (MPa)	T_{mid} (°C)	$\bar{\epsilon}_{end}$	$\bar{\sigma}_{end}$ (MPa)	T_{end} (°C)
3-7-4-1-sig	191 768	29.98	0.51	1293.81	182.01	2.16	1064.43	587.78
3-15-7-1-sig	194 432	38.54	0.51	1293.97	181.52	2.03	1060.50	587.29
Analytical	200 145	35.50	0.51	1293.59	182.47	2.16	1045.75	585.85
Built-In	199 474	28.71	0.51	1293.76	180.36	2.14	1043.19	587.66

modèle intégré et le modèle analytique pour le déplacement *mid*, alors qu'elles diffèrent légèrement pour le déplacement *end*. Cet ensemble de résultats nous permet de valider la modélisation proposée.

Un résultat intéressant concerne les valeurs de la déformation équivalente $\bar{\epsilon}$ et les températures T rapportées dans le Tableau 4.1 pour le déplacement *end*. On peut voir que, pour le déplacement *end*, les valeurs de la déformation rapportées dans le Tableau 4.1 sont de l'ordre de $\epsilon = 2.1$ alors que la plage de déformation utilisée pour l'entraînement du modèle est de $\epsilon \in [0, 1]$. La valeur maximale de la température T est d'environ 587 °C alors que la plage d'apprentissage est de $T \in [20 \text{ °C}, 500 \text{ °C}]$. Le modèle proposé est donc capable d'extrapoler avec une bonne précision certaines données hors de la plage d'apprentissage.

Il est évident, d'après le Tableau 4.1 et la Figure 4.5, que tous les modèles donnent des résultats très proches lorsque les paramètres sont dans la plage d'apprentissage, alors que les modèles RNA diffèrent légèrement lorsque les paramètres sont loin de la plage d'apprentissage (ici, avec une élongation supérieure à 6 mm de sorte que $\bar{\epsilon} > 1.7$). Cela montre que notre modèle RNA est capable, dans une certaine mesure, et avec les précautions d'usage dans la mesure où les réseaux de neurones sont généralement relativement fidèles pour l'interpolation mais moins pour l'extrapolation, de généraliser un comportement hors du domaine d'apprentissage. Si nous voulons réduire l'écart entre les valeurs prédites et les valeurs réelles en dehors de la plage d'apprentissage, nous devons élargir les plages d'apprentissage des variables d'entrée si les données sont disponibles.

Le Tableau 4.1 rapporte également les temps de calcul et le nombre total d'incrément calculés après avoir exécuté le même modèle 10 fois. On peut voir que le temps de calcul du modèle 3-7-4-1-sig est

équivalent à celui du modèle intégré et inférieur à celui du modèle analytique, alors qu'il augmente pour le RNA le plus complexe (3-15-7-1-sig). Dans l'approche proposée, les réseaux de neurones sont utilisés pour calculer la contrainte d'écoulement du matériau dans un algorithme de type Retour-Radial. Il remplace l'évaluation de la contrainte d'écoulement et de ses dérivés sur la base de certaines expressions analytiques.

L'implémentation de la VUHARD brise l'algorithme naturel optimisé intégré utilisé pour calculer les contraintes par des appels des routines Fortran, des transferts de données, entraînant une augmentation du temps CPU (voir la comparaison entre les temps CPU intégré et analytique). C'est pourquoi nous ne pouvons pas avoir une réduction du temps de calcul par rapport au modèle intégré. Nous devons donc comparer la performance CPU des modèles RNA avec le modèle analytique. Comme la loi de comportement analytique de Johnson-Cook est assez simple, le modèle RNA le plus complexe est plus lent que le modèle analytique. Mais, dans le cas d'une loi de comportement plus complexe, comme certaines lois de Johnson-Cook modifiées proposées par Zhou et *al.* [Zhou et al., 2020], cette tendance s'inversera puisque l'évaluation analytique des dérivées nécessaires au fonctionnement de l'algorithme de Retour-Radial devient assez gourmande en ressources CPU. Nous pouvons également conclure des résultats présentés dans le Tableau 4.1 que le modèle 3-7-4-1-sig est suffisant pour obtenir des résultats valables avec des temps CPU comparables par rapport à la loi constitutive intégrée.

4.2.2 Essai d'impact de Taylor

La performance de la routine VUHARD pour le modèle RNA proposée va maintenant être validée à une vitesse de déformation élevée avec la simulation de l'essai d'impact de Taylor [Taylor, 1946] où une éprouvette cylindrique impacte une cible rigide avec une vitesse initiale $V_c = 287$ m/s. La hauteur du cylindre est de $L = 32.4$ mm et le rayon de $r = 3.2$ mm comme indiqué sur la Figure 4.6. Le déplacement axial est limité sur le côté gauche de l'éprouvette tandis que le déplacement radial est libre (pour signifier un contact parfait sans friction du projectile sur la cible). Un quart de modèle 3D de l'éprouvette cylindrique de Taylor est maillé avec 4 455 éléments C3D8T (déplacement et température trilineaires à 8 nœuds). Le temps de simulation total pour l'essai d'impact de Taylor est de $t = 80$ μ s.

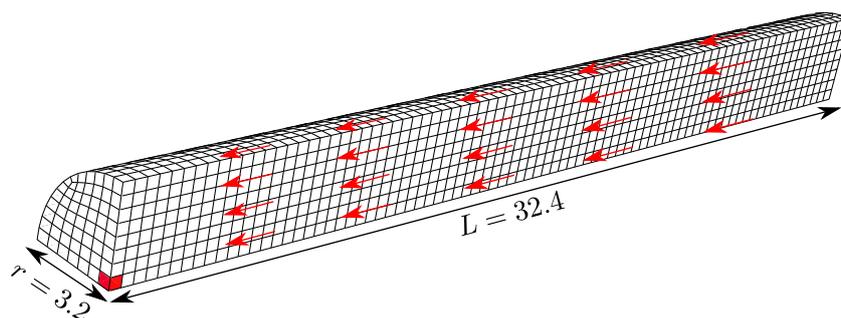


FIGURE 4.6 – Modèle numérique d'essai d'impact de Taylor

La Figure 4.7 montre le tracé de la déformation équivalente de du cylindre déformé pour deux modèles : le modèle Built-In (côté supérieur de l'image) et le modèle RNA-3-15-7-1-sig (côté inférieur de l'image). Les distributions de la déformation équivalente sont presque les mêmes pour les deux modèles. La déformation équivalente maximale $\bar{\epsilon}$ est située dans l'élément central du modèle (l'élément rouge sur la Figure 4.6). Les modèles donnent à peu près la même valeur que celle indiquée dans le Tableau 4.2 pour $\bar{\epsilon}$, T et les dimensions finales de l'éprouvette L_f (longueur finale) et D_f (diamètre final de la face d'impact).

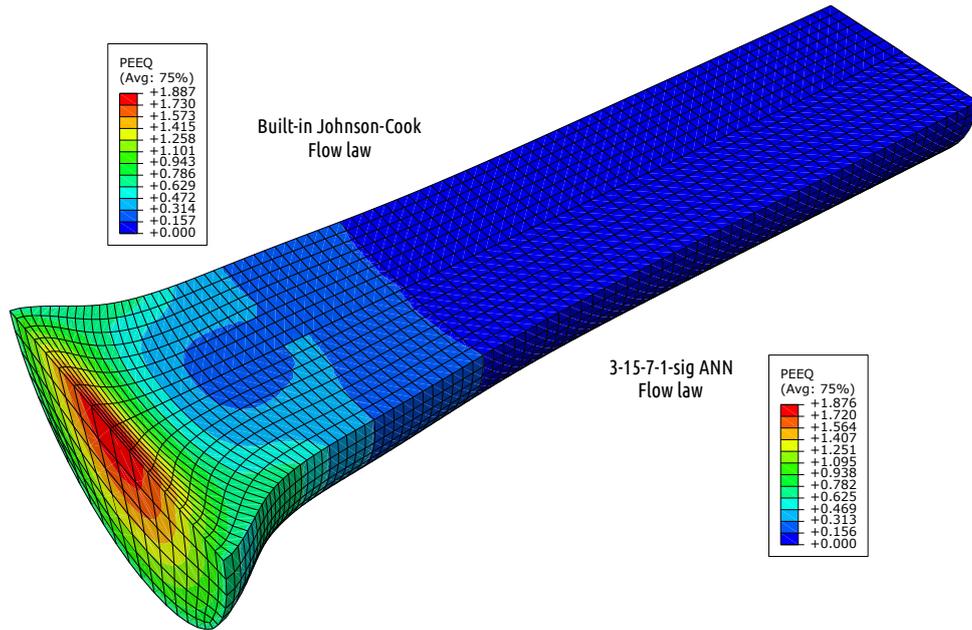


FIGURE 4.7 – Déformation équivalente $\bar{\epsilon}$ pour l'essai d'impact de Taylor, RNA 3-15-7-1-sig

Concernant les temps de simulation rapportés dans le Tableau 4.2, les mêmes tendances que celles présentées dans la Section 4.2.1 sont perceptibles dans ce cas d'essai. Encore une fois, la comparaison

TABLE 4.2 – Comparaison des résultats d'impact de Taylor 3D

Modèle	Incr.	Time (s)	L_f (mm)	D_f (mm)	T (°C)	$\bar{\epsilon}$
3-7-4-1-sig	6 344	63.08	26.52	11.18	584.28	1.83
3-15-7-1-sig	6 239	90.32	26.52	11.17	584.47	1.83
Analytical	6 419	71.71	26.53	11.19	585.64	1.84
Built-In	6 570	52.82	26.54	11.21	588.66	1.84

des résultats numériques valide l'approche proposée et montre un très bon niveau de corrélation des résultats.

4.3 Simulation d'un essai de compression

Dans l'optique d'évaluer l'efficacité de l'implémentation des modèles présentés dans les sections précédentes, un essai de compression est simulé dans cette Section. Le matériau retenu pour cette simulation est l'acier 3Cr2Mo présenté au Chapitre 1. Un modèle axisymétrique de longueur $L = 15$ mm et de diamètre $D = 10$ mm est utilisé comme illustré sur la Figure 4.8. Le maillage est

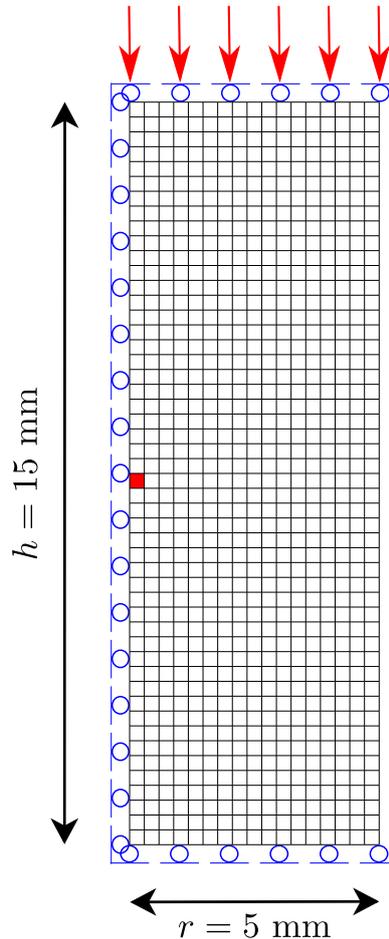


FIGURE 4.8 – Modélisation numérique de l'essai de compression du P20

composé de 986 éléments CAX4RT. Le modèle EF est simulé sur Abaqus/Explicit avec une cinétique correspondant à une vitesse de déformation $\dot{\epsilon} = 1 \text{ s}^{-1}$. Nous avons imposé un déplacement de 7 mm de la partie haute du cylindre durant la compression suivant l'axe vertical de l'éprouvette. Le déplacement radial u_r de la partie haute et de la partie basse de l'éprouvette est laissée libre. Le déplacement axial u_z de la partie haute est imposé et la partie inférieure du cylindre est fixe pendant la simulation. Une température initiale de 1050 °C est imposée à l'éprouvette.

4.3.1 Simulation basée sur le modèle de Johnson–Cook

Dans un premier modèle, nous utilisons le modèle de Johnson–Cook pour décrire le comportement du matériau. La Figure 4.9 montre la déformation équivalente ainsi que la contrainte de von Mises. On peut noter que la déformation maximale après un déplacement de 7 mm s'élève à $\epsilon = 1.15$, légèrement en dehors de la plage d'identification du modèle. Cette déformation maximale se localise au centre et aux bords de l'éprouvette. En ce qui concerne la distribution de la contrainte de von Mises, elle

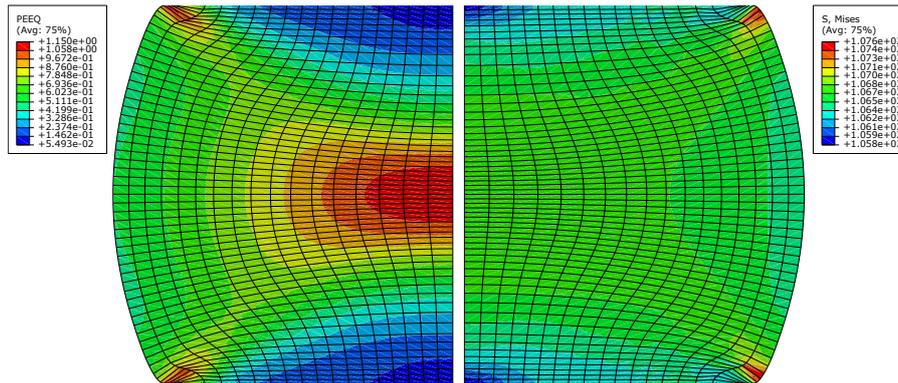


FIGURE 4.9 – Déformation équivalente et contrainte de von Mises – Modèle de Johnson–Cook

se révèle homogène à travers l'éprouvette, et sa valeur maximale atteint 107 MPa, en dessous de la valeur expérimentale d'environ 123 MPa (valeur issue des essais Gleeble pour $T = 1050\text{ °C}$ et $\dot{\epsilon} = 1\text{ s}^{-1}$). Cette disparité numérique confirme l'incapacité de ce modèle à prédire correctement le comportement de ce matériau, tel que défini lors de son identification dans le chapitre précédent.

Il est également important de noter que la déformation maximale se situe légèrement en dehors de la plage d'identification du modèle. Ainsi, pour une évaluation plus pertinente du modèle, il serait mieux de calibrer la simulation en ajustant la déformation maximale à environ $\epsilon = 0.7$. Toutefois, il est à noter que même avec un calibrage précis, les résultats de la simulation ne s'amélioreraient pas substantiellement, car le modèle en question ne convient pas au comportement de ce matériau.

4.3.2 Simulation basée sur le modèle MZA

Le modèle MZA présente des similitudes avec le modèle de Johnson–Cook en termes de formulation, mais diffère par la manière dont il prend en compte l'interdépendance entre la déformation ϵ , la vitesse de déformation $\dot{\epsilon}$ et la température T . Comme le modèle de Johnson–Cook, le modèle MZA a également été implémenté dans Abaqus en utilisant ses dérivées analytiques. La Figure 4.10 présente la déformation ainsi que la contrainte de von Mises résultant de la simulation numérique. En examinant cette figure et en considérant les valeurs maximales obtenues, on constate que la

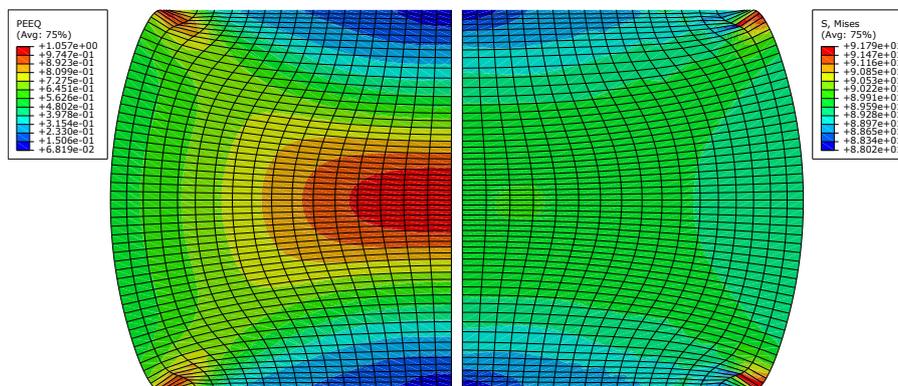


FIGURE 4.10 – Déformation équivalente et contrainte de von Mises – Modèle MZA

déformation atteint une valeur maximale de $\epsilon = 1.05$, comparée à la valeur de $\epsilon = 0.7$. Quant à la contrainte de von Mises, elle se situe autour de 92 MPa, en deçà de la valeur atteinte par le modèle de Johnson–Cook. Ces résultats confirment également que le modèle MZA n'est pas en

mesure de prédire de manière précise le comportement de ce matériau, similairement au modèle de Johnson–Cook.

4.3.3 Simulation basée sur le modèle PTM

Le modèle PTM se présente comme une généralisation du modèle MZA, puisqu'il est conçu pour surmonter les limitations associées à l'utilisation du modèle MZA. Dans le chapitre consacré à l'identification des modèles sélectionnés pour cette étude, il est établi que le modèle PTM se classe parmi ceux qui parviennent le mieux à décrire le comportement de ce matériau.

Comme pour les autres modèles, les valeurs maximales atteintes en termes de déformation et de contrainte de von Mises tournent autour de $\epsilon = 0.9$ pour la déformation et de 127 MPa pour la contrainte de von Mises (Figure 4.11). Cette dernière valeur n'est pas très éloignée de la valeur de 123 MPa observée expérimentalement. Une autre observation à souligner concernant ce modèle

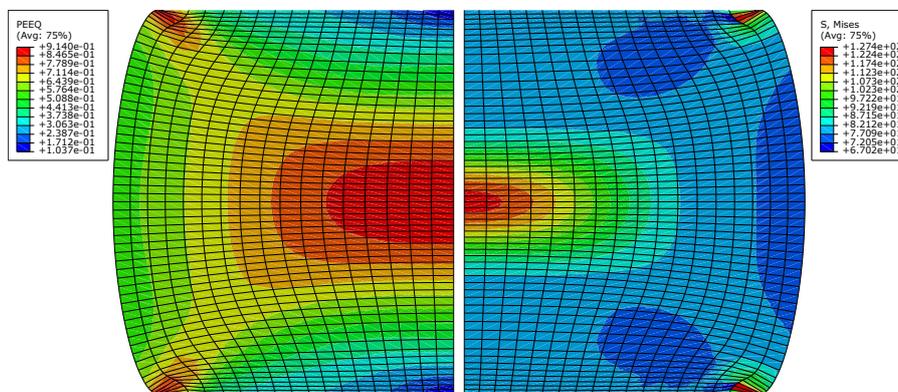


FIGURE 4.11 – Déformation équivalente et contrainte de von Mises – Modèle PTM

est la répartition de la contrainte de von Mises, qui présente une légère différence par rapport aux autres modèles. Ici, la valeur maximale de la contrainte se situe au centre de l'éprouvette.

4.3.4 Simulation basée sur le modèle RNA

Lors de la phase d'identification des modèles de comportement, le modèle RNA s'est avéré être celui qui était capable de prédire au mieux le comportement du matériau. Cette affirmation trouve une validation numérique dans les résultats présentés sur la Figure 4.12, où l'on constate que les valeurs maximales de la déformation et de la contrainte de von Mises sont d'environ $\epsilon = 0.8$ et 122 MPa respectivement. Ces valeurs sont proches de celles mesurées lors de l'identification du modèle, ce qui confirme que le modèle RNA reproduit de manière précise le comportement du matériau dans les simulations numériques.

Après avoir exposé les résultats de simulations obtenus grâce à chaque modèle, il est pertinent de procéder à une étude comparative plus approfondie de ces résultats. Pour ce faire, nous allons analyser de manière simultanée les réponses de ces modèles en nous concentrant sur un élément spécifique. Cette approche nous permettra d'examiner de près la performance de chaque modèle en effectuant une évaluation côte à côte.

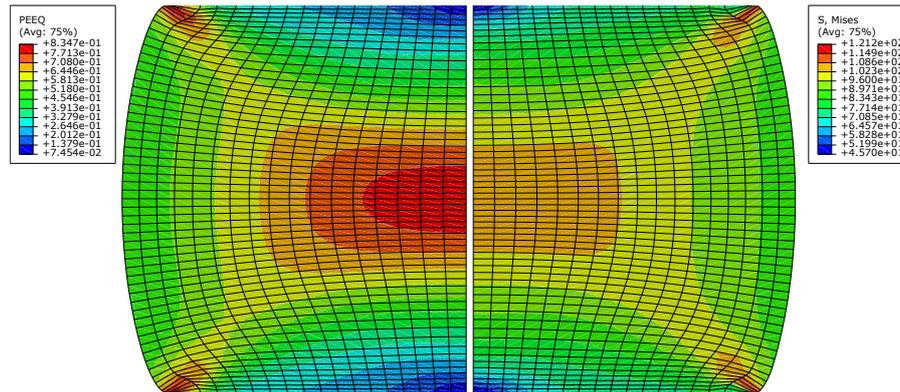


FIGURE 4.12 – Déformation équivalente et contrainte von Mises – Modèle RNA

4.3.5 Comparaison des modèles de comportement

Nous présentons ici, les résultats de plusieurs simulations réalisées avec tous les modèles pour trois températures (1050 °C, 1150 °C et 1250 °C), à une vitesse de déformation $\dot{\epsilon} = 1 \text{ s}^{-1}$. Toutes les simulations ont été menées dans les conditions précédemment détaillées pour chaque modèle. Pour effectuer cette comparaison, nous avons extrait les données de déformation et de contrainte de von Mises d'un élément localisé au centre de l'éprouvette (élément rouge situé au centre de l'éprouvette sur la Figure 4.8). Les résultats de chaque modèle sont ensuite comparés aux valeurs expérimentales correspondantes.

La Figure 4.13 présente une comparaison entre les simulations de tous les modèles et les valeurs expérimentales. L'analyse de cette figure montre que les modèles de Johnson–Cook et MZA ne par-

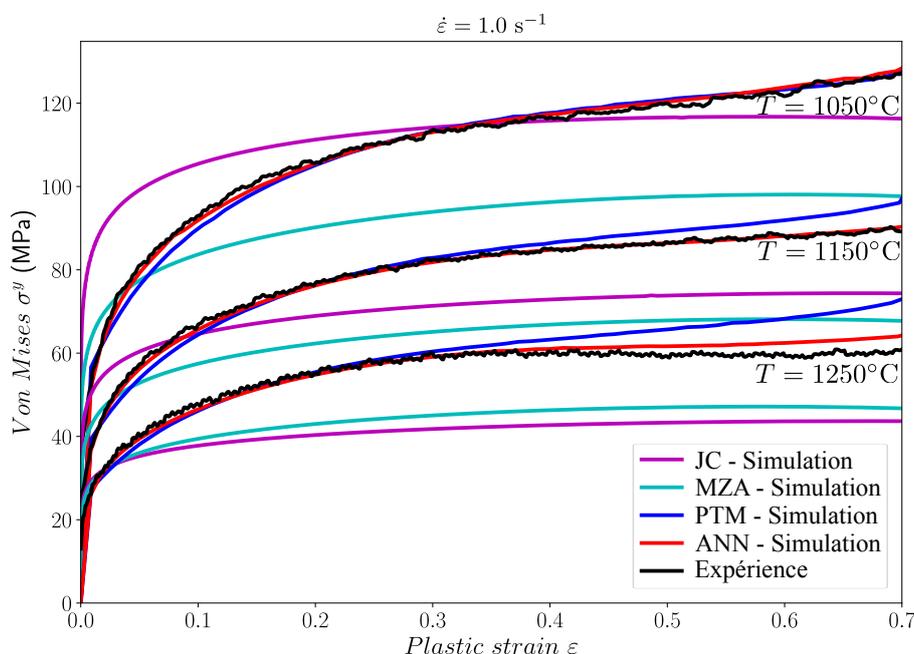


FIGURE 4.13 – Comparaison des prédictions numériques des modèles identifiés aux données expérimentales correspondantes

viennent pas à prédire avec précision le comportement du matériau pour les trois températures, ce qui confirme nos observations précédentes. En revanche, les modèles PTM et RNA présentent une excellente corrélation avec les valeurs expérimentales, validant ainsi les résultats de l'identification

réalisée dans les chapitres 2 et 3. Le Tableau 4.3 résume une comparaison entre les quatre modèles en termes de temps de simulation, d'incrément, de température maximale atteinte, de déformation et de contrainte de von Mises maximales obtenues.

TABLE 4.3 – Comparaison des résultats d'un essai de compression d'un cylindre soumis à un déplacement de 7 mm avec une température initiale de 1050 °C.

Modèles	Incréments	Temps	ε_{max}	σ_{max} (MPa)	T_{max} (°C)
JC	1,562,467	25 min 18 s	1.15	107.6	1080
MZA	1,525,368	22 min 15 s	1.058	91.79	1074
PTM	1,527,027	29 min 12 s	0.913	127.4	1080
RNA	1,496,160	40 min 55 s	0.835	121.2	1083

On peut noter que le modèle MZA présente le plus petit temps de simulation par opposition au modèle RNA, qui, en revanche, affiche le temps de simulation le plus long. Cette différence s'explique par le fait que le modèle RNA est beaucoup plus complexe dans sa formulation que le modèle MZA. En ce qui concerne les valeurs maximales de déformation et de contrainte atteintes, il est observé que les modèles RNA et PTM présentent des déformations proches de la valeur de $\varepsilon = 0.7$, qui représente la limite d'identification des modèles. Les autres modèles, en revanche, affichent des déformations autour de $\varepsilon = 1.15$.

Étant donné que le modèle RNA se révèle être le plus performant pour prédire numériquement les résultats expérimentaux, validant ainsi les observations faites lors de l'identification, nous nous concentrerons dans la suite de ce chapitre sur l'utilisation exclusive de ce modèle pour simuler l'évolution de la microstructure au cours de la simulation, en particulier en ce qui concerne la prédiction de la recristallisation dynamique et de la taille des grains.

4.4 Recristallisation dynamique

La recristallisation dynamique joue un rôle crucial dans la régulation des propriétés mécaniques des pièces forgées lors de la mise en forme à chaud. Sur la courbe de contrainte-déformation, ce phénomène se manifeste généralement par une phase d'adoucissement après un pic. Dans certains matériaux comme l'aluminium, la restauration dynamique peut contrebalancer l'écrouissage, entraînant rapidement l'apparition d'un plateau. Cependant, dans les aciers austénitiques, la restauration a une cinétique plus lente, ce qui signifie que la recristallisation dynamique peut nécessiter une accumulation critique de déformation pour se déclencher [McQueen et Jonas, 1975, Ebrahimi *et al.*, 2011a, Ebrahimi *et al.*, 2012, Schulz *et al.*, 2023, Buzolin *et al.*, 2023].

Étant donné que les propriétés d'un matériau sont étroitement liées à sa microstructure, qui est elle-même influencée par la recristallisation dynamique, il est essentiel de déterminer le début de ce processus. Cependant, divers facteurs tels que la composition chimique du matériau, la taille des grains avant déformation et les paramètres de la déformation (température, taux et valeurs de déformation) influencent la recristallisation dynamique. Celle-ci ne peut commencer qu'une fois que la déformation critique est atteinte, une valeur qui peut être identifiée par des techniques de métallographie. Cette méthode nécessite toutefois un échantillonnage intensif avant et après l'atteinte de la déformation critique. De plus, les transitions de phase qui se produisent lors du

refroidissement depuis la température de travail à chaud altèrent la structure déformée, ce qui rend la métallographie complexe. C'est pourquoi il est préférable d'utiliser des approches analytiques plus simples.

Dans cette optique, plusieurs études ont été menées pour déterminer analytiquement le début de la recristallisation dynamique. Une manière de procéder consiste à identifier la déformation critique nécessaire pour amorcer la recristallisation. Les premières investigations de Ryan et McQueen [Ryan et McQueen, 1989, Ryan et McQueen, 1990a, Ryan et McQueen, 1990b] ont permis d'observer l'apparition d'un point d'inflexion sur la courbe $\theta = \frac{\partial \sigma}{\partial \varepsilon} = f(\sigma)$ avant la contrainte pic σ_p . Ils ont associé ce point d'inflexion au début de la recristallisation dynamique, soit les conditions critiques. Par la suite, Poliak et Jonas [Poliak et Jonas, 1996, Jonas et Poliak, 2003, Poliak et Jonas, 2003a, Poliak et Jonas, 2003b] ont étayé cette hypothèse en démontrant que ce point d'inflexion était lié à une énergie thermodynamique libérée pendant le déplacement des dislocations.

Najafizadeh et Jonas [Najafizadeh et Jonas, 2006] ont simplifié le modèle thermodynamique de Poliak et Jonas en développant une technique analytique largement utilisée aujourd'hui pour déterminer les conditions critiques de déclenchement de la recristallisation dynamique. Leur approche implique l'utilisation d'un polynôme de troisième degré pour représenter la courbe $\theta = f(\sigma)$. En utilisant la seconde dérivée de cette courbe, ils ont pu identifier les valeurs critiques de contrainte et de déformation (σ_c et ε_c) nécessaires pour l'initiation de la recristallisation dynamique. Cependant, cette méthode présente des limites, ce qui a conduit au développement d'un nouveau modèle pour identifier les conditions critiques de recristallisation dynamique, objet des sections suivantes.

4.4.1 Modèle RNA pour le filtrage des données

L'érouissage est un phénomène mécanique qui se produit dans les matériaux lorsqu'ils subissent une déformation, c'est-à-dire une déformation permanente due au glissement des plans cristallins les uns par rapport aux autres. L'érouissage se traduit par une augmentation de la résistance mécanique du matériau au fur et mesure qu'il est déformé davantage et la relation qui traduit ce phénomène est donnée par $\frac{\partial \sigma}{\partial \varepsilon} = f(\sigma)$. Lorsqu'on essaie d'appliquer directement cette équation aux données expérimentales en utilisant la méthode de la dérivation numérique, les courbes obtenues ne sont pas toujours facilement interprétables.

Pour résoudre cette difficulté, il est courant d'adopter une approche consistant à trouver un polynôme capable d'approximer l'évolution de la courbe $\sigma(\varepsilon)$. Ensuite, les prédictions basées sur cette fonction polynomiale sont utilisées pour déduire la courbe $\theta(\sigma)$. Dans cette étude, nous avons choisi d'adopter une méthode basée sur les RNA, en raison de sa capacité déjà démontrée à prédire le comportement de ce matériau dans les chapitres précédents. Cette méthode implique tout d'abord le lissage de chaque courbe $\sigma(\varepsilon)$ à l'aide d'un réseau de neurones spécifique. Ensuite, les données obtenues sont utilisées pour déduire la relation $\theta(\sigma)$. A noter que le but des courbes d'érouissage est de mettre en évidence les points d'inflexion qui caractérisent le début de la recristallisation.

Le modèle RNA est utilisé dans cette section pour filtrer les données qui serviront d'entrée pour le calcul des conditions critiques de la recristallisation. L'architecture choisie pour notre application se compose de deux couches cachées, l'une avec 7 neurones et l'autre avec 5 neurones, comme illustré sur la Figure 4.14. Pour ce modèle la fonction d'activation est de type sigmoïd et n'est qu'appliquée sur les deux couches cachées. Les équations présentées dans le chapitre 3 sont valables ici et donc ne seront pas reprises.

Pour calculer le nombre de paramètres internes N_{int} dans le modèle, où n et m représentent respectivement le nombre de neurones dans la première et la deuxième couche cachée, la formule suivante peut être utilisée : $N_{int} = m(2 + n) + 2n + 1$. Ainsi, pour un réseau de type 7 – 5, $N_{int} = 60$.

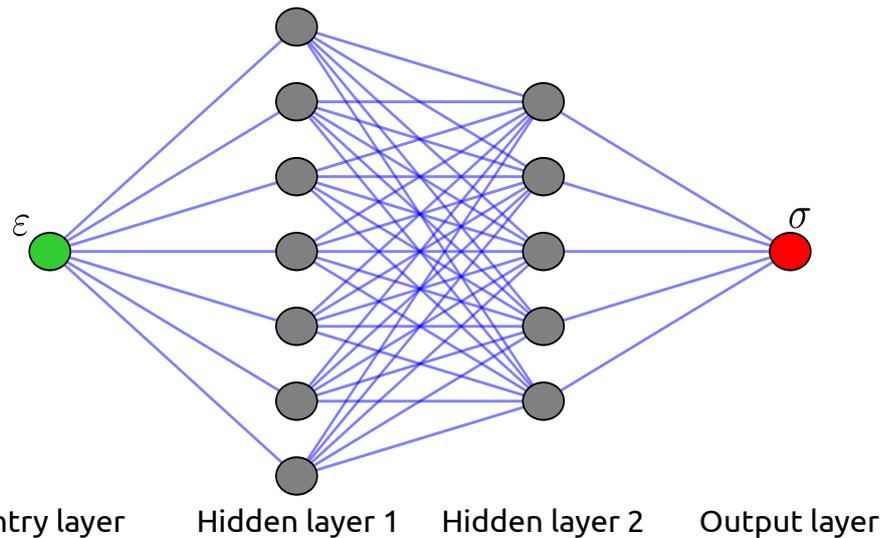


FIGURE 4.14 – Architecture du modèle RNA utilisé pour le filtrage des données.

Le programme Python utilisé pour l'entraînement du réseau de neurones a été créé avec la bibliothèque Tensorflow et l'optimisation des paramètres repose à nouveau sur l'algorithme ADAM. L'entraînement a été réalisé en utilisant 50 000 époques de l'ensemble de données expérimentales. Il a fallu environ 15 minutes pour entraîner le modèle RNA sur un ordinateur portable Dell XPS-13.

Étant donné que le phénomène de recristallisation dynamique ne se produit que pour des vitesses de déformation comprises entre $\dot{\epsilon} = 10^{-3} \text{ s}^{-1}$ et $\dot{\epsilon} = 1 \text{ s}^{-1}$, nous utiliserons uniquement les 20 premières courbes contrainte/déformation pour identifier les conditions critiques de la DRX. Chacune de ces 20 courbes se compose de 700 paires de valeurs de déformation-contrainte et a été traitée de manière indépendante par un réseau, ce qui a conduit à l'identification de 20 réseaux. A titre d'exemple, la Figure 4.15 présente les valeurs expérimentales de la courbe contrainte/déformation pour $\dot{\epsilon} = 10^{-3} \text{ s}^{-1}$ et $T = 1050 \text{ °C}$, (en bleu) ainsi que les résultats du RNA identifiés à partir de ces données (en rouge).

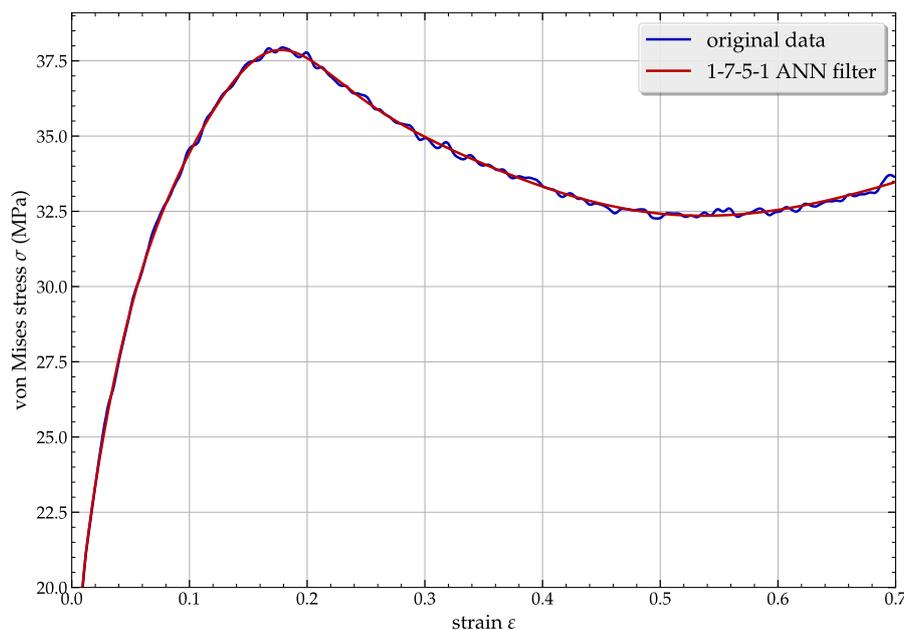


FIGURE 4.15 – Filtrage des données à l'aide du RNA pour 10^{-3} s^{-1} et 1050 °C .

Avec le filtrage RNA, $E_{RMS} = 0.093$ MPa et $E_{MAR} = 0.231$ %, tandis que l'utilisation d'une fonction polynomiale de degré 11 pour représenter les données expérimentales donne $E_{RMS} = 0.120$ MPa et $E_{MAR} = 0.275$ %. L'utilisation du modèle RNA pour représenter la courbe $\sigma(\varepsilon)$ donne de meilleurs résultats qu'un ajustement polynomial d'ordre 11, comme le démontre l'erreur estimée plus faible. Cette architecture de RNA est utilisée pour toutes les courbes suivantes, ce qui permet de déduire les courbes de durcissement, qui illustrent le taux de changement de contrainte par rapport à la déformation ($\frac{\partial\sigma}{\partial\varepsilon}$). Une illustration du calcul de la courbe de durcissement $\theta(\sigma)$ pour une seule température est donnée sur la Figure 4.16.

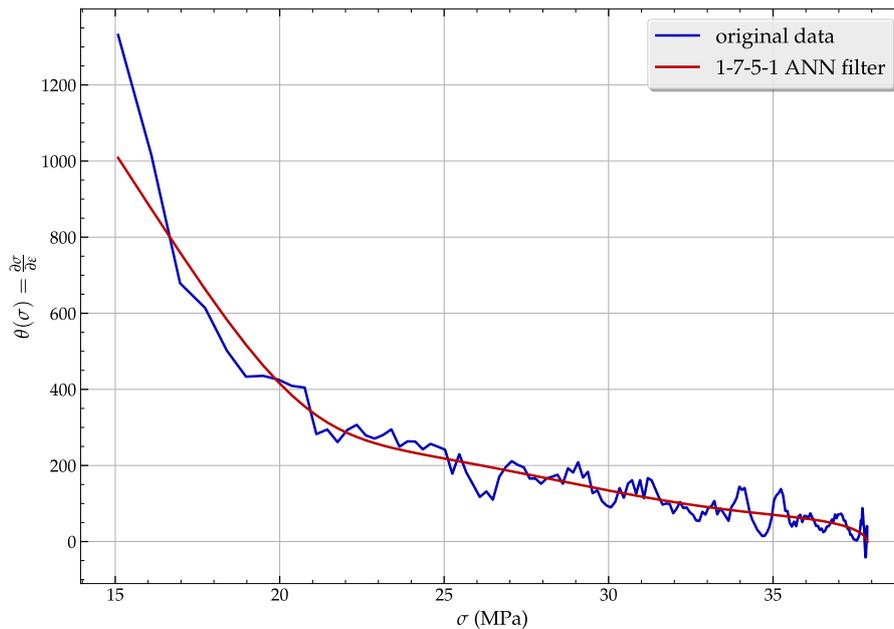


FIGURE 4.16 – Exemple de d'une courbe de durcissement.

La ligne bleue représente le calcul direct de la dérivée de la contrainte brute par rapport à la déformation, tandis que la ligne rouge représente la dérivée de la contrainte filtrée par le réseau de neurones par rapport à la déformation. Un aperçu des autres températures est donné sur la Figure 4.17. Le même processus s'applique à toutes les 20 courbes contrainte/déformation. Ces courbes fournissent des données essentielles concernant les contraintes et les déformations critiques nécessaires pour prédire la recristallisation dynamique dans l'analyse ultérieure.

Les sections suivantes aborderont deux approches distinctes. Dans un premier temps, la méthode de Najafzadeh et Jonas [Najafzadeh et Jonas, 2006] sera exposée. Elle sera suivie de la présentation d'une approche alternative visant à améliorer la précision de la détermination des conditions critiques. Comme précédemment mentionné, la méthode de Najafzadeh et Jonas repose sur un polynôme de degré 3 pour décrire les fonctions $\theta(\sigma)$ et $\ln \theta(\varepsilon)$. Cependant, dans certaines situations, cette approximation n'est pas suffisante. C'est pourquoi nous avons étendu cette approche à l'utilisation d'un polynôme de degré 5, pour lequel une méthode de résolution spécifique sera présentée.

4.4.2 Conditions critiques de la DRX

Comme abordé dans la section précédente, il est essentiel de souligner que l'initiation de la recristallisation dynamique dépend de l'atteinte d'un niveau spécifique de conditions critiques. Cependant, la détermination précise de ces conditions critiques représente un défi, nécessitant le développement de modèles capables de calculer les valeurs de contrainte et de déformation critiques correspondantes.

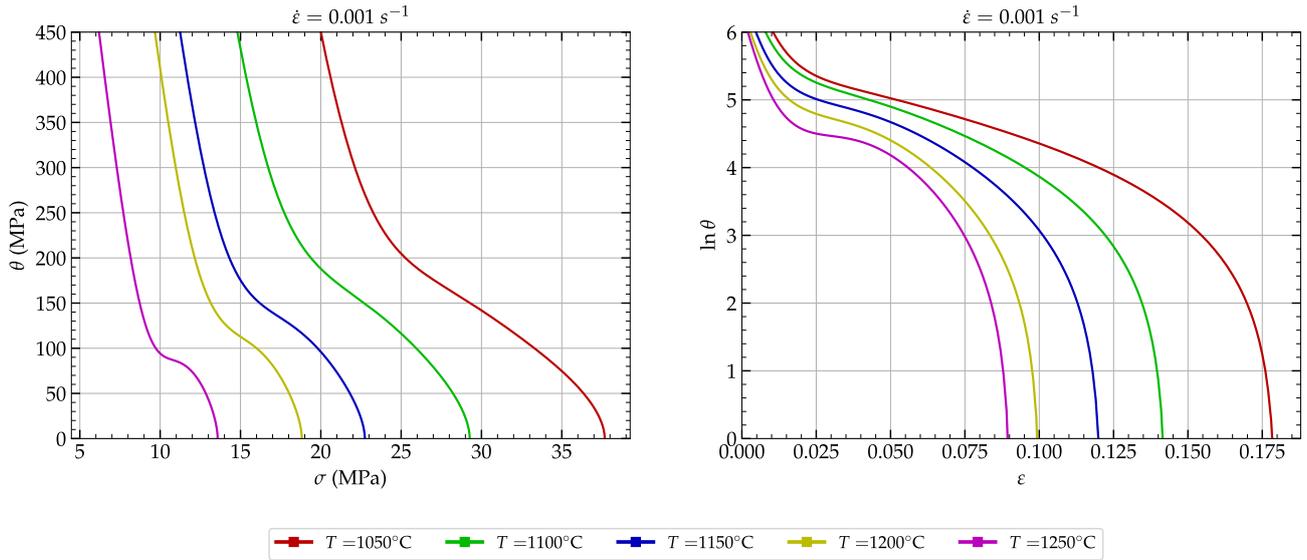


FIGURE 4.17 – Conditions critiques de la recristallisation dynamique

Dans cette optique, la méthode bien connue de [Najafizadeh et Jonas, 2006] est présentée dans un premier temps, et ensuite améliorée en utilisant deux approches.

4.4.2.1 Approche de Najafizadeh et Jonas

La détermination des points d’inflexion s’effectue en approximant la courbe de la fonction $\theta(\sigma)$ par un polynôme de degré 3 jusqu’à la contrainte pic σ_p . Ainsi, on a l’équation suivante :

$$\theta = A\sigma^3 + B\sigma^2 + C\sigma + D, \tag{4.17}$$

où A, B, C et D sont les constantes du polynôme. Le point d’inflexion d’une fonction polynomiale de degré 3 se calculant en annulant la dérivée seconde, il vient alors :

$$\frac{d^2\theta}{d\sigma^2} = 6A\sigma + 2B. \tag{4.18}$$

Ainsi la contrainte critique pour l’initiation de la DRX s’obtient comme suit :

$$6A\sigma_c + 2B = 0 \implies \sigma_c = -B/3A. \tag{4.19}$$

Dans l’approche de Najafizadeh et Jonas, la déformation critique se calcule à partir des équations suivantes :

$$\ln \theta = A_1\varepsilon^3 + B_1\varepsilon^2 + C_1\varepsilon + D_1, \tag{4.20}$$

et

$$\frac{d^2 \ln \theta}{d\varepsilon^2} = 6\varepsilon A_1 + 2B_1 \quad \text{et} \quad \frac{d^2 \ln \theta}{d\varepsilon^2} = 0 \implies \varepsilon_c = -B_1/3A_1. \tag{4.21}$$

Ainsi les contraintes σ_c et déformations critiques ε_c s’obtiennent à partir des constantes A, B, A_1 et B_1 calculés pour le polynôme correspondant, pour toutes les vitesses et températures. La Figure 4.18 donne un exemple des fonctions $\theta(\sigma)$ et $\ln \theta(\varepsilon)$ avec leurs polynômes respectifs pour une vitesse de déformation $\dot{\varepsilon} = 10^{-3} \text{ s}^{-1}$. Comme nous pouvons observer sur la Figure 4.18, le polynôme de degré 3 ne permet pas de décrire fidèlement l’expérience, et de plus cette approche présente une limite

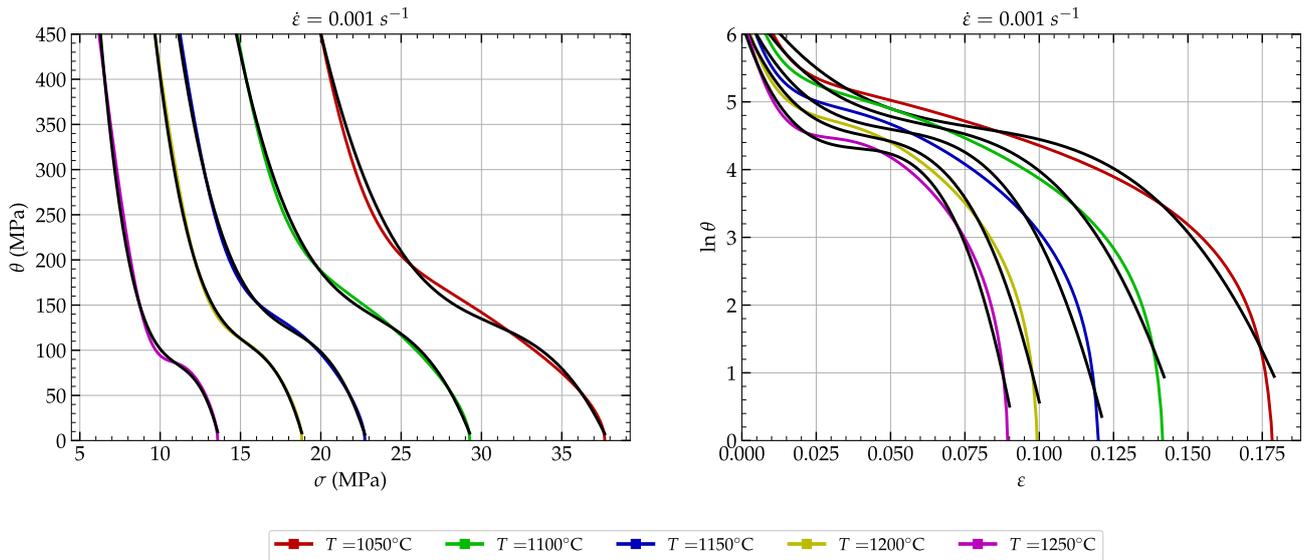


FIGURE 4.18 – Prédiction des courbes d'écrouissage basée sur l'approche de Najafizadeh et Jonas.

dans le sens où les σ_c et ϵ_c calculées ne correspondent pas exactement sur la courbe $\sigma(\epsilon)$. D'où la nécessité de trouver une alternative en modifiant la technique de calcul de ϵ_c comme présenté ci-dessous.

Une méthode simple, permettant de contourner cette difficulté, consiste à faire un calcul inverse pour déterminer ϵ_c à partir de σ_c et de la loi d'écoulement filtrée par le réseau de neurones. Ainsi, les contraintes et déformations critiques identifiées par cette méthode simplifiée sont résumées dans le Tableau 4.4 et les constantes du polynôme θ sont correspondantes données dans le Tableau 4.5.

TABLE 4.4 – Conditions critiques basées sur l'approche modifiée de Najafizadeh et Jonas.

$\dot{\varepsilon}$ (s ⁻¹)	T (°C)	ε_p	σ_p (MPa)	ε_c	σ_c (MPa)
0.001	1050	0.178	37.862	0.083	33.050
	1100	0.143	29.770	0.067	25.648
	1150	0.124	23.330	0.051	19.116
	1200	0.102	19.373	0.039	16.244
	1250	0.084	13.883	0.035	11.302
0.01	1050	0.281	57.915	0.131	51.602
	1100	0.228	46.107	0.108	40.495
	1150	0.195	38.249	0.086	33.435
	1200	0.172	30.991	0.076	26.718
	1250	0.139	24.739	0.051	20.004
0.1	1050	0.571	91.413	0.244	83.814
	1100	0.430	72.662	0.194	65.259
	1150	0.344	57.774	0.160	51.723
	1200	0.299	48.473	0.136	42.386
	1250	0.259	40.195	0.119	35.277
1	1050	0.700	127.17	0.577	116.10
	1100	0.700	106.61	0.329	94.510
	1150	0.700	89.525	0.268	89.067
	1200	0.700	72.270	0.178	57.929
	1250	0.393	59.992	0.112	54.066

TABLE 4.5 – Coefficients de θ basés sur l'approche modifiée de Najafizadeh.

$\dot{\epsilon}$ (s^{-1})	T ($^{\circ}C$)	θ (σ)
0.001	1050	$\theta = -0.183\sigma^3 + 17.050\sigma^2 - 536.275\sigma + 5817.662$
	1100	$\theta = -0.073\sigma^3 + 5.992\sigma^2 - 179.369\sigma + 1981.540$
	1150	$\theta = -0.098\sigma^3 + 4.867\sigma^2 - 96.330\sigma + 859.945$
	1200	$\theta = -0.206\sigma^3 + 8.063\sigma^2 - 120.550\sigma + 819.215$
	1250	$\theta = -0.0291\sigma^3 - 0.191\sigma^2 - 10.927\sigma + 287.532$
0.01	1050	$\theta = -0.121\sigma^3 + 17.0365\sigma^2 - 796.743\sigma + 12509.561$
	1100	$\theta = -0.142\sigma^3 + 15.924\sigma^2 - 600.836\sigma + 7712.952$
	1150	$\theta = -0.105\sigma^3 + 9.878\sigma^2 - 320.382\sigma + 3689.927$
	1200	$\theta = -0.0207\sigma^3 + 1.797\sigma^2 - 67.646\sigma + 1006.697$
	1250	$\theta = -0.103\sigma^3 + 5.212\sigma^2 - 98.948\sigma + 838.445$
0.1	1050	$\theta = -0.017\sigma^3 + 3.961\sigma^2 - 311.320\sigma + 8348.890$
	1100	$\theta = -0.033\sigma^3 + 6.003\sigma^2 - 366.759\sigma + 7631.0369$
	1150	$\theta = -0.071\sigma^3 + 10.128\sigma^2 - 488.312\sigma + 7988.310$
	1200	$\theta = -0.150\sigma^3 + 17.317\sigma^2 - 662.240\sigma + 8499.817$
	1250	$\theta = -0.199\sigma^3 + 19.292\sigma^2 - 622.041\sigma + 6782.884$
1	1050	$\theta = -0.003\sigma^3 + 1.270\sigma^2 - 159.648\sigma + 6771.970$
	1100	$\theta = -0.006\sigma^3 + 1.766\sigma^2 - 179.233\sigma + 6209.260$
	1150	$\theta = -0.010\sigma^3 + 2.337\sigma^2 - 191.401\sigma + 5394.0338$
	1200	$\theta = -0.023\sigma^3 + 4.176\sigma^2 - 263.458\sigma + 5737.964$
	1250	$\theta = -0.036\sigma^3 + 5.410\sigma^2 - 276.701\sigma + 4902.773$

4.4.2.2 Approche basée sur un degré 5

La méthode que nous introduisons ici constitue essentiellement une amélioration du modèle exposé dans la section précédente. Nous remplaçons ici le polynôme de degré 3 par un polynôme de degré 5 pour décrire les fonctions $\theta(\sigma)$ et $\ln(\varepsilon)$. Puis, nous proposons une technique de résolution pour l'équation dérivée seconde résultante des fonctions $\theta(\sigma)$ et $\ln(\varepsilon)$.

Ce qui caractérise cette technique de résolution, c'est son transfert dans le domaine des nombres complexes, utilisant des méthodes de résolution trigonométriques. De cette manière, plusieurs solutions émergent, et nous sommes en mesure de choisir celle qui correspond au domaine prédéfini par observation préalable de l'intervalle des solutions attendues, tel qu'illustré sur la Figure 4.17. La procédure proposée est basée sur les équations suivantes :

$$\theta = A_5\sigma^5 + A_4\sigma^4 + A_3\sigma^3 + A_2\sigma^2 + A_1\sigma + A_0, \quad (4.22)$$

$$\frac{d^2\theta}{d\sigma^2} = 20A_5\sigma^3 + 12A_4\sigma^2 + 6A_3\sigma + 2A_2 \quad (4.23)$$

$$\frac{d^2\theta}{d\sigma^2} = 0 \iff a\sigma^3 + b\sigma^2 + c\sigma + d = 0, \quad (4.24)$$

avec $a = 10A_5$, $b = 6A_4$, $c = 3A_3$, $d = A_2$.

Si on fait un changement de variable, on peut écrire la contrainte comme suit :

$$\sigma = X - \frac{b}{3a}. \quad (4.25)$$

En remplaçant cette relation dans l'équation (4.24), il vient donc la nouvelle relation :

$$X^3 + pX + q = 0 \quad \text{avec} \quad p = \frac{c}{a} - \frac{b^2}{3a^2} \quad \text{et} \quad q = \frac{2b^3}{27a^3} + \frac{d}{a} - \frac{bc}{3a^2}. \quad (4.26)$$

Posons à nouveau un changement de variable :

$$u + v = X \quad \text{avec} \quad 3uv = -p. \quad (4.27)$$

En remplaçant cette nouvelle expression dans l'équation (4.26) on a :

$$\begin{cases} u^3 + v^3 = -q \\ u^3v^3 = -\frac{p^3}{27} \end{cases} \iff \begin{cases} U + V = -q \\ UV = -\frac{p^3}{27} \end{cases} \implies U^2 + qU - \frac{p^3}{27} = 0, \quad U = u^3. \quad (4.28)$$

On peut alors calculer le discriminant Δ comme étant :

$$\Delta = q^2 + 4\frac{p^3}{27}. \quad (4.29)$$

- Si $\Delta \geq 0$ alors la solution est :

$$x = u + v = \sqrt[3]{-\frac{q}{2} + \sqrt{\frac{q^2}{4} + \frac{p^3}{27}}} + \sqrt[3]{\frac{q}{2} - \sqrt{\frac{q^2}{4} + \frac{p^3}{27}}} - \frac{b}{3a} \quad (4.30)$$

- Si $\Delta < 0 \Rightarrow u^3 = re^{i\alpha}$

$$\begin{cases} r = \sqrt{\frac{-p^3}{27}}, \cos(\alpha) = -\frac{q}{2r} \\ u_k = \sqrt{\frac{-p}{3}} \left[\cos\left(\frac{\alpha+2k\pi}{3}\right) + i \sin\left(\frac{\alpha+2k\pi}{3}\right) \right] = \sqrt{\frac{-p}{3}} e^{i(\alpha+2k\pi)} \\ v_k = \bar{u}_k, k \in \{0, 1, 2\} \end{cases} \quad (4.31)$$

Et les solutions sont données par :

$$\begin{cases} X_k = 2\sqrt{\frac{-p}{3}} \cos\left(\frac{\alpha+2k\pi}{3}\right) \\ x_k = X_k - \frac{b}{3a}. \end{cases} \quad (4.32)$$

Un script Python est développé pour résoudre ces équations dépendamment du signe de Δ . Les solutions de ces équations (correspondant aux contraintes critiques), sont synthétisées dans le Tableau 4.6. Les coefficients des fonctions $\theta(\sigma)$ et $\ln(\varepsilon)$ sont résumés dans les Tableaux 4.7 et 4.8. La Figure 4.19 illustre les courbes des fonctions $\theta(\sigma)$ et $\ln(\varepsilon)$ ainsi que leurs polynômes respectifs.

TABLE 4.6 – Conditions critiques pour un polynôme de degré 5

$\dot{\varepsilon}$ (s ⁻¹)	T (°C)	ε_p	σ_p (MPa)	ε_c	σ_c (MPa)
0.001	1050	0.180	37.673	0.122	29.408
	1100	0.143	29.270	0.056	22.808
	1150	0.122	22.739	0.030	17.905
	1200	0.101	18.818	0.027	15.422
	1250	0.091	13.579	0.026	11.062
0.01	1050	0.284	57.702	0.193	46.363
	1100	0.232	45.838	0.157	37.079
	1150	0.192	38.409	0.092	31.073
	1200	0.165	30.719	0.081	24.400
	1250	0.142	24.266	0.045	19.033
0.1	1050	0.586	92.085	0.155	73.166
	1100	0.420	72.796	0.116	58.675
	1150	0.344	57.730	0.097	48.230
	1200	0.299	48.440	0.086	39.930
	1250	0.257	40.084	0.067	32.689
1	1050	0.7	127.340	0.578	103.973
	1100	0.7	107.312	0.330	91.609
	1150	0.7	89.738	0.303	77.951
	1200	0.7	72.630	0.236	65.816
	1250	0.7	60.173	0.109	50.158

Sur ces différentes figures on observe une bonne amélioration apportée en utilisant un polynôme de degré 5 en termes d'ajustement des courbes prédites. Une fois que les valeurs critiques de contrainte et de déformation sont obtenues, il est possible de les exploiter pour calculer la fraction volumique de la recristallisation. En effet, ces valeurs critiques définissent les conditions requises pour le

déclenchement de la recristallisation dynamique. Dans la section suivante ces valeurs critiques seront utilisées pour prédire le processus de recristallisation en se basant sur le modèle d'Avrami [Razali et Joun, 2021], plus communément désigné sous le nom de modèle JMAK.

TABLE 4.7 – Coefficients de $\theta = f(\sigma)$ en utilisant un polynôme de degré 5

$\dot{\epsilon}$ (s ⁻¹)	T (°C)	$\theta(\sigma)$
0.001	1050	$\theta = 0.001\sigma^5 - 0.089\sigma^4 + 4.419\sigma^3 - 98.193\sigma^2 + 859.520\sigma - 695.872$
	1100	$\theta = 0.0013\sigma^5 - 0.131\sigma^4 + 4.961\sigma^3 - 80.460\sigma^2 + 430.345\sigma + 1019.710$
	1150	$\theta = 0.004\sigma^5 - 0.368\sigma^4 + 11.319\sigma^3 - 153.853\sigma^2 + 791.522\sigma - 7.731$
	1200	$\theta = 0.014\sigma^5 - 1.058\sigma^4 + 30.230\sigma^3 - 400.915\sigma^2 + 2357.110\sigma - 4122.89$
	1250	$\theta = 0.081\sigma^5 - 4.399\sigma^4 + 91.725\sigma^3 - 901.237\sigma^2 + 4066.220\sigma - 6214.420$
0.01	1050	$\theta = 0.001\sigma^5 - 0.024\sigma^4 + 1.903\sigma^3 - 67.577\sigma^2 + 988.644\sigma - 2664.88$
	1100	$\theta = 5.114 \times 10^{-5}\sigma^5 - 0.008\sigma^4 + 0.297\sigma^3 + 1.342\sigma^2 - 320.970\sigma + 5319.64$
	1150	$\theta = -0.001\sigma^5 + 0.031\sigma^4 - 1.928\sigma^3 + 64.337\sigma^2 - 1123.97\sigma + 8360.230$
	1200	$\theta = -0.001\sigma^5 + 0.047\sigma^4 - 2.515\sigma^3 + 71.523\sigma^2 - 1060.730\sigma + 6643.460$
	1250	$\theta = 0.001\sigma^5 - 0.082\sigma^4 + 2.045\sigma^3 - 9.757\sigma^2 - 282.748\sigma + 3080.630$
0.1	1050	$\theta = 1.302 \times 10^{-5}\sigma^5 - 0.005\sigma^4 + 0.484\sigma^3 + -25.456\sigma^2 - 514.78\sigma + 403.380$
	1100	$\theta = 1.948 \times 10^{-5}\sigma^5 - 0.005\sigma^4 + 0.434\sigma^3 - 15.996\sigma^2 + 137.430\sigma + 3136.790$
	1150	$\theta = 2.195 \times 10^{-5}\sigma^5 - 0.005\sigma^4 + 0.30416\sigma^3 - 6.259\sigma^2 - 112.791\sigma + 4416.35$
	1200	$\theta = -2.620 \times 10^{-6}\sigma^5 + 0.001\sigma^4 - 0.152\sigma^3 + 12.226\sigma^2 - 425.697\sigma + 5492.32$
	1250	$\theta = -5.867 \times 10^{-6}\sigma^5 + 0.002\sigma^4 - 0.270\sigma^3 + 17.462\sigma^2 - 489.963\sigma + 5123.9$
1	1050	$\theta = 2.857 \times 10^{-5}\sigma^5 - 70.002\sigma^4 + 0.180\sigma^3 - 12.210\sigma^2 - 300.642\sigma + 1076.81$
	1100	$\theta = 6.266 \times 10^{-5}\sigma^5 - 0.002\sigma^4 + 0.289\sigma^3 - 16.679\sigma^2 - 353.565\sigma + 804.856$
	1150	$\theta = 1.038 \times 10^{-5}\sigma^5 - 0.003\sigma^4 + 0.312\sigma^3 - 13.890\sigma^2 - 170.132\sigma + 2819.89$
	1200	$\theta = 1.436 \times 10^{-5}\sigma^5 - 0.003\sigma^4 + 0.260\sigma^3 - 7.143\sigma^2 - 54.670\sigma + 4332.770$
	1250	$\theta = 3.692 \times 10^{-7}\sigma^5 + 0.001\sigma^4 - 0.103\sigma^3 + 9.732\sigma^2 - 394.186\sigma + 6051.84$

TABLE 4.8 – Coefficients de $\ln \theta = f(\epsilon)$ en utilisant un polynôme de degré 5

$\dot{\epsilon}$ (s^{-1})	T ($^{\circ}C$)	$\ln \theta(\epsilon)$
0.001	1050	$\ln \theta = -721424\epsilon^5 + 317657\epsilon^4 - 52561.2\epsilon^3 + 3968.35\epsilon^2 - 145.594\epsilon + 7.194$
	1100	$\ln \theta = -2102440\epsilon^5 + 728478\epsilon^4 - 94842.4\epsilon^3 + 5616.91\epsilon^2 - 162.826\epsilon + 6.999$
	1150	$\ln \theta = -6236090\epsilon^5 - 1806880\epsilon^4 - 194286\epsilon^3 + 9343.86\epsilon^2 - 210.61\epsilon + 6.819$
	1200	$\ln \theta = -11716700\epsilon^5 + 2836160\epsilon^4 - 259422\epsilon^3 + 10823.2\epsilon^2 - 218.263\epsilon + 6.579$
	1250	$\ln \theta = -18637400\epsilon^5 + 4128750\epsilon^4 - 351246\epsilon^3 + 13899.3\epsilon^2 - 265.261\epsilon + 6.558$
0.01	1050	$\ln \theta = -78267.9\epsilon^5 + 55460.7\epsilon^4 - 14875.5\epsilon^3 + 1842.9\epsilon^2 - 111.046\epsilon + 7.554$
	1100	$\ln \theta = -209650\epsilon^5 + 116347\epsilon^4 - 24265.4\epsilon^3 + 2329.22\epsilon^2 - 111.312\epsilon + 7.160$
	1150	$\ln \theta = -592486\epsilon^5 + 266204\epsilon^4 - 44359.1\epsilon^3 + 3338.62\epsilon^2 - 1231360\epsilon + 6.945$
	1200	$\ln \theta = -1157610\epsilon^5 + 448703\epsilon^4 - 64632.2\epsilon^3 + 4198.66\epsilon^2 - 133.596\epsilon + 6.761$
	1250	$\ln \theta = -2441430\epsilon^5 + 823576\epsilon^4 - 103470\epsilon^3 + 5855.61\epsilon^2 - 160.162\epsilon + 6.678$
0.1	1050	$\ln \theta = -4048.91\epsilon^5 + 4933.08\epsilon^4 - 2320.94\epsilon^3 + 517.54\epsilon^2 - 58.6291\epsilon + 7.440$
	1100	$\ln \theta = -11998.6\epsilon^5 + 12331.1\epsilon^4 - 4772.38\epsilon^3 + 852.317\epsilon^2 - 74.8041\epsilon + 7.402$
	1150	$\ln \theta = -31345.9\epsilon^5 + 26311\epsilon^4 - 8342.77\epsilon^3 + 1227.55\epsilon^2 - 89.2542\epsilon + 7.321$
	1200	$\ln \theta = -53144.5\epsilon^5 + 38907.5\epsilon^4 - 10839.7\epsilon^3 + 1407.87\epsilon^2 - 90.811\epsilon + 7.098$
	1250	$\ln \theta = -153109\epsilon^5 + 93838.3\epsilon^4 - 21497.3\epsilon^3 + 2251.42\epsilon^2 - 113.922\epsilon + 7.046$
1	1050	$\ln \theta = -551.664\epsilon^5 + 979.501\epsilon^4 - 627.907\epsilon^3 + 190.182\epsilon^2 - 34.075\epsilon + 7.305$
	1100	$\ln \theta = -575.565\epsilon^5 + 1063.850\epsilon^4 - 712.153\epsilon^3 + 220.358\epsilon^2 - 37.5877\epsilon + 7.223$
	1150	$\ln \theta = -683.352\epsilon^5 + 1244.81\epsilon^4 - 818.105\epsilon^3 + 243.788\epsilon^2 - 39.077\epsilon + 7.092$
	1200	$\ln \theta = -1444.84\epsilon^5 + 2572.62\epsilon^4 - 1607.91\epsilon^3 + 426.213\epsilon^2 - 54.003\epsilon + 7.209$
	1250	$\ln \theta = -13356.4\epsilon^5 + 12997.4\epsilon^4 - 4758.18\epsilon^3 + 802.196\epsilon^2 - 68.579\epsilon + 7.142$

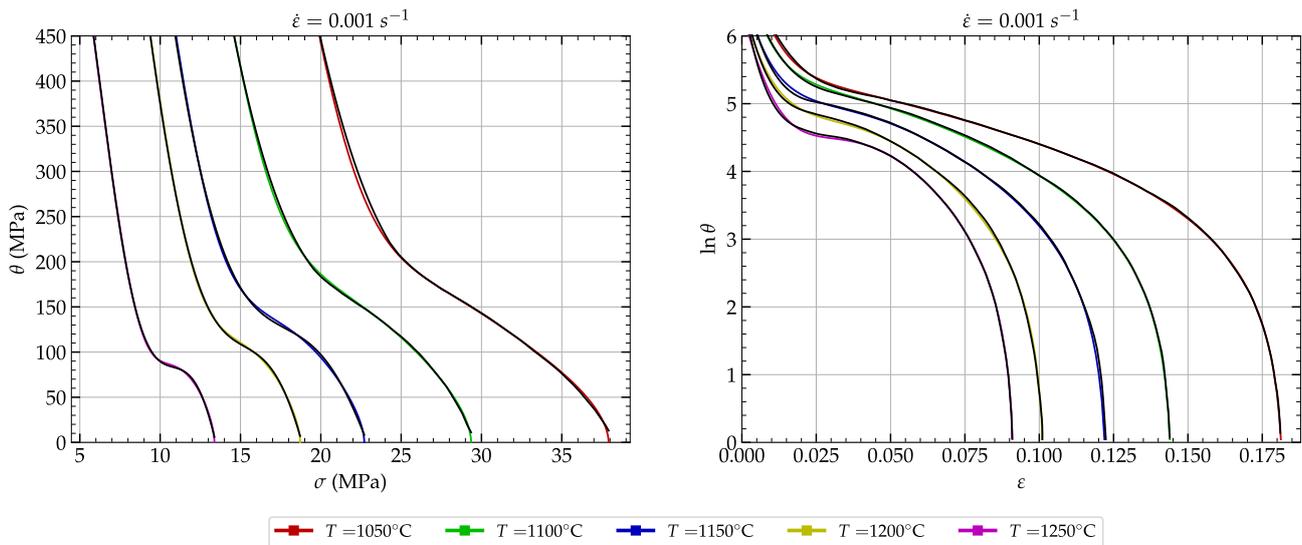


FIGURE 4.19 – Prédiction des courbes d'écoulement basée sur un polynôme de degré 5.

4.4.3 Cinétique de la DRX

Le processus de recristallisation dynamique (DRX) commence à partir d'une déformation critique comme présenté précédemment et le niveau de recristallisation peut se déterminer expérimentalement. La Figure 4.20 montre trois niveaux de la DRX : la nucléation, la croissance des grains et la coalescence des grains. La DRX affecte la courbe de contrainte d'écoulement du matériau car le taux

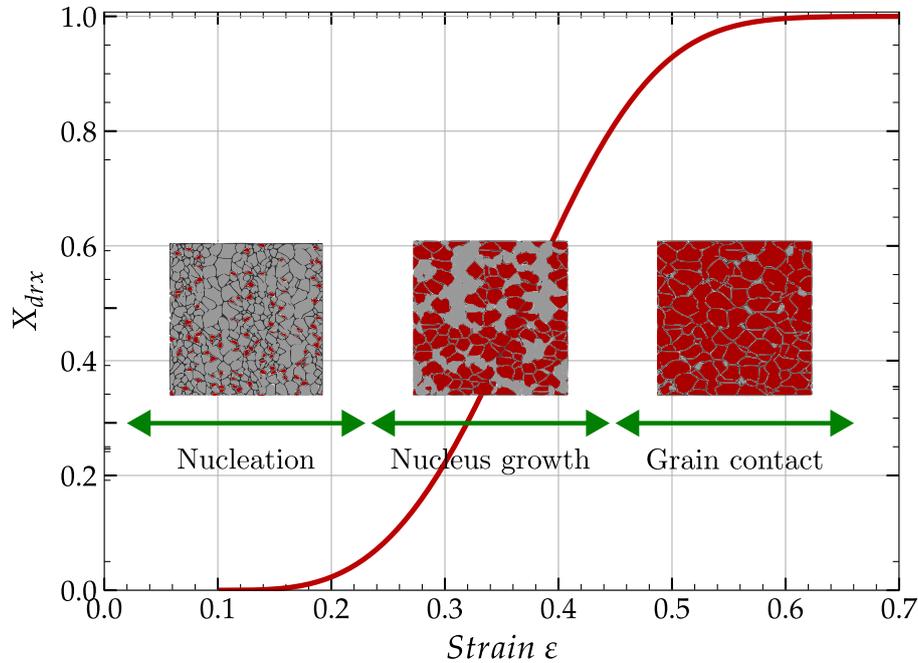


FIGURE 4.20 – Diagramme des trois étapes de la DRX

d'écrouissage est affecté par les dislocations dans les grains libres. Le modèle de Avrami [Razali et Joun, 2021], connu sous le nom du modèle JMAK, est la loi la plus utilisée pour prédire ce comportement. Son expression est donnée par l'équation suivante qui estime la fraction volumique de la recristallisation dynamique :

$$X_{drx}^p = 1 - \exp \left[-k \left(\frac{\varepsilon - \varepsilon_c}{\varepsilon_p} \right)^{n_k} \right], \quad (4.33)$$

avec ε_c et ε_p les déformations critique et pic respectivement; k et n_k les paramètres du modèle. L'équivalence expérimentale de cette fraction volumique de recristallisation dynamique est donnée par :

$$X_{drx}^e = \frac{\sigma_p - \sigma}{\sigma_p - \sigma_s}, \quad (4.34)$$

où σ_p et σ_s sont les contraintes pic et stable après le pic respectivement.

Il est possible d'établir une relation de dépendance entre ces contraintes et déformations spécifiques en utilisant le paramètre de Zener–Hollomon défini par l'équation (2.11), page 33, sous la forme suivante :

$$\begin{cases} \varepsilon_c = A_c Z^{n_c} \\ \varepsilon_p = A_p Z^{n_p} \\ \sigma_c = B_c Z^{m_c} \\ \sigma_p = B_p Z^{m_p} \\ \sigma_s = B_s Z^{m_s} \end{cases}, \quad (4.35)$$

avec $A_{(c,p)}$, $n_{(c,p)}$, $B_{(c,p,s)}$ et $m_{(c,p,s)}$ les paramètres de dépendance des déformations et contraintes spécifiques au paramètre Z . La procédure d'identification de ces paramètres, basée sur les travaux de [Chen *et al.*, 2014, Li *et al.*, 2019a, Wang *et al.*, 2021], est détaillée ci-après.

Le calcul des paramètres du modèle JMAK commence d'une part par la détermination des coefficients $A_{(c,p)}$, $n_{(c,p)}$, $B_{(c,p,s)}$ et $m_{(c,p,s)}$ et d'autre part par l'introduction de ces coefficients dans l'équation (4.35) avant leur utilisation finale dans les équations (4.33) et (4.34). Pour calculer ces paramètres, un code Python basé sur la méthode d'optimisation *curve_fit* de la librairie *scipy* est utilisé.

Une fois que les constantes définissant la dépendance du paramètre de Zener–Hollomon sont calculées, on peut les utiliser pour déduire les coefficients du modèle de JMAK. Une optimisation qui minimise l'erreur entre la recristallisation prédite X_{drx}^p et la valeur expérimentale X_{drx}^e est utilisée.

L'évolution de la microstructure n'étant pas dissociée de la taille de grains, il existe une relation qui lie la recristallisation dynamique à la taille de grain moyenne. Elle est donnée par l'équation suivante :

$$d_{drx,avg} = X_{drx} d_{drx} + d_0(1 - X_{drx}), \quad (4.36)$$

où d_{drx} est la taille de grain locale exprimée selon la relation suivante :

$$d_{drx} = n_1 d_0^{n_2} \varepsilon^{n_3} \dot{\varepsilon}^{n_4} \exp(n_5/RT), \quad (4.37)$$

pour laquelle, $d_0 = 350 \mu\text{m}$ est la taille de grain initiale et n_i sont les paramètres du modèle.

Les paramètres du modèle JMAK, de la dépendance du paramètre de Zener–Hollomon et de l'évolution de la taille de grain sont donnés dans le Tableau 4.9. Dans ce modèle sera associée également l'équation permettant de prédire la taille des grains au cours de la simulation.

TABLE 4.9 – Paramètres du modèle JMAK.

Paramètres	Approche modifiée de Najafizadeh	Polynôme de degré 5
A_c	4.427×10^{-5}	2.71×10^{-5}
A_p	29.850×10^{-5}	31.28×10^{-5}
B_c	0.072	0.065
B_p	0.103	0.094
B_s	10265.8	10265.8
n_c	0.236	-0.198
n_p	0.204	0.203
m_c	0.189	0.250
m_p	0.182	0.190
m_s	-0.198	-0.198
k	0.487	0.440
n_k	3.366	2.850
n_1	0	0
n_2	274.195	274.195
n_3	3.73×10^{-6}	3.73×10^{-6}
n_4	-0.092	-0.092
n_5	-112679	-112679

Les courbes de $X_{drx}(\varepsilon)$ sont données sur les Figure 4.21 et Figure 4.22 en utilisant les conditions critiques calculées par la méthode modifiée de Najafizadeh et celle basée sur un polynôme de degré 5 respectivement.

Une validation avec les données expérimentales (représentées par les points sur les Figures 4.21 et 4.22), des résultats obtenus par chaque approche montre clairement que les deux modèles approchent bien l'expérience et par conséquent les résultats d'un seul de ces modèles (version modifiée de Najafizadeh) seront utilisés pour la simulation numérique objet de la section suivante.

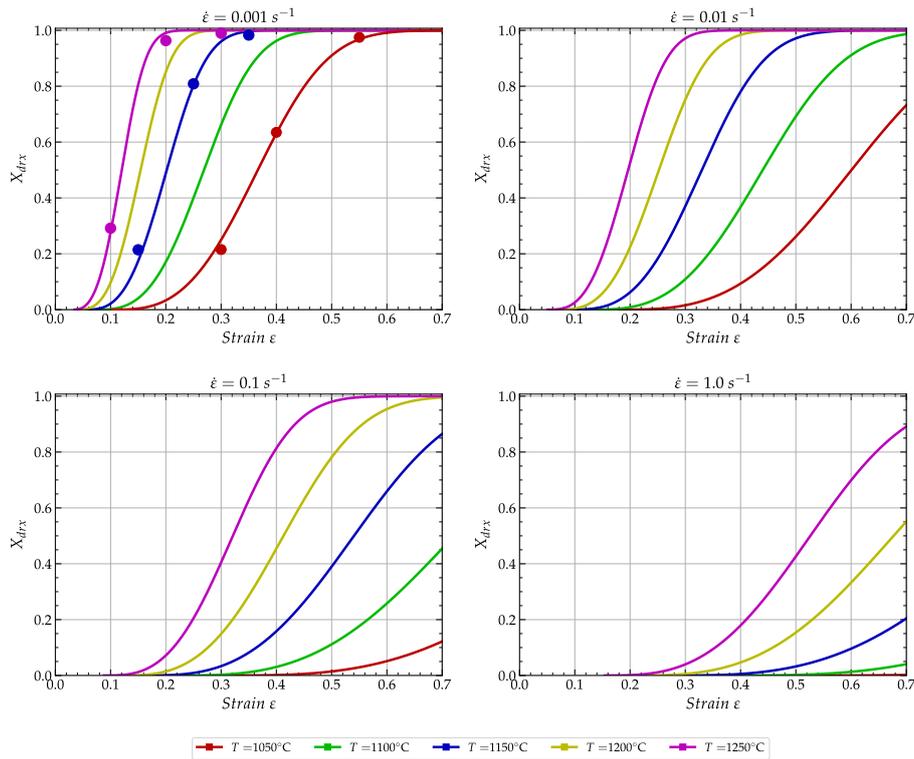


FIGURE 4.21 – Prédiction de la DRX basée sur l'approche modifiée de Najafizadeh – trait continu (modèle), points (validations expérimentales).

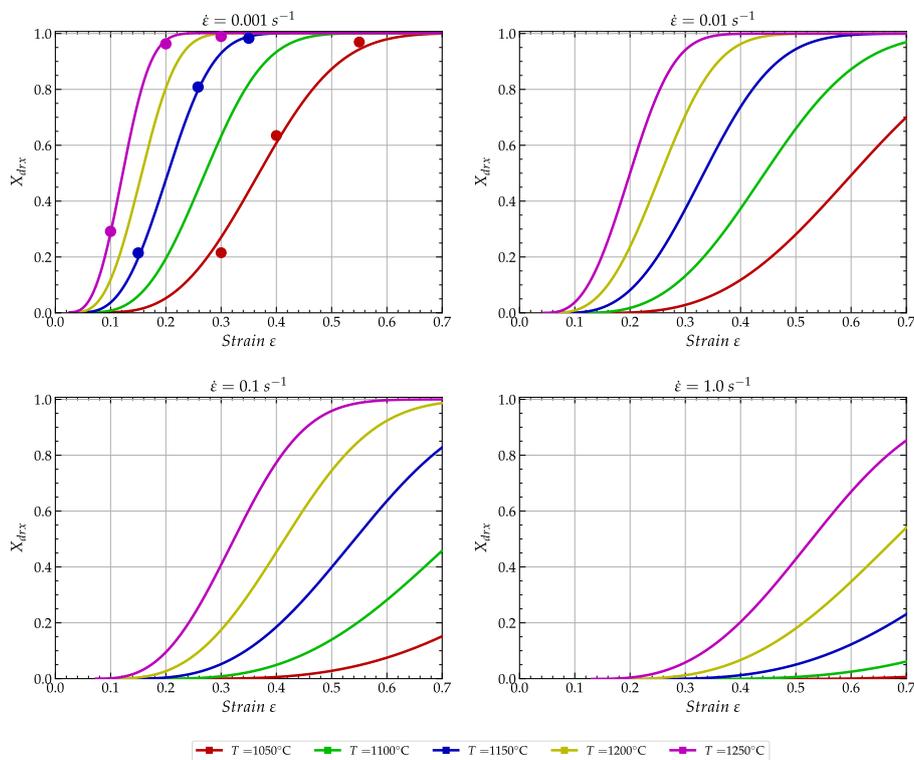


FIGURE 4.22 – Prédiction de la DRX basée sur un polynôme de degré 5 – trait continu (modèle), points (validations expérimentales).

4.5 Simulation numérique de la DRX

4.5.1 Données expérimentales

Le modèle DRX établi sur la base des courbes contrainte-déformation obtenues à l'aide d'essais de compression, est implémenté dans le logiciel élément finis Abaqus. Conformément à ce qui a été présenté au Chapitre 1, un ensemble de 30 essais de compression à haute température a été effectué sur le simulateur Gleeble-3800 sur la base de cinq températures (1050 °C, 1100 °C, 1150 °C, 1200 °C et 1250 °C) et six vitesses de déformation (10^{-3} s^{-1} , 10^{-2} s^{-1} , 10^{-1} s^{-1} , 1 s^{-1} , 2 s^{-1} et 5 s^{-1}).

Des analyses ont été effectuées sur les quatre premières vitesses de déformation (10^{-3} s^{-1} , 10^{-2} s^{-1} , 10^{-1} s^{-1} et 1 s^{-1}). En effet, les évolutions des courbes pour les deux dernières vitesses de déformation ($\dot{\epsilon} = 2 \text{ s}^{-1}$ et $\dot{\epsilon} = 5 \text{ s}^{-1}$) sont identiques à celles de la vitesse de déformation $\dot{\epsilon} = 1 \text{ s}^{-1}$ et de plus le phénomène d'adoucissement n'est pas perceptible sur les courbes $\sigma(\epsilon)$ ce qui traduirait une recristallisation non amorcée.

Lors des essais, une réduction de 60% est appliquée à toutes les éprouvettes et une trempe est appliquée après déformation pour figer la microstructure. Pour l'analyse de cette microstructure, la méthode des cercles interceptés est utilisée après que l'éprouvette ait été sectionnée, polie et attaquée chimiquement.

La taille des grains initiale mesurée avant déformation est autour de 300 μm alors que celle des grains recristallisés varie entre 23 μm et 100 μm . La Figure 4.23 montre que la DRX augmente et que les grains fins croissent graduellement au fur et mesure que la température augmente. Quant à la

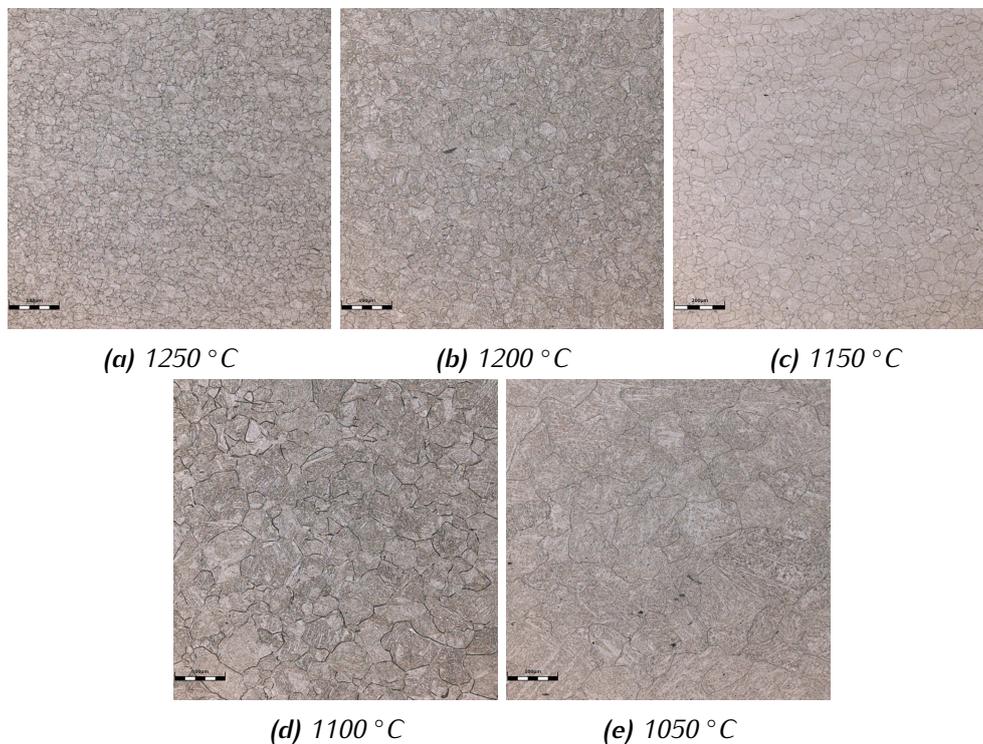


FIGURE 4.23 – Microstructure recristallisée du 3Cr2Mo au centre de l'éprouvette ($\dot{\epsilon} = 0.01 \text{ s}^{-1}$, réduction = 60 %)

Figure 4.24, il ressort qu'à la température 1150 °C avec différentes vitesses de déformation, la DRX est presque complète.

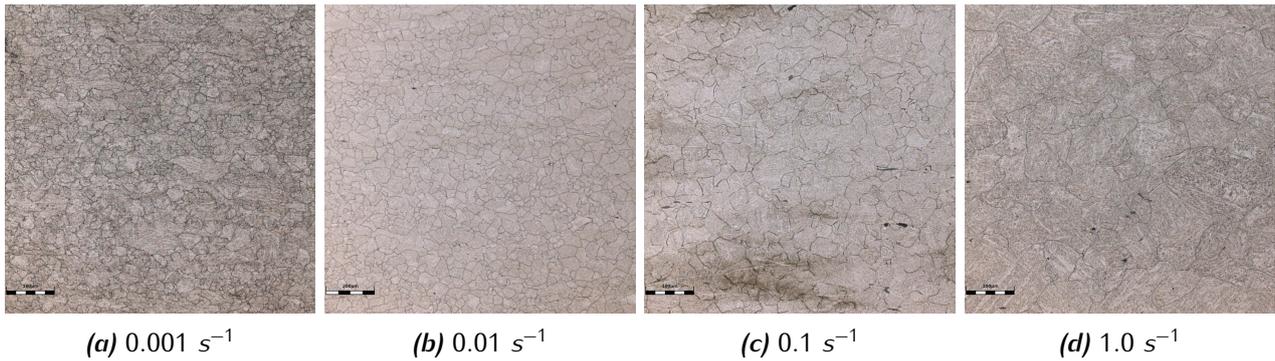


FIGURE 4.24 – Microstructure recristallisée d'un acier 3Cr2Mo au centre de l'éprouvette ($T_0 = 1150^\circ\text{C}$, réduction = 60 %)

Le Tableau 4.10 résume la taille des grains obtenus après déformation pour les conditions de travail sélectionnées.

TABLE 4.10 – Taille des grains mesurée après déformation.

$\dot{\epsilon}$ (s^{-1})	T ($^\circ\text{C}$)	Taille de grain (μm)
0.001	1050	23.35
	1100	67.35
	1150	77.25
	1200	91.52
	1250	105.25
0.01	1050	23.10
	1100	55.6
	1150	63.65
	1200	73.52
	1250	95.23
0.1	1050	21.58
	1100	47.23
	1150	53.25
	1200	65.23
	1250	72.25
1	1050	20.23
	1100	26.58
	1150	36.25
	1200	41.25
	1250	52.31

4.5.2 Modèle EF et implémentation de la DRX

Le modèle de compression utilisé pour comparer les lois de comportement dans la Section 4.3 est utilisé pour la simulation de la microstructure dans cette section à la seule différence qu'un solveur implicite est utilisé cette fois-ci pour réduire le temps de simulation. La loi de comportement RNA, implémentée via la routine utilisateur VUHARD dans Abaqus/Explicit lors de la simulation d'un essai de compression présentée précédemment est cette fois-ci implémentée dans Abaqus/Standard en utilisant la routine utilisateur UHARD. Le principe de calcul de cette routine est le même que celui de la VUHARD. Elle a juste besoin des expressions de la contraintes d'écoulement et ses dérivées respectives par rapport à la déformation ε , vitesse de déformation $\dot{\varepsilon}$ et la température T .

Quant aux X_{drx} et $d_{drx,avg}$, leur implémentation est effectuée via la même routine UHARD en utilisant les variables d'états fournies par cette routine qui permettent de stocker les valeurs calculées à chaque incrément de temps (Voir fin de code suivant) et le reste du programme est identique à celui de la VUHARD présenté en annexe C.

```

1 SUBROUTINE UHARD(SYIELD ,HARD,EQPLAS,EQPLASRT ,TIME ,DTIME ,TEMP,
2   DTEMP,NOEL,NPT,LAYER,KSPT,KSTEP,KINC,CMNAME,NSTATV,STATEV,
3   NUMFIELDV,PREDEF,DPREDEF,NUMPROPS,PROPS)
4   INCLUDE 'ABA_PARAM.INC'
5   CHARACTER*80 CMNAME
6   DIMENSION HARD(3),STATEV(NSTATV),TIME(*),PREDEF(NUMFIELDV),
7   DPREDEF(*),PROPS(*)
8   C-----
9   EPSP = EQPLAS
10  DEPSP = EQPLASRT
11  TEMP = TEMP + DTEMP + 273.15
12  C-----
13  ..... Partie LDC RNA .....
14  C-----
15  C Dynamic recrystallization
16  C-----
17  PARR = 8.314
18  PARZ = DEPSP*exp(437385.0/PARR*TEMP)
19  PAREC = 0.000027129*(PARZ)**0.245
20  PARD = 0.000312841*(PARZ)**0.202
21  if (EPSP < PAREC) then
22    PARDRX = 0.0
23  else
24    PARDRX = 1.0 - exp(-0.440*((EPSP-PAREC)/PARD)**2.850)
25  endif
26  d0 = 350.0
27  n1 = 89403.1813
28  n2 = -0.00033065
29  n3 = -0.000000018481
30  n4 = -0.09674225
31  n5 = -91734.7834
32  PARGRX = n1*d0**n2*EPSP**n3*DEPSP**n4*exp(n5/(PARR*(TEMP)))
33  AVRGRX = PARDRX*PARGRX + d0*(1-PARDRX)
34  STATEV(1) = PARDRX
35  STATEV(2) = PAREC
36  STATEV(3) = PARGRX
37  STATEV(4) = AVRGRX
38  RETURN
39  END

```

Les résultats de ces implémentations sont présentés dans les sections suivantes.

4.5.3 Effet de la température et de la vitesse de déformation sur la DRX

Dans cette partie, nous étudions l'influence de la vitesse de déformation $\dot{\epsilon}$ et de la température T sur la recristallisation en fonction de la réduction de l'échantillon. La Figure 4.25 montre les valeurs de la fraction volumique de la DRX pour différentes réductions et différentes vitesses de déformation pour une température $T_0 = 1150^\circ\text{C}$. On remarque que la DRX augmente avec l'augmentation de la réduction de l'éprouvette mais décroît avec l'augmentation de la vitesse de déformation $\dot{\epsilon}$.

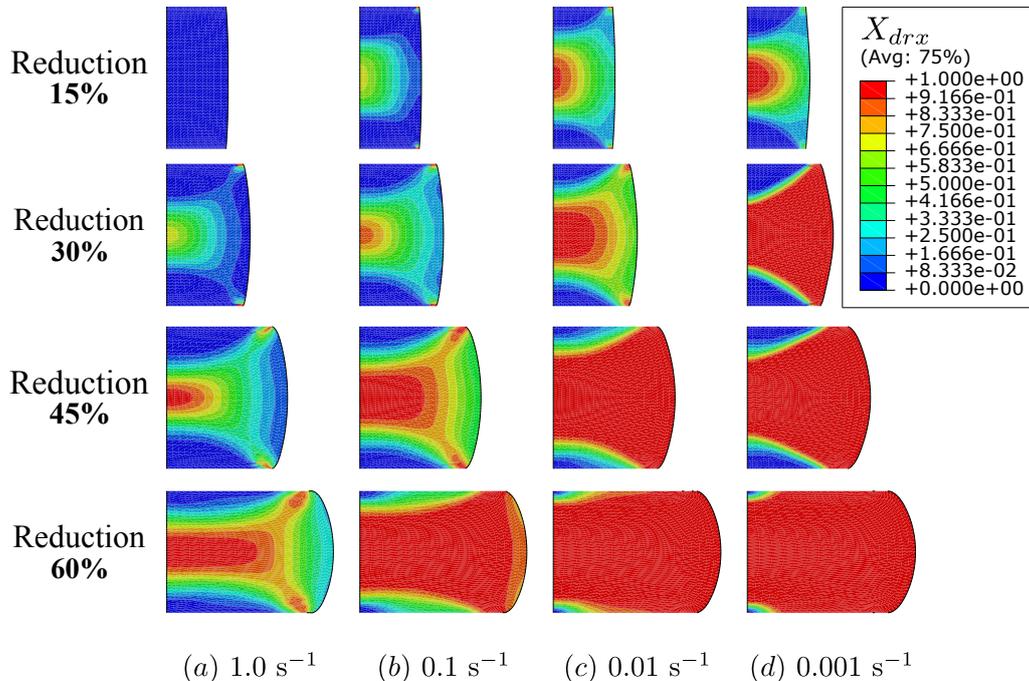


FIGURE 4.25 – X_{drx} prédites pour $T_0 = 1150^\circ\text{C}$ pour différentes réduction et vitesses de déformation

Pour une réduction de 60%, la DRX atteint 100% au centre de l'échantillon quelle que soit la vitesse de déformation $\dot{\epsilon}$. La microstructure recristallisée présentée sur la Figure 4.24 indique qu'un grand nombre de grains recristallisés sont présents au sein de l'échantillon pour une vitesse de déformation $\dot{\epsilon} = 10^{-3} \text{ s}^{-1}$.

La région où la DRX est importante est située au centre de l'échantillon (Figure 4.25); les grains sur les côtés supérieur et inférieur ainsi que sur le côté droit sont relativement moins recristallisés par rapport au centre. La distribution des DRX est inégale en raison d'un effet de frottement et de l'emboutissage inévitablement associé aux essais de compression à chaud sur un cylindre. La Figure 4.26 montre les fractions volumiques de la DRX et la taille moyenne des grains à une vitesse de déformation de $\dot{\epsilon} = 10^{-2} \text{ s}^{-1}$ et pour différentes valeurs de de températures initiales T_0 .

Plus la température augmente, la taille de grains diminue; ce qui signifie que la recristallisation est importante. De même, pour une vitesse de déformation fixe comme le montre la Figure 4.27, on observe globalement que la recristallisation augmente avec la réduction de la taille de l'éprouvette quelque soit la température initiale de l'éprouvette et cela s'explique par l'effort importa appliqué au matériau créant ainsi des dislocations internes au matériau qui se combinent les unes aux autres pour former de grains recristallisés.

Par contre pour une réduction fixée en augmentant la température, on constate que la DRX augmente également. Pour une réduction de 60%, on remarque la recristallisation est presque complète, ce qui valide les résultats de la Figure 4.23 où on observe des grains recristallisés pour une température $T_0 = 1250^\circ\text{C}$.

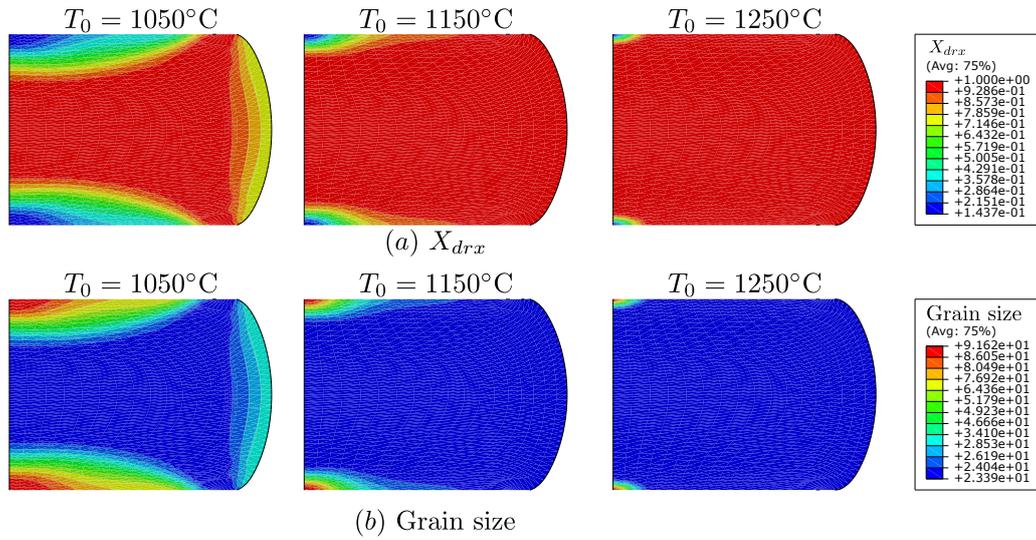


FIGURE 4.26 – Évolution de la X_{drx} et de la taille de grain pour $\dot{\epsilon} = 10^{-2} \text{ s}^{-1}$

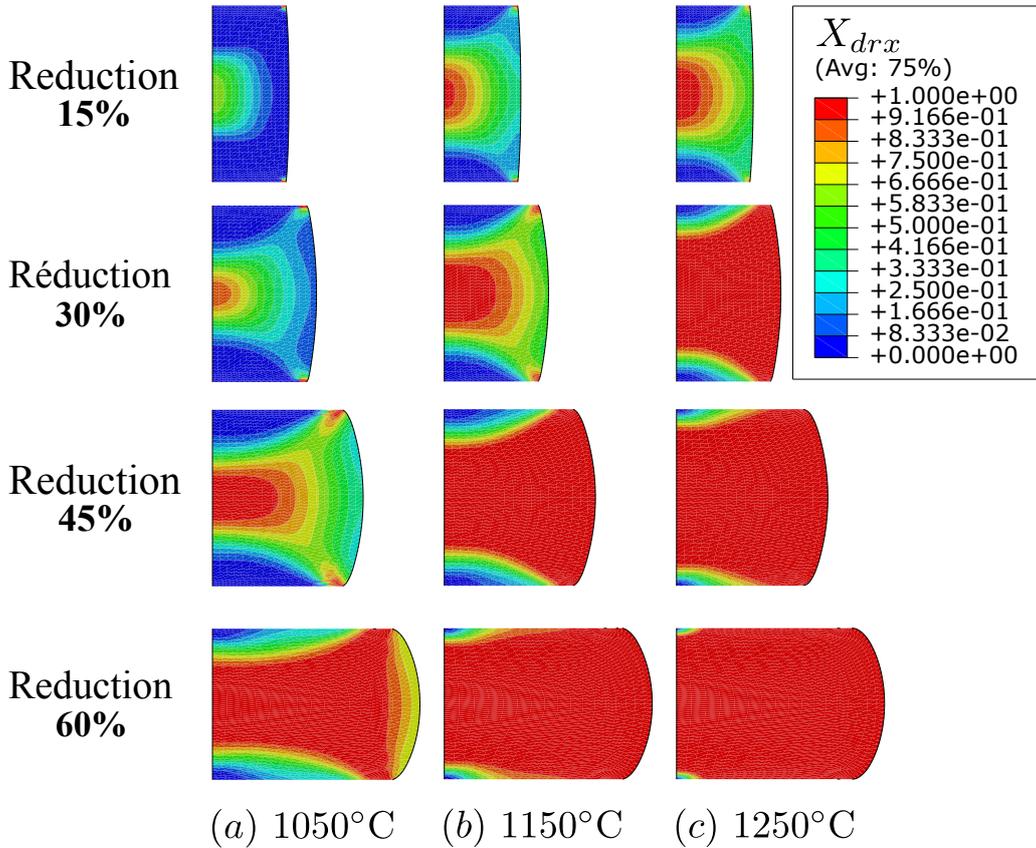


FIGURE 4.27 – X_{drx} prédites pour $\dot{\epsilon} = 10^{-2} \text{ s}^{-1}$ pour différentes réduction et températures initiales

4.5.4 Effet de la température et de la vitesse de déformation sur la taille de grain

La Figure 4.26 (a) montre que la DRX de la région la plus déformée a commencé et s'est achevée plus tôt que celle des régions moins déformées. Au fur et à mesure que T_0 augmente, l'espace occupé par la DRX s'élargit et la fraction volumique pour une DRX complète augmente. La Figure 4.26 (b) montre que la taille moyenne des grains des régions dans lesquelles la DRX est complète a augmenté en fonction de l'augmentation de la température de $T = 1050^\circ\text{C}$ à $T = 1250^\circ\text{C}$. Lorsque la température augmente, la taille des régions où la DRX n'est pas complète, réduit. Ces observations viennent corroborer l'analyse métallographique présentée précédemment.

La Figure 4.28 montre la fraction volumique de la DRX et la taille moyenne des grains à différentes vitesses de déformation pour $T_0 = 1150^\circ\text{C}$.

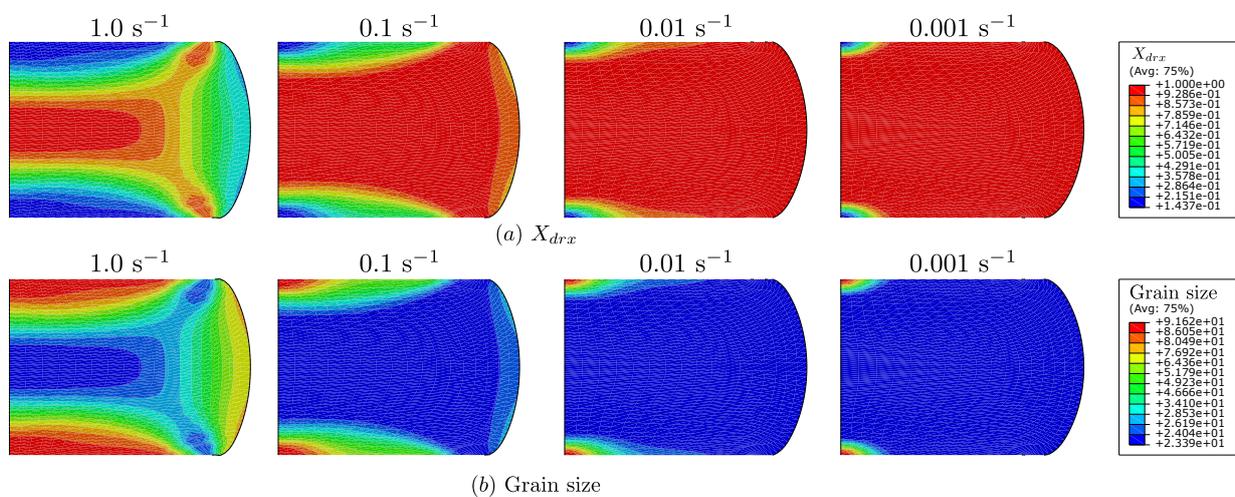


FIGURE 4.28 – Évolution de la X_{drx} et de la taille de grain pour $T_0 = 1150^\circ\text{C}$

La Figure 4.28 (a) montre qu'une augmentation de la vitesse de déformation $\dot{\epsilon}$ réduit la taille de la région dans laquelle la DRX est complète, et donc augmente celle de la région dans laquelle la DRX est incomplète. La Figure 4.28 (b) montre que la taille moyenne des grains a diminué d'environ $54.3\ \mu\text{m}$ à $18.1\ \mu\text{m}$ lorsque la vitesse de déformation passe de $1\ \text{s}^{-1}$ à $10^{-3}\ \text{s}^{-1}$.

Ainsi, les prédictions des éléments finis sont cohérentes avec les observations métallographiques de la Figure 4.25.

Les valeurs expérimentales et prédites de X_{drx} au quatre points sélectionnés sont comparées sur la Figure 4.29 et montrent que la DRX est moins observée à proximité des zones difficiles à déformer (zone morte). Cependant, l'uniformité de la DRX dépend de l'uniformité de la déformation et on constate que dans les régions recristallisées, les valeurs prédites de X_{drx} sont en conformité avec les valeurs expérimentales.

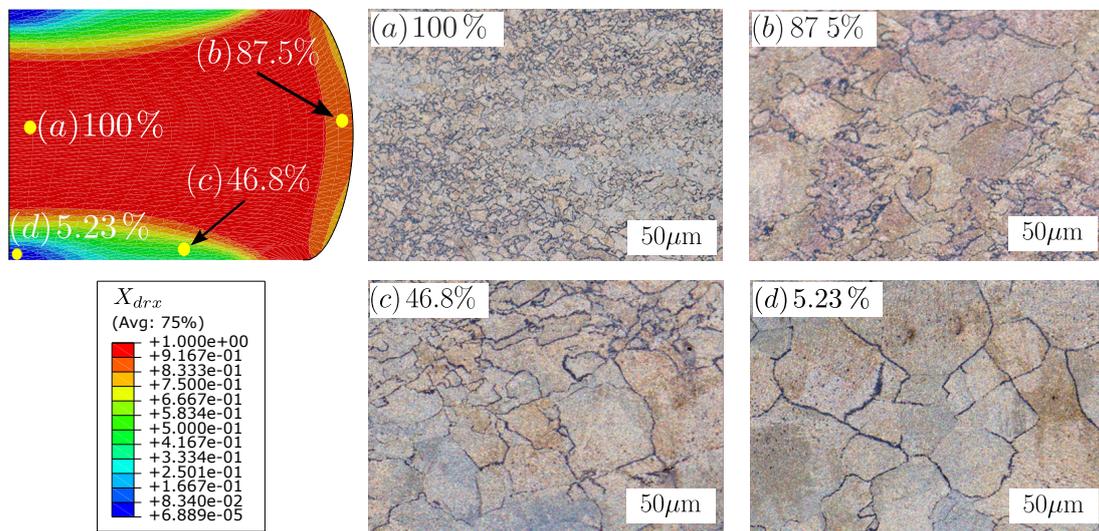


FIGURE 4.29 – Comparaison entre valeurs expérimentales et prédites de X_{drx} à $T_0 = 1150^\circ\text{C}$ et $\dot{\epsilon} = 10^{-2} \text{s}^{-1}$

4.6 Conclusions du chapitre

Ce chapitre a été consacré à la mise en pratique des modèles élaborés au sein de cette thèse. Plus précisément, il a débuté par une exposition détaillée de l'implémentation numérique d'une loi de comportement dans le logiciel Abaqus. Cette démarche a eu pour objectif de confirmer, d'une part, les identifications effectuées dans les chapitres 2 et 3 par le biais de simulations numériques portant sur des essais de rétrécissement d'une barre circulaire et un essai d'impact de Taylor. Les résultats de ces simulations ont permis de démontrer la capacité du modèle RNA à reproduire de manière précise le comportement de l'alliage 42CrMo4, comparativement aux données obtenues antérieurement avec le modèle de Johnson–Cook appliqué à ce même matériau.

La deuxième partie de ce chapitre s'est focalisée sur la simulation d'un essai de compression en utilisant un acier 3Cr2Mo identifié dans le Chapitre 2. L'objectif était de procéder à une étude comparative entre les simulations basées sur les modèles analytiques et celle basée sur les RNA. L'examen de la contrainte de von Mises et de la déformation maximale atteinte par chaque modèle a permis de conclure que le modèle RNA fournit des résultats satisfaisants, avec des valeurs très proches de celles observées lors de l'identification.

Fort de cette efficacité confirmée, la dernière partie du chapitre a exploité le modèle RNA pour simuler l'évolution de la microstructure du matériau tout au long de la simulation, en se concentrant spécifiquement sur la fraction volumique de la recristallisation dynamique et la taille de grain, comme objet de l'étude. Les résultats de la simulation ont été comparés aux données expérimentales et une bonne corrélation a été observée.

Chapitre 5

Modélisation électro-thermomécanique d'un essai Gleeble

5.1	Gradient de température lors d'un essai Gleeble	126
5.2	Modélisation électro-thermique	128
5.2.1	Régulation thermique sur Gleeble	128
5.2.2	Équations électro-thermiques	130
5.2.3	Conditions aux limites du modèle	130
5.2.4	Algorithme de contrôle de température	132
5.2.5	Modélisation électro-thermique	134
5.3	Modélisation thermomécanique	135
5.4	Modélisation électro-thermomécanique	140
5.5	Conclusions du chapitre	142

La simulation numérique des essais réalisés sur le dispositif Gleeble est un complément pour la recherche en métallurgie et en science des matériaux. Le simulateur Gleeble joue un rôle essentiel dans la recherche et le développement de nouveaux alliages, la caractérisation des propriétés des matériaux existants, ainsi que la simulation de procédés de fabrication industrielle tels que le soudage, le formage, la trempe et la fabrication additive. Afin de reproduire le plus fidèlement possible les conditions réelles des essais, il est essentiel de choisir une modélisation multiphysique prenant en compte les variations de la microstructure des matériaux pendant le chauffage et leur influence sur la manière dont ils se déforment sous des contraintes mécaniques. Alors que la modélisation thermomécanique est une approche courante pour simuler des essais thermomécaniques, il est nécessaire ici d'effectuer une modélisation électro-thermomécanique pour reproduire fidèlement les conditions d'essais Gleeble.

Lors des essais Gleeble, le défi majeur réside dans la répartition inégale de la température à l'intérieur de l'éprouvette. Cette variation de température provoque une déformation non homogène, incitant les chercheurs à réduire ce gradient thermique. La modification de la géométrie des mors telle que proposée par [Kim *et al.*, 1996, Ganapathy *et al.*, 2018] pour influencer la répartition du courant électrique est une solution explorée, mais l'élimination complète du gradient reste difficile. La méthode des éléments finis est de plus en plus utilisée pour simuler les essais Gleeble, permettant d'identifier les facteurs influençant l'inhomogénéité de la température. Cependant, les premières simulations comme c'est le cas de [Norris et Wilson, 1999, Brown *et al.*, 1997] supposent souvent une température initiale uniforme, ce qui ne reflète pas la réalité des essais. Des simulations plus représentatives [Zhang *et al.*, 2010, Bennett *et al.*, 2010, Kardoulaki *et al.*, 2014, Xiao *et al.*, 2021, Jiang

et al., 2021] ont permis de prendre en compte la partie électrothermique de ce type d'essai, mais la chaîne de simulation reste incomplète car la partie thermomécanique est effectuée par un autre logiciel ou un autre solveur et les températures initiales sont celles obtenues au dernier incrément des calculs après transfert entre les deux logiciels. Ainsi, il nous a paru important de développer un modèle capable de refléter fidèlement l'ensemble des phénomènes qui se produisent sur le dispositif Gleeble, permettant ainsi de suivre l'intégralité de l'historique du matériau pendant le chauffage puis les essais.

Dans ce chapitre, nous examinerons les raisons du choix d'une modélisation électro-thermomécanique par rapport à une modélisation thermomécanique pour simuler les essais, en mettant en avant l'impact des variations de la microstructure des matériaux pendant le chauffage sur leur comportement en termes de déformation. Cette approche prend en compte de manière plus précise les phénomènes microstructuraux, ce qui est essentiel pour reproduire fidèlement les conditions réelles d'essai.

Pour parvenir à cet objectif, un algorithme a été développé pour contrôler la température de l'éprouvette tout au long de la simulation, en ajustant la densité de courant injectée à chaque pas de temps en fonction des variations de température. Cet algorithme a été intégré dans le logiciel Abaqus grâce à une sous-routine spécifique UAMP. Afin de valider l'implémentation de cet algorithme, les mesures expérimentales de température prises par un thermocouple fixé au centre de l'éprouvette lors des essais sur Gleeble ont été comparées aux données numériques. Une excellente corrélation entre les deux ensembles de données a été observée.

Grâce à cet algorithme, deux modèles numériques ont été développés en utilisant le modèle RNA comme loi de comportement du matériau : un modèle axisymétrique et un modèle 3D. Dans le modèle axisymétrique, deux solveurs ont été utilisés : une simulation électro-thermique traitée de manière implicite, dont les résultats en température ont été transférés vers un solveur thermomécanique explicite pour la partie mécanique.

Un second modèle 3D, électro-thermo-mécanique, complètement couplé en implicite a ensuite été développé. Après comparaison avec les résultats de la simulation thermomécanique, il a été observé que le gradient de température avait une influence significative sur l'état de contrainte et de déformation du matériau 3Cr2Mo.

5.1 Gradient de température lors d'un essai Gleeble

Le principal problème rencontré lors des essais effectués sur le Gleeble réside dans le gradient de température résultant du chauffage par effet Joule qui se forme à la fois le long de l'axe longitudinal et le long de l'axe radial de l'échantillon. Afin de mettre en évidence ce gradient thermique, nous utilisons ici trois thermocouples (comme illustré sur la Figure 5.1).

Un des trois thermocouples (TC1) est positionné au milieu de l'axe longitudinal de l'échantillon, tandis que les deux autres (TC2 et TC3) sont positionnés à une distance égale à 1/4 de la longueur totale de l'éprouvette de part et d'autre du thermocouple central. L'estimation du gradient de température le long de l'axe radial est réalisée numériquement, car il est techniquement complexe d'instrumenter par des thermocouples cet axe de l'échantillon (ceci nécessite de percer l'échantillon et perturbe les mesures et les gradients).

Comme présenté au Chapitre 1, un ensemble de 30 essais de compression à haute température a été effectué sur le simulateur Gleeble-3800 avec cinq températures (1050 °C, 1100 °C, 1150 °C, 1200 °C et 1250 °C) et six vitesses de déformation (10^{-3} s^{-1} , 10^{-2} s^{-1} , 10^{-1} s^{-1} , 1 s^{-1} , 2 s^{-1} et 5 s^{-1}). Des feuilles de tantale ont été utilisées comme matériau lubrifiant entre les enclumes et l'échantillon. Les échantillons ont été chauffés à une température de 1260 °C avec une vitesse de chauffage de 2 °C/s, et ils ont été maintenus à cette température pendant une durée de 5 minutes afin d'éliminer les



FIGURE 5.1 – Utilisation de trois thermocouples fixés sur un échantillon.

gradients de température. Ensuite, les échantillons ont été refroidis à une vitesse de refroidissement de 1°C/s jusqu'à atteindre la température d'essai. Ils ont ensuite été maintenus à cette température constante pendant une période de 1 minute avant toute déformation.

La réponse thermique, obtenue par les trois thermocouples, est représentée sur la Figure 5.2. En zoomant sur la partie où la température est maintenue pendant au moins 10 minutes (un temps initialement choisi pour réduire au maximum le gradient de température mais également permettre aux précipités de dissoudre dans l'austénite), (voir Figure 5.2 (b)), on peut clairement observer que la température n'est pas uniforme le long de l'axe longitudinal.

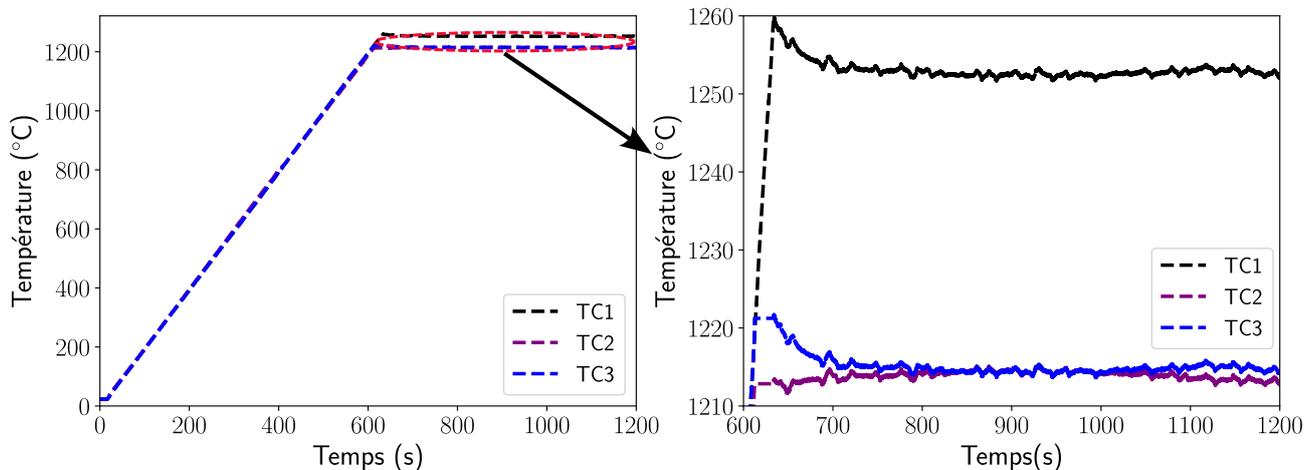


FIGURE 5.2 – Températures enregistrées par les trois thermocouples fixés sur l'échantillon.

L'écart de température entre (TC1) et (TC2 ou TC3) varie entre 30°C et 35°C pendant la phase de maintien. Bien que cet écart puisse paraître faible, il entraîne une déformation non homogène et des conditions initiales de l'essai de compression qui doivent être prises en compte dans notre analyse numérique ultérieure. Il est essentiel de noter que cette différence est observée ici suivant l'axe longitudinal, on peut supposer qu'il en existe une autre le long de l'axe radial, mais cette approche ne permet pas ici de la quantifier. A l'échelle microscopique, cette différence de température a un impact considérable sur les propriétés du matériau.

Afin de prédire numériquement cette distribution de température, il est nécessaire de développer un algorithme capable de d'évaluer la température de manière précise pendant l'essai, reproduisant ainsi les conditions réelles observées sur Gleeble. La section suivante présentera donc l'algorithme que nous avons développé pour la gestion de la densité du courant tout au long de l'essai, après avoir exposé le quelques fondements de cet algorithme.

5.2 Modélisation électro-thermique

L'analyse électro-thermique par éléments finis présentée dans cette section vise à simuler un essai de compression sur Gleeble à haute température afin de prédire le profil de température longitudinal observé expérimentalement dans la Section 5.1. Les champs de température obtenus en sortie de cette simulation serviront de conditions initiales pour la simulation thermomécanique de la compression. Cette modélisation suit le même principe que les travaux de Kardoulaki et al. [Kardoulaki et al., 2014] et Xiao et al. [Xiao et al., 2021]. Les équations qui régissent cette simulation électro-thermique ont été établies sur la base de travaux de [Marc, 2010, Walsh et al., 1986, Coër et al., 2011, Li et al., 2016, Martins et al., 2016, Fekete et Szekeres, 2015].

5.2.1 Régulation thermique sur Gleeble

Le simulateur Gleeble utilise un système de chauffage fonctionnant à partir d'un courant continu par effet Joule. Par conséquent, pour réaliser une simulation numérique d'un essai sur Gleeble, il est essentiel de comprendre le système de contrôle du courant injecté utilisé pour chauffer l'échantillon, de manière à l'intégrer dans le logiciel de calcul par éléments finis. La valeur du courant injecté lors du chauffage de l'éprouvette sur le dispositif Gleeble n'est pas directement mesurable, contrairement à d'autres grandeurs. Étant donné que le courant est transmis à l'échantillon par le biais des mors, il est nécessaire de comprendre le mécanisme de transfert de ce courant depuis les mors vers la pièce à chauffer afin de le modéliser en se basant sur les équations de conduction électrique.

Considérons un conducteur électrique de longueur L , de section S et de résistance électrique R . Lorsqu'il est traversé par un courant électrique d'intensité I , la tension à ses bornes est donnée par $U = RI$ (loi d'Ohm) et la puissance dissipée est $P = UI = RI^2$. Pour une quantité d'énergie ΔE produite durant un temps Δt la puissance s'exprime par $P = \Delta E/\Delta t$, d'où l'expression suivante :

$$P = RI^2 = \frac{\Delta E}{\Delta t} \quad \implies \quad \Delta E = RI^2 \Delta t. \quad (5.1)$$

Par ailleurs on peut écrire que :

$$\Delta E = mC_p \Delta T, \quad m = \rho SL \quad \text{et} \quad R = r \frac{L}{S}, \quad (5.2)$$

avec r la résistivité électrique du conducteur, C_p la chaleur spécifique et ρ la densité du matériau. De ces deux équations et pour une variation de la température ΔT , il vient l'écriture suivante :

$$\rho S^2 C_p \Delta T = r I^2 \Delta t \quad \implies \quad J = \frac{I}{S} = \sqrt{\frac{\rho C_p \Delta T}{r \Delta t}} \quad (5.3)$$

avec $1/r$ la conductivité électrique et J la densité du courant. Pour deux conducteurs électriques en contact (l'éprouvette et un des deux mors comme illustré sur la Figure 5.3 (a)), et si nous notons par S_1 la section de l'éprouvette et S_2 celle des mors comme montré sur la Figure 5.3 (b) dans un cas de compression, alors la densité du courant dans les mors est donnée par la relation suivante :

$$J_2 = \frac{I_2}{S_2} = \frac{I_2 S_1}{S_1 S_2} = \frac{I_1 S_1}{S_1 S_2} = J_1 \frac{S_1}{S_2} = \frac{S_1}{S_2} \sqrt{\frac{\rho C_p \Delta T}{r \Delta t}} \quad \text{car} \quad I_1 = I_2. \quad (5.4)$$

Le facteur S_1/S_2 est facile à calculer puisqu'il s'agit du rapport de la section de deux cylindres. Par contre pour un essai de traction sur Gleeble ou pour une autre géométrie des mors comme montré

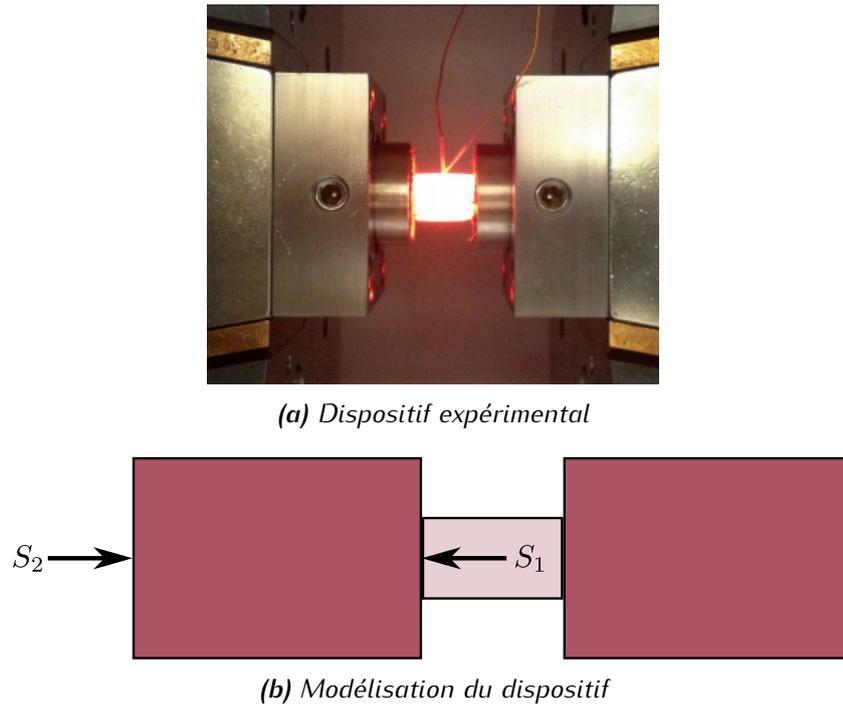


FIGURE 5.3 – Modélisation d'un essai de compression sur le dispositif Gleeble.

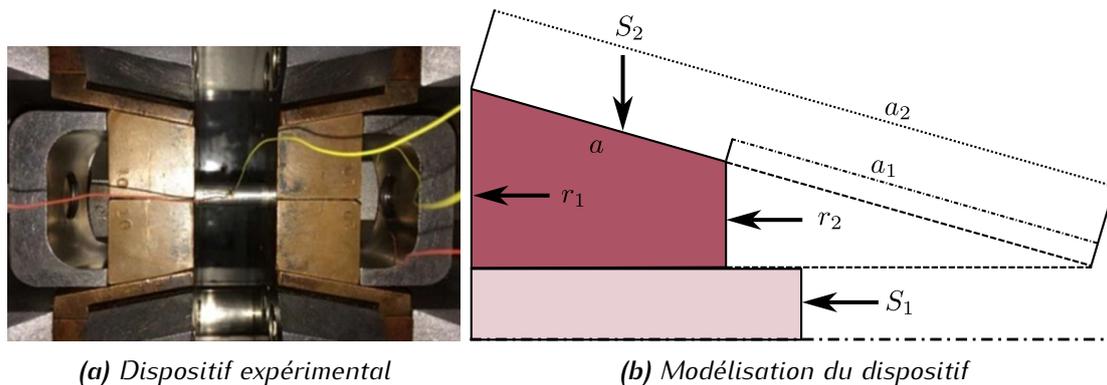


FIGURE 5.4 – Modélisation d'un essai de traction sur le dispositif Gleeble.

sur la Figure 5.4 (a), la surface des mors à considérer est la surface de contact échantillon/mors. Ce facteur S_1/S_2 doit être calculé pour la déduction complète de la densité de courant à injecter lors d'une simulation. Pour cela, ce facteur dépend aussi bien de la surface de l'éprouvette que de celle des mors. A partir de la géométrie décrite sur la Figure 5.4 (b), on peut voir que la surface latérale en question est la surface d'un tronc de cône de révolution donnée par $S_2 = \pi(r_1 + r_2)a$. Les détails de l'obtention de cette surface latérale en Annexe D. A partir du facteur S_1/S_2 , la densité de courant appliquée est déterminée pour chaque essai et utilisée dans une routine Fortran UAMP pour la simulation numérique de l'essai.

Par la suite, bien que des modélisation numériques d'essais de traction aient été réalisées au cours de ce travail de thèse, nous nous limiterons volontairement au seul cas de la simulation numérique d'un essai de compression, en relation avec les essais expérimentaux réalisés au Chapitre 1, tel qu'illustré sur la Figure 5.3.

5.2.2 Équations électro-thermiques

5.2.2.1 Génération de chaleur par effet Joule

Sur Gleeble, le chauffage est effectué par effet Joule et le courant utilisé est un courant continu en régime permanent. En régime permanent, la forme générale de la loi d'Ohm, de la loi de conservation de la charge et de la loi de Joule s'écrivent sous la forme suivante :

$$\begin{cases} J = \sigma E + J_e \\ \nabla J = 0 \\ Q_e = J \cdot E \end{cases}, \quad (5.5)$$

où J_e est la densité de courant externe (A/m^2), E est l'intensité du champ électrique ($A \cdot m^{-1}$), et se calcule à partir de la relation entre l'intensité du champ électrique et Q_e (W/m^3) la chaleur dégagée par effet joule.

5.2.2.2 Transfert de chaleur

Le transfert de chaleur en régime permanent se traduit par l'équation suivante :

$$\nabla \cdot (q_{con} + q_h + q_r) = Q_e, \quad (5.6)$$

où q_{con} , q_h , et q_r sont les flux de chaleur générés par la conduction, la convection et le rayonnement dans le solide respectivement.

Le flux de chaleur se calcule à l'aide des équation suivantes :

$$\begin{cases} q_{con} = -k \nabla T \\ -n \cdot q_h = h(T_{ext} - T) \\ -n \cdot q_r = \epsilon k_B (T_{amb}^4 - T^4) \end{cases}, \quad (5.7)$$

où k est la conductivité thermique (W/mK), h (W/m^2K) est le coefficient de transfert de chaleur par convection, T_{ext} (K) est la température externe à la pièce qui est identique à la température ambiante dans le cas d'un essai Gleeble, ϵ est l'émissivité du matériau de valeur $\epsilon = 0.8$ dans le cadre cette étude. k_B est la constante de Boltzmann dont la valeur est de $5.67 \times 10^{-8} W/m^2K^4$. T_{amb} (K) est la température ambiante.

5.2.3 Conditions aux limites du modèle

5.2.3.1 Matériau et conditions initiales

Les propriétés physiques du matériau, qui dépendent de la température, ont été obtenues à l'aide du logiciel **JmatPro** [Saunders *et al.*, 2003] à partir de la connaissance de la composition chimique du matériau (voir 1.2, page 21) donnée dans le Chapitre 1 et sont illustrées sur la Figure 5.5. Les mors sont en tungstène et la conductivité électrique qui dépend de la température est reportée dans le Tableau 5.1.

La conductivité thermique du tungstène est $70.62 W/mK$, sa chaleur spécifique est $480 J/gK$, et la densité volumique est $14900 kg/m^3$.

La densité de courant nécessaire à chaque incrément de temps est régulée par un algorithme spécifique développé et détaillé dans la suite de ce chapitre. La température initiale de l'éprouvette et des mors est fixée à $T_0 = 23^\circ C$.

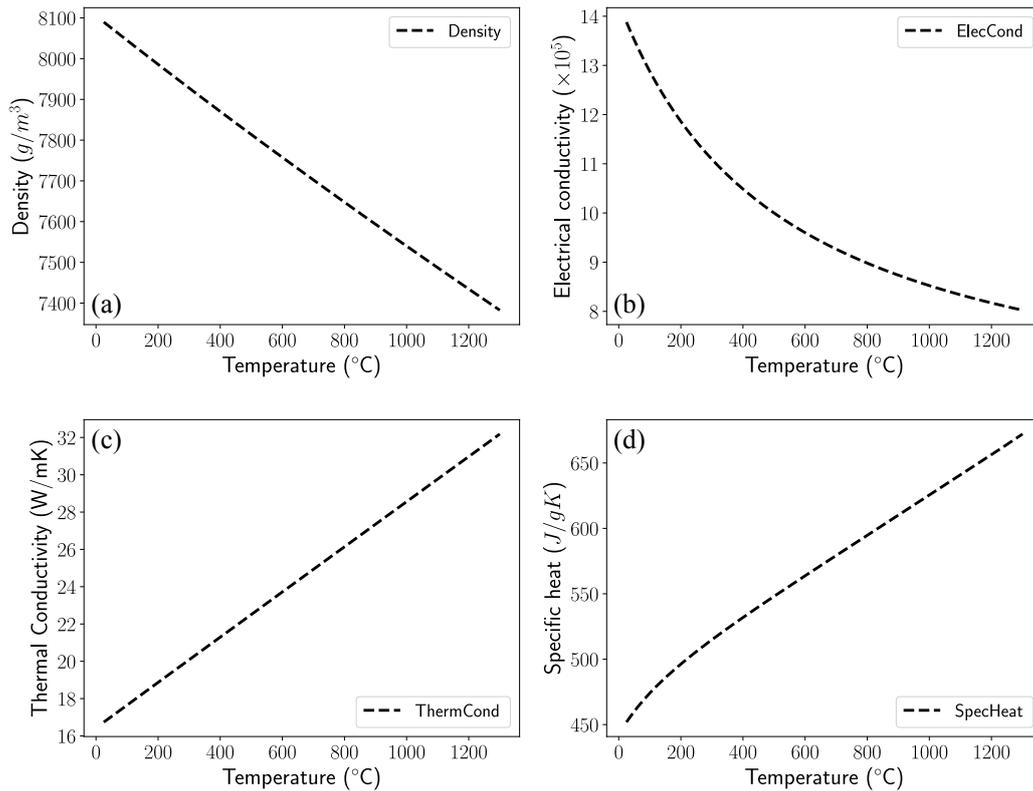


FIGURE 5.5 – Propriétés physiques de l'acier 3Cr2Mo.

TABLE 5.1 – Conductivité électrique du tungstène [Xiao et al., 2021]

Température (°C)	100	200	300	400	500	600	700	800	900
$1/r$ ($\mu\Omega^{-1}m^{-1}$)	0.41	0.31	0.32	0.26	0.24	0.12	0.12	0.12	0.11

5.2.3.2 Conditions aux limites pour la transmission électrique

On suppose que le contact entre l'éprouvette et les mors est plus ou moins parfait et la valeur de la conduction électrique est donnée par l'équation suivante :

$$n \cdot J = h_c (V_g - V_s), \quad (5.8)$$

où J est la densité du courant à l'interface, h_c est coefficient de transfert électrique et la valeur considérée dans cette étude est $3 \times 10^8 S/m^2$ [Zhang et al., 2010], V_g et V_s sont les potentiels des mors et de l'éprouvette à l'interface. Mis à part les surfaces S_{g-1} et S_{g-2} , toutes les autres surfaces en contact avec l'air sont considérées comme des surfaces d'isolation dont la description est donnée par l'équation suivante :

$$n \cdot J = 0. \quad (5.9)$$

5.2.3.3 Conditions aux limites pour le transfert de chaleur

Dans une machine Gleeble, le dispositif de refroidissement par eau installé dans les mors est le principal système de refroidissement. Il y a un échange important de chaleur entre S_{g-1} et S_{g-2}

et le flux de chaleur est significativement lié à la température des mors. L'équation décrivant cet échange est donnée par la relation suivante présentée dans les travaux de Martins [Bäker, 2006] et Kardoulaki [Kardoulaki *et al.*, 2014] :

$$q_g = h_g (T_g - T_0), \quad (5.10)$$

où q_g (W/m^2) est le flux de chaleur sur les surfaces S_{g-1} ou S_{g-2} , h_g est le coefficient de transfert de chaleur correspondant fixé à $h_g = 4000 W/m^2K$ [Xiao *et al.*, 2021], T_g est la température dans les mors, T_s est la température de l'éprouvette au cours de la simulation et T_0 est la température initiale de l'éprouvette fixée à $T_0 = 23^\circ C$.

Comme le contact entre l'éprouvette et les mors est considéré comme un contact parfait, il en découle que les températures T_g et T_s sont identiques. Par conséquent, le coefficient de transfert de chaleur a un effet minimal. L'équation de transfert de chaleur entre l'éprouvette et les mors est donnée par :

$$q_{g-s} = h_{g-s} (T_g - T_s), \quad (5.11)$$

où q_{g-s} (W/m^2) est le flux de chaleur sur la surface S_{s-g} , h_{g-s} est le coefficient de transfert de chaleur correspondant dont la valeur est $2000 W/m^2K$ [Brown *et al.*, 1997]. Pour les autres surfaces en contact avec l'air, l'équation peut être décrite par le transfert radiatif, qui s'écrit comme suit :

$$q_{s-air} = \varepsilon \sigma (T^4 - T_{amb}^4), \quad (5.12)$$

où q_{s-air} (W/m^2) est le flux de chaleur sur la surface S_{s-air} .

5.2.4 Algorithme de contrôle de température

Après avoir brièvement présenté les équations qui régissent le couplage entre les équations électriques et thermiques, dans cette section nous détaillons la mise en place de l'algorithme de contrôle de la densité de courant permettant de piloter les conditions aux limites du modèle électro-thermique. La modélisation considérée est axisymétrique et le modèle est semi-couplé. Nous avons d'une part un modèle implicite qui simule le couplage électrothermique pour obtenir la distribution de la température et d'autre part un modèle explicite pour la simulation thermomécanique.

Pour cette étude, le modèle élément finis axisymétrique présenté sur la Figure 5.6 comprend un assemblage d'une éprouvette de compression et deux mors qui sont modélisés comme des corps déformables. Les éléments DCAX4E (élément quadrilatère axisymétrique à 4 nœuds avec couplage thermo-électrique) sont utilisés pour le maillage du modèle car ils permettent de prendre en compte le chauffage par effet Joule. Le maillage de l'éprouvette est composé de 1 875 éléments et celui des mors de 882 éléments.

Dans cette analyse, l'échantillon est chauffé par effet joule et la densité de courant est appliquée sur la surface A comme montré sur la Figure 5.6 (a), qui est en fait la surface de contact des mors avec les plaques sur lesquelles est appliqué le système de refroidissement. Pour contrôler la température au cours de la simulation, un nœud est sélectionné exactement à la même position que celle du thermocouple TC1 lors de l'expérience. La valeur de la température mesurée à cette position est utilisée pour ajuster la densité de courant à injecter à chaque incrément de temps.

La différence de température ΔT , la densité du matériau ρ , la conductivité électrique $1/r$, la chaleur spécifique C_p et l'incrément de temps Δt sont utilisés pour calculer la densité de courant utilisé lors de la simulation. Les équations correspondantes ont été implémentées dans une routine utilisateur UAMP pour Abaqus/Standard décrite sur la Figure 5.8 où t_a , t_b et t_c tels que illustrés sur la Figure 5.7 représentent la fin du chauffage, la fin du maintien de l'éprouvette à une température fixée et la

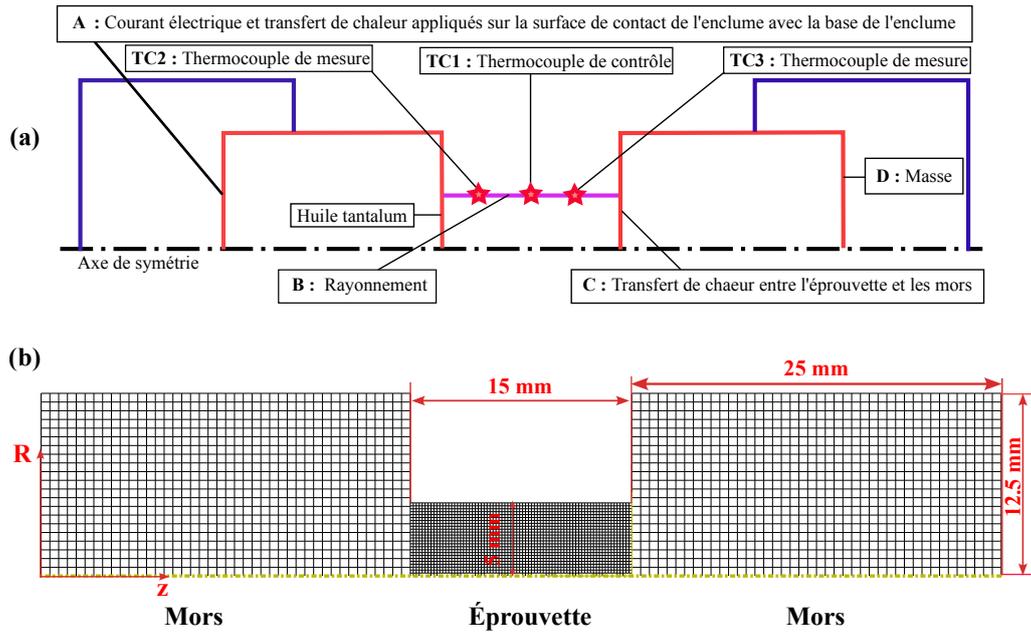


FIGURE 5.6 – Modèle numérique axisymétrique avec 3 thermocouples.

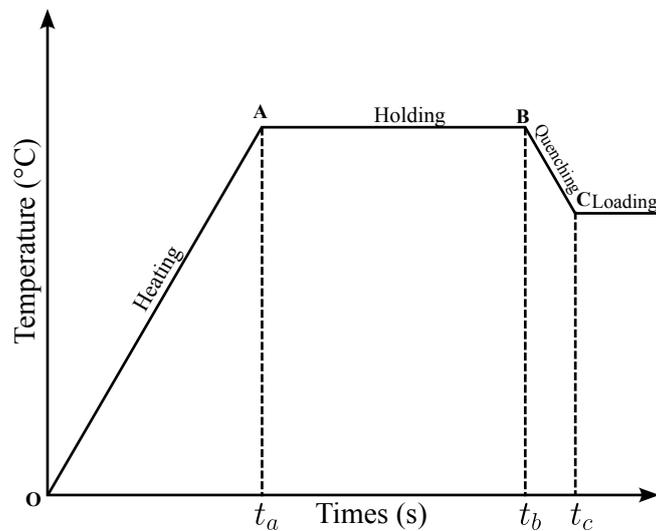


FIGURE 5.7 – Cycle thermique d'essai sur Gleeble.

fin du refroidissement. Cette routine est utilisée pour calculer en temps réel la densité de courant appliquée au système. Elle permet de compenser les pertes de chaleur par conduction au niveau des mors en tungstène en contact avec les plaques en cuivre constamment refroidies à l'eau. La perte de chaleur de l'éprouvette vers les mors est modélisée par le transfert de chaleur par conduction et le contact est supposé parfait. Les essais sont réalisés sous vide et il n'y a donc pas de pertes de chaleur par convection dans l'environnement. Bien qu'il y ait une perte de chaleur superficielle due au rayonnement, elle n'a pas été prise en compte pour simplifier les modèles. Le contrôle de la densité de courant se fait par compensation de la différence de température entre la valeur théorique et la valeur mesurée par le thermocouple.

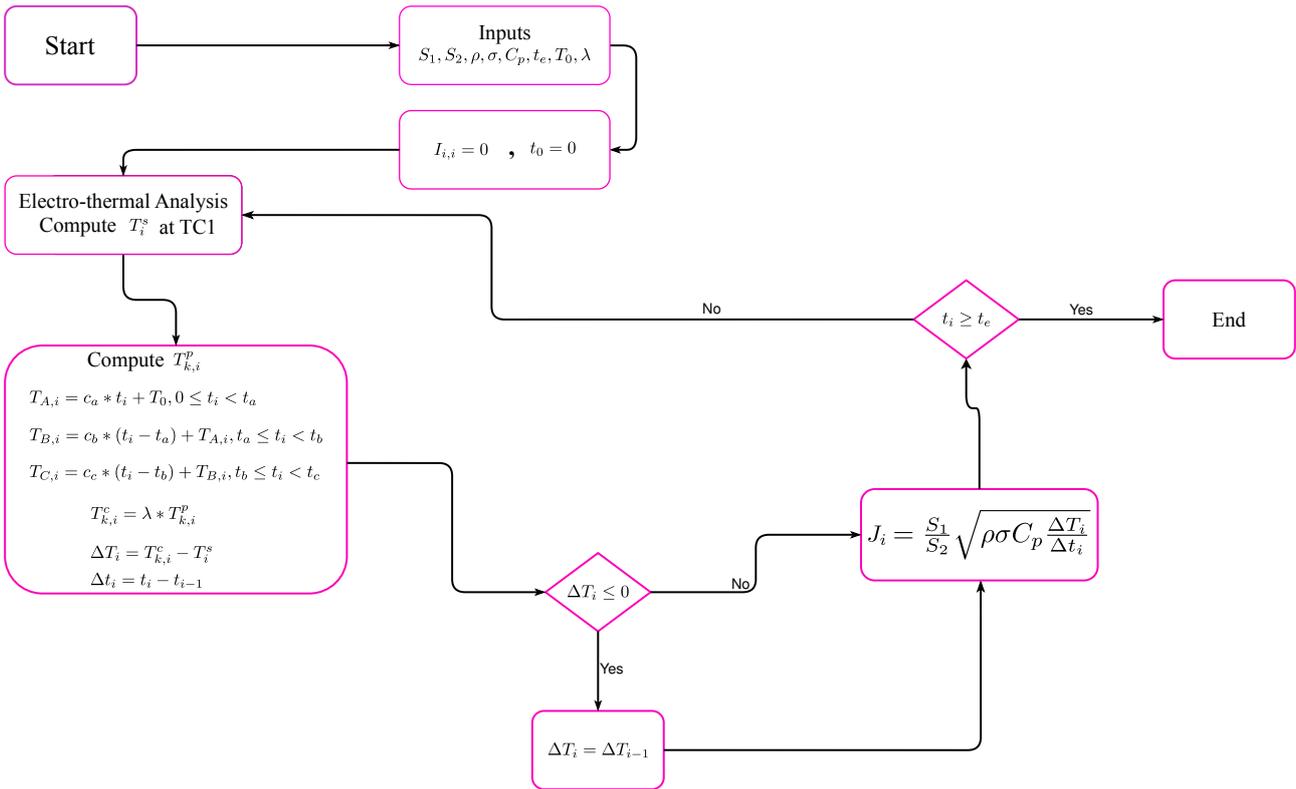


FIGURE 5.8 – Organigramme décrivant le fonctionnement de la subroutine UAMP

5.2.5 Modélisation électro-thermique

Le modèle électro-thermique développé dans cette partie est capable de reproduire le gradient de température axial le long de l'éprouvette tel qu'il peut être observé expérimentalement par le biais des trois thermocouples. La Figure 5.9 illustre le profil de température calculé numériquement et comparé aux valeurs enregistrées à partir des thermocouples TC1, TC2 et TC3 pour deux temps de maintien différents. La Figure 5.9 (a) concerne la simulation numérique d'un chauffage avec un temps

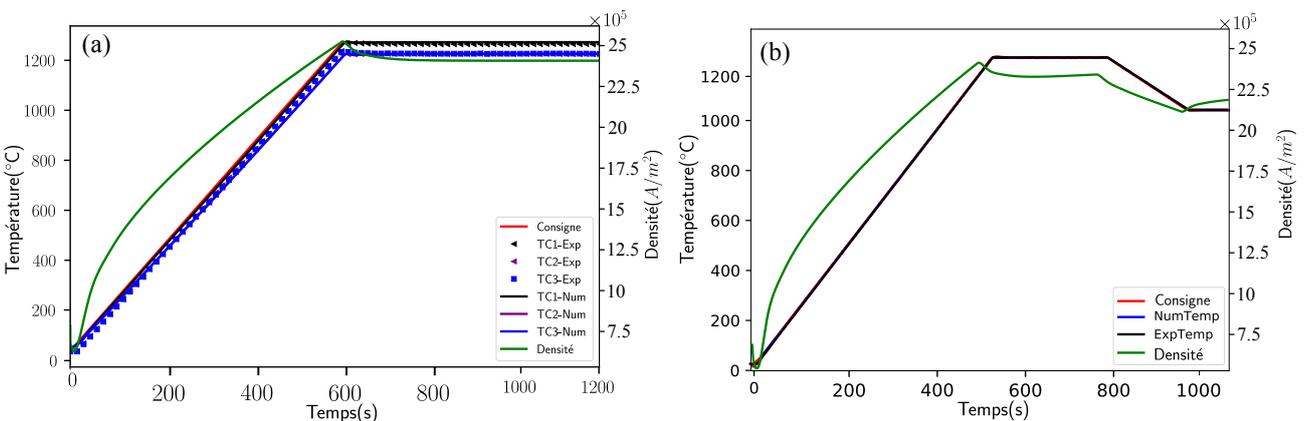


FIGURE 5.9 – Comparaison entre prédiction numérique et valeurs expérimentales de la température. (a) 10 minutes de maintien à 1260°C et (b) 5 minutes de maintien à 1260°C puis refroidissement jusqu'à la température de 1050°C.

de maintien fixé à 10 minutes à la température de 1260°C. La Figure 5.9 (b) concerne le cas d'un maintien pendant 5 minutes à 1260°C, suivi d'un refroidissement à la vitesse de 1°Cs⁻¹ jusqu'à

1050 °C, avec l'utilisation d'un seul thermocouple. L'analyse de ces courbes montre l'efficacité de la routine de régulation pour reproduire les conditions expérimentales des essais. Les prédictions à chaque nœud de mesure correspondent aux valeurs expérimentales des thermocouples fixés à la même position et donc une bonne corrélation existe entre l'expérience et les valeurs numériquement obtenues. Ces résultats valident la pertinence de l'approche proposée pour contrôler la densité de courant.

Une analyse fine des résultats présentés sur la Figure 5.10, montre que les températures mesurées par les trois thermocouples sont relativement bien prédites. Les valeurs numériques au niveau des

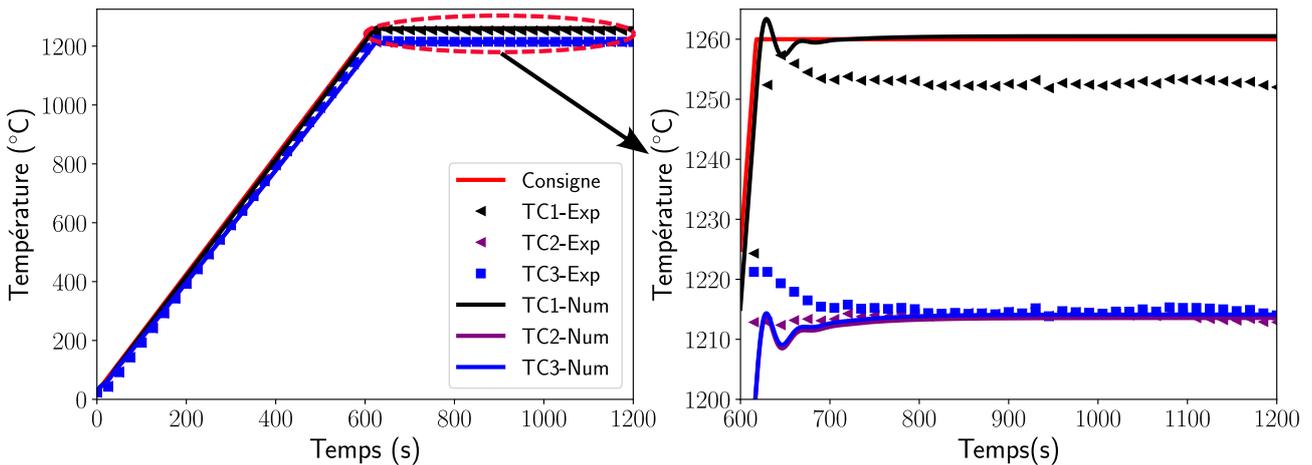


FIGURE 5.10 – Comparaison entre prédiction numérique et valeurs expérimentales de la température avec trois thermocouples

deux thermocouples TC2 et TC3 suivent très bien les valeurs mesurées expérimentalement. Par contre, il existe un écart entre la valeur mesurée par le thermocouple TC1 et la valeur numérique (qui elle est parfaitement en adéquation avec la consigne), preuve que le simulateur Gleeble a des difficultés à imposer une température égale à la consigne donnée par l'utilisateur. On peut ainsi conclure que l'algorithme de régulation et son implémentation via la routine UAMP est pertinente quant au contrôle de température au cours du chauffage.

La distribution de la température au sein de l'éprouvette est donnée sur la Figure 5.11, et montre que la température est maximale au centre de l'éprouvette, et non en surface de celle-ci.

5.3 Modélisation thermomécanique

Une fois que la distribution de la température au sein de l'éprouvette est obtenue, on peut évaluer son impact sur la déformation et les contraintes. Pour cela, un modèle thermomécanique basé sur le solveur Abaqus/Explicit est utilisé pour simuler un essai de compression à haute température sur Gleeble en considérant le même modèle avec le même maillage que celui utilisé lors du chauffage. L'utilisation d'un modèle identique permet d'extraire toutes les températures prédites par le modèle électro-thermique et de les introduire en tant que champ thermique prédéfini dans le modèle thermomécanique.

Un essai de compression à haute température est simulé en utilisant un chargement du type déplacement imposé pour une vitesse de déformation $\dot{\epsilon} = 10^{-2} \text{ s}^{-1}$. Des éléments axisymétriques couplés thermiquement à quatre nœuds de type CAX4RT ont été utilisés. Le temps de simulation est de $t = 85 \text{ s}$ et les résultats numériques concernant les champs de la déformation plastique et de contrainte de von Mises pour une température de 1050 °C sont données sur la Figure 5.12. On observe

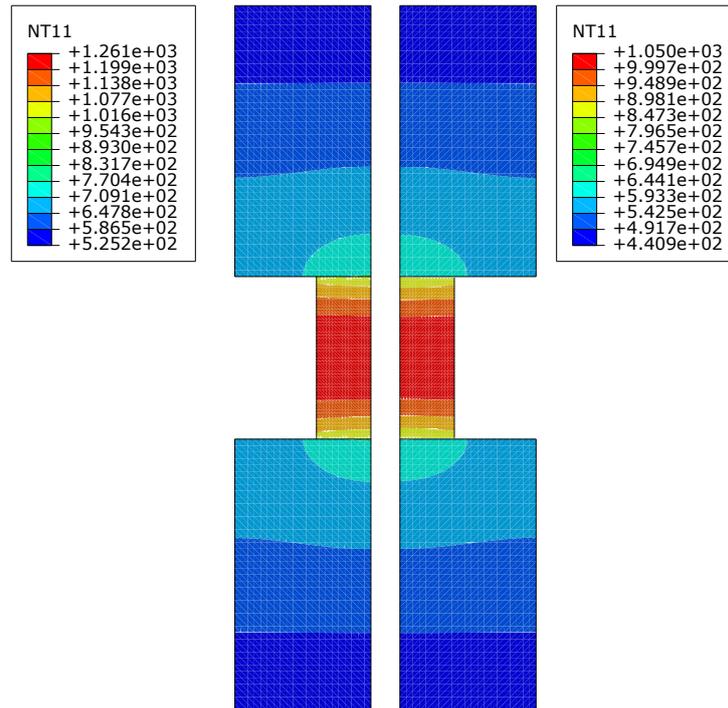


FIGURE 5.11 – Distribution de la température : (gauche) 10 minutes de maintien à 1260°C et (droite) 1 minute de maintien à 1050°C.

que la déformation plastique maximale ε_{max} se trouve au centre avec une valeur de $\varepsilon = 1.232$ alors que la contrainte de von Mises quant à elle est beaucoup plus répartie dans l'éprouvette alors que sa valeur minimale se trouve en surface de l'éprouvette. Ces répartitions de la déformation plastique et de la contrainte de von Mises s'expliquent simplement par la distribution de la température non uniforme au sein de l'éprouvette comme montré sur la Figure 5.11. En effet, la température est maximale au centre et se dissipe de l'intérieur vers la surface du matériau au cours de la déformation.

Le comportement plastique du matériau est décrit par le modèle RNA présenté dans le Chapitre 3. Ce choix a été justifié par le fait qu'il est le meilleur modèle pour la prédiction des valeurs expérimentales sur les courbes contrainte-déformations au détriment des modèles analytiques. Ce modèle a été implémenté dans Abaqus/standard cette fois-ci via une routine UHARD.

Pour bien apprécier la différence entre ce type de modélisation et celui habituellement utilisé pour simuler certains processus industriels via une analyse thermomécanique, nous allons comparer les résultats issus d'une simulation électro-thermique suivie de thermomécanique (ET-TM) avec les résultats issus d'une simulation thermomécanique simple (TM) pour laquelle la température initiale de l'éprouvette est fixée à 1050°C.

Les Figures 5.13 et 5.14 présentent une étude comparative entre une ET-TM et une étude simplement TM en fonction de la valeur de réduction de l'éprouvette sur la déformation plastique et la contrainte de von Mises. Trois niveaux sont comparés : 15%, 30% et 45% de réduction de l'éprouvette. On observe sur la Figure 5.13 que les déformations plastiques, sont presque identiques quant à la valeur maximale et la localisation de cette valeur maximale qui est au centre de l'éprouvette. Par contre leurs valeurs minimales sont localisées aux points de contact intérieurs entre l'éprouvette et les mors. On peut observer que la zone de localisation de la valeur minimale de la déformation plastique est beaucoup plus importante dans le cadre de la simulation ET-TM que la simulation TM. La forme tonneau de l'éprouvette à la fin de la simulation est beaucoup plus prononcée lors de la simulation ET-TM que sur la simulation TM et ceci s'explique toujours par la distribution de la température au sein de l'éprouvette. S'agissant de la contrainte de von Mises pour les deux simulations, on remarque que

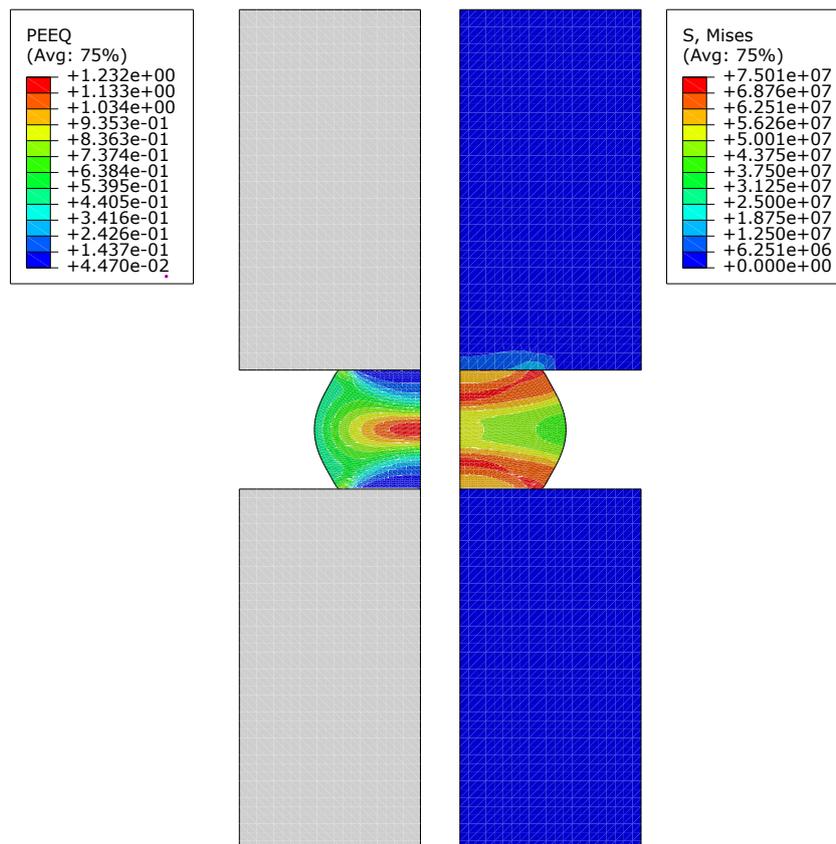


FIGURE 5.12 – Contrainte de von Mises et déformation plastique à la température 1050 °C et une vitesse de déformation 10^{-2} s^{-1}

les distributions ne sont pas similaires. Pour la simulation ET-TM la valeur maximale de la contrainte de von Mises est localisée pratiquement au même endroit au fur et à mesure que la réduction de l'éprouvette augmente alors et la valeur minimale s'étend du centre vers la surface de l'éprouvette et ces distributions sont une conséquence directe de la diffusion de la chaleur de l'intérieur vers l'extérieur permettant ainsi une baisse légère de température.

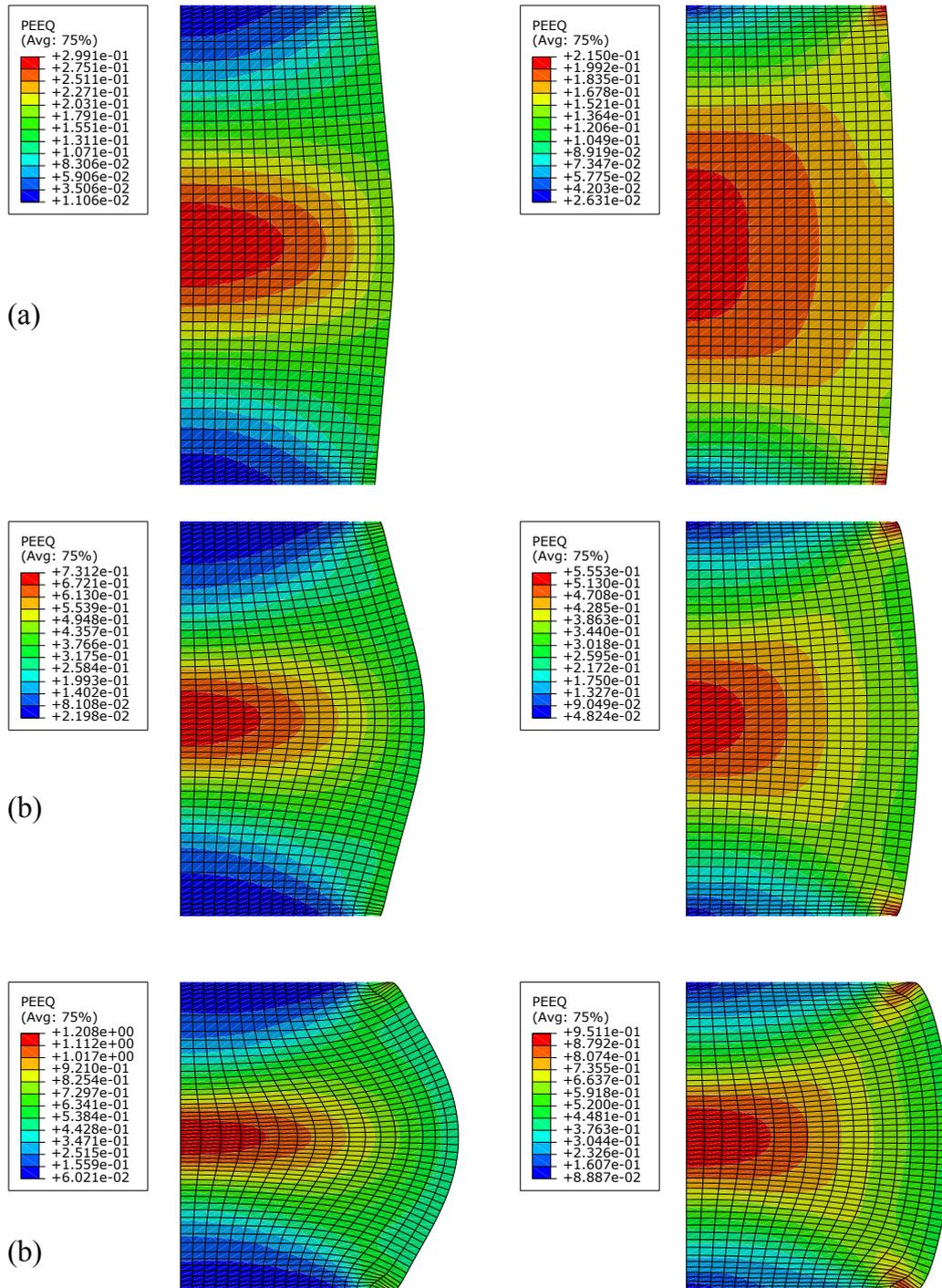


FIGURE 5.13 – Effet de réduction de l'éprouvette et de la distribution de la température sur la déformation plastique pour un essai de compression effectuée à une vitesse de déformation 10^{-2} s^{-1} et une température 1050°C pour le modèle électrothermique + thermomécanique (ET-TM) (gauche) et un modèle thermomécanique (TM) (droite) : (a) 15% de réduction - (b) 30% de réduction - (c) 45% de réduction.

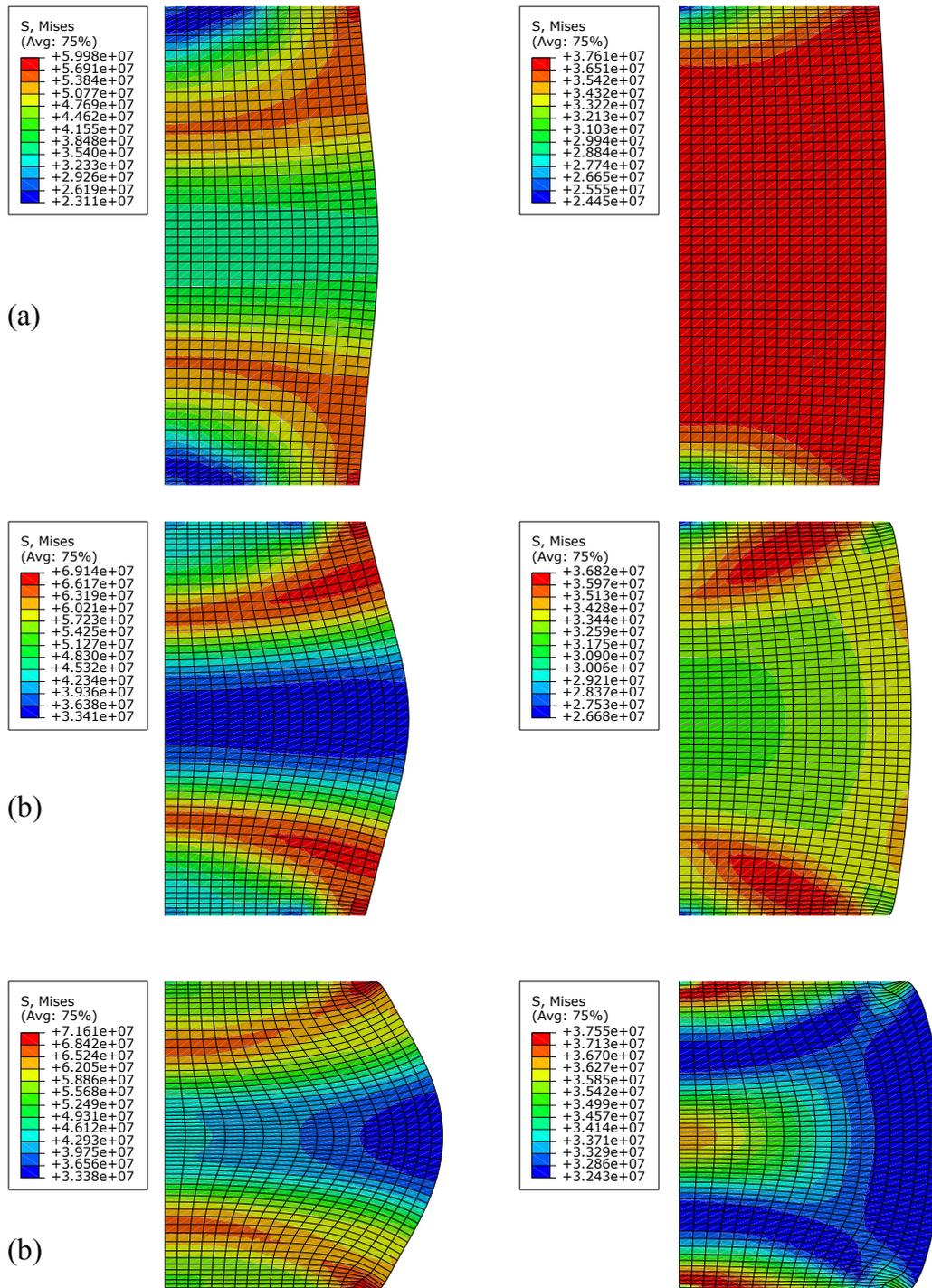


FIGURE 5.14 – Contrainte de von Mises et déformation plastique à la température 1050 °C et une vitesse de déformation 10^{-2} s^{-1} pour le modèle électrothermique + thermomécanique (ET-TM) (gauche) et un modèle thermomécanique (TM) (droite) : (a) 15% de réduction - (b) 30% de réduction - (c) 45% de réduction.

5.4 Modélisation électro-thermomécanique

Dans les sections précédentes, la modélisation électro-thermique suivie de thermomécanique (ET-TM) a été présentée ainsi qu'une comparaison avec une modélisation thermomécanique (TM). A l'issue de cette comparaison, on peut relever qu'il existe une différence entre les résultats obtenus en terme de distribution de la déformation plastique et de la contrainte de von Mises. Cependant, on ne peut pas se prononcer à ce stade sur celle qui décrit la réalité expérimentale dans le sens où aucune de ces deux approches ne reproduit exactement l'expérience. On parle de l'expérience limitée à ce qui se produit sur le simulateur Gleeble et non la réalité expérimentale au sens industriel lors de d'un processus de mise en forme à haute température. Dans cette partie, il est donc question de développer un modèle qui se rapproche plus de l'expérience sur le Gleeble et valider les résultats obtenus avec les données expérimentales.

A titre de rappel, la modélisation électro-thermomécanique (ETM) est une approche qui permet de décrire le comportement d'un système en tenant compte des interactions entre les phénomènes électriques, thermiques et mécaniques. Cette modélisation est souvent utilisée dans le domaine de l'électromécanique, où les dispositifs et systèmes sont soumis à des sollicitations électriques et thermiques qui influencent leur comportement mécanique.

La seule façon de développer un modèle électro-thermomécanique complètement couplé dans le logiciel éléments finis Abaqus est de concevoir un modèle 3D, du fait d'une limitation concernant la bibliothèque d'éléments disponibles. Ainsi, nous allons reprendre le modèle développé dans les sections précédentes avec les mêmes propriétés à la seule différence que dans cette partie, la modélisation s'effectue en 3D et en un seul step.

Pour cette partie, la Figure 5.15 est le modèle éléments finis mis en œuvre comprenant 3 quarts de cylindres alignés dont le plus petit représente l'éprouvette et les deux autres sont la modélisation des mors. Chacun de ces cylindres est modélisé comme un corps déformable.

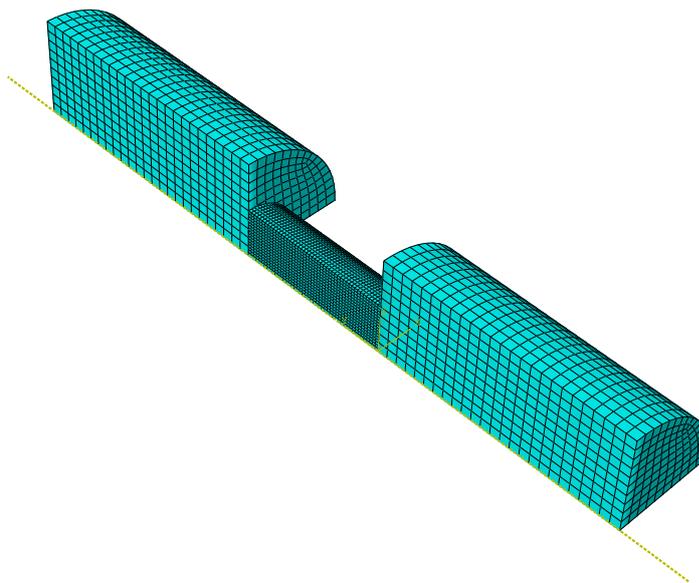


FIGURE 5.15 – Modélisation 3D du couplage électro-thermomécanique

Le maillage de l'éprouvette est composé de 11 280 éléments Q3D8 (éléments hexaédriques linéaires) alors que celui des mors est composé de 2 400 éléments. Les conditions aux limites pour la simulation

restent identiques à celles utilisées dans la simulation précédente. Pour la description du comportement plastique du matériau le modèle RNA est conservé comme c'était le cas dans la simulation partiellement couplée.

Le contrôle de la température est assuré par l'algorithme présenté dans les sections précédentes et un exemple de la distribution de la température obtenue avant la déformation est donnée sur la Figure 5.16. Il ressort de l'analyse de cette figure que la température est maximale au centre

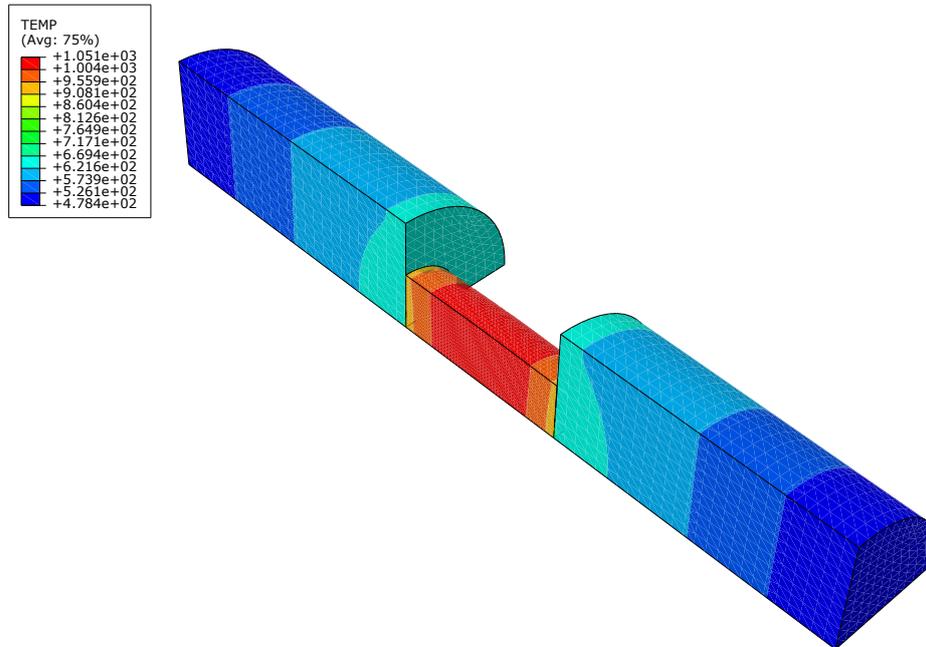


FIGURE 5.16 – Distribution de la température à 1050 °C avant la déformation pour une vitesse de déformation 10^{-2} s^{-1}

de la pièce alors qu'elle est plus basse aux extrémités. Quant à la distribution de la température à travers les mors, son intérêt n'est pas pris en compte dans cette étude d'où la présentation des résultats uniquement liés à l'éprouvette dans la suite de l'étude. Pour comparer les résultats du modèle électro-thermomécanique, un autre modèle thermomécanique 3D est effectué pour lequel la température est uniforme.

Les résultats de cette simulation sont obtenus en utilisant une vitesse de déformation de $\dot{\epsilon} = 10^{-2} \text{ s}^{-1}$ et une température $T = 1050 \text{ °C}$. La déformation plastique calculée à partir des simulations d'essai de compression pour les champs de températures non isotherme et isothermes est donnée sur la Figure 5.17 pour différentes réductions du cylindre. Les résultats de la Figure 5.17 montrent que les différences entre les déformations maximales augmentent de manière significative lorsque le taux de réduction de l'éprouvette augmente (entre 45% à 60% de réduction). Pour ce cas particulier, une déformation axiale maximale élevée ($\epsilon = 1.5$ pour un déplacement global de 4.5 mm) est atteinte dans le cas où un profil de température est calculé par un modèle éléments finis. La déformation axiale maximale simulée correspondante pour le cas isotherme est d'environ $\epsilon = 0.97$. Il faut aussi remarquer que la carte de déformation plastique pour ce cas de simulation diffère en terme d'uniformité.

Bien que les déformations axiales prédites soient relativement plus faibles dans le cas du champ isotherme (TM), les valeurs de déformation sont réparties plus uniformément à travers l'éprouvette en terme de zone de distribution. En revanche, les valeurs les plus élevées des déformations axiales de la simulation de champ thermique calculée par éléments finis sont concentrées dans la zone centrale de l'éprouvette et un changement s'observe lorsque la réduction de l'éprouvette devient plus

importante. On constate dans ce cas que les valeurs importantes de la déformation sont localisées aux extrémités de l'éprouvette ; ce qui n'est pas le cas pour la simulation isotherme (TM). Pour les mêmes conditions, lorsque la réduction la plus faible (15%) est considéré, la déformation maximale pour le cas non isotherme (ETM) est de $\epsilon = 0.34$ alors que la déformation obtenue pour le cas isotherme (TM) est de $\epsilon = 0.23$. Ici, les valeurs de déformation diffèrent moins, mais cette différence a une valeur significative sur l'état de contrainte final.

On remarque des différences dans les distributions de contraintes observées pour ces deux simulations. Sur la Figure 5.18, la contrainte de von Mises maximale augmente avec la réduction de l'éprouvette et est plus importante dans le cas de la simulation non isotherme (ETM). Quant à la distribution, les deux simulations montrent des résultats très différents : la localisation de la contrainte maximale est observée au extrémités de l'éprouvette pour la simulation (ETM) alors que pour l'autre simulation elle est plus concentrée au centre de l'éprouvette. En somme on peut dire que des profils de température non uniformes entraînent une déformation non uniforme puisque la ductilité du matériau dépend de la température, c'est-à-dire que les régions à plus basse température se déforment moins. Afin d'accommoder la réduction ou l'allongement appliqué, la région la plus chaude au centre doit se déformer plus qu'elle ne le ferait si la température était uniforme. La localisation de la déformation au centre de l'éprouvette est plus prononcée lorsque la température globale est élevée et que la vitesse de déformation appliquée est faible ; la ductilité est plus élevée dans ces conditions.

5.5 Conclusions du chapitre

Ce chapitre a abouti à la création d'un modèle électro-thermomécanique totalement couplé, conçu dans le but de reproduire les conditions des essais Gleeble. Pour accomplir cette tâche, un algorithme intégré à Abaqus/Standard via la routine utilisateur UAMP a été élaboré. Cet algorithme ajuste la densité du courant à chaque intervalle de temps en fonction des écarts de température entre la consigne et les mesures numériques, garantissant ainsi un chauffage optimal de l'éprouvette.

Afin de confirmer la validité de cette implémentation, une comparaison entre les données numériques et les mesures de température obtenues expérimentalement a été effectuée, révélant une excellente corrélation entre les deux ensembles de données.

L'utilisation de cet algorithme a permis la création de deux types de simulations, à savoir un modèle axisymétrique et un modèle 3D. A travers ces deux modèles, l'impact du gradient de température sur l'état de contrainte et de déformation du matériau 3Cr2Mo a été analysé.

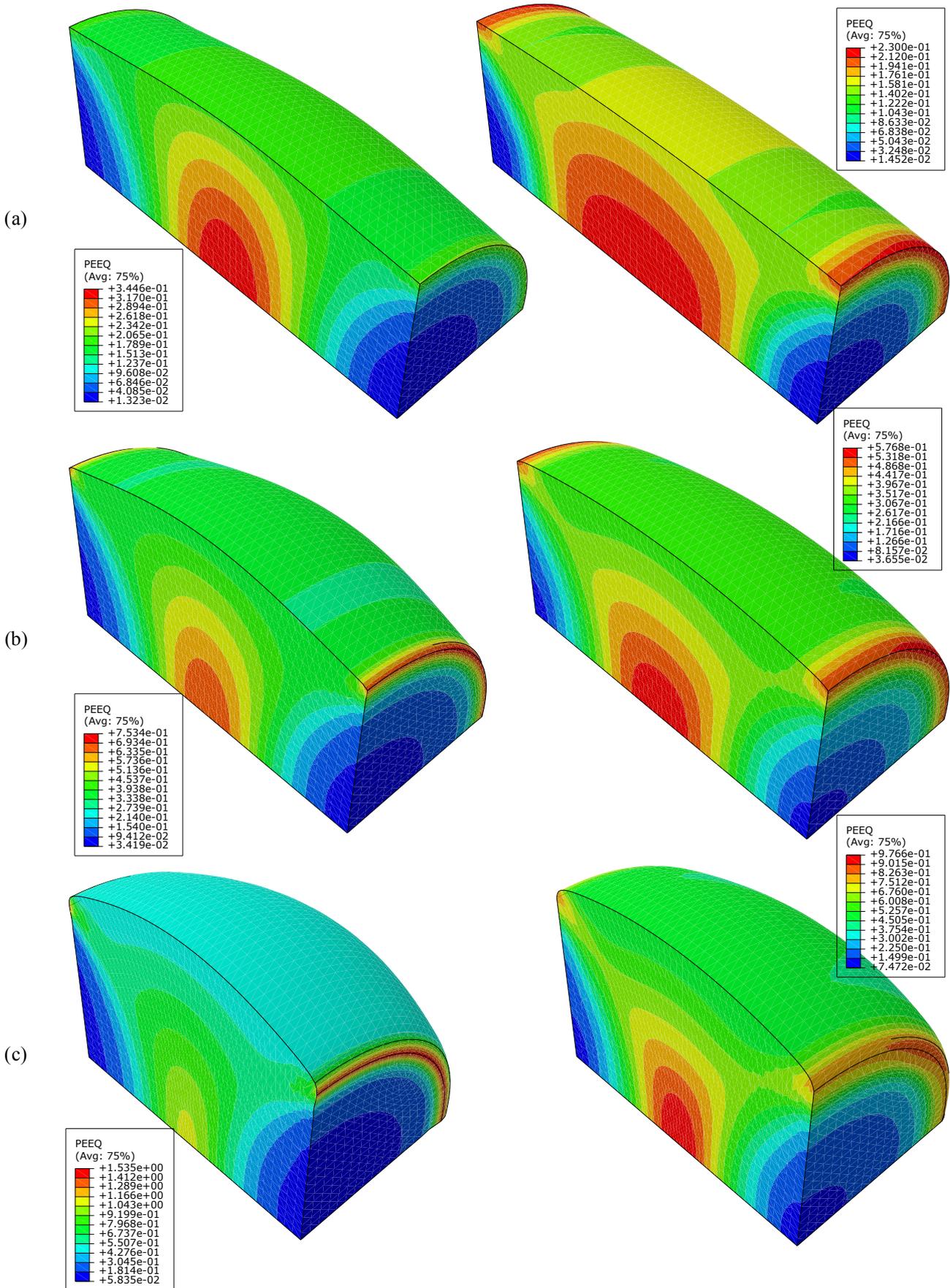


FIGURE 5.17 – Déformation plastique pour une température 1050 °C et une vitesse de déformation $10^{-2} s^{-1}$ lors d'une simulation TM (gauche) et une simulation ETM.

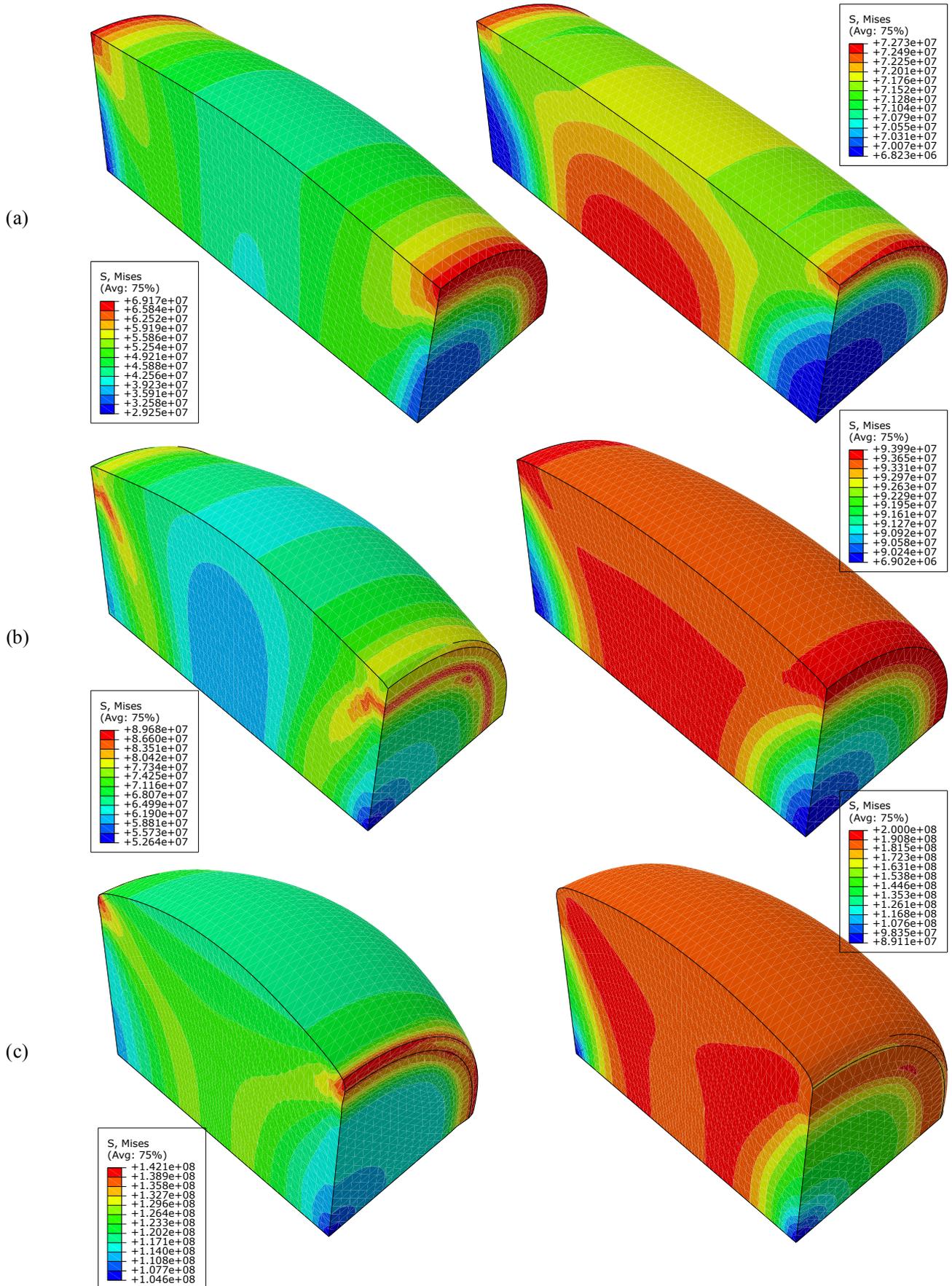


FIGURE 5.18 – Contrainte de von Mises pour une température 1050°C et une vitesse de déformation $10^{-2} s^{-1}$ lors d'une simulation TM (gauche) et une simulation ETM.

Conclusions générales et perspectives

Conclusions générales de cette étude

La modélisation numérique non linéaire des procédés de mise en forme utilisés dans l'industrie est importante pour la fabrication et l'ingénierie. Elle présente des avantages considérables pour la compréhension, l'analyse et l'optimisation d'une variété de procédés de transformation des matériaux, tels que le formage, l'emboutissage, la soudure, l'extrusion, et bien d'autres. Ces simulations numériques dépendent des propriétés matériaux et de la modélisation du comportement thermomécanique. Ces caractérisations s'effectuent à l'échelle du laboratoire avec des dispositifs d'essai, tels que le dispositif thermomécanique Gleeble. L'objectif principal de ces travaux a donc concerné le développement d'un modèle électro-thermomécanique entièrement couplé pour simuler le comportement d'alliages métalliques lors d'essais Gleeble.

Dans le cadre de cette étude, visant à obtenir des résultats expérimentaux conformes aux conditions typiquement rencontrées dans les processus industriels de mise en forme des métaux, une campagne d'essais de compression à haute température, présentée dans le Chapitre 1, a été menée. Ces essais ont été réalisés au sein du groupe CM2P en utilisant le dispositif Gleeble-3800. Les données expérimentales ainsi obtenues ont été organisées et analysées, servant de matériel de base pour l'identification de plusieurs lois de comportement non-linéaires. En extension de ces lois issues de la littérature, une nouvelle loi d'écoulement analytique (modèle PTM), a été proposée dans le Chapitre 2. Cette nouvelle loi se distingue par sa capacité d'adaptation à différents types d'essais. Le choix des lois de comportement à utiliser pour les simulations a été motivé par leur intégration aisée dans des logiciels éléments finis couramment utilisés tels qu'Abaqus, Forge, etc.

L'efficacité d'une simulation numérique étant intrinsèquement liée au modèle utilisé pour décrire le comportement du matériau, les modèles analytiques, bien qu'utilisés traditionnellement, peuvent parfois fournir des résultats insatisfaisants lorsqu'ils sont appliqués à des matériaux présentant des comportements complexes. Dans ce contexte, les modèles RNA se présentent comme une alternative prometteuse, car ils n'ont pas besoin de formes explicites et peuvent s'adapter à une variété de cas d'étude. Le Chapitre 3 est consacré au développement d'un modèle de comportement RNA, et une comparaison détaillée avec les modèles analytiques a révélé que ce modèle RNA offre des résultats plus précis.

Afin de valider les lois de comportement identifiées, des simulations thermomécaniques ont été menées dans le Chapitre 4. L'implémentation numérique de ces lois de comportement dans le logiciel éléments finis Abaqus a été présentée en amont des simulations numériques. Deux types de simulations ont été réalisées pour valider numériquement le modèle. D'une part, des essais d'impact de Taylor et de rétrécissement d'une barre circulaire sur le matériau 42CrMo4 ont été utilisés pour valider l'implémentation numérique du modèle RNA, avec des résultats comparés à ceux du modèle

de Johnson-Cook intégré dans Abaqus. Les résultats ont montré que le modèle RNA reproduit fidèlement le comportement du matériau. D'autre part, un essai de compression d'un cylindre a été réalisé en utilisant à la fois les modèles analytiques et le modèle RNA, avec une fois de plus des résultats en faveur du modèle RNA, confirmant les observations faites lors de l'identification des lois de comportement.

Le Chapitre 5 a été dédié à une simulation électro-thermomécanique entièrement couplée, basée exclusivement sur le modèle RNA en raison des résultats prometteurs obtenus dans le Chapitre 4. L'objectif était de reproduire fidèlement les conditions expérimentales sur le dispositif Gleeble. Pour ce faire, un algorithme de contrôle de température a été développé pour ajuster la densité de courant injectée à chaque incrément de calcul, et il a été implémenté dans Abaqus/Standard via la routine utilisateur UAMP, combinée à la routine UHARD pour décrire le comportement plastique du matériau. Ce chapitre a également permis d'analyser l'influence du gradient de température sur l'état de contrainte et de déformation du matériau. Bien que des résultats intéressants aient été obtenus dans le cadre de cette thèse, il reste des aspects à améliorer et à explorer dans de futures recherches.

Perspectives pour des travaux ultérieurs

Le modèle électro-thermomécanique développé représente une avancée significative, mais des améliorations sont envisageables pour une meilleure précision et une application plus étendue. Il jette les bases pour de futures avancées dans le domaine de la modélisation numérique des procédés de mise en forme des métaux à l'échelle industrielle.

- La vitesse de déformation maximale utilisée lors de nos essais est de 5 s^{-1} , ce qui est nettement en deçà des vitesses de déformation couramment observées lors des essais en dynamique. Cependant, il convient de noter que le dispositif Gleeble-3800 permet de réaliser des essais à des vitesses de déformation plus élevées en utilisant des designs spécifiques d'éprouvettes. Il serait intéressant de reproduire numériquement ces essais en prenant en compte l'aspect électro-thermomécanique couplé.
- Concernant la loi d'écoulement PTM, des efforts considérables ont été déployés pour simplifier sa formulation, afin de permettre une utilisation étendue, en particulier en ce qui concerne les difficultés lors de l'identification de ses paramètres. Dans cette étude, un programme Python a été développé pour l'identification de cette loi, mais la mise en œuvre reste délicate. Par conséquent, un travail plus approfondi reste à entreprendre, non seulement dans le but de simplifier sa formulation, mais aussi de développer une méthode d'identification plus accessible, permettant ainsi une utilisation plus répandue dans la communauté et, par extension, dans la simulation des processus industriels complexes.
- Les modèles analytiques (à l'exception des modèles HS et AR) ainsi que le modèle RNA sont des lois de comportement nouvelles pour le logiciel Forge, qui est largement utilisé de nos jours dans les procédés de mise en forme et de traitement thermique des matériaux. Il serait pertinent d'implémenter ces lois de comportement en utilisant des routines utilisateur dans le logiciel Forge. La plupart des matériaux disponibles dans la bibliothèque de Forge utilisent par défaut le modèle HS, qui ne décrit pas toujours le comportement de manière précise, comme cela a été observé dans cette étude. Afin de valider l'implémentation de ces lois dans le logiciel Forge, une comparaison devra être effectuée avec les résultats obtenus grâce à leur implémentation dans Abaqus. La même démarche pourra être suivie pour l'implémentation de ces lois de comportement dans le logiciel DynELA.

- Dans le cadre de cette thèse, l'évolution de la microstructure du matériau au cours de la simulation a été abordée en se concentrant sur la recristallisation dynamique et la taille des grains. Cependant, les résultats de la simulation se limitent actuellement à la visualisation de la fraction volumique de la recristallisation. Afin de mieux valider expérimentalement ces résultats numériques, il est essentiel d'observer les joints des grains à partir de la simulation. Dans cette optique, des travaux futurs viseront à mettre en place une procédure permettant de reconstruire les joints de grains pendant ou à l'issue d'une simulation numérique, offrant ainsi la possibilité d'observer en temps réel l'évolution de la microstructure du matériau. Le modèle électro-thermomécanique entièrement couplé joue ici un rôle fondamental, car il permettra de suivre de manière exhaustive l'historique du matériau au fil du temps. Une telle simulation devrait indubitablement avoir un impact significatif sur l'amélioration des simulations des procédés industriels, en offrant une vision plus précise et détaillée de l'évolution de la microstructure des matériaux au cours des processus de fabrication.

VUHARD du modèle de Johnson–Cook

Les pages suivantes présentent la liste de la routine VUHARD pour la loi constitutive de Johnson–Cook, qui correspond au modèle VUHARD présenté dans le paragraphe 4.1.1. Dans la routine VUHARD, seule la définition de la limite d'élasticité de Johnson–Cook et de ses dérivées par rapport aux variables appropriées (ε , $\dot{\varepsilon}$ et T) est requise, et le reste du travail est effectué par l'algorithme intégré d'Abaqus.

Ce code source contient la définition de 5 fonctions :

- **yieldStress** : Définit la fonction de calcul de la contrainte d'écoulement de Johnson–Cook σ .
- **yieldHardEpsp** : Définit la fonction de calcul de la dérivée de la contrainte d'écoulement de Johnson–Cook par rapport à la déformation $\frac{\partial \sigma}{\partial \varepsilon}$.
- **yieldHardDepsp** : Définit la fonction de calcul de la dérivée de la contrainte d'écoulement de Johnson–Cook par rapport à la vitesse de déformation $\frac{\partial \sigma}{\partial \dot{\varepsilon}}$.
- **yieldHardTemp** : Définit la fonction de calcul de la dérivée de la contrainte d'écoulement de Johnson–Cook par rapport à la température $\frac{\partial \sigma}{\partial T}$.
- **vhard** : Routine principale appelée par le code de calcul Abaqus.

```

1  C *****
2  C Function to compute the Johnson-Cook yield stress
3  C *****
4      function yieldStress (
5  C Parameters
6      1 epsp, dejsp, temp,
7  C Constants of the constitutive law
8      2 parA, parB, parC, parn, parm, pardejsp0, parT0, parTm )
9      include 'vaba_param.inc'
10 C Hardening part of the Johnson-Cook law
11     hardPart = parA + parB * epsp**parn
12 C Dependence to the deformation rate
13     if (dejsp .gt. pardejsp0) then
14         viscPart = 1.0 + parC * log (dejsp/pardejsp0)
15     else
16         viscPart = 1.0
17     end if
18 C Dependence to the temperature if parT0 < temp < parTm
19     tempPart = 1.0
20     if (temp > parT0) then
21         if (temp < parTm) then
22             tempPart = 1.0 - ((temp - parT0) / (parTm - parT0))**parm
23         else
24             tempPart = 0.0
25         end if
26     end if
27 C Compute and return the yield stress
28     yieldStress = hardPart * viscPart * tempPart
29     return
30     end
31 C
32 C *****
33 C Function to compute the Johnson-Cook hardening / epsp
34 C *****
35     function yieldHardEpsp (
36 C Parameters
37     1 epsp, dejsp, temp,
38 C Constants of the constitutive law
39     2 parA, parB, parC, parn, parm, pardejsp0, parT0, parTm)
40     include 'vaba_param.inc'
41 C Hardening part of the Johnson-Cook law
42     hardPart = parn * parB * (epsp**(parn - 1.0))
43 C Dependence to the deformation rate
44     if (dejsp .gt. pardejsp0) then
45         hardPart = hardPart * (1.0 + parC * log (dejsp/pardejsp0))
46     end if
47 C Dependence to the temperature if parT0 < temp < parTm
48     tempPart = 1.0
49     if (temp > parT0) then
50         if (temp < parTm) then
51             tempPart = 1.0 - ((temp - parT0) / (parTm - parT0))**parm
52     else

```

```

53     tempPart = 0.0
54     end if
55 end if
56 C Compute and return the yield stress
57 yieldHardEpsp = hardPart * tempPart
58 return
59 end
60 C
61 C *****
62 C Function to compute the Johnson-Cook hardening / depsp
63 C *****
64     function yieldHardDepsp (
65 C Parameters
66     1 epsp, depsp, temp,
67 C Constants of the constitutive law
68     2 parA, parB, parC, parn, parm, pardepsp0, parT0, parTm)
69     include 'vaba_param.inc'
70 C Hardening part of the Johnson-Cook law
71     hardPart = 0.0
72 C Dependence to the deformation rate
73     if (depsp .gt. pardepsp0) then
74         hardPart = (parA + parB * epsp**parn) * parC / depsp
75     end if
76 C Dependence to the temperature if parT0 < temp < parTm
77     tempPart = 1.0
78     if (temp > parT0) then
79         if (temp < parTm) then
80             tempPart = 1.0 - ((temp - parT0) / (parTm - parT0))**parm
81         else
82             tempPart = 0.0
83         end if
84     end if
85 C Compute and return the yield stress
86     yieldHardDepsp = hardPart * tempPart
87     return
88 end
89 C
90 C *****
91 C Function to compute the Johnson-Cook hardening / T
92 C *****
93     function yieldHardTemp (
94 C Parameters
95     1 epsp, depsp, temp,
96 C Constants of the constitutive law
97     2 parA, parB, parC, parn, parm, pardepsp0, parT0, parTm)
98     include 'vaba_param.inc'
99 C Hardening part of the Johnson-Cook law
100     hardPart = parA + parB * epsp**parn
101 C Dependence to the deformation rate
102     if (depsp .gt. pardepsp0) then
103         viscPart = 1.0 + parC * log (depsp/pardepsp0)
104     else

```

```

105     viscPart = 1.0
106     end if
107 C Dependence to the temperature if parT0 < temp < parTm
108     tempPart = 0.0
109     if (temp > parT0 .and. temp < parTm) then
110         tempPart = -parm*((temp - parT0)/(parTm - parT0)**(parm))
111     1 / (temp - parT0)
112     end if
113 C Compute and return the yield stress
114     yieldHardTemp = hardPart * viscPart * tempPart
115     return
116     end
117 C
118 C *****
119 C J2 Mises Plasticity with isotropic Johnson-Cook hardening
120 C *****
121     subroutine vuhard (
122 C Read only -
123         1     nblock,
124         2     nElement, nIntPt, nLayer, nSecPt,
125         3     lAnneal, stepTime, totalTime, dt, cmname,
126         4     nstatev, nfieldv, nprops,
127         5     props, tempOld, tempNew, fieldOld, fieldNew,
128         6     stateOld,
129         7     eqps, eqpsRate,
130 C Write only -
131         8     yield, dyieldDtemp, dyieldDeqps,
132         9     stateNew )
133 C
134     include 'vaba_param.inc'
135 C
136     dimension nElement(nblock),
137         1     props(nprops),
138         2     tempOld(nblock),
139         3     fieldOld(nblock,nfieldv),
140         4     stateOld(nblock,nstatev),
141         5     tempNew(nblock),
142         6     fieldNew(nblock,nfieldv),
143         7     eqps(nblock),
144         8     eqpsRate(nblock),
145         9     yield(nblock),
146         1     dyieldDtemp(nblock),
147         2     dyieldDeqps(nblock,2),
148         3     stateNew(nblock,nstatev)
149 C
150     character*80 cmname
151 C
152 C *****
153 C Start of the subroutine grab the parameters of the constitutive law
154 C *****
155     parA     = props(1)
156     parB     = props(2)

```

```

157     parn      = props(3)
158     parC      = props(4)
159     parm      = props(5)
160     pardepsp0 = props(6)
161     parT0     = props(7)
162     parTm     = props(8)
163 C
164 C *****
165 C Main computation block
166 C *****
167     do k = 1, nblock
168         epsp = eqps(k)
169         depsp = eqpsRate(k)
170         temp = tempNew(k)
171 C Compute the yield stress
172         Yield(k) = yieldStress(epsp, depsp, temp,
173             1 parA, parB, parC, parn, parm, pardepsp0, parT0, parTm)
174 C Compute derivative of yield / epsp
175         dyieldDeqps(k,1) = yieldHardEpsp(epsp, depsp, temp,
176             1 parA, parB, parC, parn, parm, pardepsp0, parT0, parTm)
177 C Compute derivative of yield / depsp
178         dyieldDeqps(k,2) = yieldHardDepsp(epsp, depsp, temp,
179             1 parA, parB, parC, parn, parm, pardepsp0, parT0, parTm)
180 C Compute derivative of yield / temp
181         dyieldDtemp(k) = yieldHardTemp(epsp, depsp, temp,
182             1 parA, parB, parC, parn, parm, pardepsp0, parT0, parTm)
183     end do
184 C
185     return
186 end
187

```


VUHARD du modèle PTM

Les pages suivantes présentent la liste de la routine VUHARD pour la loi constitutive proposée (PTM) dans cette étude, qui correspond au modèle VUHARD présenté dans la Section 4.1.3. Dans la routine VUHARD, seule la définition de la limite d'élasticité de de PTM et de ses dérivées par rapport aux variables appropriées (ε , $\dot{\varepsilon}$ et T) est requise, et le reste du travail est effectué par l'algorithme intégré d'Abaqus.

Comme pour l'implémentation du modèle de Johnson–Cook, le code source contient la définition de 5 fonctions :

- **yieldStress** : Définit la fonction de calcul de la contrainte d'écoulement de PTM σ .
- **yieldHardEpsp** : Définit la fonction de calcul de la dérivée de la contrainte d'écoulement de PTM par rapport à la déformation $\frac{\partial \sigma}{\partial \varepsilon}$.
- **yieldHardDepsp** : Définit la fonction de calcul de la dérivée de la contrainte d'écoulement de Johnson–Cook par rapport à la vitesse de déformation $\frac{\partial \sigma}{\partial \dot{\varepsilon}}$.
- **yieldHardTemp** : Définit la fonction de calcul de la dérivée de la contrainte d'écoulement de PTM par rapport à la température $\frac{\partial \sigma}{\partial T}$.
- **vuhard** : Routine principale appelée par le code de calcul Abaqus.

```

1  C *****
2  C Function to compute the Pierre-TIZE MHA yield stress
3  C *****
4      function yieldStress (
5          1 epsp, dejsp, temp,
6          2 parA0, parA1, parA2, parA3, parA4,
7          3 parB0, parB1, parB2, parB3, parB4,
8          4 parC00, parC01, parC02, parC03, parC04,
9          5 parC10, parC11, parC12, parC13, parC14,
10         6 parC20, parC21, parC22, parC23, parC24,
11         7 parC30, parC31, parC32, parC33, parC34,
12         8 parC40, parC41, parC42, parC43, parC44, pardejsp0, parT0)
13     include 'vaba_param.inc'
14     C Hardening part of the Pierre TIZE MHA law
15         A0 = parA0 + parA1*epsp + parA2*epsp**2 + parA3*epsp**3 +
16           + parA4*epsp**4
17     C Dependence of temperature (if temp /= parT0)
18         if (temp .gt. parT0) then
19             BT = (parB0 + parB1*epsp + parB2*epsp**2 + parB3*epsp**3 +
20               + parB4*epsp**4)*(temp-parT0)
21             C1 = (parC10 + parC11*epsp + parC12*epsp**2 + parC13*epsp**3 +
22               + parC14*epsp**4)*(temp-parT0)
23             C2 = (parC20 + parC21*epsp + parC22*epsp**2 + parC23*epsp**3 +
24               + parC24*epsp**4)*(temp-parT0)**2
25             C3 = (parC30 + parC31*epsp + parC32*epsp**2 + parC33*epsp**3 +
26               + parC34*epsp**4)*(temp-parT0)**3
27             C4 = (parC40 + parC41*epsp + parC42*epsp**2 + parC43*epsp**3 +
28               + parC44*epsp**4)*(temp-parT0)**4
29             CT = C1 + C2 + C3 + C4
30         else
31             BT = 0.0
32             CT = 0.0
33         end if
34     C Dependence to the deformation rate (dejsp /= pardejsp0)
35         if (dejsp .gt. pardejsp0) then
36             C0 = parC00 + parC01*epsp + parC02*epsp**2 + parC03*epsp**3 +
37               + parC04*epsp**4
38             visc = exp(BT + (C0 + CT)*log(dejsp/pardejsp0))
39         else
40             visc = exp(BT)
41         end if
42     C Compute and return the yield stress
43         yieldStress = A0 * visc
44     return
45     end
46     C
47     C *****
48     C Function to compute the Pierre TIZE MHA hardening / epsp
49     C *****
50     function yieldHardEpsp (
51         1 epsp, dejsp, temp,
52         2 parA0, parA1, parA2, parA3, parA4,

```

```

53     3 parB0, parB1, parB2, parB3, parB4,
54     4 parC00, parC01, parC02, parC03, parC04,
55     5 parC10, parC11, parC12, parC13, parC14,
56     6 parC20, parC21, parC22, parC23, parC24,
57     7 parC30, parC31, parC32, parC33, parC34,
58     8 parC40, parC41, parC42, parC43, parC44, pardepsp0, parT0)
59     include 'vaba_param.inc'
60 C Hardening part of the Pierre TIZE MHA law
61     A0 = parA0 + parA1*epsp + parA2*epsp**2 + parA3*epsp**3 +
62     + parA4*epsp**4
63     AD0 = parA1 + 2*parA2*epsp + 3*parA3*epsp**2 + 4*parA4*epsp**3
64     C0 = parC00 + epsp*parC01 + parC02*epsp**2 + parC03*epsp**3 +
65     + parC04*epsp**4
66     CD0 = parC01 + 2*parC02*epsp + 3*parC03*epsp**2 +
67     + 4*parC04*epsp**3
68 C Dependence of temperature (if temp /= parT0)
69     if (temp .gt. parT0) then
70         B0 = (parB0 + epsp*(parB1 + epsp*(parB2 + epsp*(parB3 +
71     +     epsp*parB4))))*(temp-parT0)
72         BD0 = (parB1 + epsp*(2*parB2 + epsp*(3*parB3 +
73     +     4*epsp*parB4)))*(temp-parT0)
74         C1 = (parC10 + epsp*(parC11 + epsp*(parC12 + epsp*(parC13 +
75     +     epsp*parC14))))*(temp-parT0)
76         C2 = (parC20 + epsp*(parC21 + epsp*(parC22 + epsp*(parC23 +
77     +     epsp*parC24))))*(temp-parT0)**2
78         C3 = (parC30 + epsp*(parC31 + epsp*(parC32 + epsp*(parC33 +
79     +     epsp*parC34))))*(temp-parT0)**3
80         C4 = (parC40 + epsp*(parC41 + epsp*(parC42 + epsp*(parC43 +
81     +     epsp*parC44))))*(temp-parT0)**4
82         CD1 = (parC11 + epsp*(2*parC12 + epsp*(3*parC13 +
83     +     4*epsp*parC14)))*(temp-parT0)
84         CD2 = (parC21 + epsp*(2*parC22 + epsp*(3*parC23 +
85     +     4*epsp*parC24)))*(temp-parT0)**2
86         CD3 = (parC31 + epsp*(2*parC32 + epsp*(3*parC33 +
87     +     4*epsp*parC34)))*(temp-parT0)**3
88         CD4 = (parC41 + epsp*(2*parC42 + epsp*(3*parC43 +
89     +     4*epsp*parC44)))*(temp-parT0)**4
90         CT = C1 + C2 + C3 + C4
91         CDT = CD1 + CD2 + CD3 + CD4
92     else
93         CT = 0.0
94         CDT = 0.0
95         B0 = 0.0
96         BD0 = 0.0
97     end if
98 C Dependence to the deformation rate (depsp /= pardepsp0)
99     if (depsp .gt. pardepsp0) then
100         vlg = log(depsp/pardepsp0)
101     else
102         vlg = 0.0
103     end if
104 C Compute and return the yield stress

```

```

105     yieldHardEpsp = (AD0 + A0*(BD0 + (CD0 + CDT)*vlg))*
106 + exp(B0 + (C0 + CT)*vlg)
107     return
108     end
109 C
110 C *****
111 C Function to compute the Pierre TIZE MHA hardening / depsp
112 C *****
113     function yieldHardDepsp (
114     1 epsp, depsp, temp,
115     2 parA0, parA1, parA2, parA3, parA4,
116     3 parB0, parB1, parB2, parB3, parB4,
117     4 parC00, parC01, parC02, parC03, parC04,
118     5 parC10, parC11, parC12, parC13, parC14,
119     6 parC20, parC21, parC22, parC23, parC24,
120     7 parC30, parC31, parC32, parC33, parC34,
121     8 parC40, parC41, parC42, parC43, parC44, pardepsp0, parT0)
122     include 'vaba_param.inc'
123 C Hardening part of the Pierre-TIZE MHA law
124     A0 = parA0 + parA1*epsp + parA2*epsp**2 + parA3*epsp**3 +
125     + parA4*epsp**4
126     C0 = parC00 + parC01*epsp + parC02*epsp**2 + parC03*epsp**3 +
127     + parC04*epsp**4
128 C Dependence to the temperature (if parT0 /= temp)
129     if (temp .gt. parT0) then
130         B0 = (parB0 + parB1*epsp + parB2*epsp**2 + parB3*epsp**3 +
131         + parB4*epsp**4)*(temp - parT0)
132         C1 = (parC10 + parC11*epsp + parC12*epsp**2 + parC13*epsp**3 +
133         + parC14*epsp**4)*(temp - parT0)
134         C2 = (parC20 + parC21*epsp + parC22*epsp**2 + parC23*epsp**3 +
135         + parC24*epsp**4)*(temp - parT0)**2
136         C3 = (parC30 + parC31*epsp + parC32*epsp**2 + parC33*epsp**3 +
137         + parC34*epsp**4)*(temp - parT0)**3
138         C4 = (parC40 + parC41*epsp + parC42*epsp**2 + parC43*epsp**3 +
139         + parC44*epsp**4)*(temp - parT0)**3
140         CT = C1 + C2 + C3 + C4
141     else
142         CT = 0.0
143         B0 = 0.0
144     end if
145 C Dependence to the deformation rate (depsp /= pardepsp0)
146     if (depsp .gt. pardepsp0) then
147         yieldHardDepsp = (A0*(C0 + CT)*exp(B0 + (C0 + CT)*
148         + log(depsp/pardepsp0)))/depsp
149     else
150         yieldHardDepsp = 0.0
151     end if
152 C Compute and return the yield stress
153     return
154     end
155 C
156 C *****

```

```

157 C Function to compute of the Pierre-TIZE MHA law / T
158 C ****
159     function yieldHardTemp (
160         1 epsp, dejsp, temp,
161         2 parA0, parA1, parA2, parA3, parA4,
162         3 parB0, parB1, parB2, parB3, parB4,
163         4 parC00, parC01, parC02, parC03, parC04,
164         5 parC10, parC11, parC12, parC13, parC14,
165         6 parC20, parC21, parC22, parC23, parC24,
166         7 parC30, parC31, parC32, parC33, parC34,
167         8 parC40, parC41, parC42, parC43, parC44, pardejsp0, parT0)
168     include 'vaba_param.inc'
169 C Dependence to the temperature if parT0 /= temp
170     A0 = parA0 + epsp*(parA1 + epsp*(parA2 + epsp*(parA3 +
171     + epsp*parA4)))
172     BST0 = parB0 + epsp*(parB1 + epsp*(parB2 + epsp*(parB3 +
173     + epsp*parB4)))
174     C0 = parC00 + epsp*parC01 + epsp**2*parC02 + epsp**3*parC03 +
175     + epsp**4*parC04
176     CST1 = parC10 + epsp*(parC11 + epsp*(parC12 + epsp*(parC13 +
177     + epsp*parC14)))
178     if (temp .gt. parT0) then
179         B0 = BST0*(temp - parT0)
180         CST2 = parC20 + epsp*(parC21 + epsp*(parC22 + epsp*(parC23 +
181     + epsp*parC24)))
182         CST3 = parC30 + epsp*(parC31 + epsp*(parC32 + epsp*(parC33 +
183     + epsp*parC34)))
184         CST4 = parC40 + epsp*(parC41 + epsp*(parC42 + epsp*(parC43 +
185     + epsp*parC44)))
186         C1 = CST1*(temp - parT0)
187         C2 = CST2*(temp - parT0)**2
188         C3 = CST3*(temp - parT0)**3
189         C4 = CST4*(temp - parT0)**4
190         CT2 = 2*CST2*(temp - parT0)
191         CT3 = 3*CST3*(temp - parT0)**2
192         CT4 = 4*CST4*(temp - parT0)**3
193         CT = C1 + C2 + C3 + C4
194         CTT = CT2 + CT3 + CT4
195     else
196         B0 = 0.0
197         CT = 0.0
198         CTT = 0.0
199     end if
200 C Dependence to the deformation rate
201     if (dejsp .gt. pardejsp0) then
202         vlg = log(dejsp/pardejsp0)
203     else
204         vlg = 0.0
205     end if
206 C Compute and return the yield stress
207     yieldHardTemp = exp(B0 + (C0 + CT)*vlg)*A0*(BST0 +
208     + (CST1 + CTT)*vlg)

```

```

209     return
210     end
211 C
212 C *****
213 C J2 Mises Plasticity with isotropic Pierre-TIZE MHA hardening
214 C *****
215     subroutine vuhard (
216 C Read only -
217         1     nblock,
218         2     nElement, nIntPt, nLayer, nSecPt,
219         3     lAnneal, stepTime, totalTime, dt, cmname,
220         4     nstatev, nfieldv, nprops,
221         5     props, tempOld, tempNew, fieldOld, fieldNew,
222         6     stateOld,
223         7     eqps, eqpsRate,
224 C Write only -
225         8     yield, dyieldDtemp, dyieldDeqps,
226         9     stateNew )
227 C
228     include 'vaba_param.inc'
229 C
230     dimension nElement(nblock),
231         1     props(nprops),
232         2     tempOld(nblock),
233         3     fieldOld(nblock,nfieldv),
234         4     stateOld(nblock,nstatev),
235         5     tempNew(nblock),
236         6     fieldNew(nblock,nfieldv),
237         7     eqps(nblock),
238         8     eqpsRate(nblock),
239         9     yield(nblock),
240         1     dyieldDtemp(nblock),
241         2     dyieldDeqps(nblock,2),
242         3     stateNew(nblock,nstatev)
243 C
244     character*80 cmname
245 C
246 C *****
247 C Start of the subroutine grab the parameters of the constitutive law
248 C *****
249     parA0     = props(1)
250     parA1     = props(2)
251     parA2     = props(3)
252     parA3     = props(4)
253     parA4     = props(5)
254     parB0     = props(6)
255     parB1     = props(7)
256     parB2     = props(8)
257     parB3     = props(9)
258     parB4     = props(10)
259     parC00    = props(11)
260     parC01    = props(12)

```

```

261     parC02     = props(13)
262     parC03     = props(14)
263     parC04     = props(15)
264     parC10     = props(16)
265     parC11     = props(17)
266     parC12     = props(18)
267     parC13     = props(19)
268     parC14     = props(20)
269     parC20     = props(21)
270     parC21     = props(22)
271     parC22     = props(23)
272     parC23     = props(24)
273     parC24     = props(25)
274     parC30     = props(26)
275     parC31     = props(27)
276     parC32     = props(28)
277     parC33     = props(29)
278     parC34     = props(30)
279     parC40     = props(31)
280     parC41     = props(32)
281     parC42     = props(33)
282     parC43     = props(34)
283     parC44     = props(35)
284     pardepsp0  = props(36)
285     parT0      = props(37)
286 C
287 C *****
288 C Main computation block
289 C *****
290     do k = 1, nblock
291         epsp = eqps(k)
292         depsp = eqpsRate(k)
293         temp = tempNew(k)
294 C Compute the yield stress
295         Yield(k) = yieldStress(epsp, depsp, temp,
296     1 parA0, parA1, parA2, parA3, parA4,
297     2 parB0, parB1, parB2, parB3, parB4,
298     3 parC00, parC01, parC02, parC03, parC04,
299     4 parC10, parC11, parC12, parC13, parC14,
300     5 parC20, parC21, parC22, parC23, parC24,
301     6 parC30, parC31, parC32, parC33, parC34,
302     7 parC40, parC41, parC42, parC43, parC44, pardepsp0, parT0)
303 C Compute derivative of yield / epsp
304         dyieldDeqps(k,1) = yieldHardEpsp(epsp, depsp, temp,
305     1 parA0, parA1, parA2, parA3, parA4,
306     2 parB0, parB1, parB2, parB3, parB4,
307     3 parC00, parC01, parC02, parC03, parC04,
308     4 parC10, parC11, parC12, parC13, parC14,
309     5 parC20, parC21, parC22, parC23, parC24,
310     6 parC30, parC31, parC32, parC33, parC34,
311     7 parC40, parC41, parC42, parC43, parC44, pardepsp0, parT0)
312 C Compute derivative of yield / depsp

```

```
313     dyieldDeqps(k,2) = yieldHardDepsp(eps, de, temp,
314     1 parA0, parA1, parA2, parA3, parA4,
315     2 parB0, parB1, parB2, parB3, parB4,
316     3 parC00, parC01, parC02, parC03, parC04,
317     4 parC10, parC11, parC12, parC13, parC14,
318     5 parC20, parC21, parC22, parC23, parC24,
319     6 parC30, parC31, parC32, parC33, parC34,
320     7 parC40, parC41, parC42, parC43, parC44, pardeps0, parT0)
321 C Compute derivative of yield / temp
322     dyieldDtemp(k) = yieldHardTemp(eps, de, temp,
323     1 parA0, parA1, parA2, parA3, parA4,
324     2 parB0, parB1, parB2, parB3, parB4,
325     3 parC00, parC01, parC02, parC03, parC04,
326     4 parC10, parC11, parC12, parC13, parC14,
327     5 parC20, parC21, parC22, parC23, parC24,
328     6 parC30, parC31, parC32, parC33, parC34,
329     7 parC40, parC41, parC42, parC43, parC44, pardeps0, parT0)
330     end do
331 C
332     return
333 end
334
```

Annexe

VUHARD du modèle RNA

Les pages suivantes présentent la liste de la routine VUHARD pour la loi constitutive RNA, qui correspond au modèle VUHARD présenté dans la Section 4.1.4. Dans la routine VUHARD, seule la définition de la limite d'élasticité de de RNA et de ses dérivées par rapport aux variables appropriées (ε , $\dot{\varepsilon}$ et T) est requise, et le reste du travail est effectué par l'algorithme intégré d'Abaqus.

Le code source contient la définition d'une seule fonction codée en *dur* et issue de l'entraînement du réseau :

- **vuhard** : Routine principale appelée par le code de calcul Abaqus.

```

1  C *****
2  C Function to compute the ANN : 3-15-7-1-sigmoid yield stress
3  C *****
4      subroutine vuhard (
5  C Read only -
6      + nblock, nElement, nIntPt, nLayer, nSecPt, lAnneal, stepTime,
7      + totalTime, dt, cmname, nstatev, nfieldv, nprops, props,
8      + tempOld, tempNew, fieldOld, fieldNew, stateOld, eqps, eqpsRate,
9  C Write only -
10     + yield, dyieldDtemp, dyieldDeqps, stateNew)
11  C
12     include 'vaba_param.inc'
13  C
14     dimension nElement(nblock), props(nprops), tempOld(nblock),
15     + fieldOld(nblock,nfieldv), stateOld(nblock,nstatev),
16     + tempNew(nblock), fieldNew(nblock,nfieldv), eqps(nblock),
17     + eqpsRate(nblock), yield(nblock), dyieldDtemp(nblock),
18     + dyieldDeqps(nblock,2), stateNew(nblock,nstatev)
19  C
20     character*80 cmname
21  C
22     do k = 1, nblock
23  C xemsp = (eqps(k) - minEntries[0]) / rangeEntries[0]
24     xemsp = (eqps(k) - 0.001)/0.699
25  C xdepsp = (log(depsp/logBase) - minEntries[1]) / rangeEntries[1]
26     if (eqpsRate(k) > 0.001) then
27     xdepsp = (log(eqpsRate(k)/0.001) - 0)/8.517193191416
28     else
29     xdepsp = 0
30     eqpsRate(k) = 0.001
31     endif
32  C xtemp = (temp - minEntries[2]) / rangeEntries[2]
33     xtemp = (tempNew(k) - 1050)/200
34  C za = np.exp(-wl.dot(x) - b1)
35     za0 = exp(0.017593745609*xemsp
36     + -12.47632515552*xdepsp
37     + +1.976923445843*xtemp
38     + +8.929817455731)
39     za1 = exp(3.703993330144*xemsp
40     .
41     .
42     .
43     za13 = exp(-0.255482505733*xemsp
44     + -2.586577523061*xdepsp
45     + +0.621491434974*xtemp
46     + +1.664218765708)
47     za14 = exp(0.168478978755*xemsp
48     + +2.26546650014*xdepsp
49     + +1.169081090319*xtemp
50     + -1.508758660818)
51  C zb = 1 + za
52     zb0 = 1 + za0

```

```
53      zb1 = 1 + za1
54      .
55      .
56      .
57      zb13 = 1 + za13
58      zb14 = 1 + za14
59      C zc = w2.dot(zb) + b2
60      zc0 = exp(4.395530749201/zb0
61      + +2.651343098025/zb1
62      + +4.294181435508/zb2
63      + +4.219364224678/zb3
64      + +0.660824248661/zb4
65      + +0.067452094281/zb5
66      + -2.584603869955/zb6
67      + +0.401400078155/zb7
68      + -20.550662146731/zb8
69      + +5.233257349407/zb9
70      + +6.36206014638/zb10
71      + +1.142217971388/zb11
72      + +1.26752366997/zb12
73      + +1.725767397225/zb13
74      + +0.609709858314/zb14
75      + +0.07435847579)
76      zc1 = exp(-1.166225479459/zb0
77      + -0.115470060928/zb1
78      + -2.773166081267/zb2
79      + +1.016220055512/zb3
80      + -0.368965559808/zb4
81      + -0.461073898874/zb5
82      + -2.989589459746/zb6
83      + +0.25923033473/zb7
84      + +0.746766359993/zb8
85      + -6.023251349298/zb9
86      + -1.901345645967/zb10
87      + -0.094933840408/zb11
88      + +1.19082777755/zb12
89      + -3.752345187415/zb13
90      + +1.669116385106/zb14
91      + -0.168735718483)
92      .
93      .
94      .
95      zc5 = exp(0.883809665085/zb0
96      + +1.326880877992/zb1
97      + -0.682548737798/zb2
98      + -2.298691844963/zb3
99      + -0.282283879425/zb4
100     + +0.797057888928/zb5
101     + -1.193558917743/zb6
102     + -0.061847066663/zb7
103     + +0.911573723435/zb8
104     + +1.412984242229/zb9
```

```

105     + +0.295089528315/zb10
106     + +2.213454154318/zb11
107     + +1.877832087933/zb12
108     + -0.984164948601/zb13
109     + +1.215987652376/zb14
110     + -0.14546223185)
111     zc6 = exp(-0.72674376784/zb0
112     + +0.698537198529/zb1
113     + +1.663232588362/zb2
114     + -1.245198313118/zb3
115     + -0.050642600791/zb4
116     + +3.2790902686/zb5
117     + -1.704033510565/zb6
118     + +0.45208208107/zb7
119     + -9.407508462055/zb8
120     + +1.839656209146/zb9
121     + +1.811623489208/zb10
122     + +1.732244006427/zb11
123     + +1.021344510465/zb12
124     + +2.93598511749/zb13
125     + +1.161444745381/zb14
126     + +1.061070629551)
127 C y = w3.dot(xd) + b3
128     y = -2.59086663338/(1 + zc0)
129     + +0.21978674728/(1 + zc1)
130     + +2.185398232118/(1 + zc2)
131     + -0.34819346122/(1 + zc3)
132     + +0.494459525948/(1 + zc4)
133     + +0.606731193951/(1 + zc5)
134     + -0.979264255755/(1 + zc6)
135     + +0.092274325838
136 C zd = w3v*(xd / (1 + zc)**2)
137     zd0 = -2.59086663338*(zc0/(1 + zc0)**2)
138     zd1 = 0.21978674728*(zc1/(1 + zc1)**2)
139     .
140     .
141     .
142     zd5 = 0.606731193951*(zc5/(1 + zc5)**2)
143     zd6 = -0.979264255755*(zc6/(1 + zc6)**2)
144 C ze = za / (1 + za)**2
145     ze0 = za0 / zb0**2
146     ze1 = za1 / zb1**2
147     .
148     .
149     .
150     ze13 = za13 / zb13**2
151     ze14 = za14 / zb14**2
152 C zf = (w2.T).dot(zd) * xd
153     zf0 = (-4.395530749201*zd0
154     + +1.166225479459*zd1
155     + +0.135170085433*zd2
156     + +0.614647367372*zd3

```

```

157 + +5.568292917764*zd4
158 + -0.883809665085*zd5
159 + +0.72674376784*zd6)*ze0
160   zf1 = (-2.651343098025*zd0
161 + +0.115470060928*zd1
162 + -4.566214474898*zd2
163 + +1.034030520196*zd3
164 + -1.902690131769*zd4
165 + -1.326880877992*zd5
166 + -0.698537198529*zd6)*ze1
167   .
168   .
169   .
170   zf13 = (-1.725767397225*zd0
171 + +3.752345187415*zd1
172 + +1.139370696402*zd2
173 + -1.964287777337*zd3
174 + -0.606532754218*zd4
175 + +0.984164948601*zd5
176 + -2.93598511749*zd6)*ze13
177   zf14 = (-0.609709858314*zd0
178 + -1.669116385106*zd1
179 + -2.300090860551*zd2
180 + -0.40477489877*zd3
181 + +0.940599320284*zd4
182 + -1.215987652376*zd5
183 + -1.161444745381*zd6)*ze14
184   C yd = (w1.T).dot(zf)
185   yd0 = -0.017593745609*zf0
186 + -3.703993330144*zf1
187 + -1.673885110027*zf2
188 + +1.098836613684*zf3
189 + +1.56949191701*zf4
190 + -2.252378370366*zf5
191 + -5.536205276754*zf6
192 + +1.325923310019*zf7
193 + -41.673161517862*zf8
194 + -8.687111316495*zf9
195 + -6.515657646893*zf10
196 + +3.591454125923*zf11
197 + -1.541685911176*zf12
198 + +0.255482505733*zf13
199 + -0.168478978755*zf14
200   yd1 = 12.47632515552*zf0
201 + +1.308657836701*zf1
202 + +2.737550434891*zf2
203 + +0.123153826335*zf3
204 + +11.305680103867*zf4
205 + +3.278934793317*zf5
206 + +12.006752114781*zf6
207 + -5.479824326552*zf7
208 + +0.389947666724*zf8

```

```
209 + -0.916012257649*zf9
210 + +1.763536635357*zf10
211 + -7.553673938134*zf11
212 + -2.755642041696*zf12
213 + +2.586577523061*zf13
214 + -2.26546650014*zf14
215   yd2 =-1.976923445843*zf0
216 + +1.206425173571*zf1
217 + -2.209331518582*zf2
218 + -3.156710469061*zf3
219 + -6.290945317044*zf4
220 + -4.029143345076*zf5
221 + -1.787529411656*zf6
222 + +12.196922026907*zf7
223 + -1.196618881322*zf8
224 + -0.163410260387*zf9
225 + -1.625918164675*zf10
226 + +0.077456644425*zf11
227 + +1.895383414634*zf12
228 + -0.621491434974*zf13
229 + -1.169081090319*zf14
230   Yield(k) = 152428511*y + 1310989
231   dyieldDeqps(k,1) = 218066539.341917*yd0
232   dyieldDeqps(k,2) = 17896566.107438*yd1 / eqpsRate(k)
233   dyieldDtemp(k) = 762142.555*yd2
234   end do
235 C
236   return
237   end
238
```

Annexe D

UAMP pour le contrôle du courant

Les pages suivantes présentent un exemple détaillé de la routine UAMP élaborée pour réguler la température pendant le processus de chauffage, tel que décrit dans la section 5.2.1 page 128. En effet, le contrôle de la température sur le dispositif Gleeble repose sur la différence de température mesurée par le thermocouple situé au centre de l'éprouvette. Cette différence de température est utilisée pour ajuster la densité de courant qui doit être injectée à chaque étape du calcul, conformément à l'équation (5.4) page 128. La valeur de cette densité de courant dépend de la forme de l'éprouvette. Dans le cas où cette section est modélisée par un cône de révolution (Figure 5.4 (b) page 129 par exemple), le calcul est décrit en détail comme suit :

Calcul de la surface latérale d'un cône de révolution

Considérons la Figure D.0.1 où l'élément de surface peut s'écrire comme suit dans un système de coordonnées cylindrique :

$$dS = \left\| \frac{\partial \overrightarrow{OM}}{\partial l} \wedge \frac{\partial \overrightarrow{OM}}{\partial \theta} \right\| dl d\theta \quad (D.1)$$

$$\overrightarrow{OM} = x\vec{e}_x + y\vec{e}_y + z\vec{e}_z \quad (D.2)$$

$$x = r \cos \theta \quad ; \quad y = r \sin \theta \quad ; \quad z = z \quad (D.3)$$

$$r = l \sin \alpha; \quad x = l \sin \alpha \cos \theta; \quad y = l \sin \alpha \sin \theta \quad ; \quad z = l \cos \alpha \quad (D.4)$$

$$\overrightarrow{OM} = \begin{pmatrix} r \cos \theta \\ r \sin \theta \\ z \end{pmatrix} \iff \overrightarrow{OM} = \begin{pmatrix} l \sin \alpha \cos \theta \\ l \sin \alpha \sin \theta \\ l \cos \alpha \end{pmatrix} \quad (D.5)$$

$$\frac{\partial \overrightarrow{OM}}{\partial l} = \begin{pmatrix} \sin \alpha \cos \theta \\ \sin \alpha \sin \theta \\ \cos \alpha \end{pmatrix} \quad \text{et} \quad \frac{\partial \overrightarrow{OM}}{\partial \theta} = \begin{pmatrix} -r \sin \theta \\ r \cos \theta \\ 0 \end{pmatrix} \quad (D.6)$$

$$\frac{\partial \overrightarrow{OM}}{\partial l} \wedge \frac{\partial \overrightarrow{OM}}{\partial \theta} = \begin{pmatrix} \sin \alpha \cos \theta \\ \sin \alpha \sin \theta \\ \cos \alpha \end{pmatrix} \wedge \begin{pmatrix} -r \sin \theta \\ r \cos \theta \\ 0 \end{pmatrix} = \begin{pmatrix} -r \cos \theta \cos \alpha \\ -r \sin \theta \cos \alpha \\ r \sin \alpha \end{pmatrix} \quad (D.7)$$

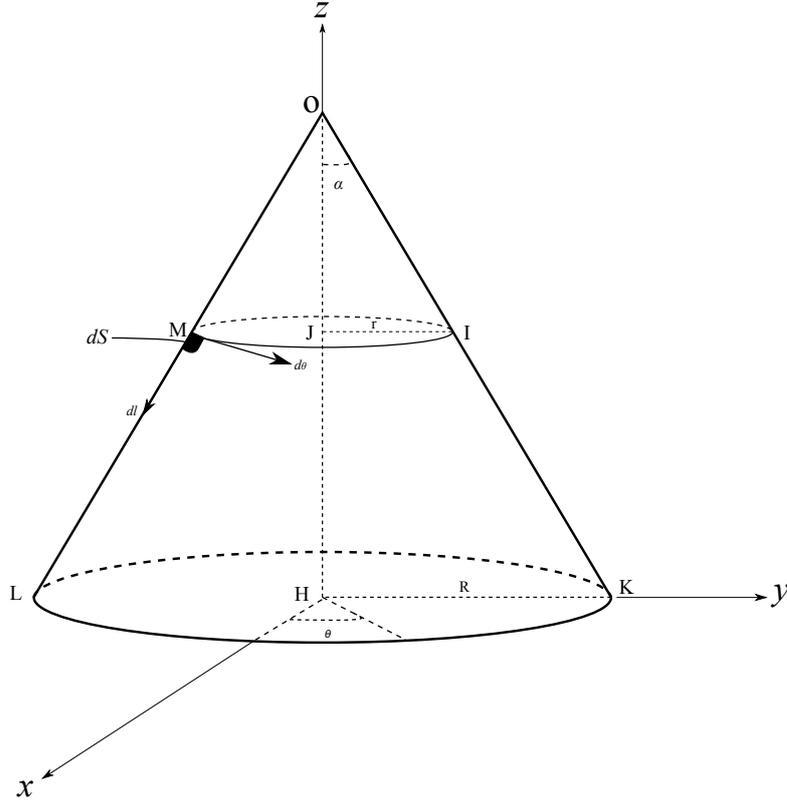


FIGURE D.0.1 – Cône de révolution

$$\left\| \frac{\partial \overrightarrow{OM}}{\partial l} \wedge \frac{\partial \overrightarrow{OM}}{\partial \theta} \right\| = \sqrt{(-r \cos \theta \cos \alpha)^2 + (-r \sin \theta \cos \alpha)^2 + (r \sin \alpha)^2} = r \quad (D.8)$$

$$dS = r dl d\theta; \quad \frac{r}{z} = \tan \alpha \Rightarrow r = z \tan \alpha; \quad z = l \cos \alpha \Rightarrow dl = \frac{dz}{\cos \alpha} \quad (D.9)$$

$$dS = \frac{\tan \alpha}{\cos \alpha} z dz d\theta \Rightarrow S = \frac{\tan \alpha}{\cos \alpha} \int z dz d\theta = \frac{\tan \alpha}{\cos \alpha} \int_0^H z dz \int_0^{2\pi} d\theta = \frac{\tan \alpha}{\cos \alpha} H^2 \pi \quad (D.10)$$

$$\tan \alpha = \frac{R}{H} \quad \text{et} \quad \cos \alpha = \frac{H}{\sqrt{R^2 + H^2}} \Rightarrow \boxed{S = \pi R \sqrt{R^2 + H^2} = \pi R a} \quad (D.11)$$

Le calcul de la surface latérale du tronc de cône de révolution peut être déduite en faisant la différence entre les surfaces du grand cône et du petit cône. Notons par S_1 la surface latérale du petit cône, S_2 celle du grand cône, a_1 l'apothème du petit cône, a_2 celui du grand cône et a la différence des apothèmes ($a_2 - a_1$).

$$S_1 = \pi r a_1; \quad S_2 = \pi R a_2; \quad \Rightarrow \quad S_3 = S_2 - S_1 = \pi(R a_2 - r a_1) \quad (D.12)$$

D'après le théorème de Thalès on a :

$$\frac{r}{R} = \frac{a_1}{a_2} \Rightarrow r a_2 = R a_1 \Rightarrow S_3 = \pi(R a_2 - r(a_2 - a)) = \pi(R a_2 - r a_2 + r a) \quad (D.13)$$

$$S_3 = \pi(R a_2 - r(a_2 - a)) = \pi(R a_2 - R a_1 + r a) = \pi(R(a_2 - a_1) + r a) = \pi(R + r) a \quad (D.14)$$

Donc la surface latérale du tronc de cône est donnée par $\boxed{S_3 = \pi(R + r) a}$. Ainsi pour le montage d'essai de traction sur Gleeble, la surface latérale $S_2 = \pi(r_1 + r_2) a$.

Implémentation numérique de la densité de courant

```

1  C   This is an UAMP for heating Gleeble system
2  C   user amplitude subroutine
3  Subroutine UAMP(
4  C   passed in for information and state variables
5  *   ampName, time, ampValueOld, dt, nProps, props, nSvars,
6  *   svars, lFlagsInfo,
7  *   nSensor, sensorValues, sensorNames,
8  *   jSensorLookUpTable,
9  C   to be defined
10 *   ampValueNew,
11 *   lFlagsDefine,
12 *   AmpDerivative, AmpSecDerivative, AmpIncIntegral,
13 *   AmpDoubleIntegral)
14
15  include 'aba_param.inc'
16
17  C   svars - additional state variables, similar to (V)UEL
18  dimension sensorValues(nSensor), svars(nSvars), props(nProps)
19  character*80 sensorNames(nSensor)
20  character*80 ampName
21
22  C   time indices
23  parameter( iStepTime      = 1,
24 *          iTotalTime     = 2,
25 *          nTime          = 2)
26  C   flags passed in for information
27  parameter( iInitialization = 1,
28 *          iRegularInc     = 2,
29 *          iCuts           = 3,
30 *          ikStep          = 4,
31 *          nFlagsInfo     = 4)
32  C   optional flags to be defined
33  parameter( iComputeDeriv   = 1,
34 *          iComputeSecDeriv = 2,
35 *          iComputeInteg    = 3,
36 *          iComputeDoubleInteg = 4,
37 *          iStopAnalysis    = 5,
38 *          iConcludeStep    = 6,
39 *          nFlagsDefine     = 6)
40
41  C   parameter( tStep=0.18d0, tAccelerateMotor = .00375d0,
42 *             omegaFinal=23.26d0,
43 *             zero=0.0d0, one=1.0d0, two=2.0d0, four=4.0d0)
44  C   parameter (pi = 3.14159265359d0)
45
46  dimension time(nTime), lFlagsInfo(nFlagsInfo),
47 * lFlagsDefine(nFlagsDefine)
48  dimension jSensorLookUpTable(*)
49
50  lFlagsDefine(iComputeDeriv)      = 0
51  lFlagsDefine(iComputeSecDeriv)   = 0
52  lFlagsDefine(iComputeInteg)      = 0

```

```

53     lFlagsDefine(iComputeDoubleInteg) = 0
54
55     C   get the current sensor value
56     locSensor = iGetSensorID('TEMP_SENSOR', jSensorLookUpTable)
57     realTemp = sensorValues(locSensor)
58     C   Temperature dependant variables
59     rE = -0.0003*realTemp**3+1.0395*realTemp**2-1248.4*realTemp+10**6
60     realDens = -0.5539 * realTemp + 8094.4
61     realSpec = 0.1633 * realTemp + 462.89
62     c   realDens = 7640.0
63     c   realSpec = 380.0
64     C   realTemp = GetSensorValue('TEMP-SENSOR',
65     * jSensorLookUpTable, sensorValues)
66     c=====
67     c-----Begin of code-----
68     c=====
69     if (ampName(1:11) .eq. 'AMP_CURRENT' ) then
70     C Initialisation of constants
71     c   ampFactor = 160.0*sqrt(5.9*realDens*realSpec)
72     ampFactor = 0.16*sqrt(rE*realDens*realSpec)
73     if (lFlagsInfo(iInitialization) .eq. 1) then
74     C Initial amplitude is sqrt(reqTemp(dt) - initTemp)
75     ampValueNew = 2.5*ampFactor
76     reqTemp = 25
77     do 10 i = 1, nSvars
78     svars(i) = ampValueNew
79     10 continue
80     else
81     C get the current time
82     tim = time(iTotalTime)
83     C ----- Start of function definition
84     if (tim.lt.617.5) then
85     reqTemp = 2.0*tim + 25
86     end if
87     if (tim.ge.617.5 .and. tim.lt.917.5) then
88     reqTemp = 1260
89     end if
90     if (tim.ge.917.5 ) then
91     c   call exit(-1)
92     end if
93     if (tim.ge.917.5 .and. tim.lt.1127.5) then
94     reqTemp = -1.0*(tim-917.5) + 1260
95     end if
96     if (tim.ge.1127.5) then
97     reqTemp = 1050
98     end if
99     C ----- End of function definition
100    corrTemp = 1.086272031*reqTemp !1.086561289*
101    if (corrTemp-realTemp .ge. 0) then
102    ampValueNew = ampFactor * sqrt(corrTemp-realTemp)
103    else
104    ampValueNew = 0.5 * ampValueOld

```

```
105         end if
106         svarSum = 0
107         do 20 i = 1, nSvars
108             svarSum = svarSum + svars(i)
109         20     continue
110     c     if (ampValueNew .gt. 0 ) then
111         ampValueNew = (ampValueNew + svarSum) / (nSvars + 1)
112     c     else
113         ampValueNew = (svarSum - ampValueOld) / (nSvars + 1)
114     c     end if
115         do 30 i = nSvars, 2, -1
116             svars(i) = svars(i-1)
117         30     continue
118         svars(1) = ampValueNew
119         end if
120         write(*,*) tim, realTemp, reqTemp, corrTemp, ampValueNew
121     end if
122     return
123 end
124
125
```

Code Python pour l'identification du modèle PTM

Les pages suivantes présentent quelques lignes de code développé pour l'identification du modèle PTM avec la bibliothèque *Lmfit*.

Les principales étapes de cette programmation sont essentiellement organisées comme suit :

- **Étape 1** : Déterminations des fonctions $I_1(\varepsilon)$ et $S_1(\varepsilon)$.
- **Étape 2** : Détermination de la fonction $S_2(\varepsilon, T)$ comme fonction dépendante de la variable T .
- **Étape 3** : Détermination des coefficients de la fonction $S_2(\varepsilon, T)$ comme des fonctions polynomiales dépendantes de la déformation ε .

```

1  # Import all the useful libraries for Python:
2
3  import numpy as np
4  import scipy as sp
5  import matplotlib.pyplot as plt
6  import pandas as pd
7  from scipy.optimize import curve_fit
8
9  colors = ['red', 'green', 'blue', 'purple', 'orange', 'black', 'yellow', 'pink', 'brown']
10
11 # Import experimental data
12 excelData = pd.read_excel('ExperimentsTreated1.xlsx', None) #srData
13 excelData_exp = pd.read_excel('ExperimentsTreated.xlsx', None) # TempData
14 excelData1 = pd.read_excel('ExperimentsTreated2.xlsx', None)
15
16 # Extract reference strain rate's data
17 dataSheet = excelData[str(eps0)]
18 tp = []
19 for i in range(1, dataSheet.shape[1]):
20     tp.append(dataSheet.columns[i])
21 tp = np.array(tp)
22 T = tp-T0
23 data = dataSheet.values
24 eps = data[:,0]
25 sig = data[:,1:]
26 logsig = np.log(sig)
27
28 # Define de constitutive law
29 def constitutiveLaw(T, I1, S1):
30     return I1 + S1*T
31
32 # Curve_fit identification
33 # S1 and I1
34 I1 = []
35 S1 = []
36 for i in range(len(eps)):
37     sig_i = logsig[i, :]
38     popt, pcov = curve_fit(constitutiveLaw, T, sig_i)
39     I_i, S_i = popt
40     I1.append(I_i)
41     S1.append(S_i)
42 J = np.exp(I1)
43
44 # Analytical formula of I1
45 def functionI1(eps, A, B, n, m, p):
46     return A + B * eps + n * eps**2 + m * eps**3 + p * eps**4
47
48 #Identification
49 popt, pcov = curve_fit(functionI1, eps, I1)
50 A, B, n, m, p = popt
51 print("A=%g, B=%g, n=%g m=%g, p=%g" %(A, B, n, m, p))
52

```

```

53 # Analytical formula of J
54 def functionJ(eps, d1, d2, d3, d4, d5):
55     return d1 + d2 * eps + d3 * eps**2 + d4 * eps**3 + d5 * eps**4
56
57 # Identification
58 popt, pcov = curve_fit(functionJ, eps, J)
59 d1, d2, d3, d4, d5 = popt
60 print("d1=%g, d2=%g, d3=%g, d4=%g, d5=%g" %(d1, d2, d3, d4, d5))
61
62 # Analytical formula of S1
63 def functionS1(eps, d13, d14, d15, d16, d17):
64     return d13 + d14 * eps + d15 * eps**2 + d16 * eps**3 + d17 * eps**4
65
66 # Identification
67 popt, pcov = curve_fit(functionS1, eps, S1)
68 d13, d14, d15, d16, d17 = popt
69 print("d13=%g, d14=%g, d15=%g, d16=%g, d17=%g" %(d13, d14, d15, d16, d17))
70
71 # Analytical formula of S2
72 def indenfunctionS2(t, S2):
73     return S2*t
74 def functionS2(T, d24, d25, d26, d27, d28):
75     return d24 + d25 * T + d26 * T**2 + d27 * T**3 + d28 * T**4
76
77 # Identification
78 d24 = []
79 d25 = []
80 d26 = []
81 d27 = []
82 d28 = []
83 for i in range(len(eps)):
84     S2 = []
85     for j in tp[0:]:
86         dataSheetj = excelData1[str(j)]
87         dataj = dataSheetj.values
88         epsj = dataj[i,0] # extraction of epsilon which corresponds to index i
89         sigj = dataj[:, 1:]
90         logsigj = np.log(sigj)
91         sig_j = logsigj[i, :] - np.log(functionJ(epsj, d1, d2, d3, d4, d5)) -      ↵
92             functionS1(epsj, d13, d14, d15, d16, d17) * (j - T0)
93         popt, pcov = curve_fit(indenfunctionS2, np.log(ep), sig_j)
94         S_j = popt
95         S2.append(S_j)
96     S2_params = np.polyfit(T, S2, 4)
97     y28, y27, y26, y25, y24 = S2_params
98     d24.append(y24)
99     d25.append(y25)
100    d26.append(y26)
101    d27.append(y27)
102    d28.append(y28)
103 # Fit of d24, d25, d26, d27, d28

```

```

104 def fun24(eps, d240, d241, d242, d243, d244):
105     return d240 + d241 * eps + d242 * eps**2 + d243 * eps**3 + d244 * eps**4
106
107     eps_j = np.linspace(eps.min(), eps.max(), 100)
108
109     d24_params = np.polyfit(eps, d24, 4)
110     d244, d243, d242, d241, d240 = d24_params
111     print("d240=%g, d241=%g, d242=%g, d243=%g, d244=%g" %(d240, d241, d242, d243, d244))
112     d25_params = np.polyfit(eps, d25, 4)
113     d254, d253, d252, d251, d250 = d25_params
114     print("d250=%g, d251=%g, d252=%g, d253=%g, d254=%g" %(d250, d251, d252, d253, d254))
115     d26_params = np.polyfit(eps, d26, 4)
116     d264, d263, d262, d261, d260 = d26_params
117     print("d260=%g, d261=%g, d262=%g, d263=%g, d264=%g" %(d260, d261, d262, d263, d264))
118     d27_params = np.polyfit(eps, d27, 4)
119     d274, d273, d272, d271, d270 = d27_params
120     print("d270=%g, d271=%g, d272=%g, d273=%g, d274=%g" %(d270, d271, d272, d273, d274))
121     d28_params = np.polyfit(eps, d28, 4)
122     d284, d283, d282, d281, d280 = d28_params
123     print("d280=%g, d281=%g, d282=%g, d283=%g, d284=%g" %(d280, d281, d282, d283, d284))
124
125     # Define the PTM model
126     def PTMLaw(eps, epsp, T):
127         d24 = d240 + d241 * eps + d242 * eps**2 + d243 * eps**3 + d244 * eps**4
128         d25 = d250 + d251 * eps + d252 * eps**2 + d253 * eps**3 + d254 * eps**4
129         d26 = d260 + d261 * eps + d262 * eps**2 + d263 * eps**3 + d264 * eps**4
130         d27 = d270 + d271 * eps + d272 * eps**2 + d273 * eps**3 + d274 * eps**4
131         d28 = d280 + d281 * eps + d282 * eps**2 + d283 * eps**3 + d284 * eps**4
132         A = d1 + d2 * eps + d3 * eps**2 + d4 * eps**3 + d5 * eps**4
133         B = d13 + d14 * eps + d15 * eps**2 + d16 * eps**3 + d17 * eps**4
134         C = d24 + d25 * (T-T0) + d26 * (T-T0)**2 + d27 * (T-T0)**3 + d28 * (T-T0)**4
135         return (A) * np.exp((B)*(T-T0) + (C) * np.log(epsp/epsp0))
136
137     # Plot the curves
138     plt.figure(figsize = (15.2, 10.75))
139     plt.rc('text', usetex = True)
140     idx = 1
141     i = 0
142     for table in range(3):
143         plt.subplot(2, 2, idx)
144         for column in range(9):
145             plt.plot(eps_exp, sig_exp[:,i], colors[5], label = 'EXP', linewidth = 3)
146             plt.plot(eps, PTMLaw(eps, epsp[idx-1], tp[column]), colors[0], label = 'PTM', marker='s', linestyle = 'none')
147             i += 1
148         plt.grid() # Plot the grid with default size
149         plt.xlabel(r'$Deformation\ \overline{\varepsilon}^p$', fontsize = 16)
150         plt.ylabel(r'$Stress\ \overline{\sigma}\ (MPa)$', fontsize = 16)
151         plt.title(r'$' + str(epsp[idx-1]) + '\ s^{-1}$', fontsize = 16)
152         idx += 1
153     plt.show()

```

Bibliographie

- [Abadi *et al.*, 2015] ABADI, M., AGARWAL, A., BARHAM, P., BREVDO, E., CHEN, Z., CITRO, C. et CORRADO, G. S. (2015). TensorFlow : Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org. 61, 74
- [Abaqus, 1989] ABAQUS (1989). *Reference Manual*. Hibbitt, Karlsson and Sorensen Inc., 100 Medway Street, Providence, RI, 02906-4402, USA. 44
- [Abspoel *et al.*, 2016] ABSPOEL, M., NEELIS, B. M. et van LIEMPT, P. (2016). Constitutive behaviour under hot stamping conditions. *Journal of Materials Processing Technology*, 228:34–42. 41
- [Abueidda *et al.*, 2021] ABUEIDDA, D. W., KORIC, S., SOBH, N. A. et SEHITOGLU, H. (2021). Deep learning for plasticity and thermo-viscoplasticity. *International Journal of Plasticity*, 136. 62
- [Alaneme et Okotete, 2019] ALANEME, K. K. et OKOTETE, E. A. (2019). Recrystallization mechanisms and microstructure development in emerging metallic materials : A review. *Journal of Science : Advanced Materials and Devices*, 4(1):19–33. 15
- [Ali *et al.*, 2019] ALI, U., MUHAMMAD, W., BRAHME, A., SKIBA, O. et INAL, K. (2019). Application of artificial neural networks in micromechanics for polycrystalline metals. *International Journal of Plasticity*, 120:205–219. 62
- [Arrhenius, 1889] ARRHENIUS, S. (1889). über die reaktionsgeschwindigkeit bei der inversion von rohrzucker durch säuren. *Zeitschrift für physikalische Chemie*, 4(1):226–248. 41
- [Ashtiani et Shahsavari, 2016] ASHTIANI, H. R. et SHAHSAVARI, P. (2016). A comparative study on the phenomenological and artificial neural network models to predict hot deformation behavior of AlCuMgPb alloy. *Journal of Alloys and Compounds*, 687:263–273. 62
- [Atkinson, 1982] ATKINSON, M. (1982). Plastic stress-strain relationships of some polycrystalline metals tested in biaxial tension to high strain. In *Strength of Metals and Alloys (ICSMA 6)*, pages 481–486. Elsevier. 30, 32
- [Bäker, 2006] BÄKER, M. (2006). Finite element simulation of high-speed cutting forces. *Journal of Materials Processing Technology*, 176(1–3):117–126. 34, 132
- [Bennett *et al.*, 2010] BENNETT, C. J., LEEN, S. B., WILLIAMS, E. J., SHIPWAY, P. H. et HYDE, T. H. (2010). A critical analysis of plastic flow behaviour in axisymmetric isothermal and Gleeble compression testing. *Computational Materials Science*, 50(1):125–137. 3, 125
- [Bomarito *et al.*, 2021] BOMARITO, G., TOWNSEND, T., STEWART, K., ESHAM, K., EMERY, J. et HOCHHALTER, J. (2021). Development of interpretable, data-driven plasticity models with symbolic regression. *Computers & Structures*, 252. 7
- [Brown *et al.*, 1997] BROWN, S. G. R., JAMES, J. D. et SPITTLE, J. A. (1997). A 3D numerical model of the temperature - time characteristics of specimens tested on a Gleeble thermomechanical simulator. *Modelling and Simulation in Materials Science and Engineering*, 5(6):539–548. Publisher : IOP Publishing. 2, 10, 125, 132
- [Buzolin *et al.*, 2023] BUZOLIN, R. H., MASSWOHL, M., FERRAZ, F. M. B., CHRZAN, K., DUDZIAK, T. et POLLETTI, M. C. (2023). Microstructure refinement of a cast high entropy alloy by thermomechanical treatments. *Materials Science and Engineering : A*, 872:144931. 101
- [Chadha *et al.*, 2018a] CHADHA, K., AHMED, Z., ARANAS JR, C., SHAHRIARI, D. et JHAZI, M. (2018a). Influence of strain rate on dynamic transformation of austenite in an as-cast medium-carbon low-alloy steel. *Materialia*, 1:155–167. 1, 6
- [Chadha *et al.*, 2018b] CHADHA, K., SHAHRIARI, D. et JHAZI, M. (2018b). An approach to develop hansel-spittel constitutive equation during ingot breakdown operation of low alloy steels. In *Frontiers in Materials Processing, Applications, Research and Technology : Select Proceedings of FiMPART 2015*, pages 239–246. Springer. 41

- [Chadha *et al.*, 2017] CHADHA, K., SHAHRIARI, D., TREMBLAY, R., BHATTACHARJEE, P. P. et JHAZI, M. (2017). Deformation and recrystallization behavior of the cast structure in large size, high strength steel ingots : experimentation and modeling. *Metallurgical and Materials Transactions A*, 48:4297–4313. 1, 6
- [Chadha *et al.*, 2020] CHADHA, K., TIAN, Y., BOCHER, P., SPRAY, J. G. et ARANAS JR, C. (2020). Microstructure evolution, mechanical properties and deformation behavior of an additively manufactured maraging steel. *Materials*, 13(10):2380. 1, 6
- [Chen *et al.*, 2014] CHEN, X.-M., LIN, Y., WEN, D.-X., ZHANG, J.-L. et HE, M. (2014). Dynamic recrystallization behavior of a typical nickel-based superalloy during hot deformation. *Materials & Design*, 57:568–577. 114
- [Coër *et al.*, 2011] COËR, J., BERNARD, C., LAURENT, H., ANDRADE-CAMPOS, A. et THUILLIER, S. (2011). The Effect of Temperature on Anisotropy Properties of an Aluminium Alloy. *Experimental Mechanics*, 51(7):1185–1195. 128
- [Coupard, 1993] COUPARD, D. (1993). *Etude du phénomène d'absorption ou de rejet d'un renfort fibreux au cours de la solidification de l'aluminium liquide*. phdthesis, Université Sciences et Technologies - Bordeaux I. 34
- [Cowper et Symonds, 1957] COWPER, G. R. et SYMONDS, P. S. (1957). Strain-hardening and strain-rate effects in the impact loading of cantilever beams. Rapport technique, Brown Univ Providence Ri. 32
- [Dalverny *et al.*, 2002] DALVERNY, O., CAPERAA, S., PANTALÉ, O. et SATTOUF, C. (2002). Identification de lois constitutives et de lois de frottement adaptées aux grandes vitesses de sollicitation. *Journal de Physique IV - Proceedings*, 12(11):275–282. 73
- [Doherty *et al.*, 1997] DOHERTY, R., HUGHES, D., HUMPHREYS, F., JONAS, J. J., JENSEN, D. J., KASSNER, M., KING, W., McNELLEY, T., McQUEEN, H. et ROLLETT, A. (1997). Current issues in recrystallization : a review. *Materials Science and Engineering : A*, 238(2):219–274. 10
- [Duc-Toan *et al.*, 2012] DUC-TOAN, N., TIEN-LONG, B., DONG-WON, J., SEUNG-HAN, Y. et YOUNG-SUK, K. (2012). A Modified Johnson–Cook Model to Predict Stress-strain Curves of Boron Steel Sheets at Elevated and Cooling Temperatures. *High Temperature Materials and Processes*, 31:37–45. 32
- [Ebrahimi *et al.*, 2012] EBRAHIMI, G., KESHMIRI, H., MALDAD, A. et MOMENI, A. (2012). Dynamic recrystallization behavior of 13% cr martensitic stainless steel under hot working condition. *Journal of Materials Science & Technology*, 28(5):467–473. 101
- [Ebrahimi *et al.*, 2013] EBRAHIMI, G., KESHMIRI, H., MAZINANI, M., MALDAR, A. et HAGHSHENAS, M. (2013). Multi-stage thermomechanical behavior of aisi 410 martensitic steel. *Materials Science and Engineering : A*, 559:520–527. 14
- [Ebrahimi *et al.*, 2011a] EBRAHIMI, G., KESHMIRI, H. et MOMENI, A. (2011a). Effect of heat treatment variables on microstructure and mechanical properties of 15cr–4ni–0.08c martensitic stainless steel. *Ironmaking & Steelmaking*, 38(2):123–128. 101
- [Ebrahimi *et al.*, 2011b] EBRAHIMI, G., KESHMIRI, H., MOMENI, A. et MAZINANI, M. (2011b). Dynamic recrystallization behavior of a superaustenitic stainless steel containing 16% cr and 25% ni. *Materials Science and Engineering : A*, 528(25–26):7488–7493. 14
- [Ebrahimi *et al.*, 2018] EBRAHIMI, G., MOMENI, A. et EZATPOUR, H. (2018). Modeling the viscoplastic behavior and grain size in a hot worked nb-bearing high-mn steel. *Materials Science and Engineering : A*, 714:25–35. 15
- [Ebrahimi *et al.*, 2017] EBRAHIMI, G., MOMENI, A., KAZEMI, S. et ALINEJAD, H. (2017). Flow curves, dynamic recrystallization and precipitation in a medium carbon low alloy steel. *Vacuum*, 142:135–145. 2, 6
- [El Mehtedi *et al.*, 2015] EL MEHTEDI, M., SPIGARELLI, S., GABRIELLI, F. et DONATI, L. (2015). Comparison study of constitutive models in predicting the hot deformation behavior of aa6060 and aa6063 aluminium alloys. *Materials Today : Proceedings*, 2(10):4732–4739. 41

- [Fekete et Szekeres, 2015] FEKETE, B. et SZEKERES, A. (2015). Investigation on partition of plastic work converted to heat during plastic deformation for reactor steels based on inverse experimental-computational method. *European Journal of Mechanics - A/Solids*, 53:175–186. 128
- [Ferguson et Lindeman, 2006] FERGUSON, D. E. et LINDEMAN, N. A. (2006). Apparatus for an anvil system that provides enhanced self-resistive specimen heating when used in a dynamic thermo-mechanical material testing system. 10, 11
- [Galos *et al.*, 2022] GALOS, J., DAS, R., SUTCLIFFE, M. P. et MOURITZ, A. P. (2022). Review of balsa core sandwich composite structures. *Materials & Design*, page 111013. 23
- [Ganapathy *et al.*, 2018] GANAPATHY, M., LI, N., LIN, J., ABSPOEL, M. et BHATTACHARJEE, D. (2018). A Novel Grip Design for High-Accuracy Thermo-Mechanical Tensile Testing of Boron Steel under Hot Stamping Conditions. *Experimental Mechanics*, 58(2):243–258. 44, 46, 125
- [Ganapathy *et al.*, 2015] GANAPATHY, M., LI, N., LIN, J., ABSPOEL, M., GUIDO, H. et BHATTACHARJEE, D. (2015). Analysis of new Gleeble tensile specimen design for hot stamping application. *MATEC Web of Conferences*, 21:05013. Publisher : EDP Sciences. 42
- [Gao, 2007] GAO, C. Y. (2007). FE realization of a thermo-visco-plastic constitutive model using VU-MAT in Abaqus/Explicit program. In *Computational Mechanics*, pages 301–301, Berlin, Heidelberg. Springer Berlin Heidelberg. 41
- [Gao *et al.*, 2020] GAO, F., LIU, W., ZHU, Q., GAO, Z., MISRA, R. D. K., LIU, Z. et YU, F. (2020). Flow behaviour and constitutive modeling for hot deformation of austenitic stainless steel. *Materials Research Express*, 7(11):116512. 41, 42, 44
- [Ghaboussi *et al.*, 1991] GHABOUSSI, J., GARRETT, J. H. et WU, X. (1991). Knowledge-Based Modeling of Material Behavior with Neural Networks. *Journal of Engineering Mechanics*, 117(1):132–153. 61
- [Ghaboussi *et al.*, 1998] GHABOUSSI, J., PECKNOLD, D. A., ZHANG, M. et HAJ-ALI, R. M. (1998). Autoprogressive training of neural network constitutive models. page 22. 61
- [Ghaboussi et Sidarta, 1998] GHABOUSSI, J. et SIDARTA, D. (1998). New nested adaptive neural networks (NANN) for constitutive modeling. *Computers and Geotechnics*, 22(1):29–52. 61
- [Goodfellow *et al.*, 2014] GOODFELLOW, I., POUGET-ABADIE, J., MIRZA, M., XU, B., WARDE-FARLEY, D., OZAIR, S., COURVILLE, A. et BENGIO, Y. (2014). Generative adversarial nets. In GHAHRAMANI, Z., WELLING, M., CORTES, C., LAWRENCE, N. et WEINBERGER, K., éditeurs : *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc. 61
- [Goodfellow *et al.*, 2016] GOODFELLOW, I. J., BENGIO, Y. et COURVILLE, A. (2016). *Deep Learning*. MIT Press, Cambridge, MA, USA. 65
- [Gorji *et al.*, 2020] GORJI, M. B., MOZAFFAR, M., HEIDENREICH, J. N., CAO, J. et MOHR, D. (2020). On the potential of recurrent neural networks for modeling path dependent plasticity. *Journal of the Mechanics and Physics of Solids*, 143. 61
- [Graves, 2013] GRAVES, A. (2013). Generating sequences with recurrent neural networks. *CoRR*, abs/1308.0850. 61
- [Gupta *et al.*, 2013] GUPTA, A. K., ANIRUDH, V. et SINGH, S. K. (2013). Constitutive models to predict flow stress in austenitic stainless steel 316 at elevated temperatures. *Materials & Design*, 43:410–418. 42, 44
- [Gurusamy et Rao, 2017] GURUSAMY, M. M. et RAO, B. C. (2017). On the performance of modified zerilli-armstrong constitutive model in simulating the metal-cutting process. *Journal of Manufacturing Processes*, 28:253–265. 42, 44
- [He *et al.*, 2014] HE, A., XIE, G., ZHANG, H. et WANG, X. (2014). A modified Zerilli–Armstrong constitutive model to predict hot deformation behavior of 20CrMo alloy steel. *Materials & Design (1980–2015)*, 56:122–127. 41, 42
- [He *et al.*, 2023] HE, T., WANG, L., HU, F., ZHOU, W., ZHANG, Z. et WU, K. (2023). Stability of retained austenite and work hardening behavior in ultra-fine medium carbon bainitic steel. *Journal of Materials Research and Technology*, 22:2690–2703. 1

- [Hensel et Spittel, 1978] HENSEL, A. et SPITTEL, T. (1978). *Kraft-und Arbeitsbedarf bildsamer Formgebungsverfahren : mit 51 Tabellen*. Dt. Verlag für Grundstoffindustrie. 41
- [Heudin, 2016] HEUDIN, J.-C. (2016). *Comprendre le deep learning : une introduction aux réseaux de neurones*. Science-eBook. 65
- [Holmquist et Johnson, 1991] HOLMQUIST, T. et JOHNSON, G. (1991). Determination of constants and comparison of results for various constitutive models. *Le Journal de Physique IV*, 1(C3):C3–853. 32
- [Hor, 2011] HOR, A. (2011). *Simulation physique des conditions thermomécaniques de forgeage et d'usinage : caractérisation et modélisation de la rhéologie et de l'endommagement*. Thèse de doctorat, Arts et Métiers ParisTech. 29, 31, 37
- [Hornik et al., 1989] HORNIK, K., STINCHCOMBE, M. et WHITE, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366. 60
- [Huang et al., 2021] HUANG, X., ZANG, Y. et GUAN, B. (2021). Constitutive models and microstructure evolution of Ti-6Al-4V alloy during the hot compressive process. *Materials Research Express*, 8(1). 62
- [Humphreys et Hatherly, 2004] HUMPHREYS, F. et HATHERLY, M. (2004). Hot deformation and dynamic restoration. *Recrystallization and related annealing phenomena*, pages 415–V. 9, 13, 14, 15, 16, 19, 20
- [Humphreys et Hatherly, 2012] HUMPHREYS, F. J. et HATHERLY, M. (2012). *Recrystallization and related annealing phenomena*. Elsevier. 11
- [Jamli et Farid, 2019] JAMLI, M. et FARID, N. (2019). The sustainability of neural network applications within finite element analysis in sheet metal forming : A review. *Measurement*, 138:446–460. 61
- [Jansen van Rensburg et Kok, 2012] JANSEN VAN RENSBURG, G. et KOK, S. (2012). Tutorial on state variable based plasticity : An Abaqus UHARD subroutine. In *Eighth South African Conference on Computational and Applied Mechanics - SACAM2012*, Johannesburg. 86
- [Javadi et Rezaia, 2009] JAVADI, A. et REZANIA, M. (2009). Intelligent finite element method : An evolutionary approach to constitutive modeling. *Advanced Engineering Informatics*, 23:442–541. 61
- [Jiang et al., 2021] JIANG, W., LONG, M., CHEN, D., AI, S. et DUAN, H. (2021). Temperature distribution in the as-cast steel specimen during gleeble hot-tensile test and its effect on high-temperature mechanical properties. *Metallurgical and Materials Transactions B*, 52:1228–1242. 125
- [Jiao et Alavi, 2020] JIAO, P. et ALAVI, A. H. (2020). Artificial intelligence-enabled smart mechanical metamaterials : Advent and future trends. *International Materials Reviews*, pages 1–29. 61
- [Johnson et Cook, 1983] JOHNSON, G. R. et COOK, W. H. (1983). A Constitutive Model and Data for Metals Subjected to Large Strains, High Strain Rates, and High Temperatures. In *Proceedings 7th International Symposium on Ballistics*, pages 541–547, The Hague. 31, 32, 73, 89
- [Jonas, 1994] JONAS, J. (1994). Dynamic recrystallization—scientific curiosity or industrial tool? *Materials Science and Engineering : A*, 184(2):155–165. 10
- [Jonas et Poliak, 2003] JONAS, J. J. et POLIAK, E. I. (2003). The critical strain for dynamic recrystallization in rolling mills. In *Materials Science Forum*, volume 426, pages 57–66. Trans Tech Publ. 102
- [Jonas et al., 2009] JONAS, J. J., QUELENNEC, X., JIANG, L. et MARTIN, E. (2009). The avrami kinetics of dynamic recrystallization. *Acta materialia*, 57(9):2748–2756. 12
- [Karanjgaokar et al., 2011] KARANJGAOKAR, N. J., OH, C.-S. et CHASIOTIS, I. (2011). Microscale Experiments at Elevated Temperatures Evaluated with Digital Image Correlation. *Experimental Mechanics*, 51(4):609–618. 11
- [Kardoulaki et al., 2014] KARDOULAKI, E., LIN, J., BALINT, D. et FARRUGIA, D. (2014). Investigation of the effects of thermal gradients present in gleeble high-temperature tensile tests on the strain state for free cutting steel. *The Journal of Strain Analysis for Engineering Design*, 49(7):521–532. 3, 42, 125, 128, 132

- [Kim *et al.*, 1996] KIM, K.-h., OH, K. H. et LEE, D. N. (1996). Mechanical behavior of carbon steels during continuous casting. *Scripta Materialia*, 34(2):301–307. 2, 125
- [Kingma et Lei, 2014] KINGMA, D. P. et LEI, J. (2014). Adam : A Method for Stochastic Optimization. *arXiv preprint arXiv :1412.6980*, page 15. 74
- [Klepaczko, 1975] KLEPACZKO, J. (1975). Thermally activated flow and strain rate history effects for some polycrystalline fcc metals. *Materials Science and Engineering*, 18(1):121–135. 38
- [Klepaczko, 1987] KLEPACZKO, J. (1987). A practical stress-strain-strain rate-temperature constitutive relation of the power form. *Journal of Mechanical Working Technology*, 15(2):143–165. 38
- [Knight et Nowotny, 2021] KNIGHT, J. C. et NOWOTNY, T. (2021). Larger GPU-accelerated brain simulations with procedural connectivity. *Nature Computational Science*, 1(2):136–142. 62
- [Kocks, 1975] KOCKS, U. (1975). Thermodynamics and kinetics of slip. *Prog. in Mater. Sci.*, 242. 39
- [Krizhevsky *et al.*, 2012] KRIZHEVSKY, A., SUTSKEVER, I. et HINTON, G. E. (2012). Imagenet classification with deep convolutional neural networks. In PEREIRA, F., BURGESS, C., BOTTOU, L. et WEINBERGER, K., éditeurs : *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc. 61
- [Kumar, 2016] KUMAR, V. (2016). Thermo-mechanical simulation using gleeble system-advantages and limitations. *Journal of Metallurgy and Materials Science*, 58(2):81–88. 3
- [Laasraoui et Jonas, 1991] LAASRAOUI, A. et JONAS, J. J. (1991). Prediction of steel flow stresses at high temperatures and strain rates. *Metallurgical transactions A*, 22:1545–1558. 9
- [Li *et al.*, 2013] LI, H.-Y., WANG, X.-F., DUAN, J.-Y. et LIU, J.-J. (2013). A modified Johnson Cook model for elevated temperature flow behavior of T24 steel. *Materials Science and Engineering : A*, 577:138–146. 36
- [Li *et al.*, 2019a] LI, L., WANG, Y., LI, H., JIANG, W., WANG, T., ZHANG, C.-C., WANG, F. et GARMESTANI, H. (2019a). Effect of the zener-hollomon parameter on the dynamic recrystallization kinetics of mg-zn-zr-yb magnesium alloy. *Computational Materials Science*, 166:221–229. 114
- [Li *et al.*, 2016] LI, N., SHAO, Z. T., LIN, J. G. et DEAN, T. A. (2016). Investigation of Uniaxial Tensile Properties of AA6082 under HFQ® Conditions. *Key Engineering Materials*, 716:337–344. Conference Name : Metal Forming 2016 ISBN : 9783038357049 Publisher : Trans Tech Publications Ltd. 128
- [Li *et al.*, 2019b] LI, X., ROTH, C. C. et MOHR, D. (2019b). Machine-learning based temperature- and rate-dependent plasticity model : Application to analysis of fracture experiments on DP steel. *International Journal of Plasticity*, 118:320–344. 62
- [Liang *et al.*, 2022] LIANG, P., KONG, N., ZHANG, J. et LI, H. (2022). A modified arrhenius-type constitutive model and its implementation by means of the safe version of newton-raphson method. *steel research international*. 55, 79, 87
- [Lin et Chen, 2010] LIN, Y. et CHEN, X.-M. (2010). A combined Johnson-Cook and Zerilli-Armstrong model for hot compressed typical high-strength alloy steel. *Computational Materials Science*, 49(3):628–633. 42
- [Lin *et al.*, 2010] LIN, Y., CHEN, X.-M. et LIU, G. (2010). A modified Johnson-Cook model for tensile behaviors of typical high-strength alloy steel. *Materials Science and Engineering : A*, 527(26):6980–6986. 36
- [Lin *et al.*, 2012] LIN, Y., LI, Q.-F., XIA, Y.-C. et LI, L.-T. (2012). A phenomenological constitutive model for high temperature flow stress prediction of Al-Cu-Mg alloy. *Materials Science and Engineering : A*, 534:654–662. 36
- [Lin *et al.*, 2008a] LIN, Y., ZHANG, J. et ZHONG, J. (2008a). Application of neural networks to predict the elevated temperature flow behavior of a low alloy steel. *Computational Materials Science*, 43(4):752–758. 61, 70

- [Lin *et al.*, 2009] LIN, Y.-C., CHEN, M.-S. et ZHANG, J. (2009). Modeling of flow stress of 42CrMo steel under hot compression. *Materials Science and Engineering : A*, 499(1):88–92. 41
- [Lin *et al.*, 2008b] LIN, Y. C., CHEN, M.-S. et ZHONG, J. (2008b). Effect of temperature and strain rate on the compressive deformation behavior of 42CrMo steel. *Journal of Materials Processing Technology*, 205(1):308–315. 41
- [Lindeman, 2008] LINDEMAN, N. A. (2008). Technique for applying direct resistance heating current to a specific location in a specimen under test while substantially reducing thermal gradients in the specimen gauge length. 11
- [Liu *et al.*, 2006] LIU, Y., FOSTER, A., LIN, J., FARRUGIA, D. et DEAN, T. (2006). On micro-damage in hot metal working. part 1 : Experimental investigation. *Engineering transactions*, 54(4):271–287. 3
- [Liu *et al.*, 2020] LIU, Y., LI, M., REN, X.-w., XIAO, Z.-b., ZHANG, X.-y. et HUANG, Y.-c. (2020). Flow stress prediction of Hastelloy C-276 alloy using modified Zerilli–Armstrong, Johnson–Cook and Arrhenius-type constitutive models. *Transactions of Nonferrous Metals Society of China*, 30(11). 42
- [Loucif *et al.*, 2018] LOUCIF, A., BEN FREDJ, E., HARRIS, N., SHAHRIARI, D., JAHAZI, M. et LAPIERRE-BOIRE, L.-P. (2018). Evolution of a-type macrosegregation in large size steel ingot after multistep forging and heat treatment. *Metallurgical and Materials Transactions B*, 49:1046–1055. 2
- [Loucif *et al.*, 2015] LOUCIF, A., FREDJ, E. B., JAHAZI, M., LAPIERREBOIRE, L.-P., TREMBLAY, R. et BEAUVAIS, R. (2015). Analysis of macrosegregation in large size forged ingot of high strength steel. 2
- [Loucif *et al.*, 2017] LOUCIF, A., SHAHRIARI, D., ZHANG, C., JAHAZI, M., LAPIERRE-BOIRE, L. P. et TREMBLAY, R. (2017). Macroscopic segregation of alloying elements in hot top of large size high strength steel ingot. *In Materials Science Forum*, volume 879, pages 1176–1181. Trans Tech Publ. 2
- [Loucif *et al.*, 2020] LOUCIF, A., ZHANG, C., JAHAZI, M. et LAPIERRE-BOIRE, L. P. (2020). Macroscopic segregation characteristics of ferrite and austenite stabilizer elements in large size high strength steel ingot. *In Key Engineering Materials*, volume 846, pages 82–86. Trans Tech Publ. 2
- [Lu *et al.*, 2011] LU, Z., PAN, Q., LIU, X., QIN, Y., HE, Y. et CAO, S. (2011). Artificial neural network prediction to the hot compressive deformation behavior of Al–Cu–Mg–Ag heat-resistant aluminum alloy. *Mechanics Research Communications*, 38(3):192–197. 61, 70
- [Ludwik, 1909] LUDWIK, P. (1909). *Elemente der technologischen Mechanik*. Springer. 30, 32
- [Luo *et al.*, 2005] LUO, J., YING, K. et BAI, J. (2005). Savitzky–golay smoothing and differentiation filter for even number data. *Signal Processing*, 85(7):1429–1434. 22
- [Lurdos, 2008] LURDOS, O. (2008). *Lois de comportement et recristallisation dynamique : approches empirique et physique*. These de doctorat, Saint-Etienne, EMSE. 29, 30, 34, 35, 36
- [Luton et Sellars, 1969] LUTON, M. et SELLARS, C. (1969). Dynamic recrystallization in nickel and nickel-iron alloys during high temperature deformation. *Acta Metallurgica*, 17(8):1033–1043. 11
- [Marc, 2010] MARC, M. (2010). Volume a : Theory and user information. *MSC. Software Corporation*. 128
- [Martins *et al.*, 2016] MARTINS, J. M. P., ALVES, J. L., NETO, D. M., OLIVEIRA, M. C. et MENEZES, L. F. (2016). Numerical analysis of different heating systems for warm sheet metal forming. *Int J Adv Manuf Technol*, page 13. 128
- [Masi *et al.*, 2021] MASI, F., STEFANO, I., VANNUCCI, P. et MAFFI-BERTHIER, V. (2021). Thermodynamics-based Artificial Neural Networks for constitutive modeling. *Journal of the Mechanics and Physics of Solids*, 147. 62
- [Mattmann, 2020] MATTMANN, C. (2020). *Machine Learning with Tensorflow*. O'REILLY MEDIA, S.I. 74
- [McCulloch et Pitts, 1943] MCCULLOCH, W. S. et PITTS, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133. 60
- [McQueen et Jonas, 1975] MCQUEEN, H. et JONAS, J. (1975). *Recovery and recrystallization during high temperature deformation*, volume 6. Elsevier. 101

- [Ming et Pantalé, 2018] MING, L. et PANTALÉ, O. (2018). An efficient and robust VUMAT implementation of elastoplastic constitutive laws in Abaqus/Explicit finite element code. *Mechanics & Industry*, 19(3):308. 72, 86, 87, 88, 92, 93
- [Minsky et Papert, 1969] MINSKY, M. L. et PAPERT, S. (1969). *Perceptrons; an Introduction to Computational Geometry*. MIT Press. 60
- [Mirzadeh et Najafzadeh, 2010] MIRZADEH, H. et NAJAFZADEH, A. (2010). Flow stress prediction at hot working conditions. *Materials Science and Engineering : A*, 527(4-5):1160–1164. 41
- [Montheillet, 1996] MONTHEILLET, F. (1996). Métallurgie en mise en forme. *Techniques de l'ingénieur. Matériaux métalliques*, (M600):M600–1. 9, 11
- [Montheillet, 2004] MONTHEILLET, F. (2004). Moving grain boundaries during hot deformation of metals : dynamic recrystallization. In *Moving interfaces in crystalline solids*, pages 203–256. Springer. 8
- [Montheillet, 2008] MONTHEILLET, F. (2008). *Métallurgie en mise en forme à froid*. Ed. Techniques Ingénieur. 8
- [Montheillet, 2009] MONTHEILLET, F. (2009). Métallurgie en mise en forme à chaud. *Techniques de l'ingénieur. Matériaux métalliques*, (M3031). 10, 13
- [Montheillet et Desrayaud, 2009] MONTHEILLET, F. et DESRAYAUD, C. (2009). Essais rhéologiques à chaud. *Techniques de l'ingénieur. Matériaux métalliques*, (M3009). 8, 17, 18, 19, 20
- [Mulyadi *et al.*, 2006] MULYADI, M., RIST, M., EDWARDS, L. et BROOKS, J. (2006). Parameter optimisation in constitutive equations for hot forging. *Journal of Materials Processing Technology*, 177(1-3):311–314. 35
- [Murugesan et Jung, 2019] MURUGESAN, M. et JUNG, D. W. (2019). Two flow stress models for describing hot deformation behavior of aisi-1045 medium carbon steel at elevated temperatures. *Heliyon*, 5(4). 1, 7
- [Murugesan *et al.*, 2019] MURUGESAN, M., SAJJAD, M. et JUNG, D. W. (2019). Hybrid machine learning optimization approach to predict hot deformation behavior of medium carbon steel material. *Metals*, 9(12):1315. 1, 6
- [Murugesan *et al.*, 2023] MURUGESAN, M., YU, J.-H., JUNG, K.-S., CHO, S.-M., BHANDARI, K. S., CHUNG, W. et LEE, C.-W. (2023). Supervised machine learning approach for modeling hot deformation behavior of medium carbon steel. *steel research international*, 94(2):2200188. 1
- [Najafzadeh et Jonas, 2006] NAJAFZADEH, A. et JONAS, J. J. (2006). Predicting the critical stress for initiation of dynamic recrystallization. *ISIJ international*, 46(11):1679–1684. 102, 104, 105
- [Nassr *et al.*, 2018] NASSR, A., JAVADI, A. et FARAMARZI, A. (2018). Developing constitutive models from EPR-based self-learning finite element analysis. *International Journal for Numerical and Analytical Methods in Geomechanics*, 42(3):401–417. 6
- [Neto *et al.*, 2020] NETO, D. M., SIMÕES, V. M., OLIVEIRA, M. C., ALVES, J. L., LAURENT, H., OUDRISS, A. et MENEZES, L. F. (2020). Experimental and numerical analysis of the heat generated by plastic deformation in quasi-static uniaxial tensile tests. *Mechanics of Materials*, 146:103398. 41
- [Newville *et al.*, 2016] NEWVILLE, M., STENSITZKI, T., ALLEN, D. B., RAWLIK, M., INGARGIOLA, A. et NELSON, A. (2016). Lmfit : Non-linear least-square minimization and curve-fitting for python. *Astrophysics Source Code Library*, pages ascl-1606. 45, 49
- [Norris et Wilson, 1999] NORRIS, S. et WILSON, I. (1999). Application of 3d numerical modelling for thermal profile optimization on the gleeble thermomechanical simulator. *Modelling and Simulation in Materials Science and Engineering*, 7(3):297. 2, 10, 125
- [Pantalé, 2021] PANTALÉ, O. (2021). ANN VUHARD repository. Software Heritage archive. 92
- [Pantalé *et al.*, 2004] PANTALÉ, O., CAPERAA, S. et RAKOTOMALALA, R. (2004). Development of an object-oriented finite element program : application to metal-forming and impact simulations. *Journal of Computational and Applied Mathematics*, 168(1):341–351. 86, 87

- [Pantalé, 2023] PANTALÉ, O. (2023). Development and implementation of an ann based flow law for numerical simulations of thermo-mechanical processes at high temperatures in fem software. *Algorithms*, 16(56). 77, 92
- [Pantalé et al., 2022] PANTALÉ, O., TIZE MHA, P. et TONGNE, A. (2022). Efficient implementation of non-linear flow law using neural network into the abaqus explicit fem code. *Finite Elements in Analysis & Design*, 198:103647. 77, 92
- [Parizeau, 2004] PARIZEAU, M. (2004). Réseaux de neurones gif-21140 et gif-64326. *Université laval*. 65
- [Park et Cho, 2021] PARK, H. et CHO, M. (2021). Multiscale constitutive model using data-driven yield function. *Composites Part B : Engineering*, 216. 28
- [Paszke et al., 2019] PASZKE, A., GROSS, S., MASSA, F., LERER, A., BRADBURY, J., CHANAN, G., KILLEEN, T., LIN, Z., GIMELSHEIN, N., ANTIGA, L., DESMAISON, A., KOPF, A., YANG, E., DEVITO, Z., RAISON, M., TEJANI, A., CHILAMKURTHY, S., STEINER, B., FANG, L., BAI, J. et CHINTALA, S. (2019). Pytorch : An imperative style, high-performance deep learning library. *In Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc. 61
- [Perdrix et al., 1981] PERDRIX, C., MY, P. et MONTHEILLET, F. (1981). Comportement mécanique et évolution structurale de l'aluminium au cours d'une déformation à chaud de grande amplitude. 9
- [Phaniraj et Lahiri, 2003] PHANIRAJ, M. P. et LAHIRI, A. K. (2003). The applicability of neural network model to predict flow stress for carbon steels. *Journal of Materials Processing Technology*, 141(2): 219–227. 47
- [Pickering, 2013] PICKERING, E. J. (2013). Macro-segregation in steel ingots : the applicability of modelling and characterisation techniques. *ISIJ international*, 53(6):935–949. 2
- [Poliak et Jonas, 1996] POLIAK, E. I. et JONAS, J. (1996). A one-parameter approach to determining the critical conditions for the initiation of dynamic recrystallization. *Acta materialia*, 44(1):127–136. 14, 15, 102
- [Poliak et Jonas, 2003a] POLIAK, E. I. et JONAS, J. J. (2003a). Critical strain for dynamic recrystallization in variable strain rate hot deformation. *Isij International*, 43(5):692–700. 102
- [Poliak et Jonas, 2003b] POLIAK, E. I. et JONAS, J. J. (2003b). Initiation of dynamic recrystallization in constant strain rate hot deformation. *ISIJ international*, 43(5):684–691. 102
- [Ponthot, 2002] PONTHOT, J. P. (2002). Unified stress update algorithms for the numerical simulation of large deformation elasto-plastic and elasto-viscoplastic processes. *International Journal of Plasticity*, page 36. 87, 92
- [Prager, 1955] PRAGER, W. (1955). The theory of plasticity : a survey of recent achievements. *Proceedings of the Institution of Mechanical Engineers*, 169(1):41–57. 30
- [Quan et al., 2016] QUAN, G.-Z., PAN, J. et ZHANG, Z.-h. (2016). Phase transformation and recrystallization kinetics in space-time domain during isothermal compressions for Ti-6Al-4V analyzed by multi-field and multi-scale coupling FEM. *Materials & Design*, 94:523–535. 42
- [Razali et Joun, 2021] RAZALI, M. K. et JOUN, M. S. (2021). A new approach of predicting dynamic recrystallization using directly a flow stress model and its application to medium mn steel. *Journal of Materials Research and Technology*, 11:1881–1894. 111, 113
- [Rosenblatt, 1957] ROSENBLATT, F. (1957). The perceptron a perceiving and recognizing automaton. Rapport technique 85-460-1, Cornell Aeronautical Laboratory, Ithaca, New York. 60
- [Rosenblatt, 1958] ROSENBLATT, F. (1958). The perceptron : A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6):386–408. Place : US Publisher : American Psychological Association. 60, 63
- [Rudnytskyj et al., 2020] RUDNYTSKYI, A., SIMON, P., JECH, M. et GACHOT, C. (2020). Constitutive modelling of the 6061 aluminium alloy under hot rolling conditions and large strain ranges. *Materials & Design*, 190:108568. 41

- [Rudra *et al.*, 2019] RUDRA, A., DAS, S. et DASGUPTA, R. (2019). Constitutive modeling for hot deformation behavior of al-5083+ sic composite. *Journal of Materials Engineering and Performance*, 28:87–99. 2, 7
- [Rumelhart *et al.*, 1986] RUMELHART, D. E., HINTON, G. E. et WILLIAMS, R. J. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088):533–536. 60
- [Ryan et McQueen, 1989] RYAN, N. et MCQUEEN, H. (1989). Dynamic recovery, strain hardening and flow stress in hot working of 316 steel. *Czechoslovak Journal of Physics B*, 39:458–465. 102
- [Ryan et McQueen, 1990a] RYAN, N. et MCQUEEN, H. (1990a). Dynamic softening mechanisms in 304 austenitic stainless steel. *Canadian Metallurgical Quarterly*, 29(2):147–162. 102
- [Ryan et McQueen, 1990b] RYAN, N. et MCQUEEN, H. (1990b). Flow stress, dynamic restoration, strain hardening and ductility in hot working of 316 steel. *Journal of Materials Processing Technology*, 21(2):177–199. 102
- [Sakai et Jonas, 1984] SAKAI, T. et JONAS, J. J. (1984). Overview no. 35 dynamic recrystallization : mechanical and microstructural considerations. *Acta Metallurgica*, 32(2):189–209. 18
- [Samantaray *et al.*, 2009a] SAMANTARAY, D., MANDAL, S. et BHADURI, A. (2009a). A comparative study on johnson cook, modified zerilli–armstrong and arrhenius–type constitutive models to predict elevated temperature flow behaviour in modified 9cr–1mo steel. *Computational Materials Science*, 47(2):568–576. 42, 44
- [Samantaray *et al.*, 2011a] SAMANTARAY, D., MANDAL, S. et BHADURI, A. (2011a). Characterization of deformation instability in modified 9cr–1mo steel during thermo–mechanical processing. *Materials & Design*, 32(2):716–722. 41
- [Samantaray *et al.*, 2011b] SAMANTARAY, D., MANDAL, S., BHADURI, A., VENUGOPAL, S. et SIVAPRASAD, P. (2011b). Analysis and mathematical modelling of elevated temperature flow behaviour of austenitic stainless steels. *Materials Science and Engineering : A*, 528(4-5):1937–1943. 44
- [Samantaray *et al.*, 2009b] SAMANTARAY, D., MANDAL, S., BORAH, U., BHADURI, A. et SIVAPRASAD, P. (2009b). A thermo–viscoplastic constitutive model to predict elevated–temperature flow behaviour in a titanium–modified austenitic stainless steel. *Materials Science and Engineering : A*, 526(1-2):1–6. 40, 42, 47
- [Samantaray *et al.*, 2011c] SAMANTARAY, D., MANDAL, S., PHANIRAJ, C. et BHADURI, A. (2011c). Flow behavior and microstructural evolution during hot deformation of AISI Type 316 L(N) austenitic stainless steel. *Materials Science and Engineering : A*, 528(29-30):8565–8572. 44
- [Sattouf *et al.*, 2000] SATTOUF, C., PANTALÉ, O. et CAPERAA, S. (2000). A methodology for the identification of constitutive and contact laws of metallic materials under High Strain Rates. *In Advances in Mechanical Behaviour, Plasticity and Damage*, pages 621–626, Tours, France. Elsevier. 73
- [Saunders *et al.*, 2003] SAUNDERS, N., GUO, U., LI, X., MIODOWNIK, A. et SCHILLÉ, J.-P. (2003). Using jmatpro to model materials properties and behavior. *Jom*, 55(12):60–65. 130
- [Schulz *et al.*, 2023] SCHULZ, B., HAGHDADI, N., LEITNER, T., HAFOK, M. et PRIMIG, S. (2023). Dynamic recrystallisation via nucleation at distorted twins in a ni–based superalloy. *Journal of Alloys and Compounds*, 936:168318. 101
- [Schwer, 2007] SCHWER, L. (2007). Optional strain–rate forms for the johnson cook constitutive model and the role of the parameter epsilon_0. *In 6th European LS_DYNA Users' conference*, pages 1–17. 89
- [Sellars, 1985] SELLARS, C. (1985). The kinetics of softening processes during hot working of austenite. *Czechoslovak Journal of Physics B*, 35(3):239–248. 33
- [Sellars, 1995] SELLARS, C. (1995). Basics of modelling for control of microstructure in thermomechanical controlled processing. *Ironmaking & steelmaking*, 22(6):459–464. 33
- [SHEN –hui *et al.*, 2020] SHEN –HUI, X., YAN, J., GAO, L. et ZHANG, J. (2020). Temperature analysis of specimens for Gleeble thermomechanical simulator. *Heat Treat Met*, 35(9):122–125. 44

- [Shi *et al.*, 2022] SHI, D., ZHANG, F., HE, Z., ZHAN, Z., GAO, W. et LI, Z. (2022). Constitutive equation and dynamic recovery mechanism of high strength cast al-cu-mn alloy during hot deformation. *Materials Today Communications*, 33:104199. 2, 6
- [Simo et Hughes, 1998] SIMO, J. C. et HUGHES, T. J. R. (1998). *Computational Inelasticity*. Interdisciplinary Applied Mathematics. Springer, New York. 86
- [Souaï, 2011] SOUAÏ, N. (2011). *Contrôle du maclage thermique et de la taille de grains par traitements thermomécaniques dans un superalliage base Ni*. Thèse de doctorat, PhD thesis, école nationale supérieure des mines de Paris. 13
- [Sripada *et al.*, 2022] SRIPADA, J., TIAN, Y., CHADHA, K., SAHA, G., JHAZI, M., SPRAY, J. et ARANAS JR, C. (2022). Effect of hot isostatic pressing on microstructural and micromechanical properties of additively manufactured 17-4ph steel. *Materials Characterization*, 192:112174. 1, 7
- [Stoffel *et al.*, 2018] STOFFEL, M., BAMER, F. et MARKERT, B. (2018). Artificial neural networks and intelligent finite elements in non-linear structural mechanics. *Thin-Walled Structures*, 131:102–106. 62
- [Stoffel *et al.*, 2019] STOFFEL, M., BAMER, F. et MARKERT, B. (2019). Neural network based constitutive modeling of nonlinear viscoplastic structural response. *Mechanics Research Communications*, 95: 85–88. 62
- [Swift, 1952] SWIFT, H. (1952). Plastic instability under plane stress. *Journal of the Mechanics and Physics of Solids*, 1(1):1–18. 30
- [Tavakoli *et al.*, 2019] TAVAKOLI, M., MIRZADEH, H. et ZAMANI, M. (2019). Ferrite recrystallisation and intercritical annealing of cold-rolled low alloy medium carbon steel. *Materials Science and Technology*, 35(16):1932–1941. 1, 6
- [Taylor, 1946] TAYLOR, G. I. (1946). The testing of Materials at High Strain Rates of Loading. *Journal of the Institution of Civil Engineers*, 26(8):486–519. 95
- [Tian *et al.*, 2023] TIAN, Y., CHADHA, K. et ARANAS JR, C. (2023). Deformation-induced strengthening mechanism in a newly designed l-40 tool steel manufactured by laser powder bed fusion. *Acta Metallurgica Sinica (English Letters)*, 36(1):21–34. 1, 6
- [Tize Mha *et al.*, 2023] TIZE MHA, P., DHONDAPURE, P., JHAZI, M., TONGNE, A. et PANTALÉ, O. (2023). Interpolation and extrapolation performance measurement of analytical and ann-based flow laws for hot deformation behavior of medium carbon steel. *Metals*, 13(3):633. 49, 83
- [Tize Mha *et al.*, 2022] TIZE MHA, P., TONGNE, A. et PANTALÉ, O. (2022). A generalized non-linear flow law based on modified zerilli-armstrong model and its implementation into abaqus/explicit fem code. *World Journal of Engineering and Technology*, 10(02):334–362. vi, 42, 43, 44, 49, 54, 55
- [Trimble *et al.*, 2020] TRIMBLE, D., AGARWAL, A., MCDONNELL, D., BARRON, S., AHEARNE, E. et O'DONNELL, G. E. (2020). Finite element simulation of orthogonal machining of biomedical grade Co–Cr–Mo alloy. *CIRP Journal of Manufacturing Science and Technology*, 28. 44
- [Vannieuwenhuyze, 2019] VANNIEUWENHUYZE, A. (2019). *Intelligence artificielle vulgarisée : Le Machine Learning et le Deep Learning par la pratique*. éditions ENI. 55
- [Vaswani *et al.*, 2017] VASWANI, A., SHAZEER, N., PARMAR, N., USZKOREIT, J., JONES, L., GOMEZ, A. N., KAISER, Ł. et POLOSUKHIN, I. (2017). Attention is all you need. In GUYON, I., LUXBURG, U. V., BENGIO, S., WALLACH, H., FERGUS, R., VISHWANATHAN, S. et GARNETT, R., éditeurs : *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc. 61
- [Versino *et al.*, 2017] VERSINO, D., TONDA, A. et BRONKHORST, C. A. (2017). Data driven modeling of plastic deformation. *Computer Methods in Applied Mechanics and Engineering*, 318:981–1004. 7
- [Vinh *et al.*, 1980] VINH, T., AFZALI, M. et ROCHE, A. (1980). Mechanical behaviour of materials. kj miller, rf smith. 32
- [Voce, 1948] VOCE, E. (1948). The relationship between stress and strain for homogeneous deformation. *Journal of the Institute of Metals*, 74:537–562. 30

- [Voyiadjis et Abed, 2005] VOYIADJIS, G. Z. et ABED, F. H. (2005). Microstructural based models for bcc and fcc metals with temperature and strain rate dependency. *Mechanics of Materials*, 37(2-3):355–378. 39
- [Walsh et al., 1986] WALSH, D. W., CIESLAK, M. J. et SAVAGE, V. F. (1986). Temperature Measurements in Resistance- Heated Specimens : Longitudinal Gradients. page 9. 128
- [Wang et al., 2021] WANG, T., CHEN, Y., OUYANG, B., ZHOU, X., HU, J. et LE, Q. (2021). Artificial neural network modified constitutive descriptions for hot deformation and kinetic models for dynamic recrystallization of novel aze311 and azx311 alloys. *Materials Science and Engineering : A*, 816: 141259. 114
- [Wang et al., 2023] WANG, X., SHI, X., HUI, Y., CHEN, B., GAN, B. et SHEN, J. (2023). Mechanical behavior and strengthening mechanism of a fine-grained medium carbon steel produced via cyclic oil quenching. *Materials Science and Engineering : A*, 866:144669. 1
- [Xiao et al., 2021] XIAO, H., FAN, X. G., ZHAN, M., LIU, B. C. et ZHANG, Z. Q. (2021). Flow stress correction for hot compression of titanium alloys considering temperature gradient induced heterogeneous deformation. *Journal of Materials Processing Technology*, 288:116868. 3, 125, 128, 131, 132
- [Yoshie et al., 1987] YOSHIE, A., MORIKAWA, H., ONOE, Y. et ITOH, K. (1987). Formulation of static recrystallization of austenite in hot rolling process of steel plate. *Transactions of the Iron and Steel Institute of Japan*, 27(6):425–431. 11
- [Yu et al., 2019] YU, D. J., XU, D. S., WANG, H., ZHAO, Z. B., WEI, G. Z. et YANG, R. (2019). Refining constitutive relation by integration of finite element simulations and Gleeble experiments. *Journal of Materials Science & Technology*, 35(6):1039–1043. 44
- [Yu et al., 2020] YU, H., SHAO, Z. et REN, F. (2020). Flow Stress Modeling using the Material Constitutive Model of Modified Zerilli-Armstrong. *Journal of Physics : Conference Series*, 1676(1):012079. Publisher : IOP Publishing. 42
- [Zaera et Fernández-Sáez, 2006] ZAERA, R. et FERNÁNDEZ-SÁEZ, J. (2006). An implicit consistent algorithm for the integration of thermoviscoplastic constitutive equations in adiabatic conditions and finite deformations. *International journal of solids and structures*, 43(6):1594–1612. 89
- [Zener et Hollomon, 1944] ZENER, C. et HOLLOMON, J. H. (1944). Effect of strain rate upon plastic flow of steel. *Journal of Applied physics*, 15(1):22–32. 33
- [Zeng et al., 2022] ZENG, S., HU, S., PENG, B., HU, K. et XIAO, M. (2022). The constitutive relations and thermal deformation mechanism of nickel aluminum bronze. *Materials & Design*, 220:110853. 2, 44, 46
- [Zerilli et Armstrong, 1995] ZERILLI, F. et ARMSTRONG, R. (1995). Constitutive equation for hcp metals and high strength alloy steels high strain rate effects on polymer, metal and ceramic matrix composites and other advanced materials. ASME. 39
- [Zerilli et Armstrong, 1987] ZERILLI, F. J. et ARMSTRONG, R. W. (1987). Dislocation-mechanics-based constitutive relations for material dynamics calculations. *Journal of applied physics*, 61(5):1816–1825. 38
- [Zhan et al., 2014] ZHAN, H., WANG, G., KENT, D. et DARGUSCH, M. (2014). Constitutive modelling of the flow behaviour of a β titanium alloy at high strain rates and elevated temperatures using the johnson-cook and modified zerilli-armstrong models. *Materials Science and Engineering : A*, 612:71–79. 42, 44
- [Zhang et al., 2010] ZHANG, C., BELLET, M., BOBADILLA, M., SHEN, H. et LIU, B. (2010). A coupled electrical-thermal-mechanical modeling of gleeble tensile tests for ultra-high-strength (uhs) steel at a high temperature. *Metallurgical and Materials Transactions A*, 41:2304–2317. 2, 3, 125, 131
- [Zhao et Gary, 1996] ZHAO, H. et GARY, G. (1996). The testing and behaviour modelling of sheet metals at strain rates from 10^{-4} to 10^4 s $^{-1}$. *Materials Science and Engineering : A*, 207(1):46–50. 32

- [Zhou *et al.*, 2020] ZHOU, Q., JI, C. et ZHU, M.-y. (2020). Research on several constitutive models to predict the flow behaviour of GCr15 continuous casting bloom with heavy reduction. *Materials Research Express*, 6(12). 95
- [Zhu *et al.*, 2022] ZHU, Y., CHEN, Y., HOU, D. et WANG, Z. (2022). Thermal effect on dislocation interactions in magnesium alloy. *Materialia*, 26:101579. 79

Index

- CDRX (Recristallisation Dynamique Continue), 9
- DDRX (Recristallisation Dynamique Discontinue),
10
- GDRX (Recristallisation Dynamique Géométrique),
9
- Gleeble, 5
- Perceptron, 60
- Produit de Hadamard, 69
- Rétropropagation du gradient, 60
- Zener–Hollomon, 33
- Zerilli–Armstrong, 38

