



UNIVERSITE KASDI MERBAH OUARGLA

Faculté des Sciences Appliquées

Département de Génie Mécanique

Mémoire MASTER ACADEMIQUE

Domaine : Sciences Appliquées

Filière : Génie Mécanique

Spécialité : Fabrication mécanique et productique



Présenté par :

Ali Abid

Souigat Mohammed Abd Ennacer

Thème :

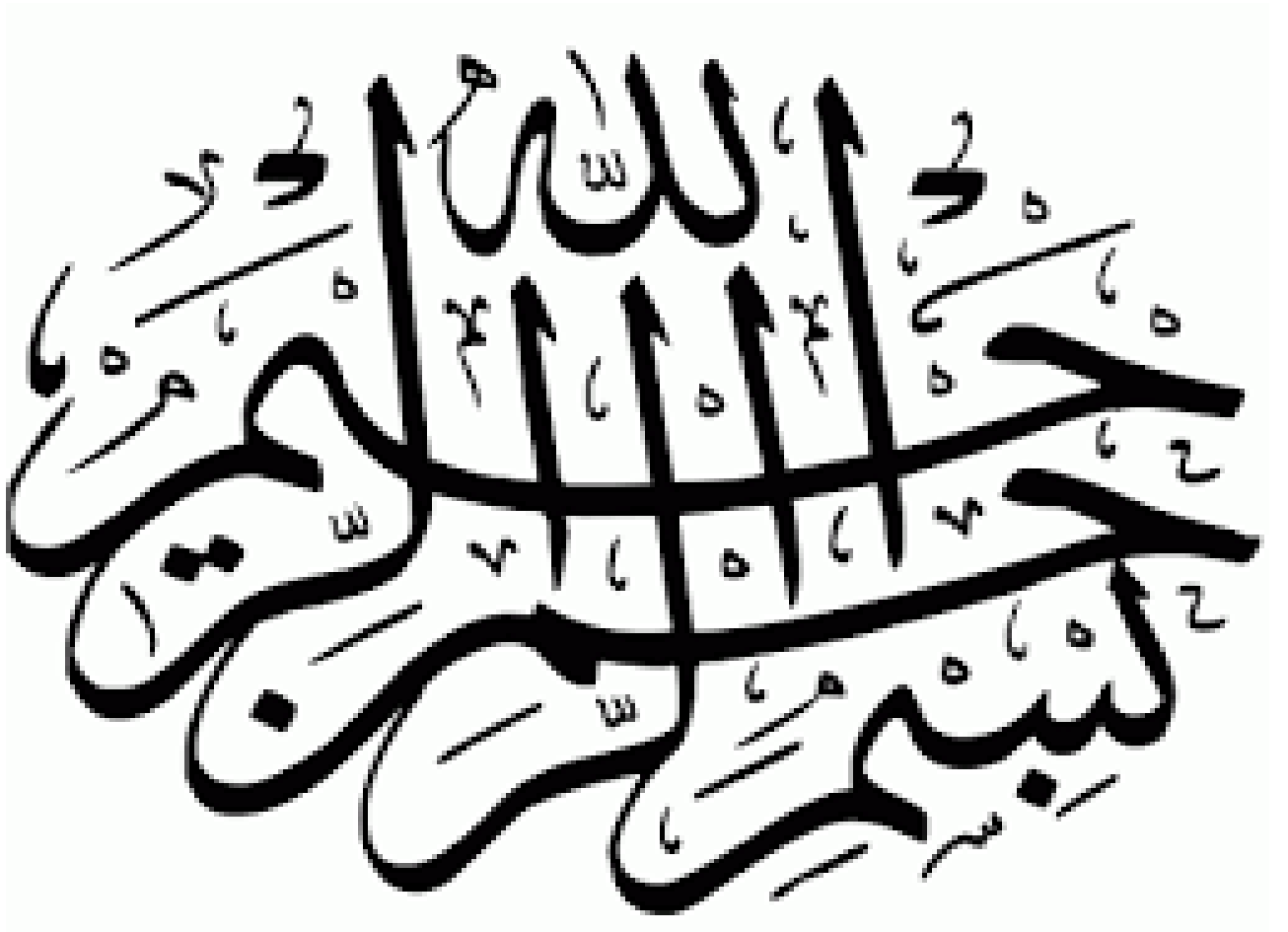
Prédiction des efforts de coupe lors de l'usinage de l'acier 20Mn Cr5 en utilisant les réseaux de neurones artificiels

Soutenu le : 14/06/2021

Devant le jury :

Dr. Bentaleb Fayçal	MC (B)	Encadreur	UKM Ouargla
Dr. Rezgui Imane	MC(A)	Présidente	UKM Ouargla
Dr. Ameer Toufik	MC(A)	Examineur	UKM Ouargla

Année universitaire : 2020 / 2021



Remerciement

Parvenus au terme de ce travail, nous remercions Dieu de nous avoir donné la santé, la patience et les moyens, afin que nous terminions ce modeste travail.

Ce présent mémoire de fin d'étude, n'aurait pu avoir le jour sans la contribution de nombreuses personnes, dont nous faisons aujourd'hui un plaisir et un devoir de les remercier :

Nous tenons à exprimer en premier lieu nos sincères remerciements à notre directeur de recherche Dr. Bentaleb Fayçal de nous avoir consacré son temps et son expérience, et surtout par ses conseils et son aide, durant la réalisation de ce travail. Nos vives gratitude et nos sincères remerciements s'adressent également à Madame et Monsieur les membres du jury : Dr. Rezgui Imane pour l'honneur qu'elle nous a fait, en acceptant de présider notre jury de soutenance, Dr. Ameer Toufik de nous avoir fait l'honneur d'examiner ce travail.

Enfin, nous nous saurions exprimer notre profonde gratitude à toutes les personnes, qui ont contribué de près ou de loin à la réalisation de ce modeste travail.

ABID Ali

Souigat Mohammed AbdEnnacer

Dédicaces

Je dédis ce modeste mémoire :

A ma mère, qui m'a encouragé d'aller en avant et
qui m'a donnée tout son amour pour reprendre mes
études ;

A mon père qui m'a donné son soutien permanent
durant toutes mes années d'études ;

A mes frères et sœurs ;

A mon binôme Souigat Mohammed AbdEnnacer
et tous mes amis sans exception ;

abid ali

Dédicaces

Je dédis ce modeste mémoire :

A ma mère, qui m'a encouragé d'aller en avant et
qui m'a donnée tout son amour pour reprendre mes
études ;

A mon père qui m'a donné son soutien permanent
durant toutes mes années d'études ;

A mes frères et sœurs ;

A mon binôme abid ali et tous mes amis sans
exception ;

Souigat Mohammed AbdEnnacer

Table des matières

Introduction Général	1
CHAPITRE I: Généralités sur le tournage	
I.1 Introduction	3
I.2 Les tours	3
I.2.1 Tournage	4
I.3 Le principe de fonctionnement	5
I.4 Différentes opérations de tournage	5
I.4.1 Chariotage	5
I.4.2 Dressage	5
I.4.3 Alésage	6
I.4.4 Perçage	6
I.4.5 Rainurage	7
I.4.6 Chanfreinage	7
I.4.7 Tronçonnage	8
I.4.8 Filetage	8
I.5 Conditions de coupe	9
I.6 Grandeurs géométriques et cinématiques en usinage	9
I.6.1 Profondeur de passe a_p (mm)	9
I.6.2 Vitesse d'avance V_f (mm/min), Avance par tour f (mm/tr)	10
I.6.3 Vitesse de coupe V_c (m/min)	10
I.7 Les efforts de la coupe	11
I.7.1 Influence des différents facteurs sur les composantes de l'effort de coupe	14
I.8 Propriétés du métal	16
CHAPITRE II : Modélisation par réseaux de neurones	
II.1 Introduction	18
II.2 Les réseaux de neurones	18
II.2.1 Les réseaux de neurones formels	19
II.2.2 Architecture des réseaux de neurones	21
II.2.2.1 Les réseaux de neurones non bouclés	21

II.2.2.2 Les réseaux de neurones bouclés	23
II.2.3 L'apprentissage des réseaux de neurones	24
II.2.3.1 Apprentissage supervisé	24
II.2.3.2 Apprentissage non supervisé	25
II.2.3.3 Apprentissage hybride	25
II.2.4 Les différents types de réseaux de neurones	25
II.2.4.1 Les réseaux 'feed-forward'	25
II.2.4.1.1 Le perceptron	25
II.2.4.1.2 Le perceptron multicouche (PMC)	25
II.2.4.1.3 Les réseaux à fonction radiale	26
II.2.4.2 Les réseaux 'feedback'	26
II.2.4.2.1 Les cartes auto-organisatrices de Kohonen	26
II.2.4.2.2 Les réseaux de Hopfield	26
II.2.4.2.3 Les réseaux ART	26
II.2.5 Propriété fondamentale des réseaux de neurones : l'approximation parcimonieuse	27
II.2.6 Réseaux de neurones et régression non linéaire	28
II.2.7 Mise en œuvre des réseaux de neurones et domaines d'application	30
CHAPITRE III : Résultats et discussion	
III.1 Introduction	35
III.2 Méthodologie de la modélisation par les réseaux de neurones artificiels (ANN)	36
III.2.1 Analyse d'un ANN pour l'effort Fa	37
III.2.1.1 Utilisation de 3 couches cachées	37
III.2.1.2 Utilisation de 5 couches cachées	40
III.2.1.3 Utilisation de 8 couches cachées	42
III.2.1.4 Comparaison les résultats obtenus pour Fa	45
III.2.2 Analyse d'un ANN pour l'effort Fc	47
III.2.2.1 Utilisation de 3 couches cachées	47
III.2.2.2 Utilisation de 5 couches cachées	50
III.2.2.3 Utilisation de 8 couches cachées	52

III.2.2.4 Comparaison les résultats obtenus pour F_c	55
III.2.3 Analyse d'un ANN pour l'effort F_p	57
III.2.3.1 Utilisation de 3 couches cachées	57
III.2.3.2 Utilisation de 5 couches cachées	60
III.2.3.3 Utilisation de 8 couches cachées	62
III.2.3.4 Comparaison les résultats obtenus pour F_p	65
Conclusion Général	68

Liste des figures

CHAPITRE I :	
Figure I.1 : Schéma d'un Tour	4
Figure I.2: Tournage avec outil à arête unique	4
Figure I.3: Chariotage	5
Figure I.4: Dressage	6
Figure I.5: Alésage	6
Figure I.6: Perçage	6
Figure I.7: Rainurage	7
Figure I.8: Chanfreinage	7
Figure I.9: Tronçonnage	8
Figure I.10: Filetage	8
Figure I.11: Réglage des conditions de coupe	9
Figure I.12: Paramètre de l'opération de tournage	10
Figure I.13: Les efforts de coupe en tournage	12
Figure. I.14: Influence de la vitesse de coupe sur l'effort de coupe	15
CHAPITRE II :	
Figure II.1: Représentation graphique d'un neurone formel	20
Figure II.2: Fonctions d'activation : (a) à seuil, (b) linéaire et (c) sigmoïde	21
Figure II.3: Réseau de neurones non bouclé « Perceptron multicouche »	22
Figure II.4: Réseau de neurones bouclé	23
Figure II.5: Influence du nombre de paramètres sur la qualité de l'approximation	31
CHAPITRE III :	
Figure III.1: Insertion des entrées dans le tableau	36
Figure III.2: Insertion des valeurs des critères de les efforts (a) F_a et (b) F_c et (c) F_p	37
Figure III.3: Diagramme optimale pour la modélisation de $F_{a_{ANN3}}$	37
Figure III.4: Le processus de formation dans le réseau neuronal pour $F_{a_{ANN3}}$	38
Figure III.5: Courbe de test de performance du système pour $F_{a_{ANN3}}$	39
Figure III.6: Régression d'entraînement des réseaux de neurones pour $F_{a_{ANN3}}$	39

Figure III.7: Diagramme optimale pour la modélisation de $F_{a_{ANN5}}$	40
Figure III.8: Le processus de formation dans le réseau neuronal pour $F_{a_{ANN5}}$	41
Figure III.9: Courbe de test de performance du système pour $F_{a_{ANN5}}$	41
Figure III.10: Régression d'entraînement des réseaux de neurones pour $F_{a_{ANN5}}$	42
Figure III.11: Diagramme optimale pour la modélisation de $F_{a_{ANN8}}$	42
Figure III.12: Le processus de formation dans le réseau neuronal pour $F_{a_{ANN8}}$	43
Figure III.13: Courbe de test de performance du système pour $F_{a_{ANN8}}$	44
Figure III.14: Régression d'entraînement des réseaux de neurones pour $F_{a_{ANN8}}$	44
Figure III.15: Comparaison entre les $F_{a_{exp}}$ et F_a simulées	46
Figure III.16: Courbe de comparaison entre les erreurs de F_c simulées	47
Figure III.17: Diagrammes optimum pour la modélisation de $F_{c_{ANN3}}$	47
Figure III.18: Le processus de formation dans le réseau neuronal pour $F_{c_{ANN3}}$	48
Figure III.19: Courbe de test des performances du système pour $F_{c_{ANN3}}$	49
Figure III.20: Régression d'entraînement des réseaux de neurones pour $F_{c_{ANN3}}$	49
Figure III.21: Diagrammes optimum pour la modélisation de $F_{c_{ANN5}}$	50
Figure III.22: Le processus de formation dans le réseau neuronal pour $F_{c_{ANN5}}$	51
Figure III.23: Courbe de test des performances du système pour $F_{c_{ANN5}}$	51
Figure III.24: Régression d'entraînement des réseaux de neurones pour $F_{c_{ANN5}}$	52
Figure III.25: Diagrammes optimum pour la modélisation de $F_{c_{ANN8}}$	52
Figure III.26: Le processus de formation dans le réseau neuronal pour $F_{c_{ANN8}}$	53
Figure III.27: Courbe de test des performances du système pour $F_{c_{ANN8}}$	54
Figure III.28: Régression d'entraînement des réseaux de neurones pour $F_{c_{ANN8}}$	54
Figure III.29: Comparaison entre les $F_{c_{exp}}$ et F_c simulées	56
Figure III.30: Courbe de comparaison entre les erreurs de F_c simulées	57
Figure III.31: Diagramme optimale pour la modélisation de $F_{p_{ANN3}}$	57
Figure III.32: Le processus de formation dans le réseau neuronal pour $F_{p_{ANN3}}$	58
Figure III.33: Courbe de test des performances du système pour $F_{p_{ANN3}}$	59
Figure III.34: Régression d'entraînement des réseaux de neurones pour $F_{p_{ANN3}}$	59
Figure III.35: Diagramme optimale pour la modélisation de $F_{p_{ANN5}}$	60
Figure III.36: Le processus de formation dans le réseau neuronal pour $F_{p_{ANN5}}$	61

Figure III.37: Courbe de test des performances du système pour Fp_{ANN5}	61
Figure III.38: Régression d'entraînement des réseaux de neurones pour Fp_{ANN5}	62
Figure III.39: Diagramme optimale pour la modélisation de Fp_{ANN8}	62
Figure III.40: Le processus de formation dans le réseau neuronal pour Fp_{ANN8}	63
Figure III.41: Courbe de test des performances du système pour Fp_{ANN8}	64
Figure III.42: Régression d'entraînement des réseaux de neurones pour Fp_{ANN8}	64
Figure III.43: Comparaison entre les Fp_{exp} et Fp simulées	66
Figure III.44: Courbe de comparaison entre les erreurs de Fp simulées	67

Listes des tableaux

CHAPITRE I :	
Tableau I.1: Effort tangentiel d'avancement F_p et effort de poussée F_a en fonction des paramètres du effort tangentiel de coupe F_c	14
CHAPITRE II :	
Tableau II.1: Réseaux de neurones et statistiques	30
CHAPITRE III :	
Tableau III.1: Résultats des efforts de la coupe (F_a , F_c et F_p) en fonction des paramètres du régime de coupe	35
Tableau III.2: Performance des structures ANN du modèle de F_a avec 3 couches cachées	37
Tableau III.3: Performance des structures ANN du modèle de F_a avec 5 couches cachées	40
Tableau III.4: Performance des structures ANN du modèle de F_a avec 3 couches cachées	43
Tableau III.5: Comparaison entre les résultats expérimentaux et simulés	45
Tableau III.6: Performance des structures ANN du modèle de F_c avec 3 couches cachées	48
Tableau III.7: Performance des structures ANN du modèle de F_c avec 5 couches cachées	50
Tableau III.8: Performance des structures ANN du modèle de F_c avec 8 couches cachées	53
Tableau III.9: Comparaison entre les résultats expérimentaux et simulés	55
Tableau III.10: Performance des structures ANN du modèle de F_c avec 3 couches cachées	58
Tableau III.11: Performance des structures ANN du modèle de F_c avec 3 couches cachées	60
Tableau III.12: Performance des structures ANN du modèle de F_c avec 3 couches cachées	63
Tableau III.13: Comparaison entre les résultats expérimentaux et simulés	65

Nomenclature

Symbole	Unité	Désignation
V_c	m/min	Vitesse de coupe
V_f	mm/min	La vitesse d'avance
ω	rad / s	Vitesse angulaire
N	<i>tr / min</i>	Vitesse de rotation
D	<i>mm</i>	Diamètre
f	<i>mm / tr</i>	Avance par tour
ap	<i>mm</i>	Profondeur de passe
F_c	N	Effort tangentiel de coupe
F_p	N	Effort tangentiel d'avancement
F_a	N	Effort de poussée
K_c	Da.N/mm ²	Coefficient spécifique de coupe
$F_{a_{exp}}$	N	Effort de poussée expérimentale
$F_{a_{ANN3}}$	N	Effort de poussée simulée avec 3 couches cachées
$F_{a_{ANN5}}$	N	Effort de poussée simulée avec 5 couches cachées
$F_{a_{ANN8}}$	N	Effort de poussée simulée avec 8 couches cachées
$EF_{a_{ANN3}}$	N	Erreur d'effort de poussée simulée avec 3 couches cachées
$EF_{a_{ANN5}}$	N	Erreur d'effort de poussée simulée avec 5 couches cachées
$EF_{a_{ANN8}}$	N	Erreur d'effort de poussée simulée avec 8 couches cachées

Introduction général :

Le tournage est l'un des procédés de fabrication les plus utilisés dans l'industrie mécanique, ainsi, le choix des paramètres de coupe le plus parfait (vitesse de coupe, avance et profondeur de coupe) est très important pour assurer un meilleur état de surface des pièces usinées et la durée de vie des outils de coupe [1].

Le but de cette étude est de prédire la force de coupe générée par les opérations de coupe à l'aide de réseaux de neurones artificiels (ANN). Pour que nous fassions le processus sur différentes couches cachées 3, 5 et 8, et choisissons la meilleure d'entre elles ?

Le travail réalisé dans cette mémoire est organisé comme suit :

- **Le premier chapitre :** Une recherche bibliographique, tout d'abord, un aperçu sur le tournage et ces différentes opérations, les conditions de coupe puis nous parlons des efforts de coupe.
- **Le deuxième chapitre :** Nous avons présenté la méthode des réseaux de neurones ainsi que les processus d'examen et de vérification.
- **Le troisième chapitre :** Nous discutons des résultats obtenus par la méthode des réseaux de neurones.

Enfin, nous terminons nos travaux par une conclusion générale.

CHAPITRE I :
Généralités sur le
tournage

I.1 Introduction :

On appelle usinage toute opération de mise en forme par enlèvement de matière à l'aide d'une machine-outil destinée à conférer à une pièce des dimensions et un état de surface situé dans un intervalle de tolérance donné. D'un point de vue économique, le secteur industriel de l'usinage a une importance non négligeable puisqu'il produit environ 2.5% du produit national brut d'un pays développé [1].

L'usinage consiste en toute opération de mise en forme par enlèvement de matière dans le but d'aboutir à une forme voulue d'une pièce brute tout en respectant les dimensions et l'état de surface prédéfinis dans la marge de tolérance donnée dans le dessin de définition. En industrie, les procédés d'usinage par enlèvement de matière des pièces métalliques à l'aide des machines-outils sont considérés comme les procédés de mise en forme les plus performants et les plus précis. Un mouvement de coupe et un mouvement d'avance sont combinés pour générer les formes escomptées. La surface voulue est générée grâce à un travail de forme du à la forme de l'outil et un travail d'enveloppe causé par la trace de l'arête de l'outil [2].

I.2 Les tours :

Les tours (figure I.1) permettent de réaliser des surfaces hélicoïdales (filetage) et des surfaces de révolution : cylindres, cônes et plans (génératrice perpendiculaire à l'axe de révolution). L'utilisation principale de ces machines est l'usinage des arbres. La pièce, généralement tenue par le mandrin, a un mouvement de rotation (mouvement de coupe) transmis par la broche.

L'outil peut se déplacer en translation suivant deux directions. Ces deux directions, perpendiculaires entre elles, appartiennent à un plan auquel l'axe de la broche est parallèle.

Le premier mouvement de translation est parallèle à l'axe de la broche. Le deuxième mouvement de translation est perpendiculaire à l'axe de la broche [3].

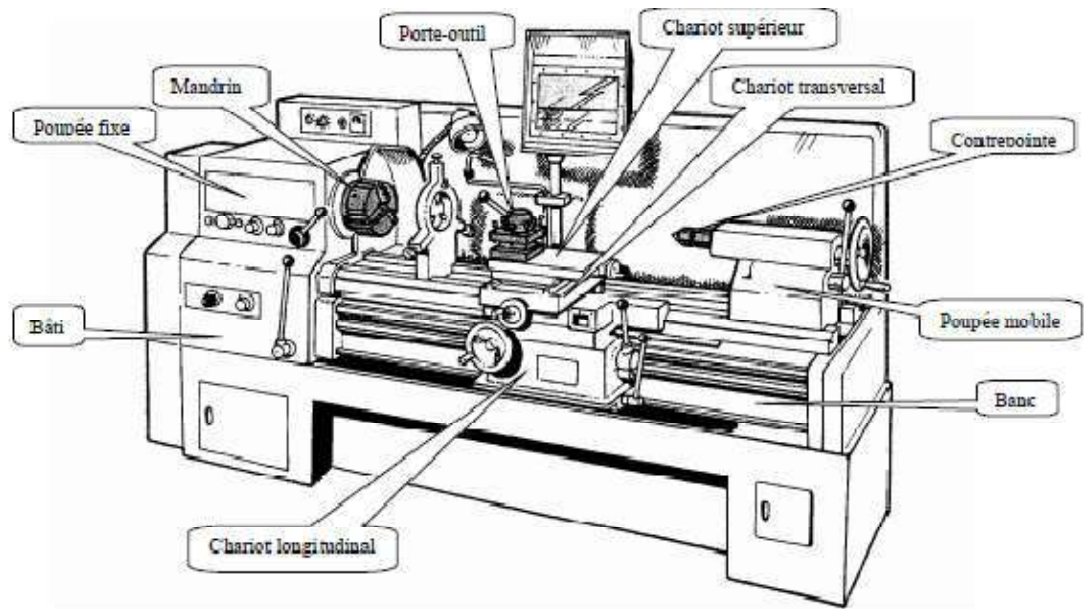


Figure I.1: Schéma d'un Tour [3].

I.2.1 Tournage :

Le tournage est un procédé de fabrication mécanique par coupe (enlèvement de matière) mettant en jeu des outils à arête unique.

La pièce est animée d'un mouvement de rotation (mouvement de coupe), qui est le mouvement principal du procédé (figure I.2) [3].

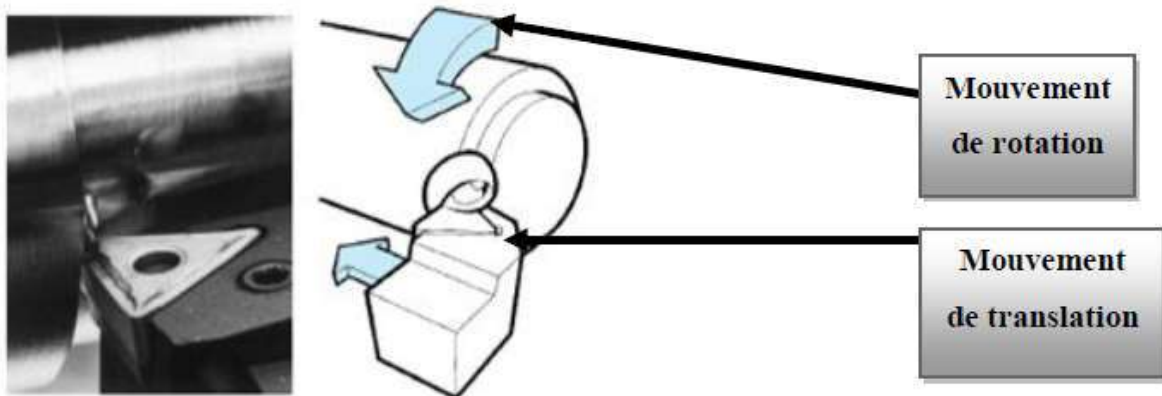


Figure I.2: Tournage avec outil à arête unique [3].

I.3 Le principe de fonctionnement :

La pièce à usiner est placée dans le mandrin et serrée par l'intermédiaire des mors. Un moteur permet la mise en rotation du mandrin fixé sur la broche. L'outil coupant est positionné dans un porte-outil et serré à l'aide de vis. Le porte-outil est mis en place sur une tourelle porte-outils. Le chariot transversal et le longitudinal assurent ainsi les mouvements de l'outil par des moteurs d'avance. La contrepointe permet de réaliser des opérations de pointage, centrage, perçage et alésage en bout de pièce [4].

I.4 Différentes opérations de tournage :

I.4.1 Chariotage :

C'est une opération qui consiste à usiner une surface cylindrique ou conique extérieure (Figure I.3)

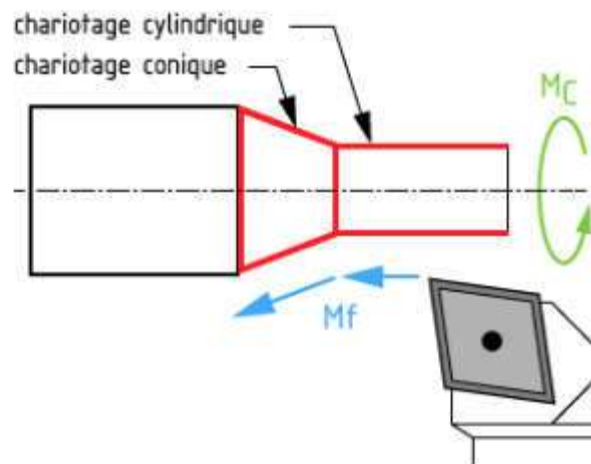


Figure I.3: Chariotage [4].

I.4.2 Dressage :

C'est une opération qui consiste à usiner une surface plane perpendiculaire à l'axe de la broche extérieure ou intérieure (Figure I.4).

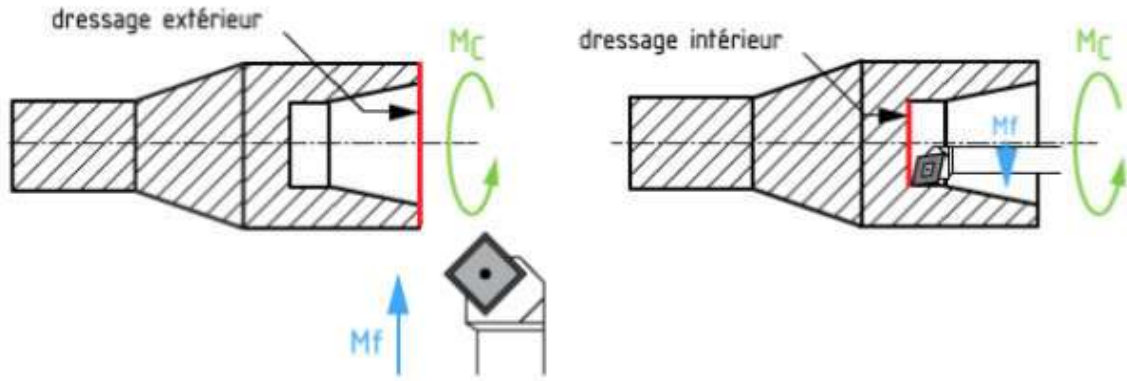


Figure I.4: Dressage [4].

I.4.3 Alésage :

C'est une opération qui consiste à usiner une surface cylindrique ou conique intérieure (Figure I.5)

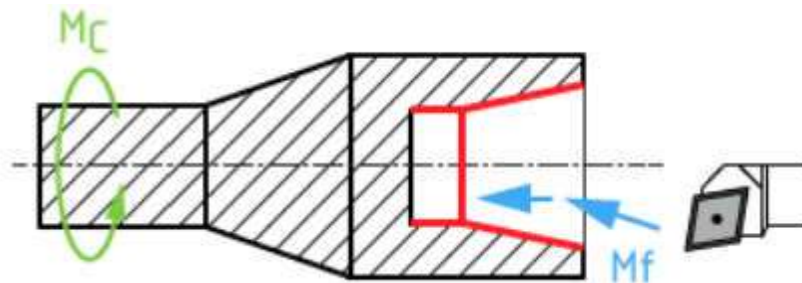


Figure I.5: Alésage [4].

I.4.4 Perçage :

C'est une opération qui consiste à usiner un trou à l'aide d'une forêt (Figure I.6).

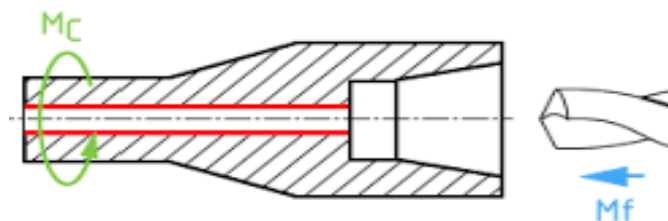


Figure I.6: Perçage [4].

I.4.5 Rainurage :

C'est une opération qui consiste à usiner une rainure intérieure ou extérieure pour le logement d'un circlips ou d'un joint torique par exemple (Figure I.7).

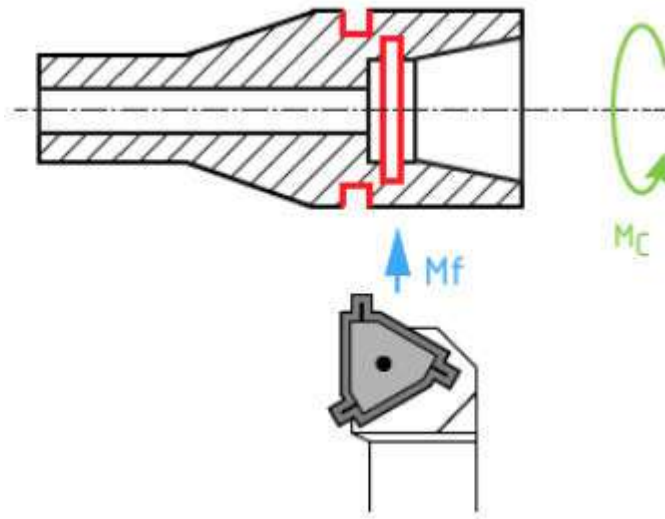


Figure I.7: Rainurage [4].

I.4.6 Chanfreinage :

C'est une opération qui consiste à usiner un cône de petite dimension de façon à supprimer un angle Vif (Figure I.8).

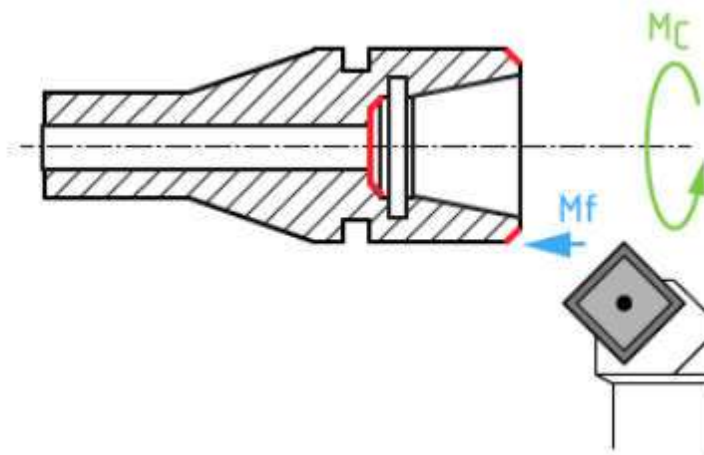


Figure I.8: Chanfreinage [4].

I.4.7 Tronçonnage :

C'est une opération qui consiste à usiner une rainure jusqu'à l'axe de la pièce afin d'en détacher un tronçon (Figure I.9).

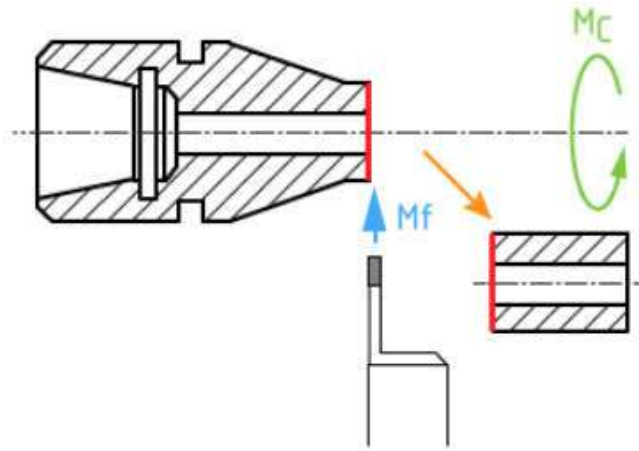


Figure I.9: Tronçonnage [4].

I.4.8 Filetage :

C'est une opération qui consiste à réaliser un filetage extérieur ou intérieur (Figure I.10).

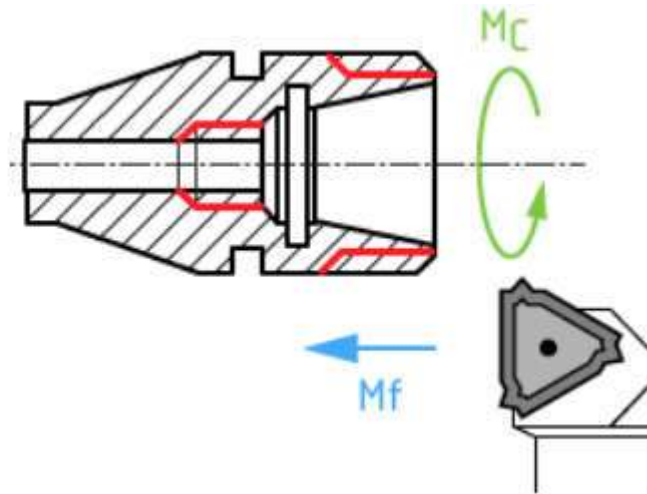


Figure I.10: Filetage [4].

Les mouvements relatifs outil-pièce en tournage sont :

- Le mouvement de coupe M_c ;
- Le mouvement d'avance M_f ;
- Le mouvement de position M_p .

I.5 Conditions de coupe :

Avant de réaliser une opération de tournage, il est nécessaire de bien choisir les conditions de coupe pour obtenir un bon résultat (précision, état de surface ...) [4].

Il y a plusieurs critères qui permettent de définir les conditions de coupe notamment :

- Le type de la machine (mécanisme, gamme des vitesses,...) ;
- La puissance de la machine ;
- La matière de l'outil (carbure...) ;
- La matière usinée (acier, aluminium...) ;
- Le type de l'opération (perçage, chariotage, surfaçage...) ;
- L'utilisation éventuelle de lubrification (destiné à refroidir ou/et à diminuer le frottement) [4].

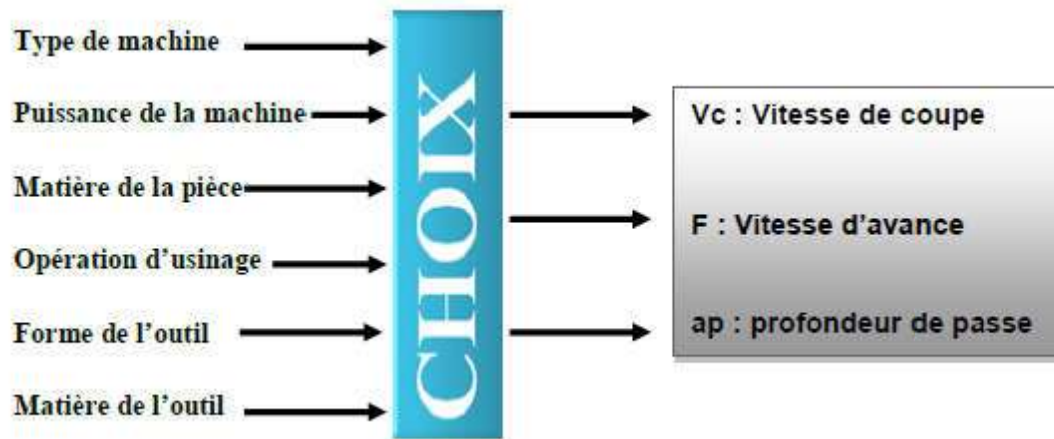


Figure I.11: Réglage des conditions de coupe [4].

I.6 Grandeurs géométriques et cinématiques en usinage :

Les grandeurs géométriques et cinématiques conditionnent les phénomènes physiques lors de la coupe [5].

I.6.1 Profondeur de passe a_p (mm) :

La profondeur de passe notée a_p , correspond à la longueur de l'arête de coupe engagée dans la matière, dans le cas de la coupe orthogonale, et à la différence entre le rayon de la pièce avant et après usinage, dans le cas du tournage.

I.6.2 Vitesse d'avance V_f (mm/min), Avance par tour f (mm/tr) :

La vitesse d'avance V_f (mm/min), est la vitesse à laquelle la machine déplace l'outil par rapport au bâti. L'avance par tour f (mm/tr) est la valeur du déplacement de l'outil, lorsque la pièce a effectué une révolution. C'est une donnée clé pour la qualité de la surface usinée.

L'avance influe non seulement sur l'épaisseur des copeaux, mais également sur la manière dont ils se brisent [5].

$$V_f = f \cdot N \quad (\text{I.1})$$

- N = fréquence de rotation réglée sur la machine en tr/min.
- f = Avance en mm/tour.

I.6.3 Vitesse de coupe V_c (m/min) :

Dépend de la configuration d'usinage, du procédé et du couple outil/matière. La définition de la vitesse de coupe est un paramètre essentiel car elle détermine la vitesse de rotation de la pièce [5].

Vitesse de rotation de la pièce N (nombre de tours), est définie par :

$$V_c = \frac{\pi D N}{1000} \quad (\text{I.2})$$

Avec :

- D : diamètre de la pièce exprimée en mm ;
- V_c : vitesse de coupe (m/min) ;
- N : vitesse de rotation (tr/min) ;

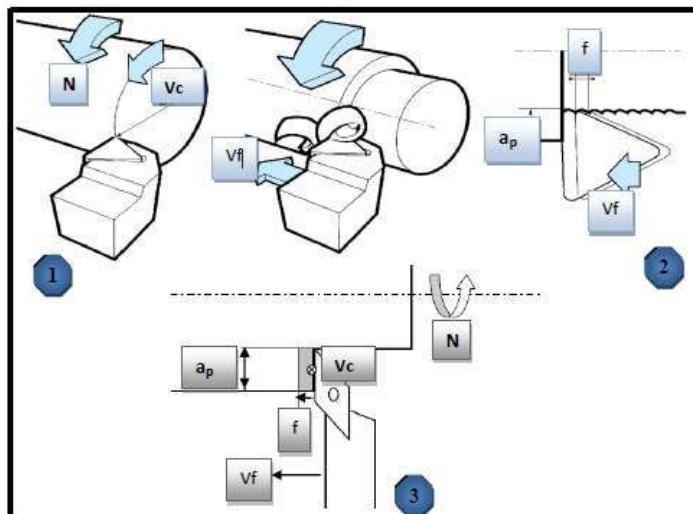


Figure I.12: Paramètre de l'opération de tournage [5].

Il convient d'observer que la vitesse de coupe n'est constante que si la vitesse de broche et le diamètre de la pièce demeurent inchangés. En dressage, par exemple où l'outil se déplace en direction du centre, la vitesse de coupe varie continuellement si la rotation de la pièce s'effectue à une vitesse de broche constante. Or, pour une productivité maximale et une meilleure qualité des surfaces obtenues, il est souhaitable de maintenir la vitesse de coupe constante. Sur un grand nombre de tours modernes, la vitesse de broche augmente au fur et à mesure que l'outil approche de l'axe, afin de compenser ainsi la diminution de diamètre. Mais dans le cas de très petits diamètres, cette compensation se révèle impossible du fait de la plage de vitesse limitée qu'autorisent les machines. De même, lorsqu'une pièce, comme cela est souvent le cas, présente des diamètres différents ou est de forme conique ou courbe, la fréquence de rotation doit être corrigée en fonction du diamètre, pour maintenir la vitesse de coupe constante. Le préparateur choisit cette vitesse de coupe en fonction, notamment [6].

- Des matériaux de la pièce et de l'outil ;
- De la lubrification ;
- De la durée de vie souhaitée pour l'outil.

I.7 Les efforts de la coupe :

Les efforts de coupe sont à l'origine des déformations plastiques et donc des élévations de température qui se produisent au cours de la coupe. Pour toutes autres conditions de coupe égales, l'augmentation de la vitesse de coupe ne s'accompagne pas d'une variation notable des efforts de coupe (10% de baisse environ), alors que l'augmentation de la vitesse d'avance entraîne une augmentation de la valeur de l'effort tangentiel (relation linéaire : l'effort double environ quand la vitesse d'avance double) [7].

L'évaluation des efforts de coupe permet : [8]

- De dimensionner les outils et les porte-pièce ;
- De déterminer les appuis du montage, en opposition à ces efforts ;
- D'évaluer la puissance de coupe afin de choisir la machine-outil.

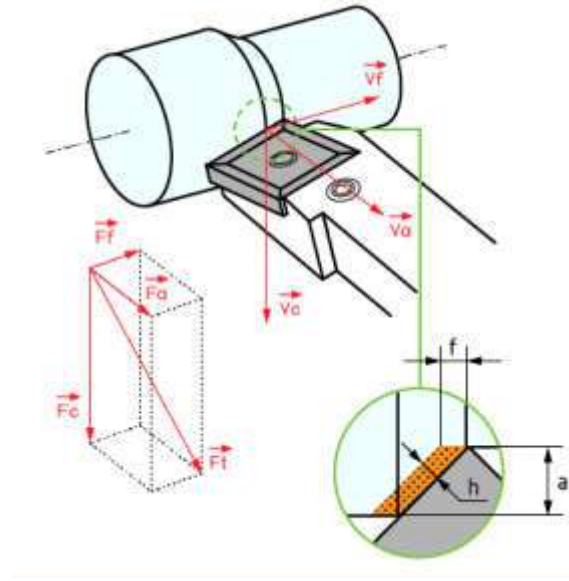


Figure I.13: Les efforts de coupe en tournage [8].

D'une manière simpliste, l'effort total est formulé donc par l'équation de résultante suivante:

$$F_c = \sqrt{F_t^2 + F_f^2 + F_r^2} \quad (I.3)$$

- F_c : l'effort tangentiel de coupe ;
- $F_f = F_p$: l'effort axial ;
- F_a : l'effort radial ;
- F_t : l'effort total .

L'effort de coupe F_c appelé effort « générateur de puissance » est décisif, tout comme la vitesse de coupe, pour le calcul de la puissance de coupe ou d'entraînement de la machine outil. L'importance de cet effort dépend essentiellement de la matière à usiner et des conditions de coupe actives.

$$F_c = K_c \cdot a_p \cdot f \quad (I.4)$$

- K_c : coefficient spécifique de coupe en [da N/mm^2].
- a_p : profondeur de passe en [mm].
- f : avance en [mm/tr].

Dans le cas général, la puissance nécessaire à la coupe est le produit scalaire entre le vecteur d'effort et vitesse :

$$P = \vec{F} \cdot \vec{V} \quad (I.5)$$

D'où

$$P = F_a \cdot V_a + F_f \cdot V_f + F_c \cdot V_c \quad (I.6)$$

En chariotage nous négligeons l'effort d'avancement et l'effort radial, en effet la puissance nécessaire à la coupe sera formulée par:

$$P_c = F_c \cdot V_c \quad (I.7)$$

Ainsi, on trouve que l'effort de coupe s'évolue inversement à la vitesse de coupe. On dit que la vitesse enlevées devisent la coupe.

L'augmentation de la vitesse conduit à une diminution des composantes de l'effort de coupe. À cause de la haute température dans la zone de coupe qui rend le métal travaillé plus plastique et par conséquent l'effort nécessaire pour la coupe diminue. Il est à noter aussi que les vitesses de coupe enlevée ne favorisent pas l'apparition de l'arête rapportée.

Les investigations antérieures montrent une nette augmentation des efforts de coupe avec l'augmentation de la profondeur de passe et cela à cause de l'augmentation de l'épaisseur (section) du copeau et par conséquent l'augmentation du volume du métal déformé. Cette augmentation est presque linéaire.

L'effort tangentiel et axial dépasse l'effort radial. Cela peut être expliqué par le travail de l'outil en tournage par son rayon du bec aux petites profondeurs de passe. Avec l'augmentation de a_p , la coupe se fait en dehors de la limite du rayon de bec et la pièce présente une résistance à la pénétration de l'outil dans le sens de l'effort tangentiel et en particulier axial, car la longueur de l'arête tranchante en contact avec la pièce augmente. Autrement dit l'outil ne travaille plus par son rayon seulement.

Avec l'augmentation des ces deux facteur, la section droite du copeau s'accroît, de même que le volume du métal déformé, il en résulte que le métal résiste plus à la formation du copeau et la valeur des efforts nécessaire pour assurer la coupe est plus importante.

En chariotage, la profondeur de passe intervient d'une manière plus accentuée sur les efforts de coupe que l'avance. Parcellements, L'augmentation de la vitesse d'avance entraîne une augmentation de la valeur de l'effort tangentiel

Tableau I.1: Effort tangentiel d'avancement F_p et effort de poussée F_a en fonction des paramètres du effort tangentiel de coupe F_c [8].

Type d'outil	Matériau usiné	F_a	F_p
Outil à charioter coudé à 45°	Acier	$2/5 F_c$	$2/5 F_c$
	Fonte	$1/3 F_c$	$1/3 F_c$
Outil couteau	Acier	$2/3 F_c$	$1/10 F_c$
	Fonte	$2/3 F_c$	$1/10 F_c$

I.7.1 Influence des différents facteurs sur les composantes de l'effort de coupe :

a) Les propriétés mécaniques du matériau à usiner :

La valeur des composantes F_a , F_c et F_p est proportionnelle à la charge rupture à la traction R_t et à la dureté HB du métal à usiner.

b) Influence de la profondeur de passe et l'avance :

Avec l'augmentation des ces deux facteur, la section du copeau s'accroît, de même que le volume du métal déformé, il en résulte que le métal résiste plus à la formation du copeau et la valeur des composantes F_a , F_c et F_p nécessaire pour assurer la coupe est plus importante. En chariotage, la profondeur de passe intervient d'une manière plus accentuée sur les efforts de coupe que l'avance.

c) Influence de la vitesse de coupe :

On peut distinguer 03 zones (Figure. I.14)

- **Zone A** : diminution de F avec les vitesses bases due à une diminution de frottement copeau - outil.

- **Zone B** : l'apparition de l'arête rapportée provoque une augmentation des frottements et par suite de l'effort tangentiel de coupe et diminution.
- **Zone C** : La V_C augmente arête rapportée diminue. L'effort tangentiel de coupe F_t diminue et se stabilise vers 200 m/min.

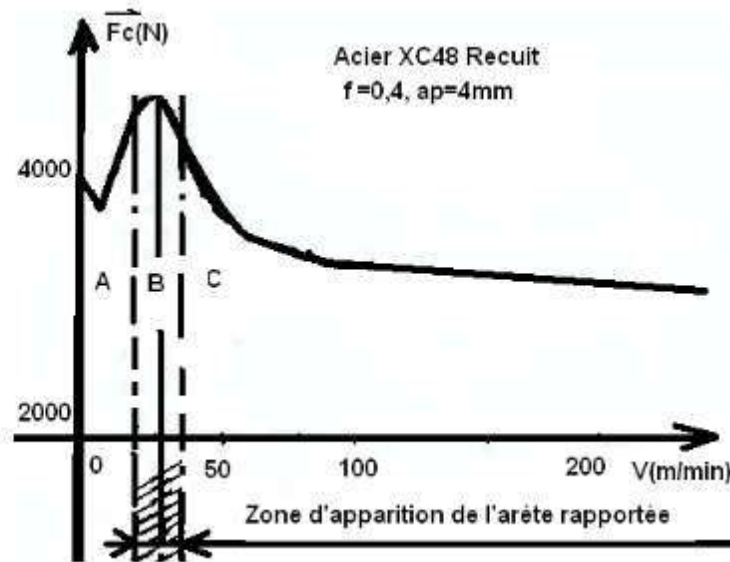


Figure. I.14: Influence de la vitesse de coupe sur l'effort de coupe [8].

d) Influence de l'angle d'attaque :

Il s'agit de l'angle entre l'arête de coupe principale des plaquettes et la surface de la pièce. L'épaisseur des copeaux, les forces de coupe et la durée de vie d'outil sont influencées par l'angle d'attaque. Lorsque l'angle d'attaque est négatif, les composantes tangentielle et radiale de l'effort de coupe sont importantes.

e) Influence de l'angle de direction principale :

Avec une avance inchangée, le copeau devient plus fin au fur et à mesure que l'angle diminue. Ceci conduit à une augmentation de l'effort de coupe, la réduction de l'angle de direction est limitée par les dégradations croissantes de la stabilité (forte augmentation des efforts d'avances et de pénétration tendance au broutage).

f) Influence de matériau de coupe :

Les matériaux de coupe revêtus, en particulier ceux avec des revêtements TiN ou Ti (C, N), accusent un frottement moindre et des efforts de coupe plus faible que les matériaux de coupe

non revêtus. Les faces d'attaque superfinies (rectifiées, rodées, polies) et dotées d'une arête de coupe dure et tranchante, réduisent les efforts de coupe [8].

I.8 Propriétés du métal :

Acier faiblement mélangé : 20 MnCr5

- **20** : 0.20% Pourcentage de Carbone ;
- **Mn** : Manganèse ;
- **Cr** : Chrome ;
- **5** : 1.25% Pourcentage le premier élément du Manganèse.

Remarque importante :

On l'appelle acier faiblement mélangé car le pourcentage de n'importe quel élément ajouté ne dépasse pas 5%

- Le pourcentage d'éléments ajoutés est enregistré double quatre fois par rapport des éléments suivants (Cu, Co, Mn, Ni, Si, Cr) ;
- Le pourcentage d'éléments ajoutés est enregistré double dix fois par rapport des éléments suivants (Mo, Nb, Pb, Ta, Ti, V, Zr).
- Ce type d'acier est utilisé dans la fabrication des ressorts, roulement et pièces résistantes du frottement

CHAPITRE II :

Modélisation par

réseaux de neurones

II.1 Introduction :

Un réseau de neurones est un processus distribué de manière massivement parallèle, qui a une propension naturelle à mémoriser des connaissances de façon expérimentale et de les rendre disponibles pour utilisation.

Les réseaux de neurones sont la réplique des neurones biologique c'est-à-dire les neurones qui existent en vous et moi. Un ensemble de cellules sont inter connectées d'une manière à répondre d'une manière bien spécifique à une excitation [9].

On recense aujourd'hui des applications des réseaux de neurones dans des domaines très variés : la reconnaissance de caractères manuscrits, la robotique, la reconnaissance de paroles, etc. Toutes ces tâches ont un point en commun : elles sont complexes à modéliser, elles ne requièrent pas une solution unique et exacte, mais plutôt une estimation de la réponse la plus plausible, et enfin elles opèrent sur des données incertaines, toujours entachées de bruit.

Une approche par réseaux de neurones permet d'esquisser des réponses à ces problèmes, en réduisant beaucoup le temps consacré par des ingénieurs à leur analyse.

Nous aborderons dans ce chapitre un résumé sur les réseaux de neurones et surtout les modèles des réseaux que nous utilisons dans notre application.

II.2 Les réseaux de neurones :

Les réseaux de neurones artificiels sont devenus en quelques années des outils précieux dans des domaines très divers de l'industrie et des services. Néanmoins, ils n'ont pas encore atteint leur plein développement, pour des raisons plus psychologiques que techniques, liées aux connotations biologiques du terme et au fait qu'ils sont considérés, à tort, comme des outils d'Intelligence Artificielle. Or l'intérêt des réseaux de neurones, dans le domaine des Sciences de l'Ingénieur, ne doit rien à la métaphore biologique : il est uniquement dû aux propriétés mathématiques spécifiques de ces réseaux. Nous expliquons dans ce qui suit, à partir de principes simples, ce que sont réellement les réseaux de neurones, et nous délimitons leurs domaines d'excellence [10] [11].

De façon générale, on situe le début des réseaux de neurones artificiels en 1943, avec les travaux de McCulloch et Pitts qui montrent qu'un réseau de neurones discret [12], sans contrainte de topologie, peut représenter n'importe quelle fonction booléenne et donc émuler un ordinateur. En 1958, Rosenblatt propose le premier algorithme d'apprentissage [13], qui

permet d'ajuster les paramètres d'un neurone. En 1969, Minsky et Papert publient le livre « Perceptrons » [14], dans lequel ils utilisent une solide argumentation mathématique pour démontrer les limitations des réseaux de neurones à une seule couche. Ce livre aura une influence négative telle que la plupart des chercheurs quitteront le champ de recherche sur les réseaux de neurones. En 1982, Hopfield propose des réseaux de neurones associatifs et l'intérêt pour les réseaux de neurones renaît chez les scientifiques [15]. En 1986, Rumelhart, Hinton et Williams publient l'algorithme de la « rétropropagation de l'erreur » qui permet d'optimiser les paramètres d'un réseau de neurones à plusieurs couches. À partir de ce moment, la recherche sur les réseaux de neurones connaît un essor fulgurant et, au cours des années 90, les applications commerciales de ce succès académique suivent [16].

Aujourd'hui, on retrouve les réseaux de neurones solidement implantés dans diverses industries, entre autres : dans les milieux financiers, pour la prédiction des fluctuations de marché ; en pharmaceutique, pour analyser le « QSAR » (Quantitative Structure-Activity Relationship) de diverses molécules organiques ; dans le domaine bancaire, pour la détection de fraudes sur les cartes de crédit et le calcul de cotes de crédit ; dans les départements de marketing de compagnies de diverses industries, pour prévoir le comportement des consommateurs ; en aéronautique, pour la programmation de pilotes automatiques.

Les applications sont nombreuses et partagent toutes un point commun essentiel à l'utilité des réseaux de neurones : les processus pour lesquels on désire émettre des prédictions comportent de nombreuses variables explicatives et, surtout, il existe éventuellement des dépendances non linéaires de haut niveau entre ces variables qui, si elles sont découvertes et exploitées, peuvent servir à l'amélioration de la prédiction du processus.

L'avantage fondamental des réseaux de neurones, par rapport aux modèles statistiques traditionnels, réside dans le fait qu'ils permettent d'automatiser la découverte des dépendances les plus importantes du point de vue de la prédiction du processus [16].

II.2.1 Les réseaux de neurones formels :

Un « neurone formel » ou « artificiel » (ou simplement « neurone ») est une fonction algébrique non linéaire et bornée, dont la valeur dépend de paramètres appelés coefficients ou poids. Les variables de cette fonction sont habituellement appelées 'entrées' du neurone, et la valeur de la fonction est appelée sa « sortie » [17].

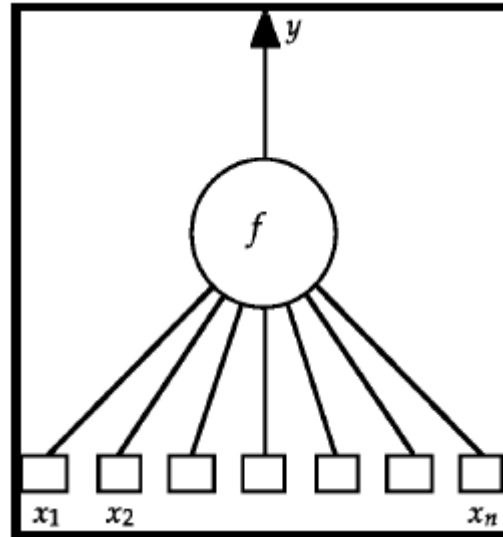


Figure II.1: Représentation graphique d'un neurone formel [17].

Un neurone réalise donc une fonction non linéaire bornée $y = f(x_1, x_2, \dots, x_n; c_1, c_2, \dots, c_p)$. Les $\{x_i\}$ sont les variables (ou entrées) et les $\{c_i\}$ sont des paramètres (ou poids). En règle générale, le calcul de la valeur de cette fonction peut se décomposer en deux étapes :

- Une combinaison linéaire des entrées :

$$V = c_0 + \sum_{i=0}^n c_i x_i \quad (\text{II.1})$$

Les c_i sont appelés poids synaptiques ou simplement poids, c_0 est appelé biais. Le biais peut être considéré comme la pondération de l'entrée 0 fixée à 1. V est appelé potentiel du neurone.

- La sortie du neurone est :

$$y = f(V) = f\left(\sum_{i=0}^n c_i x_i\right) \quad (\text{II.2})$$

La fonction f est appelée fonction d'activation ou de transfert, la figure II.2 montre les différentes fonctions d'activation utilisées dans les réseaux de neurones.

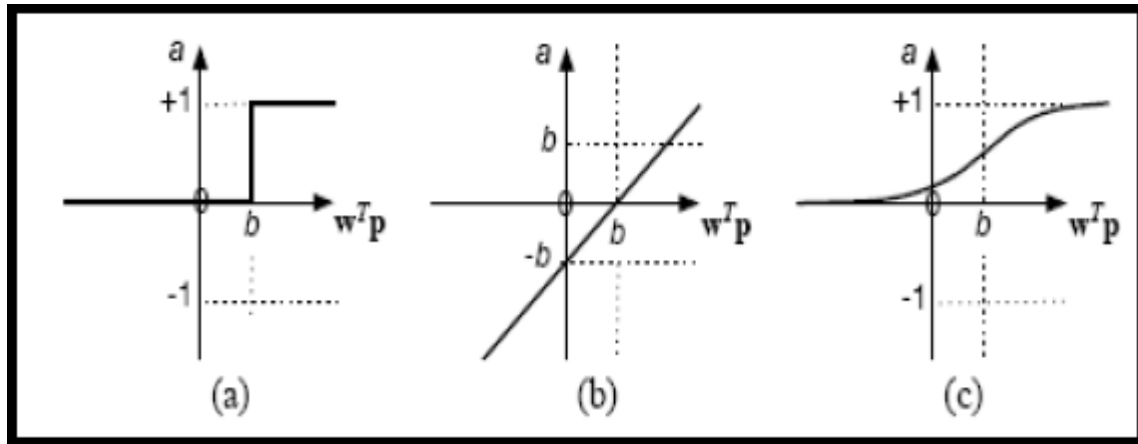


Figure II.2: Fonctions d'activation : (a) à seuil, (b) linéaire et (c) sigmoïde [17].

Un neurone formel ne réalise donc rien d'autre qu'une somme pondérée suivie d'un non linéarité. C'est l'association de tels éléments simples sous la forme de réseaux qui permet de réaliser des fonctions utiles pour des applications industrielles.

II.2.2 Architecture des réseaux de neurones :

On distingue deux grands types d'architectures de réseaux de neurones : les réseaux de neurones « non bouclés » et les réseaux de neurones « bouclés ».

II.2.2.1 Les réseaux de neurones non bouclés :

Un réseau de neurones non bouclé est représenté graphiquement par un ensemble de neurones 'connectés' entre eux, l'information circule des entrées vers les sorties, sans 'retour en arrière'. Si on représente le réseau comme un graphe dont les nœuds sont les neurones et les arêtes les 'connexions' entre eux, le graphe d'un réseau non bouclé est acyclique. Le terme de 'connexions' est une métaphore : dans la très grande majorité des applications, les réseaux de neurones sont des formules algébriques dont les valeurs numériques sont calculées par des programmes d'ordinateurs, non des objets physiques (circuits électroniques spécialisés) ; néanmoins, le terme de connexion, issu des origines biologiques des réseaux de neurones, est passé dans l'usage, car il est commode quoique trompeur. Il a même donné naissance au terme de connexionnisme.

La figure II.3 représente un réseau de neurones non bouclé qui a une structure particulière, très fréquemment utilisée : il comprend des entrées, une couche de neurones « cachés » et des

neurones de sortie. Les neurones de la couche cachée ne sont pas connectés entre eux. Cette structure est appelée « Perceptron multicouche » [18].

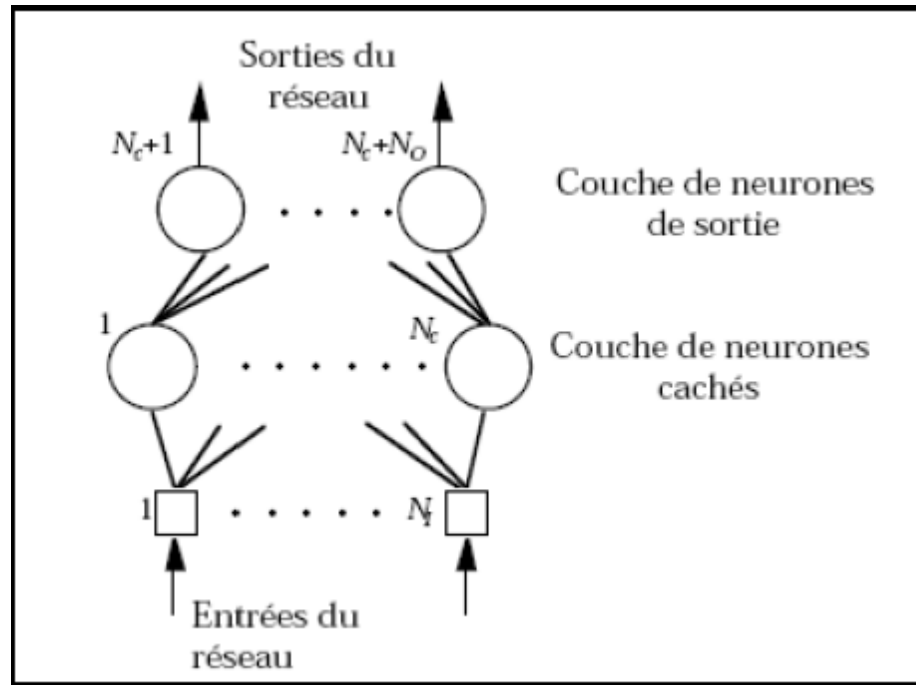


Figure II.3: Réseau de neurones non bouclé « Perceptron multicouche » [17].

Les réseaux de neurones non bouclés sont des objets statiques : si les entrées sont indépendantes du temps, les sorties le sont également. Ils sont utilisés principalement pour effectuer des tâches d'approximation de fonction non linéaire, de classification ou de modélisation de processus statiques non linéaires.

Un réseau de neurones non bouclé est constitué par plusieurs opérateurs algébriques élémentaires appelés neurones et réalise une fonction paramétrée, non linéaire par rapport aux entrées et aux paramètres.

La famille des réseaux de neurones à une couche de neurones cachés possède la propriété d'approximation parcimonieuse. Cela signifie qu'elle est capable d'approcher n'importe quelle fonction bornée et suffisamment régulière, en utilisant moins de paramètres ajustables que les familles de fonctions usuelles telles que les polynômes.

Dans l'optique d'une modélisation statistique, on utilise les réseaux de neurones pour approcher la fonction de régression du processus. L'intérêt de la parcimonie est alors de limiter le nombre d'exemples nécessaires pour obtenir une bonne estimation de la fonction de régression.

Par opposition aux modèles linéaires par rapport aux paramètres, pour lesquels la solution des moindres carrés s'obtient en résolvant un système d'équations, l'ajustement des paramètres d'un réseau de neurones (appelé aussi 'apprentissage') nécessite la mise en œuvre d'algorithmes itératifs. À cet égard, il convient de bien distinguer l'étape de calcul du gradient de la fonction de coût par rapport aux paramètres (par exemple par « la rétro propagation du gradient »), de l'étape de modification des paramètres (méthodes du gradient simple, de quasi-Newton, de Levenberg-Marquardt, etc.).

II.2.2.2. Les réseaux de neurones bouclés :

Contrairement aux réseaux de neurones non bouclés, dont le graphe de connexions est acyclique, les réseaux de neurones bouclés peuvent avoir une topologie de connexions quelconque, comprenant notamment des boucles qui ramènent aux entrées la valeur d'une ou plusieurs sorties. Pour qu'un tel système soit causal, il faut évidemment qu'à toute boucle soit associé un retard : un réseau de neurones bouclé est donc un système dynamique, régi par des équations différentielles ; comme l'immense majorité des applications sont réalisées par des programmes d'ordinateurs, on se place dans le cadre des systèmes à temps discret, où les équations différentielles sont remplacées par des équations aux différences.

Un réseau de neurones bouclé à temps discret est donc régi par une (ou plusieurs) équations aux différences non linéaires, résultant de la composition des fonctions réalisées par chacun des neurones et des retards associés à chacune des connexions.

La figure II.4 représente un réseau de neurones bouclé.

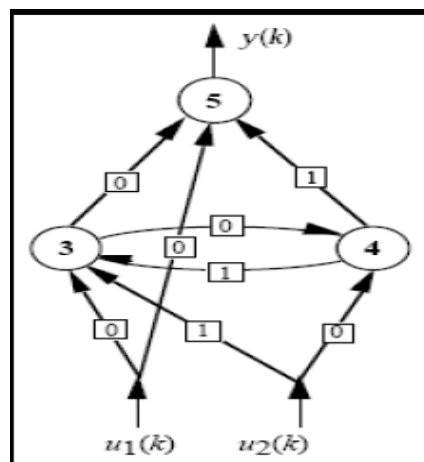


Figure II.4: Réseau de neurones bouclé [17].

Les réseaux de neurones bouclés sont utilisés pour effectuer des tâches de modélisation de systèmes dynamiques, de commande de processus, ou de filtrage [18].

II.2.3. L'apprentissage des réseaux de neurones :

L'objectif de l'apprentissage est de fournir une méthode au réseau, afin qu'il puisse ajuster ses paramètres lorsqu'on lui présente des exemples à traiter. On distingue habituellement trois types d'apprentissage : supervisé, non supervisé et hybride.

II.2.3.1 Apprentissage supervisé :

Dans ce type d'apprentissage, on fournit au réseau de neurones la donnée à traiter mais aussi la sortie attendue. Le réseau effectue une évaluation de la donnée, puis compare la valeur obtenue avec la valeur désirée, il va ensuite modifier ses paramètres internes afin de minimiser l'erreur constatée. L'apprentissage 'supervisé' consiste donc à calculer les coefficients de telle manière que les sorties du réseau de neurones soient, pour les exemples utilisés lors de l'apprentissage, aussi proches que possibles des sorties 'désirées', qui peuvent être : la classe d'appartenance de la forme que l'on veut classer, la valeur de la fonction que l'on veut approcher ou de la sortie du processus que l'on veut modéliser, ou encore la sortie souhaitée du processus à commander.

La plupart des algorithmes d'apprentissage des réseaux de neurones sont des algorithmes d'optimisation : ils cherchent à minimiser, par des méthodes d'optimisation non linéaire, une fonction de coût, qui constitue une mesure de l'écart entre les réponses réelles du réseau et ses réponses désirées. Cette optimisation se fait de manière itérative, en modifiant les poids en fonction du gradient de la fonction de coût : le gradient est estimé par une méthode spécifique aux réseaux de neurones, dite méthode de rétropropagation, puis il est utilisé par l'algorithme d'optimisation proprement dit. Les poids sont initialisés aléatoirement avant l'apprentissage, puis modifiés itérativement, jusqu'à obtention d'un compromis satisfaisant entre la précision de l'approximation sur l'ensemble d'apprentissage et la précision de l'approximation sur un ensemble de validation, distinct du précédent. Contrairement à des affirmations maintes fois répétées, l'apprentissage des réseaux de neurones n'est pas spécialement lent : il existe des algorithmes d'optimisation non linéaire extrêmement rapides qui permettent de faire des développements industriels sur de simples PC. L'apprentissage des réseaux de neurones bouclés (pour réaliser des modèles dynamiques) est très semblable à celui des réseaux non bouclés [19] [20].

II.2.3.2 Apprentissage non supervisé :

Dans ce type d'apprentissage, aucune information n'est fournie au réseau en plus des données à apprendre. Celui-ci est amené à découvrir la structure sous-jacente des données afin de les organiser en clusters.

II.2.3.3 Apprentissage hybride :

Ce type d'apprentissage est plus rare et est encore mal exploré, cette approche combine des méthodes numériques (réseaux de neurones, algorithmes génétiques) et des méthodes symboliques. Certains auteurs utilisent le terme d'apprentissage hybride pour parler d'un couplage 'supervisé non supervisé' ; dans ce cas, il s'agit d'un réseau qui met en parallèle ou en série un réseau entraîné en mode supervisé et un autre en mode non supervisé [21] [22].

II.2.4 Les différents types de réseaux de neurones :

On peut distinguer deux groupes de réseaux de neurones : les réseaux « feed-forward » et les réseaux « feedback » [23].

II.2.4.1 Les réseaux « feed-forward » :

II.2.4.1.1 Le perceptron :

C'est historiquement le premier réseau de neurones, c'est le perceptron de Rosenblatt. C'est un réseau simple, puisqu'il ne se compose que d'une couche d'entrée et d'une couche de sortie. Il est calqué, à la base, sur le système visuel et, de ce fait, est connu en premier dans la reconnaissance des formes. Cependant, il peut aussi être utilisé pour faire de la classification et pour résoudre des opérations logiques simples (telle 'ET' ou 'OU'). Sa principale limite est qu'il ne peut résoudre que des problèmes linéairement séparables. Il suit généralement un apprentissage supervisé selon la règle de correction de l'erreur (ou selon la règle de Hebb).

II.2.4.1.2 Le perceptron multicouche (PMC) :

C'est une extension du précédent, avec une ou plusieurs couches cachées entre l'entrée et la sortie. Excepté pour les couches d'entrée et de sortie, chaque neurone dans une couche est connecté à tous les neurones de la couche précédente et de la couche suivante, et il n'y a pas de connections entre les cellules d'une même couche (figure II-9). Les fonctions d'activation utilisées dans ce type de réseaux sont principalement les fonctions à seuil ou sigmoïde. Il peut résoudre des problèmes non-linéairement séparables et des problèmes logiques plus compliqués

(notamment le fameux problème du XOR), qui ne peuvent pas être résolus par un perceptron simple. Il suit aussi un apprentissage supervisé selon la règle de correction de l'erreur.

II.2.4.1.3 Les réseaux à fonction radiale :

Ce sont des réseaux que l'on nomme aussi RBF (Radial Basic Functions). L'architecture est la même que pour les PMC, cependant les fonctions de base utilisées ici sont des fonctions gaussiennes. Les RBF seront donc employés dans les mêmes types de problèmes que les PMC, en classification et en approximation de fonctions, particulièrement. L'apprentissage le plus utilisé pour les RBF est le mode hybride et les règles sont : soit la règle de correction de l'erreur, soit la règle d'apprentissage par compétition.

II.2.4.2 Les réseaux « feedback » :

II.2.4.2.1 Les cartes auto-organisatrices de Kohonen :

Ce type de réseau, appelé aussi SOM (Self Organised Maps) est un réseau à apprentissage non supervisé qui établit une carte discrète, ordonnée topologiquement en fonction de patterns d'entrée. Le réseau forme ainsi une sortie de treillis, où chaque nœud du treillis est un neurone associé à un vecteur de poids. La correspondance entre chaque vecteur de poids est calculée pour chaque entrée. Par la suite, le vecteur de poids ayant la meilleure corrélation, ainsi que certains de ses voisins vont être modifiés afin d'augmenter encore cette corrélation.

Les LVQ (Learning Vector Quantization) sont un cas particulier des SOM, où seul le nœud ayant la meilleure corrélation est adapté. On parle pour ce type de réseau, où seul le 'vainqueur' est sélectionné, de réseaux à compétition.

II.2.4.2.2 Les réseaux de Hopfield :

Les réseaux de Hopfield sont des réseaux récurrents et entièrement connectés. Dans ce type de réseaux, chaque neurone est connecté à chaque autre neurone, il n'y a aucune différenciation entre les neurones d'entrée et de sortie. Ils fonctionnent comme une mémoire associative non linéaire et sont capables de trouver un objet stocké en fonction de représentations partielles ou bruitées. L'application principale des réseaux de Hopfield est l'entrepôt de connaissances mais aussi la résolution de problèmes d'optimisation. Le mode d'apprentissage utilisé ici est le mode non supervisé [23].

II.2.4.2.3 Les réseaux ART :

Les réseaux ART (Adaptative Resonance Theory) sont des réseaux à apprentissage par compétition. Le problème majeur qui se pose dans ce type de réseaux est le dilemme

‘stabilité/plasticité’. En effet, dans un apprentissage par compétition, rien ne garantit que les catégories formées aillent rester stables. La seule possibilité pour assurer la stabilité serait que le coefficient d’apprentissage tend vers zéro, mais le réseau perdrait alors sa plasticité. Les ART sont connus spécifiquement pour contourner ce problème. Dans ce genre de réseau, les vecteurs de poids ne seront adaptés que si l’entrée fournie est suffisamment proche d’un prototype déjà connu par le réseau. On parlera alors de résonance. A l’inverse, si l’entrée s’éloigne trop des prototypes existants, une nouvelle catégorie va alors se créer, avec pour prototype l’entrée qui a engendré sa création. Le mode d’apprentissage des ART peut être supervisé ou non. Il est à noter qu’il existe deux principaux types de réseaux ART : les ART-1 pour des entrées binaires et les ART-2 pour des entrées continues [23].

II.2.5 Propriété fondamentale des réseaux de neurones :

Les réseaux de neurones, tels que nous les avons définis précédemment, possèdent une propriété remarquable, qui est à l’origine de leur intérêt pratique dans des domaines très divers : ce sont des « approximateurs » universels parcimonieux.

Sans entrer dans les détails mathématiques, la propriété d’approximation peut être énoncée de la manière suivante : toute fonction bornée suffisamment régulière peut être approchée avec une précision arbitraire, dans un domaine fini de l’espace de ses variables, par un réseau de neurones comportant une couche avec un nombre fini de neurones cachés, possédant tous la même fonction d’activation, et un neurone de sortie linéaire [17]. Cette propriété n’est pas spécifique aux réseaux de neurones : il existe bien d’autres familles de fonctions paramétrées possédant cette propriété ; c’est le cas notamment des ondelettes, des fonctions radiales, des fonctions splines, par exemple.

La spécificité des réseaux de neurones réside dans le caractère parcimonieux de l’approximation : à précision égale, les réseaux de neurones nécessitent moins de paramètres ajustables (les poids des connexions) que les approximateurs universels couramment utilisés ; plus précisément, le nombre de poids varie linéairement avec le nombre de variables de la fonction à approcher, alors qu’il varie exponentiellement pour la plupart des autres approximateurs [24].

C'est cette remarquable parcimonie qui justifie l'intérêt industriel des réseaux de neurones. En pratique, dès qu'un problème fait intervenir plus de deux variables, les réseaux de neurones sont, en général, préférables aux autres méthodes.

Qualitativement, la propriété de parcimonie peut se comprendre de la manière suivante : lorsque l'approximation est une combinaison linéaire de fonctions élémentaires fixées (des monômes par exemple, où des gaussiennes à centres et écarts-types fixes), on ne peut ajuster que les coefficients de la combinaison ; en revanche, lorsque l'approximation est une combinaison linéaire de fonctions non linéaires à paramètres ajustables (un Perceptron multicouche par exemple), on ajuste à la fois les coefficients de la combinaison et la forme des fonctions que l'on combine. Ainsi, dans un Perceptron multicouche, les poids de la première couche déterminent la forme de chacune des sigmoïdes réalisées par les neurones cachés, et les poids de la seconde couche déterminent une combinaison linéaire de ces fonctions. On conçoit facilement que cette souplesse supplémentaire, conférée par le fait que l'on ajuste la forme des fonctions que l'on superpose, permet d'utiliser un plus petit nombre de fonctions élémentaires, donc un plus petit nombre de paramètres ajustables. Nous allons voir dans le paragraphe suivant pourquoi cette propriété de parcimonie est précieuse dans les applications industrielles [24].

II.2.6 Réseaux de neurones et régression non linéaire :

Dans la pratique, on n'utilise pas les réseaux de neurones pour réaliser des approximations de fonctions connues. Le plus souvent, le problème qui se pose à l'ingénieur est le suivant : il dispose d'un ensemble de mesures de variables d'un processus de nature quelconque (physique, chimique, économique, financier, ...), et du résultat de ce processus ; il suppose qu'il existe une relation déterministe entre ces variables et ce résultat, et il cherche une forme mathématique de cette relation, valable dans le domaine où les mesures ont été effectuées, sachant que : les mesures sont en nombre fini, elles sont certainement entachées de bruit et toutes les variables qui déterminent le résultat du processus ne sont pas forcément mesurées. En d'autres termes, l'ingénieur cherche un modèle du processus qu'il étudie, à partir des mesures dont il dispose, et d'elles seules : on parle de modélisation « boîte noire ». Dans le jargon des réseaux de neurones, les données à partir desquelles on cherche à construire le modèle s'appellent des exemples.

En quoi la propriété d'approximation parcimonieuse peut-elle être utile pour résoudre ce genre de problèmes ? Ce que l'ingénieur cherche à obtenir à l'aide de son modèle, c'est la « vraie » fonction qui relie la grandeur y_p que l'on veut modéliser aux variables $\{x\}$ qui la déterminent, c'est-à-dire la fonction que l'on obtiendrait en faisant une infinité de mesures de y_p pour chaque valeur possible de $\{x\}$: en termes de statistiques, on cherche la fonction de régression de la grandeur à modéliser. Cette fonction est inconnue, mais on peut chercher une approximation à partir des mesures disponibles : les réseaux de neurones sont donc de bons candidats pour cela, si la fonction de régression cherchée est non linéaire. Cette approximation est obtenue en estimant les paramètres d'un réseau de neurones au cours d'une phase dite d'apprentissage. C'est ici que la propriété d'approximation parcimonieuse des réseaux de neurones est précieuse : en effet, le nombre de mesures nécessaires pour estimer les paramètres de manière significative est d'autant plus grand que le nombre de paramètres est grand. Ainsi, pour modéliser une grandeur avec une précision donnée à l'aide d'un réseau de neurones, il faut moins de données que pour la modéliser, avec une précision comparable, à l'aide d'une régression linéaire multiple ; de manière équivalente, un réseau de neurones permet, avec les mêmes données disponibles, de réaliser une approximation plus précise qu'une régression linéaire multiple.

De manière générale, un réseau de neurones permet donc de faire un meilleur usage des mesures disponibles que les méthodes de régression non linéaires conventionnelles. Ce gain peut être considérable, lorsque le processus à modéliser dépend de plusieurs variables : rappelons en effet que le nombre de paramètres (donc de mesures) varie exponentiellement pour les méthodes conventionnelles de régression non linéaire, alors qu'elle varie linéairement pour les réseaux de neurones [17].

Ainsi, à la lumière de cette propriété fondamentale, la technique des réseaux de neurones apparaît comme une puissante méthode de régression non linéaire : ce n'est donc rien d'autre qu'une extension des méthodes de régression linéaire ou multilinéaires proposées par tous les logiciels qui permettent de faire de la modélisation de données. Contrairement à une croyance répandue, elle ne relève donc pas de l'Intelligence Artificielle au sens classique du terme, mais elle constitue une branche des statistiques appliquées. Il ne faut donc pas être victime du vocabulaire anthropomorphique utilisé (neurones, apprentissage, etc.) ; le tableau II-1 résume les équivalences entre le vocabulaire des statistiques et celui des réseaux de neurones.

Tableau II.1: Réseaux de neurones et statistiques [17].

RÉSEAUX DE NEURONES	STATISTIQUES
Choix de l'architecture	Choix de la famille de fonctions destinées à approcher la fonction de régression
Ensemble d'apprentissage	Observations
Apprentissage	Estimation des paramètres de l'approximation de la fonction de régression
Généralisation	Interpolation, extrapolation

II.2.7 Mise en œuvre des réseaux de neurones et domaines d'application :

Pour réaliser l'approximation de la fonction de régression cherchée, à partir d'échantillons généralement bruités, à l'aide d'un réseau de neurones, trois étapes successives sont nécessaires:

- Il faut tout d'abord choisir l'architecture du réseau, c'est-à-dire les entrées externes, le nombre de neurones cachés, et l'agencement des neurones entre eux, de telle manière que le réseau soit en mesure de reproduire ce qui est déterministe dans les données ; le nombre de poids ajustables est un des facteurs fondamentaux de la réussite d'une application : si le réseau possède un trop grand nombre de poids, c'est-à-dire si le réseau est trop 'souple', il risque de s'ajuster au bruit qui est présent dans les données de l'ensemble d'apprentissage, et, même en l'absence de bruit, il risque de présenter des oscillations non significatives entre les points d'apprentissage, donc de posséder de mauvaises propriétés d'interpolation (ou, dans le jargon des réseaux de neurones, de 'généralisation') ; si par contre ce nombre est trop petit, le réseau est trop 'rigide' et ne peut reproduire la partie déterministe de la fonction. Le problème de la détermination de l'architecture optimale est resté pendant longtemps un problème ouvert, mais il existe actuellement diverses méthodes, mettant notamment en jeu des tests statistiques, qui permettent de déterminer cette architecture pour une vaste classe de réseaux ; la figure II-6 illustre l'influence du nombre de paramètres sur la qualité de l'approximation : le réseau le plus parcimonieux (4 neurones cachés, soit 13 coefficients) produit un bien meilleur ajustement qu'un réseau trop riche en coefficients (8 neurones cachés, soit 25 coefficients).
- Il faut calculer ensuite les poids du réseau ou, en d'autres termes, 'estimer les paramètres de la régression non linéaire' à partir des exemples, en minimisant l'erreur d'approximation sur

les points de l'ensemble d'apprentissage, de telle manière que le réseau réalise la tâche désirée. Ce calcul des coefficients synaptiques constitue l'apprentissage supervisé pour le réseau de neurones.

- Il faut estimer enfin la qualité du réseau obtenu, en lui présentant des exemples qui ne font pas partie de l'ensemble d'apprentissage.

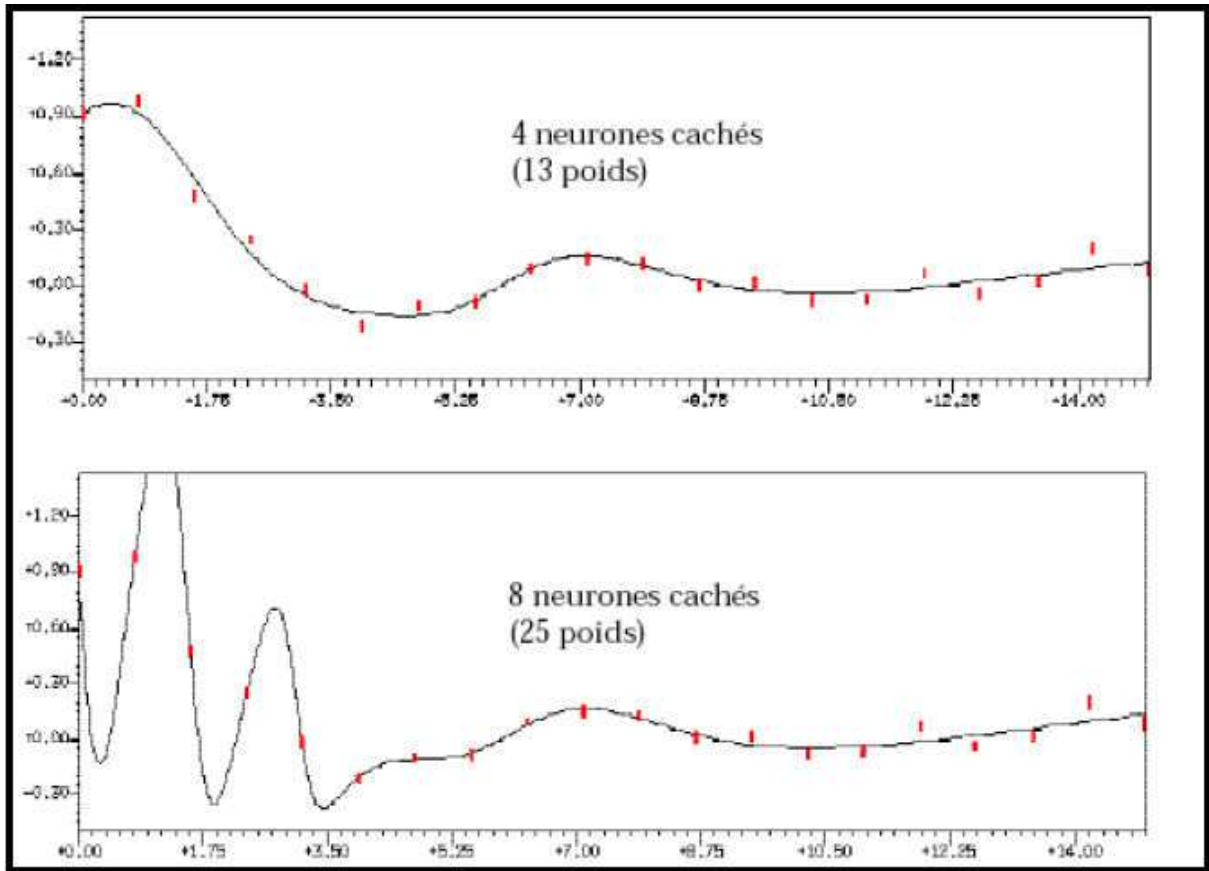


Figure II.5: Influence du nombre de paramètres sur la qualité de l'approximation [17].

Les grands domaines d'application des réseaux de neurones découlent naturellement de leurs propriétés fondamentales :

- **La régression non linéaire, ou modélisation de données statiques :** Il existe une immense variété de phénomènes statiques qui peuvent être caractérisés par une relation déterministe entre des causes et des effets ; les réseaux de neurones sont de bons candidats pour modéliser de telles relations à partir d'observations expérimentales, sous réserve que celles-ci soient suffisamment nombreuses et représentatives.

- **La modélisation de processus dynamiques non linéaires :** Modéliser un processus, c'est trouver un ensemble d'équations mathématiques qui décrivent le comportement dynamique du processus, c'est-à-dire l'évolution de ses sorties en fonction de celle de ses entrées ; c'est donc typiquement un problème qui peut être avantageusement résolu par un réseau de neurones, si le phénomène qu'on veut modéliser est non-linéaire. La prévision de séries chronologiques (prévisions financières, prévision de consommation, etc.) entre dans ce cadre [17].
- **La commande de processus :** Commander un processus, c'est imposer à celui-ci un comportement défini à l'avance en fonction des signaux de commande ; l'ensemble (commande + processus) peut donc être considéré comme un système qui réalise une fonction non-linéaire qu'un réseau de neurones peut approcher.
- **La classification :** Supposons que l'on désire classer des formes en deux catégories, A ou B, en fonction de certaines caractéristiques de ces formes ; on peut définir une fonction φ qui vaut (+1) pour toutes les formes de la classe A et (-1) pour toutes les formes de la classe B. Les réseaux de neurones sont de bons candidats pour réaliser une approximation de cette fonction φ et on peut démontrer que cette approximation constitue une estimation de la probabilité d'appartenance de la forme inconnue à la classe A. Les réseaux de neurones fournissent donc une information très riche, qui est loin d'être une simple réponse binaire. Cette propriété remarquable, que les réseaux de neurones partagent avec d'autres classifieurs, n'est malheureusement pas mise à profit dans la plupart des applications.

En conclusion, les points fondamentaux qu'il convient de toujours garder à l'esprit lorsqu'on cherche à mettre en œuvre des réseaux de neurones sont donc les suivants :

- Les réseaux de neurones sont des outils statistiques, qui permettent d'ajuster des fonctions non linéaires très générales à des ensembles de points ; comme toute méthode statistique, l'utilisation de réseaux de neurones nécessite que l'on dispose de données suffisamment nombreuses et représentatives ;
- Les réseaux de neurones sont des approximateurs parcimonieux ;
- Les réseaux de neurones permettent de modéliser des phénomènes statiques (réseaux non bouclés) et dynamiques (réseaux bouclés) ;

- Il est toujours souhaitable, et souvent possible, pour la conception du réseau, d'utiliser les connaissances mathématiques dont on dispose sur le phénomène à modéliser : les réseaux de neurones ne sont pas nécessairement des « boîtes noires » [17].

CHAPITRE III :

Résultats et discussions

III.1 Introduction :

Dans ce chapitre, nous utilisons les réseaux de neurones pour étudier les efforts de coupe pendant le chariotage de l'acier 20 Mn Cr5 avec un outil en carbure métallique en fonction des paramètres de coupe (V_c , f et a_p) en se basant sur une base de données élaboré par Bentaleb Fayçal [25].

Les résultats expérimentaux sont illustrés dans le tableau III-1.

Tableau III-1: Résultats des efforts de la coupe (F_a , F_c et F_p) en fonction des paramètres du régime de coupe.

Essais	Factures			Efforts de coup		
	V_c [m/min]	f [mm/tr]	a_p [mm]	F_a [N]	F_c [N]	F_p [N]
1	110	0.09	1	115.170	143.830	224.093
2	110	0.09	2	144.204	189.204	317.466
3	110	0.09	3	160.171	228.500	358.426
4	110	0.11	1	118.884	130.263	229.514
5	110	0.11	2	155.254	182.236	321.020
6	110	0.11	3	169.678	211.523	366.433
7	110	0.14	1	133.189	150.219	260.777
8	110	0.14	2	165.513	220.449	360.844
9	110	0.14	3	182.371	249.539	398.188
10	125	0.09	1	102.790	130.424	202.220
11	125	0.09	2	130.986	150.754	273.934
12	125	0.09	3	157.165	180.504	326.336
13	125	0.11	1	100.888	105.736	202.057
14	125	0.11	2	127.758	166.213	285.146
15	125	0.11	3	157.967	191.632	323.977
16	125	0.14	1	114.238	145.634	221.560
17	125	0.14	2	148.139	222.531	323.267
18	125	0.14	3	197.668	240.832	387.921
19	155	0.09	1	87.182	106.750	169.977
20	155	0.09	2	115.359	157.744	227.431
21	155	0.09	3	154.024	186.566	279.833
22	155	0.11	1	85.452	116.918	181.133
23	155	0.11	2	125.489	161.200	249.986
24	155	0.11	3	153.400	187.133	289.210
25	155	0.14	1	91.422	131.768	192.162
26	155	0.14	2	132.332	190.788	275.781
27	155	0.14	3	175.462	238.502	348.668

III.2 Méthodologie de la modélisation par les réseaux de neurones artificiels (ANN) :

Pour modéliser les efforts de coupe par ANN, nous avons utilisé le logiciel MATLAB R2013a.

Pour les entrées ‘inputs’ sur workspace :

- La première ligne contient les valeurs de la variable Vc (27 Colonnes) ;
- La deuxième ligne contient les valeurs de la variable f (27 Colonnes) ;
- La troisième ligne contient les valeurs de la variable ap (27 Colonnes).

	1	2	3	4	5	6	7	8	9	10
1	110	110	110	110	110	110	110	110	110	1
2	0.0900	0.0900	0.0900	0.1100	0.1100	0.1100	0.1400	0.1400	0.1400	0.09
3	1	2	3	1	2	3	1	2	3	

Figure III.1: Insertion des entrées dans le tableau.

Pour les sorties « outputs » sur workspace :

Chaque ligne contient les valeurs de les efforts (27 Colonnes) (a) Fa et (b) Fc et (c) Fp.

	1	2	3	4	5	6	7	8	9	10
1	115.1700	144.2040	160.1710	118.8840	155.2540	169.6780	133.1890	165.5130	182.3710	102.79
2										
3										

(a)

	1	2	3	4	5	6	7	8	9	10
1	143.8300	189.2040	228.5000	130.2630	182.2360	211.5230	150.2190	220.4490	249.5390	130.42
2										
3										

(b)

	1	2	3	4	5	6	7	8	9	10
1	224.0930	317.4660	358.4260	229.5140	321.0200	366.4330	260.7770	360.8440	398.1880	202.22
2										
3										

(c)

Figure III.2: Insertion des valeurs des critères de les efforts (a) Fa et (b) Fc et (c) Fp.

Dans notre cas, nous n'avons pas entré de valeurs de poids, donc le programme les prend avec des valeurs initiales aléatoires.

Notre travail consiste à étudier les efforts de coupe ont utilisons les réseaux de neurones avec 3, 5 et 8 couches cachées et de faire la comparaison entre eux.

III.2.1 Analyse d'un ANN pour l'effort Fa :

III.2.1.1 Utilisation de 3 couches cachées :

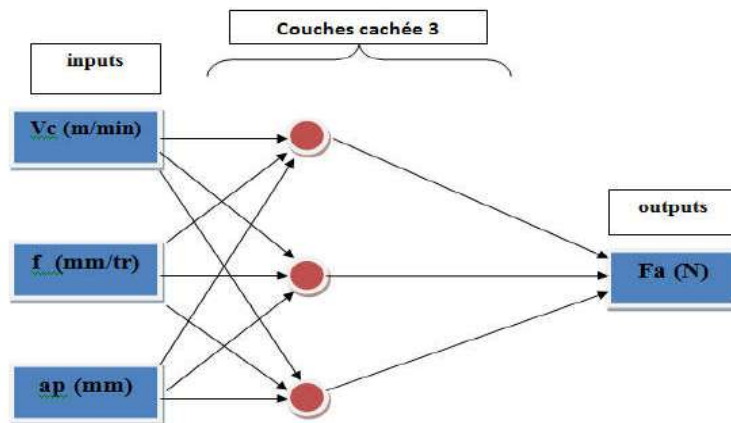


Figure III.3: Diagramme optimale pour la modélisation de Fa ANN3.

Tableau III.2: Performance des structures ANN du modèle de Fa avec 3 couches cachées.

Apprentissage		Validation	
Fa (N)	Mesures	Fa (N)	Mesures
R ²	0.99896027	R ²	0.99898026
Racine de l'erreur quadratique moyenne (RMSE)	3.14880637	Racine de l'erreur quadratique moyenne (RMSE)	13.9651518

Ecart absolu moyen	2.56473889		2.8378
Somme des carrés des écarts (SSE)	178.469668	Somme des carrés des écarts (SSE)	125.686367
Somme des fréquences	18	Somme des fréquences	9

Après avoir assemblé et arrangé le programme, nous commençons le processus de formation:

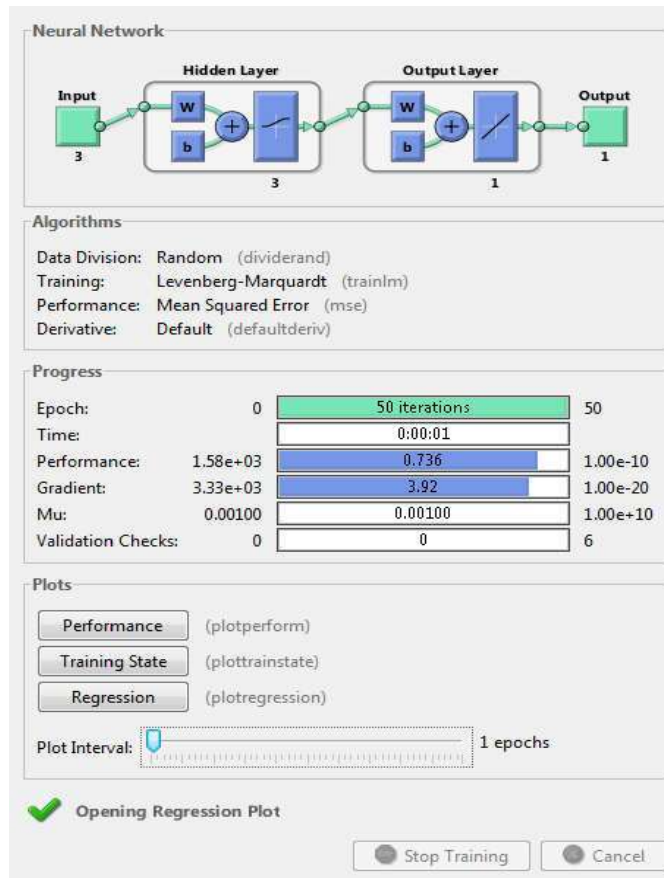


Figure III.4: Le processus de formation dans le réseau neuronal pour Fa_{ANN3}.

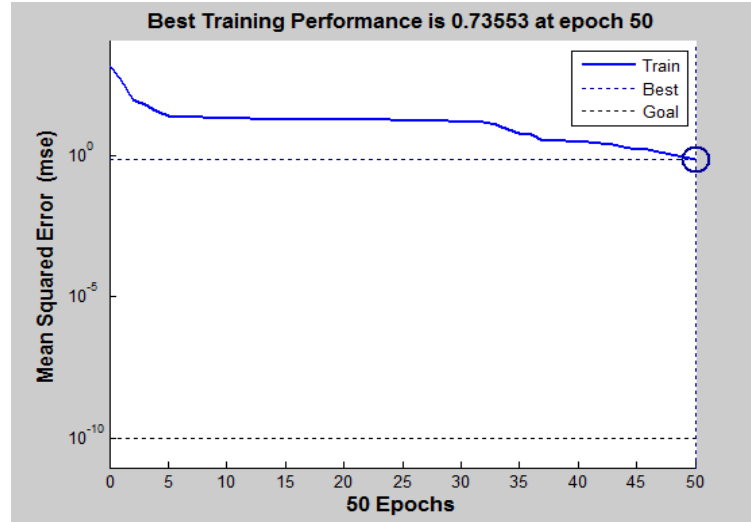


Figure III.5: Courbe de test de performance du système pour Fa_{ANN3} .

La figure montre la courbe d'erreur de performance du système. On peut dire que l'erreur quadratique moyenne de la courbe qui vaut 0.73553 à 50 itérations.

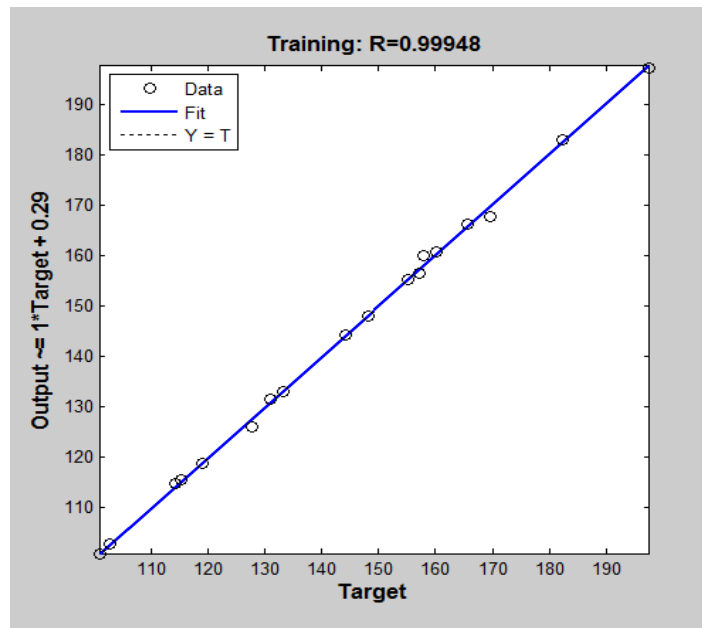


Figure III.6: Régression d'entraînement des réseaux de neurones pour Fa_{ANN3} .

Le résultat du test de régression du modèle obtenu par les réseaux de neurones nous donne un apprentissage égal à 99,948%. Cette valeur indique que 99,948% de la variance simple de l'effort Fa s'explique par les variables indépendantes, et 0,052% reste inexpliqués.

III.2.1.2 Utilisation de 5 couches cachées :

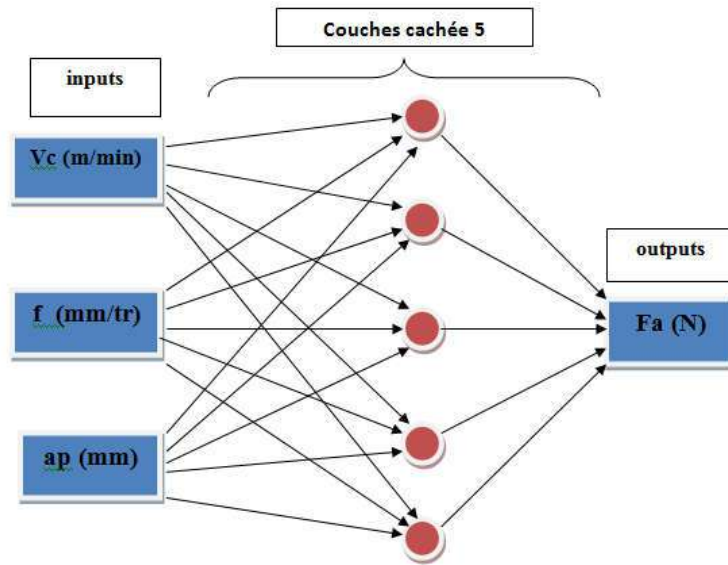


Figure III.7: Diagramme optimale pour la modélisation de F_a ANN5.

Tableau III.3: Performance des structures ANN du modèle de F_a avec 5 couches cachées.

Apprentissage		Validation	
Fa(N)	Mesures	Fa(N)	Mesures
R ²	0.998880314	R ²	0.99982001
Racine de l'erreur quadratique moyenne (RMSE)	2.63831226	Racine de l'erreur quadratique moyenne (RMSE)	2.18491499
Ecart absolu moyen	1.866411111	Ecart absolu moyen	1.4959
Somme des carrés des écarts (SSE)	125.292448	Somme des carrés des écarts (SSE)	42.9646815
Somme des fréquences	18	Somme des fréquences	9

Après avoir assemblé et arrangé le programme, nous commençons le processus de formation:

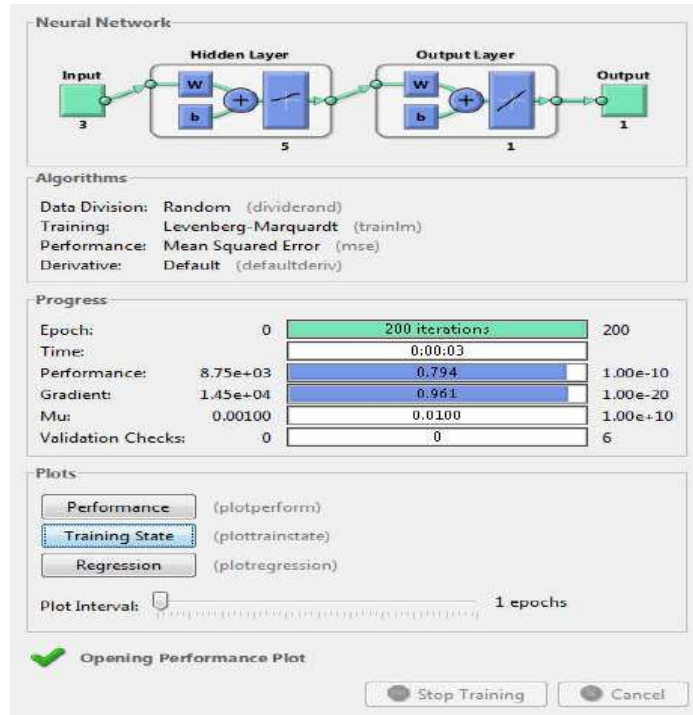


Figure III.8: Le processus de formation dans le réseau neuronal pour Fa_{ANN5}.

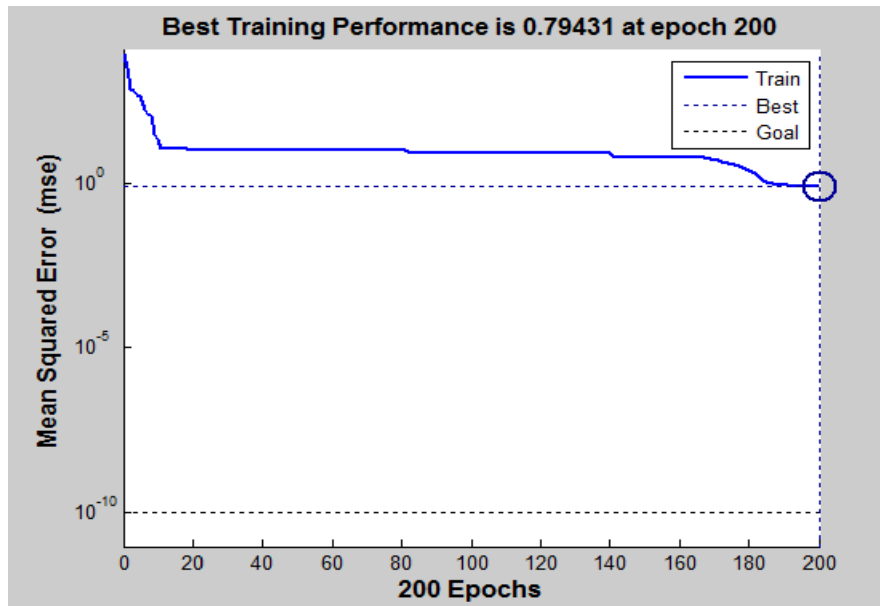


Figure III.9: Courbe de test de performance du système pour Fa_{ANN5}.

La figure montre la courbe d'erreur de performance du système. On peut dire que l'erreur quadratique moyenne de la courbe qui vaut 0.79431 à 200 itérations.

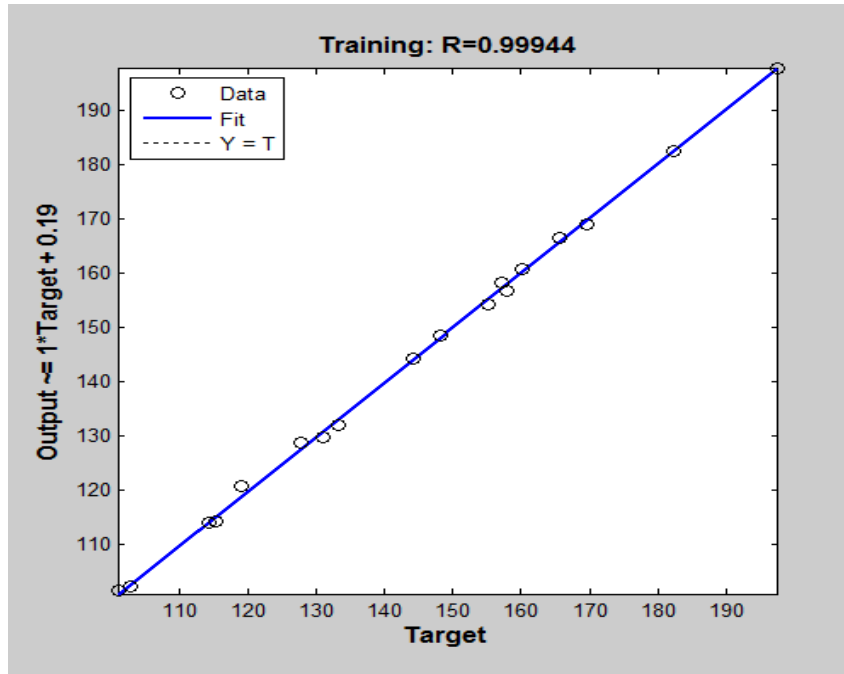


Figure III.10: Régression d'entraînement des réseaux de neurones pour Fa_{ANN5} .

Le résultat du test de régression du modèle obtenu par les réseaux de neurones nous donne un apprentissage égal à 99,944%. Cette valeur indique que 99,944% de la variance simple de l'effort Fa s'explique par les variables indépendantes, et 0,056% reste inexpliqués.

III.2.1.3 Utilisation de 8 couches cachées :

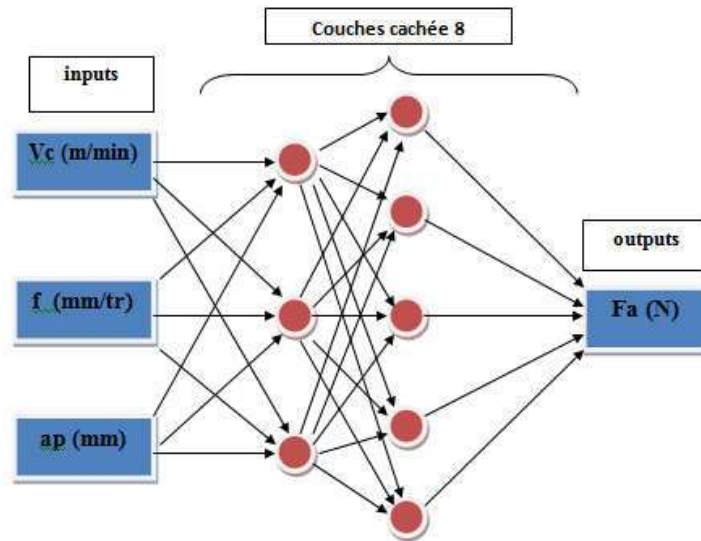


Figure III.11: Diagramme optimale pour la modélisation de Fa_{ANN8} .

Tableau III.4: Performance des structures ANN du modèle de Fa avec 8 couches cachées.

Apprentissage		Validation	
Fa (N)	Mesures	Fa (N)	Mesures
R ²	0.99992	R ²	0.99980001
Racine de l'erreur quadratique moyenne (RMSE)	0.71287427	Racine de l'erreur quadratique moyenne (RMSE)	0.03065539
Ecart absolu moyen	0.45342222		0.11086667
Somme des carrés des écarts (SSE)	9.14741498	Somme des carrés des écarts (SSE)	0.27589848
Somme des fréquences	18	Somme des fréquences	9

Après avoir assemblé et arrangé le programme, nous commençons le processus de formation:

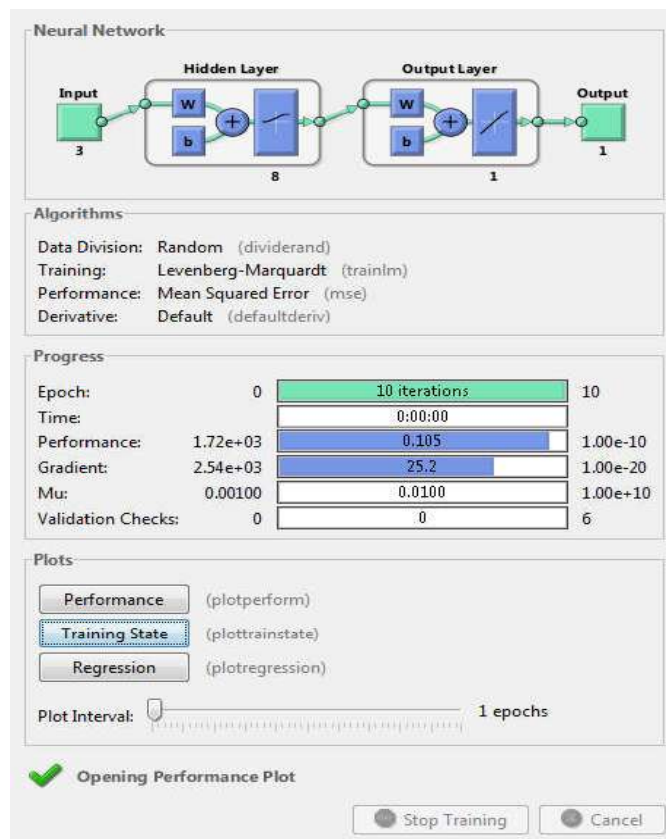


Figure III.12: Le processus de formation dans le réseau neuronal pour Fa_{ANN8}.

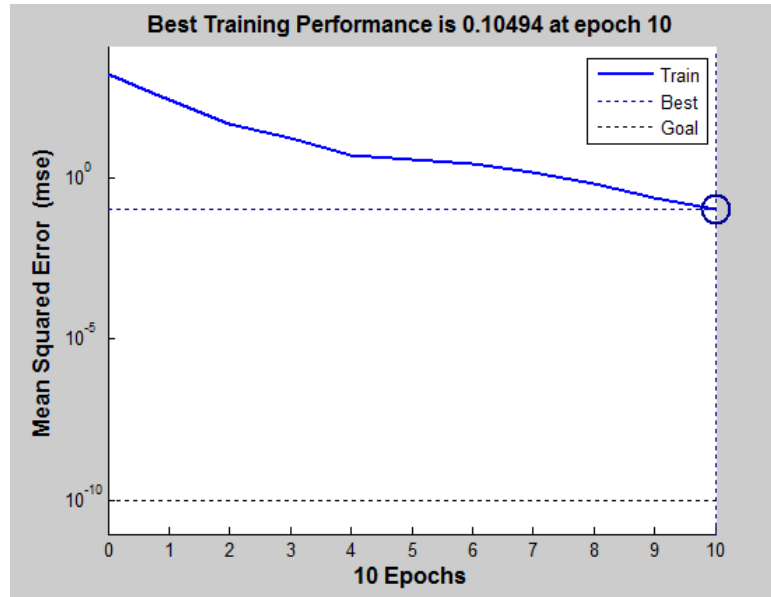


Figure III.13: Courbe de test de performance du système pour Fa_{ANN8}.

La figure montre la courbe d'erreur de performance du système. On peut dire que l'erreur quadratique moyenne de la courbe qui vaut 0,10494 à 10 itérations.

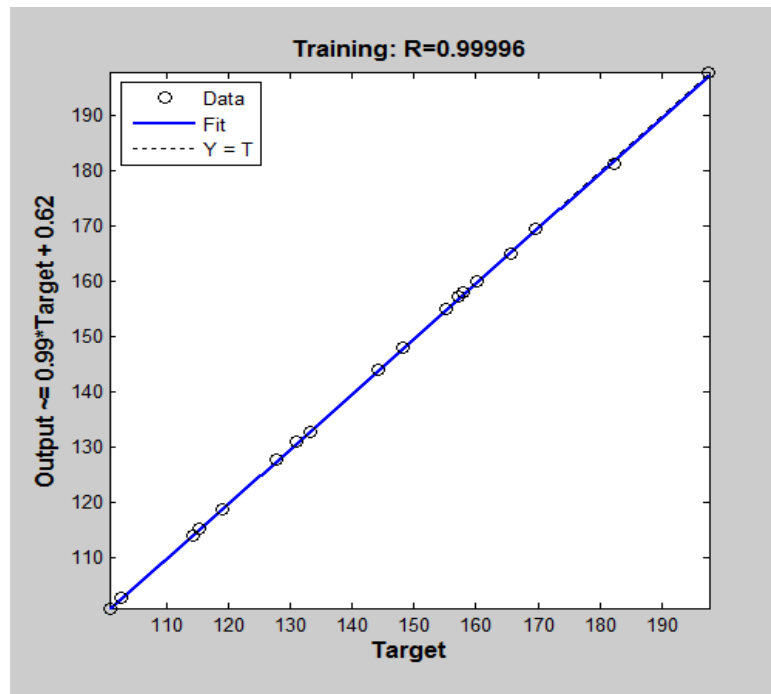


Figure III.14: Régression d'entraînement des réseaux de neurones pour Fa_{ANN8}.

Le résultat du test de régression du modèle obtenu par les réseaux de neurones nous donne un apprentissage égal à 99,996%. Cette valeur indique que 99,996% de la variance simple de l'effort F_a s'explique par les variables indépendantes, et 0,004% reste inexpliqués.

III.2.1.4 Comparaison les résultats obtenus pour F_a :

Le tableau suivant montre la comparaison entre les résultats obtenus en utilisant 3, 5 et 8 couches cachées.

Tableau III.5: Comparaison entre les résultats expérimentaux et simulés.

Essais	$F_{a_{ANN3}}$	$F_{a_{ANN5}}$	$F_{a_{ANN8}}$	$F_{a_{exp}}$	$EF_{a_{ANN3}}$	$EF_{a_{ANN5}}$	$EF_{a_{ANN8}}$
1	113.0106	115.17	114.1742	115.17	2.1594	0	0.9958
2	145.2312	141.0457	145.2229	144.204	1.0272	3.1583	1.0189
3	162.8765	160.171	159.7127	160.171	2.7055	0	0.4583
4	119.5435	122.6623	120.0286	118.884	0.6595	3.7783	1.1446
5	152.533	151.1179	153.3206	155.254	2.721	4.1361	1.9334
6	172.131	169.678	170.5748	169.678	2.453	0	0.8968
7	135.9955	127.83	133.0623	133.189	2.8065	5.359	0.1267
8	165.8648	164.077	166.4265	165.513	0.3518	1.436	0.9135
9	186.6733	182.7469	181.9296	182.371	4.3023	0.3759	0.4414
10	101.8448	100.4671	102.7407	102.79	0.9452	2.3229	0.0493
11	136.9363	130.986	130.9759	130.986	5.9503	0	0.0101
12	159.2179	157.8194	157.1599	157.165	2.0529	0.6544	0.0051
13	98.2605	100.888	100.9399	100.888	2.6275	0	0.0519
14	134.2093	131.9496	127.754	127.758	6.4513	4.1916	0.004
15	157.7289	162.9998	158.0178	157.967	0.2381	5.0328	0.0508
16	112.591	115.3236	114.2264	114.238	1.647	1.0856	0.0116
17	149.6365	146.4067	148.1459	148.139	1.4975	1.7323	0.0069
18	192.0987	197.3358	197.6255	197.668	5.5693	0.3322	0.0425
19	82.7137	87.182	87.1222	87.182	4.4683	0	0.0598
20	118.2058	118.9342	115.323	115.359	2.8468	3.5752	0.036
21	145.8974	149.7944	154.4133	154.024	8.1266	4.2296	0.3893
22	90.2409	85.6068	85.4899	85.452	4.7889	0.1548	0.0379
23	126.1458	122.4041	125.5345	125.489	0.6568	3.0849	0.0455
24	155.9771	153.4	153.3428	153.4	2.5771	0	0.0572
25	90.6923	92.2246	91.4529	91.422	0.7297	0.8026	0.0309
26	133.26	132.1575	132.339	132.332	0.928	0.1745	0.007
27	175.88	174.0205	175.1278	175.462	0.418	1.4415	0.3342
Somme					71.7055	47.0585	9.1594
Moyenne					2.65575926	1.742907407	0.33923704

La comparaison entre les valeurs expérimentales avec les valeurs simulées est exprimée par la figure suivante :

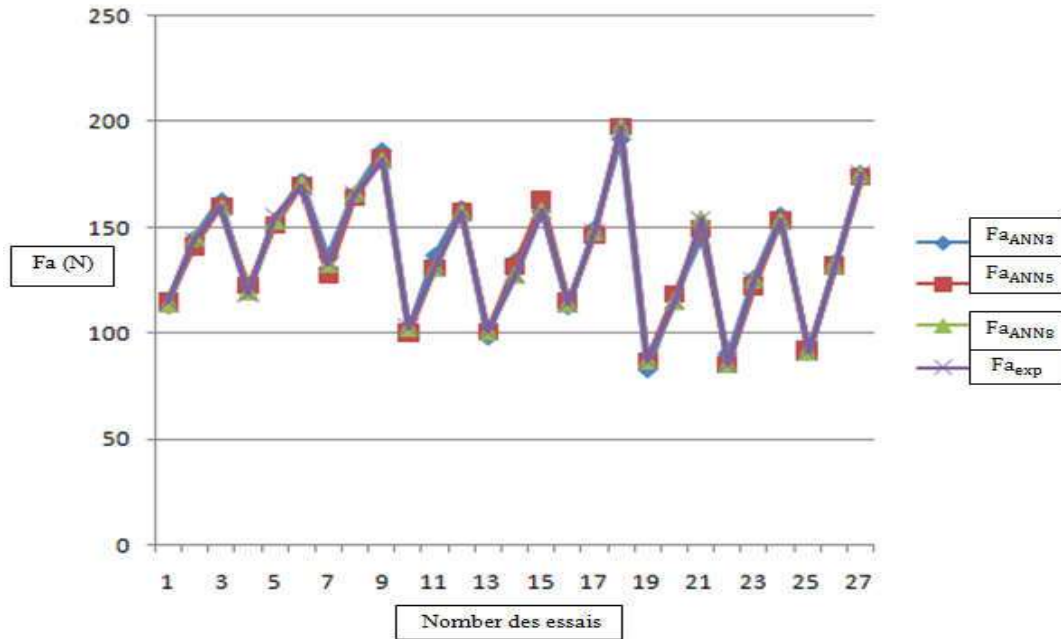


Figure III.15: Comparaison entre les Fa_{exp} et Fa Simulées.

Nous notons qu'il existe une nette différence entre les résultats expérimentaux de Fa_{exp} et les résultats des réseaux de neurones Fa_{ANN3} et Fa_{ANN5} , et nous notons qu'il y a une bonne correspondance entre les résultats expérimentaux de Fa_{exp} et les résultats de la Fa_{ANN8} expérimentale.

L'erreur obtenue entre les valeurs expérimentales et les valeurs simulées est présentée par la figure suivante :

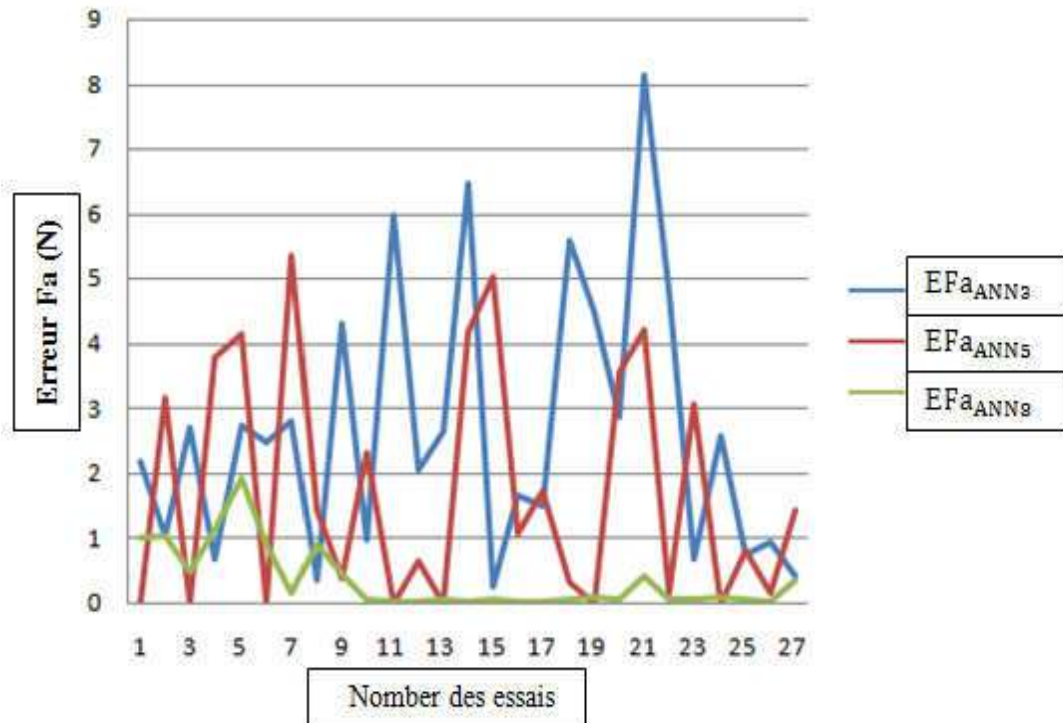


Figure III.16: Courbe de comparaison entre les erreurs de Fa simulées.

III.2.2 Analyse d'un ANN pour l'effort F_c :

III.2.2.1 Utilisation de 3 couches cachées :

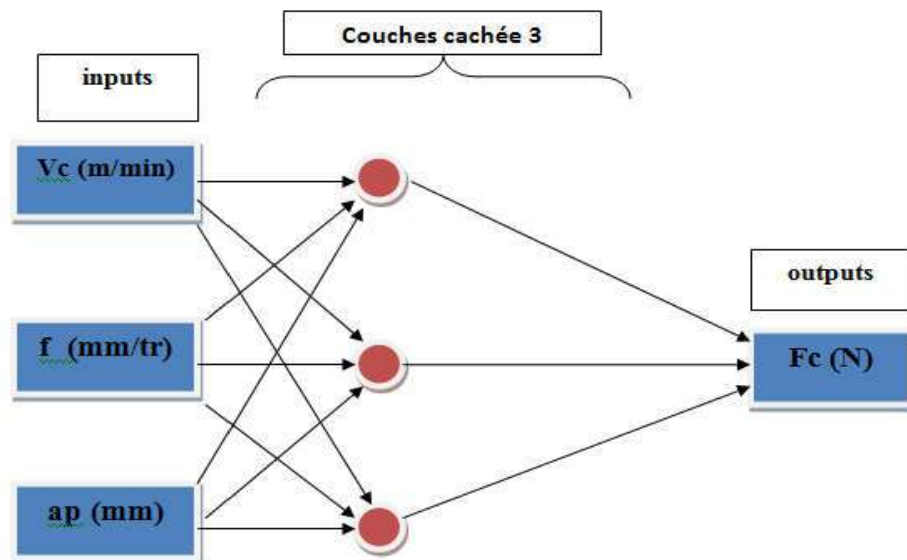


Figure III.17: Diagrammes optimum pour la modélisation de $F_{c_{ANN3}}$.

Tableau III.6: Performance des structures ANN du modèle de Fc avec 3 couches cachées.

Apprentissage		Validation	
Fc (N)	Mesures	Fc (N)	Mesures
R ²	0.99408876	R ²	0.99672269
Racine de l'erreur quadratique moyenne (RMSE)	5.51457999	Racine de l'erreur quadratique moyenne (RMSE)	8.36478275
Ecart absolu moyen	4.4364		2.12382222
Somme des carrés des écarts (SSE)	547.390665	Somme des carrés des écarts (SSE)	75.2830447
Somme des frequencies	18	Somme des frequencies	9

Après avoir assemblé et arrangé le programme, nous commençons le processus de formation:

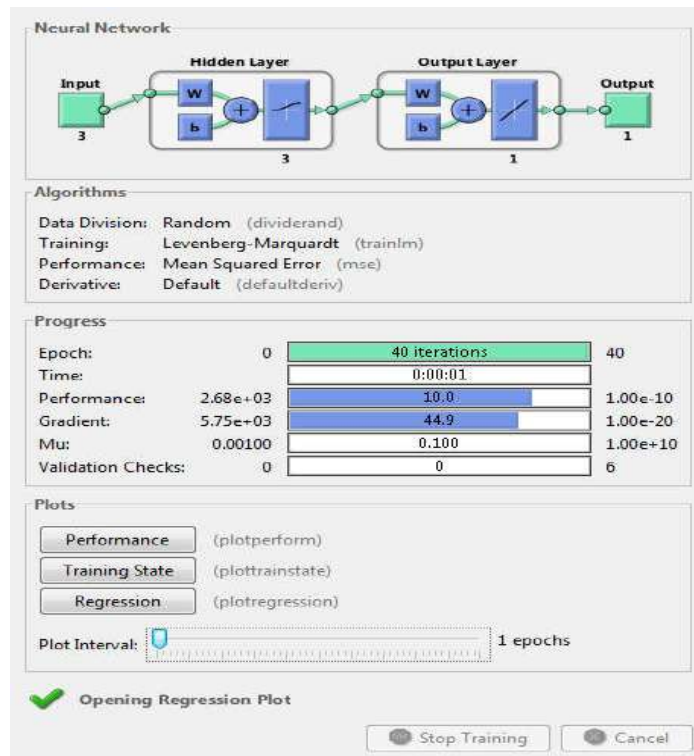


Figure III.18: Le processus de formation dans le réseau neuronal pour Fc_{ANN3}.

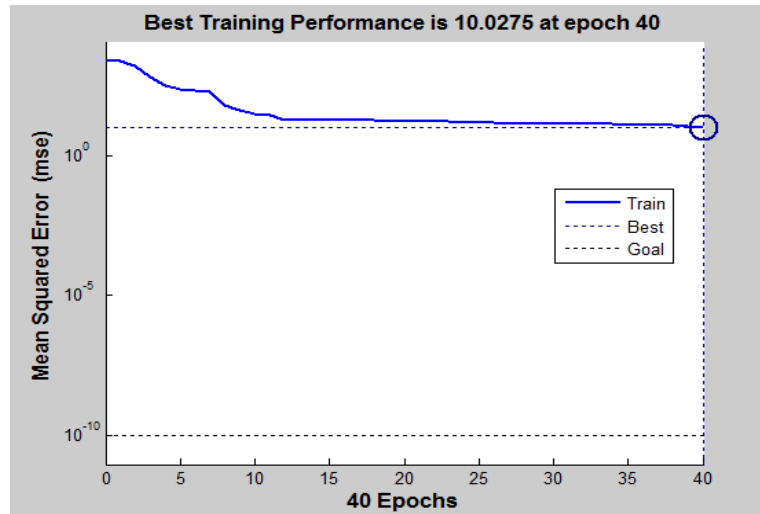


Figure III.19: Courbe de test des performances du système pour Fc_{ANN3} .

La figure montre la courbe d'erreur de performance du système. On peut dire que l'erreur quadratique moyenne de la courbe qui vaut 10,2692 à 40 itérations.

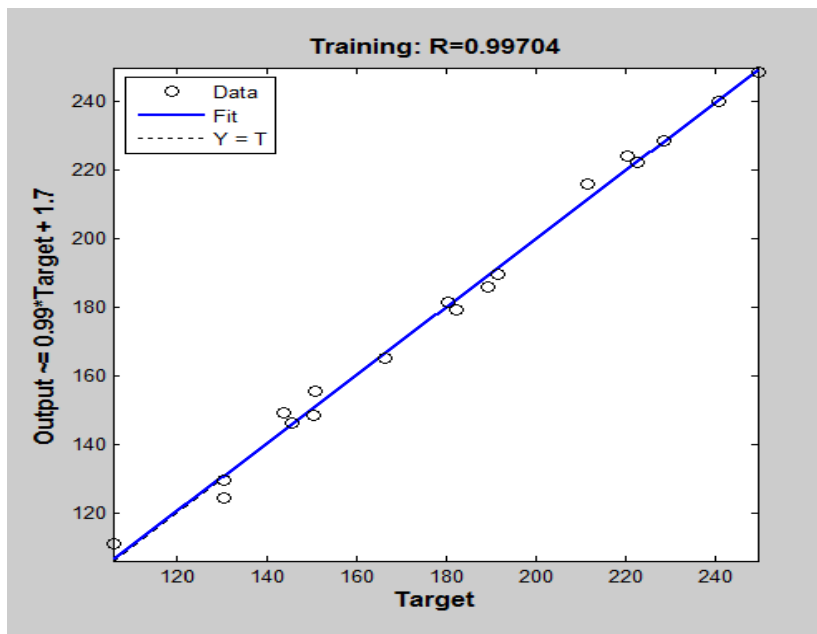


Figure III.20: Régression d'entraînement des réseaux de neurones pour Fc_{ANN3} .

Le résultat du test de régression du modèle obtenu par les réseaux de neurones nous donne un apprentissage égal à 99,704%. Cette valeur indique que 99,704% de la variance simple de l'effort Fc s'explique par les variables indépendantes, et 0,296% reste inexpliqués.

III.2.2.2 Utilisation de 5 couches cachées :

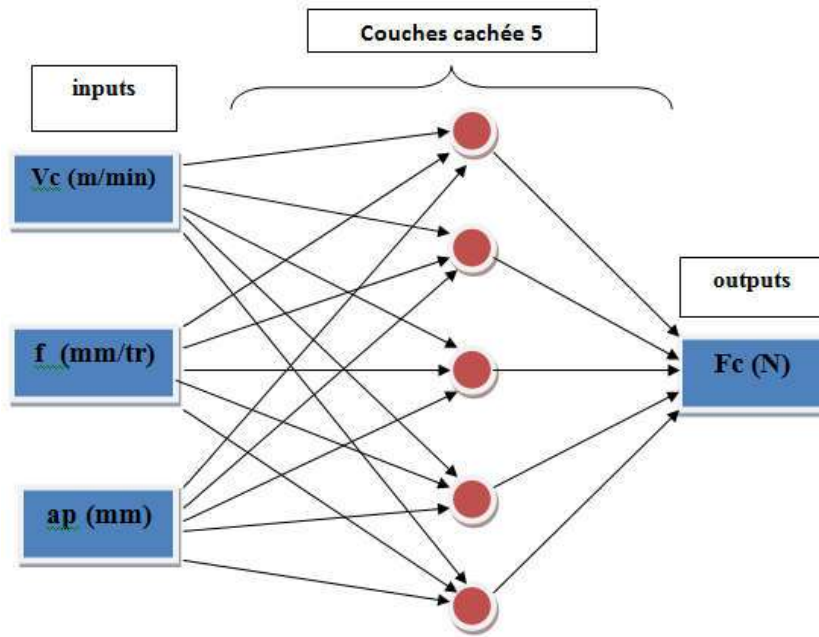


Figure III.21: Diagrammes optimum pour la modélisation de F_c ANN5.

Tableau III.7: Performance des structures ANN du modèle de F_c avec 5 couches cachées.

Apprentissage		Validation	
F_c (N)	Mesures	F_c (N)	Mesures
R^2	0.99978001	R^2	0.99942008
Racine de l'erreur quadratique moyenne (RMSE)	2.80295668	Racine de l'erreur quadratique moyenne (RMSE)	2.09725781
Ecart absolu moyen	1.92886667	Ecart absolu moyen	1.33788889
Somme des carrés des écarts (SSE)	141.41819	Somme des carrés des écarts (SSE)	39.586413
Somme des fréquences	18	Somme des fréquences	9

Après avoir assemblé et arrangé le programme, nous commençons le processus de formation:

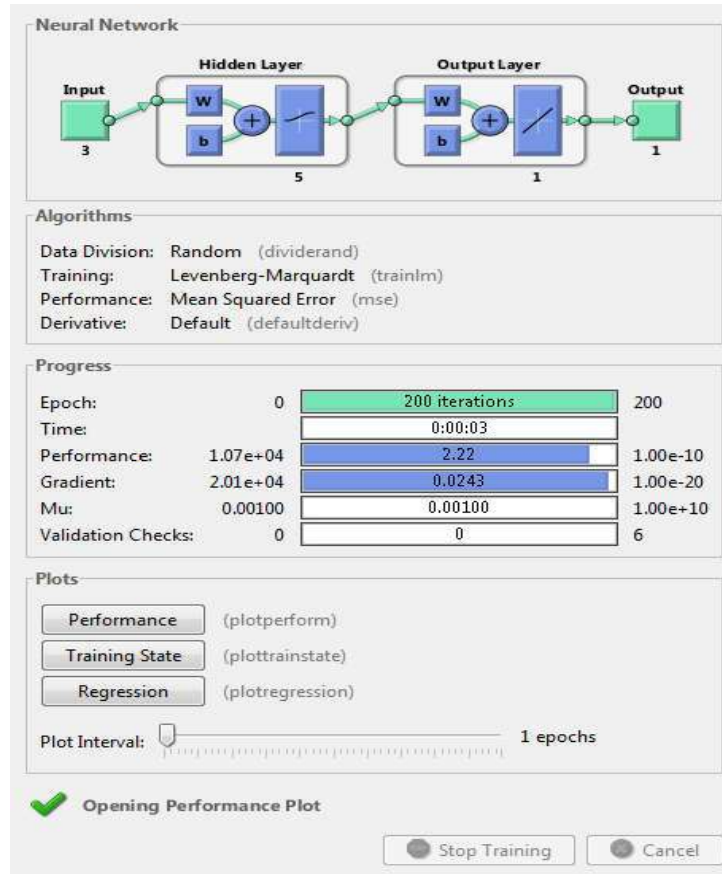


Figure III.22: Le processus de formation dans le réseau neuronal pour Fc_{ANN5} .

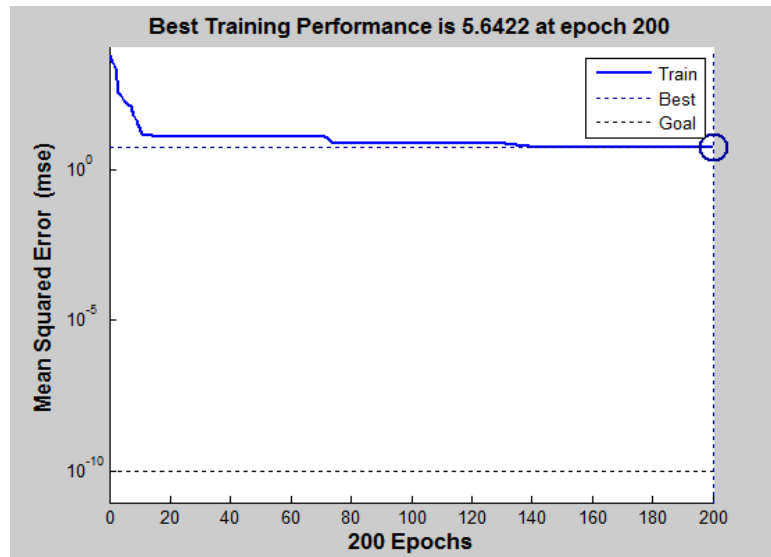


Figure III.23: Courbe de test des performances du système pour Fc_{ANN5} .

La figure montre la courbe d'erreur de performance du système. On peut dire que l'erreur quadratique moyenne de la courbe qui vaut 5,6422 à 200 itérations.

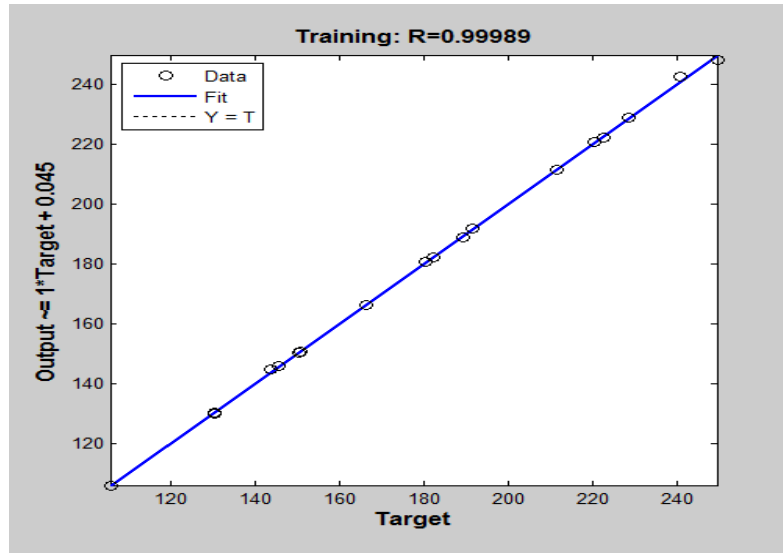


Figure III.24: Régression d'entraînement des réseaux de neurones pour F_c_{ANN5} .

Le résultat du test de régression du modèle obtenu par les réseaux de neurones nous donne un apprentissage égal à 99,989%. Cette valeur indique que 99,989% de la variance simple de l'effort F_c s'explique par les variables indépendantes, et 0,011% reste inexpliqués.

III.2.2.3 Utilisation de 8 couches cachées :

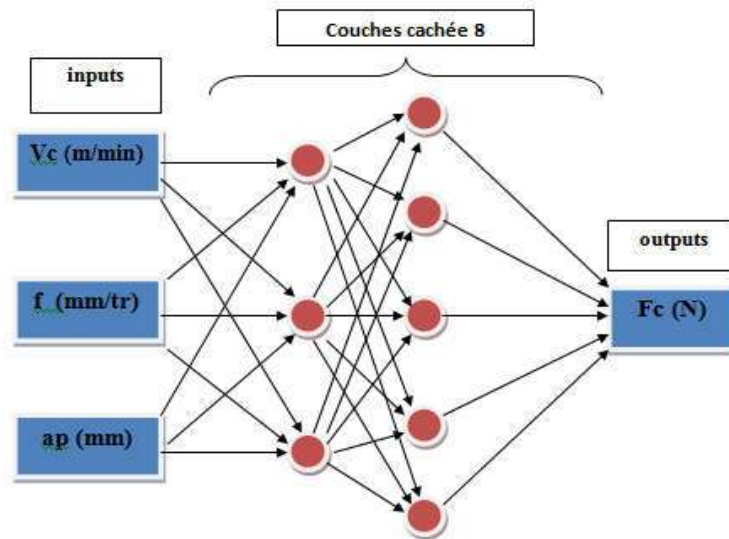


Figure III.25: Diagrammes optimum pour la modélisation de F_c_{ANN8} .

Tableau III.8: Performance des structures ANN du modèle de Fc avec 8 couches cachées.

Apprentissage		Validation	
Fc (N)	Mesures	Fc (N)	Mesures
R ²	0.9972419	R ²	0.99924014
Racine de l'erreur quadratique moyenne (RMSE)	1.39331003	Racine de l'erreur quadratique moyenne (RMSE)	4.2713E-05
Ecart absolu moyen	0.69713889	Ecart absolu moyen	0.00368889
Somme des carrés des écarts (SSE)	34.9436313	Somme des carrés des écarts (SSE)	0.00038442
Somme des fréquences	18	Somme des fréquences	9

Après avoir assemblé et arrangé le programme, nous commençons le processus de formation:

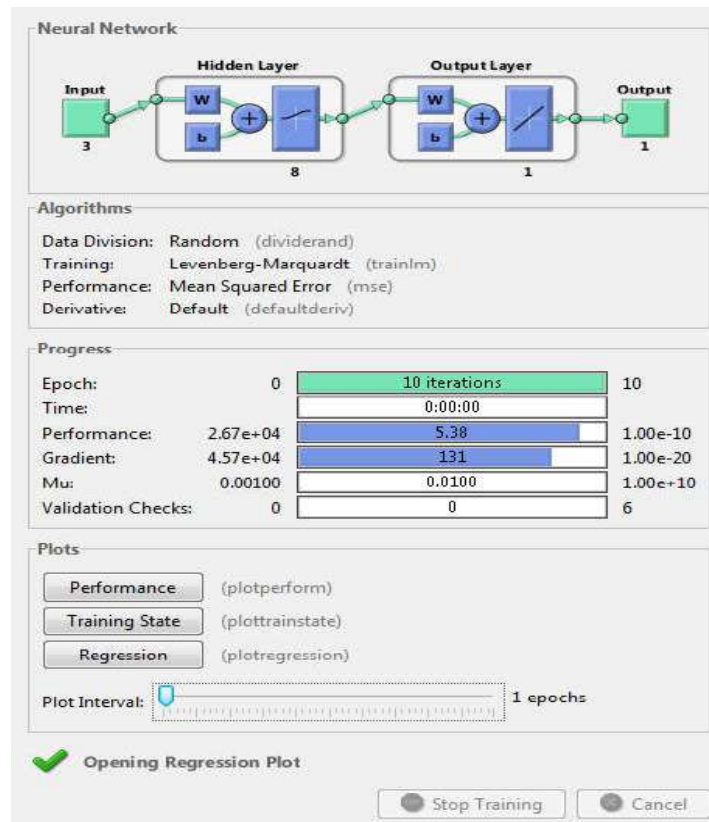


Figure III.26: Le processus de formation dans le réseau neuronal pour Fc_{ANN8}.

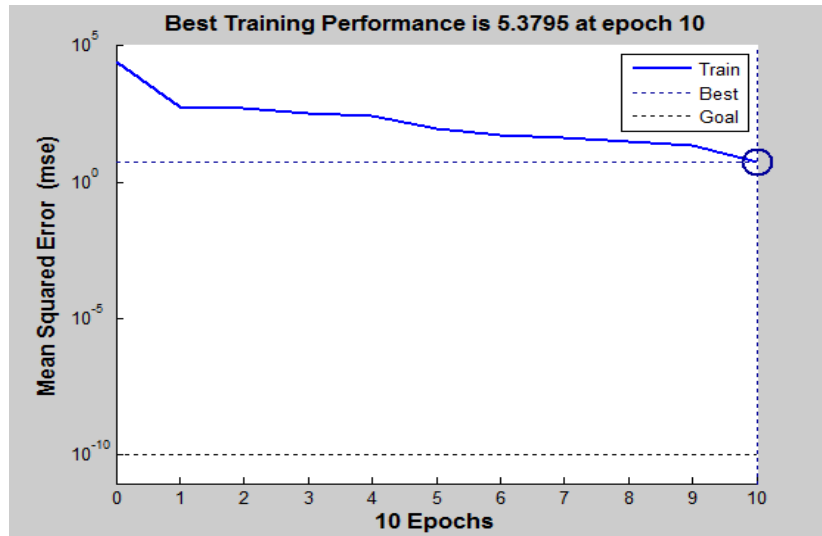


Figure III.27: Courbe de test des performances du système pour Fc_{ANN8} .

La figure montre la courbe d'erreur de performance du système. On peut dire que l'erreur quadratique moyenne de la courbe qui vaut 5,3795 à 10 itérations.

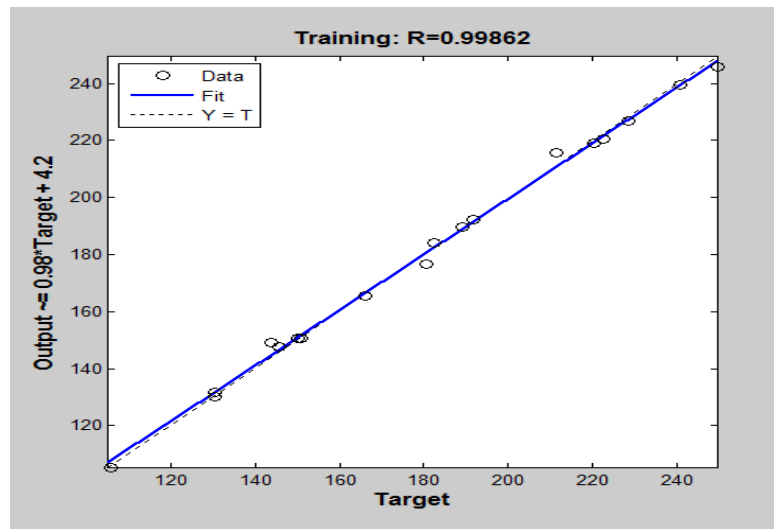


Figure III.28: Régression d'entraînement des réseaux de neurones pour Fc_{ANN8} .

Le résultat du test de régression du modèle obtenu par les réseaux de neurones nous donne un apprentissage égal à 99,862%. Cette valeur indique que 99,862% de la variance simple de l'effort Fc s'explique par les variables indépendantes, et 0,138% reste inexpliqués.

III.2.2.4 Comparaison les résultats obtenus pour Fc :

Le tableau suivant montre la comparaison entre les résultats obtenus en utilisant 3, 5 et 8 couches cachées.

Tableau III.9: Comparaison entre les résultats expérimentaux et simulés.

Essais	Fc _{ANN3}	Fc _{ANN5}	Fc _{ANN8}	Fc _{exp}	EFc _{ANN3}	EFc _{ANN5}	EFc _{ANN8}
1	137.2061	143.83	146.6882	143.83	6.6239	0	2.8582
2	185.5603	185.1019	189.3493	189.204	3.6437	4.1021	0.1453
3	219.5504	228.7963	228.457	228.5	8.9496	0.2963	0.043
4	137.2061	129.2416	127.5461	130.263	6.9431	1.0214	2.7169
5	185.5604	181.9935	181.9698	182.236	3.3244	0.2425	0.2662
6	209.1799	211.523	211.518	211.523	2.3431	0	0.005
7	150.219	150.219	150.1089	150.219	0	0	0.1101
8	221.9643	222.9448	220.5732	220.449	1.5153	2.4958	0.1242
9	245.5713	245.382	249.5428	249.539	3.9677	4.157	0.0038
10	130.424	134.8333	127.304	130.424	0	4.4093	3.12
11	155.8426	158.2514	150.755	150.754	5.0886	7.4974	0.001
12	191.1716	181.4384	180.5588	180.504	10.6676	0.9344	0.0548
13	108.1638	107.0421	108.8237	105.736	2.4278	1.3061	3.0877
14	155.3541	164.633	166.2115	166.213	10.8589	1.58	0.0015
15	198.2657	191.3756	191.6239	191.632	6.6337	0.2564	0.0081
16	147.007	146.1464	145.6335	145.634	1.373	0.5124	0.0005
17	218.5864	220.4073	222.5309	222.531	3.9446	2.1237	0.0001
18	242.3822	244.6168	240.8299	240.832	1.5502	3.7848	0.0021
19	105.3178	104.5153	106.7635	106.75	1.4322	2.2347	0.0135
20	154.4224	155.0872	157.744	157.744	3.3216	2.6568	0
21	185.1885	186.566	186.5682	186.566	1.3775	0	0.0022
22	116.9026	117.4001	116.9171	116.918	0.0154	0.4821	0.0009
23	165.2684	161.2	161.2017	161.2	4.0684	0	0.0017
24	193.5705	181.9895	187.1335	187.133	6.4375	5.1435	0.0005
25	130.5139	131.3675	131.7541	131.768	1.2541	0.4005	0.0139
26	191.3382	190.3914	190.7879	190.788	0.5502	0.3966	0.0001
27	239.1595	239.2288	238.5024	238.502	0.6575	0.7268	0.0004
Somme					98.9696	46.7606	12.5817
Moyenne					3.66554074	1.73187407	0.46598889

La comparaison entre les valeurs expérimentales avec les valeurs simulées est exprimée par la figure suivante :

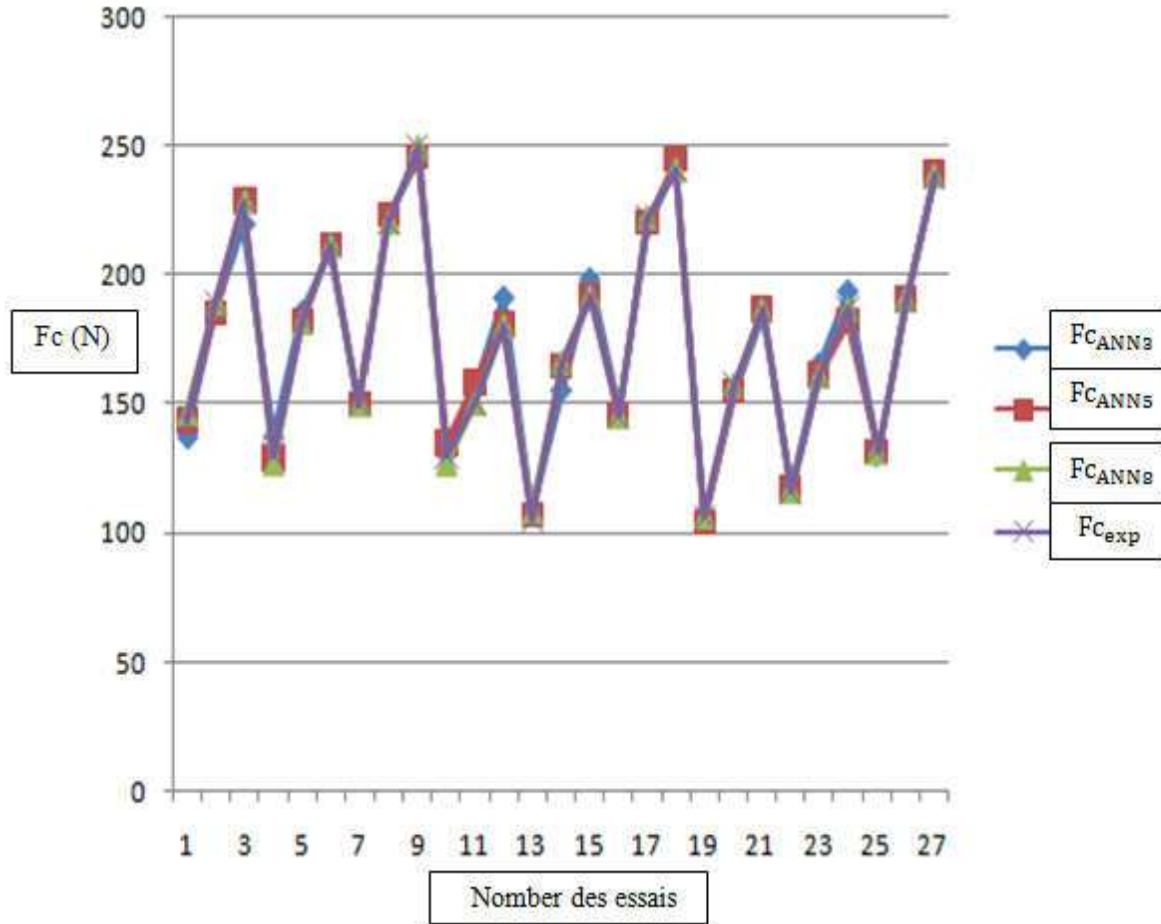


Figure III.29: Comparaison entre les F_{c_exp} et F_c simulées.

Nous notons qu’il existe une nette différence entre les résultats expérimentaux de F_{c_exp} et les résultats des réseaux de neurones F_{c_ANN3} et F_{c_ANN5} , et nous notons qu’il y a une bonne correspondance entre les résultats expérimentaux de F_{c_exp} et les résultats de la F_{c_ANN8} expérimentale.

L’erreur obtenue entre les valeurs expérimentales et les valeurs simulées est présentée par la figure suivante :

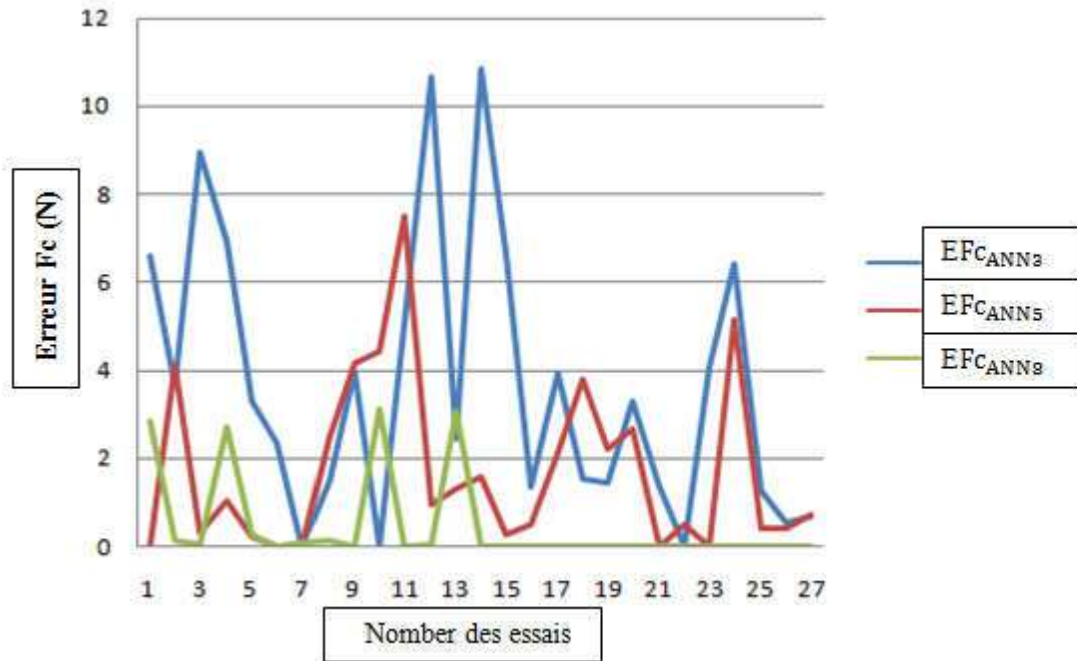


Figure III.30: Courbe de comparaison entre des EFC_{ANN3}, EFC_{ANN5} et EFC_{ANN8}

III.2.3 Analyse d'un ANN pour l'effort Fp :

III.2.3.1 Utilisation de 3 couches cachées :

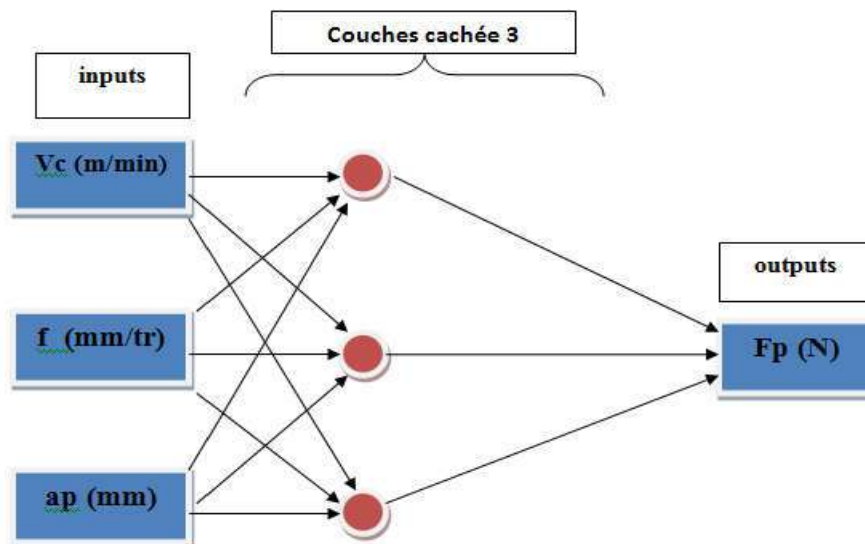


Figure III.31: Diagramme optimale pour la modélisation de Fp_{ANN3}.

Tableau III.10: Performance des structures ANN du modèle de Fp avec 3 couches cachées.

Apprentissage		Validation	
Fp (N)	Mesures	Fp (N)	Mesures
R ²	0.99265354	R ²	0.99784117
Racine de l'erreur quadratique moyenne (RMSE)	5.15857095	Racine de l'erreur quadratique moyenne (RMSE)	4.67725135
Ecart absolu moyen	4.20255		1.54912222
Somme des carrés des écarts (SSE)	478.995377	Somme des carrés des écarts (SSE)	42.0952621
Somme des fréquences	18	Somme des fréquences	9

Après avoir assemblé et arrangé le programme, nous commençons le processus de formation:

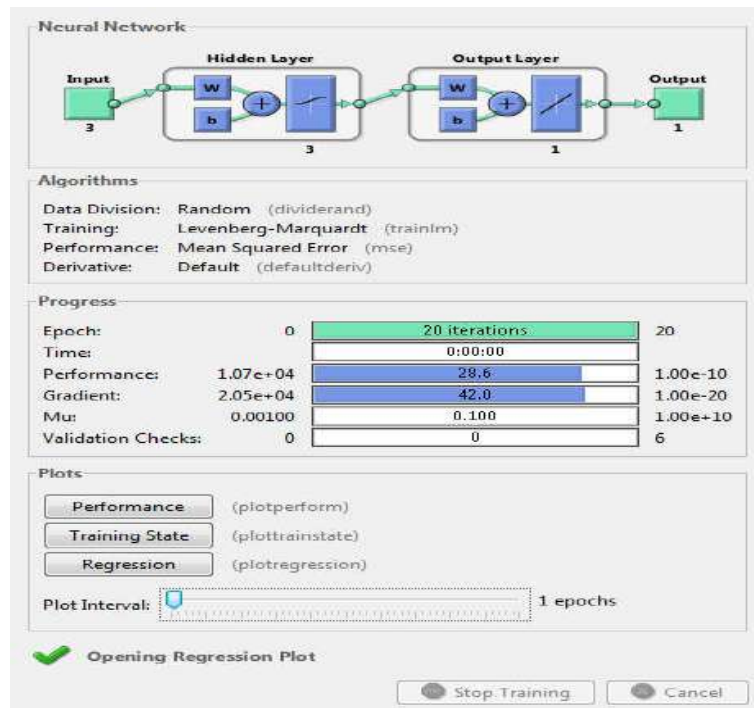


Figure III.32: Le processus de formation dans le réseau neuronal pour Fp ANN3.

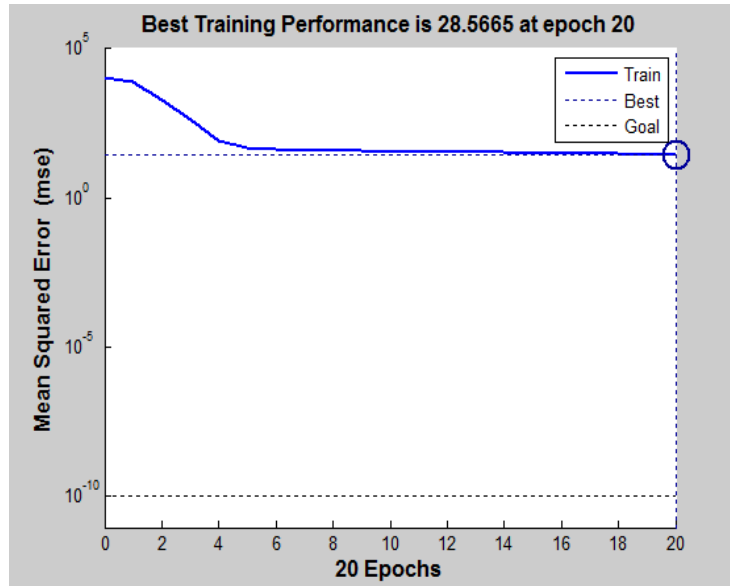


Figure III.33: Courbe de test des performances du système pour Fp_{ANN3} .

La figure montre la courbe d'erreur de performance du système. On peut dire que l'erreur quadratique moyenne de la courbe qui vaut 28,5665 à 20 itérations.

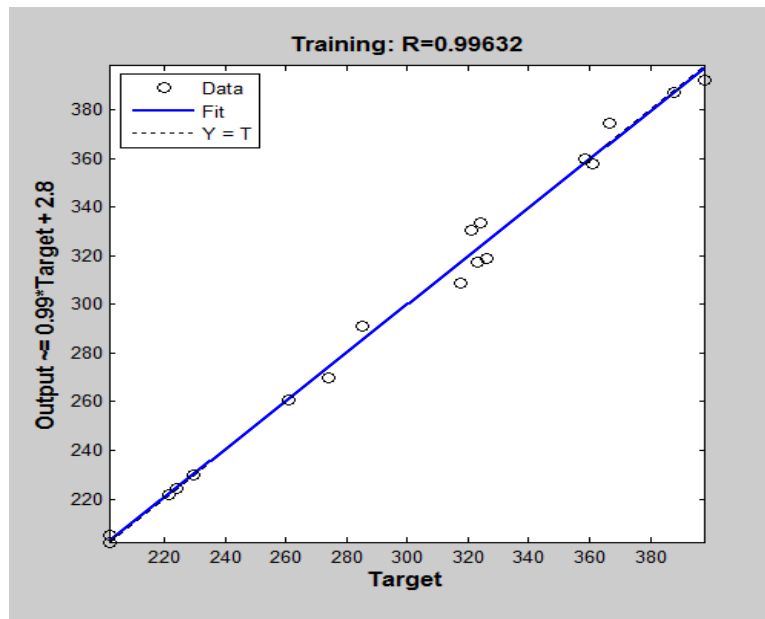


Figure III.34: Régression d'entraînement des réseaux de neurones pour Fp_{ANN3} .

Le résultat du test de régression du modèle obtenu par les réseaux de neurones nous donne un apprentissage égal à 99,632%. Cette valeur indique que 99,632% de la variance simple de l'effort F_p s'explique par les variables indépendantes, et 0,3268% reste inexpliqués.

III.2.3.2 Utilisation de 5 couches cachées :

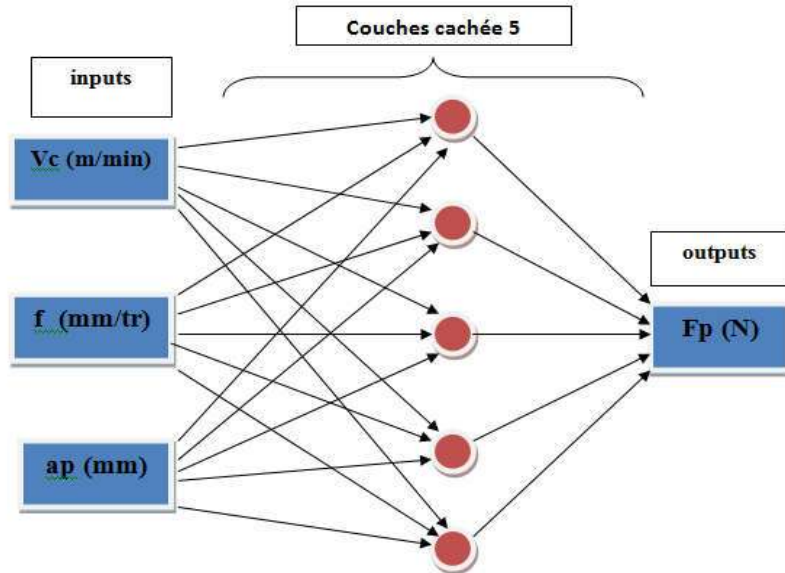


Figure III.35: Diagramme optimale pour la modélisation de F_p ANN5.

Tableau III.11: Performance des structures ANN du modèle de F_p avec 5 couches cachées.

Apprentissage		Validation	
Fp (N)	Mesures	Fp (N)	Mesures
R ²	0.99854053	R ²	0.99962004
Racine de l'erreur quadratique moyenne (RMSE)	2.02868136	Racine de l'erreur quadratique moyenne (RMSE)	3.28624786
Ecart absolu moyen	1.51130556	Ecart absolu moyen	2.72002222
Somme des carrés des écarts (SSE)	74.0798654	Somme des carrés des écarts (SSE)	97.1948249
Somme des fréquences	18	Somme des fréquences	9

Après avoir assemblé et arrangé le programme, nous commençons le processus de formation:

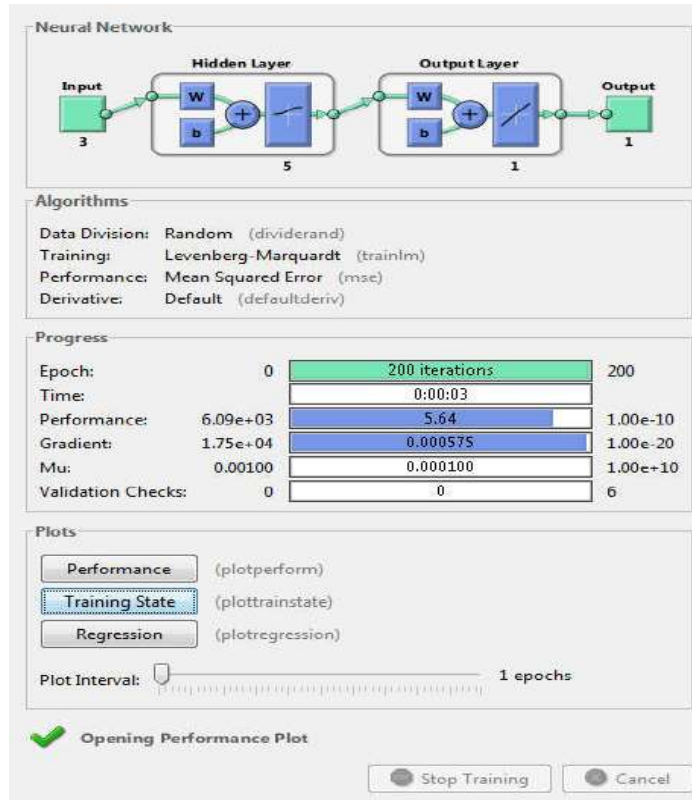


Figure III.36: Le processus de formation dans le réseau neuronal pour Fp_{ANN5} .

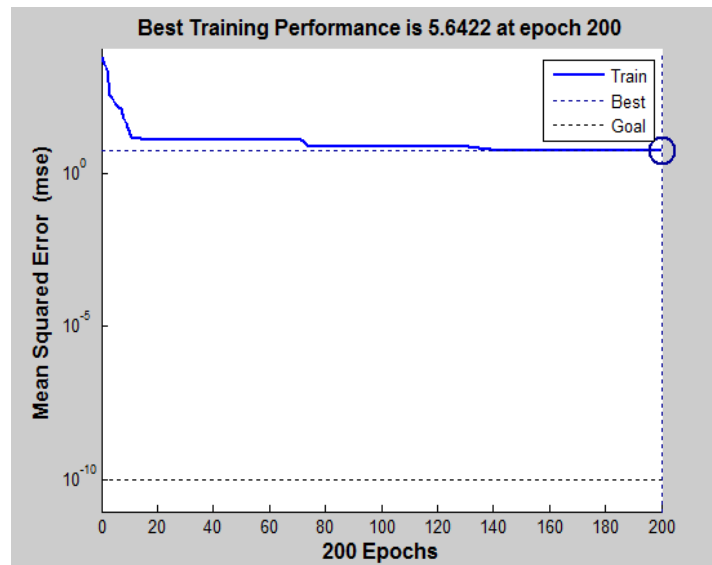


Figure III.37: Courbe de test des performances du système pour Fp_{ANN5} .

La figure montre la courbe d'erreur de performance du système. On peut dire que l'erreur quadratique moyenne de la courbe qui vaut 5,6422 à 200 itérations.

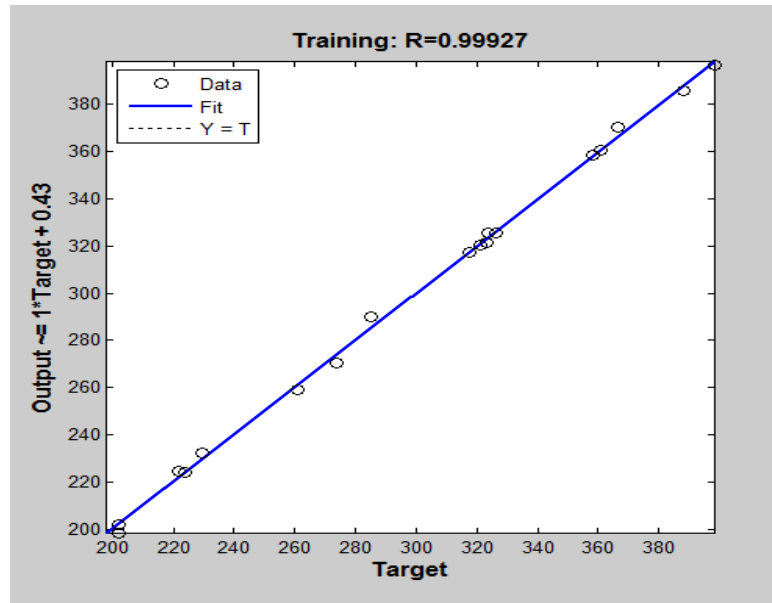


Figure III.38: Régression d'entraînement des réseaux de neurones pour Fp_{ANN5} .

Le résultat du test de régression du modèle obtenu par les réseaux de neurones nous donne un apprentissage égal à 99,927%. Cette valeur indique que 99,927% de la variance simple de l'effort Fp s'explique par les variables indépendantes, et 0,073% reste inexpliqués.

III.2.3.3 Utilisation de 8 couches cachées :

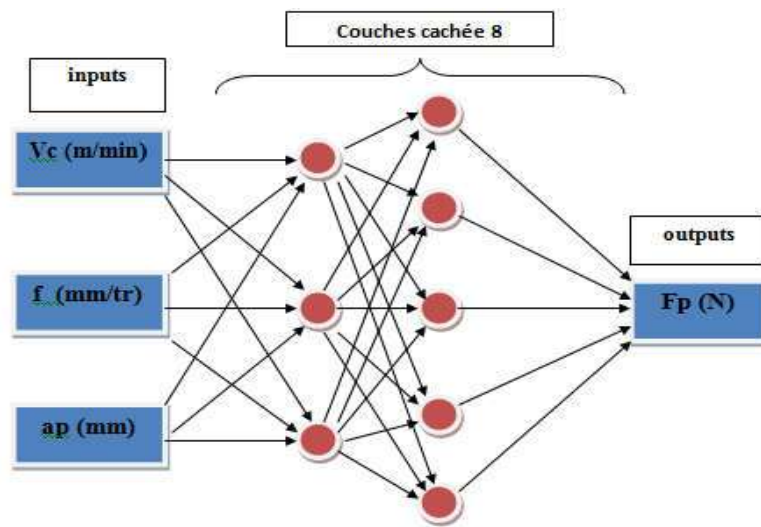


Figure III.39: Diagramme optimale pour la modélisation de Fp_{ANN8} .

Tableau III.12: Performance des structures ANN du modèle de Fp avec 8 couches cachées.

Apprentissage		Validation	
Fp (N)	Mesures	Fp (N)	Mesures
R ²	0.99994	R ²	0.99998
Racine de l'erreur quadratique moyenne (RMSE)	0.28657264	Racine de l'erreur quadratique moyenne (RMSE)	2.44362021
Ecart absolu moyen	0.12456667	Ecart absolu moyen	0.83485556
Somme des carrés des écarts (SSE)	1.47822984	Somme des carrés des écarts (SSE)	21.9925819
Somme des fréquences	18	Somme des fréquences	9

Après avoir assemblé et arrangé le programme, nous commençons le processus de formation:

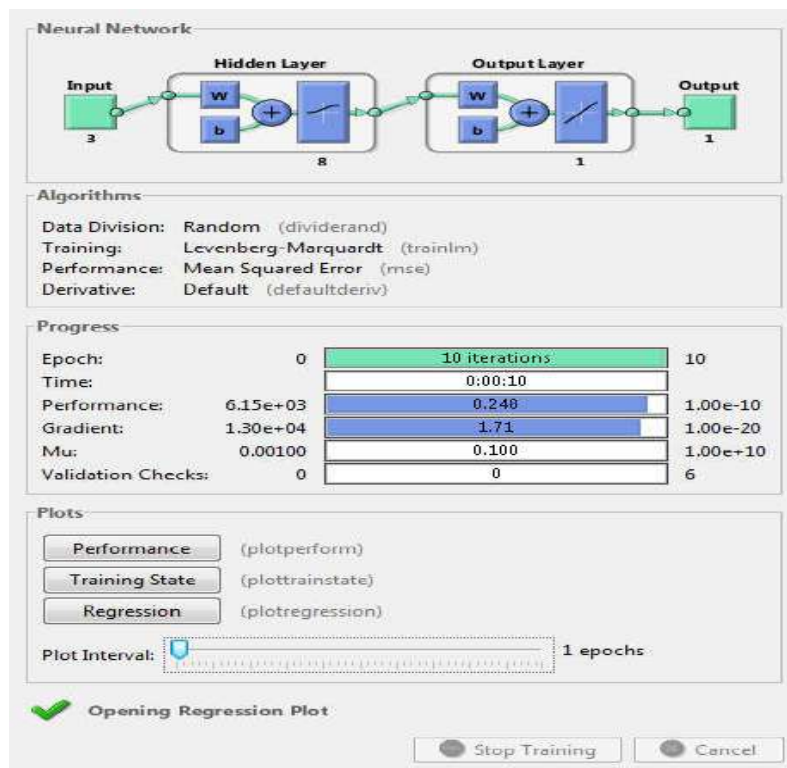


Figure III.40: Le processus de formation dans le réseau neuronal pour Fp_{ANN8}.

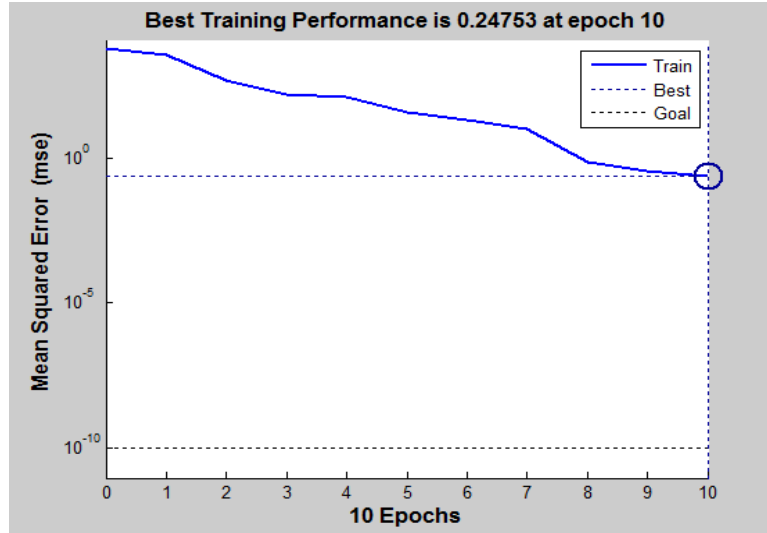


Figure III.41: Courbe de test des performances du système pour Fp_{ANN8} .

La figure montre la courbe d'erreur de performance du système. On peut dire que l'erreur quadratique moyenne de la courbe qui vaut 0,24753 à 10 itérations.

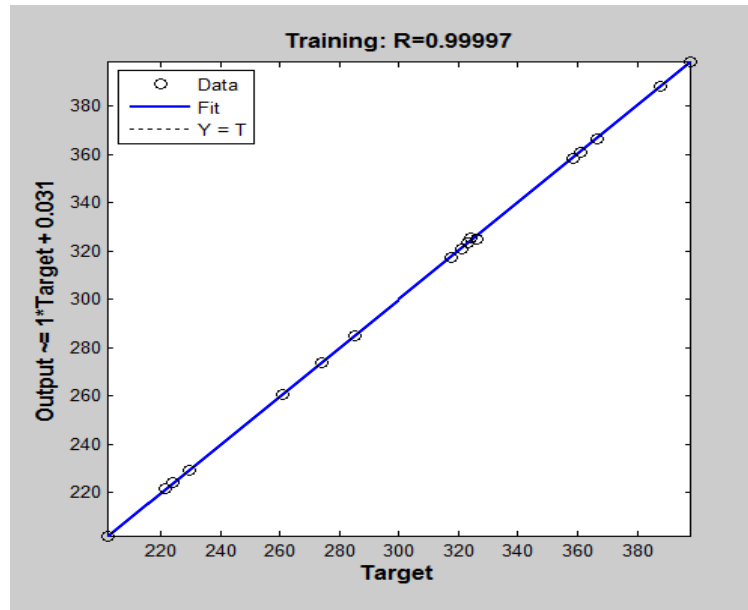


Figure III.42: Régression d'entraînement des réseaux de neurones pour Fp_{ANN8} .

Le résultat du test de régression du modèle obtenu par les réseaux de neurones nous donne un apprentissage égal à 99,997%. Cette valeur indique que 99,997% de la variance simple de l'effort Fp s'explique par les variables indépendantes, et 0,003% reste inexplicés.

III.2.3.4 Comparaison les résultats obtenus pour F_p :

Le tableau suivant montre la comparaison entre les résultats obtenus en utilisant 3, 5 et 8 couches cachées.

Tableau III.13: Comparaison entre les résultats expérimentaux et simulés.

Essais	$F_{p_{ANN3}}$	$F_{p_{ANN5}}$	$F_{p_{ANN8}}$	$F_{p_{exp}}$	$EF_{p_{ANN3}}$	$EF_{p_{ANN5}}$	$EF_{p_{ANN8}}$
1	222.2475	226.3268	224.0932	224.093	1.8455	2.2338	0.0002
2	312.8082	315.3703	317.25	317.466	4.6578	2.0957	0.216
3	353.7094	360.1296	359.339	358.426	4.7166	1.7036	0.913
4	234.5073	229.2888	229.5116	229.514	4.9933	0.2252	0.0024
5	312.8432	320.9328	321.0599	321.02	8.1768	0.0872	0.0399
6	365.6571	365.6893	365.6978	366.433	0.7759	0.7437	0.7352
7	259.8929	260.2015	260.7727	260.777	0.8841	0.5755	0.0043
8	361.4412	360.844	360.8768	360.844	0.5972	0	0.0328
9	400.8704	398.188	398.1859	398.188	2.6824	0	0.0021
10	193.9615	200.0663	202.2105	202.22	8.2585	2.1537	0.0095
11	273.7201	277.9382	273.9315	273.934	0.2139	4.0042	0.0025
12	321.9415	321.6437	326.34	326.336	4.3945	4.6923	0.004
13	203.7738	202.7836	202.2895	202.057	1.7168	0.7266	0.2325
14	276.8198	283.2535	285.1465	285.146	8.3262	1.8925	0.0005
15	334.7597	327.2252	323.9714	323.977	10.7827	3.2482	0.0056
16	224.934	220.6995	221.5718	221.56	3.374	0.8605	0.0118
17	327.3186	324.8956	323.2831	323.267	4.0516	1.6286	0.0161
18	382.7229	387.5888	387.9072	387.921	5.1981	0.3322	0.0138
19	171.0254	171.002	172.6924	169.977	1.0484	1.025	2.7154
20	232.8474	232.3727	227.4856	227.431	5.4164	4.9417	0.0546
21	281.0919	277.6935	279.8301	279.833	1.2589	2.1395	0.0029
22	180.3652	178.1565	177.3943	181.133	0.7678	2.9765	3.7387
23	248.62	243.8581	249.9483	249.986	1.366	6.1279	0.0377
24	289.1221	292.7293	289.213	289.21	0.0879	3.5193	0.003
25	192.0137	194.9729	192.9506	192.162	0.1483	2.8109	0.7886
26	273.5538	275.3736	275.8616	275.781	2.2272	0.4074	0.0806
27	347.0468	348.136	348.5758	348.668	1.6212	0.532	0.0922
Somme					89.588	51.6837	9.7559
Moyenne					3.31807407	1.91421111	0.36132963

La comparaison entre les valeurs expérimentales avec les valeurs simulées est exprimée par la figure suivante :

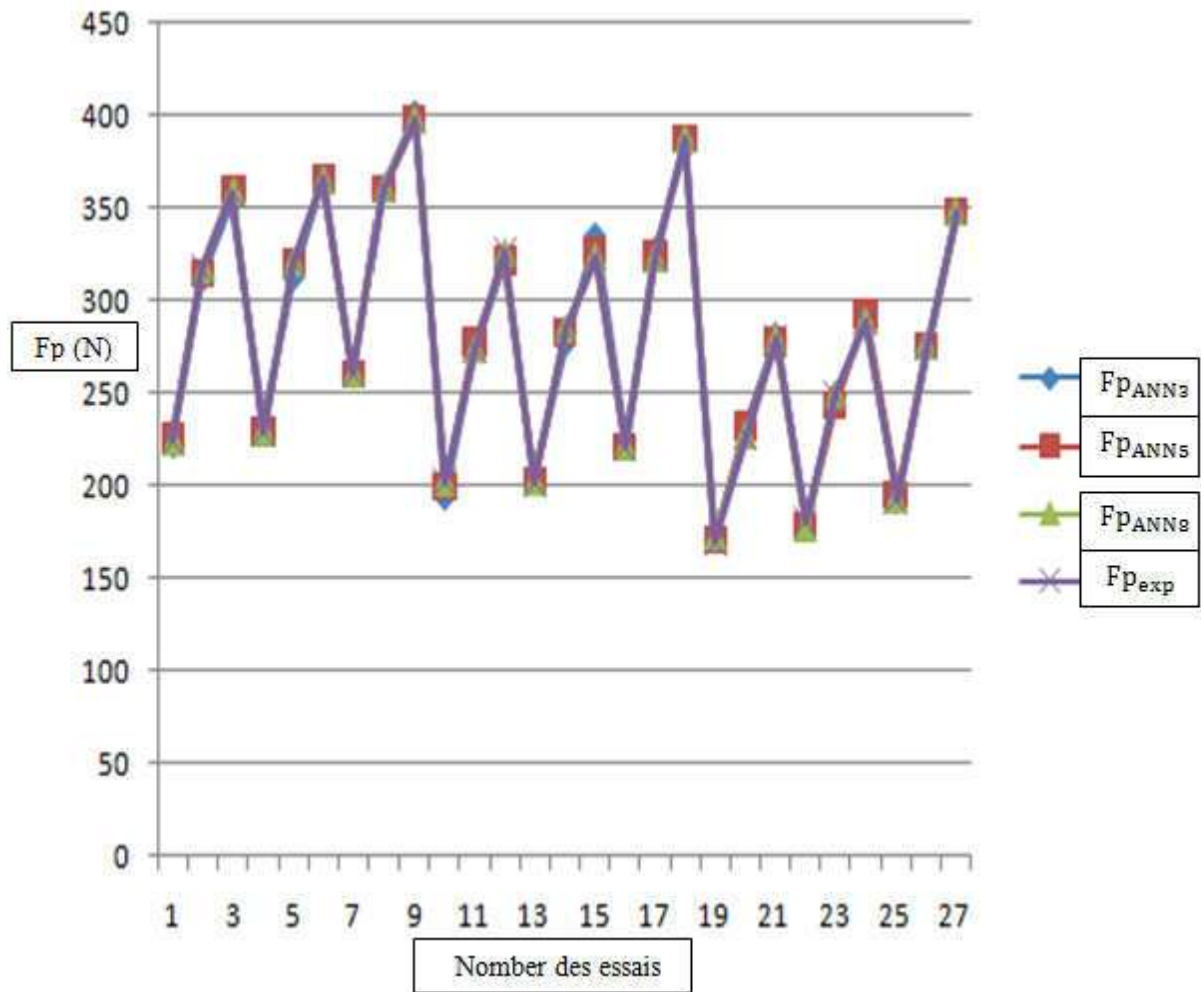


Figure III.43: Comparaison entre les F_{p_exp} et F_p simulées.

Nous notons qu’il existe une nette différence entre les résultats expérimentaux de F_{p_exp} et les résultats des réseaux de neurones F_{p_ANN3} et F_{p_ANN5} , et nous notons qu’il y a une bonne correspondance entre les résultats expérimentaux de F_{p_exp} et les résultats de la F_{p_ANN8} expérimentale

L’erreur obtenue entre les valeurs expérimentales et les valeurs simulées est présentée par la figure suivante :

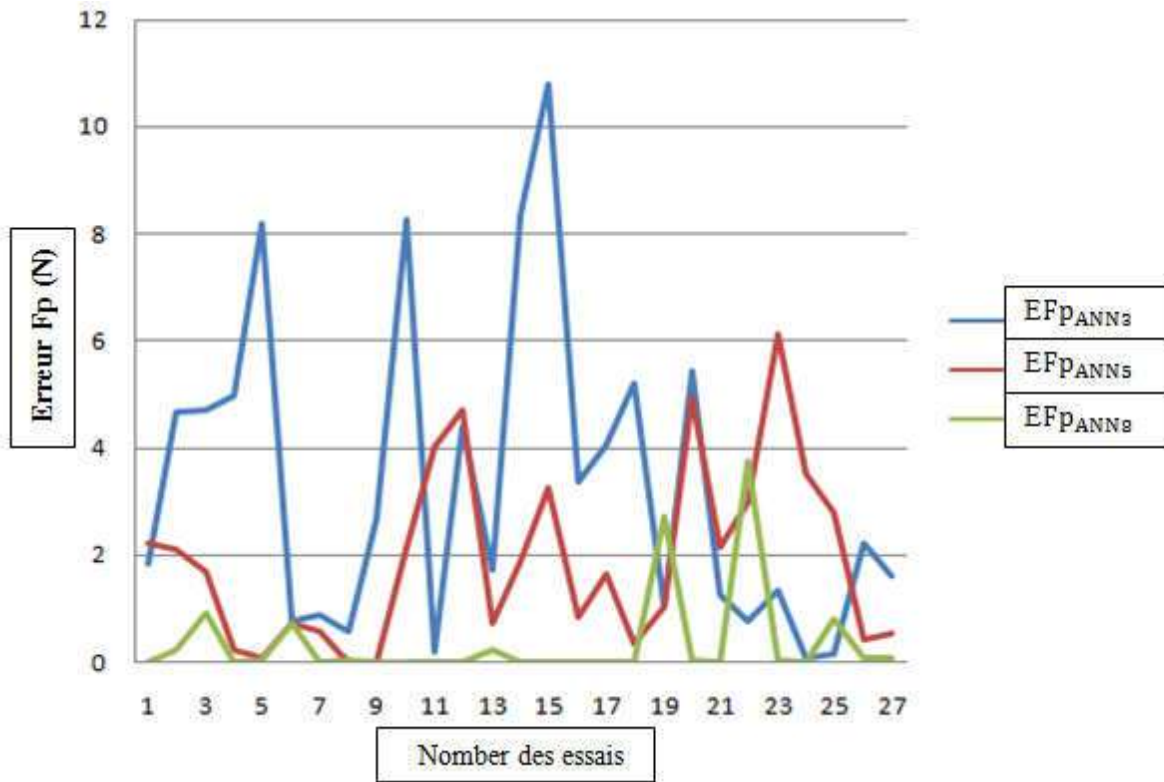


Figure III.44: Courbe de comparaison entre les erreurs de Fp simulées.

CONCLUSION GENERALE

Dans le présent mémoire, nous avons procédé à l'étude des efforts de coupe pendant le tournage de l'acier 20Mn Cr5 par l'utilisation de la méthode des réseaux de neurones ANN,

Cette étude nous a permis de faire une comparaison entre les résultats expérimentaux et celles obtenus avec la méthode des réseaux de neurones en appliquant trois types de couches cachées ; 3, 5 et 8.

Vu les résultats obtenus, on peut dire que les modèles obtenus avec un nombre de couches cachées égale à 8 sont les meilleures.

On conclut que les résultats obtenus en simulant les efforts de coupe par la méthode des réseaux de neurones comme méthode de vérification coïncident bien avec les résultats expérimentaux, donc elle peut être utilisée comme méthode alternative pour étudier les efforts de coupe en tournage de l'acier 20Mn Cr5.

LES REFERENCES

- [1] : R. Berghida, Impact de la signature des outils de coupe sur la variation de la géométrie des pièces mécaniques, Mémoire de magistère, Université de Constantine, 2006.
- [2] : Souhaila Boulahmi, Etude expérimentale Des effets de l'outil de coupe, des conditions d'usinage de lubrification sur la qualité de surface et le comportement en traction des pièces en alliages D'Aluminium, mémoire, l'école de Technologie supérieure.
- [3] : Techniques de l'ingénieur JEAN HENG, 2002 (Pratique de la maintenance préventive (Mécanique. Pneumatique. Hydraulique. Electricité. Froid), Dunod, Paris.
- [4] : H. Iotfi, Prédiction de la rugosité de surface lors de l'usinage de l'acier inoxydable martensitique X20Cr13 en utilisant les réseaux de neurones artificiels, la logique floue et le modèle 2015/2016.
- [5] : R. BUTIN, M. PINOT, (fabrication mécaniques tome 2), les conditions FOUCHER, Paris. de regression . 2015 / 2016.
- [6] : R. Fethi .L'effet d'usinage par tournage sur un acier XC18 . 16/10/2016.
- [7] : Selaimia M., Etude des efforts de coupe et de la rugosité lors de l'usinage du PE80 <http://www.memoireonline.com>
- [8] : Guillaume .F : Développement d'un modèle généralisé d'efforts de coupe pour l'usinage de l'alliage de titane Ti6Al4V . le 15 Novembre 2016
- [9] : N. Athamena, «Reconnaitances de l'écriture manuscrite », Thèse de doctorat. Université de Batna.
- [10] : G. Dreyfus, J. M. Martinez, M. Samuelides, M. B. Gordon, F. Badran, S. Thiria, L. Hérault, 'Réseaux de neurones: méthodologie et applications', Eyrolles, France, 2004.
- [11] : G. Dreyfus, 'Les réseaux de neurones', Mécanique Industrielle et Matériaux, N° 5, Septembre, 1998.
- [12]: W. McCulloch and W. Pitts, 'A logical calculus of the ideas immanent in nervous activity', Bulletin of Mathematical Biophysics, 5: 115-133, 1943.
- [13]: F. Rosenblatt, 'The Perceptron: a probabilistic model for information storage and organization in the brain', Psychological Review, 65: 386-408, 1958.
- [14]: M. Minsky, S. Papert, 'Perceptrons', Expanded Edition. MIT Press, 1988.

- [15]: J. J. Hopfield, 'Neural networks and physical systems with emergent collective computational abilities', *Proceedings of the National Academy of Sciences*, 79: 2554-2558, 1982.
- [16]: D. Rumelhart, G. Hinton, R. Williams, 'Learning internal representations by error propagation', *Parallel Distributed Processing*, Vol. 1, D. Rumelhart and J. McClelland Eds, Cambridge: MIT Press, pp. 318-362, 1986.
- [17]: G. DREYFUS . *Mécanique Industrielle et Matériaux*, Les réseaux de neurones, de la Ville de Paris (ESPCI), N°51 (septembre 1998).
- [18]: C. Bishop, 'Neural Networks for Pattern Recognition', Oxford University Press, 1995.
- [19]: W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, 'Numerical Recipes in C: the Art of Computing', Cambridge University Press, 1992.
- [20]: O. Nerrand, P. Roussel-Ragot, L. Personnaz, G. Dreyfus, 'Neural networks and nonlinear adaptive filtering: unifying concepts and new algorithms', *Neural Computation*, Vol.5, pp. 165-199, 1993.
- [21]: W. Chen, P. Garder, 'Word level discriminative training for handwritten word recognition', *Proc. 7th IWFHR*, pp. 393-402, Amsterdam, 2000.
- [22]: H. Bunke, M. Roth, E. G. Schukat-Talamazzini, 'Off-line cursive handwriting recognition using hidden markov models', *Pattern Recognition*, Vol. 28, N° 9, pp. 1399-1413, 1995.
- [23]: H. A. Beghdadi, M. Senouci, 'Réseaux de neurones : théorie et pratique', Office des Publications Universitaires, Algérie, 2005.
- [24]: K. Hornik, M. Stinchcombe, H. White, P. Auer, 'Degree of approximation results for feedforward networks approximating unknown mappings and their derivatives', *Neural Computation*, Vol. 6, pp. 1262-1275, 1994.
- [25]: Bentaleb Fayçal, Contribution à l'étude de la dégradation de la zone de coupe lors de l'évolution de la contrainte de l'outil sur la matière, Thèse de doctorat, Université de Constantine, 2017.

Résumé

La qualité et la précision des appareils mécaniques sont devenues un objectif majeur dans le processus de fabrication, le bon choix des outils et des informations de coupe est essentiel pour atteindre les objectifs souhaités.

Dans ce mémoire, on a fait une modélisation des efforts de coupe par la méthode des réseaux de neurones avec l'utilisation des trois entrées (vitesse de coupe, avance et profondeur de passe) avec trois sorties (Fa, Fc et Fp) on utilisons trois types de couches cachées (3, 5 et 8) avec un nombre d'itérations.

Après comparaison des écarts types et des résidus, nous avons constaté que les meilleurs résultats sont celles obtenus avec un nombre de couches cachées égale à huit.

Mots-clés: Modélisation, Efforts de coupe, Réseaux de neurones et couches cachées.

الملخص:

أصبحت جودة و دقة الأجهزة الميكانيكية هدف رئيسا في التصنيع, و يعد الاختيار الصحيح للأدوات و المعلومات القطع أمرا ضروريا لتحقيق الأهداف المرجوة.

في هذه المذكرة ، قمنا بنمذجة قوى القطع باستخدام طريقة الشبكة العصبية باستخدام ثلاثة مدخلات (سرعة القطع ، التغذية وعمق القطع) بثلاثة مخرجات (Fp, Fc, Fa) ثلاثة أنواع من الطبقات المخفية (3 ، 5 و 8) بعدد من التكرارات.

بعد مقارنة الانحرافات المعيارية والمخلفات، وجدنا أن أفضل النتائج هي تلك التي تم الحصول عليها بعدد من الطبقات المخفية يساوي ثمانية.

الكلمات المفتاحية: النمذجة و قوى القطع و الشبكات العصبية و الطبقات الخفية

Abstract:

The quality and accuracy of mechanical devices have become a major goal in the manufacturing process, the right choice of tools and cutting information is essential to achieve desired goals.

In this thesis, we modeled the cutting forces using the neural network method with the use of three inputs (cutting speed, feed and depth of cut) with three outputs (Fa, Fc and Fp). Three types of hidden layers (3, 5 and 8) with a number of iterations.

After comparing the standard deviations and the residuals, we found that the best results are those obtained with a number of hidden layers equal to eight.

Keywords: Modeling, Cutting forces, Neural networks and hidden layers.