

IDENTIFICATION DES PARAMÈTRES DU MOTEUR DE
L'AVION CESSNA CITATION X POUR LA PHASE DE
CROISIÈRE À PARTIR DES TESTS EN VOL ET À BASE
DES RÉSEAUX DE NEURONES

PAR
Mahdi ZAAG

MÉMOIRE PRÉSENTÉ À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
COMME EXIGENCE PARTIELLE À L'OBTENTION DE LA MAÎTRISE
AVEC MÉMOIRE EN GÉNIE DE LA PRODUCTION AUTOMATISÉE
M.Sc.A.

MONTRÉAL, LE 17 MARS 2017

ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

©Tous droits réservés

Cette licence signifie qu'il est interdit de reproduire, d'enregistrer ou de diffuser en tout ou en partie, le présent document. Le lecteur qui désire imprimer ou conserver sur un autre media une partie importante de ce document, doit obligatoirement en demander l'autorisation à l'auteur.

PRÉSENTATION DU JURY

CE MÉMOIRE A ÉTÉ ÉVALUÉ

PAR UN JURY COMPOSÉ DE :

Mme Ruxandra Botez, directrice de mémoire
Département de génie de la production automatisée à l'École de technologie supérieure

M. Christian Belleau, président du jury
Département de génie mécanique à l'École de technologie supérieure

M. Tony Wong, membre du jury
Département de génie de la production automatisée à l'École de technologie supérieure

IL A FAIT L'OBJET D'UNE SOUTENANCE DEVANT JURY ET PUBLIC

LE 16 MARS 2017

À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

REMERCIEMENTS

J'adresse mes sincères remerciements à madame Ruxandra Botez, ma directrice de recherche, pour son soutien et son aide tout le long du projet. Je la remercie également pour sa disponibilité, ses conseils judicieux qui m'ont permis d'accomplir ce travail.

Je tiens à exprimer ma gratitude aux membres de jury qui m'ont fait l'honneur de participer à l'évaluation de ce mémoire.

Je dédie ce modeste travail à mes chers parents en témoignage de ma reconnaissance pour leur amour, leur affectation ainsi que leurs sacrifices qu'ils ont consentis pour mon éducation et ma formation, à mes frères et ma fiancée pour leurs soutiens continus, à toute ma famille et à mes meilleurs amis, qu'ils gardent de bons souvenirs du bon temps qu'on a passé ensemble.

J'aimerais par la même occasion, remercier tous les étudiants et collègues du laboratoire de recherche en commande active, avionique et en aéro-servoélasticité LARCASE, et surtout mes collègues Georges Ghazi et Abdallah Ben Mosbah.

IDENTIFICATION DES PARAMÈTRES DU MOTEUR DE L'AVION CESSNA CITATION X POUR LA PHASE DE CROISIÈRE À PARTIR DES TESTS EN VOL ET À BASE DES RÉSEAUX DE NEURONES

Mahdi ZAAG

RÉSUMÉ

La disponibilité des modèles précis des avions est parmi les éléments clés permettant d'assurer leurs améliorations. Ces modèles servent à améliorer les commandes de vol et de concevoir de nouveaux systèmes aérodynamiques pour la conception des ailes déformables des avions.

Ce projet consiste à concevoir un système d'identification de certains paramètres du modèle du moteur de l'avion d'affaires américain Cessna Citation X pour la phase de croisière à partir des essais en vol. Ces essais ont été effectués sur le simulateur de vol conçu et fabriqué par CAE Inc. qui possède le niveau D de la dynamique de vol. En effet, le niveau D est le plus haut niveau de précision donné par l'autorité fédérale de réglementation FAA de l'aviation civile aux États-Unis.

Une méthodologie basée sur les réseaux de neurones optimisés à l'aide d'un algorithme intitulé le "grand déluge étendu" est utilisée dans la conception de ce système d'identification. Plusieurs tests de vol pour différentes altitudes et différents nombres de Mach ont été réalisés afin de s'en servir comme bases de données pour l'apprentissage des réseaux de neurones. La validation de ce modèle a été réalisée à l'aide des données du simulateur. Malgré la non-linéarité et la complexité du système, les paramètres du moteur ont été très bien prédits pour une enveloppe de vol déterminée. Ce modèle estimé pourrait être utilisé pour des analyses de fonctionnement du moteur et pourrait assurer le contrôle de l'avion pendant cette phase de croisière.

L'identification des paramètres du moteur pourrait être réalisée aussi pour les autres phases de montée et de descente afin d'obtenir son modèle complet pour toute l'enveloppe du vol de l'avion Cessna Citation X (montée, croisière, descente). Cette méthode employée dans ce

travail pourrait aussi être efficace pour réaliser un modèle pour l'identification des coefficients aérodynamiques du même avion à partir toujours des essais en vol.

Mots-clés : identification, modèle, essais en vol, réseaux de neurones, moteur, avion

CESSNA CITATION X ENGINE PARAMETRES IDENTIFICATION IN THE CRUISE REGIME FROM FLIGHT TESTS BASED ON NEURAL NETWORKS

Mahdi ZAAG

ABSTRACT

The availability of accurate aircraft models is one of the key elements in ensuring aircraft improvements. These models are used to improve flight controls and design new aerodynamic systems for the design of deformable aircraft wings.

This project consists of designing a method for identifying certain parameters of the engine model of the US Cessna Citation X business aircraft for the cruise phase from the flight tests. These tests were performed on the designed flight simulator manufactured by CAE Inc. which has flight dynamics D level. Level D is the highest level of accuracy given by the FAA Civil Aviation Authority in the United States.

A methodology based on optimized neural networks using an algorithm called the "Extended Great Deluge" is used in the design of this identification model. Several flight tests for different altitudes and Mach numbers were performed to serve as databases for learning neural networks. Model Validation was carried out using the simulator data. Despite the non-linearity and complexity of the system, engine parameters were predicted very well for a particular flight envelope. This estimated model could be used for engine performance analyzes and could provide aircraft control during this cruise phase.

Engine model identification could also be carried out for the other phases of climb and descent in order to obtain its complete model for the whole Cessna Citation X aircraft flight envelope (climb, cruise, descent). This method used in this work could also be efficient to realize a model to identify aerodynamic coefficients of the same airplane always from flight tests.

Keywords: identification, model, flight tests, neural networks, engine, airplane

TABLE DES MATIÈRES

	Page
INTRODUCTION	1
CHAPITRE 1 REVUE DE LA LITTÉRATURE	5
1.1 Objectif de la recherche	5
1.2 Les types des moteurs des avions	6
1.2.1 Les moteurs turboréacteurs	6
1.2.2 Les turboréacteurs à double flux (turbofan).....	8
1.3 L'avion Cessna Citation X.....	10
1.4 Identification des systèmes	13
1.5 Les différents modèles d'identification des moteurs des avions réalisés	15
1.5.1 Les modèles des moteurs existants relatifs à la poussée.....	15
1.5.2 Les modèles des moteurs existants relatifs à la consommation du carburant	18
1.5.3 Les modèles existants des moteurs basés sur les réseaux de neurones.....	19
1.6 Les différentes méthodes d'optimisation	21
1.6.1 Les algorithmes génétiques.....	21
1.6.2 Le recuit simulé.....	23
1.6.3 La recherche tabou.....	24
1.6.4 Le grand déluge étendu.....	25
CHAPITRE 2 INTRODUCTION AUX RÉSEAUX DE NEURONES	29
2.1 Le modèle mathématique.....	30
2.2 Architecture des réseaux de neurones.....	31
2.3 Perceptron multicouche	34
2.4 L'apprentissage des réseaux de neurones	35
2.4.1 L'algorithme de rétro-propagation du gradient.....	38
2.4.2 L'algorithme Gauss-Newton.....	41
2.4.3 Algorithme Levenberg-Marquardt.....	42
2.5 La validation croisée	45
CHAPITRE 3 IDENTIFICATION DES PARAMÈTRES DU MOTEUR	47
3.1 Les données d'apprentissage.....	47
3.1.1 Le simulateur de vol.....	48
3.1.2 La phase de croisière.....	49
3.2 La conception du modèle de prédiction des paramètres du moteur.....	50
3.2.1 Le principe de construction du modèle.....	50
3.2.2 La correction des paramètres d'apprentissage	57
3.2.3 L'apprentissage des données avec les réseaux de neurones	59
CHAPITRE 4 RÉSULTATS ET DISCUSSIONS	67

4.1	Modèle du moteur pour l'altitude et la manette de gaz fixes et pour les nombres de Mach variables	67
4.2	Modèle du moteur aux altitudes variables, nombres de Mach variables, températures variables et à la position de la manette de gaz (TLA) fixe	71
4.2.1	Résultats du premier bloc (N1PC)	71
4.2.2	Résultats du deuxième bloc (N2PC, FNC, WFC).....	75
4.2.2.1	Les résultats de la vitesse de rotation du moteur à haute pression N2PC	76
4.2.2.2	Les résultats de la poussée nette du moteur (FNC).....	79
4.2.2.3	Les résultats de la consommation du carburant du moteur (WFC)	81
4.2.3	Résultats du modèle complet	84
4.2.3.1	Résultats obtenus pour la vitesse de rotation du moteur à haute pression N2PC	84
4.2.3.2	Les résultats de la poussée nette du moteur FNC	87
4.2.3.3	Les résultats obtenus pour la consommation du carburant WFC	90
	CONCLUSION.....	97
	LISTE DE RÉFÉRENCES BIBLIOGRAPHIQUES	99

LISTE DES TABLEAUX

	Page
Tableau 1.1	Les caractéristiques et les performances de Cessna Citation X.....11
Tableau 2.1	Les différentes fonctions d'activation32
Tableau 3.1	Les variables utilisées pour la conception du modèle.....53
Tableau 3.2	Les différents modes de moteur54
Tableau 3.3	Les paramètres à utiliser dans la conception du modèle.....56
Tableau 4.1	Erreur relative moyenne pour le NIPC pour plusieurs altitudes fixes et pour la plage des nombres de Mach = 0.22 à 0.84.....70
Tableau 4.2	Comparaison des erreurs de prédiction de la poussée nette des moteurs par différents modèles.....94
Tableau 4.3	Comparaison des erreurs de la prédiction de la consommation du carburant par différents modèles.....94

LISTE DES FIGURES

		Page
Figure 1.1	Les différentes parties d'un turboréacteur	7
Figure 1.2	Les différentes parties d'un turboréacteur à double flux	9
Figure 1.3	L'avion Cessna Citation X	10
Figure 1.4	Enveloppe de vol de Cessna Citation X (poids en fonction de Xcg).....	12
Figure 1.5	Enveloppe de vol de Cessna Citation X (Altitude, TAS)	13
Figure 1.6	Principe du modèle	14
Figure 1.7	Architecture hybride d'estimation des paramètres de durée de vie du moteur	20
Figure 2.1	Définition d'un neurone artificiel	31
Figure 2.2	Architecture d'un réseau de neurone	31
Figure 2.3	Perceptron multicouches avec une couche cachée.....	35
Figure 2.4	Algorithme de rétro-propagation du gradient	40
Figure 2.5	Algorithme de Levenberg-Marquardt	44
Figure 3.1	Le simulateur de vol de Cessna Citation X de LARCASE.....	48
Figure 3.2	Les phases de vol à partir du décollage à l'atterrissage.....	49
Figure 3.3	Le modèle de prédiction des paramètres du moteur proposé.....	52
Figure 3.4	Les outils utilisés pour la conception du modèle	53
Figure 3.5	Le modèle de prédiction des paramètres du moteur	55
Figure 3.6	Algorithme combiné EGD et réseaux de neurones.....	61
Figure 3.7	Enveloppe du vol TLA = 35 degrés	63
Figure 3.8	Bloc de prédiction de NIPC pour altitudes variables, Mach variable et TLA 35 degrés	64

Figure 3.9	Architecture du modèle de prédiction des paramètres du moteur du Cessna Citation X en croisière.....	65
Figure 4.1	Enveloppe de vol TLA = 35 degrés.....	69
Figure 4.2	N1PC du simulateur et du modèle pour plusieurs altitudes gardées fixes, pour le nombre de Mach = 0.22 à 0.84 et TLA = 35 degrés.....	69
Figure 4.3	N1PC du simulateur et N1PC du modèle pour TLA = 35 degrés.....	72
Figure 4.4	Enveloppe de vol pour le TLA = 65 degrés.....	73
Figure 4.5	N1PC du simulateur et N1PC du modèle pour le TLA = 65 degrés.....	74
Figure 4.6	Erreur relative moyenne de N1PC dans l'identification du modèle.....	74
Figure 4.7	Erreur relative moyenne de N1PC dans la validation du modèle.....	75
Figure 4.8	N2PC du simulateur et N2PC du modèle pour TLA = 35 degrés.....	76
Figure 4.9	N2PC du simulateur et N2PC du modèle pour TLA = 65 degrés.....	77
Figure 4.10	Erreur relative moyenne de N2PC pour l'identification du modèle.....	78
Figure 4.11	Erreur relative moyenne de N2PC pour la validation du modèle.....	78
Figure 4.12	FNC du simulateur et FNC du modèle pour TLA = 35 degrés.....	79
Figure 4.13	FNC du simulateur et FNC du modèle pour TLA = 65 degrés.....	80
Figure 4.14	Erreur relative moyenne de FNC en identification du modèle.....	80
Figure 4.15	Erreur relative moyenne de FNC en validation du modèle.....	81
Figure 4.16	WFC du simulateur et WFC du modèle pour TLA = 35 degrés.....	82
Figure 4.17	WFC du simulateur et WFC du modèle pour TLA = 65 degrés.....	82
Figure 4.18	Erreur relative moyenne de WFC en identification du modèle.....	83
Figure 4.19	Erreur relative moyenne de WFC en validation du modèle.....	83
Figure 4.20	N2PC du simulateur et N2PC du modèle pour TLA = 35 degrés.....	85
Figure 4.21	N2PC du simulateur et N2PC du modèle pour TLA = 65 degrés.....	86
Figure 4.22	Erreur relative moyenne de N2PC pour toutes les TLA en identification.....	86

Figure 4.23	Erreur relative moyenne de N2PC pour toutes les TLA en validation	87
Figure 4.24	FNC du simulateur et FNC du modèle pour TLA = 35 degrés.....	88
Figure 4.25	FNC du simulateur et FNC du modèle pour TLA = 65 degrés.....	88
Figure 4.26	Erreur relative moyenne de FNC pour toutes les TLA en identification ...	89
Figure 4.27	Erreur relative moyenne de FNC pour toutes les TLA en validation	90
Figure 4.28	WFC du simulateur et WFC du modèle pour TLA = 35 degrés.....	91
Figure 4.29	WFC du simulateur et WFC du modèle pour TLA = 65 degrés.....	91
Figure 4.30	Erreur relative moyenne de WFC pour toutes les TLA en identification ..	92
Figure 4.31	Erreur relative moyenne de WFC pour toutes les TLA en validation	93

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

Abréviations

FAA	Federal Aviation Administration
LARCASE	Laboratoire de Recherche en Commande Active, Avionique et Aéroserveoélasticité
EGD	Extended Great Deluge
PID	Proportionnel Intégrateur Dérivateur
FADEC	Full Authority Digital Engine Control
TLA	Thrust Level Angle
TAS	True Air Speed

INTRODUCTION

Les besoins de se déplacer, ainsi que les exigences accrues des agences de certification des transports aériens combinent la nécessité d'assurer la sécurité, le confort des passagers et aussi la réduction des coûts de vol des aéronefs. Ces besoins obligent les compagnies aériennes à chercher des méthodes d'optimisation des coûts de conception et de fabrication des avions. Plusieurs essais en vol sont indispensables au développement et à l'exploitation des aéronefs afin d'assurer leur stabilité et sécurité pendant les différentes phases de vol. Cependant, ces essais en vol peuvent être très coûteux et surtout peuvent risquer la vie des ingénieurs et des pilotes. Ainsi, nous pouvons comprendre la nécessité d'une meilleure modélisation possible des aéronefs qui permettrait de valider plusieurs critères et lois de commande avant de reproduire les essais en vol sur le prototype réel.

La modélisation d'un système fait partie des étapes importantes de sa conception. En effet, le modèle réalisé pour n'importe quel système doit être judicieux et précis afin de garantir l'obtention de très bons résultats. Cette modélisation pourrait servir dans le développement de nouvelles technologies très efficaces pour les avions, rendre plus performants les structures des avions et mieux gérer le trafic aérien.

L'amélioration des performances des avions civils et militaires nécessite la conception et l'intégration multidisciplinaires de leurs différents modèles pour les différentes composantes de ces avions incluant les moteurs. Cette amélioration devrait assurer les moindres coûts de consommation de carburant en gardant la meilleure qualité possible du modèle.

Parmi les objectifs à accomplir dans le cadre des améliorations des performances des avions, les industries aérospatiales visent à réduire les consommations de carburant et les émissions nocives. En effet, afin d'atteindre ces objectifs, les tests en vol qui doivent être réalisés peuvent s'avérer coûteux en termes de temps et d'argent.

Les tests sur les avions et leurs moteurs, tiennent compte d'un très grand nombre de paramètres, c'est pour cette raison qu'ils prennent beaucoup de temps. Ainsi, une excellente modélisation de ces moteurs pourrait être la meilleure solution possible pour éviter ce problème de temps.

Le moteur est la partie la plus complexe d'un aéronef à modéliser à cause de la complexité de son fonctionnement et la grande diversité des paramètres qui interviennent dans sa modélisation. La conception d'un modèle pertinent du moteur dépend de ses paramètres. De ce fait, l'excellente identification de ces paramètres permet l'amélioration des performances des avions.

Les moteurs sont des composantes (systèmes) responsables de maintenir la stabilité de l'avion pendant ses différentes phases de vol. Ils servent aussi à gérer d'autres composantes de l'avion tout au long de son vol. Le fonctionnement des moteurs des avions est complexe. En effet, le design d'un modèle fiable pour l'identification des paramètres d'un moteur nécessite beaucoup de connaissances multidisciplinaires.

Une simulation par ordinateur qui est très précise permettrait de modéliser le fonctionnement réel d'un moteur d'un aéronef. La modélisation d'un moteur d'avion pourrait être effectuée à plusieurs niveaux de détails. Par exemple, une modélisation des moteurs pourrait s'effectuer à l'aide de super ordinateurs pour la simplification des fonctions mathématiques reliant leurs paramètres et leurs fonctions de transfert utilisées pour le contrôle (Sonny et al., 2008).

L'équipe du Laboratoire de Recherche en Commande Active, Avionique et Aéroservoélasticité (LARACASE) développe de nouvelles méthodes de modélisation des avions dans le but d'améliorer la conception des avions des compagnies avec lesquelles elle travaille. Plusieurs projets qui ont été réalisés, et d'autres projets sont en cours de réalisation au LARACASE. Les principaux objectifs de ces projets sont l'optimisation des trajectoires de vol (Murrieta et al., 2016) (Murrieta et al., 2015), l'amélioration des performances des ailes des avions (Korenschi, A. et al., 2016), (Grigorie et al., 2013), l'estimation des coefficients

aérodynamiques à partir des géométries des ailes (Ben Mosbah et al., 2016) , ainsi que l'optimisation des performances de la gestion de vol (Pollender et al., 2011).

Cette recherche réalisée au LARCASE a pour objectif de construire une méthode d'identification des paramètres du moteur de l'avion d'affaires Cessna Citation X. Cette identification est basée sur la méthodologie des réseaux de neurones qui sont adaptés particulièrement à l'identification et au contrôle des systèmes dynamiques qui varient au cours du temps. Le premier chapitre de ce mémoire présente les différents modèles réalisés des moteurs des avions avec différentes méthodes ainsi que les moyens d'optimisation des erreurs utilisés.

Le deuxième chapitre montre en détails l'application des méthodes des réseaux de neurones, leur rôle et leur principe de fonctionnement. Ce chapitre permet de bien comprendre le principe de conception de la méthodologie d'identification des paramètres du moteur.

Les principes et les étapes de conception du modèle de moteur sont abordés à travers le troisième chapitre dans le but d'expliquer les relations entre les différents paramètres qui ont des influences sur le fonctionnement du moteur. La modélisation du fonctionnement de ce moteur permet la prédiction des valeurs de ces paramètres influents qui sera présentée à travers ce chapitre.

Le quatrième chapitre présente les résultats obtenus à partir du modèle conçu et seront comparés et validés par rapport aux données du simulateur de vol de recherche de niveau D du LARCASE. Ces résultats sont exprimés sous forme d'erreurs relatives pour 69 cas de vol pour l'identification et pour la validation. L'erreur relative représente la différence entre la valeur prédite et la valeur réelle divisée par cette valeur réelle. Cette erreur est utilisée dans les problèmes de minimisation des erreurs de prédiction.

CHAPITRE I

REVUE DE LA LITTÉRATURE

Ce chapitre présente la recherche bibliographique concernant les travaux effectués sur l'idée générale du projet. Tout d'abord, les types des moteurs des avions ainsi que le principe d'identification des systèmes sont présentés. Ensuite, les différentes méthodes de conception des modèles permettant l'identification des paramètres des moteurs des aéronefs sont discutées. Ensuite, nous présentons les différentes disciplines et outils qui sont nécessaires au déroulement du projet. Finalement, les algorithmes d'optimisation des erreurs sont présentés.

1.1 Objectif de la recherche

La mise en contexte présentée dans l'introduction permet de dégager l'objectif principal de ce mémoire qui se résume par la conception d'une méthodologie d'identification des paramètres du moteur de l'avion Cessna Citation X au cours de la phase de croisière qui est la phase la plus longue du vol. Afin d'atteindre cet objectif, nous proposons d'étudier en premier les différents types de moteurs des avions qui existent dans le but de déterminer les paramètres qui interviennent essentiellement dans leur fonctionnement. Ensuite, les principes des méthodes d'identification utilisées pour la réalisation des modèles de ces moteurs seront étudiés. La méthode de modélisation du moteur que nous avons préconisé a été choisie suite à l'étude des modèles de moteurs d'avions réalisés au cours des dernières années et après l'analyse des résultats obtenus pour ces différents modèles.

Notre modèle est basé sur l'apprentissage artificiel et plus précisément sur les réseaux de neurones qui sont expliqués au deuxième chapitre de ce mémoire. Dans le but d'obtenir un modèle précis d'identification des paramètres du moteur, il faudrait optimiser les réseaux de neurones à l'aide d'un algorithme d'optimisation. En effet, une étude des différents algorithmes d'optimisation est nécessaire pour choisir le meilleur d'entre eux, l'appliquer au modèle et garantir le minimum d'erreur de prédiction possible.

1.2 Les types des moteurs des avions

Il existe plusieurs types de moteurs d'avions :

- Les turboréacteurs;
- Les turbopropulseurs;
- Les turbomoteurs;
- Les statoréacteurs;
- Les moteurs à pistons;
- Les moteurs à hélices;
- Les moteurs-fusées.

De nos jours, les types de moteurs les plus souvent utilisés dans les avions civils sont les turboréacteurs choisis pour notre étude. Ces deux types de moteurs fonctionnent sur le même principe de "turbine" (Goder, 2015) qui sera expliqué dans les sections suivantes.

1.2.1 Les moteurs turboréacteurs

Un turboréacteur se compose d'un tube qui contient :

- Un compresseur à plusieurs étages;
- Une chambre à combustion;
- Une turbine à gaz;
- Une tuyère de sortie.

Le principe de fonctionnement de ce type de moteur se résume par l'entrée d'air froid par l'avant et qui sera mis sous pression par le compresseur. Ensuite, l'air comprimé entre dans la chambre à combustion afin de se mélanger au carburant puis s'enflammer ce qui permet la production d'une très grande force de poussée causée par la dilatation de la chaleur et l'explosion du mélange. Ensuite, la turbine à gaz tourne par l'air explosé qui sort du moteur par la tuyère de sortie. Un schéma visualise mieux les étapes citées précédemment (Girard, G. et al., 2016) :

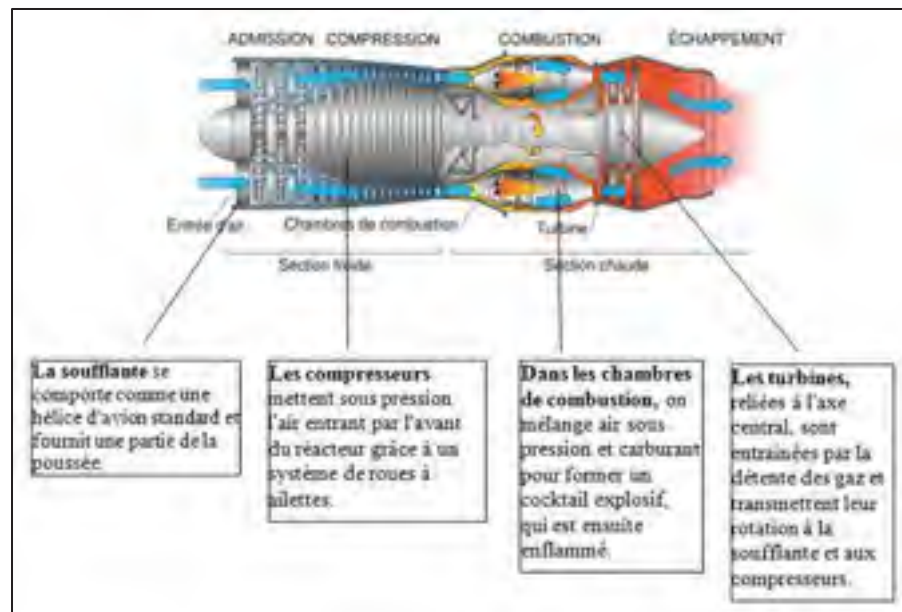


Figure 1.1 Les différentes parties d'un turboréacteur
Tirée de Girard, G. et al., (2016)

La force de poussée du moteur pourrait être exprimée (en thermodynamique) en fonction de la masse d'air et la différence entre la vitesse d'entrée et la vitesse de sortie des écoulements de gaz. L'équation 1.1 présente cette relation :

$$F_{poussée} = M(V_{sortie} - V_{entrée}) \quad (1.1)$$

Où M est le débit massique de l'air passant dans le moteur (kg/s)

$V_{entrée}$: Vitesse d'entrée des gaz dans le compresseur (m/s)

V_{sortie} : Vitesse de sortie des gaz de la tuyère du turboréacteur (m/s)

Lorsque la totalité de l'air traverse le moteur et actionne les turbines, le turboréacteur est dit à "simple flux". En effet, pour ce type de moteur lorsque la température de l'air ambiant augmente, l'air chaud et le flux massique de l'air passant dans le moteur diminuent ce qui provoque la diminution de la poussée du réacteur. Pour augmenter cette poussée au décollage

et durant les périodes chaudes, il faudrait injecter de l'eau à l'entrée du compresseur qui devrait se vaporiser et refroidir l'air (Gaëtan, 2009). Dans le but d'accroître l'efficacité et la performance du moteur, le compresseur a été divisé en deux parties, l'une à "basse pression" et l'autre à "haute pression", ce qui définit les turboréacteurs à double flux.

1.2.2 Les turboréacteurs à double flux (turbofan)

Ce type de moteur exprime une amélioration du moteur turboréacteur à simple flux dont nous avons parlé dans le paragraphe précédent. Le principe de fonctionnement de ce moteur est un peu différent de celui du turboréacteur à simple flux. En effet, seulement une partie de l'air qui entre dans le moteur sera par la suite comprimée ; l'autre partie retourne à l'extérieur. Ensuite, l'air se mélange avec le gaz provenant de la turbine avant d'arriver à la tuyère (Gaëtan, 2009). Les avantages des turboréacteurs à double flux sont les suivants : (Weng et al.) (Gaëtan, 2009)

- Une plus grande poussée au décollage par rapport à la poussée d'un turboréacteur à simple flux.
- Un refroidissement du moteur grâce à la déviation de l'air.
- Une diminution du bruit en provenance de la combustion grâce au gaz d'échappement qui est noyé dans le flux d'air supplémentaire.

Un modèle de turboréacteur à double flux a été présenté. Il se compose d'un ventilateur à un étage, d'un compresseur axial à basse pression à 3 étages, d'un compresseur axial à haute pression à 9 étages, d'une chambre de combustion, une turbine à haute pression à un étage et d'une turbine à basse pression à 4 étages (Aydin et al., 2015).

La turbine à haute pression gère le compresseur à haute pression qui permet de pressuriser l'air et passer à l'étage de combustion. Alors que la turbine à basse pression gère le ventilateur qui pressurise et accélère la déviation de l'air. Une partie d'air prise à travers le noyau du moteur restant de l'écoulement (20% du débit), passe par le ventilateur et se mélange avec l'air chaud juste en sortant de la turbine. L'air comprimé sera mélangé au carburant et fournit ainsi la combustion. Le gaz à très haute pression et à très haute température sera dirigé en direction de la turbine à basse pression (comme montré dans la figure 1.2) afin d'obtenir la poussée

nécessaire pour la propulsion de l'avion. Le gaz sortant de la turbine sera éliminé à travers le tuyau d'échappement à haute vitesse. Le ventilateur est l'une des parties importantes du moteur. En effet, 75% à 80% de la poussée du moteur est générée par le ventilateur. Pour le turboréacteur à double flux, il y a deux axes de turbine. Un axe relie la turbine à haute pression au compresseur à haute pression, tandis que l'autre axe relie la turbine à basse pression au compresseur à basse pression et au soufflante (Aydin et al., 2015).

La figure suivante 1.2 représente les différentes parties d'un turboréacteur à double flux :

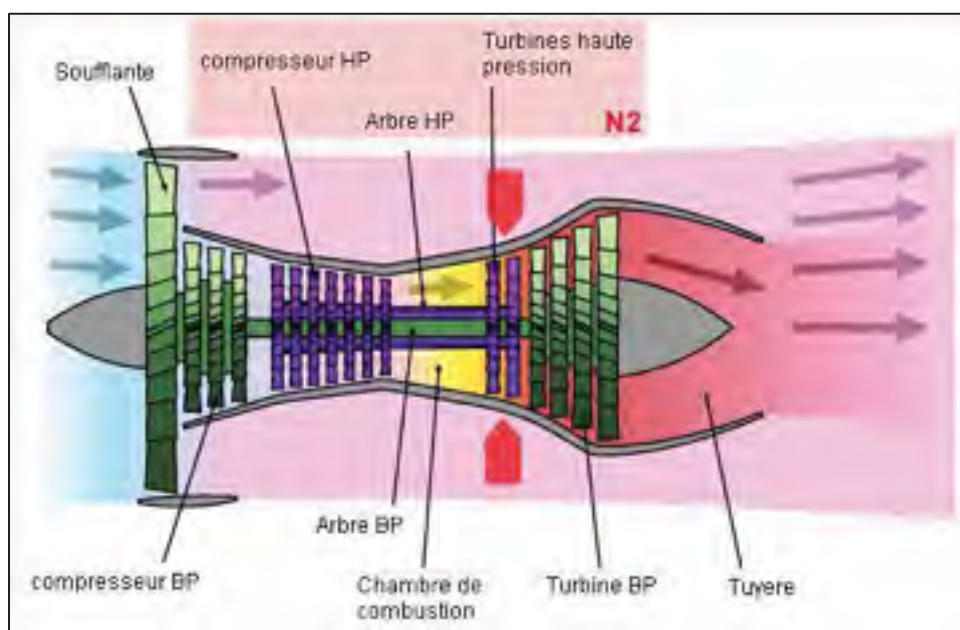


Figure 1.2 Les différentes parties d'un turboréacteur à double flux
Tirée de De Magnitot et al., (2010)

Cette étude des moteurs nous a permis de différencier leurs types et de distinguer leurs différentes parties et leurs fonctionnements. Lors de notre projet, l'identification du moteur de l'avion Cessna Citation X devrait être réalisée, une étude de cet avion et de son moteur qui est un turboréacteur à double flux est nécessaire afin d'identifier les paramètres qui gèrent le comportement de cet aéronef.

1.3 L'avion Cessna Citation X

L'avion Cessna Citation X est un avion d'affaire conçu et fabriqué par la compagnie Cessna Aircraft Company aux États Unis. Le premier vol de cet avion a été effectué le 21 décembre 1993. C'est l'avion civil le plus rapide au monde et il a la capacité d'emporter entre 8 et 12 passagers. Cet avion est conçu dans le but de répondre aux besoins des clients qui veulent toujours se déplacer le plus rapidement possible entre les continents et les différents pays. Étant un avion d'affaires, il est comparable au Falcon 2000 LXS de Dassault Aviation ou au Challenger 300 de Bombardier. Cessna Citation X possède deux moteurs turboréacteurs à double flux qui sont construits par Rolls-Royce (Chahbani, 2015), l'avion Cessna est présenté dans la figure 1.3 ci-dessous :



Figure 1.3 L'avion Cessna Citation X
Tirée de Leclercq, (2015)

Le tableau 1.1 ci-dessous représente les caractéristiques et les performances de cet avion :

Tableau 1.1 Les caractéristiques et les performances de Cessna Citation X

Données géométriques du Cessna Citation X	Valeurs numériques
Longueur	22.04 m
Envergure	19.48 m
Hauteur	05.86 m
Surface des ailes	49 m ²
Masse maximale	16,511 kg
Données des performances du Cessna Citation X	
Vitesse maximale de croisière	527 nœuds (976 km/h)
Altitude maximale	51,000 pi (15,545 m)
Rayon d'action	6,004 km
Vitesse maximale par rapport à la vitesse du son	Mach 0.93
Distance au décollage	5,280 pi (1,609 m)
Distance d'atterrissage	3,430 pi (1,045 m)
Taux de montée	3,650 pi/min (43,000 pi en 28 min à pleine charge)
Données des moteurs	
Types des moteurs	Turbofan Rolls-Royces AE 3007C1
Poussée maximale	30.09 kN

L'enveloppe de vol de l'avion Cessna Citation X est le domaine dans lequel son intégralité structurelle est assurée par le constructeur. En effet, aucun dépassement n'est permis en dehors des limites de cette enveloppe car ceci peut endommager ou détruire la structure de l'avion.

Généralement, chaque avion possède sa propre enveloppe de vol unique. Le simulateur de recherche de vol de Cessna Citation X fournit par CAE Inc. qui se trouve à l'ÉTS au LARCASE contient les deux types d'enveloppes de vol :

- La première enveloppe est donnée par le poids total de l'avion (Weight (klb)) en fonction de la position du centre de gravité ($X_{cg}(\%)$)
- La seconde enveloppe est donnée par l'altitude (pi) en fonction de la vitesse vraie de l'avion par rapport à l'air (TAS (nœuds)) mais aussi en fonction du nombre de Mach

La figure 1.4 représente le premier type d'enveloppe de vol et la figure 1.5 montre le deuxième type de l'enveloppe de vol pour le Cessna Citation X.

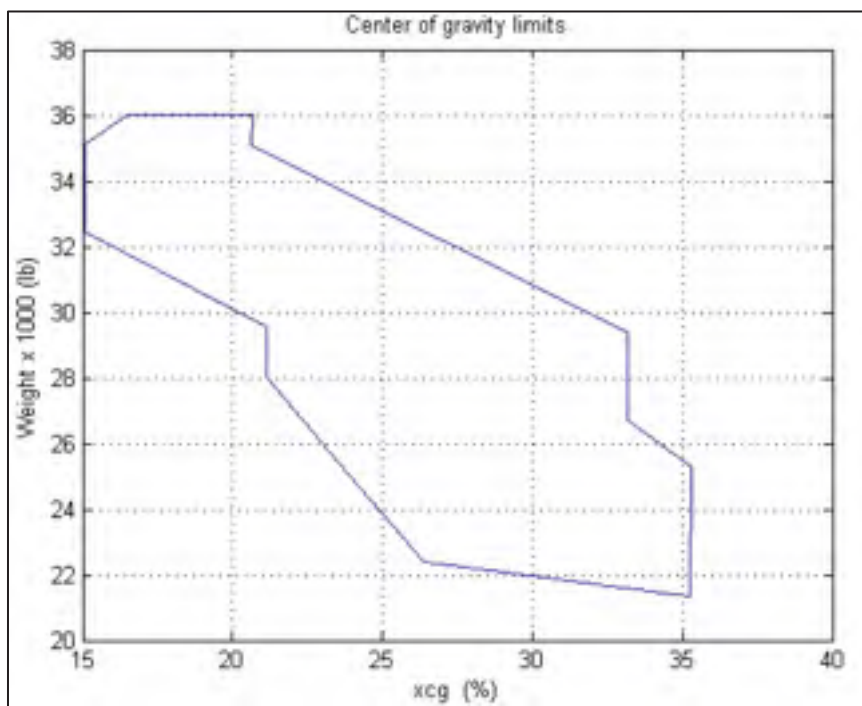


Figure 1.4 Enveloppe de vol de Cessna Citation X
(poids en fonction de Xcg)

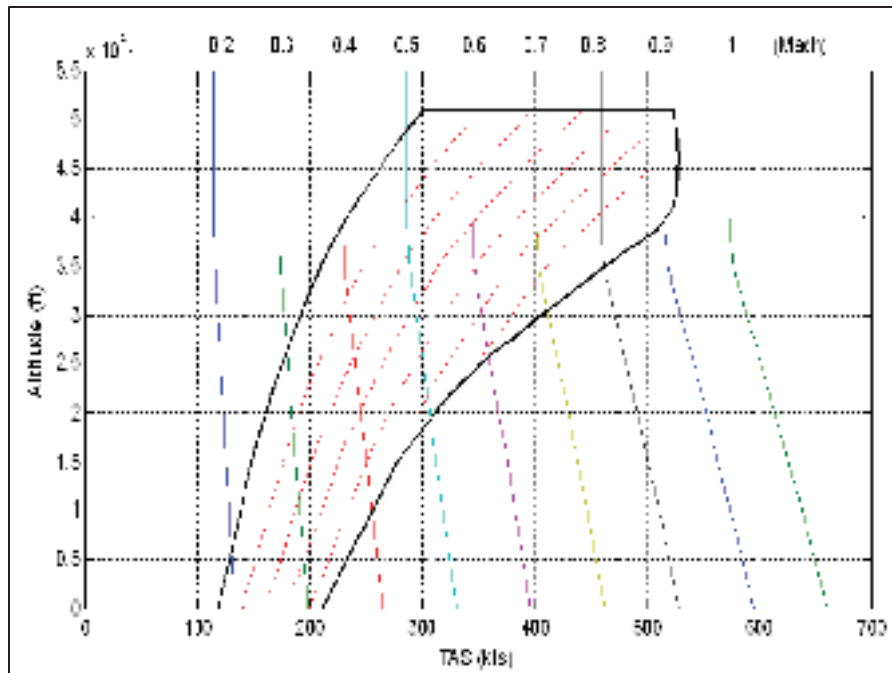


Figure 1.5 Enveloppe de vol de Cessna Citation X
(Altitude, TAS)

L'identification des paramètres du moteur de Cessna doit tenir compte de ces deux types d'enveloppes de vol ainsi que des performances de cet avion. La compréhension du principe de cette identification est nécessaire afin de réaliser la modélisation la plus utile du moteur de l'avion.

1.4 Identification des systèmes

Le modèle doit reproduire au mieux le comportement du système (processus) pour toutes les conditions nécessaires à son fonctionnement

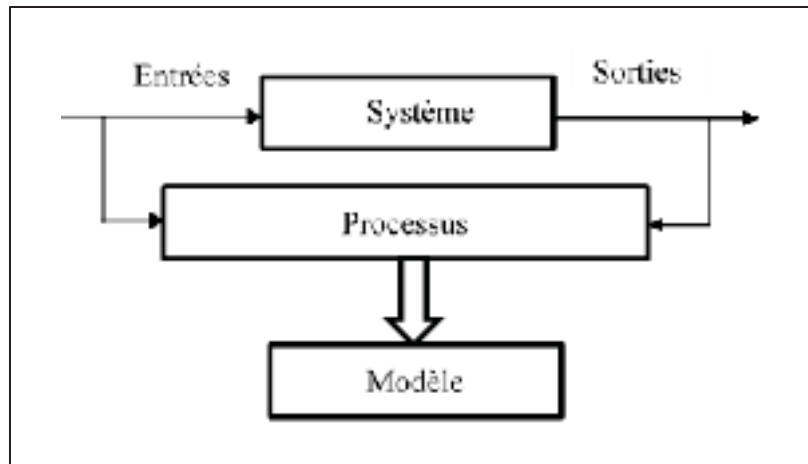


Figure 1.6 Principe du modèle

Les objectifs de conception d'un modèle sont les suivants :

- Augmenter la connaissance d'un système physique "modèle de connaissance" ; ces connaissances ont pour but de diminuer les difficultés de la description fidèle des phénomènes complexes.
- Représenter le comportement global d'un système physique "modèle de représentation" ; nous utilisons un système de type "boîte noire" dans lequel les paramètres du modèle n'ont pas de sens physique, par contre ce modèle est utile pour développer les lois de commande.

Les domaines d'application sont plusieurs, parmi lesquels nous pouvons énumérer :

- "Simulation" : les simulateurs des avions;
- "Prédiction des sorties d'un système" : la prédiction des paramètres de vol d'un avion, la commande prédictive...
- "Synthèse des régulateurs" : le contrôle des procédés industriels.

"L'identification" est une approche expérimentale qui permet de déterminer un modèle dynamique en utilisant des algorithmes différents. Les étapes de l'identification d'un modèle sont : (Ioan-Doré, 2002)

1. L'acquisition des entrées/sorties sous un protocole d'expérimentation, ce protocole permet l'extraction d'un modèle significatif de procédé.

2. Le choix de la structure du modèle ; déterminer les paramètres à estimer, ainsi que leur nombre.
3. L'estimation (identification) des paramètres du modèle ; utiliser des algorithmes qui fournissent les paramètres du modèle à partir des entrées/sorties du modèle.
4. La validation du modèle après son identification ; vérifier la représentation du modèle identifié du comportement du système par la connaissance de ses entrées et sorties.

Les types des méthodologies d'identification des systèmes sont :

- "Paramétrique" : procédure en temps réel, elle peut s'effectuer lorsque le système est en exploitation;
- "Non paramétrique" : cette procédure nécessite des essais particuliers et le système doit être "hors exploitation".

1.5 Les différents modèles d'identification des moteurs des avions réalisés

1.5.1 Les modèles des moteurs existants relatifs à la poussée

Plusieurs modèles ont été réalisés afin d'identifier des paramètres spécifiques aux moteurs des aéronefs. Les objectifs essentiels de ces modèles sont : d'estimer leurs performances de la conception des moteurs, de prédire et représenter leur comportement et fonctionnement et de faciliter la prise des décisions lors des étapes de fabrication des avions (Daniel, 1992). En effet, la majorité des modèles réalisés est basée sur des équations polynomiales. Ces modèles analytiques ont permis l'estimation des paramètres des moteurs tout en assurant la haute qualité de sa performance, ainsi que de sa précision.

Une méthode d'identification des paramètres du moteur de l'avion Cessna Citation X a été réalisée dans notre laboratoire LARCASE à partir des données du simulateur de vol de niveau D (Ghazi et al., 2015). Ce modèle a considéré deux algorithmes d'estimation ("la méthode des moindres carrés" ainsi que la "méthode d'optimisation par essais particulières") dans le but de prédire la force de poussée du moteur dans n'importe quelle condition de vol. De très bons résultats ont été obtenus grâce à ce modèle avec une erreur relative moyenne entre les données

de simulateur de la force de la poussée du moteur et les données expérimentales qui a été à égale à 1.58% ; sa valeur maximale a été de 3.2%.

Un modèle générique a été développé par Rodriguez et al. (2012) ; la poussée d'un turboréacteur d'un avion civil a été estimée dans ce modèle. Le principe de réalisation de ce modèle a été différent de celui proposé par Ghazi et al. (2015) car celui-ci n'a pas tenu compte des données provenant du simulateur ; par contre, il s'était basé sur des données qui pourraient être changées, telles que le taux de dilution, ainsi que la poussée maximale au niveau de la mer. La validation de ce modèle a été réalisée grâce à la comparaison de ses résultats par rapport aux données de 4 types de moteurs (Honeywell TFE 731-5, General Electric TF39, Garrett ATF3 et General Electric CF6-80). Ces types de moteurs ont été choisis suite à la présence de leurs données dans la littérature. Les erreurs relatives des résultats du modèle par rapport aux données des trois premiers moteurs que nous venons de citer n'a pas dépassé plus de 6%, alors que l'erreur relative maximale du modèle par rapport aux données du dernier type du moteur (CF6-80) a été de 14%. En effet, le modèle n'a pas tenu compte de la géométrie et des caractéristiques de chaque moteur, en plus, le manque de données a affecté aussi la précision du modèle. Nous avons déduit que le modèle du moteur basé sur les données du simulateur de vol réalisé par Ghazi et al. (2015) a été plus efficace que celui de Rodriguez et al. (2012) car la quantité des données a été plus importante ; ceci a permis d'assurer une très grande précision de prédiction du modèle de Ghazi et al. (2015) par rapport au modèle basé sur une petite quantité de données. Ceci, nous permet de conclure sur la nécessité d'avoir une grande base de données pour la conception et la réalisation du modèle de moteur de l'avion Cessna Citation X présenté dans ce mémoire.

D'autres modèles de moteurs d'avions ont été proposés sur la base des relations mathématiques liant les différents paramètres qui affectent la poussée du moteur. En effet, le modèle proposé par Mattingly et al. (1987) tenait compte de la densité de l'air et de la variation du nombre de Mach pour déterminer la poussée maximale du moteur ; le modèle proposé par Wanner (1976) tenait compte de la position de la manette de gaz. Ces deux modèles étaient basés sur les

résultats obtenus par des analyses aérodynamiques et thermodynamiques obtenues pour chaque phase de vol.

L'équation du modèle de Wanner (1976) est présentée par :

$$F = K_f \rho (\alpha M)^{\lambda_f} \delta_X \quad (1.2)$$

Où F : la poussée du moteur

K_f : une constante déterminée

ρ : la densité de l'air à une altitude donnée

M : le nombre de Mach

α : la vitesse du son à une altitude donnée

λ_f : un coefficient qui dépend du type du moteur (entre -1 et 2)

δ_X : la position de la manette de gaz

Le modèle de Mattingly et al. (1987) utilise l'équation 1.3 ci-dessous :

$$\frac{F_{max}}{F_0} = \left(\frac{\rho}{\rho_0}\right)^6 [0.568 + 0.25(1.2 - M)^3] \quad (1.3)$$

Où F_{max} : la poussée maximale du moteur en croisière

F_0 : la poussée maximale du moteur au sol et à l'arrêt

ρ : la masse volumique de l'air pour l'avion en vol, alors à une certaine altitude

ρ_0 : la masse volumique de l'air de l'avion au sol

M : le nombre de Mach

Nous remarquons pour les deux modèles, la dépendance de la poussée du moteur de l'altitude et du nombre de Mach dans les équations (1.2) et (1.3). En plus, le modèle de Wanner (1976) dépend de la position de la manette de gaz. Mais, ce modèle-là, ne tient pas compte de l'effet de taux de dilution pour les moteurs civils, ainsi la fonction (relation) entre la poussée et la position de la manette de gaz est linéaire.

D'après l'analyse présentée dans l'article de Ghazi et al. (2015), l'erreur maximale de la poussée du moteur obtenue pour le modèle de Wanner (1976) a été de 6% pour le nombre de Mach 0.1, et, elle a été de 3.4% pour le modèle de Mattingly et al. (1987) pour le nombre de Mach de 0.4. En effet, pour ces deux modèles, l'erreur relative moyenne a été aussi calculée pour les altitudes et elle a été de 3.4% pour le modèle de Mattingly et al. (1987) tandis que pour le modèle de Wanner (1976), l'erreur relative a été très grande de l'ordre de 20% pour l'altitude de 30,000 pieds.

Tous les modèles qui ont été présentés précédemment s'intéressaient à la détermination de la poussée du moteur, alors qu'il y a des modèles de moteurs qui s'intéressent à la consommation du carburant par l'avion. Les deux paramètres : la "poussée" du moteur ainsi que sa "consommation du carburant" sont importants dans le but de gérer et d'optimiser son fonctionnement.

1.5.2 Les modèles des moteurs existants relatifs à la consommation du carburant

La consommation de carburant se présente, pour les réacteurs, sous la forme de produit d'une consommation spécifique et de la poussée du moteur. Le modèle du moteur proposé par Mattingly (1996) a présenté une consommation spécifique qui dépendait de l'altitude, du nombre de Mach, du cycle du moteur, ainsi que de sa température. Ce modèle a offert une consommation spécifique égale à $1.89 \cdot 10^{-5}$ (kg/s)/N en croisière et supérieure à la valeur donnée par le modèle qui considère une consommation de carburant d'une valeur constante égale à $1.75 \cdot 10^{-5}$ (kg/s)/N. Donc, 8% de plus de consommation a été obtenue (Roux, 2002). Le modèle de Torenbeek (1982) était basé sur une analyse détaillée du fonctionnement du moteur. Dans ce modèle la consommation du carburant ainsi que le nombre de Mach augmentaient, alors qu'une augmentation du taux de dilution provoquait la diminution de cette consommation. Mais, cette consommation ne dépendait pas uniquement que du nombre de Mach et du taux de dilution, mais aussi de l'altitude et du rapport de pressions (entrée/sortie) du compresseur (Roux, 2002).

À partir du modèle proposé par Torenbeek (1982), Roux a proposé un modèle de consommation de carburant spécifique à une poussée maximale avec une précision de 3.6% pour les moteurs à taux de dilution supérieur à 3 pour le régime de croisière. Les valeurs de la consommation spécifique de carburant du modèle de Torenbeek (1982) présentaient une erreur de 6.6% par rapport à leurs valeurs expérimentales obtenues en croisière.

Un autre modèle pour les moteurs à double flux a été proposé par Engineering Sciences Data Unit (1982). Il donne la consommation du carburant en fonction du nombre de Mach, du rapport des températures au sol et en vol ainsi qu'un coefficient qui dépend du taux de dilution et de l'altitude. De la même manière que pour les modèles cités précédemment, ce modèle montre que la consommation du carburant augmente lorsque le nombre de Mach augmente aussi. La limitation de ce modèle est sa validité en régime de croisière pour le nombre de Mach situé entre 0.6 et 0.9 (Roux, 2002).

D'autres méthodes ont été utilisées pour l'identification du modèle ; le diagnostic des performances des moteurs se base sur les réseaux de neurones. Ces réseaux servent à l'apprentissage des données et surtout au traitement des systèmes non linéaires. Une explication des réseaux de neurones sera présentée dans le deuxième chapitre.

1.5.3 Les modèles existants des moteurs basés sur les réseaux de neurones

Dans le contexte des réseaux de neurones appliqués à la modélisation des moteurs d'avions, une technique de calcul des caractéristiques de la turbine à gaz d'un moteur d'aéronef a été développée par Grigor'ev et al. (2015) en se basant sur les réseaux de neurones. L'avantage de l'utilisation de cette technique est de représenter les approximations des différentes caractéristiques d'une turbo machine sous forme d'un tableau pour les analyser et assurer leurs meilleures qualités. Elle permet également de réaliser des interpolations et des extrapolations des approximations de ces caractéristiques.

Par ailleurs, un autre modèle basé sur une combinaison des réseaux de neurones avec des algorithmes génétiques a été réalisé pour les diagnostics des performances d'un moteur d'un avion (Kobayashi et al., 2005). En effet, l'utilisation des réseaux de neurones pour ce modèle a servi à estimer l'état interne du moteur, tandis que les algorithmes génétiques ont servi à l'amélioration de la robustesse de mesures des incertitudes. Cette technique hybride a permis de proposer des solutions potentielles pour la résolution des anomalies ou mesures des capteurs dans le but de réduire les fausses alarmes. L'architecture proposée pour ce modèle hybride d'estimation citée précédemment est présentée dans la figure 1.7 :

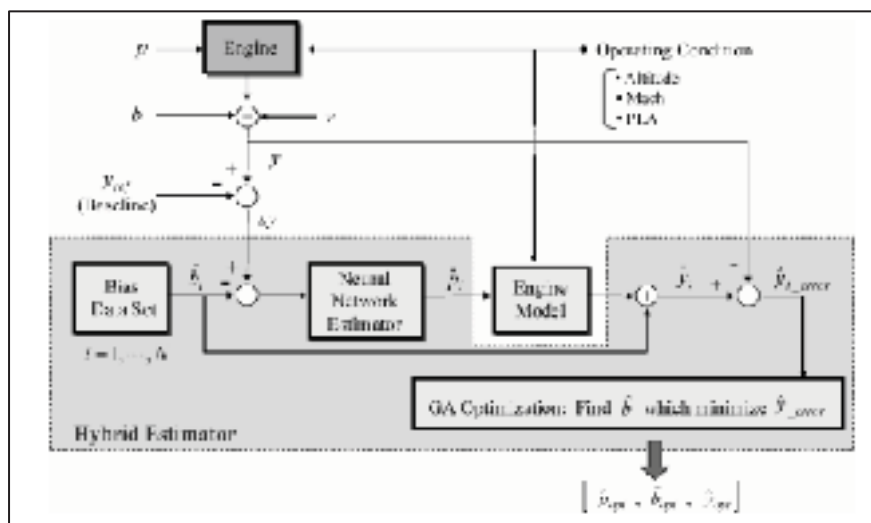


Figure 1.7 Architecture hybride d'estimation des paramètres de durée de vie du moteur

Tirée de Kobayashi et al., (2005)

L'apprentissage des réseaux de neurones pourrait prendre beaucoup de temps. Afin de remédier à ce problème, l'algorithme génétique est utilisé. Pour l'apprentissage des données, le modèle des réseaux de neurones tient compte des paramètres du moteur tels que l'altitude, le nombre de Mach ainsi que la position de la manette de gaz. Par ailleurs, l'ensemble des données représente l'intervalle de solutions sur lequel l'algorithme génétique cherche une solution optimale. Cette solution annule la différence entre les mesures réelles et celles données par les capteurs. L'algorithme génétique minimise l'erreur entre les données du modèle du moteur et ses valeurs expérimentales réelles. L'utilisation des réseaux de neurones pour l'estimation de certains paramètres du moteur a permis le traitement de la non linéarité du

système correspondant. Ainsi, l'utilisation de réseaux de neurones pourrait être efficace dans le cadre de notre projet pour l'estimation des paramètres du moteur de l'avion Cessna Citation X, tels que la poussée du moteur et la consommation du carburant qui dépendent de plusieurs paramètres.

Les algorithmes génétiques trouvent les solutions optimales pour la résolution de certains problèmes dans les meilleurs délais. Même si ces algorithmes sont les plus utilisés dans les problèmes d'optimisation, d'autres méthodes existent pouvant être plus efficaces.

1.6 Les différentes méthodes d'optimisation

Dans le but de résoudre certains problèmes difficiles (grandes dimensions et multicritères), les algorithmes heuristiques et méta-heuristiques sont employés. Les heuristiques sont des méthodes algorithmiques permettant la résolution de ces problèmes d'une façon rapide et permettent de trouver d'excellents compromis entre la rapidité et la qualité de la solution optimale (Amiguet, 2008).

Alors que les méta-heuristiques sont des techniques visant à guider la recherche de la solution optimale ainsi d'explorer l'espace de cette recherche d'une manière efficace afin de déterminer les solutions optimales (Osman et al., 1996). Dans ce cadre, plusieurs méthodes heuristiques et méta-heuristiques sont utilisées.

1.6.1 Les algorithmes génétiques

Les algorithmes génétiques tels mentionnés ci-haut, sont des algorithmes d'optimisation qui utilisent des techniques de l'évolution de la nature. Ils sont utiles pour résoudre les problèmes non linéaires. Leur principe est de partir d'un ensemble de solutions initiales, ensuite obtenir un nouvel ensemble composé de ses meilleures solutions. Ces opérations sont répétées plusieurs fois pour finalement choisir la meilleure solution (LaBarre, M.-O., 2002).

Afin de mieux comprendre le principe d'un algorithme génétique, les étapes de cet algorithme sont présentées ci-dessous (LaBarre, M.-O., 2002) :

1. Générer l'ensemble initial des solutions (la population initiale) de taille N ;
2. Calculer le facteur de qualité pour chaque solution;
3. Vérifier si le critère de la terminaison ou le critère final est atteint ou non par rapport à la valeur du facteur de la qualité;
4. Sinon, une paire de solutions est choisie selon le facteur de qualité calculé à l'étape 2 pour la reproduction de nouvelles solutions;
5. Placer les nouvelles solutions produites dans le nouvel ensemble;
6. Vérifier si la taille du nouvel ensemble de solutions est correcte, sinon retourner à l'étape 4;
7. Remplacer l'ancien ensemble de solutions par le nouvel ensemble;
8. Retourner à l'étape 2;
9. Répéter jusqu'à l'atteinte de la taille N du nouvel ensemble.

Le critère de terminaison cité à l'étape 3 est atteint lorsqu'une certaine condition ou un certain paramètre est obtenu par exemple lorsque le facteur de qualité calculé dépasse un certain seuil ou une certaine valeur.

Ces algorithmes génétiques sont utilisés dans différents domaines, incluant le domaine aéronautique. Dans le modèle de Kobayashi et al. (2005), l'algorithme génétique utilisé a permis d'assurer la robustesse des mesures, de gérer la grande taille des données d'entraînement des réseaux de neurones et de minimiser les erreurs entre les données prédites par les réseaux de neurones et les données réelles du moteur.

Un modèle d'une voilure avec sa surface déformable supérieure (Korenschi, Andreea et al., 2016) a été conçu en se basant sur un algorithme génétique pour l'optimisation de ses performances aérodynamiques. Cet algorithme génétique a permis d'optimiser la forme de la surface supérieure de la voilure en utilisant les actionneurs responsables de cette déformation. En effet, le grand avantage d'un algorithme génétique est sa capacité de trouver une solution optimale pour des problèmes complexes. La meilleure des deux solutions possibles est choisie. Des problèmes considérant un très grand nombre de paramètres peuvent être résolus par ce

type d'algorithmes dans quelques itérations. Mais les algorithmes génétiques manipulent plusieurs solutions en même temps, ainsi ils sont coûteux en temps de calcul.

1.6.2 Le recuit simulé

Le "recuit simulé" est une autre méthode méta-heuristique qui résout des problèmes d'optimisation difficiles. La définition du "recuit" se résume par le fait de laisser refroidir lentement un métal pour améliorer ses qualités. Le refroidissement lent de ce métal permet à ses molécules de se regrouper dans une configuration plus stable que dans celle initialement. Le "recuit simulé" inspiré par ce principe a été proposé par Kirkpatrick et al. (1983) en se basant sur l'algorithme de Metropolis et al. (1953). En effet, le "recuit simulé" possède des paramètres d'entrée de l'algorithme qui sont la "température initiale" et la "vitesse de refroidissement". A chaque itération la température diminue car le métal est entrain de se refroidir. Il présente une facilité et rapidité d'implémentation ainsi que d'adaptation qui sont utiles pour la résolution des différents problèmes. Le choix manuel des paramètres initiaux de cet algorithme est l'un de ses inconvénients. Également, la modification d'un seul paramètre à la fois a le désavantage de ralentir l'algorithme (Ben Mosbah, 2011). Les étapes du "recuit simulé" sont les suivantes :

1. Initialiser la température T ;
2. Modifier l'énergie du système avec ΔE ;
3. Utiliser le critère de Metropolis : - si $\Delta E < 0$, la modification est acceptée, sinon elle est acceptée avec une probabilité déterminée;
4. Vérifier si l'équilibre thermodynamique est atteint, sinon retourner à l'étape 2;
5. Vérifier si le système est figé (optimisé), sinon continuer à réduire la température et retourner à l'étape 2;
6. Fin de l'algorithme.

L'influence de la vitesse de diminution de température influe énormément la performance de l'algorithme du recuit simulé car si l'algorithme atteint un minimum local à une basse température, la possibilité d'obtenir le minimum global est difficile.

Cet algorithme a été utilisé dans plusieurs domaines d'applications notamment l'aérospatiale grâce à sa capacité de résoudre les problèmes d'optimisation difficiles. En effet, un modèle proposé par Hung et al. (2008) utilise l'algorithme du recuit simulé pour la conception d'un contrôleur PID (Proportionnel Intégral Dérivé) pour les systèmes qui contiennent des incertitudes et des perturbations. Le problème d'optimisation proposé avait les trois objectifs suivants : la stabilité de la robustesse, l'atténuation des perturbations et des erreurs. Ce modèle a présenté de bons résultats suite à son application à un système d'un avion de combat.

L'algorithme du "recuit simulé" a été utilisé aussi pour résoudre le problème des rotations planifiées des membres de l'équipage d'un avion (Lucic et al., 1999). Ce modèle a été testé pour des petits et moyens transporteurs aériens et ses résultats étaient bons.

1.6.3 La recherche tabou

La "recherche tabou" est une méthode d'optimisation proposée par Glover (1986). Elle est une méthode heuristique de recherche locale utilisée pour la résolution des problèmes complexes ou de grandes tailles. Son principe de base est de poursuivre la recherche de la solution même si un optimum local est rencontré. Cette méthode utilise une mémoire à court terme (ou "liste tabou") qui empêche le processus de recherche de trouver une solution déjà obtenue. Cette "liste tabou" contient les différentes solutions interdites trouvées au cours de la recherche. Si la taille de cette liste est importante, alors les solutions voisines ou proches sont inaccessibles. Par contre si la taille de la "liste tabou" est petite, il y a un risque de blocage de la solution dans un "minimum local". Donc, la taille de la liste est un paramètre très important. Cependant, il y a des exceptions par lesquelles le processus de recherche ne respecte pas la liste tabou si une solution de très bonne qualité est obtenue en considérant surtout un "mouvement tabou" ("critère d'aspiration") (Ben Mosbah, 2011). Ce critère permet d'accepter un mouvement si celui-ci conduit à une configuration meilleure que celle déjà trouvée.

Cette technique d'optimisation est facile à implémenter car elle permet de gagner du temps de résolution même pour des problèmes de grandes tailles et donne également de bons résultats

pour certains cas. En revanche, la "recherche tabou" n'assure pas une convergence vers un optimum global, ce qui mène à régler délicatement la taille de la "liste taboue" en fonction de plusieurs critères qui varient d'un problème à un autre. En effet, une demande en ressources importante car la "liste taboue" est trop imposante (Moumen, 2011). Les étapes de cet algorithme sont explicitées ci-dessous : (Kammakh et al., 2014)

1. Choisir une solution initiale;
2. Créer une liste de solutions possibles;
3. Choisir la meilleure solution en tenant compte de la liste taboue et du critère d'aspiration qui permet d'accepter un "mouvement tabou";
4. Obtenir une nouvelle solution qui sera enregistrée dans le cas où elle sera meilleure que la précédente solution;
5. Appliquer un critère d'arrêt. Il faudra arrêter le fonctionnement de l'algorithme, sinon il faudra retourner à l'étape 2.

Le "critère d'arrêt" pourrait être par exemple un nombre d'itérations prédéterminé à l'avance. La "recherche Tabou" est appliquée surtout dans les problèmes d'ordonnancement cellulaire. Il y a également un modèle d'optimisation paramétrique d'un avion qui a été conçu à l'aide de l'algorithme de la "recherche Tabou" (Zhiping et al., 2010).

1.6.4 Le grand déluge étendu

L'algorithme du grand déluge étendu est une technique de recherche locale proposée par Dueck (1993). Son principe se base sur l'acceptation de certaines mauvaises solutions ne dépassant pas une limite B prédéfinie. Cette limite B décroît avec un pas ΔB fixe (pour les problèmes de minimisation) et augmente dans le cas des problèmes de maximisation. Le pas ΔB est le paramètre à fixer au début de l'algorithme.

Les résultats obtenus dans le modèle de Burke et al. (2004) ont montré que le temps de convergence de l'algorithme dépendait essentiellement de ΔB . Si ΔB augmentait, le temps de convergence pouvait diminuer dans le cas des problèmes de minimisation, alors qu'il y avait un risque de dégradation de la solution.

En effet, la limite B représente une séparation entre l'espace de recherche réalisable et celui non réalisable. Autrement dit, cette limite B sert à obtenir la solution dans l'espace réalisable puisqu'elle coupe le voisinage de cette solution. Ensuite, la recherche de la solution se fait d'un seul côté (en dessous ou en dessus de la limite) selon la fonction objective (minimisation ou maximisation). Plus précisément, cette limite B rapproche de plus en plus à la valeur de la solution recherchée ce qui réduit l'espace de recherche et diminue la possibilité d'amélioration de la solution, donc le processus est finalisé (Ben Mosbah, 2011).

L'avantage de cet algorithme est le fait de régler un seul paramètre qui est ΔB . Il faut bien évidemment très bien choisir ce paramètre afin de garantir la qualité de la solution et le gain du temps de calcul. Les étapes de l'algorithme sont présentées ci-dessous :

1. Définir la solution initiale S ainsi que le pas ΔB ;
2. Calculer l'efficacité de S : $f(S)$;
3. Associer l'efficacité $f(S)$ à B : $B = f(S)$;
4. Définir le voisinage de la solution : $N(S)$;
5. Sélectionner une solution aléatoire appartenant à ce voisinage : $S^* \in N(S)$;
6. Vérifier la condition : - si $f(S^*) \leq f(S)$ ou $f(S) \leq B$ alors accepter S^* , diminuer la limite B : $B = B - \Delta B$ et vérifier le critère d'arrêt;
 - sinon revenir à l'étape 4;
7. Fin de l'algorithme.

Cet algorithme a été appliqué pour résoudre le problème d'optimisation des horaires des examens (Burke et al., 2004) ainsi que pour modéliser le contrôleur des actionneurs de l'aile déformable de l'avion ATR-42 (Mosbah et al., 2016). Les résultats obtenus pour ce modèle de l'ATR-42 réalisé à base de réseaux de neurones et de l'algorithme grand déluge étendu ont été meilleurs que ceux obtenus pour un contrôleur PID, ce qui met en évidence l'efficacité de l'algorithme cité précédemment.

A travers ce chapitre, nous avons présenté les différents types des moteurs d'avions, ainsi que les différentes méthodologies utilisées pour la conception des modèles d'identification de leurs paramètres. Également, nous avons présenté quelques algorithmes d'optimisation qui ont montré une grande efficacité dans le cadre des améliorations de certains modèles. Après avoir étudié plusieurs modèles de moteurs, nous avons choisi de réaliser le modèle du moteur de l'avion Cessna Citation X en se basant sur des réseaux de neurones et de l'algorithme du grand déluge étendu (EGD). Ce choix a été fait grâce aux avantages présentés par les réseaux de neurones pour le traitement des systèmes non linéaires et grâce à la facilité de l'implémentation de l'algorithme EGD et sa rapidité de convergence qui pourrait être assurée en combinaison avec les réseaux de neurones. Le chapitre suivant expliquera les réseaux de neurones en détails ; leur fonctionnement et leur principe d'apprentissage. Le troisième chapitre donnera les détails de la réalisation du modèle du moteur de Cessna Citation X. Finalement, le quatrième chapitre présentera les résultats obtenus pour les différents paramètres identifiés grâce au modèle.

CHAPITRE 2

INTRODUCTION AUX RÉSEAUX DE NEURONES

L'apprentissage des machines se fait grâce à des mécanismes adaptifs : soit par expérience, par exemple ou par analogie. En effet, les deux moyens les plus utilisés pour l'apprentissage des machines sont les algorithmes génétiques et les réseaux de neurones (Cormier, 2005).

Un "neurone" est exprimé par sa fonction non linéaire, paramétré, de ses variables d'entrée. L'intérêt des études par des neurones réside grâce à leurs propriétés qui résultent de leur association en réseaux c'est-à-dire de la composition des fonctions non linéaires réalisées par chacun des neurones (Dreyfus et al., 2002).

Les réseaux de neurones artificiels sont comparés aux réseaux de neurones biologiques qui sont de très petits éléments permettant de traiter les influx nerveux qui arrivent (les entrées), chacun selon son importance relative, et qui émettent un seul signal de sortie. Ce sont des éléments importants traités dans le domaine de l'intelligence artificielle. L'objectif essentiel de ces réseaux est de reproduire des raisonnements intelligents d'une manière artificielle grâce à des fonctions mathématiques parfois complexes. La reproduction du fonctionnement d'un système au moyen des réseaux de neurones doit être simple et logique. Les réseaux de neurones sont appliqués dans plusieurs domaines. En effet, un auto pilote pour avion, des systèmes de lecture d'adresses postales et des chèques bancaires, des systèmes de traitement de signal, des systèmes de vision par ordinateurs, des systèmes de d'exploration pétrolière... ont été développés en utilisant les réseaux de neurones (Parizeau, 2006).

Dans cette section, la modélisation mathématique d'un neurone est présentée en premier lieu. Puis, les principes de fonctionnement et les algorithmes d'apprentissage des réseaux sont expliqués. Et finalement, la validation croisée est explicitée.

2.1 Le modèle mathématique

Les réseaux de neurones sont composés de fonctions mathématiques et des unités de calcul connectées les unes aux autres ayant l'objectif d'effectuer des opérations complexes grâce aux fonctions d'apprentissage.

Chaque neurone est composé :

- d'une entrée représentée par un vecteur de plusieurs éléments;
- d'un poids associé à chaque entrée;
- d'un biais ajouté à l'entrée;
- d'une fonction de transfert;
- d'une sortie scalaire.

Un neurone artificiel présenté dans la figure 2.1 représente une fonction algébrique paramétrée.

Pour calculer la valeur de cette fonction, nous pouvons procéder en deux étapes comme suit :

- 1) La "combinaison linéaire des entrées" :

$$U = W_0 + \sum W_{ij}X_i \quad (2.1)$$

Où U est le potentiel du neurone

W_0 est le biais qui est considéré comme la pondération de l'entrée 0 fixée à 1

W_{ij} sont les poids

X_i sont les entrées

- 2) La "sortie du neurone" est exprimée par la relation suivante :

$$Y = f(U) = f(\sum W_{ij}X_i) \quad (2.2)$$

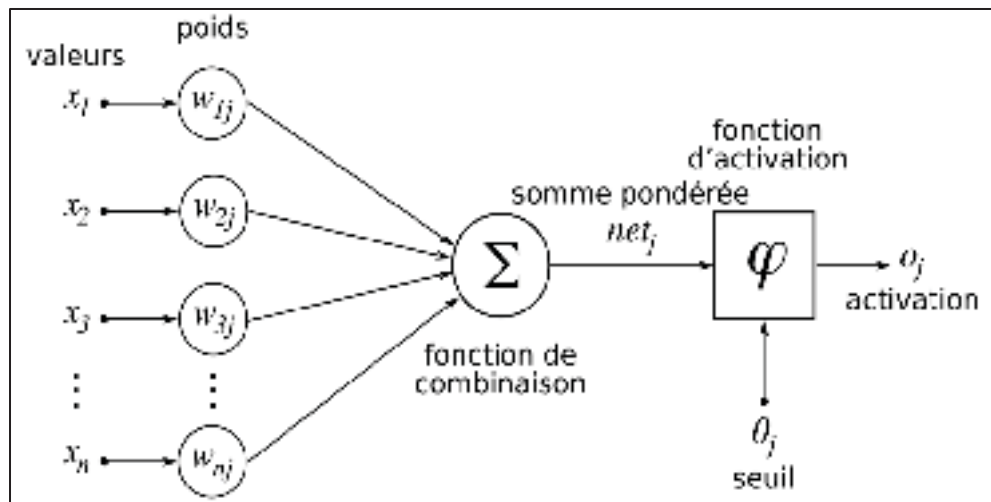


Figure 2.1 Définition d'un neurone artificiel

2.2 Architecture des réseaux de neurones

L'architecture générale des réseaux de neurones consiste en la représentation des neurones en couches successives : la première couche représente la couche des entrées, alors que la dernière est la couche de sortie et pour les couches intermédiaires, ce sont les couches cachées du réseau puisque nous ne connaissons pas vraiment leur fonctionnement. Les neurones de la couche des entrées ainsi que ceux de la couche de sortie sont connus.

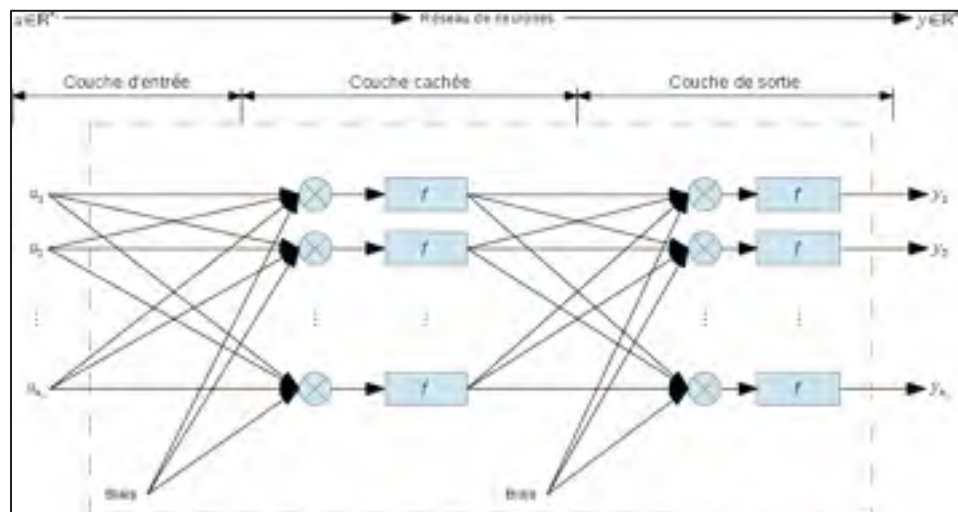


Figure 2.2 Architecture d'un réseau de neurone

Chaque couche cachée peut être constituée de plusieurs neurones. Chaque sortie de neurones d'une couche est reliée à l'ensemble des entrées de la couche suivante des neurones.

Des paramètres de l'architecture du réseau de neurones peuvent être optimisés comme le nombre de couches cachées et le nombre de neurones. Un modèle de réseau de neurones a été proposé par S.McCulloch et al. (1943) utilisant la sommation pondérée des entrées puis appliquant une fonction d'activation à seuil appelé Θ . La fonction d'activation multiplie chaque entrée par son poids correspondant et effectue la somme présentée dans l'équation 2.3 (Cormier, 2005) :

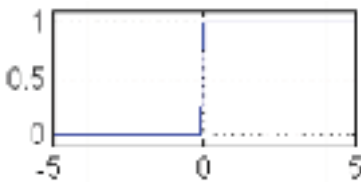
$$X = \sum_{i=1}^n x_i w_i^2 \quad (2.3)$$

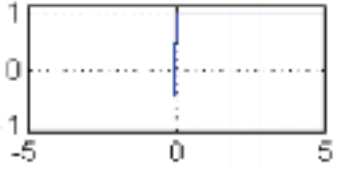
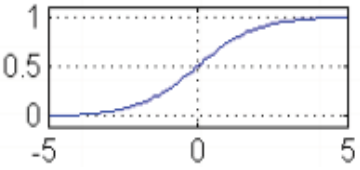
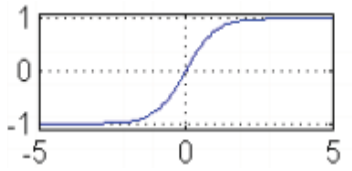
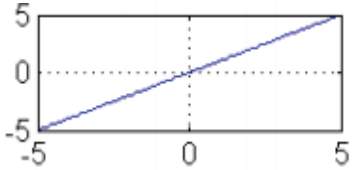
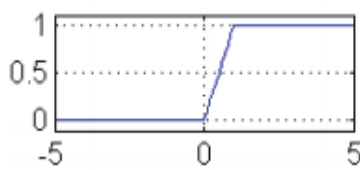
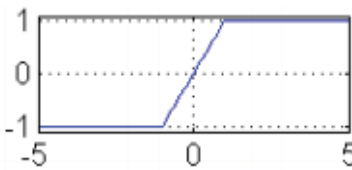
Si l'entrée nette du système est moins que la valeur du seuil, la sortie du neurone est -1, sinon le neurone est activé et la sortie du neurone est +1 :

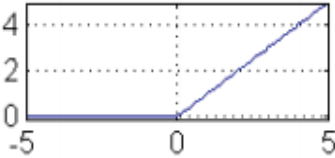
$$Y = \begin{cases} +1 & \text{si } X \geq \theta \\ -1 & \text{si } X \leq \theta \end{cases} \quad (2.4)$$

La fonction d'activation peut être linéaire ou non. Le choix de la fonction à utiliser, dépend du problème à résoudre, ainsi de sa représentation. Ces fonctions sont présentées dans le tableau 2.1 ci-dessous (Parizeau, 2006).

Tableau 2.1 Les différentes fonctions d'activation

Nom de la fonction	$Y = f(X)$	Représentation	Fonction Matlab
Seuil	$\begin{cases} Y = 0 & \text{si } X < 0 \\ Y = 1 & \text{si } X \geq 0 \end{cases}$		Hardlim

Seuil symétrique	$\begin{cases} Y = -1 & \text{si } X < 0 \\ Y = 1 & \text{si } X \geq 0 \end{cases}$		hardlims
Sigmoïde	$Y = \frac{1}{1 + \exp^{-X}}$		logsig
Tangente hyperbolique	$Y = \frac{e^X - e^{-X}}{e^X + e^{-X}}$		tansig
Linéaire	$Y = X$		purelin
Linéaire saturée	$\begin{cases} Y = 0 & \text{si } X < 0 \\ Y = X & \text{si } 0 \leq X \leq 1 \\ Y = 1 & \text{si } X > 1 \end{cases}$		satlin
Linéaire saturée symétrique	$\begin{cases} Y = -1 & \text{si } X < -1 \\ Y = X & \text{si } -1 \leq X \leq 1 \\ Y = 1 & \text{si } X > 1 \end{cases}$		satlins

Linéaire positive	$\begin{cases} Y = 0 \text{ si } X < 0 \\ Y = X \text{ si } X \geq 0 \end{cases}$		Poslin
----------------------	---	--	---------------

Les fonctions de transfert les plus utilisées comme fonctions d'activation sont :

- La fonction seuil;
- La fonction linéaire;
- La fonction sigmoïde.

Concernant la "fonction seuil", elle applique un seuil sur son entrée : c'est-à-dire une entrée négative qui ne dépasse pas le seuil donc la sortie retourne 0. Si une entrée est positive ou nulle, elle dépasse le seuil et la sortie retourne 1. La fonction linéaire est simple, elle affecte l'entrée à sa sortie. La fonction sigmoïde est un compromis entre les deux fonctions précédentes.

2.3 Perceptron multicouche

Le "perceptron multicouche" se trouve parmi les classes importantes des réseaux de neurones. En effet, il permet de résoudre les problèmes d'identification, de la classification et de contrôle des systèmes dynamiques non linéaires. Le perceptron multicouche contient des couches cachées composées de plusieurs neurones. Son apprentissage est de type "supervisé" avec la correction des erreurs. Un exemple de perceptron multicouches est présenté ci-dessous :

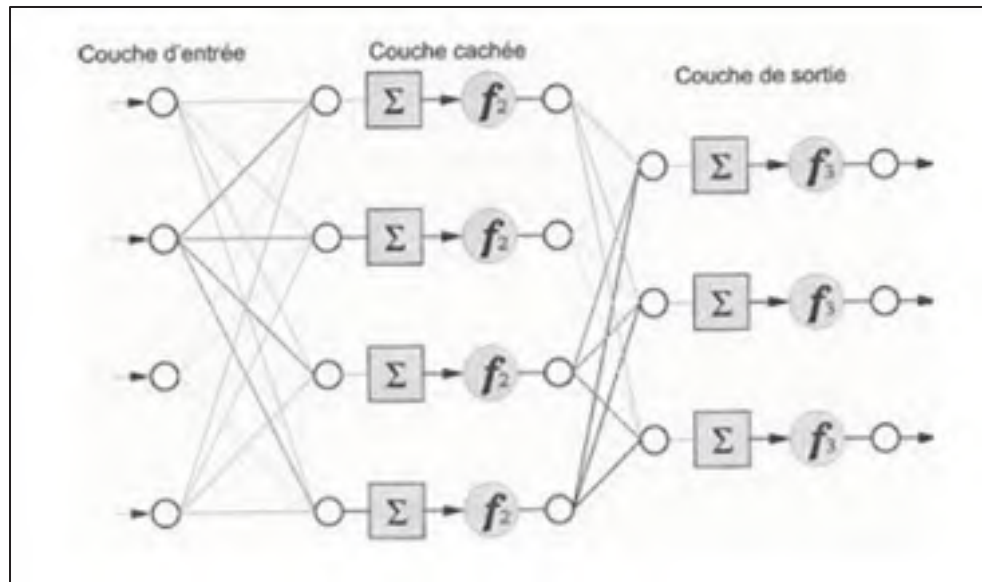


Figure 2.3 Perceptron multicouches avec une couche cachée

Le "perceptron multicouche" utilise une fonction d'activation généralement non linéaire pour tous les neurones. Les poids et les biais du réseau sont deux paramètres qui influent son comportement. Donc, l'apprentissage est le processus qui permet d'obtenir les paramètres optimaux d'un réseau.

2.4 L'apprentissage des réseaux de neurones

"L'apprentissage" est une phase qui permet le développement du réseau de neurones durant laquelle le comportement de ce réseau est modifié jusqu'à l'obtention du comportement désiré. Il considère la mise à jour des poids des connexions au sein du réseau pour réussir la tâche qui lui est demandée. Ainsi, l'apprentissage est une caractéristique principale des réseaux de neurones qui peut se faire de différentes manières et selon plusieurs règles.

Les règles d'apprentissage sont les suivantes :

- **Règle de correction d'erreurs** : elle est classée dans l'apprentissage supervisé lorsque nous introduisons dans le réseau une entrée et sa sortie correspondante. Nous considérons (y) la sortie calculée par le réseau et (d) la sortie désirée.

Le principe de cette règle est d'utiliser l'erreur ($d-y$) pour modifier les connexions et diminuer l'erreur globale du système. Par la suite, le réseau va s'adapter jusqu'à ce que (y) soit égal à (d), c'est-à-dire jusqu'à ce que l'erreur soit nulle.

- **Apprentissage de Boltzmann** : les réseaux de Boltzmann sont des réseaux symétriques récurrents. Ils possèdent deux sous-groupes de cellules :
 - ✓ Cellules visibles;
 - ✓ Cellules cachées.

Ce type d'apprentissage fonctionne de deux façons différentes ; la première façon est le mode figé où une valeur de l'environnement est affectée aux cellules visibles. L'autre façon est le mode libre évolution où les cellules (cachées ou visibles) peuvent changer d'état librement. Cette règle d'apprentissage relève partiellement du hasard et consiste à ajuster les poids des connexions d'une façon aléatoire pour satisfaire une distribution probabiliste souhaitée des cellules visibles (Volpi, 2016).

- **Règle de Hebb** : cette règle est basée sur des données biologiques, elle permet de modéliser le fait que si des neurones sont activés de façon synchrone et répétée, l'efficacité de transmission des signaux de ces neurones augmente. L'apprentissage ici est "localisé" c'est-à-dire que la modification d'un poids synaptique (w_{ij}) dépend seulement de l'activation d'un neurone i et d'un autre neurone j (Volpi, 2016).
- **Règle d'apprentissage par compétition** : Cet apprentissage concerne un seul neurone. Ceci correspond au regroupement des données du système à modéliser en catégories. Dans un réseau à compétition simple, chaque neurone de la couche de sortie est connecté aux neurones de la couche d'entrée, aux autres neurones de la couche de sortie (connexions "inhibitrices") et à lui-même (connexions "excitatrices"). En effet, la sortie du réseau de neurones va dépendre de la compétition entre les connexions inhibitrices et les connexions excitatrices.

L'apprentissage s'effectue grâce à la minimisation d'une fonction de sortie qui est calculée à partir de la base des données réservée à l'apprentissage et de la sortie du réseau de neurones (Stricker, 2004). Cette minimisation se réalise à l'aide des méthodes non-linéaires d'optimisation tout en réduisant l'écart entre les réponses réelles du réseau et les réponses désirées.

Il existe deux types d'apprentissage :

- Apprentissage supervisé;
- Apprentissage non supervisé.

Concernant un apprentissage **supervisé**, le réseau s'adapte par comparaison entre la sortie qu'il a calculé, en fonction des entrées fournies et la sortie désirée du système à modéliser. Ainsi, le réseau va se modifier (en boucle fermée) jusqu'à ce qu'il trouve la bonne sortie attendue qui correspond à une entrée donnée. Pour le mode "d'apprentissage supervisé", il y a le "renforcement" qui permet au réseau d'apprendre la corrélation ("entrée/sortie") à travers une estimation de son erreur (rapport échec/succès). Par suite, le réseau va tendre à maximiser un index de performance qui lui est fourni (signal de renforcement). En utilisant ce signal, le système est capable de savoir si la réponse qu'il fournit est correcte ou non, toutefois il ne connaît pas la bonne réponse. Tandis que pour l'apprentissage **non supervisé**, le système est basé sur des probabilités, ceci veut dire que le réseau se modifie en fonction des régularités statistiques de l'entrée et établit l'ordre des catégories en attribuant et en optimisant une valeur de qualité aux catégories reconnues.

Il existe aussi deux modes essentiels d'apprentissages qui sont :

- L'apprentissage "en ligne";
- L'apprentissage "hors ligne".

Pour l'apprentissage "en ligne", chaque valeur correspondante à une variable est jetée après son entraînement et aussi la mise à jour des paramètres. Ce type d'apprentissage ne permet pas de réaliser les calculs précédents car les données ne sont pas enregistrées. Son mode de fonctionnement tient compte des deux phases d'apprentissage et d'utilisation du système simultanément. En effet, le système continue à apprendre tout le long de son utilisation jusqu'à

obtenir une donnée qui permettra l'amélioration de ses performances. Tandis que pour l'apprentissage "hors ligne", les phases d'apprentissage et d'utilisation sont séparés. Ce type d'apprentissage permet au système d'être entraîné d'une manière statique (les données sont prédéfinies dès le début de l'entraînement) ou incrémentale (les données peuvent être introduites au fur et à mesure de l'entraînement) à partir d'une base de données servie à l'entraînement. Puis, le système appris sera utilisé (Bouillon, 2012). La minimisation de l'erreur entre la sortie réelle du réseau de neurones et celle désirée du système est obtenue en appliquant l'algorithme de rétro-propagation du gradient.

2.4.1 L'algorithme de rétro-propagation du gradient

La fonction de l'algorithme de rétro-propagation du gradient est de propager les erreurs obtenues à la sortie vers les couches cachées afin de minimiser l'erreur quadratique moyenne d'apprentissage. C'est une technique d'apprentissage supervisée

L'erreur à la sortie d'un neurone est donnée par la différence entre la valeur de la sortie désirée et la valeur réelle donnée par le neurone :

$$e = y_{désirée} - y \quad (2.5)$$

Où e : l'erreur calculée

$y_{désirée}$: la réponse désirée du système

y : la réponse réelle du neurone

Les étapes de cet algorithme sont les suivantes : (Doncescu, 2007)

- 1- Initialisation des poids des liens entre les neurones. Le poids peut avoir une valeur entre 0 et 1 qui est déterminée aléatoirement;
- 2- Application des entrées-sorties à introduire au réseau;
- 3- Calculs des réponses du réseau de neurones correspondantes aux entrées introduites;
- 4- Calculs de l'erreur entre les réponses du réseau et les réponses désirées du système;

- 5- Correction des poids des liens entre les neurones de la couche de sortie et de la première couche cachée, tout en tenant compte de l'erreur calculée à l'étape précédente;
- 6- Propagation de l'erreur sur la couche précédente et correction des poids des liens entre les neurones de la couche cachée et ceux de la couche d'entrée;
- 7- Retour à la deuxième étape si l'erreur n'est pas acceptable, sinon finaliser l'algorithme.

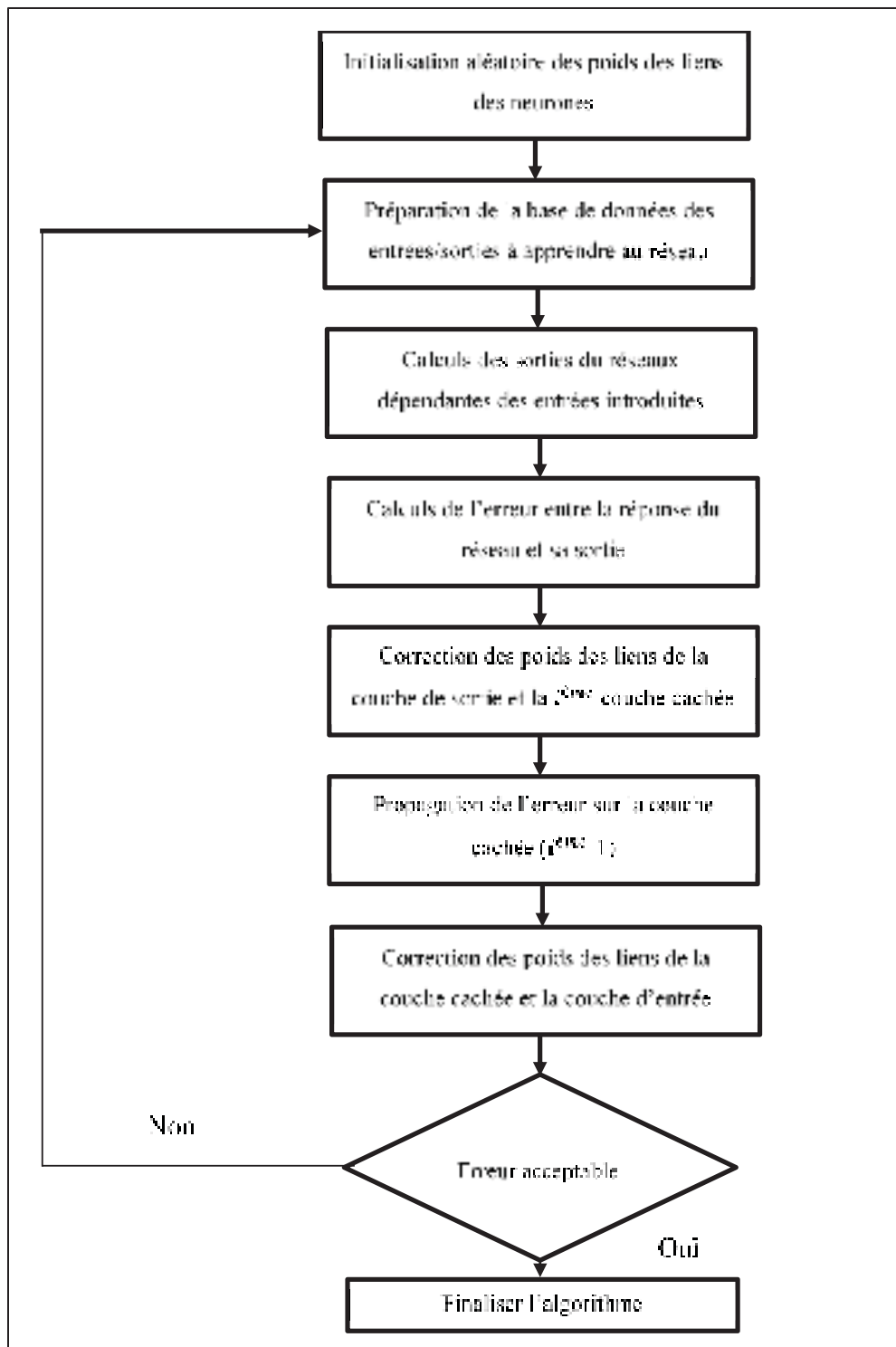


Figure 2.4 Algorithme de rétro-propagation du gradient
Adapté de Doncescu, (2007)

Cette méthode de rétro-propagation peut prendre beaucoup de temps avant que la somme des erreurs carrées ne soit réduite à zéro ce qui est un cas presque impossible. Pour cette raison, un critère d'erreurs devrait être mis en place et contenant une valeur limite à ne pas dépasser par la somme des erreurs carrées (LaBarre, M. O., 2002).

L'entraînement de perceptron multicouche avec cet algorithme de rétro-propagation du gradient se fait de la même manière que l'entraînement d'un perceptron à un seul neurone. Pour ce faire, il faut calculer le gradient d'erreur pour chaque neurone et par suite calculer la variation des poids pour les liens rattachés à ce neurone. Après, la couche précédente sera traitée de la même manière.

L'utilisation de la fonction « sigmoïde » pour cet algorithme est justifiée par le fait qu'il n'existe pas une réponse correcte ou non, la réponse aurait une valeur entre 0 et 1 (c'est-à-dire si la valeur est 0, elle ne veut pas dire qu'elle est mauvaise mais acceptable et elle pourrait être améliorée et tendre vers 1, qui est la valeur idéale). En effet, même si les réponses sont acceptables mais non satisfaisantes, il reste une petite marge d'erreur qui serait corrigée par l'algorithme en modifiant les poids pour que la réponse tend vers 1 (l'idéal).

2.4.2 L'algorithme Gauss-Newton

Cet algorithme est utilisé pour résoudre les problèmes des moindres carrés non linéaires. Il permet de trouver le "minimum" d'une fonction. Il permet la minimisation de la somme de fonctions au carré sans avoir recours aux valeurs des dérivées secondes qui sont complexes. La méthode Gauss-Newton permet de minimiser $F(x)$ qui représente la somme des carrés des fonctions non linéaires : (Magnani et al., 2009)

$$F(x) = \sum_{i=1}^m |f_i(x)|^2 \quad (2.6)$$

Cette méthode permet de calculer une séquence de solutions approximatives x_k à travers la linéarisation de f_i autour de x_k et de déterminer x_{k+1} comme solution au problème linéaire des moindres carrés. Elle ne garantit pas aussi la convergence pour des fonctions f_i non linéaires. L'algorithme de rétro-propagation ne permet pas une convergence rapide de l'erreur, alors que la méthode de Newton assure une convergence rapide qui nécessite le calcul de l'inverse du Hessien qui est complexe. Afin de résoudre ces problèmes, l'algorithme de Levenberg-Marquardt propose une approximation du Hessien tout en utilisant la méthode Newton-Gauss qui assure la rapidité et la précision de la solution autour du minimum.

2.4.3 Algorithme Levenberg-Marquardt

Cet algorithme est une amélioration de l'algorithme utilisé par la méthode Gauss-Newton. Il est utilisé généralement pour les réseaux qui servent à la prédiction des données et ceci grâce à sa capacité de trouver une solution numérique au problème de minimisation de l'erreur, plus précisément pour les systèmes non linéaires et dépendant de plusieurs variables. Il permet aussi de trouver une solution, même si, elle est loin du minimum. L'algorithme Levenberg-Marquardt optimise l'erreur quadratique tout en assurant sa rapidité et robustesse. Il fonctionne selon les techniques des moindres carrés non-linéaires et de l'algorithme de Gauss-Newton à voisinage restreint.

$$\Delta W = -(J^T \cdot J + \mu \cdot I)^{-1} \cdot J^T e \quad (2.7)$$

Où ΔW est le taux de mise à jour de la matrice des paramètres du réseau

J est la matrice Jacobienne de la fonction erreur quadratique

e est le vecteur d'erreur

I est la matrice identité

μ est le coefficient adaptatif et dépendant de l'évolution de l'apprentissage

Si μ est grand alors l'algorithme est équivalent à la descente du gradient avec un taux d'apprentissage qui est faible, et si, μ est nul alors c'est la méthode de Newton-Gauss qui est

appliquée en utilisant une approximation de Hessien. L'objectif de la méthode Levenberg-Marquardt est de s'approcher de la méthode de Newton le plus rapidement possible puisque cette dernière présente une très grande précision et rapidité au voisinage d'un minimum (Girard, A., 2007).

L'initialisation des poids de neurones lors de l'apprentissage par l'algorithme de Levenberg-Marquardt doit être très bien étudiée. En effet, une mauvaise initialisation peut empêcher d'atteindre un optimum local.

Les étapes de l'algorithme de Levenberg-Marquardt sont expliquées en (Rondepierre et al., 2013) et (More, 1977). Tout d'abord, il faudrait introduire les données à apprendre par les réseaux et initialiser le coefficient μ suivant le type du problème à résoudre. Ensuite, calculer une solution de l'équation 2.7 afin d'obtenir la matrice J de la fonction de l'erreur quadratique et le vecteur d'erreur e . Le pas Δ de Levenberg-Marquardt est calculé par la suite pour avoir la nouvelle solution de l'équation 2.7 permettant d'avoir les paramètres optimaux du réseau.

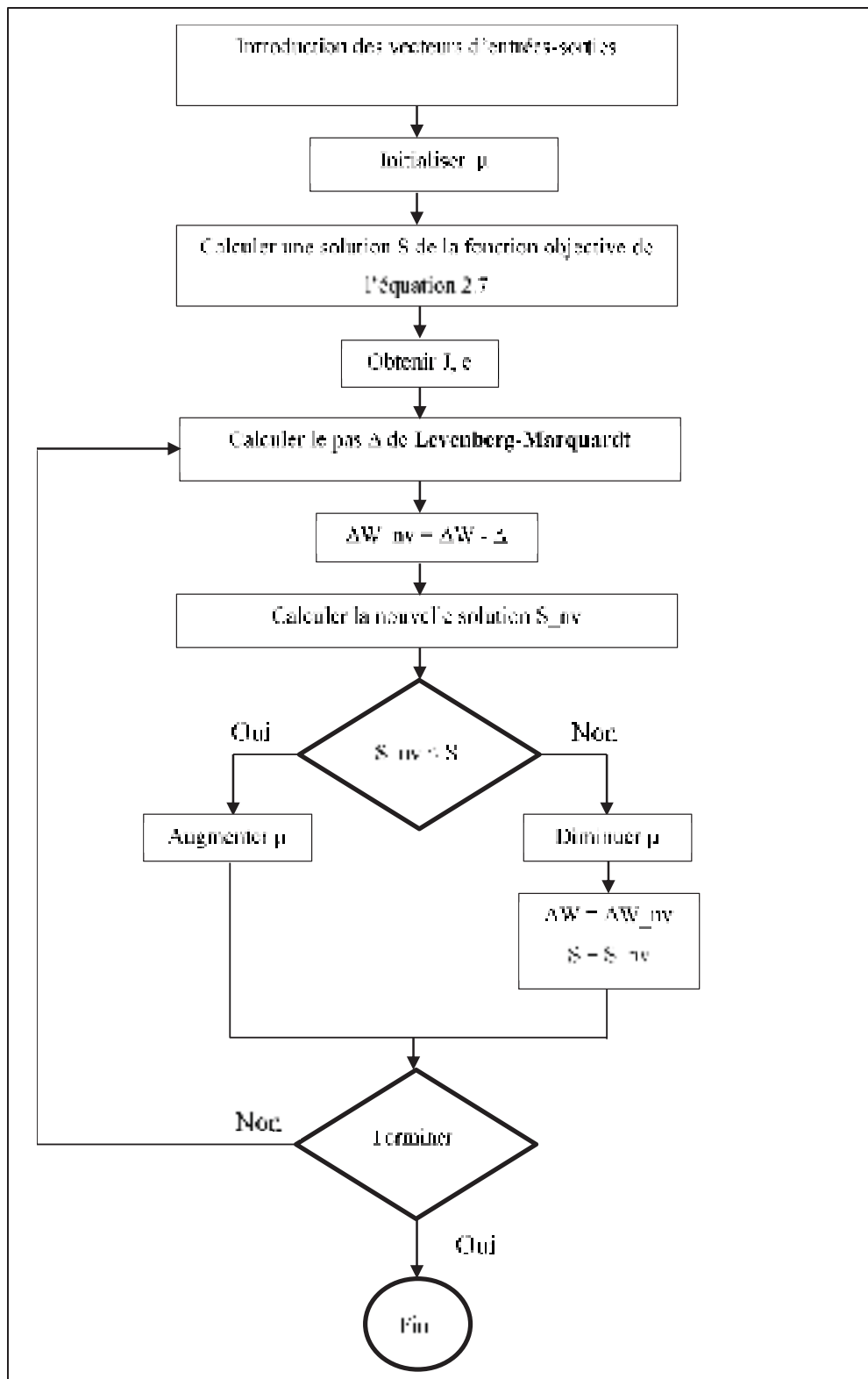


Figure 2.5 Algorithme de Levenberg-Marquardt

Les données à introduire au réseau de neurones pour l'apprentissage, ainsi que le nombre des époques durant lesquels le réseau s'entraîne sont des éléments très importants. En effet, ils influent la performance finale du réseau. La non-linéarité des réseaux de neurones impose une quantité suffisante de données d'entrées afin de garantir une identification correcte des systèmes.

Après avoir réalisé l'apprentissage du réseau pour identifier un système, il faut bien évidemment le valider. Pour ce faire, des méthodes de validation existent comme la "validation croisée", le "ré-échantillonnage" (bootstrap), la "validation par découpage". Mais, étant donné que les données ne sont pas assez suffisantes pour utiliser certaines méthodes de validation, la validation croisée est la plus utilisée.

2.5 La validation croisée

Cette méthodologie de validation permet de tester la précision d'un modèle dans un échantillon test par rapport à sa précision dans un échantillon d'apprentissage. En effet, l'utilisation des données non servies à l'apprentissage pour le test du réseau permet de déterminer l'erreur moyenne du modèle. De ce fait, plus cette erreur est faible et plus le modèle du réseau est fiable.

Le principe de ce type de validation est de découper le jeu des données en K groupes sélectionnés aléatoirement qui vont servir aux tests. Ensuite, l'erreur est calculée pour chaque groupe et une moyenne de ces erreurs sera déduite. Cette moyenne sera comparée à la moyenne calculée par le critère d'acceptation qui est fixé selon le système étudié dans le but de juger si le modèle réalisé par le réseau de neurones est valide ou non. Le pourcentage des données non utilisées pour l'apprentissage qui doit être conservé pour la validation est 25% pendant que 75% des données sont utilisés pour l'apprentissage (Haykin, 1999).

À travers ce chapitre, nous avons présenté tout d'abord les réseaux de neurones, leur fonctionnement et leur architecture. Ensuite, nous avons exposé le principe d'apprentissage

basé sur ces réseaux de neurones et les différents algorithmes utilisés. Cette étude présentée dans ce chapitre nous a permis de choisir l'algorithme d'apprentissage que nous allons utiliser pour la conception du modèle du moteur. Ce choix s'est basé sur les avantages que présente l'algorithme de Levenberg-Marquardt. Enfin, nous avons présenté la méthode de la validation croisée qui servira à la validation du modèle. Les chapitres 3 et 4 présenteront la méthodologie de conception du modèle permettant d'identifier les paramètres du moteur de l'avion Cessna Citation X et les résultats obtenus pour chacune des étapes de cette méthodologie.

CHAPITRE 3

IDENTIFICATION DES PARAMÈTRES DU MOTEUR

Un objectif essentiel du processus d'identification est d'obtenir un modèle du moteur qui représente d'une manière précise le fonctionnement et le comportement du moteur de l'avion Cessna Citation X lors du vol. En effet, la méthode des réseaux de neurones est utilisée puisqu'elle traite les problèmes d'identification des systèmes non linéaires.

Ce chapitre présente la méthodologie du travail réalisé afin d'obtenir le modèle du moteur qui permet de prédire les paramètres intervenant dans son fonctionnement. Nous introduisons en premier lieu les données servant à l'apprentissage et leur récupération. En second lieu, nous expliquons la méthodologie de conception du modèle. Et finalement, nous présentons la méthode d'apprentissage des données.

3.1 Les données d'apprentissage

Comme il a été mentionné précédemment, les réseaux de neurones nécessitent d'avoir une grande base de données pour garantir une très bonne précision de la prédiction des paramètres. Ce qui fait que, plus la dimension de la couche d'entrées est grande, plus l'apprentissage nécessite un plus grand ensemble de données. Ces données devraient être traitées avant l'apprentissage en fonction de leur usage.

Toutes les données utilisées sont récupérées à partir du simulateur de vol disponible au laboratoire LARCASE. Ces données doivent représenter les fonctions qui relient l'ensemble des paramètres du modèle du moteur à prédire. Afin de gagner du temps sur l'apprentissage, il faut éliminer la redondance des données.

3.1.1 Le simulateur de vol

Le laboratoire LARCASE est équipé d'un simulateur de vol de l'avion Cessna Citation X fabriqué par CAE Inc qui est une compagnie travaillant dans le domaine de la modélisation et la simulation en aviation. Ce simulateur certifié au plus haut niveau (dynamique de vol) permet de représenter d'une façon exacte la dynamique de vol de l'avion Cessna. Les conditions de vol du simulateur sont identiques aux conditions réelles de vol.



Figure 3.1 Le simulateur de vol de Cessna Citation X de LARCASE

L'acquisition des données à partir du simulateur se fait à partir d'un fichier **main.cts** dans lequel il faut définir :

- La durée : au bout de laquelle la prise de données est réalisée;
- La fréquence d'échantillonnage : le pas à définir pour enregistrer les données;
- Les paramètres à enregistrer : les variables calculées par le simulateur ainsi que leurs noms, descriptions et unités.

Le prélèvement des paramètres désirés se fait après avoir étudié le fonctionnement du moteur et trouver les variables qui l'influencent.

La croisière est la phase la plus représentative lors du vol d'un avion puisqu'elle est la plus longue partie. Le modèle du moteur ici présenté a été conçu pour identifier ses paramètres pendant cette phase de vol.

3.1.2 La phase de croisière

La phase de croisière est la plus longue partie du vol. Elle est située entre le décollage et l'atterrissage de l'avion. Lors de cette phase, l'avion consomme le moins de carburant suivant une configuration optimale. En effet, pour obtenir une meilleure fiabilité, il faudrait utiliser le moteur entre 55% et 75% de sa puissance maximale (Paris, 2013).

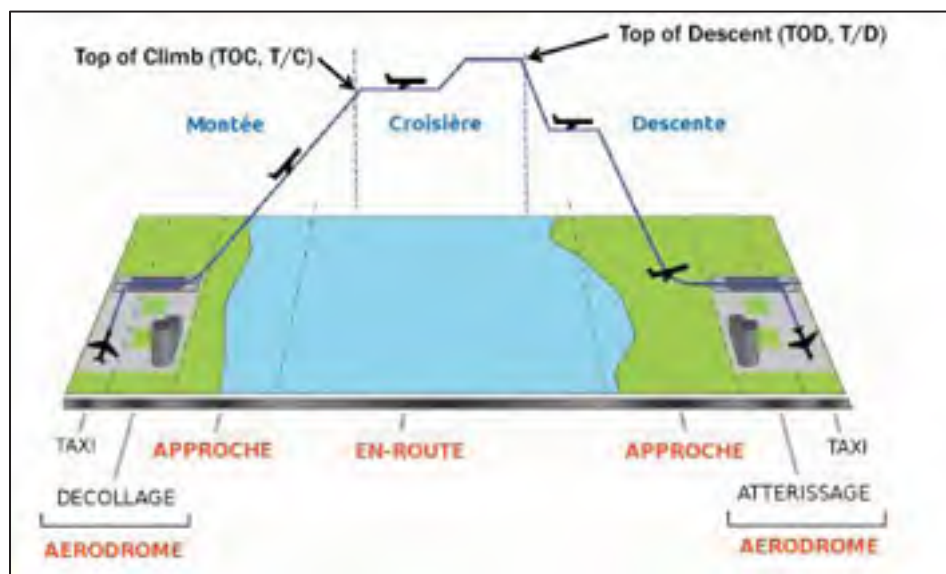


Figure 3.2 Les phases de vol à partir du décollage à l'atterrissage
Tirée de Girardet, (2014)

Le régime de croisière (Morieux, 2014) :

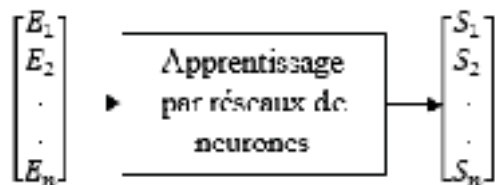
- Une croisière en "régime économique" est réalisée pour une consommation réduite de carburant et une vitesse par rapport au sol raisonnable. Ce régime est obtenu à 65% de la puissance maximale du moteur
- Une croisière en "régime normal" correspond à une utilisation normale avec une meilleure performance de vitesse de l'avion et une consommation raisonnable. Ce

régime est obtenu pour la puissance du moteur entre 70% et 80% de la puissance maximale du moteur.

- Une croisière "rapide" correspond à une utilisation prolongée du moteur et à une puissance élevée de plus que 80%, ce qui provoque une consommation plus importante de carburant.

Pendant la croisière, les puissances des deux moteurs de l'aéronef doivent être symétriques ; la même pression d'admission et le même nombre de tours des moteurs.

Dans le but d'entraîner les réseaux de neurones, une mise en forme des données est établie. Ces données sont représentées sous forme d'une matrice, elles peuvent être des entrées (E_1, E_2, \dots, E_n) ou/et des sorties (S_1, S_2, \dots, S_n) du système à modéliser.



Les entrées ainsi que les sorties provenant du simulateur vont être utilisées pour l'apprentissage. Ensuite, après l'apprentissage, les sorties seront obtenues en fonction des entrées, mais tout en assurant la précision de prédiction. De ce fait, des sorties non utilisées pour l'apprentissage seront comparées à celles obtenues à partir du simulateur.

3.2 La conception du modèle de prédiction des paramètres du moteur

3.2.1 Le principe de construction du modèle

Le comportement des moteurs de l'avion Cessna Citation est non linéaire. Un modèle mathématique d'identification à partir des tests en vol a été développé pour estimer la poussée du moteur pour n'importe quelle condition de vol (Ghazi et al., 2015).

Les paramètres à prédire par notre modèle (les sorties du système) sont :

- La poussée nette du moteur (FNC) en livres (lbs);
- La consommation du carburant (WFC) en livres/heure (lbs/h);
- La vitesse de rotation du moteur à haute pression (N2PC) en tours par minute (RPM).

La poussée nette du moteur de Cessna Citation X (turboréacteur), ainsi que la consommation du carburant varient en fonction de la vitesse et de l'altitude de l'avion. La variation de la poussée nette du moteur est relative de la variation de la vitesse de l'avion. (Anderson, 1999)

Soit : $F_{v=0}$ est la poussée nette du moteur au niveau de la mer et à une vitesse nulle

F est la poussée nette du moteur

La variation de la poussée en fonction du nombre de Mach peut être représentée par la relation ci-dessous : (Anderson, 1999)

$$\frac{F}{F_{v=0}} = A M_{\infty}^{-n} \quad (3.1)$$

Les deux valeurs A et n sont deux paramètres qui varient en fonction de l'altitude, M_{∞} est le nombre de Mach. La variation de la poussée en fonction de l'altitude pourrait être représentée par : (Anderson, 1999)

$$\frac{F}{F_{v=0}} = \left(\frac{\rho}{\rho_0}\right)^m \quad (3.2)$$

Où ρ est la densité de l'air à une altitude donnée variant en fonction de la température, ρ_0 est la densité au niveau de la mer, m est un coefficient qui dépend de la conception du moteur et qui est proche de 1. La densité ρ peut s'écrire sous la forme suivante (à partir de la loi des gaz parfaits) :

$$\rho = \frac{P M_{air}}{R T} \quad (3.3)$$

Où T est la température à l'altitude donnée, R est la constante universelle des gaz parfaits, P est la pression de l'air et M_{air} est la masse molaire de l'air. En effet, à partir des équations (3.1), (3.2) et (3.3), la poussée nette du moteur varie essentiellement avec l'altitude, le nombre de Mach et de la température.

Concernant la consommation du carburant : soit C_t la consommation du carburant spécifiée à une altitude et à un nombre de Mach M_∞ , elle peut s'exprimer comme suit (Anderson, 1999) :

$$C_t = B (1+k M_\infty) \quad (3.4)$$

Où B et k sont deux coefficients empiriques

La consommation du carburant d'un aéronef dépend essentiellement du nombre de Mach. Comme considéré précédemment, les réseaux de neurones sont la base de notre travail. En effet, les données qui seront utilisées pour l'apprentissage sont déduites à partir des équations de (3.1) à (3.4). Les entrées et les sorties du système sont obtenues à partir du simulateur de vol de l'avion Cessna Citation X. Un modèle de prédiction des sorties telles que la poussée nette du moteur et la consommation du carburant est présenté ci-dessous :

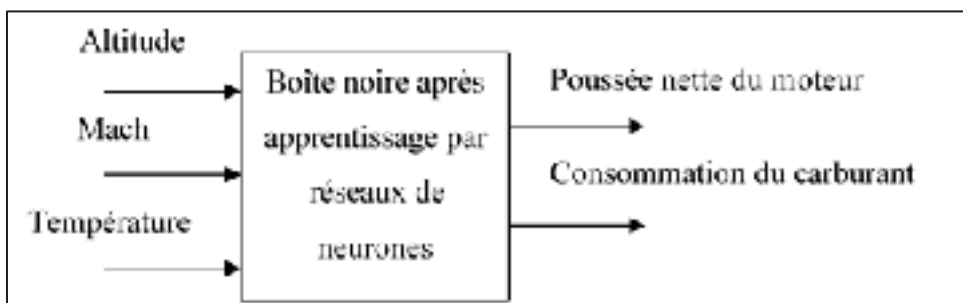


Figure 3.3 Le modèle de prédiction des paramètres du moteur proposé

Les données qui ont été choisies pour effectuer l'apprentissage des boîtes noires de prédiction de paramètres du moteur sont présentées dans le tableau 3.1 :

Tableau 3.1 Les variables utilisées pour la conception du modèle

Variables logicielles	Définition des variables	Valeurs	Unités
G04_EOM_ALT_PRESS_F8	Altitude	[5000...45000]	pieds
G04_EOM_MACH_PRESS_F8	Nombre de Mach	[0.2...0.9]	-----
G71_FN_F4	Poussé nette	[930...4921]	lbs
G71_WF_F4	Consommation du carburant	[483...2337]	lbs/ heure

La température sera calculée à partir de la densité, de la pression et à l'aide de la fonction **atmosisa** de MATLAB. Sachant que la pression varie selon l'altitude, donc la température varie aussi selon l'altitude. Les valeurs de la poussée nette du moteur et de la consommation du carburant sont données à partir du simulateur de vol en fonction de la variation du Mach et de l'altitude. Un organigramme qui représente les outils utilisés pour chaque étape du travail réalisé est présenté dans la figure 3.4 ci-dessous :

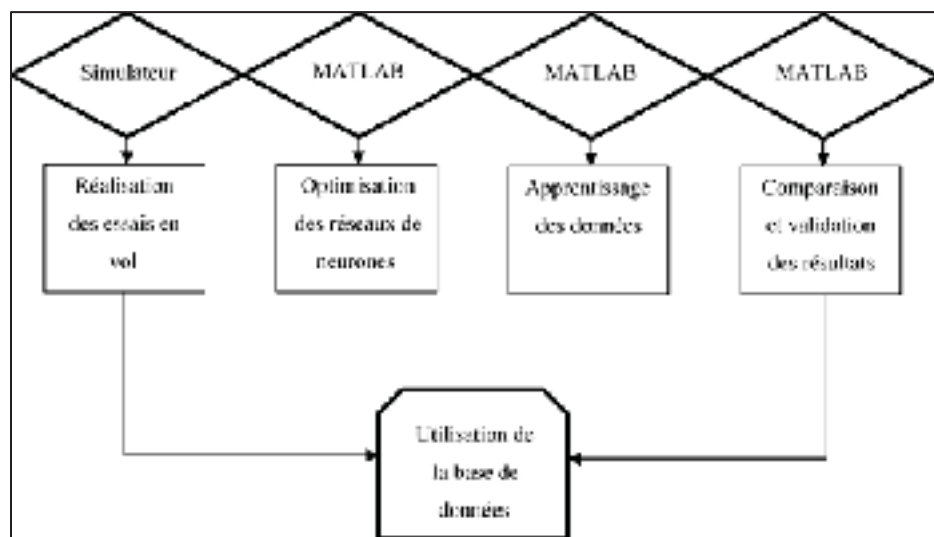


Figure 3.4 Les outils utilisés pour la conception du modèle

Un apprentissage de données a été réalisé avec le modèle de prédiction présenté précédemment toutefois les résultats n'ont pas été bons. En effet, une amélioration des résultats est nécessaire afin de pouvoir atteindre les objectifs fixés. En se basant sur le même principe des réseaux de neurones et grâce aux études réalisées des moteurs, une méthode basée sur le principe de FADEC et les performances du moteur a été conçue et utilisée.

Le FADEC est un contrôleur qui se trouve pratiquement dans tous les moteurs des récents aéronefs. Ce contrôleur est responsable essentiellement de l'alimentation du moteur en carburant. Plus précisément, le FADEC contrôle la rotation du compresseur à basse pression en fonction de la position de la manette de gaz, du nombre de Mach ainsi que de l'altitude et la température. Grâce à la rotation du moteur de l'avion à basse pression N1P, nous pouvons déterminer la rotation du moteur de l'avion Cessna à haute pression N2P ainsi que la poussée nette du moteur FN et sans oublier la consommation du carburant WF. Chaque position de la manette de gaz correspond à un mode moteur. Le FADEC gère ces différents modes (Blake, 2009).

Tableau 3.2 Les différents modes de moteur

Modes	Positions de la manette de gaz TLA (degrés)
CUT OFF	5 – 8
IDLE (Le régime à ralenti)	8 – 13
MCR (Max Cruise) mode de croisière	40
MCL (Max Climb): mode de montée	50
NTO (Normal Take-Off): mode de décollage	60
MAX : la position extrême de la manette de gaz	65

Une nouvelle conception du modèle du moteur a été réalisée en tenant compte de la position de la manette de gaz et du principe du FADEC. Cette conception s'est basée sur le premier modèle qui donne la poussée nette du moteur, la consommation du carburant en fonction de l'altitude, du nombre de Mach et de la température, il faut ajouter que le modèle est maintenant composé de deux boîtes noires. La première boîte représente le rôle du FADEC en donnant en sortie la rotation du moteur à basse pression (N1P) en fonction de l'altitude, la température, la position de la manette de gaz (TLA) et le nombre de Mach. La deuxième boîte va prédire la poussée nette du moteur, la consommation du moteur ainsi que la rotation du moteur à haute pression (N2P) en fonction de la sortie de la première boîte qui n'est autre que la rotation du premier moteur et aussi le nombre de Mach. Ce modèle a été réalisé sous forme de deux boîtes noires représentant un système physique car nous ne pouvons pas avoir les données géométriques du moteur de Cessna qui sont confidentielles. Le nouveau modèle conçu est représenté ci-dessous :

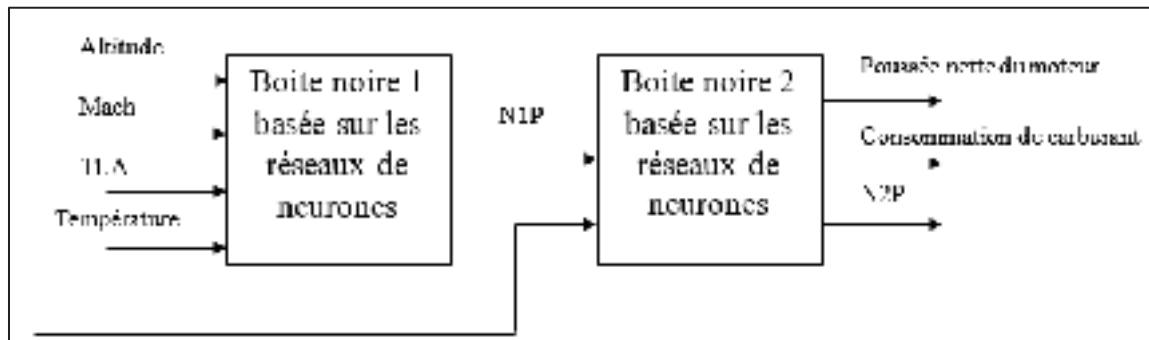


Figure 3.5 Le modèle de prédiction des paramètres du moteur

Nous devons avoir les données (TLA, N1P, N2P) à partir du simulateur dans le but de réaliser l'apprentissage des réseaux de neurones des deux boîtes noires permettant l'identification des paramètres désirés du moteur de Cessna. Le tableau 3.2 présente les nouveaux paramètres à utiliser pour l'apprentissage des deux boîtes noires :

Tableau 3.3 Les paramètres à utiliser dans la conception du modèle

Noms des variables dans le simulateur	Définition des variables	Valeurs	Unités
G71_TLA_F4	Position de la manette de gaz	[40...65]	Degrés
G71_N1P_F4	Vitesse de rotation du moteur 1 à basse pression	[77...94]	% (Tours/min)
G71_N2P_F4	Vitesse de rotation du moteur 2 à haute pression	[87...96]	% (Tours/min)

Comme cité précédemment, les entrées et les sorties du modèle seront utilisées pour l'apprentissage des deux boîtes noires. Après cet apprentissage, ces deux boîtes seront capables de prédire la poussée nette du moteur, la consommation du carburant ainsi que la rotation du moteur à haute pression en introduisant l'altitude, le nombre de Mach, la position de la manette de gaz et la température. Ainsi, la dépendance des sorties par rapport aux entrées est validée à partir des essais réalisés à l'aide du simulateur de vol du Cessna. Les valeurs de N1P et N2P sont données (en pourcentage de tours par minute) à partir de ce simulateur de vol mais aussi en fonction de l'altitude, le nombre de Mach et la position de la manette de gaz.

Tout d'abord, nous procédons à une correction de quelques paramètres ce qui permettra : (Blake, 2009)

- D'éliminer la dépendance de certains paramètres comme le nombre de Mach;
- De tenir compte des conditions ambiantes du moteur;
- D'avoir des variables avec un comportement simple à modéliser.

3.2.2 La correction des paramètres d'apprentissage

La façon de corriger certaines variables est d'utiliser des rapports adimensionnés de température et de pression (Blake, 2009), soit :

T : La température absolue

T_0 : La température au niveau de la mer (= 288.15 K)

θ : Le ratio de températures

$$\theta = \frac{T}{T_0} \quad (3.5)$$

P : La pression ambiante

P_0 : La pression au niveau de la mer

δ : le ratio de pressions

$$\delta = \frac{P}{P_0} \quad (3.6)$$

Ces ratios (températures et pressions) sont utilisés pour les calculs des facteurs de correction des paramètres tels que la rotation du moteur à basse pression (N1P) et celle du moteur à haute pression (N2P), la poussée nette du moteur (FN) ainsi que la consommation du carburant (WF). Ces facteurs sont présentés comme suit :

$$\theta^* = \frac{T}{T_0} (0.2 M^2 + 1) \quad (3.7)$$

$$\delta^* = \frac{P}{P_0} (0.2 M^2 + 1)^{3.5} \quad (3.8)$$

Où M est le nombre de Mach

Grâce à ces deux facteurs de corrections relatives à la température et à la pression (équations 3.7 et 3.8), les paramètres à corriger deviennent :

- La vitesse de rotation du moteur 1 à basse pression : N1P

$$N1PC = \frac{N1P}{\sqrt{\theta^*}} \quad (3.9)$$

- La vitesse de rotation du moteur 2 à haute pression : N2P

$$N2PC = \frac{N2P}{\sqrt{\theta^*}} \quad (3.10)$$

- La poussée nette du moteur : FN

$$FNC = \frac{FN}{\delta^*} \quad (3.11)$$

- Le débit de la consommation de carburant : WF

$$WFC = \frac{WF}{\delta^* \sqrt{\theta^*}} \quad (3.12)$$

Ensuite, après avoir préparé la base de données, nous devons passer à l'étape d'apprentissage qui permettrait d'obtenir le modèle désiré pour l'identification des paramètres. En effet, une méthodologie d'apprentissage doit être suivie afin d'avoir le modèle le plus précis. Pour ce faire, il faudrait utiliser un réseau de neurones optimisé qui garantirait la qualité d'apprentissage.

3.2.3 L'apprentissage des données avec les réseaux de neurones

L'apprentissage est basé sur l'algorithme de Levenberg-Marquardt qui a été cité et expliqué précédemment dans le deuxième chapitre. Mais, dans le but d'effectuer un meilleur apprentissage, il faut utiliser des réseaux de neurones optimisés. En effet, l'utilisation d'un algorithme de recherche locale qui est le grand déluge étendu (EGD) cité dans le premier chapitre, qui peut être combiné à l'algorithme de réseaux de neurones Levenberg-Marquardt est une solution pour obtenir de meilleurs résultats (Ben Mosbah, 2011).

Cet algorithme présente des avantages par rapport à sa rapidité de convergence et sa facilité de mettre en œuvre dans le cas de notre modèle particulièrement. Le principe de l'EGD est le fait d'utiliser une méthode permettant d'accepter certaines mauvaises solutions ne dépassant pas une certaine limite, ce qui permet de rapprocher le plus possible la solution cherchée de la solution réelle. Le nombre de couche maximal, le nombre des itérations ainsi que l'erreur acceptée sont des paramètres essentiels pour le processus d'apprentissage. Alors, nous fixons ces paramètres pour satisfaire aux critères d'apprentissage, en sachant que le nombre de neurones est modifié d'une façon aléatoire et automatique.

- Nombre maximal de couches = 4;
- Le pas $\Delta B = 0.00001$;
- Nombre maximal d'itérations = 10.

Ces paramètres ont été fixés suite à des tests réalisés tout au long du projet.

Le principe de l'EGD est appliqué à l'algorithme Levenberg-Marquardt mentionné dans la figure 2.4 du deuxième chapitre en tenant compte des paramètres cités précédemment (le nombre de couches, le pas ΔB , le nombre d'itérations). Le principe de l'EGD est expliqué dans le premier chapitre. Nous allons effectuer l'apprentissage des données grâce à l'algorithme présenté dans la figure 3.6 qui combine Levenberg-Marquardt et l'EGD. Le principe de cet algorithme est de commencer tout d'abord par l'introduction des données (entrées/sorties) à apprendre par le réseau. Ensuite, il faudrait appliquer les équations (de 3.5 à 3.12) permettant de corriger les paramètres cités dans la section 3.2.2. Il faudrait initialiser le nombre initial de couches et de neurones. L'algorithme effectue par la suite l'apprentissage afin de calculer

l'erreur d'apprentissage. L'initialisation du pas ΔB , le nombre maximal d'itérations et le nombre maximal de couches permettrait d'arrêter le programme. Le nombre de neurones est affecté aléatoirement à chaque couche afin de minimiser au maximum le réseau. Finalement, si l'un des critères cités est atteint, le programme enregistre le réseau optimisé que nous utiliserons pour la prédiction des paramètres. En effet, deux réseaux de neurones seront utilisés ; le premier réseau remplacerait le rôle du FADEC pour la prédiction de N1PC pour chacune des positions de la manette de gaz (TLA) choisies et le deuxième réseau fournirait la poussée nette du moteur FNC, la consommation du carburant et la rotation du moteur à haute pression N2PC pour les mêmes TLA.

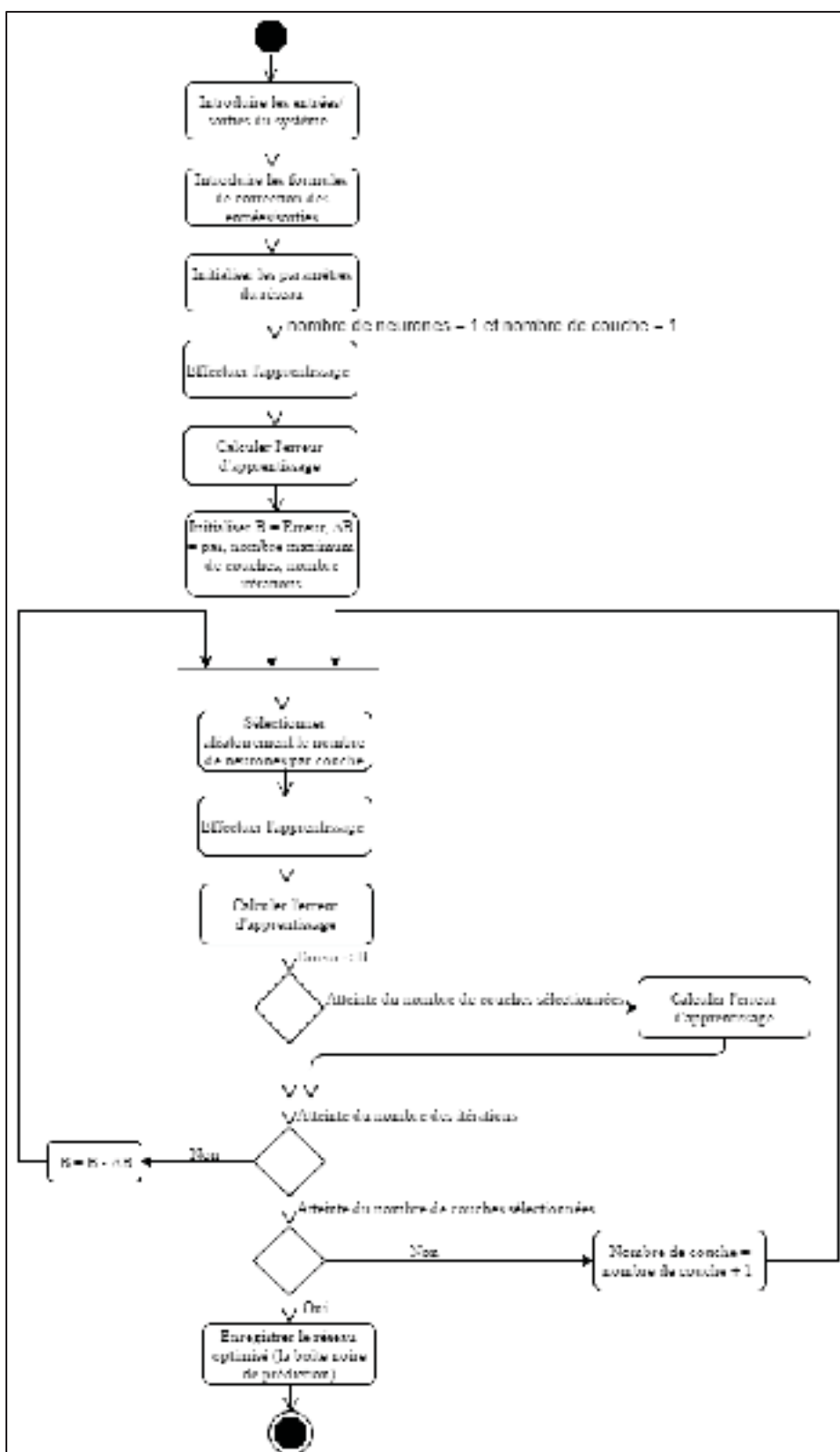


Figure 3.6 Algorithme combiné EGD et réseaux de neurones

Après avoir récupéré les données du simulateur, nous allons fixer l'altitude ainsi que la position de la manette de gaz et faire varier le nombre de Mach. La température va être calculée automatiquement en fonction de la variation de l'altitude et va influencer les corrections des paramètres citées précédemment.

Nous avons prélevé 25% des données d'une façon aléatoire dans l'ensemble de l'enveloppe de vol dans lequel nous avons travaillé afin de s'en servir pour les tests de validation. La façon aléatoire permet de prendre des bas, des moyens et des hauts nombres de Mach pour chaque altitude et pour chaque position de manette de gaz ce qui permet de couvrir l'enveloppe de vol. Les autres 75% des données sont utilisées pour l'apprentissage des deux réseaux de neurones. Pour commencer, nous avons besoin des valeurs des altitudes, des nombres de Mach, de la position de la manette de gaz TLA et de la vitesse de rotation du moteur N1P à basse pression. Nous corrigeons les valeurs de N1P obtenues à partir du simulateur en utilisant les équations de 3.7 à 3.9.

La première étape est réalisée pour une position de manette de gaz fixée (TLA = 35 degrés), pour des altitudes fixes variant de 5,000 pieds à 10,000 pieds. Le nombre de Mach varie de 0.22 à 0.84 relativement à chaque altitude et les valeurs de N1P varient aussi suivant ces nombres de Mach. Cette étape servirait juste à la vérification du programme d'apprentissage et de ses paramètres d'initialisation car le but du modèle est d'avoir des prédictions pour altitudes et nombre de Mach variables. Les valeurs du paramètre (N1PC) prédites seront comparées à celles obtenues à partir des tests de vols du simulateur et seront présentées dans le quatrième chapitre. Ensuite, l'apprentissage des données est réalisé pour chaque position de manette de gaz variant de 35 degrés à 65 degrés avec un pas de 5 degrés chacune pour des altitudes variables de 5,000 pieds à 45,000 pieds avec un pas de 5,000 pieds. La température varie suivant les altitudes et serait calculée directement par MATLAB. Le choix des pas entre les différentes entrées a été pris suite aux plusieurs tests réalisés avec différentes altitudes et nombres de Mach qui varient de 0.2 à 0.9 avec un pas de 0.0001. Donc, le nombre total des données est à peu près 441000 cas de vol (7000 nombres de Mach, 9 altitudes et 7 TLAs). En effet, chacune des positions TLA possède une enveloppe de vol sur laquelle nous avons réalisé

notre modèle. Cette enveloppe permet une meilleure modélisation du comportement du moteur et sert comme une base de données intéressante permettant d'effectuer l'apprentissage désiré des réseaux de neurones. L'enveloppe de vol pour TLA = 35 degrés est présenté dans la figure 3.7.

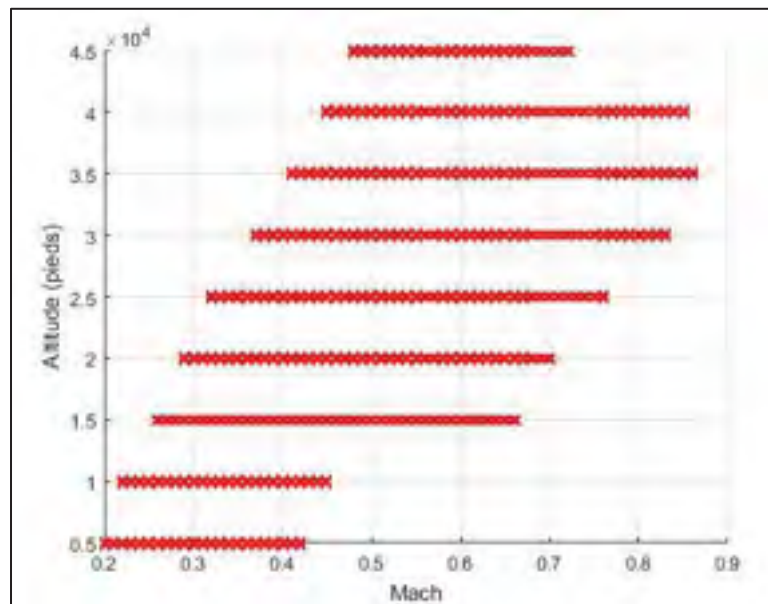


Figure 3.7 Enveloppe du vol TLA = 35 degrés

L'algorithme appliqué est celui dans la figure 3.6. La variation de l'altitude, nombre de Mach, et température permettrait d'avoir de très bons résultats puisque la base de données serait plus grande. En d'autres termes, les réseaux de neurones apprendront mieux lorsque la base de données sera plus importante. Pour chaque position de manette de gaz TLA, nous obtenons les valeurs prédites de NIPC pour chaque altitude et pour chaque nombre de Mach à partir du premier bloc de réseaux neurones (boîte noire). La figure 3.8 illustre la boîte noire de prédiction de NIPC qui reçoit en entrée, l'altitude, le nombre de Mach et TLA qui est fixé à 35 degrés. La température est calculée automatiquement comme nous l'avons expliqué précédemment. NIPC théorique contient les données provenant du simulateur de vol et servira à la comparaison par rapport aux données prédites à partir du bloc 1.

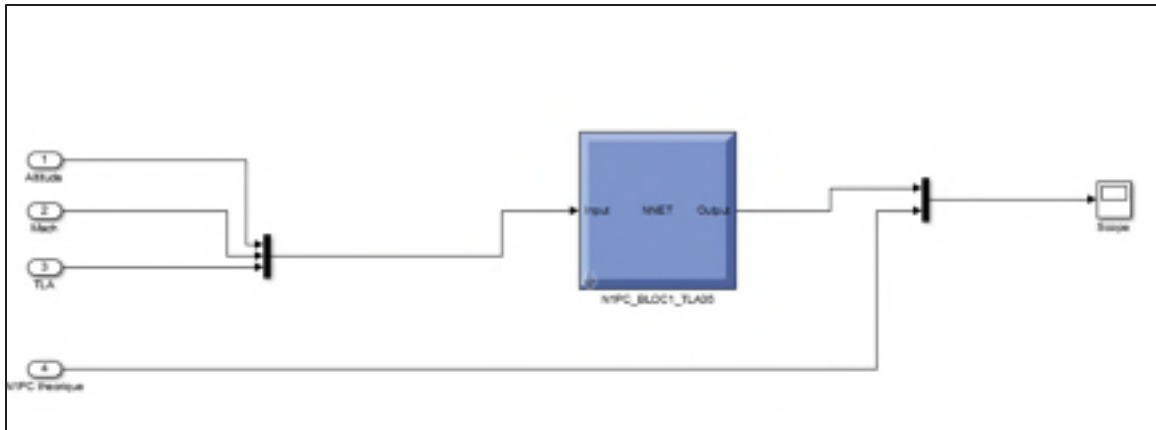


Figure 3.8 Bloc de prédiction de N1PC pour altitudes variables, Mach variable et TLA 35 degrés

La prédiction de N1PC permet de servir à l'apprentissage du deuxième réseau en plus du nombre de Mach et les autres paramètres tels que : la poussée nette du moteur FN, la consommation du carburant WF et la vitesse de rotation du moteur à haute pression N2P provenant du simulateur. Après avoir obtenu toutes les valeurs de N1PC pour chaque position de la manette de gaz de 35 degrés à 65 degrés, nous passons à l'apprentissage du deuxième système de prédiction (deuxième boîte noire). Pour ce faire, nous préparons la base de données nécessaire composée du nombre de Mach, les valeurs de N1PC, les valeurs de la poussée nette du moteur FNC, de la consommation du carburant WFC et la vitesse de rotation du moteur à haute pression N2PC après leurs corrections par les mêmes équations (de 3.5 à 3.12) utilisées précédemment. Nous introduisons ces paramètres au programme d'apprentissage présenté à la figure 3.6.

Les valeurs obtenues des sorties prédites sont très bonnes en les comparant aux valeurs obtenues par le simulateur de vol. La validation de ce deuxième bloc de prédiction a été effectuée de la même manière que le premier bloc.

Nous arrivons à l'étape de combinaison des deux blocs de prédiction pour avoir le modèle conçu au début du travail. Nous testons tout le modèle et nous constatons que l'erreur moyenne de prédiction a légèrement augmenté tout en restant acceptable. Cette augmentation est expliquée par la mise en cascade des deux boîtes noires de prédiction possédant chacune des erreurs différentes.

Le modèle conçu pourrait être ajouté au modèle aérodynamique de l'avion. En effet, pour chaque position de manette de gaz lors de la croisière, nous pouvons déterminer la poussée nette du moteur et la consommation du carburant qui sont deux paramètres importants et qui interviennent dans le fonctionnement de l'avion. Le modèle de prédiction des paramètres du moteur est représenté dans la figure 3.9. Les entrées sont l'altitude, le nombre de Mach et la position de la manette de gaz TLA qui permettraient d'obtenir N1PC en premier lieu, et par la suite obtenir les paramètres désirés qui sont : la poussée nette du moteur FNC, de la consommation du carburant WFC et la vitesse de rotation du moteur à haute pression N2PC. FNC, WFC, N2PC théoriques sont les données provenant du simulateur de vol après leurs corrections et seraient utilisées à la comparaison par rapport aux données prédites par le modèle réalisé. L'architecture des deux blocs de réseaux de neurones permettant la prédiction des paramètres du moteur est présentée dans la figure 3.9 :

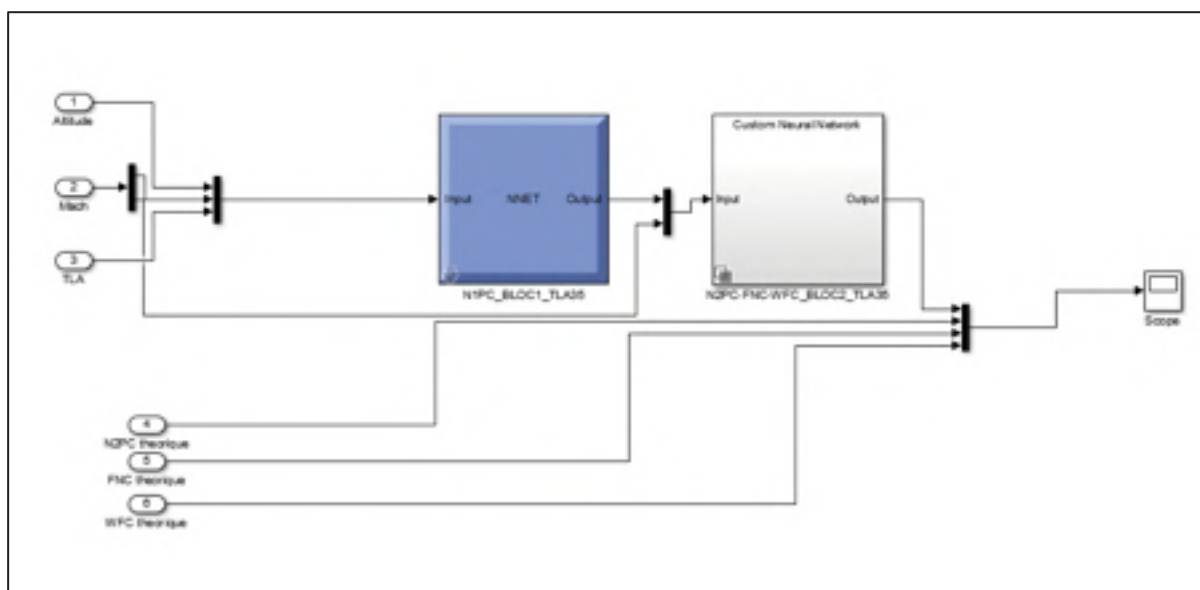


Figure 3.9 Architecture du modèle de prédiction des paramètres du moteur du Cessna Citation X en croisière

La conception d'un modèle basé sur les réseaux de neurones (Levenberg-Marquardt) et un algorithme d'optimisation (EGD) ont permis l'obtention d'un système de prédiction de la poussée nette, de la consommation du carburant du moteur du Cessna Citation X ainsi que de la vitesse de rotation du moteur à haute pression en fonction de l'altitude, du nombre de Mach,

de la position de la manette de gaz et de la température. Un tel modèle permettrait d'obtenir le gain de temps perdu dans l'acquisition des données à partir du simulateur de vol, ainsi que l'essai des nouvelles lois de commandes. La méthode générale de conception du modèle s'est basée sur l'extraction des relations entre les paramètres influant le fonctionnement du moteur. Nous avons utilisé les réseaux de neurones car ils présentaient une simplicité d'implémentation et qu'ils pourraient gérer la non linéarité du système. Les résultats obtenus pour chaque étape de conception et chaque paramètre seront présentés et discutés dans le chapitre suivant.

CHAPITRE 4

RÉSULTATS ET DISCUSSIONS

Ce chapitre présente les résultats obtenus au cours et à la fin de la conception du modèle du moteur de l'avion Cessna Citation X expliquée à travers le chapitre précédent. Le critère de validation du modèle conçu va s'effectuer par la comparaison des résultats obtenus de ce modèle avec les données provenant du simulateur de vol et en tenant compte de la règle de validation croisée des réseaux de neurones qui permet de valider le modèle à partir d'un pourcentage (25%) des données non utilisées pour l'apprentissage. Un autre critère de validation qui a été tenu en compte est la valeur d'erreur de $\pm 5\%$ qu'il ne faut pas dépasser pour la modélisation des moteurs des avions (Aviation, 2003). Comme nous l'avons expliqué à travers le chapitre précédent, le modèle du moteur à réaliser, consiste à identifier sa poussée nette, sa consommation de carburant ainsi que sa rotation à haute pression. Ces paramètres dépendent de l'altitude, du nombre de Mach, de la position de la manette de gaz et de la température.

Tout d'abord, nous allons présenter le modèle à l'altitude fixe et Mach variable qui nous a permis uniquement de vérifier la dépendance de certains paramètres et tester l'algorithme d'apprentissage (vérifier ses paramètres à gérer). Ensuite, nous allons varier le nombre de Mach et l'altitude au même temps et voir les résultats obtenus pour chacune des positions choisies de la manette de gaz pour la phase de croisière.

4.1 Modèle du moteur pour l'altitude et la manette de gaz fixes et pour les nombres de Mach variables

Le modèle conçu et présenté dans le chapitre précédent, inclut deux blocs. Le premier bloc sert à obtenir la vitesse de rotation du moteur à basse pression (N1PC) et le second bloc permet d'obtenir la poussée nette du moteur (FNC), sa consommation de carburant (WFC) ainsi que la vitesse de rotation du moteur à haute pression (N2PC).

En effet, nous avons commencé par l'apprentissage des données avec un nombre de Mach variable et gardé la position de la manette de gaz et l'altitude fixes (donc la température fixe

car l'altitude est fixe). Ainsi, nous avons testé l'apprentissage pour la position de la manette de gaz fixe (TLA = 35 degrés) afin de vérifier si N1PC varie réellement selon le nombre de mach et l'altitude.

Les tests utilisés pour la conception du modèle ont été réalisés pour chaque altitude de 5,000 pieds à 45,000 pieds avec un pas choisi de 5,000 pieds, pour une position TLA qui varie de 35 degrés à 65 degrés avec un pas de 5 degrés. Concernant les nombres de Mach, ils varient évidemment selon les altitudes. Ces valeurs ont été choisies pour la phase de croisière qui représente la plus longue phase de vol. Tout d'abord, ces tests ont été réalisés sur le simulateur de vol de niveau D. Ensuite, pour la première étape de travail, une base de données a été effectuée pour l'apprentissage des réseaux de neurones. Cette base de données se compose des tableaux contenant chacun une altitude fixe, un nombre de Mach variable, une TLA fixe et la température sera aussi fixe ici car l'altitude est fixe.

Suite au prélèvement des données à partir du simulateur, nous corrigeons les paramètres à utiliser dans la base des données avec les formules de corrections qui ont été présentées dans le chapitre précédent dans le but d'améliorer les résultats. Nous effectuons par la suite les tests du premier bloc grâce à notre algorithme présenté dans la figure 3.6 dans le but d'obtenir N1PC pour chaque altitude de 5,000 pieds à 45,000 pieds pour TLA fixe à 35 degrés. Nous calculons les erreurs relatives obtenues entre les valeurs de N1PC du simulateur et celles du premier bloc après l'apprentissage. Suite à un premier apprentissage, nous avons obtenu un bon résultat représentatif du comportement réalisé par l'avion pour une altitude, température et une position de manette de gaz TLA fixe. Nous représentons l'enveloppe de vol dans lequel nous allons travailler pour la position de la manette de gaz TLA = 35 degrés dans la figure 4.1 :

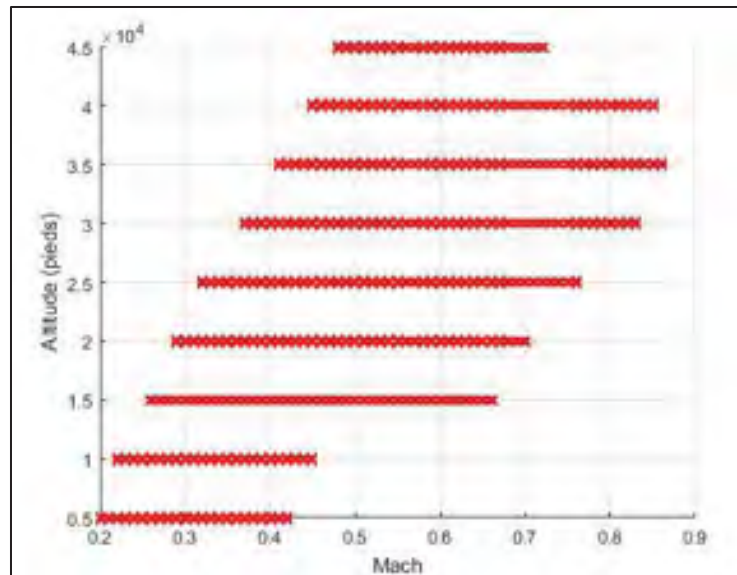


Figure 4.1 Enveloppe de vol TLA = 35 degrés

Ensuite, nous présentons la comparaison entre le NIPC obtenu numériquement par notre modèle et le NIPC obtenu suite aux tests sur le simulateur pour différents nombres de Mach variant de 0.22 à 0.84 suivant les différentes altitudes pour TLA = 35 degrés dans la figure 4.2 :

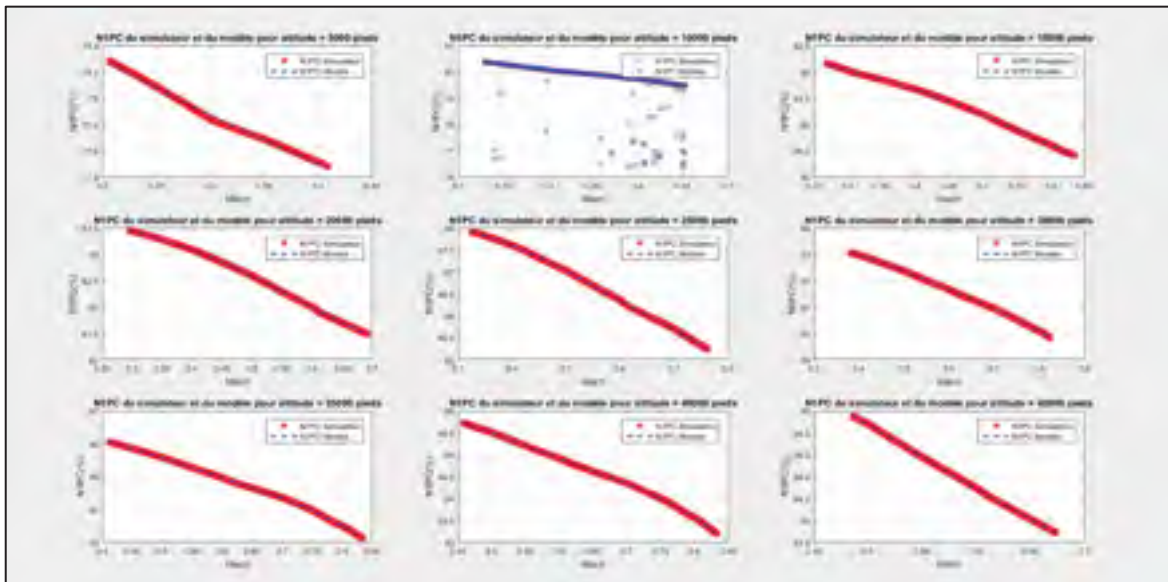


Figure 4.2 NIPC du simulateur et du modèle pour plusieurs altitudes gardées fixes, pour le nombre de Mach = 0.22 à 0.84 et TLA = 35 degrés

Nous remarquons à travers ces résultats obtenus que le N1PC varie selon la vitesse (nombre de Mach) et que l'apprentissage a réussi car les valeurs du simulateur et ceux du modèle sont très proches, avec une très petite erreur en bas de 0.04 %. Le tableau 4.1 représente la moyenne des erreurs relatives de N1PC pour chaque altitude :

Tableau 4.1 Erreur relative moyenne pour le N1PC pour plusieurs altitudes fixes et pour la plage des nombres de Mach = 0.22 à 0.84

Altitude (pieds)	Erreur relative moyenne pour le N1PC (%)
5000	0.0143
10000	0.0407
15000	0.0085
20000	0.0071
25000	0.0071
30000	0.0118
35000	0.0098
40000	0.0049
45000	0.0035

L'erreur relative de N1PC ne dépasse pas 0.1 % pour les différentes altitudes ce qui mène à déduire que la précision de prédiction est assurée par ce modèle. Nous remarquons que cette moyenne de l'erreur relative est trop petite car la variation du nombre de Mach avec un pas de 10^{-4} lors des tests de vol a permis d'avoir un grand nombre de données. L'objectif du modèle final est de calculer les paramètres du moteur (la poussée nette, la consommation du carburant et la rotation du moteur à haute pression) en fonction de la variation de l'altitude et du nombre de Mach simultanément. En effet, il faudrait avoir un modèle valide pour des altitudes et des nombres de Mach variables. C'est pour cette raison que nous n'avons pas effectué les tests pour toutes les positions de manette de gaz. Effectivement, cette étape était utile pour vérifier la précision de l'algorithme utilisé pour l'apprentissage des données et qui a été cité dans la

figure 3.6 dans le chapitre précédent. Outre, elle nous a permis de nous assurer de la variation du N1PC en fonction du nombre de Mach.

L'étape suivante de la recherche est de faire l'apprentissage des données de chacun des blocs du modèle en fonction des paramètres provenant du simulateur, tout en faisant varier l'altitude et le Mach en même temps pour une position de manette de gaz qui est fixe et ceci pour remplacer le rôle du FADEC (la première boîte noire du modèle présenté dans le chapitre 3). L'apprentissage va se faire avec les réseaux de neurones en combinaison avec l'algorithme de grand déluge étendu comme nous l'avons expliqué précédemment dans le but d'utiliser un réseau optimisé de neurones et de garantir la précision des résultats.

4.2 Modèle du moteur aux altitudes variables, nombres de Mach variables, températures variables et à la position de la manette de gaz (TLA) fixe

4.2.1 Résultats du premier bloc (N1PC)

D'emblée, nous avons commencé par la réalisation de la première boîte noire du modèle qui va donner comme sortie le N1PC. Pour ce faire, nous préparons une base de données contenant les altitudes variant de 5,000 à 45,000 pi avec leurs nombres de Mach et N1PC correspondants prélevés à partir du simulateur, tout en gardant la position de la manette de gaz fixe. Nous avons réalisé l'apprentissage des données comme cité précédemment à base des réseaux de neurones qui ont été optimisés par l'algorithme du grand déluge étendu afin d'assurer les meilleurs résultats. La variation de l'altitude a influencé automatiquement la variation de la température qui a été utilisée comme une entrée parmi celles du modèle conçu. Une partie de la base des données non utilisée à l'apprentissage (25% des données) a été utilisée pour la validation de chaque bloc du modèle.

L'apprentissage a été réalisé avec l'initialisation des paramètres de base de l'algorithme. Ces paramètres sont le nombre de couches à ne pas dépasser (nombre maximal de couches = 4), le pas qui est utilisé pour l'algorithme d'optimisation (EGD), $\Delta B = 0.00001$, et le nombre maximal d'itérations comme nous avons expliqué dans le chapitre précédent. Le nombre de

couches qui a permis d'obtenir les meilleurs résultats est de 2 couches. Dans notre cas, le nombre de neurones par couches diffère d'un apprentissage à un autre car il est sélectionné aléatoirement afin de garantir la précision du réseau optimisé.

Au début, nous affichons les résultats d'identification de NIPC pour la plus basse position de manette de gaz sélectionnée TLA = 35 degrés en fonction du nombre de Mach et de l'altitude; ainsi, l'enveloppe de vol est celle donnée dans la figure 4.1 :

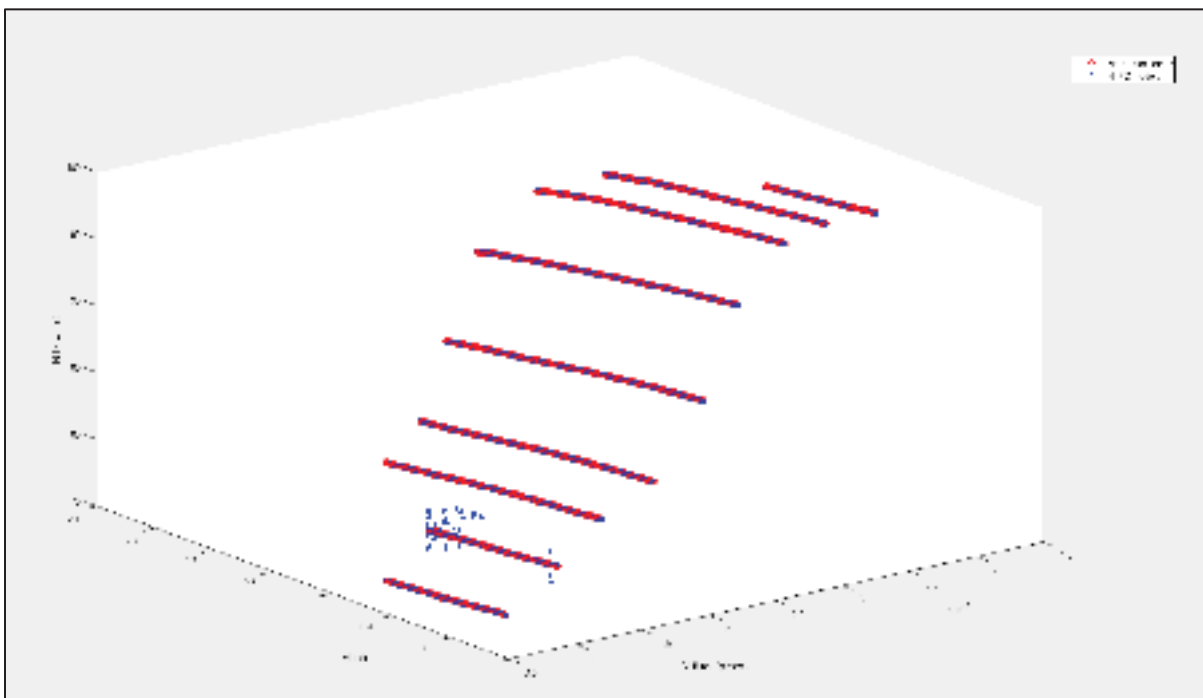


Figure 4.3 NIPC du simulateur et NIPC du modèle pour TLA = 35 degrés

Le modèle présente une haute précision de prédiction de NIPC par rapport aux données du simulateur en fonction de l'altitude et du nombre de Mach ainsi que la température. Il n'est pas nécessaire de présenter la variation de NIPC avec la température puisqu'elle varie en fonction de l'altitude. L'erreur relative moyenne de NIPC est de $3.7 \cdot 10^{-3} \%$ ce qui est vraiment un excellent résultat, aidant à la prédiction des autres paramètres du deuxième bloc du modèle. Nous remarquons qu'il y a des points présentés de NIPC du modèle dans la figure 4.3 (en bleu) qui ne sont pas très proches des valeurs de NIPC provenant du simulateur pour une altitude de 10,000 pi et pour le TLA = 35 degrés. Cette erreur est expliquée par une mauvaise réalisation de ce test de vol à cette altitude. En d'autres termes, une mauvaise manipulation lors de la

réalisation des tests en vol a provoqué la mauvaise prédiction de NIPC pour certains nombres de Mach. Le même travail a été réalisé pour toutes les positions de manette de gaz (TLA) choisies de [35,40,45,50,55,60,65] degrés. Nous présentons aussi les résultats pour la plus haute position de manette de gaz choisie TLA = 65 degrés, l'enveloppe de vol obtenue pour cette position de TLA ainsi que la variation du NIPC sont présentées respectivement dans la figure 4.4 et la figure 4.5 :

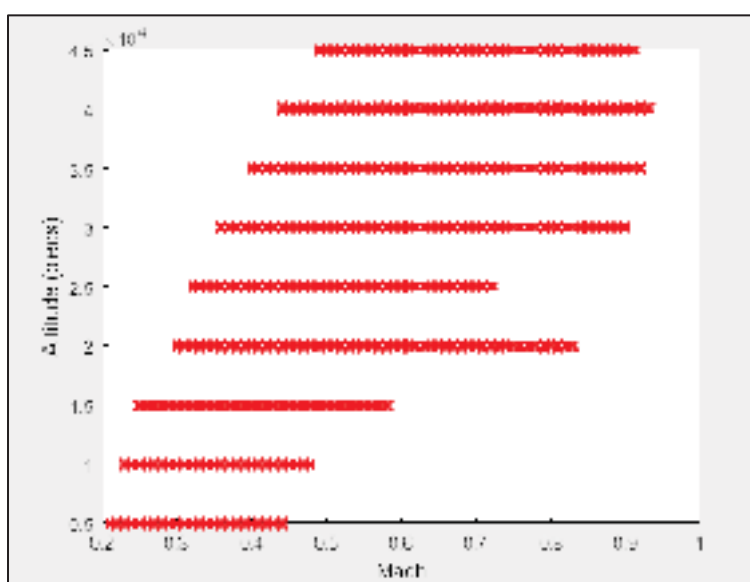


Figure 4.4 Enveloppe de vol pour le TLA = 65 degrés

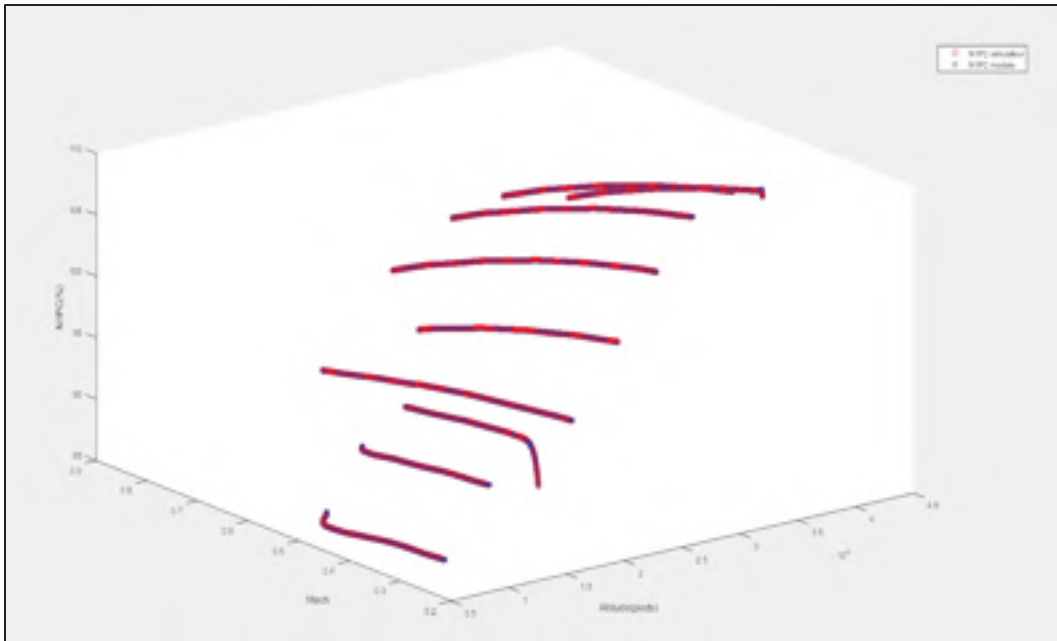


Figure 4.5 N1PC du simulateur et N1PC du modèle pour le TLA = 65 degrés

La figure 4.5 montre que les valeurs de N1PC du modèle sont bien prédites ce qui mène à déduire la réussite d'apprentissage lors de l'identification du modèle. La figure 4.6 ci-dessous montre les erreurs relatives moyennes de N1PC pour toutes les positions de manette de gaz TLA que nous avons choisi pour l'identification du modèle :

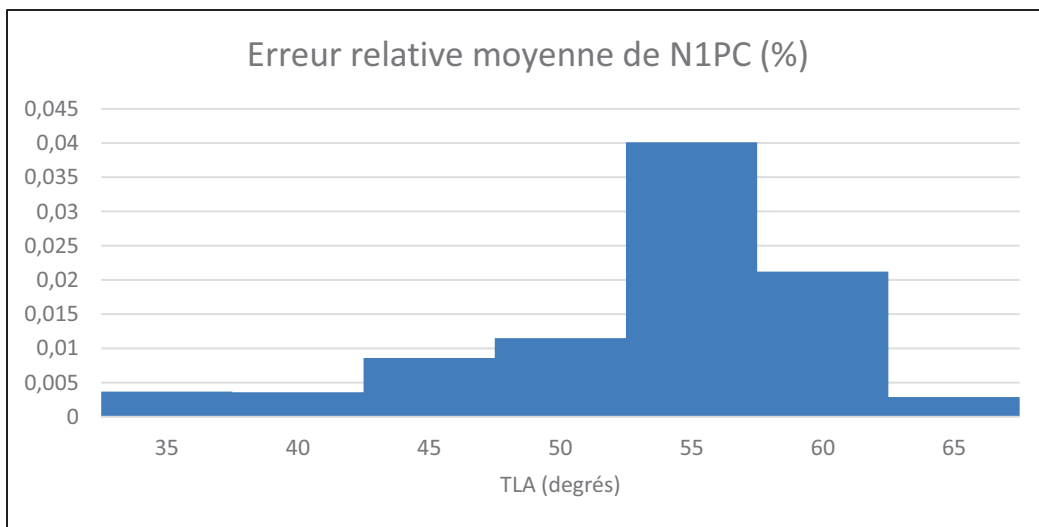


Figure 4.6 Erreur relative moyenne de N1PC dans l'identification du modèle

Pour la validation du modèle, nous l'avons testé avec les données qui n'ont pas été utilisées pour l'apprentissage et avons obtenu de bons résultats. Nous présentons dans la figure 4.7 les erreurs relatives moyennes de N1PC pour chaque position TLA dans la validation du modèle :

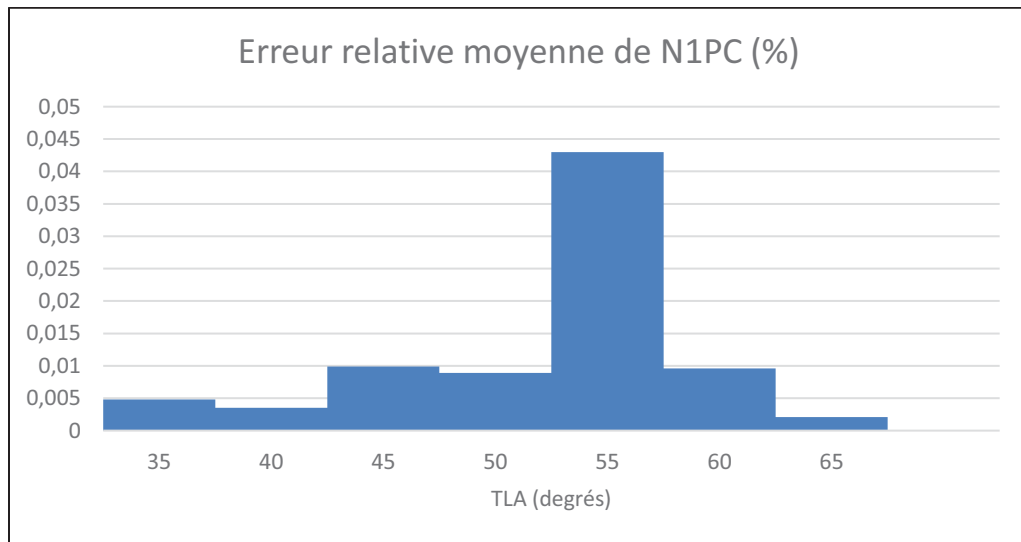


Figure 4.7 Erreur relative moyenne de N1PC dans la validation du modèle

Les prédictions de N1PC effectuées par le premier bloc du modèle conçu paraissent très intéressantes puisque la moyenne des erreurs relatives entre N1PC du simulateur et ceux obtenus par le modèle a été moins que 0.1%. En effet, un faible pourcentage d'erreur permet d'assurer une bonne prédiction des autres paramètres qui sont la consommation du carburant, la poussée nette du moteur et la rotation du moteur à haute pression qui sont prédites par le deuxième bloc. L'étape suivante serait d'effectuer l'apprentissage de ce deuxième bloc qui servira à prédire les paramètres du moteur que nous venons de citer.

4.2.2 Résultats du deuxième bloc (N2PC, FNC, WFC)

Afin de réaliser cette étape, une base de données a été préparée comme précédemment pour faire leur apprentissage grâce au même algorithme utilisé pour les étapes précédentes, et expliqué au troisième chapitre. Cette base de données contient les nombres de Mach, la vitesse de rotation du moteur à basse pression (N1P), la vitesse de rotation du moteur à haute pression

(N2P), la consommation du carburant (WF) et la poussée nette du moteur (FN) avant les corrections par les équations (de 3.5 à 3.12) citées dans la section 3.2.2.

Nous introduisons ces équations de corrections comme les étapes précédentes afin de faciliter la modélisation de certains paramètres. Nous obtenons N1PC, N2PC, WFC et FNC corrigés. Les nombres de Mach et les N1PC seront introduits à l'entrée du deuxième bloc de prédiction et les autres paramètres seront introduits à sa sortie pour effectuer l'apprentissage.

Nous présentons les résultats de chacune des sorties par rapport aux données du simulateur (les tests de vol) pour la position de la manette de gaz la plus basse (TLA = 35 degrés) et la plus haute (TLA = 65 degrés) comme montré dans l'étape précédente montrant les résultats de la prédiction du N1PC par le premier bloc de réseaux de neurones.

4.2.2.1 Les résultats de la vitesse de rotation du moteur à haute pression N2PC

Nous commençons par l'affichage de N2PC pour la position de la manette de gaz TLA = 35 degrés en fonction de l'altitude et des nombres de Mach :

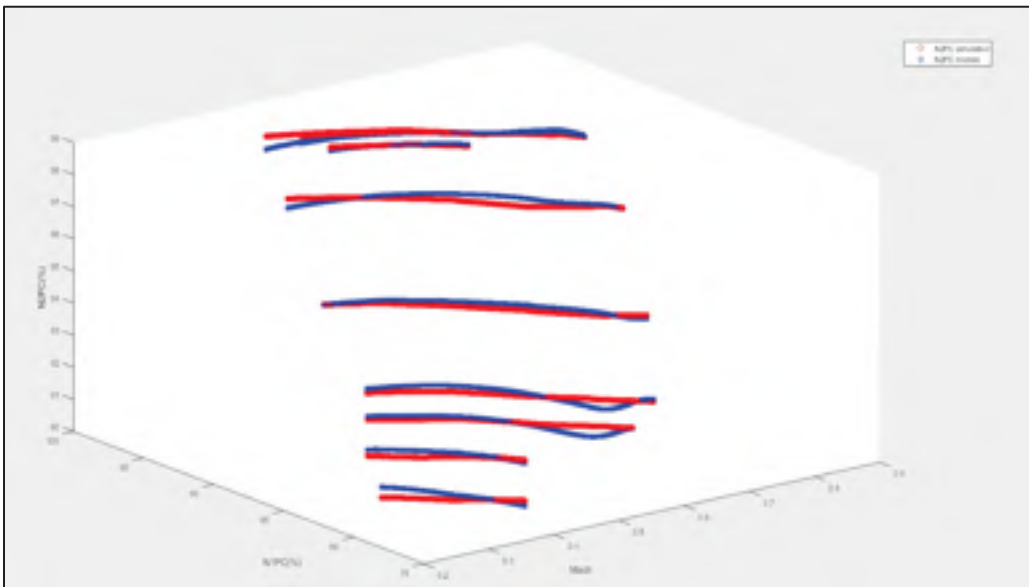


Figure 4.8 N2PC du simulateur et N2PC du modèle pour TLA = 35 degrés

La figure 4.8 montre la variation de N2PC en fonction de N1PC et du nombre de Mach. Ainsi, la comparaison entre N2PC du simulateur et N2PC du modèle, présente une erreur relative moyenne de $9.71 \cdot 10^{-2} \%$ pour l'identification du modèle. De ce fait, la précision de la prédiction de N2PC est assurée pour cette position de la manette de gaz TLA. La figure 4.9 présente les résultats concernant la prédiction de N2PC pour la position de manette de gaz TLA = 65 degrés en fonction de N1PC et du nombre de Mach :

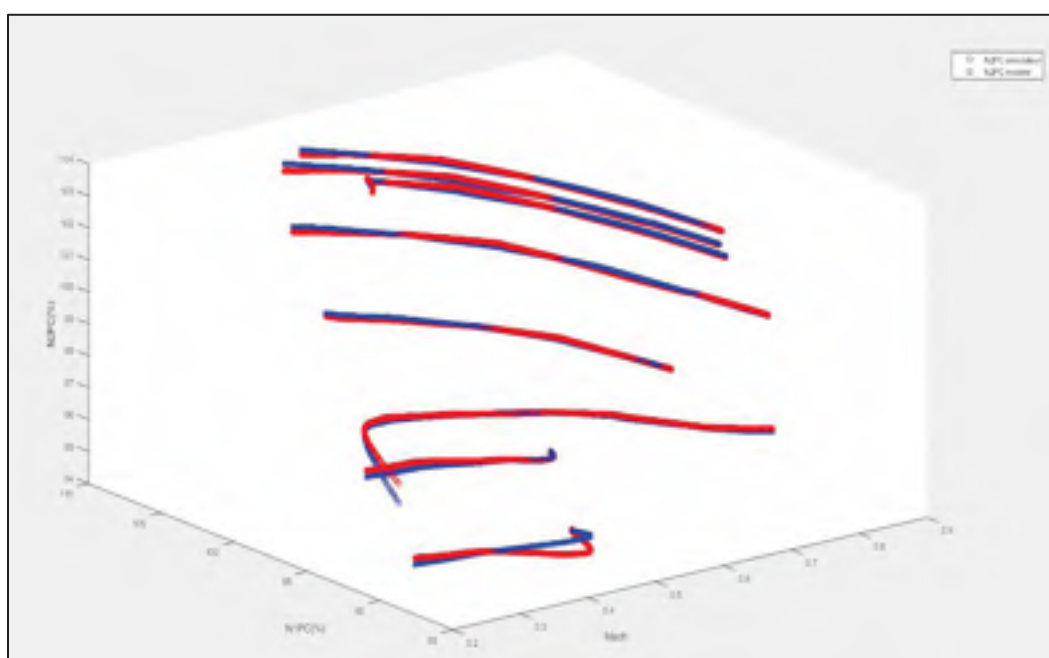


Figure 4.9 N2PC du simulateur et N2PC du modèle pour TLA = 65 degrés

Nous présentons les erreurs relatives moyennes de N2PC pour chacune des positions de la manette de gaz TLA lors de l'identification du modèle dans la figure 4.10 ci-dessous :

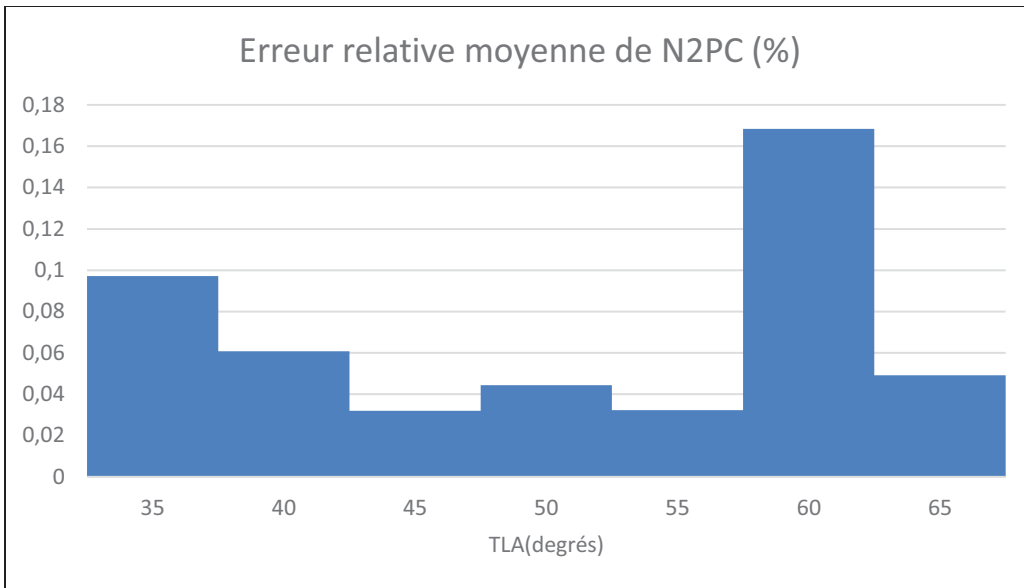


Figure 4.10 Erreur relative moyenne de N2PC pour l'identification du modèle

Une moyenne d'erreur maximale est de 0.16% pour la position TLA = 65 degrés permettrait d'obtenir une très bonne identification de N2PC en fonction du nombre de Mach et de N1PC (figure 4.10). La validation avec les données non utilisées à l'apprentissage est présentée ci-dessous :

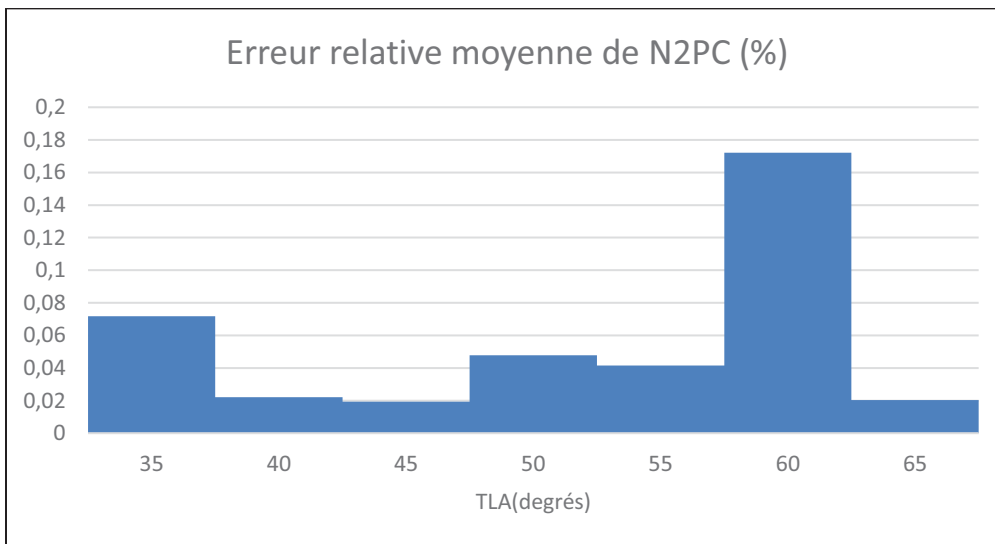


Figure 4.11 Erreur relative moyenne de N2PC pour la validation du modèle

Concernant la validation du modèle, l'erreur relative moyenne maximale de N2PC est de 0.17% pour la position de TLA = 60 degrés. La prédiction de N2PC dépend du comportement de N1PC et du nombre de Mach. Nous présentons ensuite la poussée nette du moteur FNC pour chacune des positions de TLA.

4.2.2.2 Les résultats de la poussée nette du moteur (FNC)

Après l'apprentissage de la deuxième boîte noire, nous obtenons aussi les valeurs prédites de la poussée nette du moteur FNC par le modèle en fonction de N1PC et du nombre de Mach. La comparaison de ces valeurs sera réalisée par rapport aux données du simulateur. Nous présentons comme précédemment les valeurs de la poussée FNC pour TLA = 35 degrés et TLA = 65 degrés, ainsi que les erreurs relatives pour toutes les autres positions intermédiaires de la manette de gaz TLA. Nous commençons par exposer la variation de FNC (du modèle et du simulateur) avec le nombre de Mach et FNC pour la position TLA = 35 degrés :

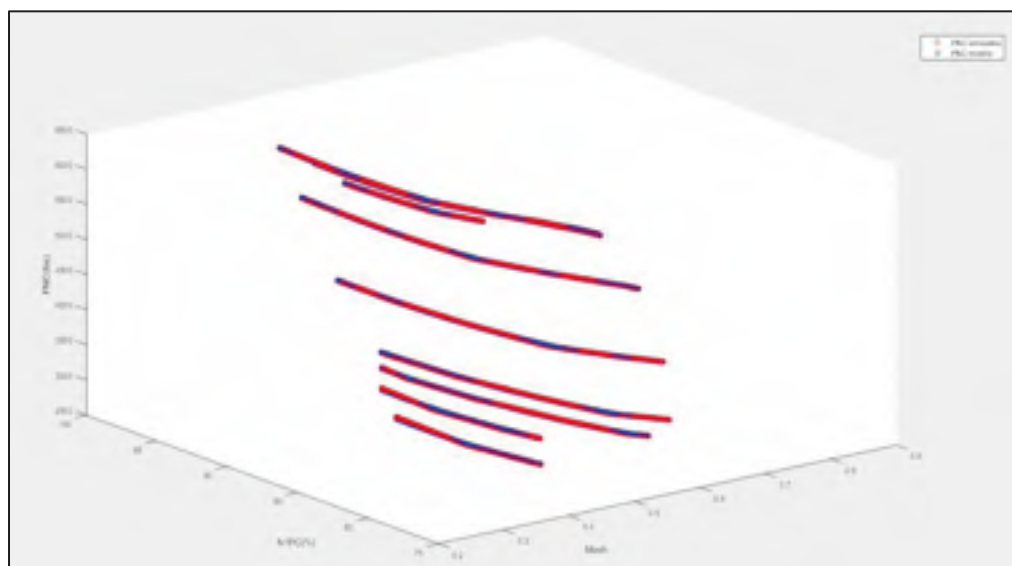


Figure 4.12 FNC du simulateur et FNC du modèle pour TLA = 35 degrés

La moyenne de l'erreur de prédiction de la poussée nette du moteur FNC en fonction de N1PC et du nombre de Mach pour cette position TLA = 35 degrés est $7.64 \cdot 10^{-2} \%$. Nous présentons

également la variation du FNC avec NIPC et le nombre de Mach pour la position TLA = 65 degrés :

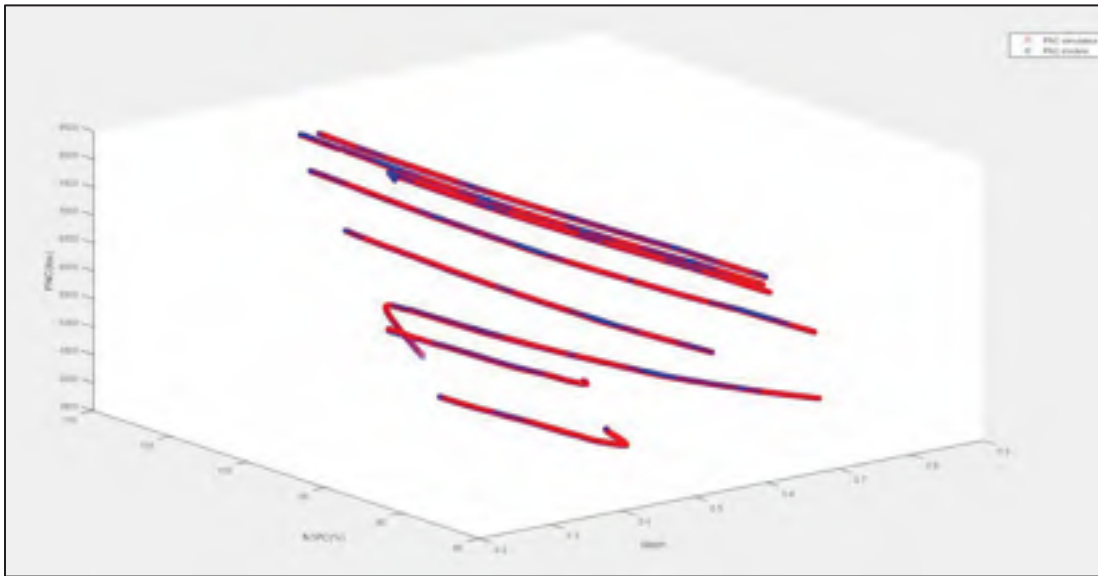


Figure 4.13 FNC du simulateur et FNC du modèle pour TLA = 65 degrés

La figure 4.14 présente la variation des moyennes des erreurs relatives de FNC pour les différentes positions intermédiaires (entre 35 et 65 degrés) de la manette de gaz après avoir effectué l'apprentissage.

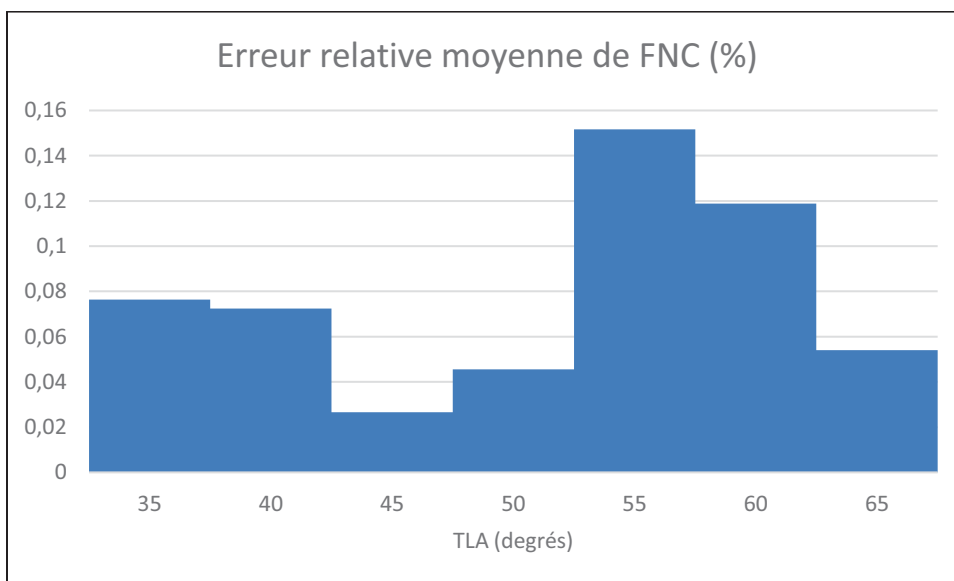


Figure 4.14 Erreur relative moyenne de FNC en identification du modèle

Nous présentons également la variation de l'erreur relative moyenne de FNC en validation du modèle :

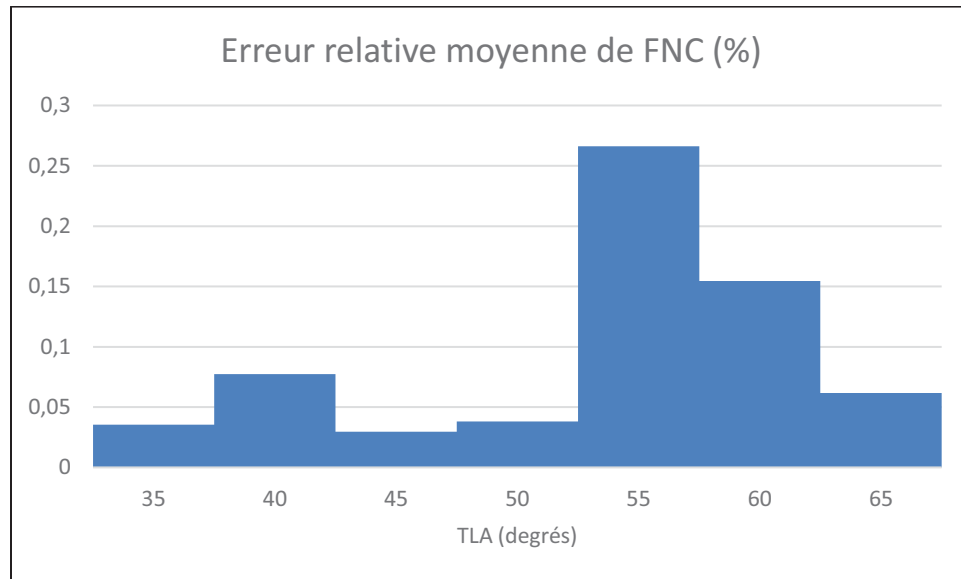


Figure 4.15 Erreur relative moyenne de FNC en validation du modèle

La prédiction de la poussée nette du moteur FNC a présenté une moyenne d'erreur relative "maximale" en identification du modèle de 0.15 % pour la position de manette de gaz TLA = 55 degrés et 0.26 % en validation modèle pour la même position de la TLA. Cette poussée FNC varie selon le nombre de Mach et varie indirectement selon l'altitude car N1PC varie selon l'altitude, ce qui prouve l'équation 3.2 du troisième chapitre montrant la variation de la poussée du moteur en fonction de la densité de l'air qui elle-même varie en fonction de l'altitude.

La boîte noire qui nous a donné la rotation du moteur à haute pression N2PC ainsi que sa poussée nette FNC, nous permet d'obtenir la consommation du carburant WFC.

4.2.2.3 Les résultats de la consommation du carburant du moteur (WFC)

La prédiction de la consommation du carburant du moteur WFC sera présentée pour TLA = 35 degrés et 65 degrés. Nous commençons tout d'abord par présenter la variation du WFC en

fonction du N1PC et du nombre de Mach pour la TLA = 35 degrés dans la figure 4.16, et ensuite nous montrons la variation du WFC en fonction du N1PC et du nombre de Mach pour la TLA = 65 degrés dans la figure 4.17 :

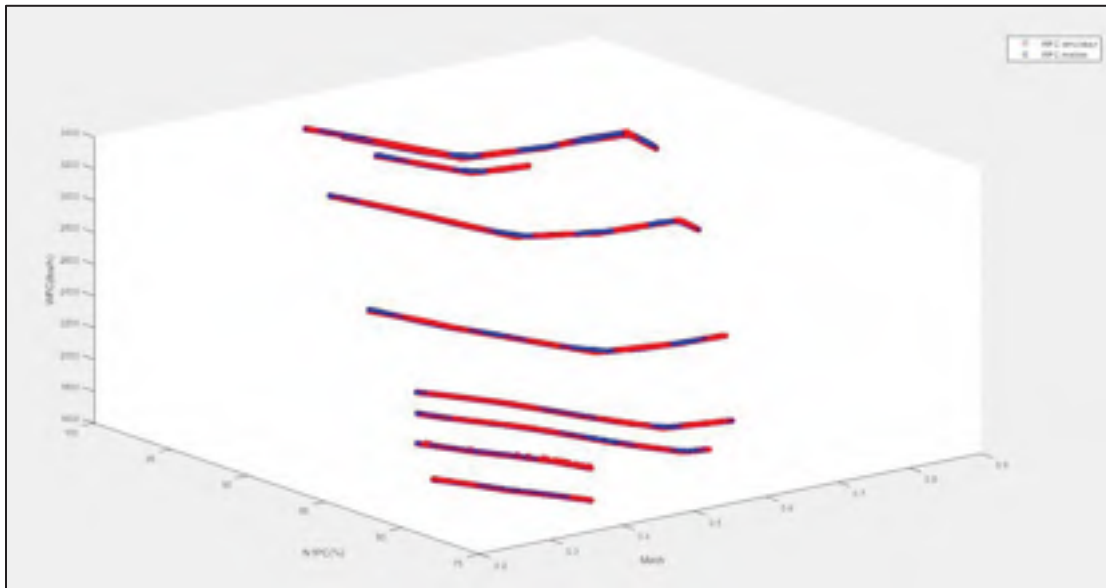


Figure 4.16 WFC du simulateur et WFC du modèle pour TLA = 35 degrés

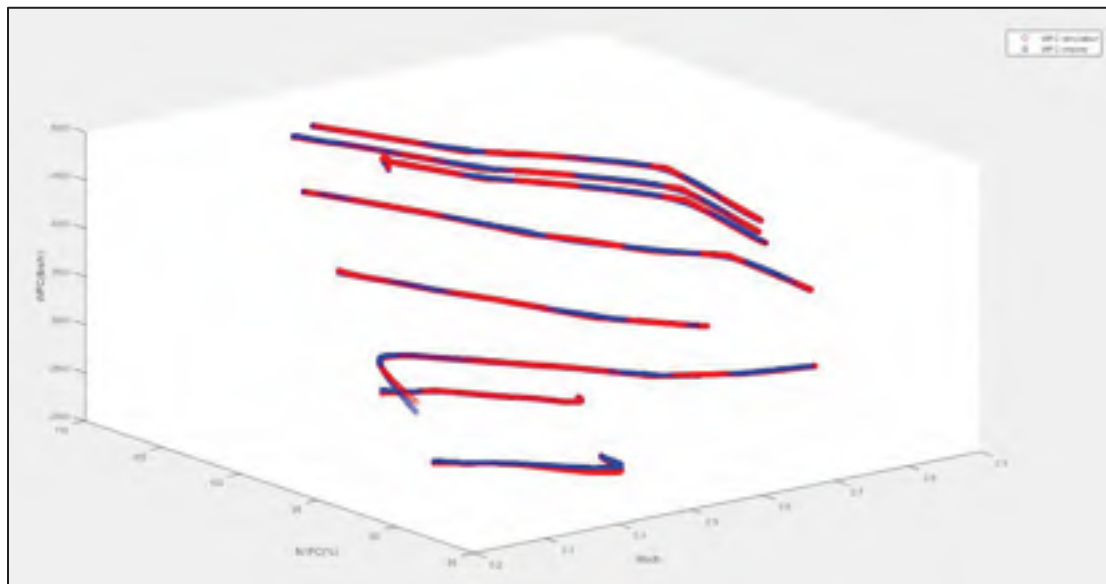


Figure 4.17 WFC du simulateur et WFC du modèle pour TLA = 65 degrés

La moyenne d'erreur de prédiction de la consommation de carburant du moteur WFC pour la position TLA = 35 degrés est de 0.10 % en identification. Nous montrons dans la figure 4.18 les erreurs relatives moyennes de prédiction de la consommation de carburant WFC par rapport aux valeurs du simulateur en identification et en validation du modèle :

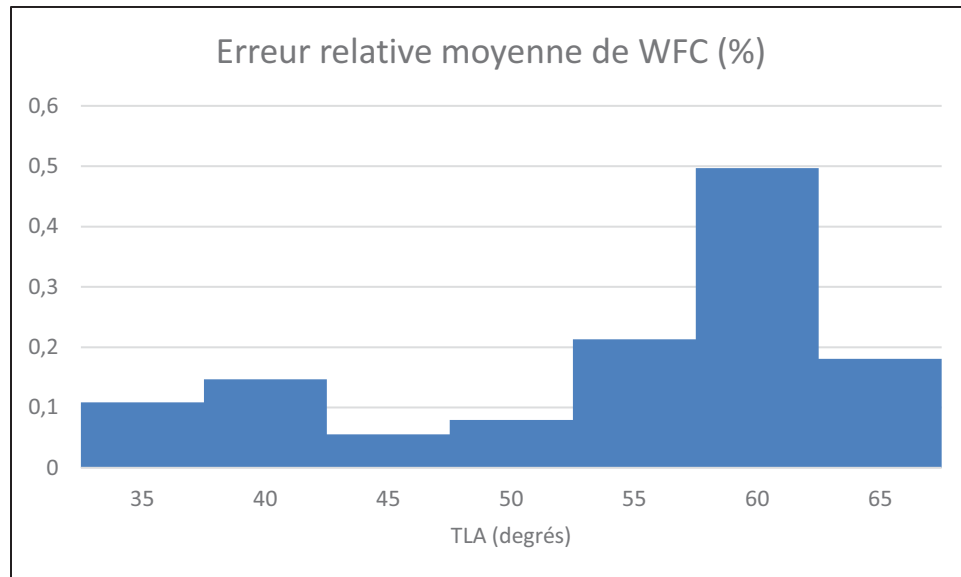


Figure 4.18 Erreur relative moyenne de WFC en identification du modèle

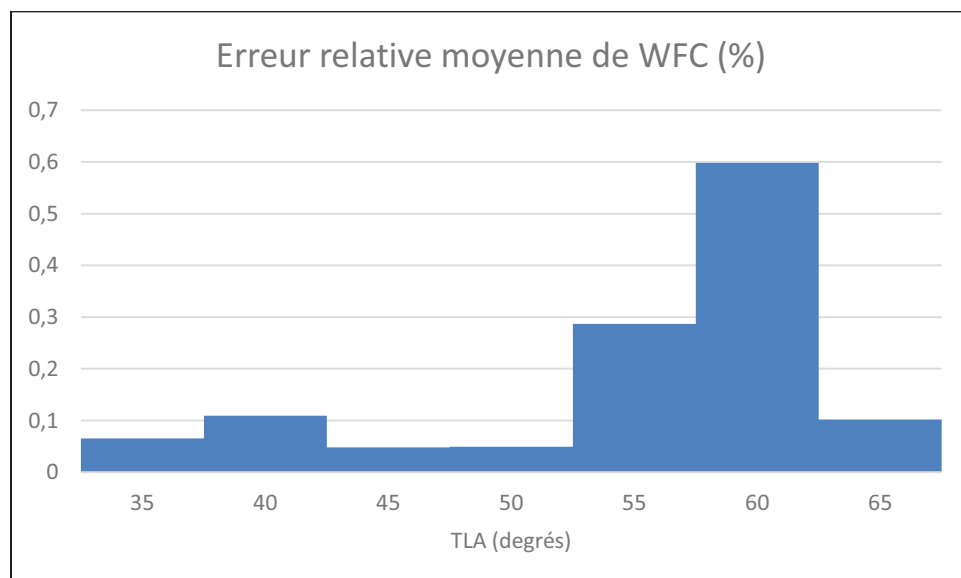


Figure 4.19 Erreur relative moyenne de WFC en validation du modèle

Une erreur relative moyenne maximale de WFC est de 0.49 % en identification et 0.59 % en validation pour la position TLA = 60 degrés. Après avoir présenté les résultats suite à l'apprentissage du deuxième bloc en fonction des paramètres tels que le nombre de Mach, N1PC, N2PC, FNC et WFC, nous constatons que la conception du deuxième bloc a été confirmée grâce au faible pourcentage d'erreur présenté pour ces paramètres et pour les différentes positions de manette de gaz.

Nous allons maintenant arranger les deux blocs en cascade afin d'obtenir les résultats pour notre modèle du moteur.

4.2.3 Résultats du modèle complet

Le but de notre modèle du moteur est de prédire la poussée nette du moteur, sa consommation du carburant ainsi que la vitesse de rotation à haute pression en introduisant l'altitude désirée, le nombre de Mach désiré, la position de manette de gaz TLA et la température qui sera calculée en fonction de l'altitude. Dans cette partie, nous exposerons les résultats obtenus pour chacune des positions de manette de gaz TLA en fonction des altitudes et des Mach.

4.2.3.1 Résultats obtenus pour la vitesse de rotation du moteur à haute pression N2PC

Comme dit précédemment, nous afficherons les résultats obtenus pour TLA = 35 degrés et TLA = 65 degrés. Nous commençons tout d'abord par afficher la variation de N2PC en fonction du nombre de Mach et de l'altitude pour la TLA = 35 degrés :

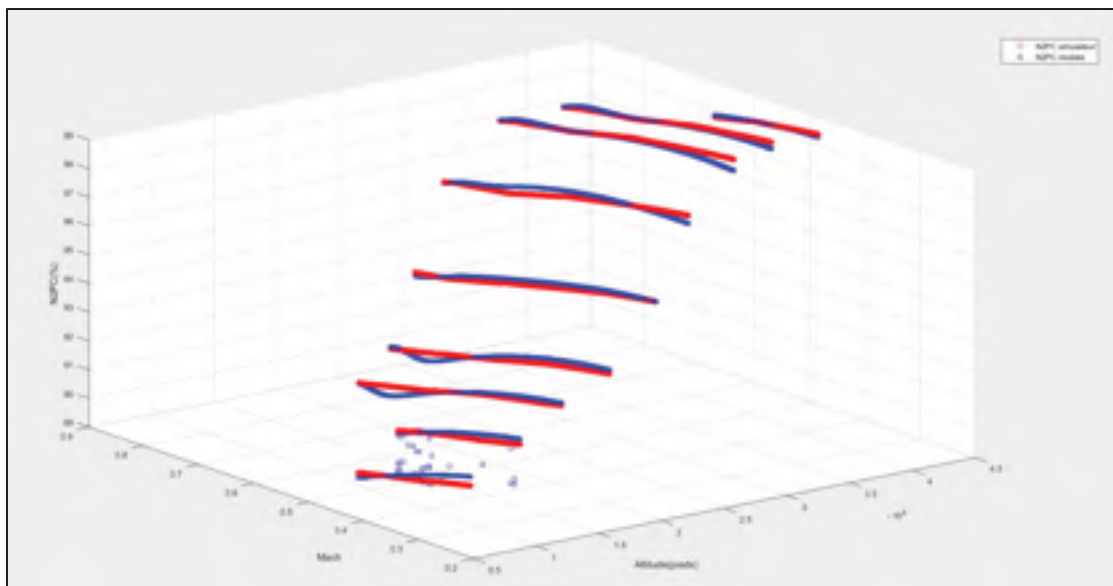


Figure 4.20 N2PC du simulateur et N2PC du modèle pour TLA = 35 degrés

Cette prédiction de N2PC a révélé une erreur moyenne de $9.76 \cdot 10^{-2} \%$. En d'autres termes, le modèle a réussi d'assurer une bonne identification des paramètres pour cette position TLA en fonction des paramètres d'entrées (altitude et nombre de Mach). Les points de N2PC qui sont présentés (en bleu) dans la figure 4.20 et qui ne sont pas bien prédits par rapport au N2PC du simulateur sont dus à la mauvaise réalisation du test de vol à cette altitude (10,000 pi) comme nous l'avons expliqué dans la section 4.2.1. Les résultats obtenus pour la variation du N2PC en fonction du nombre de Mach et de l'altitude pour la TLA = 65 degrés sont présentés ci-dessous dans la figure 4.21 :

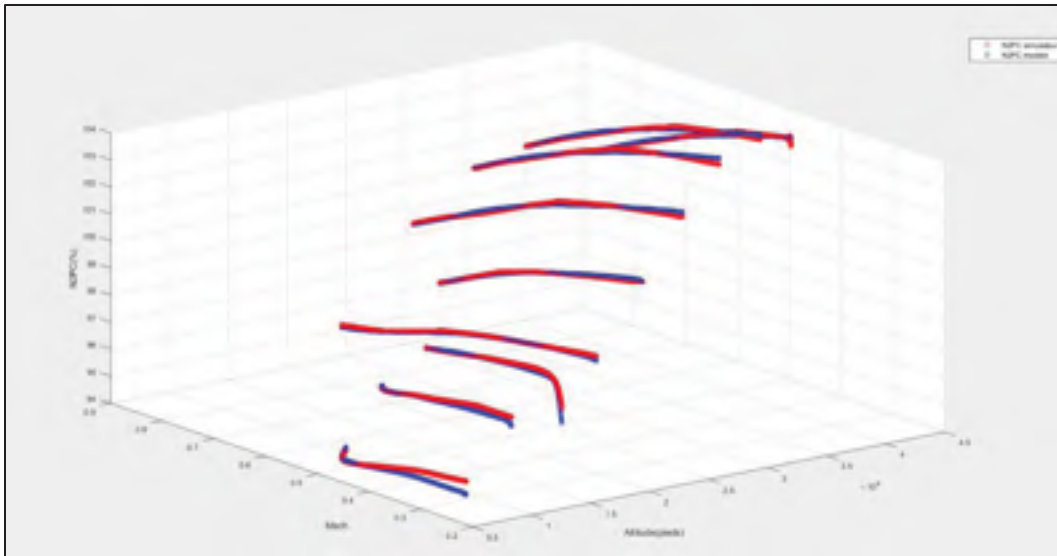


Figure 4.21 N2PC du simulateur et N2PC du modèle pour TLA = 65 degrés

L'apprentissage est réussi pour cette position TLA = 65 degrés puisque l'erreur relative moyenne entre N2PC du simulateur et celle prédite par le modèle est basse. Nous présentons dans les figures 4.22 et 4.23 les résultats obtenus pour l'erreur relative moyenne de N2PC pour les autres positions de TLA :

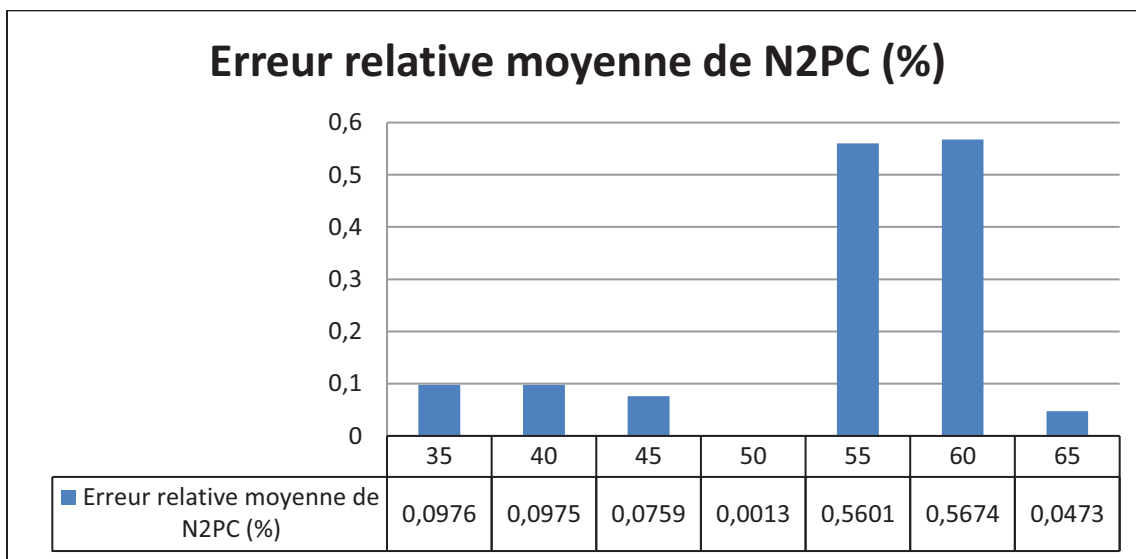


Figure 4.22 Erreur relative moyenne de N2PC pour toutes les TLA en identification

Nous avons testé les données non utilisées à l'apprentissage (25% des données) pour obtenir la réponse du modèle et nous présentons les résultats obtenus ci-dessous :

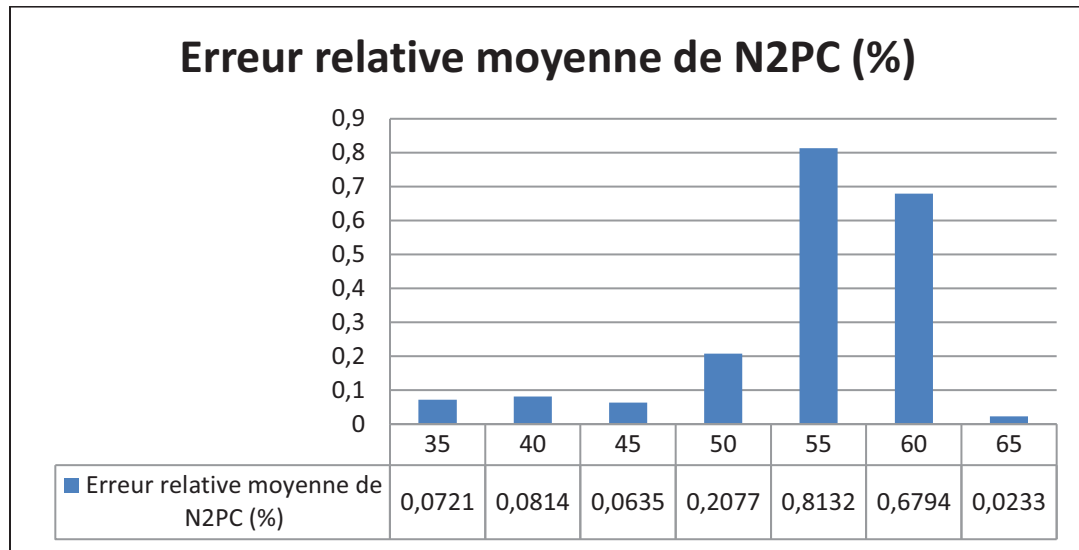


Figure 4.23 Erreur relative moyenne de N2PC pour toutes les TLA en validation

Le modèle fournit une très bonne prédiction de la vitesse de rotation du moteur à haute pression N2PC puisque l'erreur relative moyenne maximale entre les valeurs du simulateur et celles prédites par le modèle ne dépasse pas 0.82 %.

Nous suivons les mêmes étapes pour la présentation des résultats de la poussée nette du moteur FNC et de sa consommation de carburant WFC.

4.2.3.2 Les résultats de la poussée nette du moteur FNC

La poussée nette du moteur FNC est donnée en fonction de l'altitude et du nombre de Mach et bien évidemment de la température qui se calcule en fonction de l'altitude comme nous l'avons expliqué pour chacune des positions de manette de gaz TLA. Nous commençons par l'affichage de la variation de FNC avec le nombre de Mach et l'altitude pour le TLA = 35 degrés :

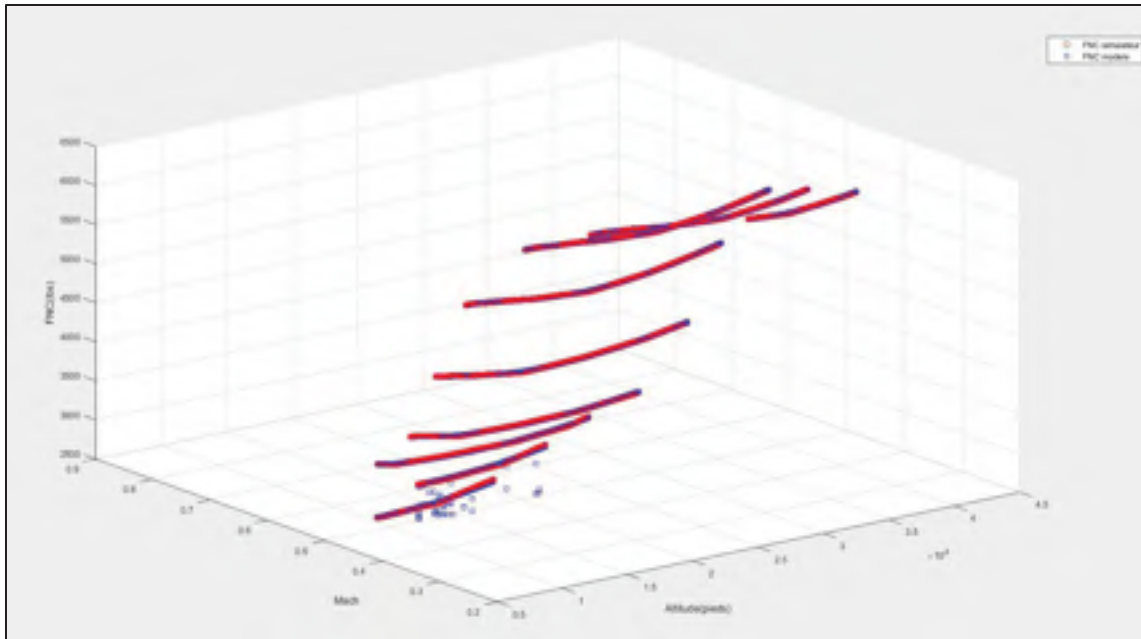


Figure 4.24 FNC du simulateur et FNC du modèle pour TLA = 35 degrés

Les résultats concernant la variation de FNC avec l'altitude et le nombre de Mach pour la position de la manette de gaz TLA = 65 degrés sont présentés dans la figure 4.25 :

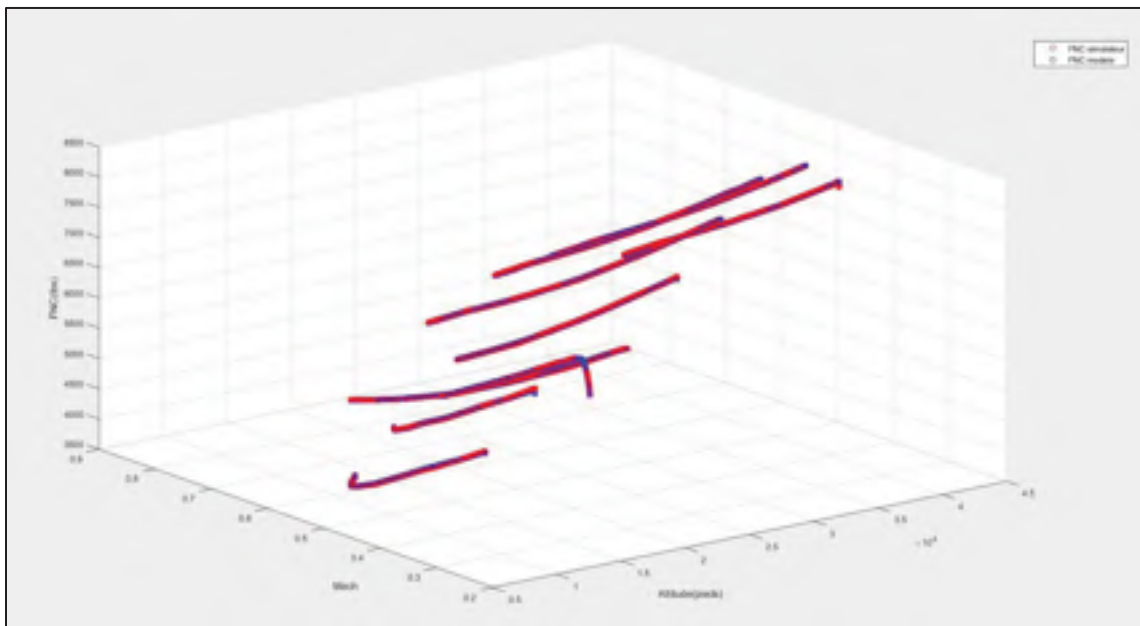


Figure 4.25 FNC du simulateur et FNC du modèle pour TLA = 65 degrés

Cette prédiction de FNC réalisé à l'aide de notre modèle a présenté une erreur relative moyenne de $8 \cdot 10^{-2} \%$ pour la position de manette de gaz TLA = 35 degrés. La figure 4.26 présente les résultats obtenus en termes d'erreurs relatives moyennes de FNC pour les autres positions de manette de gaz TLA afin de vérifier si le modèle est valide pour les autres positions de TLA ou non :

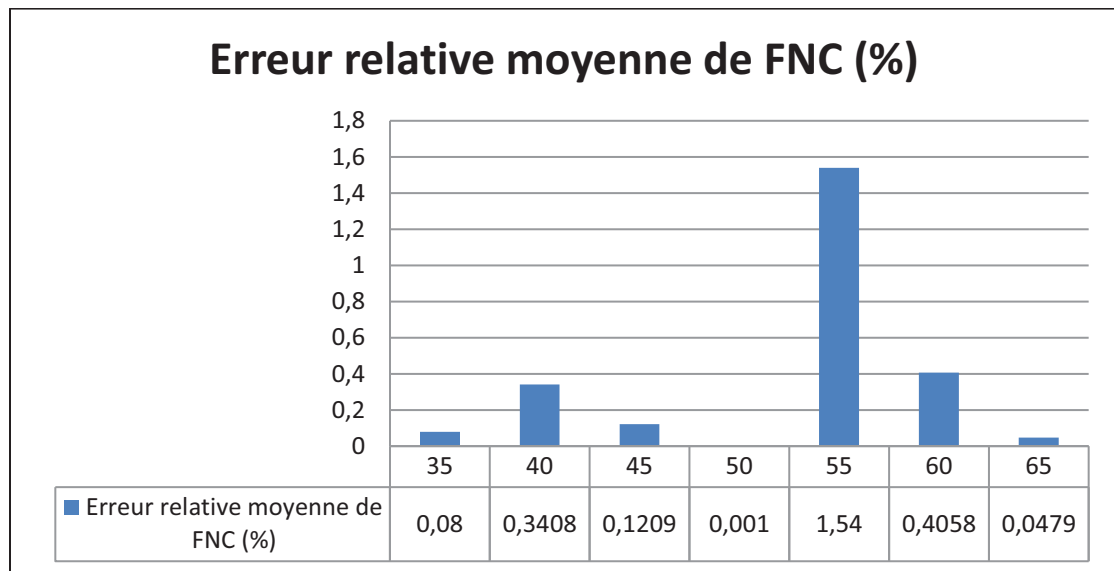


Figure 4.26 Erreur relative moyenne de FNC pour toutes les TLA en identification

L'erreur relative moyenne "maximale" de FNC en identification est 1.54% pour la position TLA = 55 degrés. Pour la validation, l'erreur relative moyenne maximale de prédiction n'a pas dépassé 1.56% pour la même position de TLA de 55 degrés :

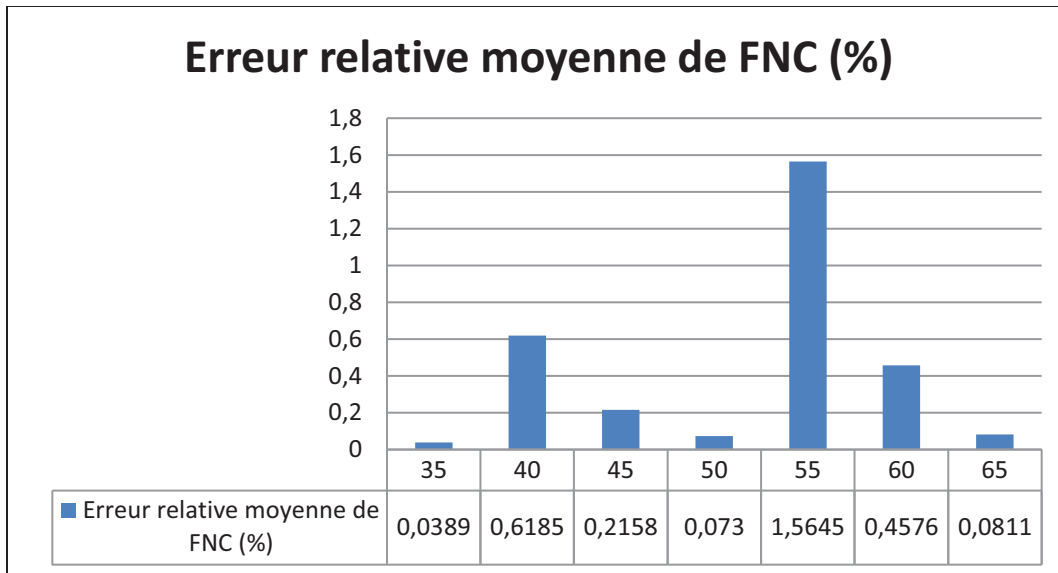


Figure 4.27 Erreur relative moyenne de FNC pour toutes les TLA en validation

Le modèle réalisé assure une très bonne prédiction de la poussée nette du moteur FNC grâce à la performance de l'apprentissage et au choix de la base de données. La validité de ces données obtenues par modélisation est réalisée par leur comparaison par rapport aux données du simulateur. L'erreur relative maximale pour la modélisation de la consommation du carburant pour les moteurs des avions ne doit pas dépasser 5% (Aviation, 2003). En d'autres termes, notre modèle confirme sa bonne identification pour la vitesse de rotation du moteur à haute pression N2PC ainsi que sa poussée nette FNC. Nous confirmons également son identification et sa validation pour la prédiction de la consommation du carburant WFC dans la section suivante.

4.2.3.3 Les résultats obtenus pour la consommation du carburant WFC

Les résultats obtenus pour la variation de la consommation du carburant du moteur prédite par le modèle en fonction du nombre de Mach et de l'altitude, sont affichés dans cette section. Au début, nous commençons par afficher la variation de WFC avec l'altitude et le nombre de Mach pour TLA = 35 degrés dans la figure 4.28 :

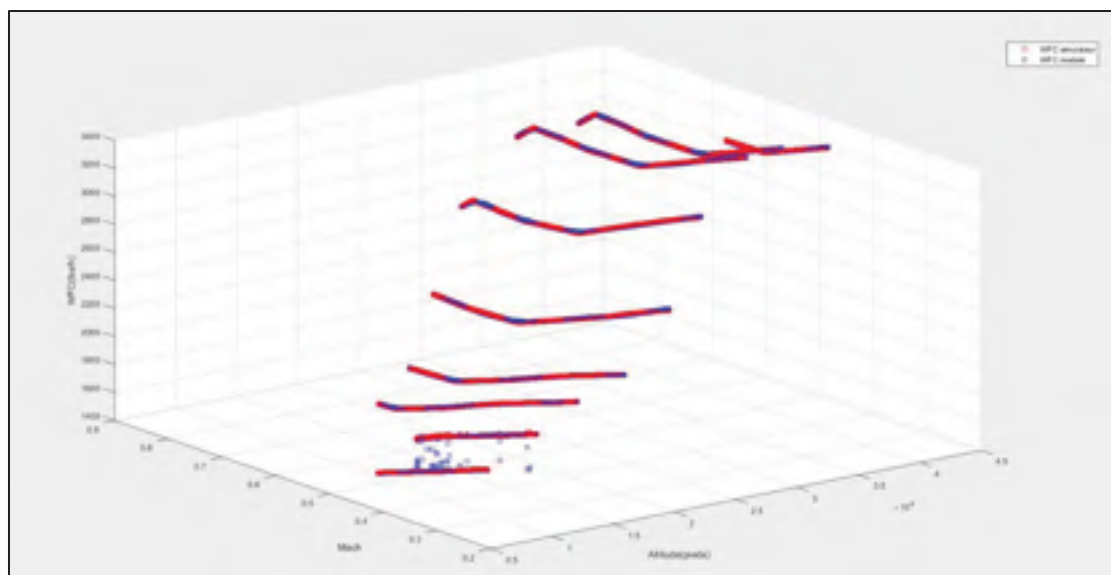


Figure 4.28 WFC du simulateur et WFC du modèle pour TLA = 35 degrés

Ensuite, la variation de WFC avec l'altitude et le nombre de Mach pour TLA = 65 degrés est présentée dans la figure 4.29 :

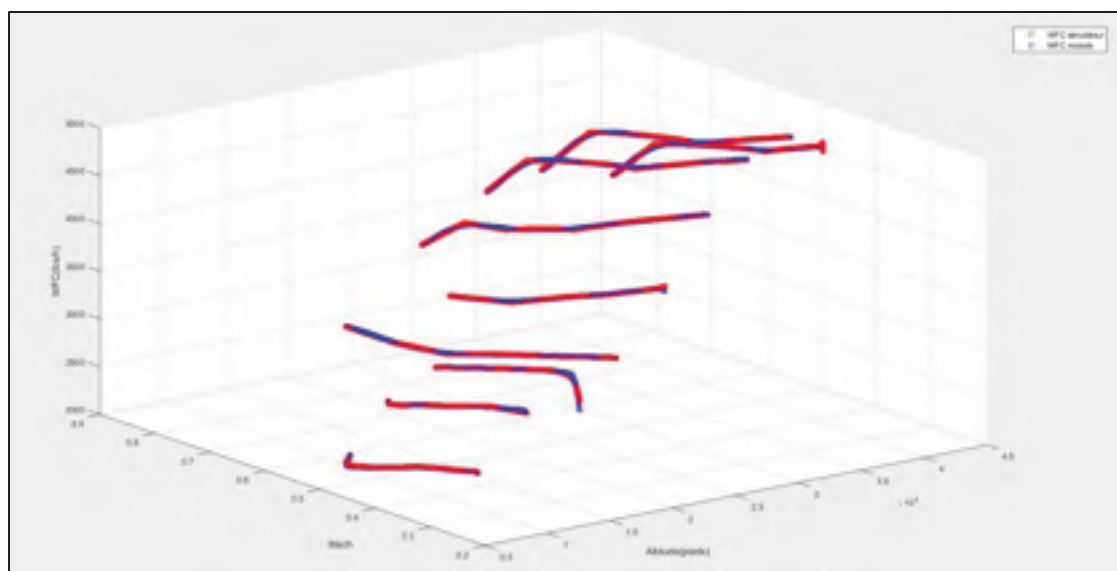


Figure 4.29 WFC du simulateur et WFC du modèle pour TLA = 65 degrés

La prédiction de la consommation du carburant WFC a présenté une erreur relative moyenne de $11.65 \cdot 10^{-2} \%$ par rapport aux données du simulateur pour la position de TLA = 35 degrés. Par contre, cette erreur a été de $13.63 \cdot 10^{-2} \%$ pour la position de TLA = 65 degrés. Nous affichons les résultats obtenus pour WFC pour les autres positions TLA afin de mettre en évidence la variation de l'erreur relative moyenne pour chacune des positions de manette de gaz TLA en identification (figure 4.30) :

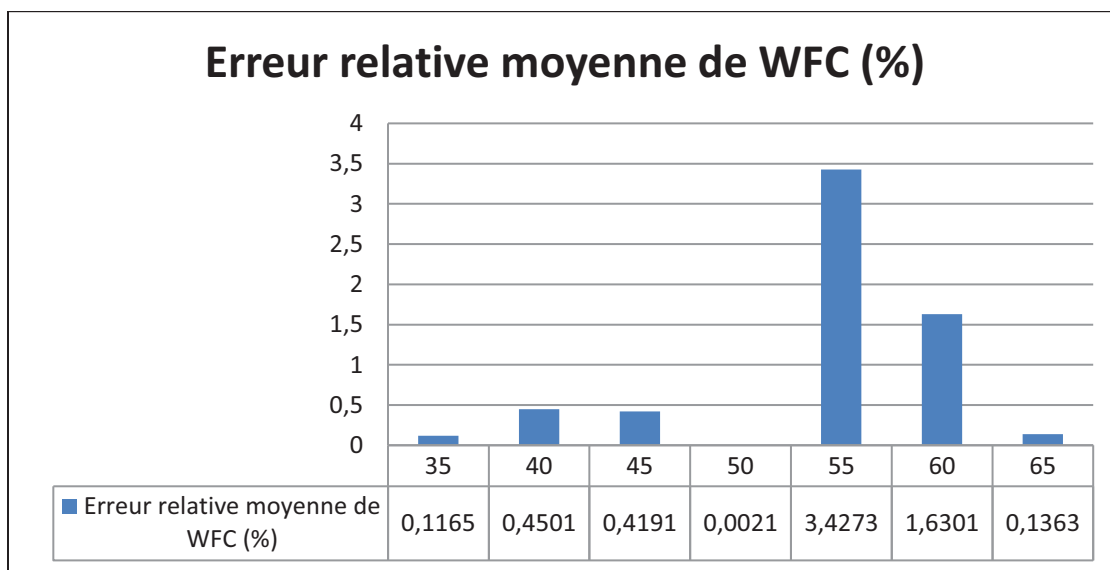


Figure 4.30 Erreur relative moyenne de WFC pour toutes les TLA en identification

Nous remarquons que l'erreur relative moyenne de la consommation du carburant WFC est plus importante que celle trouvée précédemment pour les autres paramètres et elle est maximale de 3.42 % en identification pour une position de TLA = 55 degrés. Cette valeur reste toujours acceptable puisqu'elle ne dépasse pas 5% (la limite d'erreur acceptable pour la prédiction de ce paramètre WFC). En validation, l'erreur relative maximale est de 3.28 % pour la même position de manette de gaz TLA = 55 degrés :

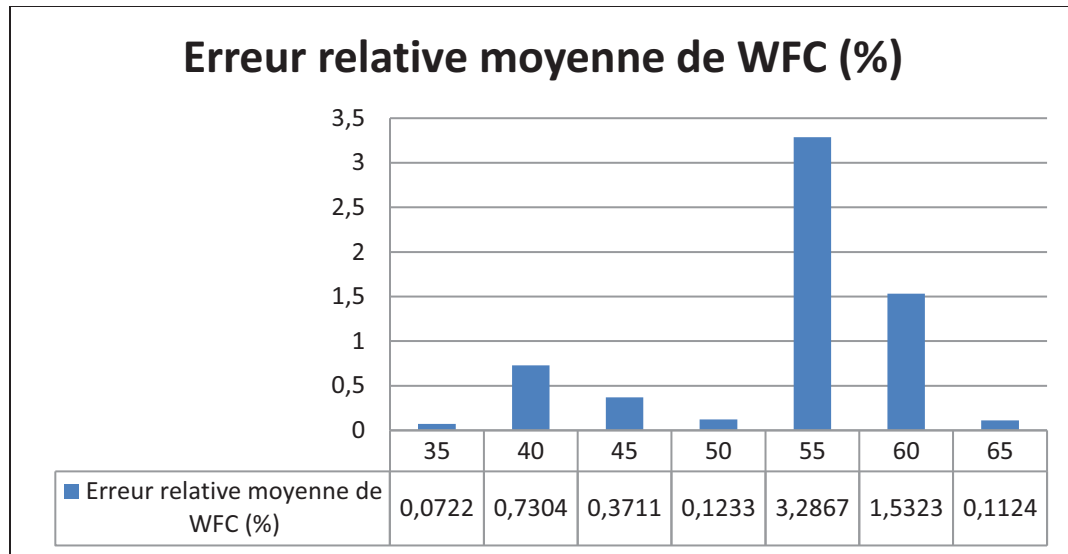


Figure 4.31 Erreur relative moyenne de WFC pour toutes les TLA en validation

Nous constatons que l'erreur relative de prédiction a augmenté après avoir associé les deux blocs qui ont subi l'apprentissage des données ensemble. Ceci est expliqué par le fait que l'erreur due à l'apprentissage du premier bloc affecte l'erreur du deuxième bloc qui se propage, ce qui fait augmenter les erreurs des sorties du modèle (sorties du deuxième bloc).

Après avoir présenté les résultats obtenus pour les différents paramètres du moteur, nous avons constaté que ces résultats ont été parfaits pour un modèle statique en termes des positions de manette de gaz TLA. Nous comparons les résultats obtenus par rapport aux travaux cités dans la revue de littérature (Ghazi et al., 2015) (Rodriguez et al., 2012) qui sont représentés dans le tableau 4.2 :

Tableau 4.2 Comparaison des erreurs de prédiction de la poussée nette des moteurs par différents modèles

	Modèle basé sur les réseaux de neurones	Modèle mathématique de (Ghazi et al.,2015)	Modèle générique (Rodriguez et al.,2012)
Erreur relative moyenne de la poussée nette des moteurs	$\leq 1.56 \%$	1.58 %	entre 6 % et 14 % selon le type des moteurs

L'efficacité des réseaux de neurones est claire à travers ces résultats. En effet, la non linéarité du moteur de l'avion pourrait être modélisé par ces réseaux. La performance de ces réseaux a été assurée par l'algorithme d'optimisation du grand déluge étendu qui a permis un apprentissage réussi avec des réseaux optimisés.

Nous pouvons comparer aussi les résultats de la consommation du carburant WFC de notre modèle par rapport à d'autres modèles existants (Roux, 2002), (Torenbeek, 1982) :

Tableau 4.3 Comparaison des erreurs de la prédiction de la consommation du carburant par différents modèles

	Modèle basé sur les réseaux de neurones	Modèle de Roux, 2002	Modèle de Torenbeek, 1982
Erreur relative moyenne de la consommation du carburant des moteurs	$\leq 3.42 \%$	8 %	6.6 %

Concernant la prédiction de la consommation du carburant, le modèle réalisé à base des réseaux de neurones présente la meilleure (la plus basse) erreur de prédiction parmi les erreurs des autres modèles existants et cités dans la littérature. En d'autres termes, les réseaux de neurones

peuvent être utiles pour la modélisation des systèmes possédant un grand nombre de paramètres avec un fonctionnement plus ou moins complexe.

À travers ce chapitre, nous avons présenté les résultats obtenus pour le modèle du moteur réalisé qui se base sur les techniques des réseaux de neurones. Ces résultats ont été présentés pour chacune des étapes de sa réalisation. La validation du modèle a été réalisée par rapport aux données des tests de vol réalisés avec le simulateur du LARCASE et en se basant sur les informations provenant du guide des critères de qualification des simulateurs de vol présenté par l'Organisation Internationale de l'Aviation Civile. L'objectif essentiel du projet a été atteint grâce à ce modèle qui a permis d'identifier les paramètres du moteur pour les différentes positions de manette de gaz TLA pendant la phase de croisière. Le principe suivi dans ce mémoire pourrait être considéré pour la réalisation du modèle dans d'autres phases de vol. Grâce à ce modèle, Le principe de l'intelligence artificielle et plus précisément les réseaux de neurones s'est avéré utile pour l'identification des paramètres des moteurs des aéronefs.

CONCLUSION

La modélisation des moteurs des avions tient compte de plusieurs hypothèses afin de garantir l'obtention d'un modèle de moteur fiable et précis. En effet, un tel modèle sert à gagner en temps des tests de vol qui se réalisent à l'aide du simulateur de vol et qui pourraient prendre énormément de temps. L'analyse du fonctionnement du moteur et l'anticipation des prises de décisions ont pour but d'améliorer les performances de l'avion.

L'objectif de cette étude a été d'identifier certains paramètres du moteur de l'avion Cessna Citation X. Dans ce but, une méthodologie d'identification basée sur les réseaux de neurones a été créée. Le choix des réseaux de neurones a été réalisé grâce à leurs possibilités de prédiction des paramètres des systèmes non linéaires ainsi que leurs facilités d'utilisation. L'utilisateur doit obtenir un modèle du moteur, afin d'introduire ses paramètres d'entrées et obtenir ses paramètres de sortie correspondantes désirées.

L'identification du modèle se base essentiellement sur les données du simulateur de vol niveau D de LARCASE. Tout d'abord, une étude a été réalisée sur les différents types des moteurs des aéronefs et les méthodologies qui ont été utilisées pour leur modélisation dans la Section 1.5. Certes, les résultats trouvés dans la littérature étaient intéressants ; mais une nouvelle méthodologie a été trouvée en se basant sur les réseaux de neurones et sur les tests de vol du simulateur. En d'autres termes, l'apprentissage artificiel qui se base sur les réseaux de neurones permet un apprentissage de n'importe quel fonctionnement d'un système ainsi qu'une précision de prédiction de ses paramètres.

Les réseaux de neurones et leurs algorithmes d'apprentissage ont été présentés dans le deuxième chapitre dans le but de comprendre leur fonctionnement et pour les utiliser ensuite dans la conception du modèle du moteur.

La méthodologie de cette conception s'est basée sur une étude des paramètres représentant le fonctionnement du moteur ainsi que les relations qui relient ses fonctions. Évidemment, les

tests de vol réalisés avec le simulateur de vol de LARCASE ont permis de valider ces relations entre ces différents paramètres, ainsi de servir à l'apprentissage et la validation de ce modèle.

Les résultats obtenus pour ce modèle ont montré une excellente précision pour la prédiction des paramètres du moteur. Heureusement, dans notre cas, le simulateur de vol nous a facilité la tâche grâce à la possibilité de réaliser tous les tests de vol désirés pour l'avion Cessna Citation X.

En contrepartie, ce modèle est valide pour une enveloppe de vol bien déterminée. Il a été conçu pour la phase de croisière uniquement. De ce fait, nous suggérons de refaire des tests pour les phases de montée et de descente mais aussi de refaire l'apprentissage avec la même méthodologie dans le but d'avoir un modèle représentant tout le fonctionnement du moteur pour les trois phases de vol. L'utilisation d'autres algorithmes d'optimisation comme par exemple "la recherche tabou" ou "le recuit simulé" pourraient être combinés avec les méthodologies des réseaux de neurones, mais ne peuvent pas garantir l'obtention de meilleurs résultats que ceux obtenus par notre méthodologie.

Nous proposons aussi, d'intégrer le modèle du moteur de l'avion Cessna Citation X dans le modèle global aéro-propulsif (avion et moteur) réalisé par l'équipe de simulation de LARCASE afin d'étudier les performances et de prédire ses coefficients aérodynamiques de ce modèle global.

En conclusion, nous pouvons dire que les méthodologies des réseaux de neurones en combinaison avec celles du grand déluge étendu, ont assuré la prédiction des paramètres du moteur pour un type d'avion dans certaines conditions de vol. Le même modèle pourrait être utilisé pour d'autres types d'avions mais, il faudrait s'assurer de la disponibilité de leurs données de vol nécessaires pour l'apprentissage. Les paramètres estimés du moteur à partir du modèle d'identification réalisé pourraient être utilisés dans d'autres modèles plus complexes des moteurs.

LISTE DE RÉFÉRENCES BIBLIOGRAPHIQUES

- Amiguet, M. (2008). Recherche heuristique et méta-heuristique. *matthieu amiguet*. Tiré de <http://www.matthieuamiguet.ch/media/documents/MA-IARTI-03-RechHeuristique.pdf>
- Anderson, J. D. (1999). *aircraft performance and design* Boston: McGraw-Hill.
- Aviation, O. I. C. (2003). *Manual of criteria for the qualification of flight simulators*. ICAO Journal.
- Aydin, H., Turan, O., Karakoc, T. H., & Midilli, A. (2015). Exergetic Sustainability indicators as a tool in commercial aircraft : A case of study for a turbofan engine. *International Journal of Green Energy* 12(1), 28-40.
- Bardela, P. A., & Botez, R. M. (2017). Identification and Validation of the Cessna Citation X Engine Component Level Modeling with Flight Tests. In *AIAA Modeling and Simulation Technologies Conference* (p. 1942).
- Bardela, P. A., Botez, R. M., & Pageaud, P. (2017) *CESSNA CITATION X ENGINE MODEL EXPERIMENTAL VALIDATION*.
- Ben Mosbah, A. (2011). *Optimisation de l'ordonnancement cellulaire avec métaheuristiques*. Montréal:
- Ben Mosbah, A., Botez, R., & Dao, T. M. (2016). Hybrid Original Approach for the Prediction of the Aerodynamic Coefficients of an ATR-42 Scaled Wing Model. *Chinese Aeronautical Journal*, 29(1), 41-52.
- Blake, W. (2009). *Jet Transport Performance Methods*. Washington: Boeing.
- Bouillon, M. (2012). *Apprentissage incrémental et décrémental* Rennes: HAL.
- Burke, E., Bykov, Y., Newall, J., & Petrovic, S. (2004). A time-predefined local search approach to exam timetabling problems. *IIE transactions*, 36(6), 509-528.
- Chahbani, S. (2015). *Estimation des masses, des centres de gravité ainsi que des moments d'inertie de l'avion Cessna Citation X*. Montréal école de technologie supérieure
- Cormier, G. (2005). Tiré de http://www8.umoncton.ca/umcm-cormier_gabriel/SystemesIntelligents/GIND5439_Chapitre2.pdf
- Daniel, P. R. (1992). *Aircraft design : a conceptual approach*. Washington: American Institute Of Aeronautics and Astronautics.

- De Magnitot, D., de Fougerolle, T., & Hamel, M. (réalisateurs). (2010). L'avion du futur. *avionaprespetrole.e-monsite.com*. Tiré de <http://avionaprespetrole.e-monsite.com/pages/ameliorations-techniques/quelles-ameliorations-concernant-la-propulsion-de-l-avion-pourraient-permettre-une-economie-e-carburant.html>
- Doncescu, A. (2007). Tiré de http://conf.laas.fr/ignotus/archives/Doncescu_reseaux_neurones.pdf
- Dreyfus, G., Martinez, J.-M., Samuelides, M., Gordon, M. B., Badran, F., Thiria, S., & Héroult, L. (2002). *réseaux de neurones*: Eyrolles.
- Duech, G. (1993). New Optimization Heuristics : The Great Deluge Algorithm and the Record-to-Record Travel. *Journal Of Computational Physics*, 104(1), 86-92.
- Engineering Sciences Data Unit, E. (Mai 1982). Approximate methods for estimation of cruise range and endurance : aeroplanes with turbojet and turbofan engines. *IHS ESDU*, 8.
- Gaëtan (réalisateur). (2009, Janvier 01). les types de propulseurs à réactions Tiré de www.avionslegendaires.net/dossier/la-propulsion-a-reaction/typologie-des-propulseurs
- Ghazi, G., Botez, R., & Achigui, J. M. (2015). Cessna Citation X Engine Model Identification from Flight Tests. *SAE International Journal Aerospace*, 8(2), 203-213.
- Girard, A. (2007). *identification des coefficients aérodynamiques et commande de vol non linéaire*. Montréal: école polytechnique de Montréal.
- Girard, G., Yang, A., Sebamalai, C., & Magne, R. (réalisateurs). (2016). Le moteur à réaction Tiré de <http://moteurareaction.e-monsite.com/pages/physique/fonctionnement.html>
- Girardet, B. (2014). *Trafic Aérien : détermination optimale et globale des trajectoires d'avions en présence du vent*. Toulouse
- Glover, F. (1986). Future Paths for Integer Programming and Links to Artificial Intelligence. *Computers and Operations Research*, 13(5), 533-549.
- Goder, S. (2015). Comment ça vole ? Tiré de www.commentcavole.com/les-moteurs-d'avion-generalites/
- Grigor'ev, V. A., Kalabukhov, D. S., & Rad'ko, V. M. (2015). Application of Neural Network Approximation Methods in the Generalization and Presentation of th Aircraft Gas Turbine Engine Turbomachinery Characteristics. *Russian Aeronautics*, 58, 48-53.
- Grigorie, L. T., Botez, R. M., & Popov, A.-V. (2013). How the airfoil shape of a morphing wing is actuated and controlled in a smart way. *Journal of Aerospace Engineering*, 28(1).

- Haykin, S. (1999). *Neural networks : a comprehensive foundation*. New york: Prentice Hall
- Hung, M. H., Li-Sun, S., Ho, S.-J., Hwang, S.-F., & Ho, S.-Y. (2008). A Novel Intelligent Multiobjective Simulated Annealing Algorithm for Designing Robust PID Controllers. *IEEE Transactions on Systems, Man, and Cybernetics*, 319-330.
- Ioan-Doré, L. (2002). *Identification des modèles de procédés* Grenoble:
- Kammakh, M. H., & Sarr, S. (2014). *Méthode de recherche Tabou* Paris:
- Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by Simulated Annealing. *Science*, 220(4598), 671-680.
- Kobayashi, T., & Simon, D. L. (2005). Hybrid Neural-Network Genetic-Algorithm Technique for Aircraft Engine Performance Diagnostics. *Journal Of Propulsion and Power*, 21(4), 751-758.
- Koreanschi, A., Gabor, O. S., Acotto, J., Botez, R. M., Mamou, M., & Mebarki, Y. (2016). *A Genetic Algorithm Optimization Method for a Morphing Wing Tip Demonstrator Validated Infra Red Experimental Data*. Communication présentée à 34th AIAA Applied Aerodynamics Conference Washington (p. 4037).
- Koreanschi, A., Sugar-Gabor, O., & Botez, R. M. (2016). Drag Optimization of a Wing Equipped with a Morphing Upper Surface. *The Aeronautical Journal*, 120(1225), 473-493.
- LaBarre, M.-O. (2002). *Algorithmes génétiques*
- LaBarre, M. O. (2002). *Réseaux de neurones*.
- Leclercq, T. (réalisateur). (2015, Aout 31). Historiaviation. Tiré de <http://www.historiaviation.com/pages/fiches-des-avions/cessna/cessna-citation-x.html>
- Lucic, P., & Teodorovic, D. (1999). Simulated annealing for the multi-objective aircrew rostering problem. *Transportation Research Part A : Policy and Practice*, 33(1), 19-45.
- Magnani, A., & Boyd, S. P. (2009). Convex piecewise-linear fitting. *Optimization and Engineering*, 1-17.
- Mattingly, J. D. (1996). *Elements of Gas Turbine Propulsion*: McGraw-Hill series in mechanical engineering.
- Mattingly, J. D., Heiser, W. H., & Daley, D. H. (1987). *Aircraft engine design*. Washington: American Institute Of Aeronautics and Astronautics.

- Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., & Teller, E. (1953). Equation of State Calculations by Fast Computing Machines. *Journal of Chemical Physics*, 21(6), 1087-1092.
- More, J. J. (1977). *Levenberg--Marquardt algorithm: implementation and theory*. Communication présentée à Conference on numerical analysis, Dundee, UK.
- Morieux, D. (réalisateur). (2014). les operations moteur. Tiré de http://home.nordnet.fr/dmorieux/op_moteur0001.htm
- Mosbah, A. B., Botez, R. M., Dao, T. M., Guezguez, M. S., & Zaag, M. (2016). A neural network controller new methodology for the ATR-42 morphing wing actuation. *INCAS Bulletin*, 8(2), 59.
- Moumen, N. (2011). *recherche tabou*. Algérie: Université de la science et de la technologie Mohamed Boudlaf.
- Murrieta, A., & Botez, R. M. (2015). New Methodology for an Aircraft Vertical Navigation Flight Trajectory Cost calculation using a Performance Database. *AIAA Journal of Aerospace Information Systems*, 12(8), 519-532.
- Murrieta, A., Gagné, J., & Botez, R. M. (2016). New Search Space Reduction Algorithm for Vertical Reference Trajectory Optimization. *INCAS Bulletin*, 8(2), 77-95.
- Osman, I. H., & Gilbert, L. (1996). Metaheuristics : A bibliography. *Annals of Operations research*, 63(5), 511-623.
- Paris, A. (2013). *Aérodynamique Mécanique du vol Performances avions*.
- Parizeau, M. (2006). Tiré de <http://wcours.gel.ulaval.ca/2014/h/GIF7005/default/7references/reseauxdneurones.pdf>
- Pollender, O., & Botez, R. M. (2011). *Practical sequencing method between aerodynamic modeling using the Vortex Lattice Method and a simulation platform for an autopilot using optimal control techniques*. Communication présentée à 58th Aeronautics Conference and AGM, Montreal.
- Rodriguez, L. F., & Botez, R. M. (2012). *Civil turbofan engines thrust generic model*. Communication présentée à IEEE Industrial Electronics Society, Montréal (p. 5444-5450).
- Rondepierre, A., & Weiss, P. (2013). *Méthodes standards en optimisation non linéaire déterministe*. Toulouse: INSA Toulouse.
- Roux, E. (2002). *Pour une approche analytique de la dynamique de vol* Toulouse:

- S.McCulloch, W., & H.Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity *Bulletin of Mathematical Biophysics*, 115-133.
- Sonny, M., Iain, W., & Declan, G. B. (2008). Development and Validation of a Civil Aircraft Engine Simulation Model for Advanced Controller Design. *Journal of Engineering for Gas Turbines and Power*, 130(5), 15.
- Stricker, M. (2004). *Réseaux de neurones pour le traitement automatique du langage : conception et réalisation de filtres d'informations*. paris: ESPCI ParisTECH.
- Torenbeek, E. (1982). *Synthesis of Subsonic Airplane Design*. Delft: Springer Science et Business Media
- Volpi, K. (2016). *Réseau de neurones*.
- Wanner, J.-C. (1976). *Dynamique du vol et pilotage des avions*: Office national d'études et de recherches aérospatiales.
- Weng, J.-L., Kabran, C., Billy, T., & Guillain, M. (réalisateurs). Les moteurs d'avion. Tiré de <http://lesmoteursdavion.lescigales.org/>
- Zaag, M., & Botez, R. M. (2017). Cessna Citation X Engine Model Identification and Validation in the Cruise Regime from Flight Tests based on Neural Networks combined with Extended Great Deluge Algorithm. In *AIAA Modeling and Simulation Technologies Conference* (p. 1941).
- Zhiping, Q., & Yuxing, Z. (2010). Parametric Optimization Design of Aircraft Based on Hybrid Parallel Multi-objective Tabu Search Algorithm. *Chinese Journal of Aeronautics*, 23(4), 430-437.