

MODÉLISATION DES PERFORMANCES DU MOTEUR
GENERAL ELECTRIC CF34-8C5B1 DE L'AVION CRJ700
À PARTIR DES RÉSEAUX DE NEURONES ET DES
TESTS EN VOL

par

Mahefatiana Rojo Princy ANDRIANANTARA

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

MONTREAL, LE 13 JUIN 2021

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 média 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 :

Prof. Ruxandra Botez, directrice de mémoire
Département de génie des systèmes à l'École de technologie supérieure

Prof. David St-Onge, président du jury
Département de génie mécanique à l'École de technologie supérieure

Prof. Georges Ghazi, membre du jury
Département de génie des systèmes à l'École de technologie supérieure

Prof. David Rancourt, examinateur externe
Département de génie mécanique à l'Université de Sherbrooke

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

LE 4 AOÛT 2021

À L'ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

REMERCIEMENTS

Ce mémoire de maîtrise a été réalisé à l'École de technologie supérieure, au Laboratoire de Recherche en Commande Active, Avionique et AéroServoélasticité (LARCASE).

Je remercie avant tout ma directrice de recherche, la Professeure Ruxandra Botez pour l'opportunité qu'elle m'a accordé de faire partie de son équipe de recherche au laboratoire du LARCASE, ainsi que pour son soutien et son aide tout au long de ce mémoire et de mes deux années de maîtrise.

Je remercie le Professeur Jeremy Laliberté de l'Université de Carlton pour le financement d'une partie de ce mémoire dans le cadre du programme CREATE Uninhabited aircraft system Training, Innovation and Leadership Initiative financé par le CRSNG. Je remercie également M. Ian Glenn de ING Robotics Aviation de m'avoir donné l'opportunité d'effectuer un stage dans sa compagnie. Je remercie le CRSNG pour l'octroi de la Chaire de Recherche du Canada en Technologie de Modélisation et Simulation des Aéronefs, dont les fonds ont financé une partie de ce mémoire. Je remercie aussi les organismes subventionnaires FCI (La Fondation Canadienne d'Innovation) et MDEIE (Ministère du Développement Économique, de l'Innovation et de l'Exportation) pour l'obtention des deux simulateurs de vol de niveau D au LARCASE pour l'avion d'affaire Cessna Citation X et le CRJ700, et qui ont été utiles à la réalisation de cette recherche.

Je remercie les membres du jury d'avoir pris le temps d'évaluer ce mémoire.

Je remercie le Dr Georges Ghazi pour la patience dont il fait preuve et l'aide qu'il a fournie. Ses connaissances et son savoir-faire sont salutaires, et travailler avec lui est un honneur. Je remercie également tous les membres du laboratoire du LARCASE pour les idées qu'on a pu échanger et leur soutien sous quelques formes que ce soit. Je tiens aussi à mentionner que la plus grande partie de ce mémoire a été réalisée durant le temps de pandémie de la COVID-19, ce qui m'a permis de découvrir les joies du télétravail et des cours en ligne.

Je remercie ma maman Emma pour l'amour inconditionnel qu'elle me porte chaque jour. Je remercie mon père, mon frère, ma sœur, toute ma famille et tous mes amis qui m'ont toujours soutenu dans mes choix et mes projets, et qui sont aujourd'hui littéralement dans les quatre coins du globe. Je laisse également une pensée à celle qui partage mon cœur depuis le secondaire et avec qui mes discussions sont infinies.

Je remercie mes colocataires Parham et Abdurrashid pour leurs encouragements quotidiens, et avec qui j'ai vécu tout au long de mes deux années de maîtrise. L'humour fut à son paroxysme et en anglais.

Enfin, par ce mémoire je rends hommage à un Grand Homme, M. Randimbisoa Rakotoarinoro, pour avoir cru en moi, et pour avoir contribué à réaliser mes rêves depuis mes années en Russie jusqu'au Canada. Comme promis, j'ai fini ce mémoire, et j'espère que de là où tu es, tu en es fier. Je te dédie ce travail.

MODÉLISATION DES PERFORMANCES DU MOTEUR GENERAL ELECTRIC CF34-8C5B1 DE L'AVION CRJ700 À PARTIR DES RÉSEAUX DE NEURONES ARTIFICIELS ET DES TESTS EN VOL

Mahefatiana Rojo Princy ANDRIANANTARA

RÉSUMÉ

Une nouvelle méthodologie a été développée afin d'identifier le modèle statique de performances de la turbosoufflante General Electric CF34-8C5B1 en utilisant deux réseaux de neurones en cascade de types perceptron simple couche avec une propagation directe (en anglais *FeedForward Neural Network*) et à l'aide des tests en vol. Les tests ont été réalisés sur le simulateur de vol du CRJ700 certifié niveau D pour toute l'enveloppe de vol de l'avion et pour quatre phases de vol, dont le décollage, la montée, la croisière et la descente.

La méthodologie a été réalisée en deux étapes. La première étape a consisté en l'estimation des paramètres de design du moteur, dont le rapport de pression du moteur (*Engine Pressure Ratio* ou *EPR*), le rapport de pression de la soufflante (*Fan Pressure Ratio* ou *FPR*) et le rapport de pression totale (*Overall Pressure Ratio* ou *OPR*), la température d'entrée de la turbine *ITT*, la vitesse de rotation de la soufflante N_1 et la vitesse de rotation de la turbine N_2 à partir d'un premier réseau de neurones ECS-NET. L'objectif de cette étape tend à reproduire le système de contrôle du moteur en modélisant les niveaux de poussée ainsi que les paramètres de chaque composant du moteur. La deuxième étape a consisté en l'estimation des paramètres de performances dont le débit de carburant W_F et la poussée F_N à partir d'un deuxième réseau de neurones ENG-NET. Les entrées du modèle sont notamment l'altitude, le nombre de Mach et les conditions atmosphériques comme la température ambiante, la pression ambiante et la densité.

L'identification du modèle a été réalisée avec succès sur le logiciel d'ingénierie Matlab/Simulink et elle a donné d'excellents résultats. Les réseaux de neurones ont été entraînés avec l'algorithme de régularisation bayésienne, qui a été capable d'identifier un système non linéaire à partir d'un nombre minimum de données. Différentes combinaisons de sorties du modèle ECS-NET ont été ensuite proposées, ce qui a démontré la capacité des réseaux de neurones à s'adapter par rapport aux paramètres disponibles.

La phase de validation a été réalisée à partir d'une analyse comparative entre les sorties de chaque modèle et celles de l'avion de référence. Pour ce faire, une analyse statistique a été réalisée pour l'ensemble des tests, puis différents scénarios de simulation ont été étudiés, dont un scénario pour la panne du moteur lors de la phase de croisière. Les résultats obtenus pour les paramètres de sortie ont montré que les modèles ont une excellente capacité de prédiction, avec une erreur relative inférieure à 2 % et une déviation standard inférieure à la tolérance de 5 % définie par la FAA.

GENERAL ELECTRIC CF34-8C5B1 TUROFAN PERFORMANCE MODELLING FOR THE CRJ700 AIRCRAFT USING ARTIFICIAL NEURAL NETWORKS AND FLIGHT TESTS

Mahefatiana Rojo Princy ANDRIANANTARA

ABSTRACT

A new methodology was developed and then validated in order to identify a static performance model of the General Electric CF34-8C5B1 Turbofan using two successive Feedforward Neural Networks and flight tests. The flight tests were carried on a Level D flight simulator of the CRJ700 for the aircraft flight envelope including major flight phases such as takeoff, climb, cruise and descent.

The identification process was divided in two steps. In the first step, the following engine design parameters were estimated : Engine Pressure Ratio EPR , Fan Pressure Ratio FRR , Overall Pressure Ratio OPR , Inlet Turbine Temperature ITT , Fan speed N_1 and Core speed N_2 using a first neural networks ECS-NET. This first step aimed to reproduce the Engine Control System by modelling the thrust ratings and each engine component characteristics. Then, the second step consisted in predicting the engine performance parameters, such as the fuel flow W_F and the thrust F_N using a second neural network ENG-NET. The model inputs were the altitude, the Mach number and the atmosphere conditions such as ambient pressure, ambient temperature and air density.

The identification process was successfully carried on Matlab/Simulink engineering software and gave excellent performances. The neural networks were trained using the Bayesian regularization algorithm, which was able to identify a non-linear system from a minimum amount of data. Several output combinations for the ECS-NET model were proposed, which demonstrated the ability of the neural networks to adapt according to the given set of parameters.

The validation phase was performed using a comparative analysis between the model, and the reference aircraft outputs. To do this phase, a statistical analysis was processed for the overall flight tests, and then several simulation scenarios were studied, including an engine degradation scenario during the cruise phase. The results have shown that the models have an excellent ability of prediction, with a relative error less than 2% and a standard deviation less than 5% for each output, as required by the FAA.

TABLE DES MATIÈRES

INTRODUCTION	21
CHAPITRE 1 REVUE DE LITTÉRATURE	29
1.1 L’historique du turboréacteur	29
1.2 Problème d’identification de systèmes	31
1.3 Principe d’identification de systèmes	33
1.4 Méthodes de modélisation	35
1.4.1 Modèle de « boîte blanche »	36
1.4.2 Modèle de « boîte noire »	37
1.4.3 Modèle de « boîte grise »	45
1.5 Algorithmes d’estimation.....	46
1.5.1 Maximum de vraisemblance (MLE).....	46
1.5.2 Optimisation d’une fonction de coût.....	49
1.5.3 Algorithme de Gauss-Newton.....	51
1.5.4 Algorithme de Levenberg-Marquardt	52
1.5.5 Algorithme de régularisation bayésienne.....	54
1.6 Présentation de la turbosoufflante.....	56
1.6.1 Numérotation des composants	58
1.6.2 Principes de conception d’un turboréacteur.....	59
1.7 Modèle de performance du turboréacteur à double flux.....	60
1.7.1 Principe de la propulsion	60
1.7.2 Modèle de l’atmosphère standard	61
1.7.3 Cycle de Brayton.....	64
1.7.4 Modèle de performance à partir de l’Approche par Composant.....	67
1.7.5 Modèle de performance simplifié	73
1.7.6 Modes de poussées.....	75
1.8 Modèles empiriques de performance	76
1.9 Paramètres de performances additionnels.....	77
1.10 Ordinateur de contrôle du moteur FADEC.....	79
1.11 Modélisation sur Matlab/Simulink	81
1.12 Modélisation du moteur par les logiciels spécialisés.....	82
1.13 Modélisation du moteur par analyse en CFD	83
1.14 Conclusion de la revue de littérature	85
CHAPITRE 2 MÉTHODOLOGIE ET IDENTIFICATION DU MODÈLE DE PERFORMANCE	87
2.1 Présentation de l’outil de recherche.....	87
2.1.1 Description du CRJ700.....	87
2.1.2 Description du turboréacteur CF34-8C5B1	90
2.1.3 Présentation du simulateur de vol	93
2.2 Réalisation des tests en vol	96

2.2.1	Phase de décollage	97
2.2.2	Phase de montée.....	102
2.2.3	Phase de croisière.....	104
2.2.4	Phase de descente.....	107
2.2.5	Base de données complète du modèle de performance	110
2.3	Identification du modèle de performance	118
2.3.1	Réseau de neurones de type perceptron multicouche	118
2.3.2	Structure du modèle de moteur.	121
2.3.3	Apprentissage du réseau de neurones.	123
2.3.4	Résultats de l'identification du réseau de neurones.....	125
2.3.5	Résultats préliminaires.....	125
CHAPITRE 3 VALIDATION DU MODÈLE DE RÉSEAUX DE NEURONES		131
3.1	Combinaisons du modèle de moteur ECS-ENG	131
3.2	Validation des modèles de performances	133
3.2.1	Critères d'analyse statistique	133
3.2.2	Résultats de la validation	134
3.3	Critères de sélection du modèle	141
3.4	Simulation du modèle complet pour différentes applications.....	145
3.4.1	Résultats de la validation du modèle complet.....	145
3.4.2	Simulation en phase de décollage.....	146
3.4.3	Simulation en phase de montée.....	149
3.4.4	Simulation du modèle en phase de descente.....	152
3.4.5	Simulation en phase de croisière lors d'une panne.....	155
3.4.6	Changement de modes de poussée.....	157
CHAPITRE 4 DISCUSSIONS, REMARQUES ET OBSERVATIONS		165
4.1	Méthodologie	165
4.2	Réalisation des tests en vol	166
4.3	Identification du modèle	167
4.4	Validation des modèles de performances	167
4.5	Observations et remarques.....	168
CONCLUSION		171
ANNEXE I AIAA Propulsion & Energy Forum 2021		173
BIBLIOGRAPHIE		189

LISTE DES TABLEAUX

Page

Tableau 2.1 Limites maximales des paramètres pour chaque mode de poussée	92
Tableau 2.2 Positions de la manette des gaz en fonction du mode de poussée	93
Tableau 2.3 Nombre des tests en vol pour chaque régime de vol.....	110
Tableau 2.4 Résultats de l'entraînement des réseaux de neurones	125
Tableau 3.1 Numérotation des sorties du bloc "Modèle de moteur"	132
Tableau 3.2 Dénomination des modèles de réseaux de neurones	132
Tableau 3.3 Erreurs moyennes, déviations standards et erreurs maximales du ratio de pression du moteur <i>EPR</i> en [%] pour les régimes de décollage, de montée, de croisière et de descente	135
Tableau 3.4 Erreurs moyennes, déviations standards et erreurs maximales du ratio de pression de la soufflante <i>FPR</i> en [%] pour les régimes de décollage, de montée, de croisière et de descente.....	136
Tableau 3.5 Erreurs moyennes, déviations standards et erreurs maximales de la température d'entrée de la turbine <i>ITT</i> en [%] pour les régimes de décollage, de montée, de croisière et de descente	136
Tableau 3.6 Erreurs moyennes, déviations standards et erreurs maximales du ratio de pression totale <i>OPR</i> en [%] pour les régimes de décollage, de montée, de croisière, et de descente	137
Tableau 3.7 Erreurs moyennes, déviations standards et erreurs maximales de la vitesse de rotation de la soufflante <i>N1</i> en [%] pour les régimes de décollage, de montée, de croisière et de descente.....	138
Tableau 3.8 Erreurs moyennes, déviations standards et erreurs maximales de la vitesse de rotation de la turbine <i>N2</i> en [%] pour les régimes de décollage, de montée, de croisière et de descente.....	138
Tableau 3.9 Erreurs moyennes, déviations standards et erreurs maximales de la poussée <i>FN</i> en [%] pour les régimes de décollage, de montée de croisière et de descente	139
Tableau 3.10 Erreurs moyennes, déviations standards et erreurs maximales du débit de carburant <i>WF</i> en [%] pour les régimes de décollage, de montée de croisière et de descente.....	140

Tableau 3.11 Critères de sélection <i>RSS</i> , et critères d'information <i>AIC</i> et <i>BIC</i> de chaque modèle de réseau de neurones	144
Tableau 3.12 Erreurs moyennes, déviations standards et erreurs maximales en [%] des sorties du modèle complet E123456 pour les régimes de décollage, de montée, de croisière et de descente.....	146

LISTE DES FIGURES

Figure 1.1 Principe de la modélisation mathématique d'un système.....	31
Figure 1.2 Algorithme d'identification d'un système.....	34
Figure 1.3 Schéma d'un réseau de neurones biologiques	38
Figure 1.4 Représentation d'un neurone artificiel	39
Figure 1.5 Fonction de transfert linéaire.....	40
Figure 1.6 Fonction de transfert hyperbolique tangente-sigmoïde	41
Figure 1.7 Réseau de neurones à plusieurs entrées et plusieurs sorties (MIMO).....	42
Figure 1.8 Architecture d'un réseau de neurones récurrent	43
Figure 1.9 Les principaux composants du turboréacteur à double flux	57
Figure 1.10 Numérotation des stations du turboréacteur à double flux.....	58
Figure 1.11 Diagramme Température <i>versus</i> Entropie $T - s$	65
Figure 1.12 Diagramme Pression <i>versus</i> Volume $P - V$	66
Figure 2.1 Surfaces de contrôle du CRJ700.....	88
Figure 2.2 Schéma de la turbosoufflante General Electric CF34-8C5B1.....	90
Figure 2.3 Modes de poussée de la manette des gaz.....	92
Figure 2.4 Photo du simulateur de vol du CRJ700	94
Figure 2.5 Profil d'un vol complet.....	96
Figure 2.6 Schéma des phases de décollage et de montée.....	97
Figure 2.7 Forces agissant sur l'avion lors du roulage au sol.....	98
Figure 2.8 Forces agissant sur l'avion lors de la montée.....	99
Figure 2.9 Ensemble des profils de décollage	101
Figure 2.10 Ensemble des profils de montée.....	104
Figure 2.11 Forces agissant sur l'avion lors de la croisière.....	104

Figure 2.12 Variation du TLA en fonction du mode de poussée.....	105
Figure 2.13 Ensemble des tests de croisière	107
Figure 2.14 Forces agissant sur l'avion lors de la descente.....	107
Figure 2.15 Ensemble des profils de descente	109
Figure 2.16 Ratio de pression du moteur <i>EPR</i> en fonction de l'altitude et du nombre de Mach pour les régimes de décollage, de montée, de croisière et de descente	112
Figure 2.17 Ratio de pression de la soufflante <i>FPR</i> en fonction de l'altitude et du nombre de Mach pour les régimes de décollage, de montée, de croisière et de descente	113
Figure 2.18 Température d'entrée de la turbine <i>ITT</i> en °K en fonction de l'altitude et du nombre de Mach pour les régimes de décollage, de montée, de croisière et de descente.....	114
Figure 2.19 Ratio de pression totale <i>OPR</i> en fonction de l'altitude et du nombre de Mach pour les régimes de décollage, de montée, de croisière et de descente	115
Figure 2.20 Vitesse de rotation de la soufflante corrigée <i>N1, C</i> en fonction de l'altitude et du nombre de Mach pour les régimes de décollage, de montée, de croisière et de descente	116
Figure 2.21 Vitesse de rotation de la turbine corrigée <i>N2, C</i> en fonction de l'altitude et du nombre de Mach pour les régimes de décollage, de montée, de croisière et de descente.....	117
Figure 2.22 Poussée corrigée <i>FN, C</i> et débit de carburant corrigée <i>WF, C</i> en fonction du nombre de Mach et de la vitesse de rotation de la soufflante corrigée <i>N1, C</i> pour les régimes de décollage, de montée, de croisière et de descente	118
Figure 2.23 Représentation du neurone artificiel.....	119
Figure 2.24 Représentation d'un réseau de neurones de type perceptron multicouche avec une propagation directe FFNN	120
Figure 2.25 Structure complète du modèle de performance	121
Figure 2.26 Identification du réseau de neurones sur Matlab.....	124
Figure 2.27 Variations de MSE pour chaque réseau de neurones (ECS-NET and ENG-NET) en fonction du nombre de neurones dans la couche cachée	124

Figure 2.28 Erreurs résiduelles du ratio de pressions du moteur <i>EPR</i> pour les régimes de (a) décollage, (b) montée, (c) croisière et (d) descente.....	126
Figure 2.29 Erreurs résiduelles du ratio de pressions de la soufflante <i>FPR</i> pour les régimes de (a) décollage, (b) montée, (c) croisière et (d) descente	126
Figure 2.30 Erreurs résiduelles de la températures d'entrée de la turbine <i>ITT</i> pour les régimes de (a) décollage, (b) montée, (c) croisière et (d) descente	126
Figure 2.31 Erreurs résiduelles du ratio de pressions total <i>OPR</i> pour les régimes de (a) décollage, (b) montée, (c) croisière et (d) descente	127
Figure 2.32 Erreurs résiduelles de la vitesse de rotation de la soufflante corrigée <i>N1, C</i> pour les régimes de (a) décollage, (b) montée, (c) croisière et (d) descente	127
Figure 2.33 Erreurs résiduelles de la vitesse de rotation de la turbine corrigée <i>N2, C</i> pour les régimes de (a) décollage, (b) montée, (c) croisière et (d) descente.....	127
Figure 2.34 Erreurs résiduelles de la poussée corrigée <i>FN, C</i> pour les régimes de (a) décollage, (b) montée, (c) croisière et (d) descente	128
Figure 2.35 Erreurs résiduelles du débit de carburant corrigé <i>WF, C</i> pour les régimes de (a) décollage, (b) montée, (c) croisière et (d) descente.....	128
Figure 3.1 Conditions de vol lors de la simulation en phase de décollage	147
Figure 3.2 Sorties des modèles <i>EPR, FPR, ITT</i> et <i>OPR</i> lors de la simulation en phase de décollage	148
Figure 3.3 Sorties des modèles <i>N1, N2, FN</i> et <i>WF</i> lors de la simulation en phase de décollage	149
Figure 3.4 Conditions de vol lors de la simulation en phase de montée.....	150
Figure 3.5 Sorties des modèles <i>EPR, FPR, ITT</i> et <i>OPR</i> lors de la simulation en phase de montée	151
Figure 3.6 Sorties des modèles <i>N1, N2, FN</i> et <i>WF</i> lors de la simulation en phase de montée	152
Figure 3.7 Conditions de vol lors de la simulation en phase de descente.....	153
Figure 3.8 Sorties des modèles <i>EPR, FPR, ITT</i> et <i>OPR</i> lors de la simulation en phase de descente	154
Figure 3.9 Sorties des modèles <i>N1, N2, FN</i> et <i>WF</i> lors de la simulation en phase de descente	155

Figure 3.10 Résultats de la simulation du modèle complet lors de la phase de croisière et dans une situation de panne	157
Figure 3.11 Conditions de vol lors de la variation de la manette des	158
Figure 3.12 Sorties des modèles <i>EPR</i> , <i>FPR</i> , <i>ITT</i> et <i>OPR</i> lors de la	159
Figure 3.13 Sorties des modèles <i>N1</i> , <i>N2</i> , <i>FN</i> et <i>WF</i> lors de la.....	160
Figure 3.14 Conditions de vol lors de la variation de la manette des gaz de 30° à 0°	161
Figure 3.15 Sorties des modèles <i>EPR</i> , <i>FPR</i> , <i>ITT</i> , <i>OPR</i> lors de la variation de <i>TLA</i> de 30 à 0°	162
Figure 3.16 Sorties des modèles <i>N1</i> , <i>N2</i> , <i>FN</i> et <i>WF</i> lors de la variation de <i>TLA</i> de 30 à 0°	163

LISTE DES ABRÉVIATIONS, SIGLES ET ACRONYMES

AIAA	American Institute of Aeronautics and Astronautics
AIC	Akaike Information Criterion
ANN	Artificial Neural Networks
BIC	Bayesian Information Criterion
BPR	Bypass Ratio
CAS	Calibrated Airspeed
CASI	Canadian Aeronautics and Space Institute
CLM	Component Level Modeling
CRJ	Canadair Regional Jet
EASA	European Union Aviation Safety Agency
ECS	Engine Control System
ECS-NET	Engine Control System Networks
ENG-NET	Engine Performance Networks
EPR	Engine Pressure Ratio
FAA	Federal Aviation Administration
FADEC	Full Authority Digital Engine Controller
FFNN	Feedforward Neural Networks
FMS	Flight Management System
FOM	Flight Operation Manual
FPR	Fan Pressure Ratio
HPC	High Pressure Compressor
HPT	High Pressure Turbine

IA	Intelligence Artificielle
IDLE	Idle Thrust setting
ISA	International Standard Atmosphere
ITT	Inlet Turbine Temperature
LARCASE	Laboratoire de Recherche en Commande active, Avionique et AeroServoÉlasticité
LPC	Low Pressure Compressor
LPT	Low Pressure Turbine
MCLB	Maximum Climb Thrust setting
MIMO	Multi-Input Multi-Output
MTO	Maximum Take-Off Thrust setting
MSE	Mean Square Error
OPR	Overall Pressure Ratio
RSS	Root Sum Square
RU	Ratio Unit
SFC	Specific Fuel Consumption
SISO	Single Input Single Output
TAS	True Airspeed
TC	Transport Canada
TLA	Throttle Lever Angle
VRESIM	Virtual Research Simulator

INTRODUCTION

Le modèle mathématique d'un moteur permet de simuler son comportement dans le but d'évaluer ses performances, de diagnostiquer les erreurs, ou de réduire le coût des tests en vol. De nos jours, la méthode la plus utilisée dans l'industrie est celle de l'approche par composant (en anglais *Component Level Modeling* ou CLM). Cette méthode consiste à modéliser chaque composant du moteur à partir des équations de thermodynamique (Chapman et al., 2016). Malheureusement, les données sur les composants sont souvent confidentielles et difficilement accessibles au grand public. C'est pour cette raison qu'il est préférable d'utiliser des modèles génériques dans les cas de simulations et d'appliquer des méthodes d'approximation. Cependant, cette démarche complique parfois la modélisation et réduit la précision du modèle.

Une autre alternative consiste à l'approche par métamodèle. Différentes techniques de l'Intelligence Artificielle (IA) sont proposées dans la littérature dans les cas de modélisation. Dans l'industrie des turbomachines, l'application de l'IA a pris place au début du 21^{ème} siècle. Mais son application actuelle dans l'aviation civile est encore limitée. Les modèles sont surtout utilisés pour le diagnostic, l'estimation des performances, la détection et la localisation des défaillances ou encore l'évaluation du niveau de dégradation du moteur. Parmi ces techniques de l'IA on peut citer les Machines à vecteur de support (en anglais *Support Vector Machine* ou SVM) (Zhao et al., 2014), les réseaux de neurones artificielles (en anglais *Artificial Neural Networks* ou ANN) (Lazzaretto et Toffolo, 2001) ou encore la logique floue (Ruichao et al., 2018).

Un exemple de métamodèle sont les réseaux de neurones. Ceux-ci sont capables de reproduire le comportement complexe des systèmes non linéaires avec un minimum de calcul et avec une plus grande rapidité que les méthodes analytiques comme l'approche par composant (Lazzaretto et Toffolo, 2001). Ils permettent entre autres de compenser le manque d'informations sur les composants du moteur. Quelques exemples d'application décrits dans la revue de littérature ont permis de susciter l'intérêt des réseaux de neurones dans la modélisation des turbines à gaz

La certification des systèmes basés sur l'Intelligence Artificielle (IA) reste encore un défi dans l'industrie aérospatiale, et surtout dans l'aviation civile. Les autorités de certification comme le *Federal Aviation Administration* (FAA), *Transport Canada* (TC) (Wilkinson et al., 2016) et le *European Union Aviation Safety Agency* (EASA) (Cluzeau et al., 2020) ont émis des recommandations quant aux procédures de certification de tels systèmes. La plus grande difficulté est de certifier la base de données nécessaire à l'identification du modèle, c'est-à-dire, de déterminer si tous les états du système sont proprement représentés par l'ensemble de données.

Ainsi, les projets de recherches sur l'IA se multiplient dans le but de valider leurs applications dans le domaine aérospatial, et aussi de faciliter leurs implémentations dans l'industrie. L'IA est déjà largement utilisée en pratique dans le domaine des « aéronefs sans pilotes » (en anglais *Unmanned Aerial Vehicle* ou UAV), aussi connu par « drones », étant donné que son application comporte moins de risques pour la vie humaine en cas de défaillance.

0.1 Intérêts de recherche et retombées

Afin de calculer la trajectoire optimale d'un avion d'un point à un autre, les algorithmes encodés dans le système de gestion de vol (en anglais *Flight Management System* ou FMS) utilisent des modèles mathématiques très précis. De tels modèles sont supposés de représenter les performances aérodynamiques de l'avion en termes de force de portance et traînée, ainsi que les performances propulsives des moteurs en termes de poussée et débit de carburant (Ghazi et Botez, 2019). Le FMS calcule la trajectoire optimale qui minimise le coût de vol. Étant donné que la trajectoire optimale dépend directement du modèle de performance, les avionneurs investissent largement dans le développement des modèles de prédiction fiables. Un modèle de poussée et de consommation de carburant permettrait ainsi d'avoir une meilleure prédiction de la consommation en carburant, et donc de s'assurer une meilleure optimisation.

Le développement d'une plateforme de simulation permettrait également de réduire le nombre d'essais en vol, qui sont parfois coûteux et demandent beaucoup de temps. Le nombre

d'aéronefs augmentant chaque année, on peut noter que la réduction du nombre d'essais en vol sera bénéfique sur les plans économique et environnemental. Un modèle mathématique adéquat permettrait de réduire le temps de calcul et de prédire les paramètres de performance avec un minimum de données.

Ce mémoire vise à mettre en pratique les connaissances en matière de modélisation mathématique en vue d'identifier un modèle de performance. Les connaissances dans les techniques de simulation et d'estimation de paramètres, ainsi que dans la modélisation du cycle d'un turboréacteur à double flux seront utilisées. Il nécessite notamment des connaissances en thermodynamique afin de comprendre la physique et le principe de fonctionnement du moteur. Ce mémoire traite également les thèmes de l'apprentissage machine (en anglais *Machine Learning*), grâce à l'application des réseaux de neurones artificiels. Plus précisément, il traite l'apprentissage supervisé (en anglais *Supervised Learning*), c'est à dire celui dans lequel le modèle est identifié à partir des données d'entrées et de sorties préalablement déterminées.

0.2 Objectifs de recherche

Comme il a été dit dans la section précédente, ce mémoire de recherche traite le thème de l'identification des systèmes. L'objectif principal de ce mémoire vise à investiguer la possibilité d'utiliser les réseaux de neurones artificiels pour modéliser les performances des moteurs General Electric CF34-8C5B1 de l'avion régional CRJ700. Pour ce faire, une nouvelle méthodologie sera proposée en utilisant deux réseaux de neurones en cascade et des tests en vol. Le modèle devra être capable de prédire les performances du moteur pour toutes les phases de vol et sur toute l'enveloppe de vol de l'avion.

Le premier objectif sera de découvrir les différentes techniques de simulation dans le domaine de la turbomachine et de réaliser une revue de littérature sur les méthodes de modélisation mathématiques. La théorie des turbines à gaz sera également revue afin de comprendre la physique du moteur à réaction et son mode de fonctionnement. À l'issue de la recherche bibliographique, la méthodologie appliquée dans ce mémoire sera également identifiée.

Le second objectif sera de se familiariser avec l'avion CRJ700, le moteur GE CF34-8C5B1 et l'outil de recherche qui est le simulateur de vol du CRJ700. En effet, afin de réaliser les tests sur le simulateur de vol, il est nécessaire de maîtriser les éléments de contrôle de l'avion, de reconnaître, d'interpréter et de manipuler l'instrumentation à bord de l'avion. Il est également nécessaire de se familiariser avec le langage de programmation CTS, qui est le langage utilisé par CAE inc. sur la plateforme de simulation.

Le troisième objectif sera d'identifier les paramètres nécessaires à l'identification du modèle, comme les pressions, les températures et les conditions de vol. Il sera nécessaire d'identifier les variables que l'utilisateur devra recueillir du simulateur de vol.

Le quatrième objectif sera de réaliser les tests en vol sur le simulateur de vol du CRJ700 afin d'établir une base de données détaillée. Cette étape occupera la majeure partie du temps alloué à ce mémoire. Les tests en vol devront comprendre les phases de décollage, de montée, de croisière et de descente et ils devront également couvrir toute l'enveloppe de vol de l'avion. La base de données sera ensuite divisée en deux parties, dont la base de données pour l'identification, et la base de données pour la validation du modèle.

Le cinquième objectif sera d'identifier le modèle de performance proprement dit. Pour ce faire, la modélisation des moteurs GE CF34-8C5B1 est réalisée en deux étapes. La première étape consiste à modéliser le système de contrôle du moteur grâce à un premier réseau de neurones ECS-NET. La deuxième étape consiste à modéliser les paramètres de performances, dont la poussée nette et le débit de carburant à partir d'un second réseau de neurones ENG-NET. Les deux réseaux de neurones seront entraînés sous Matlab, en utilisant l'algorithme de régularisation bayésienne. Différents modèles de ECS-NET seront également proposés et modélisés à partir des réseaux de neurones.

Le sixième objectif sera enfin de valider le modèle. Pour ce faire, une analyse statistique sera réalisée, ainsi que des exemples de simulation seront proposés. L'objectif consiste à comparer les sorties du simulateur avec celles du modèle pour l'ensemble des tests de validation.

0.3 Organisation du mémoire

Le CHAPITRE 1 présente une revue de littérature sur les techniques innovantes en matière de modélisation dans la turbomachine. Un bref historique des turbines à gaz a été établi et un rappel sur l'identification de système a été effectué dont les principes de « boîte blanche », « boîte noire » et « boîte grise ». La technique des réseaux de neurones a été exposée, et elle fait l'objet d'étude dans ce mémoire.

Le CHAPITRE 2 discute la méthodologie de recherche. Une présentation de l'outil de recherche est effectuée, et les résultats des tests en vols sont exposés. L'identification du modèle de performance est également réalisée grâce aux réseaux de neurones de type « perceptron ». Les résultats préliminaires de validation sont enfin discutés afin d'émettre une première conclusion sur la précision du modèle.

Le CHAPITRE 3 présente les résultats de l'identification et de validation ainsi que les simulations des modèles de performances. Différents modèles du réseau de neurones ECS-NET sont analysés, et une comparaison est effectuée entre les sorties du modèle et celles de l'avion de référence. La validation est effectuée pour chaque modèle, chaque phase de vol et chaque paramètre.

Le CHAPITRE 4 présente enfin une récapitulation du travail effectué, ainsi que des discussions sur les résultats et sur les futurs axes de recherches.

0.4 Contributions

Ce mémoire a fait l'objet de deux présentations à des conférences. La première présentation sera réalisée au forum de l'AIAA Propulsion & Energy 2021. Elle concerne la modélisation des performances à partir des réseaux de neurones avec comme sortie la vitesse de rotation de la soufflante N_1 , ainsi que les paramètres de performances dont le débit de carburant W_F et la poussée nette F_N . La deuxième présentation sera réalisée au symposium CASI Propulsion AÉRO 2021. Elle concerne la modélisation des performances à partir de l'approche par composant combinée aux réseaux de neurones avec comme sorties, les paramètres $EPR, FPR, OPR, ITT, N_1, N_2$ et les paramètres de performances, dont le débit de carburant W_F et la poussée nette F_N .

Ce mémoire présente aussi une analyse des différents modèles de moteur ECS-NET, ainsi qu'un modèle de performance sans la composante ECS-NET, toujours à partir des réseaux de neurones. Une méthode de sélection et une méthode de validation sont appliquées pour évaluer chaque modèle. Pour ce faire, une tolérance de 5 % définie par la FAA a été appliquée sur l'erreur relative pour les paramètres de performance W_F et F_N .

Ce mémoire constitue une continuité des travaux de recherche effectués au laboratoire du LARCASE sur la modélisation des performances du moteur. Les travaux de recherches précédents ont été réalisés pour l'avion d'affaire Cessna Citation X, d'abord par Zaag et Botez en 2017, qui ont appliqué les réseaux de neurones pour l'identification d'un modèle de performance pour la phase de croisière. Ses travaux ont ensuite été continués par Bardela et Botez en 2017, qui ont appliqué l'approche par composant ainsi que les méthodes empiriques pour la modélisation des performances en phase de croisière; et enfin, Ghazi et Botez en 2019, ont modélisés les performances à partir des méthodes empiriques et des données disponibles dans le manuel de vol du pilote (en anglais *Flight Operating Manual* ou FOM) dans le cadre de l'optimisation du système de gestion de vol (en anglais *Flight Managment System* ou FMS).

En plus de la phase de croisière, ce mémoire traite les phases de décollage, de montée et de descente pour l'avion régional CRJ700. La méthodologie proposée résulte d'une combinaison de l'approche par composant avec les réseaux de neurones. La structure du modèle est

identifiée dans la littérature et l'identification du modèle est réalisée sur le logiciel Matlab. Des simulations sont réalisées afin de valider les modèles dans des conditions de décollage, de montée, de descente, de détection de panne du moteur en régime de croisière et de changement de régime de poussée.

CHAPITRE 1

REVUE DE LITTÉRATURE

1.1 L'historique du turboréacteur

L'industrie aéronautique est un secteur en large expansion. Il recouvre plusieurs domaines d'activité et utilise plusieurs technologies de l'information (IT). Elle tente en particulier de mettre en pratique les nouvelles innovations. Ses acteurs regroupent à la fois les dynamiciens, les aérodynamiciens, les thermodynamiciens, le personnel navigant ou encore les contrôleurs aériens.

La conception d'un aéronef est le résultat d'une synergie importante entre différents acteurs et différentes compagnies. L'aéronef résulte d'une série d'études réparties entre plusieurs équipes dont une équipe de conception, une équipe de structure, une équipe de propulsion, une équipe d'avionique et systèmes de commande, ou encore une équipe pour la stabilité et le contrôle. Un effort prédominant d'une équipe sur une autre affecterait considérablement le type d'aéronef obtenu (Mattingly, 2002).

L'aéronautique a pris un essor considérable depuis la découverte du moteur à réaction, aussi appelé turboréacteur. Il a fait sa première apparition en 1910 au Deuxième Salon de l'Aéronautique à Paris, suite à sa découverte par Henri Coanda. Les missiles V-1 et V-2, développés par les Allemands entre 1935 et 1940 furent ensuite les premiers missiles à être équipés de turboréacteurs et ont largement prouvés leur efficacité durant la Seconde Guerre mondiale (Mattingly, 2002). En 1939, le Heinkel He-178 conçu par Von Ohain a été le premier avion propulsé par un turboréacteur.

Après la Seconde Guerre mondiale, plusieurs projets de développement de turboréacteurs ont vu le jour grâce à des compagnies industrielles comme Pratt & Whitney Canada, General Electric et Rolls-Royce. Certains avions militaires ont été peu à peu reconvertis en avions civils et en 1953, le *De Havilland Comet* a été le premier avion commercial muni de turboréacteurs.

La recherche en turbomachine n'a alors cessé d'évoluer et ensuite d'être appliquée à l'espace. Le principe de la propulsion a été découvert depuis le début du 20^{ème} siècle, notamment avec Tsiolkovski et les *Rocket equations* ou les équations d'une fusée (Austin et al., 2019).. Cependant, le voyage spatial a connu son essor seulement dans les années 1960. Il a mis en place des théories sur la propulsion et la conception de fusées, munies de moteurs à combustion, connu sous le nom de Ramjet. Ses recherches ont été reconnues et mises en pratique seulement à partir des années 1960 où la recherche spatiale est devenue importante surtout pour les États-Unis et de l'Union Soviétique.

D'autre part, la recherche sur les performances a pris place dans l'industrie aéronautique. D'abord dans le domaine militaire, les aviateurs cherchent à atteindre une vitesse maximale, et une meilleure manœuvrabilité des avions. On cherchera à optimiser les performances aérodynamiques et propulsives de l'avion, jusqu'à surpasser le mur du son. Le turboréacteur avec une postcombustion a été développé en vue d'atteindre ces performances. Ainsi, le 4 octobre 1957, l'avion CF-105 Arrow construit par la compagnie Avro Canada était le premier avion supersonique capable d'atteindre le nombre de Mach 2 (Force, n.d.). L'aviation civile connaîtra ensuite son moment de gloire avec le Concorde et le Tu-144 au début des années 1970. Malheureusement, le vol supersonique a démontré un déficit à la fois économique qu'environnemental. Par conséquent, les avions civils ont été développés surtout pour les vols transsoniques.

Pour ce faire, de nouvelles recherches sont menées en vue d'optimiser le vol commercial dans le domaine transsonique. Le turboréacteur à double flux fait son apparition et équipe la plupart des avions commerciaux actuels (Mattingly et al. 2002). La turbosoufflante (en anglais *Turbofan*) est un exemple de turboréacteur à double flux. Elle consiste à un turboréacteur à simple flux autour duquel traverse un second flux d'air froid, généré par une soufflante.

Les performances du turboréacteur à double flux dépendent essentiellement du taux de dilution du moteur (aussi désigné en anglais par *Bypass Pressure Ratio* ou *BPR*), exprimé par le rapport

de pression entre l'écoulement interne et l'écoulement externe. Le principe de fonctionnement ainsi que la numérotation de ses composants seront présentés dans la section 1.6.

De nos jours, il existe trois types de turbosoufflantes, dont la turbosoufflante avec un taux de dilution élevé supérieur à 5, la turbosoufflante avec un taux de dilution faible inférieur à 5, et la turbosoufflante munie d'un système de postcombustion (Mattingly et *al.* 2002). La turbosoufflante General Electric CF34-8C5B1 de l'avion Bombardier CRJ700 fera l'objet d'étude de ce mémoire de maîtrise qui appartient à la catégorie des turbosoufflantes avec un taux de dilution élevé.

1.2 Problème d'identification de systèmes

Un système est un ensemble d'éléments et d'équations. On peut citer les équations de forces et de moments, les lois de la thermodynamique, les lois de la physique quantique, etc.

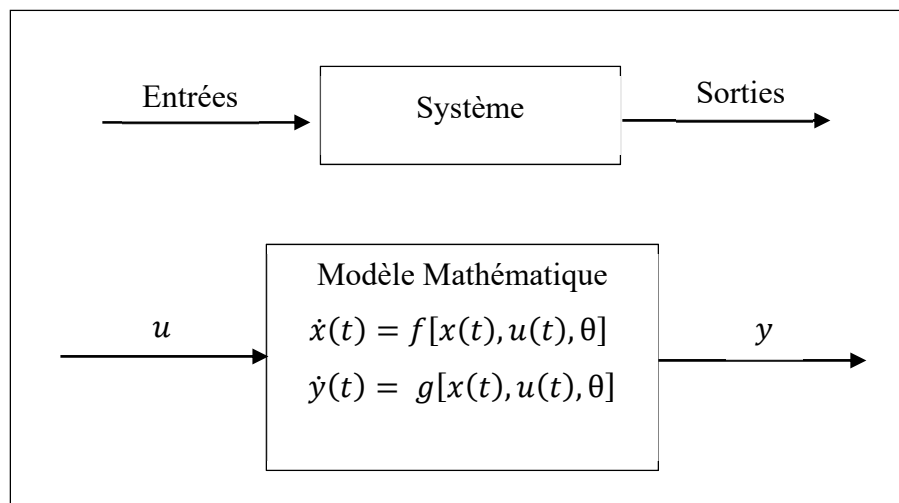


Figure 1.1 Principe de la modélisation mathématique d'un système

En général, un système est composé de 3 éléments, dont les entrées u , les sorties y , et les fonctions du système f et g comme montré sur la Figure 1.1. Le vecteur x représente l'état du système, tandis que θ représente le vecteur de paramètres des fonctions f et g .

Selon Jategaonkar, 2015, il existe trois types de problèmes possibles dans une de modélisation de systèmes :

- Le premier est le problème de simulation, qui consiste à l'identification d'une sortie y à partir d'une entrée x et d'un système de fonctions f et g .
- Le deuxième est le problème de contrôle, qui consiste à l'identification d'une entrée x à partir d'une sortie y et d'un système de fonctions f et g .
- La troisième est le problème d'identification de système, qui consiste à l'identification des fonctions f et g à partir d'une entrée x et d'une sortie y .

Dans le cadre de ce mémoire, on propose la résolution du troisième problème.

L'identification d'un système peut être une tâche complexe, car elle demande beaucoup de temps et de données. En effet, l'identification de systèmes est similaire à un problème inverse et répond au besoin de déterminer le comportement (ou la physique) d'un ensemble à partir d'observations. L'identification de système voit son application dans plusieurs domaines comme l'économie, l'ingénierie civile, l'ingénierie mécanique, la géologie, ou encore la sociologie, où les mesures expérimentales sont réalisées à partir d'observations en vue d'obtenir un modèle.

La validité d'un nouveau système identifié dépend en premier lieu de la validité des données d'entrées et de sorties utilisées lors de sa modélisation. Le système à identifier comporte des paramètres θ que l'on doit estimer indirectement à partir des mesures expérimentales.

L'identification d'un système n'est pas arbitraire, et dépend du savoir-faire de l'auteur. Cependant, une règle générale est appliquée, qui repose sur le principe de parcimonie établi par Ockham's Razor au 14ème siècle et repris par Jategaonkar, 2015. Cette règle stipule que le nombre de paramètres du modèle mathématique à estimer ne doit pas dépasser le nombre nécessaire pour expliquer le comportement (ou la physique) du système. C'est-à-dire que le modèle doit être le plus simple possible, avec un minimum de paramètres, ce qui permet de

minimiser les erreurs. En effet, les erreurs, les ajustements excessifs, et les redondances engendrent souvent des calculs non essentiels et ralentissent les simulations.

Le processus d'identification répond aux principes suivants :

- La dynamique du système est déterministe.
- La physique du système peut être modélisée par des lois mathématiques.
- L'expérimentation est réalisable.
- Les entrées et les sorties du système doivent être des quantités mesurables directement par des capteurs ou indirectement par des calculs.

1.3 Principe d'identification de systèmes

L'identification de système est un processus visant à déterminer un modèle mathématique exprimé par des équations linéaires ou non linéaires comportant des paramètres inconnus θ . Ceux-ci doivent être déterminés indirectement à partir des mesures expérimentales et des méthodes d'estimation. Les entrées et les sorties du système sont tout d'abord mesurées. Un modèle est ensuite construit par le modélisateur afin de lier les sorties avec les entrées. En réalité, l'identification parfaite est impossible, l'objectif principal étant de minimiser l'erreur entre les sorties mesurées t et les sorties du modèle mathématique y .

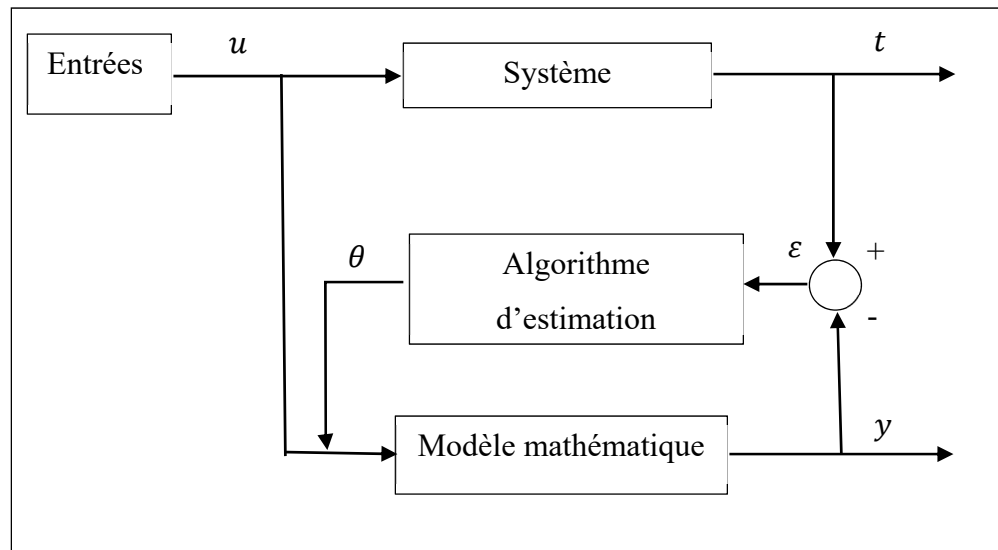


Figure 1.2 Algorithme d'identification d'un système

Pour ce faire, l'auteur fait face à plusieurs difficultés lors de l'identification de système, comme le choix des paramètres d'entrées et de sorties, l'hypothèse d'un ou de plusieurs modèles lors des premières étapes de l'identification, et le choix du modèle adéquat parmi ces modèles. Cela implique que le modèle mathématique identifié n'est pas unique, et représente une solution parmi plusieurs solutions.

L'identification de système est réalisée selon un processus générique. La majeure partie des efforts est allouée à l'identification et l'estimation des paramètres θ . Une fois cette étape achevée, il est nécessaire de procéder à une validation du modèle selon les critères choisis par l'auteur. Si ces critères ne sont pas respectés, le processus de modélisation doit reprendre son cycle avec de nouveaux paramètres θ , et/ou avec de nouvelles structures et de nouvelles données d'entrées u et de sorties t . Le processus est arrêté lorsque le modèle final répond à tous les critères de performance. Ainsi l'identification de système est un processus itératif, qui repose sur le principe d'essai-erreur. La Figure 1.2 illustre le principe d'identification d'un système. Le vecteur de paramètres θ est déterminé grâce à un algorithme d'estimation qui optimise une fonction de coût, dont ses contraintes sont définies par le modélisateur.

Lors de la première itération, le vecteur θ_0 doit être rigoureusement choisi par l'utilisateur, de sorte que l'estimation converge le plus rapidement possible et dans la bonne direction. Un mauvais premier choix pourrait ralentir le processus d'estimation et/ou faire diverger le processus d'identification et/ou arrêter le processus d'identification sur un extremum local. Ce premier choix repose ainsi sur l'intuition du modélisateur au début du processus itératif. Si la structure du système est préalablement connue, selon des connaissances *a priori* de l'auteur sur le système, alors l'identification reviendrait seulement à estimer les paramètres θ .

1.4 Méthodes de modélisation

Un modèle est l'ensemble des équations, relations mathématiques et paramètres choisis par l'auteur. En de simples termes, c'est la manière par laquelle l'auteur perçoit le système en général. De ce fait, un modèle n'est pas unique pour un système.

Deux types de modèles existent selon l'approche choisie par l'auteur (Jategaonkar, 2015, chap.4) : 1) le modèle phénoménologique et 2) le modèle comportemental.

Le modèle « phénoménologique » repose sur une connaissance de la physique du système, d'une formulation théorique du problème et dérive essentiellement de la mécanique newtonienne. Les paramètres à estimer dans ce cas ont leurs significations physiques. C'est ce que l'on nomme par modèle de « boîte blanche ».

Le modèle « comportemental » ne s'intéresse pas à la physique du système, ni même aux interactions internes au système. L'auteur veille seulement à établir des liens entre ses entrées et sorties. Dans ce cas, les paramètres du système ne sont que des valeurs numériques sans aucune interprétation physique et sont appelés « les poids » du système. L'auteur cherchera seulement de reproduire le signal à la sortie du système. C'est ce que l'on nomme par modèle de « boîte noire ».

Un troisième modèle de « boîte grise » est possible à réaliser. Comme l'expression l'indique, il s'agit d'une combinaison des deux modèles phénoménologique et comportemental. Les sections 1.4.1, 1.4.2 et 1.4.3 développeront ces concepts, ainsi que leurs applications dans un contexte d'identification d'un modèle de moteur et d'un modèle de performance.

1.4.1 Modèle de « boîte blanche »

Le concept de « boîte blanche » s'intéresse à l'identification d'un modèle à partir de la physique du système. L'auteur choisit de représenter son modèle à partir des équations fondamentales de la physique afin de déterminer un lien entre les entrées et les sorties du système. Une telle approche est nécessaire lors de ses premières phases d'identification. Il permet à l'auteur d'avoir une idée globale de son modèle et sa structure, de la nature du système et de ses paramètres à identifier.

Dans ce cas, l'auteur devrait avoir une connaissance précise de la physique du système. En effet, le modèle phénoménologique peut être très complexe, et l'erreur en sortie du modèle est souvent très grande. Cette approche mène principalement à une représentation sous forme d'équations différentielles, ou des équations avec des dérivées partielles. Il nécessite une connaissance de la structure du modèle, de chaque sous-système, et d'une description détaillée des interactions internes au système et de chacun de ses paramètres.

Le modèle de « boîte blanche » peut être qualifié de « paramétrique » et de « non paramétrique ». Le modèle d'état et la fonction de transfert appartiennent à la catégorie du modèle « paramétrique » et permettent d'organiser et de structurer le modèle. Étant donné la complexité que peut présenter cette approche, un modèle comportemental est proposé dans la section suivante, celui de « boîte noire », qui soulève les contraintes physiques et les connaissances *a priori* du modélisateur.

1.4.2 Modèle de « boîte noire »

Comme énoncé précédemment, le modèle de « boîte noire » ne s'intéresse ni à la physique du système ni aux interactions internes au système. Ce concept tend à créer des liens entre les entrées et les sorties, de sorte que la réponse en sortie du modèle concorde avec celle du système. Parmi les techniques de modélisation « boîte noire », on peut citer celles de l'intelligence artificielle, dont son application a pris un essor dans le domaine aérospatial. Un exemple d'application sont les réseaux de neurones et la logique floue.

1.4.2.1 Réseaux de neurones artificiels

Une description du principe de fonctionnement des réseaux de neurones est proposée dans cette sous-section ainsi que quelques exemples d'application dans le domaine de la turbomachine.

Les réseaux de neurones artificiels (en anglais *Artificial Neural Networks* ou NN) sont largement utilisés dans les problèmes d'identification d'un modèle mathématique. La première allusion aux réseaux de neurones remonte vers la fin du 19^{ème} siècle dans des travaux interdisciplinaires de physique, de psychologie et de neurophysiologie menés par des scientifiques comme Hermann von Helmholtz, Ernst Mach et Ivan Pavlov (Hagan et *al.*, 2002).

Les réseaux de neurones modernes voient le jour dans les années 1940 avec les travaux de Warren McCulloch et Walter Pitts (Cité dans Hagan et *al.*, 2002), qui ont démontré leur efficacité. La première application des réseaux de neurones est les « perceptrons », construits avec des règles d'apprentissage, créés par Frank Rosenblatt dans les années 1950. Plusieurs autres types de réseaux de neurones, plus performants et plus robustes seront ensuite obtenus.

Les réseaux de neurones appartiennent aux techniques de l'intelligence artificielle ou IA. Ils sont également connus sous le nom de « réseau de neurones artificiels », en anglais *Artificial Neural Networks* ou ANN. En principe, les réseaux de neurones artificiels tentent de résoudre un problème de la même manière que le cerveau humain, dans le cas de ce mémoire, des problèmes de régression linéaire et non linéaire. D'autres applications des ANN sont possibles,

surtout en imagerie pour la reconnaissance de formes, pour la reconstitution d'images et de signaux, et pour la classification d'objets. Il est également possible d'estimer des probabilités pour les problèmes de prédiction.

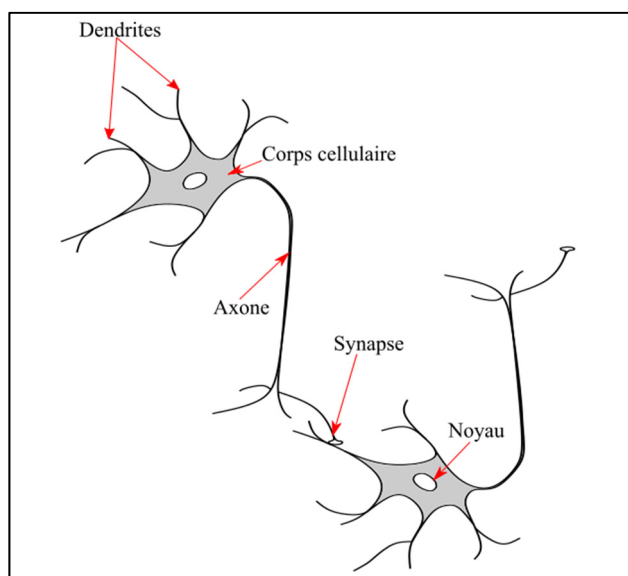


Figure 1.3 Schéma d'un réseau de neurones biologiques

La Figure 1.3 montre le schéma d'un réseau de neurones biologiques contenu dans le cerveau humain. Les signaux sont envoyés par le cerveau vers les différents organes et muscles du corps, permettant différents processus comme la respiration, le mouvement ou encore la pensée. Comme leurs noms l'indique, les réseaux de neurones sont composés par des neurones, qui sont connectés les uns avec les autres par des liens appelés « synapses » et chaque neurone a une fonction bien précise. Un neurone est capable de recevoir des signaux provenant d'autres neurones grâce aux « dendrites », qui sont ses entrées, de traiter les informations grâce au « noyau », et de créer un signal de sortie unique qu'il transmet au prochain neurone grâce aux « axones » et « synapse ». La fonction biologique neuronale est conservée dans la mémoire du neurone qui est le « noyau », ce qui signifie que le neurone traite des informations précises, d'où la nécessité de l'utilisation des règles d'apprentissage.

L'apprentissage est le processus de construction de liens entre les neurones ou la modification de certains liens déjà existants. Il existe plusieurs algorithmes d'apprentissage, dont les

exemples sont l'algorithme de Levenberg-Marquardt (section 1.5.4) et l'algorithme de régularisation bayésienne (section 1.5.5).

En général, la structure du neurone artificiel est la même que celle d'un neurone naturel comme présenté sur la Figure 1.4. Les neurones sont interconnectés entre eux pour former une architecture complexe. Un neurone peut avoir plusieurs entrées, mais il ne peut avoir qu'une seule sortie.

La Figure 1.4 illustre les composantes d'un neurone artificiel. Mathématiquement, le vecteur d'entrées u est multiplié par la matrice de poids w pour former le produit matriciel wu . Chaque produit $u_i w_{1,i}$ est ensuite additionné grâce à un sommateur. La seconde entrée du sommateur correspond au biais b . La sortie du sommateur n est ensuite connectée à la fonction de transfert f , qui calcule enfin la sortie du neurone a . La fonction de transfert f peut être une fonction linéaire ou non linéaire. Elle est choisie en fonction de la difficulté du problème que le neurone doit résoudre.

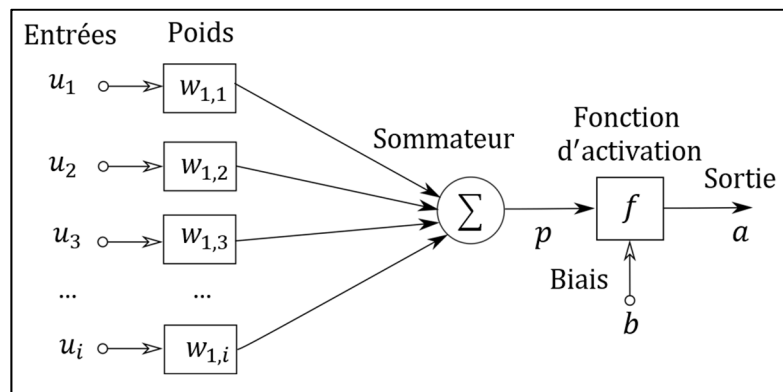


Figure 1.4 Représentation d'un neurone artificiel

Il existe plusieurs types de fonctions de transfert dont les plus utilisées sont les fonctions de transfert linéaires et les fonctions de transfert de type tangente-sigmoïde. La fonction de transfert tangente-sigmoïde permet de mieux généraliser l'identification, et elle est capable d'approximer des systèmes complexes avec un degré élevé de non-linéarité. Ces deux

fonctions de transfert seront appliquées dans ce mémoire pour l'identification d'un modèle de performance.

La fonction de transfert linéaire est utilisée pour modéliser la ou les sorties du réseau de neurones. Elle est décrite par la relation suivante :

$$a = f(p) = cp + d \quad (1.1)$$

où a est la valeur de la fonction, p est la variable, c est le coefficient directeur de la droite, et d est l'ordonnée à l'origine. En général, la fonction de transfert est strictement linéaire, c'est-à-dire que $d = 0$ et $c = 1$. La fonction de transfert se réécrit alors comme suit :

$$a = p \quad (1.2)$$

La fonction de transfert linéaire est modélisée graphiquement sur la Figure 1.5.

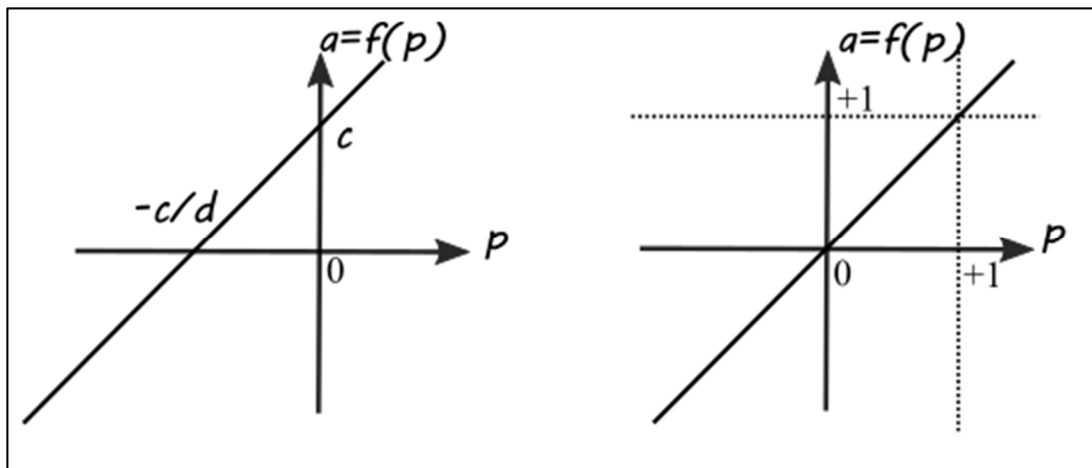


Figure 1.5 Fonction de transfert linéaire

La fonction de transfert hyperbolique tangente-sigmoïde est utilisée pour modéliser la ou les couches cachées du réseau de neurones. Elle est décrite par la relation suivante :

$$a = \frac{e^p - e^{-p}}{e^p + e^{-p}} \quad (1.3)$$

La fonction de transfert hyperbolique tangente sigmoïde est modélisée graphiquement sur la Figure 1.6.

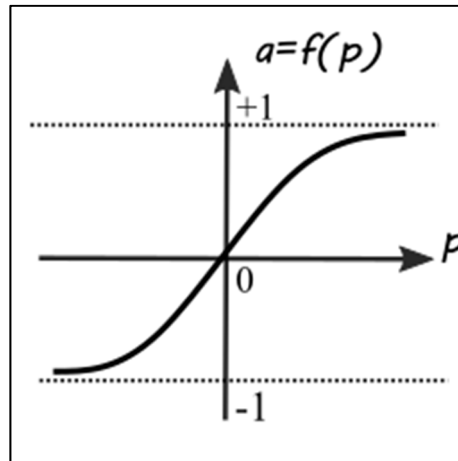


Figure 1.6 Fonction de transfert hyperbolique tangente-sigmoïde

Le logiciel de modélisation numérique Matlab/Simulink est muni d'une boîte d'outils pour la modélisation des réseaux de neurones (*Neural Network Toolbox*) permettant à l'auteur de concevoir et de personnaliser son propre réseau de neurones.

Les paramètres du réseau de neurones que l'auteur devra définir sont l'architecture, le nombre de couches cachées, le nombre de neurones, et les fonctions de transfert pour chaque couche. Ainsi, un réseau de neurones peut avoir plusieurs entrées et sorties (en anglais *Multi-Input Multi-Output* ou MIMO), comme le montre la Figure 1.7.

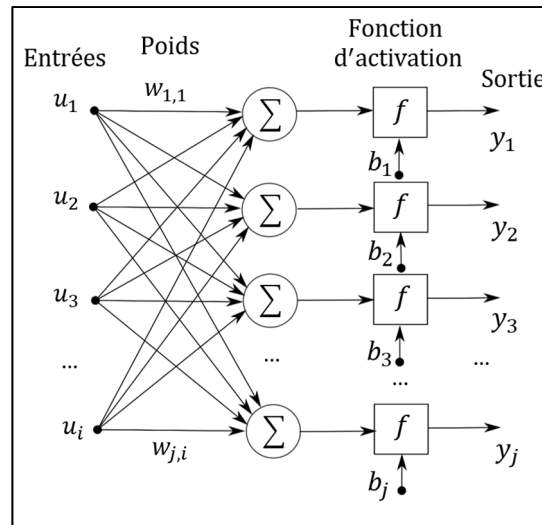


Figure 1.7 Réseau de neurones à plusieurs entrées et plusieurs sorties (MIMO)

Dépendamment de son architecture, un réseau de neurones peut être « direct » comme montré dans la Figure 1.7, ou « récurrent » avec un retour sur la sortie comme sur la Figure 1.8. Le choix de la structure du réseau de neurones dépend également de la dynamique du système que l'auteur doit modéliser. Par exemple, un réseau de neurones récurrent est plus approprié pour modéliser des systèmes dynamiques dont les variables dépendent du temps t , tandis que les réseaux de neurones de type « direct » sont plus appropriés pour modéliser des systèmes statiques dont leurs variables ne dépendent pas du temps.

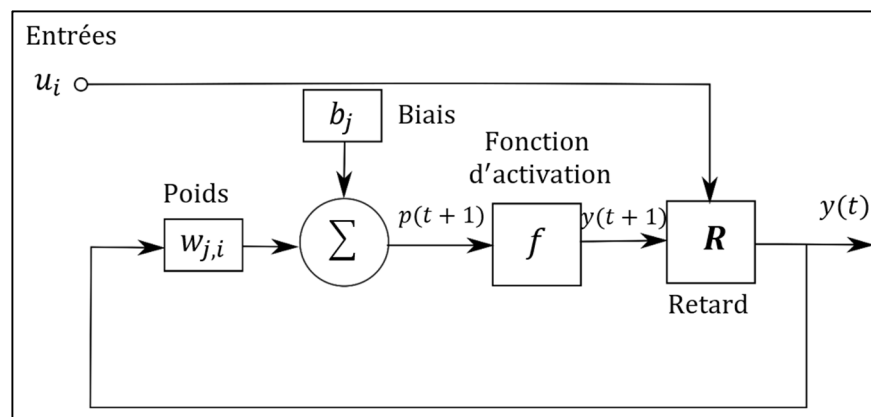


Figure 1.8 Architecture d'un réseau de neurones récurrent

La méthodologie de recherche développera en détail l'architecture du réseau de neurones choisie pour le modèle de moteur et le modèle de performances.

1.4.2.2 Exemples d'application des réseaux de neurones

Dans le domaine des turbomachines, les réseaux de neurones ont été initiés par (Kanelopoulos et *al.*, 1997) (Mathioudakis et Romessis, 2004) (Kobayashi et Simon, 2005), qui les utilisent pour le diagnostic et pour l'estimation de performance du moteur. De même (Lazaretto et Toffolo, 2001) proposent un modèle de moteur hybride en combinant les méthodes d'approche par composant (en anglais *Component Level Modeling* ou CLM) avec les réseaux de neurones artificiels ANN afin de compenser le manque de connaissances sur les composants du moteur.

Zaag et *al.*, 2019, au laboratoire LARCASE, ont appliqué les réseaux de neurones pour l'identification d'un modèle de performance de l'avion d'affaire Cessna Citation X, optimisé par l'algorithme du grand déluge étendu (en anglais *Extended Great Deluge* ou *EGD*). Ils ont identifié un modèle de poussée et un modèle de consommation de carburant pour la phase de croisière, ainsi qu'un modèle de soufflante N_1 . Les données d'entrées et de sorties ont été l'altitude, le nombre de Mach et la position de la manette des gaz. Dans ce cas, il n'a pas été nécessaire de déterminer les équations décrivant la physique de chaque composante du moteur. Le nombre optimal de neurones et de couches a été déterminé grâce à l'algorithme du grand déluge étendu (*EGD*), ce qui a permis d'accélérer l'identification des paramètres du réseau de neurones. Le modèle final a été comparé avec le modèle analytique de Torrenbeek disponible dans la littérature. Les simulations ont donné une erreur de moins de 3 % lors de la prédiction du débit de carburant et de la poussée pour la phase de croisière.

Ibrahim et *al.*, 2019, proposent une comparaison de modèles de performances à l'aide de différents types de réseaux de neurones. Ils mettent en application les réseaux de neurones statiques et dynamiques avec plusieurs entrées et plusieurs sorties de type MIMO NARX, et

avec différentes architectures en série et en dérivation. Ils varient la structure à savoir le nombre de couches, le nombre de neurones et ils appliquent différents algorithmes d'apprentissage. Les poids et biais ont été calculés en minimisant une fonction de coût, qui est exprimée par la racine de l'erreur moyenne quadratique (en anglais *Root Mean Square Error* ou RMSE).

Récemment, d'autres auteurs comme (Zhou et *al.*, 2020), (Zhao et *al.*, 2020) et (Gishvarov et Raherinjatovo, 2019) proposent des réseaux de neurones plus sophistiqués, capables de détecter les erreurs, de faire un diagnostic du moteur et de prendre des décisions intelligentes.

Les résultats ont montré que l'apprentissage du réseau de neurones dynamique prenait plus de temps de calcul que celui du réseau de neurones statique. Cependant, le réseau de neurones dynamique est capable de mieux simuler le comportement du système en régime transitoire.

1.4.2.3 Logique floue

La logique floue appartient à la catégorie d'algorithmes métaheuristiques, dont son application fait partie du concept de « boîte noire ».

Ruichao et *al.*, 2018 ont identifié un modèle de moteur à partir de la logique floue de Takagi Sugeno (TS). Il a mis en évidence la synthèse d'un contrôleur, basé sur la détermination des gains sur toute l'enveloppe de vol de l'avion, assurant ainsi la stabilité du moteur. L'objectif du modèle de moteur est de minimiser une fonction de coût lors de l'identification des paramètres, d'établir des contraintes de stabilité, et ensuite d'appliquer la méthode de TS. En effet, la méthode de TS permet de convertir un système initialement non linéaire en un système linéaire selon le principe de classement flou dont les contraintes sont établies par l'auteur. Il faudra ainsi résoudre un problème d'optimisation quadratique.

Boëly et *al.*, 2010, ainsi que Kouba et *al.*, 2010, ont appliqué la logique floue pour l'identification d'un modèle aérodynamique non linéaire de l'avion F/A-18 grâce à la méthode de Sugeno, et à son couplage avec les réseaux de neurones. Le taux de prédiction des

performances aérodynamiques de l'avion est supérieur à 98 % avec un temps de calcul minimal.

1.4.3 Modèle de « boîte grise »

Le modèle de « boîte grise » combine les caractéristiques des modèles de « boîte noire » avec ceux de la « boîte blanche ». L'objectif du modèle est de sauvegarder la physique du système, tout en introduisant de nouveaux paramètres, n'ayant aucune signification physique, mais permettant d'augmenter la précision de l'identification. Pour illustrer ce principe, des travaux sur l'identification de modèle de performance ont été effectués au laboratoire du LARCASE d'abord par (Ghazi et Botez, 2015) et ensuite par (Botez et *al.* 2019).

L'identification d'un modèle de performance pour l'avion d'affaire Cessna Citation X par Ghazi et Botez a été effectuée au LARCASE dans le cadre de l'optimisation du système de gestion de vol (en anglais *Flight Management System* ou FMS). La méthode repose sur l'identification d'un modèle à partir des données de performances disponibles dans le manuel de vol du pilote (en anglais *Flight Crew Operating Manual* ou FCOM). Les équations mathématiques pour le débit de carburant, la poussée et la vitesse de rotation de la soufflante ont été préalablement établies comme présenté à la section 1.7.5. Les données d'entrées et de sorties ont été obtenues grâce à l'interpolation des tables de performances disponibles dans le FOM. Les paramètres inconnus ont ensuite été estimés afin d'obtenir une erreur relative inférieure à 5 % lors des phases de validation.

Une deuxième approche a été proposée au LARCASE par (Bardela et Botez, 2017), (Botez et *al.*, 2019) pour l'avion d'affaire Cessna Citation X. La méthode d'« approche par composant » a été appliquée pour la phase de croisière. Cette méthode consiste à modéliser chaque composant du moteur grâce aux équations de thermodynamique à la section 1.7.4, qui décrivent le cycle de Brayton présenté à la section 1.7.3. Pour ce faire, la modélisation a été réalisée en deux étapes. La première étape consiste à identifier un modèle de soufflante dont la sortie correspond à la vitesse de rotation de la soufflante N_1 , tandis que la deuxième étape

consiste à identifier un modèle de la turbine et à modéliser la vitesse de rotation de la turbine N_2 . Les données d'entrées ont été l'altitude, le nombre de Mach, la position de la manette des gaz (en anglais *Throttle Lever Angle* ou TLA), le taux de dilution du moteur, la température ambiante, la pression ambiante, le débit d'air et le débit de carburant. Les données de sorties ont été la vitesse de rotation de la soufflante N_1 , la vitesse de rotation de la turbine N_2 , mais aussi les paramètres de performances, comme la consommation en carburant W_F et la poussée F_N . Une technique d'estimation de paramètres a ensuite été appliquée pour ajuster les équations mathématiques. La modélisation a réussi avec une erreur relative moyenne inférieure à 5 % sur toute l'enveloppe de vol de l'avion pour la phase de croisière.

Qui et *al.*, 2007, ont développé un modèle de diagnostic du moteur pour l'évaluation du niveau de détérioration du turboréacteur. Les paramètres d'entrées ont été obtenus grâce à une plateforme du moteur, qui a été modélisé à partir de l'« approche par composant » (en anglais *Component Level Modeling* ou CLM) présenté dans la section 1.7.4. La partie « diagnostic » quant à elle a été estimée grâce aux « réseaux de neurones ».

1.5 Algorithmes d'estimation

L'identification d'un système dépend des paramètres, qui sont calculés grâce à un algorithme d'estimation. Le choix d'un algorithme par rapport à un autre dépend de la décision du modélisateur et des critères de performance qu'il a établis. En général, ces critères sont la fonction de coût et le temps de calcul. Ce choix pourrait également dépendre de la capacité de calcul de l'ordinateur. Par exemple, la solution d'un algorithme complexe peut prendre plus de temps à converger.

1.5.1 Maximum de vraisemblance (MLE)

La méthode de maximum de vraisemblance (en anglais *Maximum Likelihood* ou ML) a été conçue par Fisher (cité dans Jategaonkar, 2015), en vue de déterminer les paramètres θ , et de maximiser une fonction de vraisemblance $p(t|\theta)$ défini par l'équation (1.4).

$$p(t|\theta) = p(t_1|\theta) \cdot p(t_2|\theta) \cdots p(t_N|\theta) = \prod_{i=1}^N p(t_i|\theta) \quad (1.4)$$

où $p(t|\theta)$ est la probabilité des observations t et θ est le vecteur de paramètres à estimer.

La théorie de Fisher dit qu'une population peut être décrite à l'aide d'une expression mathématique déterminée, dont ses valeurs numériques sont calculées par le vecteur θ . La population représente l'ensemble des observations t pour un nombre d'expérimentations $N \rightarrow \infty$. Il est alors nécessaire de déterminer les valeurs de θ sur un intervalle défini. Dans ce cas, la fonction de vraisemblance p représente la probabilité des variables observées t et non pas la probabilité des paramètres θ . La valeur maximale de la fonction de vraisemblance notée $\hat{\theta}_{ML}$ est estimée par l'équation (1.5) :

$$\hat{\theta}_{ML} = \arg \left(\max_{\theta} \ln p(t|\theta) \right) \quad (1.5)$$

En pratique nécessite la résolution de l'équation suivante :

$$\frac{\partial \ln p(t|\theta)}{\partial \theta} = 0 \quad (1.6)$$

où $\ln[p(t|\theta)]$ est le logarithme de la fonction de vraisemblance p . L'existence d'une solution de l'équation (1.6) suggère la continuité de la fonction de vraisemblance $p(t|\theta)$ et implique que la dérivée seconde de la fonction de vraisemblance existe également. Cependant, étant donné que l'équation (1.6) s'agit d'un système d'équations non linéaire, la solution ne peut être trouvée que par une série d'approximations. Le développement linéaire de la fonction de vraisemblance $p(t|\theta)$ lors de la première itération θ_0 de θ est donné par l'équation (1.7) :

$$\frac{\partial \ln p(t|\theta_1)}{\partial \theta} \approx \frac{\partial \ln p(t|\theta_0)}{\partial \theta} + \frac{\partial^2 \ln p(t|\theta_0)}{\partial \theta^2} \Delta \theta \quad (1.7)$$

où $\theta_1 = \theta_0 + \Delta \theta$ est la première approximation de θ . En égalisant l'équation (1.7) à zéro, on obtient le système d'équations suivant :

$$\frac{\partial^2 \ln p(t|\theta_0)}{\partial \theta^2} \Delta \theta = - \frac{\partial \ln p(t|\theta_0)}{\partial \theta} \quad (1.8)$$

Cette équation peut être résolue par rapport à $\Delta \theta$. La matrice de gradient $\frac{\partial^2 \ln(p(t|\theta_0))}{\partial \theta^2}$ est connue comme la matrice d'information de Fisher.

Il est possible d'appliquer l'algorithme du « maximum de vraisemblance » sur un système dynamique. Dans ce cas, il faudrait considérer la distribution Gaussienne E , déterminée par la matrice de covariance R , tel que :

$$E\{u(a_i)u^T(a_j)\} = R\delta_{ij} \quad (1.9)$$

où δ_{kl} représente le « delta de Kronecker », alors l'expression de la fonction de vraisemblance $p(t|\theta)$ peut se réécrire comme suit :

$$\begin{aligned} p[t(a_1), \dots, t(a_N)|\theta, R] &= \prod_{i=1}^N p[t(a_i)|\theta, R] \\ &= \{(2\pi)^{n_y}|R|\}^{-\frac{N}{2}} \exp \left[-\frac{1}{2} \sum_{i=1}^N [t(a_i) - y(a_i)]^T \times R^{-1} [t(a_i) - y(a_i)] \right] \end{aligned} \quad (1.10)$$

où t est le vecteur d'observations à n_y dimensions pour N mesures, $t(a_i)$ est la valeur de l'observation pour un point de mesure à l'instant a_i , θ est le vecteur de paramètres, et R est la matrice de covariance. On peut noter que la fonction de vraisemblance p correspond à la

probabilité des observations t et non pas à la probabilité des paramètres θ . L'estimation du maximum de vraisemblance signifie que le vecteur θ doit être déterminé afin de maximiser la fonction $p(t|\theta)$. Le vecteur $\hat{\theta}_{ML}$ obtenu est alors le plus « acceptable » afin d'obtenir le minimum d'erreur possible.

Il est possible de réécrire l'équation (1.10) en appliquant le logarithme de part et d'autre de l'égalité. Le problème d'optimisation devient alors un problème de minimisation de la fonction suivante :

$$L(t|\theta, R) = \frac{1}{2} \sum_{i=1}^N [t(a_i) - y(a_i)]^T R^{-1} [t(a_i) - y(a_i)] + \frac{N}{2} \ln[\det(R)] + \frac{Nn_y}{2} \ln(2\pi) \quad (1.11)$$

Il s'agit du logarithme de la fonction de vraisemblance p . La dérivée $\partial(\ln p) / \partial \theta = \frac{1}{p} \cdot \partial p / \partial \theta$ montre que le logarithme de p est nul lorsque $\partial p / \partial \theta$ est nul, ce qui amène au même résultat que celui obtenu par l'équation (1.6). Ainsi, l'équation (1.11) est plus facile à appliquer que l'équation (1.10) étant donné que le logarithme d'une fonction est plus facile à utiliser que la fonction elle-même.

1.5.2 Optimisation d'une fonction de coût

La plupart des algorithmes d'optimisation ont la tendance à minimiser une fonction que l'on nomme par « fonction de coût ». La fonction de coût est généralement notée par $J(\theta)$ et elle est dépendante des paramètres θ et R , où θ est le vecteur de paramètres que l'on doit identifier et R est la matrice de covariance, qui correspond aux erreurs entre les observations t et leurs valeurs simulées y .

Deux cas d'optimisation de la fonction de coût existent. Ces cas sont détaillés ci-après, dépendamment si la matrice de covariance R est connue ou inconnue.

1- Si la matrice R est connue, alors la fonction de coût peut être modélisée comme suit :

$$J(\theta) = \frac{1}{2} \sum_{i=1}^N [t(a_i) - y(a_i)]^T R^{-1} [t(a_i) - y(a_i)] \quad (1.12)$$

où N est le nombre d'observations, y est la valeur simulée et t est la valeur observée. Dans ce cas, la fonction de coût correspond à la somme des carrés des erreurs et elle est quadratique.

2- Si la matrice R est inconnue alors il est possible d'estimer la matrice de covariance R pour un vecteur de paramètres θ , tel que :

$$R = \frac{1}{N} \sum_{i=1}^N [t(a_i) - y(a_i)][t(a_i) - y(a_i)]^T \quad (1.13)$$

Ce résultat est obtenu grâce à l'égalisation par zéro de la dérivée par R de l'équation (1.11).

En substituant l'équation (1.13) dans (1.11), on remplace la matrice de covariance R par l'estimation de R , et on obtient la fonction de coût suivante :

$$J(\theta) = \frac{1}{2} n_y N + \frac{N}{2} \ln[\det(R)] + \frac{N n_y}{2} \ln(2\pi) \quad (1.14)$$

où n_y et N sont fixes pour un ensemble d'observations. Le premier et le troisième terme de la partie droite de l'équation (1.14) sont négligeables, étant donné que ce sont des valeurs constantes qui n'ont pas d'influence lors de la minimisation de la fonction de coût $J(\theta)$. Par conséquent, la fonction de coût peut se réécrire comme suit :

$$J(\theta) = \det(R) \quad (1.15)$$

Étant donné que la matrice R dépend du vecteur θ , il est primordial de très bien choisir les valeurs initiales de θ .

1.5.3 Algorithme de Gauss-Newton

La minimisation de la fonction de coût par rapport à θ est donnée par l'égalisation à zéro de la dérivée de la fonction de coût $J(\theta)$, comme décrit dans l'équation (1.16) suivante :

$$\frac{\partial J(\theta)}{\partial \theta} = 0 \quad (1.16)$$

L'approximation de l'équation (1.16) par le développement de la série de Taylor est donnée comme suit :

$$\left(\frac{\partial J(\theta)}{\partial \theta} \right)_{i+1} \approx \left(\frac{\partial J(\theta)}{\partial \theta} \right)_i + \left(\frac{\partial^2 J(\theta)}{\partial \theta^2} \right)_i \Delta \theta \quad (1.17)$$

où $\Delta \theta = \theta_{i+1} - \theta_i$ est la variation de θ à l' i ème itération, et $\left(\frac{\partial^2 J(\theta)}{\partial \theta^2} \right)_i$ est le gradient de second degré, qui correspond à la dérivée seconde par rapport à θ de la fonction de coût $J(\theta)$ à l' i ème itération. La matrice de gradients de second degré est désignée par « Hessien ». En appliquant la condition écrite dans l'équation (1.16), et en égalisant la partie droite de l'équation (1.17) par zéro, on obtient la valeur de $\Delta \theta$:

$$\Delta \theta = - \left[\left(\frac{\partial^2 J(\theta)}{\partial \theta^2} \right) \right]^{-1} \left(\frac{\partial J(\theta)}{\partial \theta} \right)_i \quad (1.18)$$

Le gradient $\left(\frac{\partial J(\theta)}{\partial \theta} \right)_{i+1}$ tend vers 0 lorsque l'itération augmente à $i + 1$. Il est alors nécessaire d'utiliser une valeur initiale θ_0 pour débiter l'algorithme, que l'auteur devra soigneusement choisir dès le début du processus itératif. La dérivée partielle par rapport à θ de la fonction de coût $J(\theta)$ est donnée par l'équation (1.19) suivante :

$$\frac{\partial J(\theta)}{\partial \theta} = - \sum_{i=1}^N \left[\frac{\partial y(a_i)}{\partial \theta} \right]^T R^{-1} [t(a_i) - y(a_i)] \quad (1.19)$$

Le calcul de cette dernière équation nécessite d'utiliser les valeurs des observations t , des variables y correspondantes aux t et du gradient $\frac{\partial y}{\partial \theta}$. La dérivée de deuxième ordre par rapport à θ de la fonction de coût $J(\theta)$ est alors exprimée comme suit :

$$\frac{\partial^2 J(\theta)}{\partial \theta^2} = \sum_{k=1}^N \left[\frac{\partial y(a_i)}{\partial \theta} \right]^T R^{-1} \frac{\partial y(a_i)}{\partial \theta} + \sum_{k=1}^N \left[\frac{\partial^2 y(a_i)}{\partial \theta^2} \right]^T R^{-1} [t(a_i) - y(a_i)] \quad (1.20)$$

Le calcul de cette dérivée est plus complexe que le calcul de la première dérivée et il demande plus de temps, étant donné qu'il nécessite le calcul de la seconde dérivée de la valeur simulée $\frac{\partial^2 y(a_i)}{\partial \theta^2}$. Toutefois, le second terme de l'équation (1.20) tend vers 0 lors après plusieurs itérations. Par conséquent il est possible de négliger le second terme de l'équation (1.20) et de la simplifier comme suit :

$$\frac{\partial^2 J(\theta)}{\partial \theta^2} \approx \sum_{i=1}^N \left[\frac{\partial y(a_i)}{\partial \theta} \right]^T R^{-1} \frac{\partial y(a_i)}{\partial \theta} \quad (1.21)$$

1.5.4 Algorithme de Levenberg-Marquardt

L'algorithme de Levenberg-Marquardt a lieu comme un processus itératif. Il s'agit d'une combinaison entre la méthode de Gauss – Newton et la méthode du gradient. La zone de convergence de l'algorithme est plus large étant donné que l'algorithme utilise une combinaison de deux méthodes pour minimiser la fonction de coût. L'ensemble des paramètres θ est décrit par le système d'équations (1.22) suivant :

$$\begin{aligned}\theta_{i+1} &= \theta_i + \Delta\theta \\ (F + \lambda I)\Delta\theta &= -G\end{aligned}\tag{1.22}$$

où λ est le paramètre de Levenberg-Marquardt (*LM*) qui peut varier afin de contrôler si sa direction tend vers celle de l'algorithme du gradient à l'équation (1.19) ou vers celle de l'algorithme de Gauss – Newton décrit par l'équation (1.21). Si $\lambda \rightarrow \infty$, alors l'algorithme devient celui du gradient et si $\lambda \rightarrow 0$ alors l'algorithme devient celui de Gauss – Newton. Les fonctions F et G sont calculées de la manière suivante :

$$\begin{aligned}F &= \sum_{i=1}^N \left[\frac{\partial y(a_i)}{\partial \theta} \right]^T R^{-1} \left[\frac{\partial y(a_i)}{\partial \theta} \right] \\ G &= - \sum_{i=1}^N \left[\frac{\partial y(a_i)}{\partial \theta} \right]^T R^{-1} [t(a_i) - y(a_i)]\end{aligned}\tag{1.23}$$

où F est la matrice d'information ou le « Hessien » décrit à l'équation (1.21), G est le vecteur gradient à l'équation (1.19), et $\Delta\theta$ est le vecteur de variation de θ . Après les calculs du gradient G et de la matrice de second degré F , les paramètres θ sont recalculés à partir de l'équation (1.22) pour la prochaine itération $i + 1$. Il est à noter que le Hessien F est une matrice définie positive et symétrique. Il reste alors à trouver un compromis pour la valeur de λ qui consistera dans le choix d'une solution optimale entre les deux méthodes. L'algorithme pour le choix du paramètre λ est présenté comme suit :

Étape 1 : Choisir un facteur de réduction $\nu > 1$. Initialiser comme suit : $\nu = 10$ et $\lambda_0 = 0.001$.

Étape 2 : Résoudre une première approximation de θ à l'itération i grâce à l'algorithme de Gauss – Newton, et passer à son approximation suivante à l'itération $i + 1$. On note λ^{i-1} la valeur issue de l'itération précédente.

Étape 3 : Résoudre le système d'équations linéaires (1.22) en remplaçant le paramètre λ par deux différents paramètres λ^{i-1} et λ^{i-1}/ν .

Étape 4 : Pour les deux solutions de $\Delta\theta$ obtenus dans l'étape 3, recalculer les valeurs de θ à l'équation (1.22), et calculer la fonction de coût pour les deux valeurs de θ grâce à l'équation (1.11), à savoir $L_i = L[\theta(\lambda^{i-1})]$ et $L_i^\nu = L[\theta(\lambda^{i-1}/\nu)]$.

Étape 5 : Comparer les valeurs de L_i et L_i^ν obtenues et déterminer la valeur de $\Delta\theta$ pour la prochaine approximation.

- a- Si $L_i^\nu \leq L_{i-1}$, alors $\lambda^i = \lambda^{i-1}/\nu$ et $\Delta\theta = \Delta\theta \left(\frac{\lambda^{i-1}}{\nu}\right)$
- b- Si $L_i^\nu > L_{i-1}$ et $L_i \leq L_{i-1}$, alors $\lambda^i = \lambda^{i-1}$ et $\Delta\theta = \Delta\theta(\lambda^{i-1})$, c'est-à-dire accepter le nouveau point, mais retenir le paramètre λ de l'itération précédente.
- c- Sinon si $L_i^\nu > L_{i-1}$ et $L_i > L_{i-1}$, alors $\lambda^{i-1} = \lambda^{i-1}\nu$, c'est-à-dire augmenter la valeur de λ et répéter les étapes 2 à 5 jusqu'à ce qu'un nouveau point acceptable soit trouvé.

Le choix de ν est arbitraire et dépend de l'auteur. La littérature énonce que $\nu = 10$ est un bon choix initial pour l'estimation de paramètres (Jategaonkar, 2015 chap. 4). En d'autres termes, l'algorithme de Levenberg-Marquardt calcule et compare la fonction de coût $L(\theta)$ entre les itérations initiales et les itérations précédentes pour les deux paramètres λ et ν . La fonction de coût qui présente une plus grande réduction est choisie et le programme lance ensuite les calculs pour la prochaine itération. Si les deux fonctions de coût sont toutes les deux plus grandes que celles de l'itération précédente, alors la valeur du paramètre λ est augmentée et l'itération recommence. De cette manière, l'algorithme de Levenberg – Marquardt garantit la convergence de la solution vers sa valeur optimale.

1.5.5 Algorithme de régularisation bayésienne

L'algorithme de régularisation bayésienne est principalement appliqué dans l'apprentissage des réseaux de neurones pour déterminer les vecteurs de poids w et les vecteurs de biais b associés à chaque neurone (section 1.4.2.1). Il possède une excellente propriété de généralisation et il utilise l'algorithme de Levenberg – Marquardt présenté dans la section 1.5.4 afin de calculer le minimum x d'une fonction de coût $F(x)$. Dans ce cas, la fonction de coût

$F(x)$ correspond à la somme quadratique des poids et biais E_w avec la somme quadratique des erreurs E_D , présentée dans l'équation (1.24) suivante :

$$F(x) = \alpha E_w + \beta E_D \quad (1.24)$$

où x est la combinaison du vecteur de poids w et du vecteur de biais b , et α et β sont les paramètres de la fonction de coût $F(x)$. La somme quadratique des poids et biais E_w et la somme quadratique des erreurs E_D sont calculées comme suit :

$$E_w = \sum_{i=1}^n x_i^2 \quad (1.25)$$

$$E_D = \sum_{i=1}^N (t(a_i) - y(a_i))^T (t(a_i) - y(a_i))$$

où t est le vecteur d'observations et y est le vecteur des valeurs simulées correspondant à t . Ainsi, l'algorithme de régularisation bayésienne calcule la combinaison optimale des paramètres linéaires α et β afin de minimiser la fonction de coût $F(x)$. Les étapes de l'algorithme sont les suivantes :

Étape 0 : Initialiser le vecteur de poids et biais x de façon aléatoire et calculer les valeurs de E_w et E_D à partir des équations (1.25). Calculer les valeurs initiales des paramètres α et β comme suit :

$$\alpha = \frac{\gamma}{2E_w(x)} \quad (1.26)$$

$$\beta = \frac{N - \gamma}{2E_D(x)}$$

où γ est le nombre initial de poids et de biais nécessaire pour minimiser la fonction de coût $F(x)$. Initialiser $\gamma = n$, où n est le nombre total de paramètres dans le neurone.

Étape 1 : Minimiser la fonction de coût $F(x)$ grâce à l'algorithme de Levenberg-Marquardt, présenté à la section 1.5.4.

Étape 2 : Recalculer le nombre de poids et de biais γ comme suit :

$$\gamma = n - 2\alpha \cdot \text{tr}(H)^{-1} \quad (1.27)$$

où la matrice H est le « Hessien », calculé à partir de l'approximation de Gauss – Newton comme suit :

$$H = \nabla^2 F(x) \approx 2\beta R^T R + 2\alpha I_n \quad (1.28)$$

où R est la matrice de covariance et I_n est une matrice unitaire de taille n

Étape 3 : Recalculer les nouvelles estimations de α et β à partir des équations (1.26).

Étape 4 : Répéter les étapes 1 à 3 jusqu'à la convergence de la fonction de coût $F(x)$ ou jusqu'à ce que les critères d'arrêts soient respectés. Ces critères d'arrêts de l'algorithme peuvent être : le nombre de cycles d'itérations, le temps d'apprentissage du réseau de neurones, et la limite de la variation de la fonction de coût (*e.g.* $\Delta F(x) < 0.001$).

1.6 Présentation de la turbosoufflante

La turbosoufflante (en anglais *Turbofan*) est un turboréacteur à double flux qui fournit la poussée nécessaire pour l'accélération et la décélération de l'avion lors du vol. Il permet de régulariser la vitesse de vol, d'effectuer des manœuvres comme le virage, de balancer l'avion en cas de panne d'un ou de plusieurs moteurs. Toutes les directives concernant le contrôle du moteur ainsi que les manœuvres à effectuer sont indiquées au pilote dans le manuel d'opération de vol (en anglais *Flight Operations Manuel* ou FOM).

Les principaux composants du turboréacteur à double flux sont : l'entrée d'air, la soufflante, la tuyère de sortie secondaire, le compresseur à haute pression (HPC), la chambre de combustion, la turbine à haute pression (HPT), la turbine à basse pression (LPT), l'arbre à basse pression, l'arbre à haute pression et la tuyère de sortie principale. Ces composants sont montrés dans la Figure 1.9.

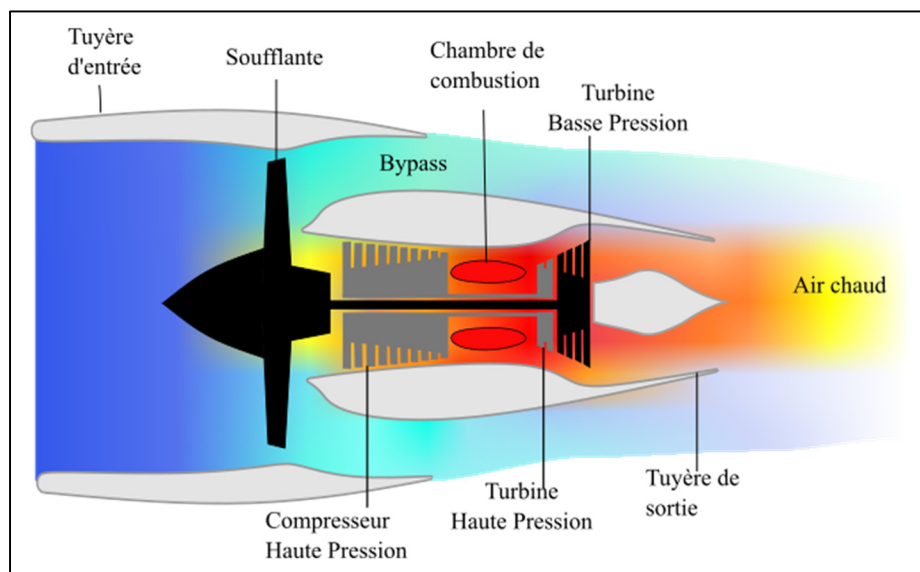


Figure 1.9 Les principaux composants du turboréacteur à double flux

L'air entre par l'entrée, traverse la soufflante, puis il est séparé en deux flux axiaux. Le flux principal qui est le flux chaud traverse le compresseur à haute pression HPC, où il est comprimé. Il entre ensuite dans la chambre de combustion où il est mélangé avec le carburant pour être brûlé. Le gaz formé traverse ensuite la turbine à haute pression HPT et la turbine à basse pression LPT pour être ensuite éjecté dans l'atmosphère à travers la tuyère de sortie. Le flux froid en contour produit environ 80 % de la poussée nette du turboréacteur à double flux.

La turbine à haute pression HPT est reliée au compresseur à haute pression HPC *via* l'arbre à haute pression (en gris sur la Figure 1.9). De la même manière, la turbine à basse pression LPT est reliée à la soufflante *via* l'arbre à basse pression (en noir sur la Figure 1.9). L'arbre à basse

pression et l'arbre à haute pression sont disposés de manière concentrique sur un axe, et ils tournent à différentes vitesses N_1 et N_2 respectivement où N_2 est généralement supérieure à N_1 .

1.6.1 Numérotation des composants

Le moteur est sectionné et numéroté par « station » afin de mieux repérer ses différents composants comme l'illustre la Figure 1.10.

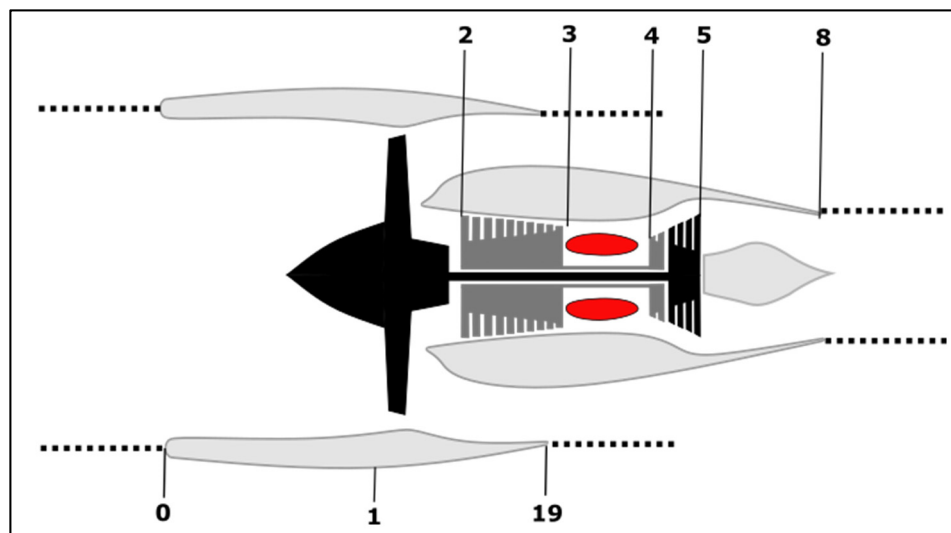


Figure 1.10 Numérotation des stations du turboréacteur à double flux

Pour un turboréacteur à double flux, avec un flux séparé en deux flux axiaux, les stations sont numérotées comme suit sur la Figure 1.10 :

Station 0 : Atmosphère ambiante.

Station 1 : Sortie de l'entrée d'air et l'entrée de la soufflante.

Station 2 : Sortie de la soufflante et l'entrée du compresseur à haute pression.

Station 3 : Sortie du compresseur à haute pression et l'entrée de la chambre de combustion.

Station 4 : Sortie de la chambre de combustion et l'entrée de la turbine.

Station 5 : Sortie de la turbine et l'entrée de la tuyère de sortie principale.

Station 8 : Sortie de la tuyère de sortie principale.

Station 19 : Sortie de la tuyère de sortie secondaire.

Les paramètres de pression, de température et du débit d'air varient à travers le moteur suivant un cycle thermodynamique déterminé connu par « cycle de Brayton », et présenté dans la section 1.7.3. Chaque paramètre est indexé avec les numéros de chaque composant selon la position des capteurs dans le moteur.

1.6.2 Principes de conception d'un turboréacteur

La conception d'un turboréacteur à double flux doit prendre en compte 3 études essentielles : l'étude thermodynamique, l'étude aérodynamique, et l'étude mécanique.

L'étude thermodynamique sert à déterminer le débit d'air, les ratios de pressions, les ratios de températures et la température à l'entrée de la turbine (*ITT*) comme décrit à la section 1.7.4. L'étude aérodynamique permet de déterminer les dimensions du moteur et de ses composants, à savoir le nombre et la forme des ailettes du compresseur et de la turbine, la vitesse de rotation de la soufflante, le diamètre de la soufflante, l'aire du col de la tuyère de sortie, *etc.* L'étude mécanique permet enfin de résoudre les problèmes liés aux vibrations, qui pourraient conduire à d'autres modifications au niveau des études thermodynamiques et aérodynamiques. Il sera alors nécessaire de reprendre le cycle de conception.

L'étude sur le contrôle du moteur devrait garantir sa stabilité. Pour ce faire, il est nécessaire de prédire les paramètres représentatifs de la température, de la pression, et du débit d'air/carburant à travers le moteur en vue de sélectionner les « points de design ». Les points de design sont choisis pour des conditions de vol stables. Ces conditions de vol incluent l'altitude, le nombre de Mach, la position de la manette des gaz et les propriétés de l'atmosphère ambiante, dont la température, la pression et la densité de l'air.

Les paramètres du turboréacteur sont définis lors de la phase de conception. Ils dépendent du type d'avion, du type de mission ainsi que des critères de performance attendus lors des manœuvres et des différentes phases de vol (décollage, montée, croisière, descente et atterrissage). Un avion civil long-courrier aurait par exemple besoin de moins de

consommation en carburant pour un vol en croisière et à haute altitude, qui est la plus longue phase de vol.

1.7 Modèle de performance du turboréacteur à double flux

Dans cette section, on discutera en détail la physique du turboréacteur à double flux, c'est-à-dire, du cycle thermodynamique et de son analyse afin de l'appliquer à l'identification d'un modèle de performance.

1.7.1 Principe de la propulsion

Dans le cas d'un turboréacteur, la propulsion consiste à éjecter une masse de gaz dans le sens opposé à son mouvement, suite à une succession de compression, combustion et détente, permettant ainsi à un aéronef d'acquérir une vitesse.

L'air entre dans le moteur avec une vitesse V_a égale à l'opposé de la vitesse de l'avion et sort par la tuyère de sortie avec une vitesse V_s . On considère que la masse d'air m est constante et que la masse de carburant est négligeable. D'une manière générale, la force de poussée F est due à la variation de la vitesse et elle est déterminée comme suit :

$$F = \dot{m}(V_s - V_a) \quad (1.29)$$

où $\dot{m}V_s$ est la poussée brute et $\dot{m}V_a$ est la force de traînée à l'entrée du moteur créée par les frictions de l'air.

En considérant que la pression p_a du gaz à l'entrée du moteur est plus petite que celle à la sortie p_s , à l'équation de la poussée (1.29) s'ajoute l'expression de la force générée par la différence de pression. La nouvelle expression de la poussée se réécrit alors comme suit :

$$F = \dot{m}(V_s - V_a) + A_s(p_s - p_a) \quad (1.30)$$

où A_s est l'aire du col de la tuyère de sortie.

Dans les prochaines sections, on développera en détail la force de poussée F_N qui résulte d'une analyse du cycle thermodynamique, appliquée sur une turbosoufflante. Avant de parvenir à cette étape, il est indispensable d'avoir un modèle d'atmosphère capable de prédire la température, la pression et la densité de l'air ambiante. La section 1.7.2 décrit le modèle d'atmosphère standard que l'on appliquera dans ce mémoire pour la modélisation des performances.

1.7.2 Modèle de l'atmosphère standard

Afin de modéliser les performances d'un avion, il est nécessaire d'avoir un modèle d'atmosphère qui pourra décrire l'environnement de vol de l'avion (Botez, 2006). Les propriétés de l'air qu'il faudrait modéliser sont notamment la température, la pression, et la densité. L'information sur la vitesse du son ne sera pas nécessaire, étant donné que le nombre de Mach sera directement mesuré à bord de l'avion.

L'atmosphère de la Terre est généralement divisée en plusieurs couches. Ces couches sont délimitées par des sections, qui sont des lignes imaginaires à des altitudes précises. La première couche correspond à la troposphère, et s'étend entre 0 km et 11 km d'altitude, où $h_{tropopause} = 11$ km correspond à la tropopause. La deuxième couche correspond à la stratosphère, et s'étend de 11 à 47 km, où $h_{stratopause} = 47$ km correspond à la stratopause. La plupart des vols commerciaux sont réalisés dans ces deux couches de l'atmosphère, et les aéronefs pilotés peuvent opérer jusqu'à environ 20 km d'altitude.

L'atmosphère est considérée comme statique, ce qui fait que la vitesse est partout uniforme et elle est égale à zéro. La densité, la température et la pression de l'air sont variables, et elles sont décrites à partir des équations hydrostatiques présentées dans les prochaines sections.

1.7.2.1 Température

La température est calculée en fonction de l'altitude et des conditions au niveau de la mer. Lorsque l'altitude est inférieure à la tropopause, la température en K est calculée en fonction de l'altitude comme suit :

$$T(h) = T_0 - L \cdot h \quad (1.31)$$

où h est l'altitude de vol de l'avion en m, $L = 0.0065$ K/m est le gradient de température en fonction de l'altitude, et $T_0 = 288.15$ K (15°C) est la température au niveau de la mer. Lorsque l'altitude est supérieure à la tropopause $h_{tropopause} = 11$ km, la température est constante et elle est égale à 216.65 K.

1.7.2.2 Pression statique

Lorsque l'altitude est inférieure à l'altitude de la tropopause, la pression statique en Pa est calculée en fonction de l'altitude comme suit :

$$P_s(h) = P_0 \left(1 - \frac{L \cdot h}{T_0} \right)^{\frac{g_0}{L \cdot R}} \quad (1.32)$$

Lorsque l'altitude est supérieure à celle de la tropopause, la pression statique en Pa est calculée en fonction de l'altitude comme suit :

$$P_s(h) = P_{tropopause} \cdot e^{\frac{-g_0}{R \cdot T_{tropopause}} (h - h_{tropopause})} \quad (1.33)$$

où $P_{tropopause} = P_s(h_{tropopause})$ est la pression statique au niveau de la tropopause calculée à partir de l'équation (1.31), $T_{tropopause} = 216.65$ K est la température au niveau de la tropopause, $P_0 = 101\,325$ Pa (2 116.2 lb/ft²) est la pression statique au niveau de la mer,

$g_0 = 9.81 \text{ m/s}^2$ est la constante gravitationnelle, et $R = 287 \text{ J/kg} \cdot \text{K}$ est la constante des gaz (Botez, 2006, Notes de cours).

1.7.2.3 Densité

Lorsque l'altitude est inférieure à celle de la tropopause, la densité en kg/m^3 est calculée en fonction de l'altitude comme suit :

$$\rho(h) = \frac{P_s}{R \cdot T} = \rho_0 \left(1 - \frac{L \cdot h}{T_0} \right)^{\frac{g_0}{L \cdot R} - 1} \quad (1.34)$$

où $\rho_0 = 1.225 \text{ kg/m}^3$ est la densité de l'air au niveau de la mer.

Lorsque l'altitude est supérieure à celle de la tropopause, la densité en kg/m^3 est calculée en fonction de l'altitude comme suit :

$$\rho = \rho_{tropopause} \cdot e^{\frac{-g_0}{R \cdot T_{tropopause}} \cdot (h - h_{tropopause})} \quad (1.35)$$

où $\rho_{tropopause} = \rho(h_{tropopause})$ est la densité au niveau de la tropopause, et elle est calculée à partir de l'équation (1.34) pour $h = h_{tropopause}$.

1.7.2.4 Ratio de températures et ratio de pressions

D'autres paramètres additionnels comme le ratio de températures ou le ratio de pressions peuvent être considérés afin d'obtenir un modèle adimensionnel de l'atmosphère. Le ratio de températures θ correspond au rapport de la température à une altitude h par la température au niveau de la mer. Il est défini comme suit :

$$\theta = \frac{T(h)}{T_0} \quad (1.36)$$

Le ratio de pressions δ correspond au rapport de la pression statique à une altitude h par la pression statique au niveau de la mer. Il est défini comme suit :

$$\delta = \frac{P_s(h)}{P_0} \quad (1.37)$$

Ces deux paramètres servent généralement comme des facteurs de correction pour les paramètres du moteur, présenté dans la section 1.7.5.

1.7.3 Cycle de Brayton

Afin de faire voler un avion, il est nécessaire d'avoir un système de propulsion qui génère la force de poussée. En aéronautique, le système de propulsion le plus répandue est la turbine à gaz. Il existe plusieurs types de turbines à gaz dont le turbomoteur, le turbopropulseur, le turboréacteur et le turboréacteur à double flux. Leur fonctionnement est régi par un même principe : le corps central est composé d'un compresseur, d'une chambre de combustion et d'une turbine qui entraîne le compresseur. Les lois de thermodynamique qui régissent le fonctionnement des moteurs à réaction sont les mêmes. Pour un moteur d'avion, le cycle thermodynamique est ouvert, c'est-à-dire que l'air provenant de la turbine est éjecté dans l'atmosphère, au lieu de circuler vers le compresseur. L'énergie chimique obtenue grâce à la combustion du mélange air/carburant est convertie en travail suivant deux principes : soit que la turbine produit plus de puissance mécanique, soit que la tuyère de sortie augmente le moment du gaz pour produire une force de poussée.

Pour comprendre le système de propulsion, il faut connaître les propriétés thermodynamiques des gaz. En effet, un gaz est caractérisé par sa pression P , son volume V et sa température T . Lors des processus d'échauffement ou de compression, ces trois paramètres sont liés entre eux et varient selon le cycle thermodynamique de Brayton. Le diagramme Température (T) *versus*

Entropie (s) de la Figure 1.11 illustre le cycle idéal de Brayton pour un turboréacteur à double flux :

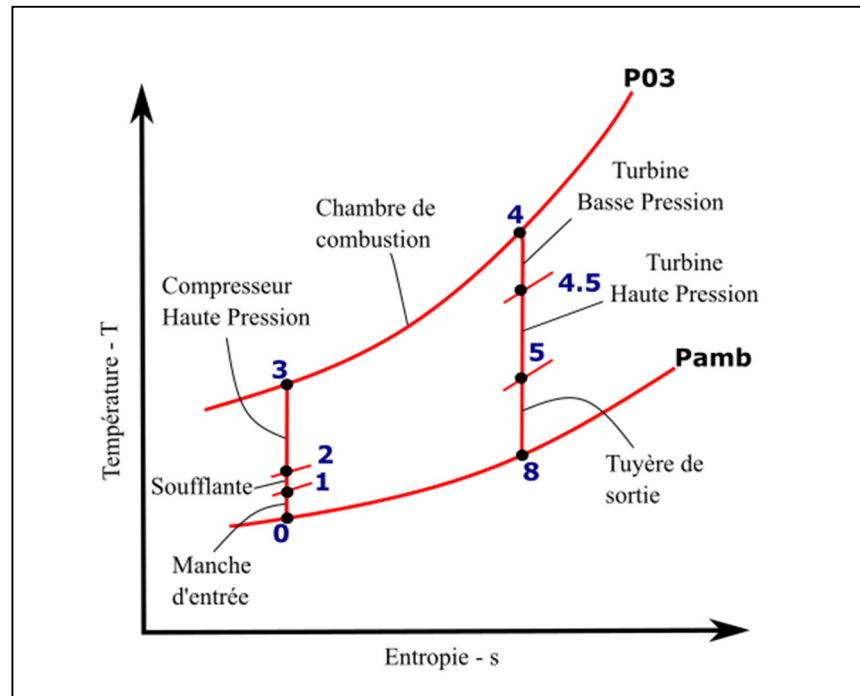


Figure 1.11 Diagramme Température *versus* Entropie ($T - s$)

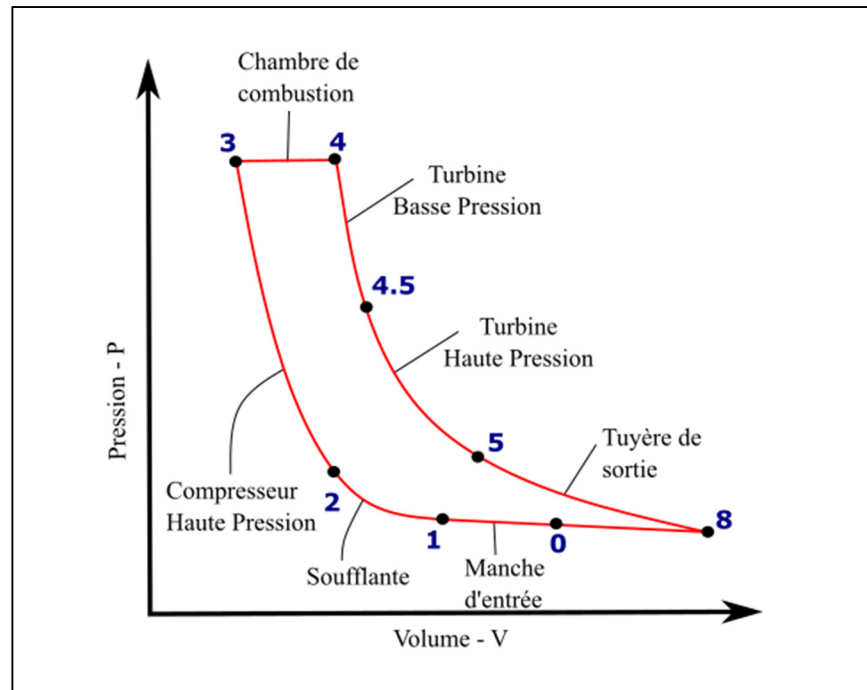


Figure 1.12 Diagramme Pression *versus* Volume ($P - V$)

Le cycle commence par les conditions atmosphériques à la station 0 de la Figure 1.10. En condition de vol, l'air entre par la manche d'entrée d'air à la station 0 et il est décéléré jusqu'à l'entrée du compresseur à la station 2. Comme l'air est décéléré, l'énergie cinétique de l'avion augmente la pression statique de l'air. L'air est alors compressé selon un processus isentropique avec une entropie constante, et la température augmente entre les stations 0 et 3, comme on le voit sur la Figure 1.11. Le compresseur exerce un travail sur le gaz et augmente la pression et la température de manière isentropique jusqu'à la sortie du compresseur à la station 3. Dans le cas d'un cycle réel (*i.e.* non idéal), la compression n'est pas isentropique, et l'entropie augmente légèrement. L'air est ensuite mélangé avec le carburant et il est brûlé pour former un gaz dans la chambre de combustion entre les stations 3 et 4. La température augmente tandis que la pression reste constante selon un processus isobare. L'air passe ensuite à travers la turbine entre les stations 4 et 5. En régime permanent, étant donné que la turbine et le compresseur sont liés par un arbre, le travail effectué par le gaz sur la turbine est égal au travail effectué par le compresseur sur le gaz; la variation de température est donc la même pour la turbine et le compresseur. Finalement, la tuyère de sortie rejette l'air dans l'atmosphère

entre les stations 5 et 8 et le gaz retourne dans les conditions atmosphériques initiales pour fermer le cycle. La surface sous le diagramme $T - s$ représente le travail de la force de poussée effective générée par le turboréacteur.

1.7.4 Modèle de performance à partir de l'Approche par Composant

L'analyse du cycle thermodynamique de la turbosoufflante sera effectuée selon (Hill and Peterson. 1992) et (Saravanamuttoo, 2014).

Les principaux paramètres du turboréacteur à double flux sont :

- Le taux de dilution du moteur, en anglais *Bypass Ratio* ou *BPR*.
- Le ratio de pression du moteur, en anglais *Engine Pressure Ratio* ou *EPR*.
- Le ratio de pression de la soufflante, en anglais ou *Fan Pressure Ratio* ou *FPR*.
- Le ratio de pression totale, en anglais *Overall Pressure Ratio* ou *OPR*.
- La vitesse de rotation de la soufflante N_1 .
- La vitesse de rotation de la turbine N_2 .
- La température à l'entrée de la turbine, en anglais *Interturbine temperature* ou *ITT*.
- L'aire du col de la tuyère de sortie du flux principal et du flux secondaire.
- Le rendement de chaque composant, à savoir le rendement de la soufflante η_f , le rendement du compresseur η_c , le rendement de combustion η_b , le rendement de la turbine η_t , le rendement des tuyères de sortie principale η_n et secondaire η_{fn} .

Le *OPR* est le rapport de pression entre la pression maximale du moteur et la pression atmosphérique, et il définit le nombre de phases du compresseur et de la turbine, ainsi que la longueur et le poids du moteur lors de sa conception. La température à l'entrée de la turbine *ITT* définit les limites de températures au niveau de la turbine. Le taux de dilution *BPR*, la vitesse de rotation de la soufflante N_1 et le ratio de pression de la soufflante *FPR* sont essentiels pour définir le diamètre de la soufflante.

Comme décrit dans le cycle de Brayton (section 1.7.3), le cycle d'un turboréacteur à double flux commence par l'établissement des conditions atmosphériques. Les ratios de températures et de pressions totales sont calculés comme suit pour les conditions atmosphériques :

$$\frac{T_{01}}{T_a} = 1 + \frac{\gamma_a - 1}{2} M^2 \quad (1.38)$$

$$\frac{P_{01}}{P_a} = \left(\frac{T_{01}}{T_a} \right)^{\frac{\gamma_a}{\gamma_a - 1}} \quad (1.39)$$

où T_{01} et P_{01} sont les température et pression totales à la station 1 au niveau de l'entrée d'air, T_a et P_a sont les température et pression statiques de l'atmosphère ambiante à la station 0, $\gamma_a = 1.4$ est le ratio de chaleur spécifique de l'air. Pour des conditions de vol transsoniques et subsoniques ($M < 1$), les ratios de températures et de pressions au niveau de la soufflante (station 1 à 2) sont définis comme suit :

$$\frac{T_{02}}{T_{01}} = FPR^{\frac{\gamma_a - 1}{\eta_f \gamma_a}} \quad (1.40)$$

$$\frac{P_{02}}{P_{01}} = FPR \quad (1.41)$$

où FPR est donné et η_f est le rendement de la soufflante.

Les ratios de températures et de pressions dans le compresseur (station 2 à 3), sont définis comme suit :

$$\frac{P_{03}}{P_{02}} = \frac{OPR}{FPR} \quad (1.42)$$

$$\frac{T_{03}}{T_{02}} = \frac{1}{\eta_c} \left(OPR^{\frac{\gamma_a - 1}{\gamma_a}} - 1 \right) \quad (1.43)$$

où OPR et FPR sont donnés et η_c est le rendement du compresseur.

Les ratios de température et de pression dans la chambre de combustion (station 3 à 4), sont définis comme suit :

$$T_{04} = ITT \quad (1.44)$$

$$\frac{T_{04}}{T_{03}} = T_{04} \left(\frac{T_{02}}{T_{03}} \right) \left(\frac{1}{T_{01}} \right) \quad (1.45)$$

$$\frac{P_{04}}{P_{03}} = \eta_b \quad (1.46)$$

où ITT est donné et η_b est le rendement de la combustion.

Les ratios de températures et de pressions au niveau de la turbine (station 4 à 5) sont définis comme suit :

$$\frac{T_{05}}{T_{04}} = 1 - \left(\frac{T_{03}}{T_{02}} - 1 \right) \left(\frac{T_{01}}{T_{04}} \right) \quad (1.47)$$

$$\frac{P_{05}}{P_{04}} = \left(1 - \frac{1}{\eta_t} \left(1 - \frac{T_{05}}{T_{04}} \right) \right)^{\frac{\gamma_g}{\gamma_g - 1}} \quad (1.48)$$

où η_t est le rendement de la turbine et $\gamma_g = 1.33$ est le ratio de chaleur spécifique du gaz. Cette valeur de γ_g a été choisie pour un mélange de sgaz après combustion (Saravanamuttoo, 2014).

Ainsi, le ratio de température totale (en anglais *Engine Temperature Ratio* ou ETR) et le ratio de pression totale (en anglais *Engine Pressure Ratio* ou EPR) peuvent être calculés comme suit :

$$ETR = \left(\frac{T_{05}}{T_{04}} \right) \left(\frac{T_{04}}{T_{03}} \right) \left(\frac{T_{03}}{T_{02}} \right) \left(\frac{T_{02}}{T_{01}} \right) \quad (1.49)$$

$$EPR = \left(\frac{P_{05}}{P_{04}}\right) \left(\frac{P_{04}}{P_{03}}\right) \left(\frac{P_{03}}{P_{02}}\right) \left(\frac{P_{02}}{P_{01}}\right) = \left(\frac{P_{05}}{P_{01}}\right) \approx \left(\frac{P_{05}}{P_a}\right) \quad (1.50)$$

En supposant une condition de choc sur le col de la tuyère de sortie principale, la vitesse de sortie du gaz à la station 8 est définie par l'équation (1.51) suivante :

$$V_8 = \sqrt{2c_{pg}T_{05}\eta_j \left[1 - \left(\frac{1}{EPR} \right)^{\frac{\gamma_g-1}{\gamma_g}} \right]} \quad (1.51)$$

où η_j est le rendement de la tuyère de sortie principale, T_{05} et P_{05} sont calculées à partir de ETR et EPR ainsi que des ratios de températures et de pressions décrites par les équations (1.38) et (1.39), et $c_{pg} = 1.148$ définit la capacité thermique massique du gaz.

En supposant l'existence d'une onde de choc sur le col de la tuyère de sortie secondaire, la vitesse de sortie du gaz à la station 19 est définie comme suit :

$$V_{19} = \sqrt{2c_{pa}T_{02}\eta_{fn} \left[1 - \left(\frac{1}{FPR} \right)^{\frac{\gamma_a-1}{\gamma_a}} \right]} \quad (1.52)$$

où η_{fn} est le rendement de la tuyère de sortie secondaire, T_{02} et P_{05} sont calculées à partir de ETR et EPR ainsi que des ratios de températures et de pressions décrites par les équations (1.38) et (1.39), et $c_{pa} = 1.005$ définit capacité thermique massique de l'air.

Le débit massique de l'air m_0 est calculé en considérant les ondes de choc. Le débit massique corrigé noté par $\frac{m_0\sqrt{\theta}}{\delta A}$ dépendra seulement du nombre de Mach M comme suit:

$$\frac{\dot{m}_0 \sqrt{\theta}}{\delta A_{19}} = \frac{\sqrt{\frac{\gamma_a}{R_{air}}} P_0}{\sqrt{T_0}} M \left[1 + \frac{\gamma_a - 1}{2} M^2 \right]^{\frac{-\gamma_a + 1}{2(\gamma_a - 1)}} = f(M) \quad (1.53)$$

où $\dot{m}_0$ est le débit massique de l'air entrant par l'entrée, θ et δ sont les ratios de températures et de pressions définis à la section 1.7.2.4, A_{19} est l'aire du col de la tuyère de sortie secondaire à la station 19, et $R = 287 \text{ J/kg} \cdot \text{K}$ est la constante des gaz. Il est alors possible de retrouver la valeur de $\dot{m}_0$ en multipliant le débit massique corrigé par $\left(\frac{\delta \cdot A_{19}}{\sqrt{\theta}} \right)$:

$$\dot{m}_0 = f(M) \left(\frac{\delta \cdot A_{19}}{\sqrt{\theta}} \right) \quad (1.54)$$

où $f(M)$ est une notation signifiant une dépendance du nombre de Mach.

Le taux de dilution du moteur (en anglais *Bypass Pressure Ratio* ou *BPR*) est un paramètre de conception généralement connu. Il correspond au rapport entre le débit d'air en contour du moteur $\dot{m}_f$ avec le débit d'air entrant dans le corps principal du moteur $\dot{m}_c$, soit :

$$BPR = \frac{\dot{m}_f}{\dot{m}_c} \quad (1.55)$$

En supposant que le débit massique d'air $\dot{m}_0$ est connu à partir de l'équation (1.54), alors les débits $\dot{m}_f$ et $\dot{m}_c$ peuvent être déterminés comme suit :

$$\dot{m}_f = \frac{\dot{m}_0 \cdot BPR}{BPR + 1} \quad (1.56)$$

$$\dot{m}_c = \frac{\dot{m}_0}{BPR + 1} \quad (1.57)$$

La force de poussée définie par les équations (1.29) et (1.30) peut alors être réécrite en assumant que la vitesse de l'air est négligeable à l'entrée de la turbosoufflante et que $V_s \gg V_a$. La « poussée froide » notée par F_{N_1} est alors obtenue comme suit :

$$F_{N_1} = \dot{m}_f \cdot V_{19} \quad (1.58)$$

La poussée générée par le corps principal du moteur, notée par F_{N_2} est obtenue comme suit :

$$F_{N_2} = \dot{m}_c \cdot V_8 \quad (1.59)$$

Si un choc intervient sur le col de la tuyère de sortie principale A_8 ou sur le col de la tuyère de sortie secondaire A_{19} , alors il faut prendre en compte la chute de pression lors du calcul de la poussée. Dans ce cas, la pression au niveau du col de la tuyère, appelée « pression critique » est calculée comme suit :

$$P_c = P_{sortie} * \left[\frac{\eta}{\eta - \frac{\gamma - 1}{\gamma + 1}} \right]^{\frac{\gamma}{\gamma - 1}} \quad (1.60)$$

où $(P_{sortie}, \eta, \gamma)$ prennent les valeurs $(P_{02}, \eta_f, \gamma_a)$ pour la tuyère de sortie secondaire à la station 19, et prennent les valeurs $(P_{05}, \eta_{fn}, \gamma_g)$ pour la tuyère de sortie principale à la station 8. La poussée créée par la chute de pression s'ajoute alors aux poussées F_{N_1} et F_{N_2} déterminées par les relations (1.58) et (1.59) qui se réécrivent comme suit :

$$F_{N_1} = \dot{m}_f \cdot V_{19} + A_{19}(P_c - P_a) \quad (1.61)$$

$$F_{N_2} = \dot{m}_c \cdot V_8 + A_8(P_c - P_a) \quad (1.62)$$

Enfin, la poussée totale correspond à la somme des deux poussées F_{N_1} et F_{N_2} .

$$F_N = F_{N_1} + F_{N_2} \quad (1.63)$$

La consommation en carburant W_F peut être exprimée comme suit :

$$W_F = RU * (m_c) \quad (1.64)$$

où RU (en anglais par *Ratio Unit*) est connu pour chaque type de moteur, car il représente le rapport entre le débit d'air du corps principal m_c et le débit de carburant $m_{carburant}$.

$$RU = \frac{m_c}{m_{carburant}} \quad (1.65)$$

Cette analyse thermodynamique du cycle de Brayton montre que les températures et les pressions locales du moteur sont principalement dépendantes des paramètres de design du moteur qui sont le OPR , le FPR et le ITT . Dans le cas d'une turbosoufflante, le taux de dilution du moteur BPR est également pris en compte et il est généralement constant.

Dans le cadre de ce mémoire, le principe de « l'approche par composant » consiste à modéliser les paramètres de design du moteur à partir des réseaux de neurones, à savoir des paramètres suivants : EPR , FPR , ITT , OPR , N_1 et N_2 , ainsi que les paramètres de performances dont la poussée F_N et la consommation carburant W_F .

1.7.5 Modèle de performance simplifié

Une linéarité devrait exister entre la position de la manette des gaz TLA et la poussée F_N . Cependant, la poussée ne peut pas être directement mesurée. D'où la nécessité de contourner sa non-linéarité. Ceci peut être réalisé en utilisant la vitesse de rotation corrigée de la soufflante $N_{1,C}$, afin de modéliser la poussée et le débit de carburant. Ainsi une deuxième approche est

possible pour établir un modèle de performance en considérant seulement la vitesse de rotation de la soufflante N_1 .

En effet, la poussée F_N et le débit de carburant W_F dépendent généralement de la vitesse de rotation de la soufflante N_1 , de différents paramètres de l'air comme la vitesse de vol réelle V_T , la température ambiante T , la pression ambiante P , la constante de l'air R_{air} , ainsi que différentes caractéristiques physiques du moteur $\lambda = \{\lambda_1, \dots, \lambda_n\}$. Ces deux paramètres de performance peuvent être écrits mathématiquement sous la forme suivante :

$$F_N = f(N_1, V_T, T, P, R_{air}, \lambda) \quad (1.66)$$

$$W_F = f(N_1, V_T, T, P, R_{air}, \lambda) \quad (1.67)$$

où F_N est la poussée et W_F est le débit de carburant.

Les équations (1.66) et (1.67) représentent des fonctions mathématiques incluant un nombre minimum de 6 paramètres, dépendamment du nombre des paramètres λ , équivalents aux paramètres du moteur décrits dans la section précédente. Aussi, il est évident qu'identifier un tel système peut être une tâche très complexe, et nécessite un nombre important de données, afin d'identifier l'influence de chaque paramètre sur les performances du moteur. Toutefois, en utilisant une analyse dimensionnelle, comme celle présentée par (Ghazi et Botez, 2019) il est possible de réduire le nombre de paramètres, et par conséquent, il est possible de simplifier la représentation des performances, comme suit :

$$\frac{F_N}{\delta} = f\left(\frac{N_1}{\sqrt{\theta}}, M\right) \quad (1.68)$$

$$\frac{W_F}{\delta\sqrt{\theta}} = f\left(\frac{N_1}{\sqrt{\theta}}, M\right) \quad (1.69)$$

où $F_{N,C} \equiv F_N/\delta$ est la poussée corrigée, $W_{F,C} \equiv W_F/(\delta\sqrt{\theta})$ est le débit de carburant corrigé, δ est le ratio de pressions de l'air et θ est le ratio de températures de l'air définis dans la section 1.7.2.4.

Le fait de représenter les modèles de performance sous cette forme est un avantage, du fait que les résultats ne dépendent que de deux variables, dont la vitesse de rotation de la soufflante corrigée $N_{1,C} \equiv N_1/\sqrt{\theta}$ et le nombre de Mach M . Ces équations permettent de réduire considérablement la complexité des systèmes d'équations que l'on doit modéliser, et indirectement, de réduire le nombre de données nécessaires pour l'identification du modèle.

1.7.6 Modes de poussées

Le paramètre $\frac{N_1}{\sqrt{\theta}}$ des équations (1.68) et (1.69) correspond à la vitesse de rotation corrigée de la soufflante et il détermine la puissance du moteur. Ce paramètre est directement étroitement lié à la position de la manette des gaz et permet de varier la puissance du moteur lors de différentes phases de vol.

Lorsque le pilote monte la manette des gaz vers le haut, un signal est envoyé au Système de Contrôle du Moteur (en anglais *Engine Control System* ou ECS) afin d'augmenter la vitesse de rotation de la soufflante, et ainsi d'augmenter la poussée. De la même manière, lorsque le pilote abaisse la manette des gaz, un signal est envoyé à l'ECS afin de diminuer la vitesse de rotation de la soufflante et ainsi de réduire la poussée.

Afin d'éviter que les composants du moteur n'opèrent pas au-delà des limites établies, l'ECS impose des limites de poussée qu'il ne faut pas dépasser. Ces limites, qu'on appelle également

« niveau de poussée », définissent la vitesse maximale de rotation de la soufflante qu'il est possible de contrôler pour des conditions de vol spécifiques.

Pour la plupart des avions commerciaux, il existe trois niveaux de poussée que le pilote est amené à utiliser, dépendamment de la phase de vol. Ces niveaux de poussée sont : 1) la poussée maximale de décollage (en anglais *Maximum Takeoff Thrust* ou MTO), 2) la poussée maximale de montée (en anglais *Maximum Climb Thrust* ou MCLB), et 3) la poussée minimale (en anglais *idle* ou IDLE). Chacun de ces niveaux de poussée peut être sélectionné en plaçant la manette des gaz sur une position spécifique, et devrait être utilisé respectivement lors des phases de décollage, de montée et de descente.

D'autre part, la phase de croisière est la seule phase où la manette des gaz doit être constamment ajustée entre les positions IDLE et MCLB. Dans ce cas, la vitesse de rotation de la soufflante N_1 correspondante est calculée par l'ECS en présumant une variation continue entre les valeurs IDLE et MCLB.

Ainsi, il est nécessaire de définir les conditions de vol qui serviront de base pour l'identification du réseau de neurones. Pour ce faire, chaque point de vol aura ses données d'altitudes, de nombres de Mach, de positions de la manette des gaz, de températures, de pressions et de densités.

1.8 Modèles empiriques de performance

Bartel et Young, 2008, appliquent les modèles de performances émis par Torenbeek, qui consistent en une approximation de la consommation spécifique de carburant (en anglais *Thrust Specific Fuel Consumption* ou TSFC) et de la poussée. Les paramètres du modèle ont été déterminés à l'aide de courbes pour trois différentes phases de vol, dont la montée, la croisière et la descente. Les auteurs ont notamment analysé l'influence de l'altitude et de

l'extraction d'air au niveau du compresseur sur la poussée. Les résultats de la modélisation ont montré une erreur relative inférieure à 4.5 % pour la poussée lors de la phase de montée.

Outre le modèle de performance par « approche par composant » (Botez et *al.*, 2019) identifie les modèles de performances de l'avion Cessna Citation X à partir du modèle de poussée énoncé par (Mattingly et *al.* 2002), du modèle de poussée et de consommation de carburant énoncé par Torenbeek, et du modèle de Yoder. Ces derniers modèles de performances sont basés sur des formules mathématiques empiriques dont leurs paramètres ont été déterminés à partir des algorithmes d'estimations comme celui de Levenberg – Marquardt et l'algorithme d'Optimisation par Essaim de Particule (en anglais *Particle Swarm Optimization* ou PSO). L'étude comparative a pu démontrer une erreur relative inférieure à 5 % pour la prédiction de la poussée, et une erreur relative inférieure à 15 % pour la consommation en carburant pour la phase de croisière.

1.9 Paramètres de performances additionnels

En plus du débit de carburant W_F , et de la poussée F_N , d'autres paramètres de performances peuvent être pris en compte, comme la consommation spécifique de carburant, en anglais *Specific Fuel Consumption* ou SFC, le rendement de propulsion η_p , le rendement de conversion d'énergie η_e et le rendement total du moteur η_o (Saravanamuttoo, 2014).

La consommation spécifique de carburant SFC correspond au rapport du débit de carburant par la poussée.

$$SFC = \frac{W_F}{F_N}$$

Le *SFC* est un critère important lors de la conception et du choix d'un turboréacteur.

Le rendement de propulsion η_p correspond au rapport de l'énergie effective de propulsion par la somme de cette énergie avec l'énergie cinétique non utilisée. Le rendement η_p peut être défini comme suit :

$$\eta_p = \frac{2}{1 + \left(\frac{V_s}{V_a}\right)^2} \quad (1.70)$$

où V_s est la vitesse de sortie du gaz et V_a est la vitesse de l'air ambiant à l'entrée du moteur. En reprenant les équations (1.29) et (1.30), on peut établir que la force de poussée F est maximale lorsque $V_s \gg V_a$, c'est-à-dire pour des conditions statiques où la vitesse de l'air ambiant est négligeable, dans ce cas, le rendement de propulsion η_p est minimal. Cependant le rendement de propulsion η_p est maximal lorsque $V_a = V_s$, mais dans ce cas la poussée F est minimale. Ainsi, afin d'obtenir un rendement élevé, mais aussi une force de poussée élevée, il faut trouver un compromis entre les vitesses V_a et V_s , de sorte que V_s soit supérieure à V_a , mais que cette différence ne dépasse pas une certaine limite.

Un autre paramètre de performance est le rendement de la conversion d'énergie que l'on peut déterminer comme suit :

$$\eta_e = \frac{\dot{m}(V_s^2 - V_a^2)/2}{\dot{m}_{carburant} Q_{net,p}} \quad (1.71)$$

où $\dot{m}_{carburant}$ est le débit de carburant et $Q_{net,p}$ est la chaleur nette délivrée à la sortie de la chambre de combustion. L'énergie thermique est ensuite convertie en énergie cinétique.

Le dernier paramètre de performance est le rendement total du moteur η_o . Il correspond au produit du rendement de propulsion η_p avec le rendement de la conversion d'énergie η_e et il est défini comme suit :

$$\eta_o = \eta_p \eta_e \quad (1.72)$$

Dans le cadre de ce mémoire, on prendra en compte la consommation en carburant W_F et la poussée F_N comme critères de performance, ce qui implique directement la consommation spécifique de carburant SFC .

1.10 Ordinateur de contrôle du moteur FADEC

Le turboréacteur est muni d'un ordinateur de contrôle connu en anglais sous le nom de *Full Authority Digital Engine Controller* ou FADEC. Selon le *Federal Aviation Administration*, le FADEC est un élément indispensable de l'avion pour assurer la sécurité du vol. Il permet de réduire la charge de travail du pilote, de superviser le moteur et d'avertir le pilote de tout problème. Il est entre autres responsable de l'allumage du moteur et de son bon fonctionnement. Ainsi, il n'existe aucun contrôle direct du pilote sur le moteur. Si le FADEC tombe en panne, alors le moteur tombe également en panne. Toutefois, les constructeurs se sont assurés du bon fonctionnement du FADEC en y ajoutant des systèmes de redondance, ce qui fait qu'il risque moins les défaillances.

Le FADEC transforme les signaux de contrôle provenant de la manette des gaz, des différents capteurs de la soufflante et du débit de carburant en un seul signal de contrôle. Chaque position de la manette des gaz est la position optimale qui assure une combinaison optimale entre puissance/propulsion et vitesse de rotation de la soufflante/débit de carburant pour chaque altitude de vol. Cela permet ainsi au pilote d'économiser en carburant et de réduire l'émission de CO_2 dans l'atmosphère.

Le FADEC utilise deux méthodes de contrôle (Chapman et Litt, 2017) : 1) le contrôle de la vitesse de rotation de la soufflante N_1 , dont l'élément de contrôle est l'aire du col de la tuyère de sortie secondaire (en anglais *Variable Area Fan Nozzle* ou VAFN). 2) le contrôle du ratio de pression du moteur EPR , dont ses éléments de contrôle sont les valves d'extraction d'air (en anglais *Variable Bleed Valve* ou VBV), qui sont installés sur les étages du compresseur.

Mais il existe d'autres boucles de contrôle dont le FADEC est responsable, comme : 1) le contrôle de la température de la turbine (en anglais *Interturbine Temperature* ou *ITT*). Dans ce cas, le FADEC contrôle la valve de mesure du carburant (en anglais *Fuel Metering Valve*), plus précisément du débit de carburant qui entre dans la chambre de combustion, mais aussi le système de refroidissement de la turbine grâce aux valves d'extraction d'air au niveau de la turbine et 2) le contrôle de la vitesse de rotation de la turbine N_2 .

Selon (Csank, J. et al., 2010), le système de contrôle du moteur aurait les trois principales fonctions : 1) maintenir la poussée à une valeur déterminée selon la position de la manette des gaz. 2) assurer une excellente performance lors de la transition entre deux modes de poussée et 3) assurer le bon déroulement et la sécurité des opérations. Les deux dernières fonctions sont difficilement réalisables lors des modes de poussées maximales, durant lesquels le moteur opère près des limites de l'enveloppe de vol, tandis que les deux premières fonctions consistent à contrôler la poussée du moteur. Ainsi, la logique du FADEC peut être divisée en deux catégories : 1) la gestion de la poussée et 2) la mise en place de barrières de protection.

La supervision se fait de manière automatique et permet de "protéger" le moteur en fixant des limites de vitesse, de pressions, et de température de la turbine. Ceci permet d'éviter le décrochage ou encore le phénomène de pompage, qui se traduit par l'écoulement inverse du flux. Pour ce faire, le FADEC contrôle automatiquement les valves d'extraction d'air au niveau du compresseur et de la turbine afin de réduire le ratio de pression du compresseur *OPR* et/ou le ratio de pression du moteur *EPR*.

Différentes structures de contrôle sont présentes dans la littérature (Csank et al., 2010) (Chapman et Litt, 2017) (Chapman et Litt, 2020) (Bazazzadeh et al., 2009) (Ruichao et al., 2018) (Roberts et Eastbourne, 2014) afin de modéliser un système dynamique. Cependant, seulement le modèle statique est abordé dans ce mémoire, dont l'objectif est de « mimer » le FADEC à partir des réseaux de neurones. Aussi, dans le cadre de ce mémoire, la modélisation sera réalisée sur toute l'enveloppe de vol de l'avion. On ne modélisera donc pas les barrières

de protection du moteur, étant donné que celles-ci sont automatiquement prises en compte dans les données d'identification lors de la réalisation des tests en vol.

1.11 Modélisation sur Matlab/Simulink

Roberts et Eastbourne, 2014, déterminent un modèle de propulsion dynamique d'un turboréacteur à double flux à partir de la plateforme Simulink intégrée à Matlab en adoptant la technique d'« approche par composants ». Chaque composant du turboréacteur à double flux, dont le compresseur à haute pression, la soufflante, la chambre de combustion, la turbine à haute pression, la turbine à basse pression, la tuyère de sortie principale et la tuyère de sortie secondaire ont été modélisés et validés d'une manière indépendante. Cette approche permet de très bien représenter les conditions physiques du moteur, de réduire le temps de simulation, et d'augmenter la précision des calculs grâce à des programmes numériques performants. Cette approche permet également de modéliser le comportement du moteur suite à différents types de perturbations selon les conditions de vol.

Bazazzadeh et *al.*, 2009 modélisent le système de contrôle du débit de carburant à partir d'un modèle établi sur Matlab/Simulink. Ce modèle permet de prédire les paramètres du système de contrôle, de réduire les essais coûteux et de satisfaire les critères de stabilité et de contrôle.

Chapman et Litt, 2020, proposent une loi de contrôle pour le système de moteur d'un turboréacteur. Le système identifié a été implémenté sur Matlab/Simulink en utilisant la boîte d'outils pour la modélisation et l'analyse de système thermodynamique (T-MATS). Ce modèle utilise la méthode de la balance d'énergie ainsi que les performances de chaque composant afin d'identifier un système non linéaire. Les paramètres de contrôle ont été le débit de carburant W_F , les valves d'extraction d'air (en anglais *Variable Bleed Valve* ou VBV) et l'aire du col de la tuyère de sortie (en anglais *Variable Area Fan Nozzle* ou VAFN). Les simulations ont démontré que le modèle obtenu était équivalent à celui d'un turboréacteur dynamique complètement opérationnel et que son utilisation serait recommandée pour de futures recherches en turbomachine.

La modélisation par Simulink facilite l'application du modèle sur un autre moteur. Il nécessite seulement de remplacer les valeurs numériques des paramètres de performances et des gains dans le modèle. Cependant, cette approche pourrait être coûteuse en termes de temps, étant donné qu'elle nécessite une vérification minutieuse de chaque composant du moteur. Cela dit, le modèle développé pourrait servir de référence pour de futures études dont notamment pour la prédiction de performances, pour le calcul des rendements des composants et du moteur, pour le diagnostic du moteur et pour l'évaluation du niveau de dégradation du moteur. Dans ce cas, le modèle mis en place peut être comparé à celui obtenu par des logiciels spécialisés en turbomachine, présentés dans la prochaine section.

1.12 Modélisation du moteur par les logiciels spécialisés

Certains logiciels permettent d'analyser le cycle thermodynamique du moteur comme Gasturb et Prooxis. Ils permettent entre autres de modéliser un turboréacteur dont ses paramètres sont calculés à partir des modèles génériques. D'autres programmes comme le C-MAPSS et son analogue C-MAPSS40K, développés par la *NASA Dryden Research Laboratory* permettent de modéliser les performances d'un turboréacteur ou encore de concevoir des boucles de contrôle (Csank et al., 2010). Le logiciel C-MAPSS40k permet de calculer les performances d'un moteur de la même catégorie, c'est-à-dire pour les turbosoufflantes avec une force de poussée équivalente à 40 000 lbf.

L'analyse des performances d'un turboréacteur peut également être réalisée grâce au code *Numerical Propulsion System Simulation* ou NPSS (Jones, S.M., 2007). La modélisation est basée sur le cycle de Brayton, et le moteur est divisé en trois composants principaux, dont le compresseur, la chambre de combustion et la turbine. Étant donné la complexité des interactions entre les composants du moteur il est nécessaire de recourir à des codes numériques afin de prédire les performances du moteur. Le NPSS est un programme orienté objet, permettant de modéliser l'architecture d'un moteur à réaction de manière très précise.

Le programme AEDsys développé par Mattingly en 2002 permet de réaliser différentes tâches comme l'analyse de contraintes, l'analyse de mission, l'analyse des performances du turboréacteur ou encore l'analyse des performances de chaque composant du moteur. Ce programme intègre 14 différents types de contraintes, 8 modèles de performance de composants, 7 polaires aérodynamiques, 7 modèles de poussées et 7 modèles de consommation de carburant.

Les logiciels présentés dans cette section prennent en compte la physique du moteur et peuvent être classés parmi les modèles de « boîte grise ». En effet, dans les modèles de « boîte grise », la structure générale du moteur est déjà connue et il ne reste plus qu'à identifier les paramètres de corrections et/ou les rendements η de chaque composant afin d'adapter le modèle générique au modèle existant. Étant donné que le simulateur de vol du CRJ700 est certifié de niveau D, on considèrera le moteur GE CF34-8C5B1 comme un moteur réel qui servira de référence. La validation à partir du simulateur est donc suffisante. Ainsi, la modélisation présentée dans ce mémoire ne requiert pas l'utilisation de logiciels spécialisés, cependant ces-derniers pourraient compléter ces recherches grâce à une validation plus poussée.

1.13 Modélisation du moteur par analyse en CFD

La modélisation par la mécanique des fluides (en anglais *Computational Fluid Dynamics* ou CFD) est une autre technique pour l'identification des systèmes.

La modélisation en CFD est particulièrement nécessaire dans l'identification d'un modèle d'écoulement, ou dans la simulation de l'écoulement autour d'un modèle que l'on doit identifier. Elle utilise principalement la méthode d'éléments finis (en anglais *Finite Element Method* ou FEM), qui consiste à discrétiser un volume en vue de déterminer les propriétés de l'écoulement dans un point fixe de ce volume.

Plusieurs logiciels d'analyse en CFD sont disponibles. On peut citer ANSYS Fluent qui est largement utilisé dans les domaines de l'ingénierie. Dépendamment du niveau de fidélité de la

modélisation, d'autres logiciels spécialisés en aéronautique comme Open FOAM et Open VSP permettent de simuler l'écoulement autour d'un profil d'aile, du fuselage ou encore de l'empennage, en vue de déterminer les coefficients aérodynamiques du modèle d'avion (Ségui et *al.*, 2018).

Dans le domaine de la turbomachine, l'analyse en CFD est particulièrement nécessaire pour modéliser l'écoulement au niveau des composants du turboréacteur comme les ailettes de la soufflante, les ailettes de la turbine et du compresseur (Dufour et *al.*, 2015). Cette technique permet notamment de simuler et de visualiser les paramètres physiques comme l'entropie. Cependant, la modélisation dans la chambre de combustion est plus complexe que les autres composants, étant donné que les réactions chimiques produites par la combustion sont à très petite échelle au niveau moléculaire. Un niveau de discrétisation élevé lors du maillage serait nécessaire pour la chambre de combustion.

La modélisation par CFD requiert également la connaissance de la géométrie du moteur et de ses composants, ainsi que celle des paramètres de pression, de température, du volume, du nombre de Mach, du nombre de Reynolds, et du type de gaz pour chaque point du maillage de type CFD (Dufour et *al.*, 2015). L'analyse en CFD permet notamment d'étudier le comportement du moteur lors des écoulements laminaires, turbulents, et lors des phénomènes de décrochage et de pompage. L'objectif est de valider un modèle d'écoulement, qui dépendra par la suite des paramètres géométriques du moteur et/ou des paramètres de chaque composant (ratios de pressions, ratios de température, rendement). Il est parfois nécessaire de revalider les simulations réalisées par *CFD* avec les simulations classiques.

Cette méthode permettrait également de prédire les propriétés de l'écoulement en utilisant les paramètres de pression, de température, de débit d'air/carburant en un point donné, afin de prédire les paramètres de performances comme la poussée, et le débit de carburant.

1.14 Conclusion de la revue de littérature

Dans ce chapitre, nous avons expliqué les grandes lignes de la turbomachine. On a notamment détaillé plusieurs techniques de modélisation et d'identification de systèmes.

Cette revue de littérature a particulièrement permis de détailler les technologies les plus récentes en matière d'identification de système, dont les réseaux de neurones, les algorithmes d'estimation et l'approche par composant, qui seront appliquées dans ce mémoire.

On a pu également prendre connaissance des travaux de recherches réalisés dans le domaine de la turbomachine, dont dans l'identification du système par composants destiné à l'analyse, au diagnostic, à la détection des erreurs, à la prédiction de performance et au contrôle du moteur.

CHAPITRE 2

MÉTHODOLOGIE ET IDENTIFICATION DU MODÈLE DE PERFORMANCE

Ce chapitre présentera la méthode de recherche appliquée dans ce mémoire. Dans un premier temps, une description complète de l'outil de recherche sera réalisée dans la section 2.1. On décrira d'abord au lecteur l'avion de référence qui est le CRJ700, ensuite on présentera le moteur, et enfin on présentera le simulateur de vol. La réalisation des tests en vol sera présentée par la suite dans la section 2.2 et le modèle de performance sera identifié dans la section 2.3.

2.1 Présentation de l'outil de recherche

Cette section présente une description de l'avion de référence, de son moteur et du simulateur de vol utilisés pour l'identification du modèle de performance. Il s'agit du jet régional CRJ700 conçu par Bombardier Aerospace¹. Son premier vol s'est effectué le 27 mai 1999 et l'avion peut accueillir près de 80 passagers. Le CRJ700 est propulsé par deux turbosoufflantes General Electric CF34-8C5B1.

2.1.1 Description du CRJ700

Le CRJ700 possède un empennage avec une configuration en « T » et il est capable d'opérer à des altitudes atteignant les 41 000 pi et au nombre de Mach maximal de 0.83. Les dernières générations du CRJ700 sont équipées de deux moteurs General Electric CF34 – 8C5B1. Grâce à ces moteurs, la distance de vol maximale de l'avion peut atteindre jusqu'à 2 900 miles. Le moteur consiste à un turboréacteur à double flux, monté sur chaque côté du fuselage comme le montre la Figure 2.1. Chaque moteur est capable de produire une poussée maximale de 13 790 lbf pour des conditions de température standard de 15°C au niveau de la mer.

¹ Le groupe a été acquis par le groupe Mitsubishi Aviation Group en 2020

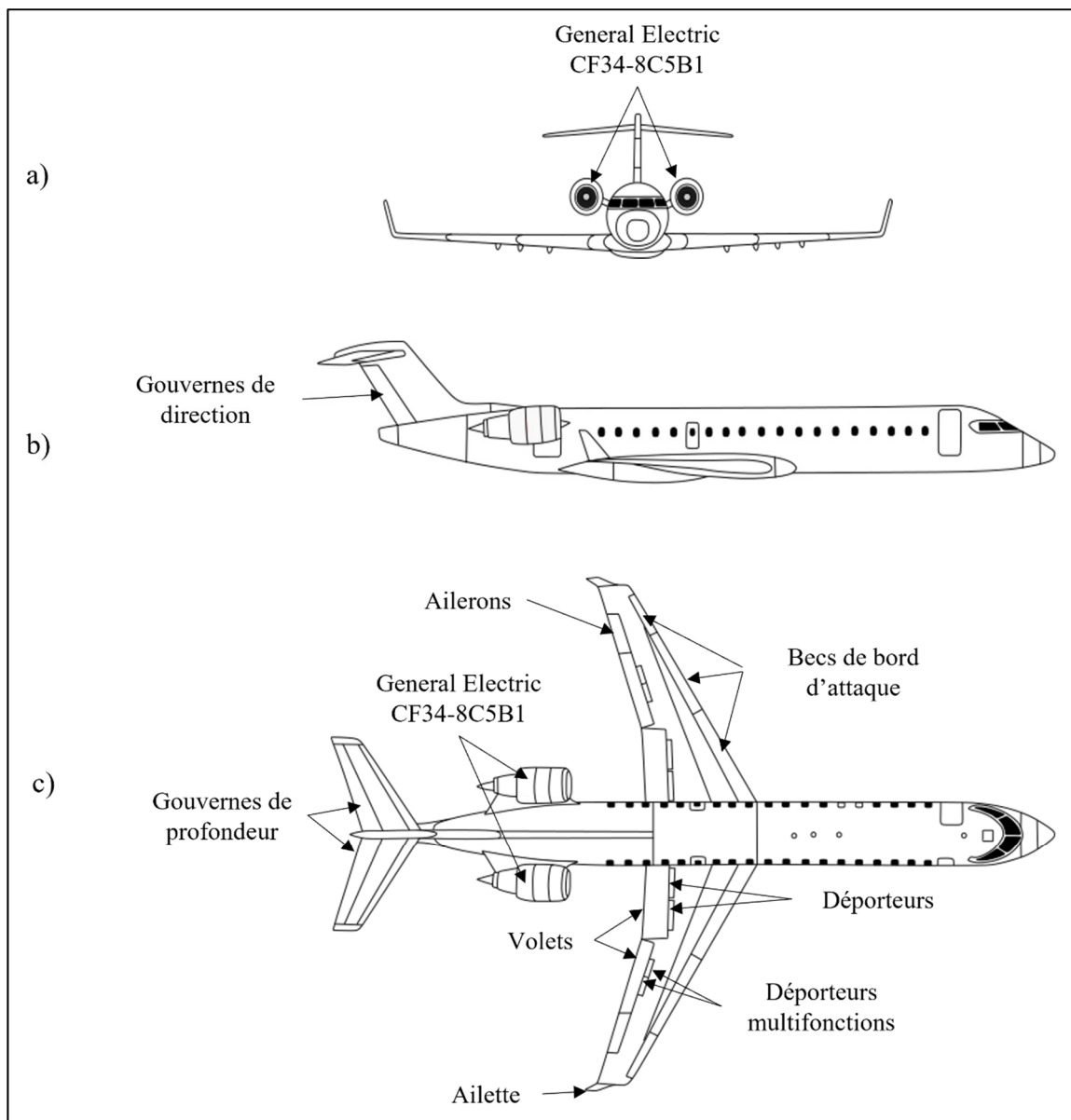


Figure 2.1 Surfaces de contrôle du CRJ700

Les éléments de contrôle du CRJ700 sont de type conventionnel avec une manche de contrôle monté sur une colonne, ainsi qu'avec des palonniers pour la commande des gouvernes de direction comme l'illustre la photo du simulateur de vol sur la Figure 2.4.

Le contrôle du mouvement longitudinal de l'avion est assuré par les gouvernes de profondeurs, fixées sur un stabilisateur horizontal amovible. Le mouvement de tangage est actionné lorsque le pilote tire ou pousse la colonne. Le contrôle du mouvement latéral de l'avion est assuré par les ailerons, qui sont assistés par les déporteurs multifonctionnels. Le mouvement de roulis est actionné lorsque le pilote tourne le manche vers la gauche ou vers la droite. Enfin, le mouvement directionnel de l'avion est assuré par les gouvernes de direction assistées par les amortisseurs de lacet. Le mouvement de lacet est amorcé lorsque le pilote appuie sur les palonniers.

Les surfaces de contrôles sont actionnées à l'aide d'un réseau de câbles, de torseurs et de leviers liés à un système hydraulique ou à un système électrique (de type *Fly-by-wire*). Le système de contrôle de chaque surface de contrôle est séparé en deux systèmes identiques: un système de gauche et un système de droite. Le système gauche est contrôlé par le pilote, tandis que le système de droite est contrôlé par le copilote. Lors des conditions normales de vol, les deux systèmes sont interconnectés entre eux à l'aide d'un mécanisme, et fonctionnent d'une manière simultanée. Si une anomalie survient sur une surface de contrôle, les systèmes gauche et droit peuvent être séparés, et ainsi la surface de contrôle fonctionnel servira à contrôler le mouvement de l'avion.

Les déporteurs permettent de réduire la portance de l'avion grâce à un déploiement symétrique. Les déporteurs multifonctionnels permettent en plus d'assister les ailerons lors de virages coordonnés lorsqu'ils sont déployés de façon asymétrique. Ils assistent le contrôle du roulis et stabilisent l'avion sur une trajectoire fixe lors de sa descente et de son atterrissage. Les déporteurs multifonctionnels permettent notamment d'améliorer le contrôle latéral de l'avion lors des vols à basse vitesse.

Les volets ainsi que les becs de bord d'attaque augmentent la portance maximale de l'avion lors des phases de décollage et d'atterrissage. Chaque section de l'aile est munie de trois becs de bord d'attaque et de deux volets. Les deux systèmes sont actionnés par un système électrique. Lorsqu'ils sont actionnés, les becs de bord d'attaque sont défléchis vers l'avant et

vers le bas, tandis que les volets se déplacent légèrement vers l'arrière et vers le bas grâce à un pivot.

Un système de prévention du décrochage est activé lorsque l'angle d'attaque de l'avion se rapproche de sa limite maximale. Il s'agit d'alertes visuelles qui apparaissent sur l'écran d'affichage principal (appelé EICAS), et des signaux sonores qui avertissent l'équipage. L'autopilote est alors désengagé et la commande devient manuelle. Si l'angle d'attaque continue de se rapprocher de l'angle d'attaque critique de décrochage, un mécanisme pousse automatiquement la colonne vers l'avant pour piquer le nez de l'avion.

2.1.2 Description du turboréacteur CF34-8C5B1

Comme dit précédemment, le CRJ700 est propulsé par deux moteurs General Electric CF34-8C5B1, qui consistent à un turboréacteur à double flux comme le montre la Figure 2.2. Le principe de propulsion est détaillé dans la section 1.7.1.

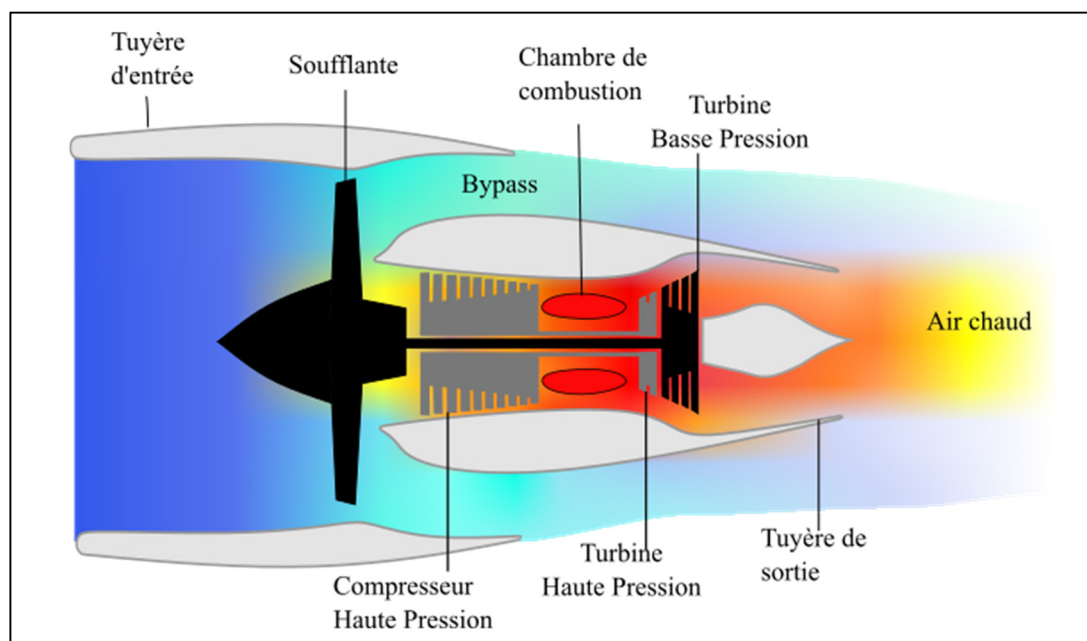


Figure 2.2 Schéma de la turbosoufflante General Electric CF34-8C5B1

Le moteur General Electric CF34-8C5B1 est un turboréacteur à double flux de type conventionnel : l'air de l'entrée du moteur est accéléré à travers une soufflante N_1 . Le flux d'air est ensuite séparé en deux : 1) le flux d'air principal, ou air chaud, correspond à l'air qui entre dans le corps principal du moteur (en jaune et rouge). 2) le flux d'air secondaire, aussi connu par « poussée froide », correspond à l'air sur les contours du moteur (en bleu). La poussée froide génère près de 80 % de la poussée totale. Le cycle thermodynamique complet est détaillé dans la section 1.7.3.

Le moteur est muni d'un système de refroidissement qui consiste à un système d'air secondaire. Il permet de refroidir les composants, dont les ailettes de la turbine, les différents compartiments du moteur et les autres éléments du moteur. L'air comprimé est également utilisé pour pressuriser le système de lubrification du moteur.

Le contrôle du moteur est effectué par le FADEC (voir section 1.10) qui contrôle le système de géométrie variable, ainsi que les valves d'extraction d'air et les valves de contrôle du carburant. Le système de Géométrie Variable (en anglais *Variable Geometry* ou VG) est un système d'actuation qui contrôle le débit d'air entrant dans le compresseur. Il permet notamment de changer la position de la manche d'entrée du compresseur et la position des stators sur les quatre premiers étages du compresseur. Cette opération permet d'optimiser l'angle d'attaque de chaque ailette. Le FADEC permet également d'évaluer les performances de chaque composant et de placer des barrières de protections afin d'éviter les phénomènes de décrochage ou de pompage. Le Tableau 2.1 indique les valeurs maximales des paramètres du moteur CF34-8C5B1, dont celles de la vitesse de rotation de la soufflante N_1 , la température à l'entrée de la turbine ITT et la vitesse de rotation de la turbine N_2 pour chaque mode de poussée. Les modes de poussée sont expliqués dans la prochaine section.

Tableau 2.1 Limites maximales des paramètres pour chaque mode de poussée

Paramètres du moteur	Mode IDLE	Mode MCLB	Mode Régime de croisière	Mode MTO
N_1	20 – 25 %	99.5%	99.5%	99.5%
ITT	-	960°C	-	964°C pour 2 min
N_2	55 – 65 %	98%	98%	99.4%

Contrôle du moteur

Les éléments de contrôle de la turbosoufflante sont la manette des gaz (en anglais *Throttle Lever Angle* ou TLA) et la réserve automatique. Le moteur est contrôlé par le FADEC, qui répond aux commandes du pilote *via* le TLA. Le FADEC envoie ensuite les signaux de commande vers l'unité de contrôle du carburant (en anglais *Fuel Metering Unit*), qui contrôle le débit de carburant injecté dans la chambre de combustion. L'information sur l'état du moteur est affichée au pilote sur l'écran affichage principal EICAS. Il s'agit principalement de la vitesse de rotation de la soufflante N_1 , de la vitesse de rotation de la turbine N_2 , de la température à l'entrée de la turbine ITT, et du niveau de vibration pour chaque moteur.

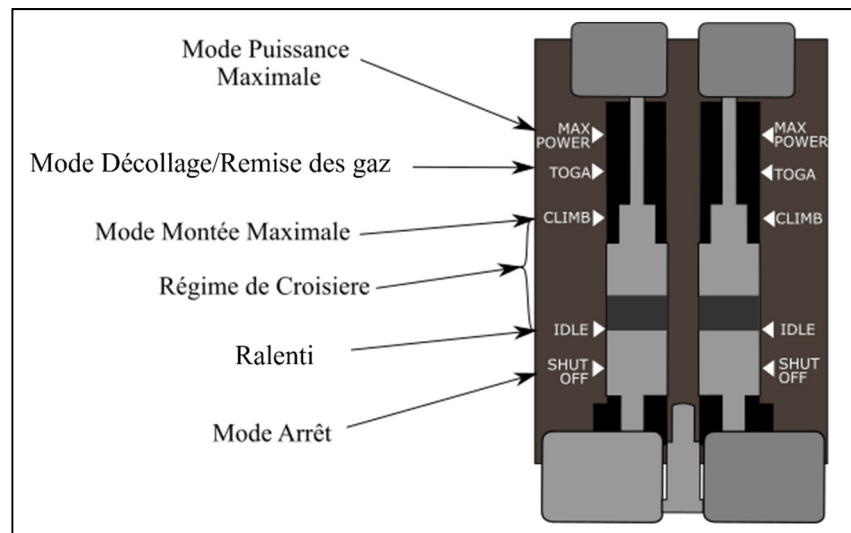


Figure 2.3 Modes de poussée de la manette des gaz

Comme illustré sur la Figure 2.3, les différentes positions de la manette des gaz sont : le mode « arrêt » ou SHUTOFF, le mode « poussée minimale » ou IDLE, le mode « montée maximale » ou MCLB, le mode « décollage/2nd tour » ou TOGA (ou MTO), et le mode « puissance maximale » ou MAX POWER. Comme il a été décrit dans la section 1.7.6, chaque position correspond à un niveau de poussée bien déterminé, et chaque niveau de poussée est utilisé pour une phase de vol spécifique.

Tableau 2.2 Positions de la manette des gaz en fonction du mode de poussée

Mode de poussée	TLA en °
IDLE	0
Régime de croisière	0-45
MCLB	45
TOGA ou MTO	51
MAX POWER	100

Ainsi, les modes IDLE, CLIMB et MTO sont utilisés respectivement lors des phases de descente, de montée et de décollage. La phase de croisière est la seule phase où la manette des gaz est constamment ajustée entre les positions IDLE et MCLB. Enfin, le mode MAX POWER est utilisé par le système de réserve automatique (en anglais *Automatic Performance Reserve* ou APR) en cas de panne d'un moteur lors du décollage ou pour arrêter l'avion grâce à la reverse de poussée lors du roulage au sol. Le Tableau 2.2 indique la valeur de TLA en ° correspondant à chaque mode de poussée.

2.1.3 Présentation du simulateur de vol

Cette section présente le simulateur de vol du CRJ700 qui constituera l'outil de recherche utilisé dans ce mémoire. Le simulateur de vol du CRJ700 présenté sur la Figure 2.4 a été développé par la compagnie CAE, inc. en collaboration avec Bombardier, et il est disponible au local A-3420 de l'ÉTS. Il a été acquis par le Professeur Botez pour le laboratoire du

LARCASE par l'intermédiaire des fonds CFI et MDEIE et il est dédié à la recherche. Le simulateur de vol possède une certification de niveau D, qui est la plus haute qualification pour la dynamique du vol et la modélisation des moteurs selon le *Federal Aviation Administration*. Ainsi, on peut considérer que les données collectées du simulateur de recherche sont précises et représentent les performances réelles des turboréacteurs General Electric CF34-8C5B1.



Figure 2.4 Photo du simulateur de vol du CRJ700

L'interface du simulateur comporte trois écrans d'affichage, dont un écran principal (en face du pilote) et deux écrans auxiliaires (sur la gauche du pilote). L'écran principal affiche la cabine du pilote, ainsi que le tableau de bord. Il est possible d'accéder aux différents instruments de vol et aux différents systèmes de contrôle de l'avion. Le deuxième écran (en haut à gauche) peut être personnalisé par le pilote. Par défaut, il affiche les instruments de vol principaux, dont l'altimètre, le compas, le variomètre, l'horizon artificiel, l'écran d'affichage EICAS, le système de gestion de vol (en anglais *Flight Management System* ou FMS), la manette des gaz, la position des volets, la position des trains d'atterrissage et l'autopilote. Le troisième écran (en bas à gauche) permet de visualiser l'état des différents systèmes en temps réel comme le système de carburant, la pneumatique, le système d'huile, la température du moteur, *etc.*

Comme décrit précédemment dans la section 2.1.1 et illustré sur la Figure 2.4, les éléments de contrôle du CRJ700 sont la colonne pour le contrôle des gouvernes de profondeur, le manche fixé sur la colonne pour le contrôle des ailerons et les palonniers pour le contrôle des gouvernes de direction. Le manche est également équipé d'un bouton pour désactiver l'autopilote. Les autres commandes sont accessibles à l'écran comme la manette des gaz, la position des volets et la position des trains d'atterrissage.

Le simulateur de vol du CRJ700 permet de réaliser diverses missions de vol. Pour ce faire, il possède différentes fonctions permettant de modifier l'environnement de vol de l'avion comme la température ambiante, les conditions de vent (vitesse, directions) ou encore les turbulences. Ce simulateur permet de paramétrer le poids de l'avion et la quantité initiale de carburant. Il permet également de simuler les défaillances comme les anomalies des surfaces de contrôle et les pannes du moteur.

Le langage de communication utilisé est le CTS qui est un langage propre à la compagnie CAE. Un guide d'utilisation du simulateur et un guide d'introduction au langage CTS sont mis à la disposition de l'utilisateur. L'utilisateur peut entrer les commandes de deux façons : 1) soit en utilisant les fonctions intégrées à la barre des tâches qui est affichée sur l'interface du simulateur, 2) soit en entrant les lignes de codes programmées en langage CTS via la console de commande prévue à cet effet.

La collecte de données est réalisée grâce à la console de commande. Pour ce faire l'utilisateur doit définir les variables qu'il souhaite obtenir avant d'entrer la commande de collecte. Une fois que la collecte est terminée, les données sont stockées dans un fichier sous le format *CSV, qui peut être ensuite exporté sur Matlab pour l'étude.

2.2 Réalisation des tests en vol

L'objectif de cette section consiste à mettre en place une base de données nécessaire à l'identification du modèle de performance. La base de données a été établie à partir des tests en vol effectués sur le simulateur de vol du CRJ700 présenté dans la section précédente.

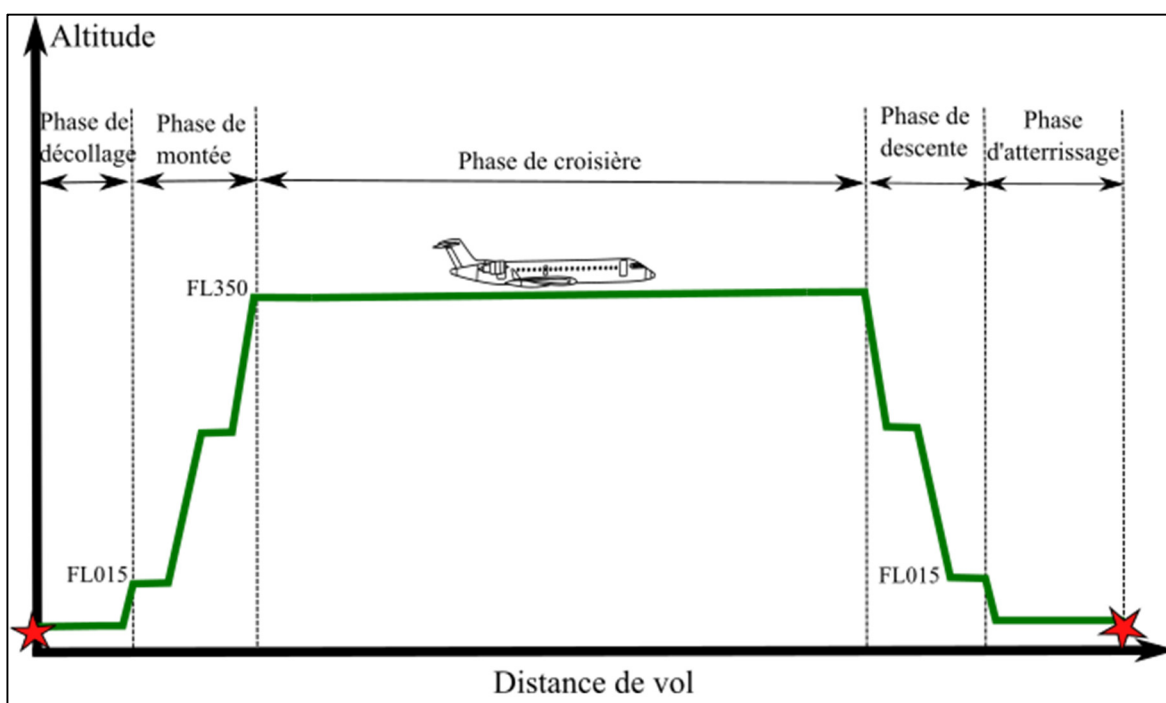


Figure 2.5 Profil d'un vol complet

Quatre phases de vol ont été étudiées afin de modéliser les performances du moteur sur toute l'enveloppe de vol de l'avion. La réalisation des tests pour chaque phase de vol est détaillée dans les sections suivantes et la base de données complète est décrite dans la section 2.2.5.

2.2.1 Phase de décollage

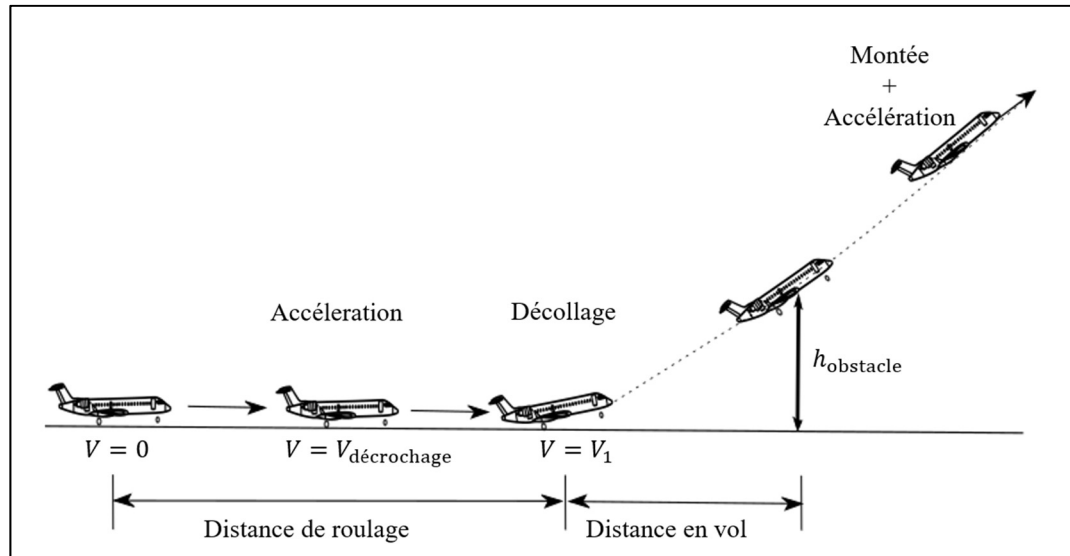


Figure 2.6 Schéma des phases de décollage et de montée

Les premières catégories de vols sont les tests de décollage, qui correspondent au premier segment d'un profil de vol complet, comme le montre la Figure 2.5. La distance totale de décollage est la somme de la distance de roulage avec la distance parcourue en vol comme le montre la Figure 2.6. En général, la distance de décollage doit être la plus courte possible pour une opération rapide et économique. Cependant, certains avions décollent à puissance réduite sur une plus longue distance afin de réduire l'usure des moteurs.

La phase de décollage est réalisée en 3 étapes : 1) Le pilote place la manette des gaz sur la position « Décollage maximal », en anglais *Maximum Takeoff* ou MTO. L'avion accélère alors jusqu'à atteindre la vitesse de décollage V_1 . 2) Le pilote tire ensuite la colonne pour décoller le nez de l'avion, augmentant l'angle d'attaque pour générer la portance. 3) Le pilote rentre enfin les trains d'atterrissage pour réduire la traînée une fois que l'altitude de sécurité $h_{obstacle}$ soit atteinte. L'avion continue enfin sa trajectoire en augmentant sa vitesse verticale et en accélérant jusqu'à une altitude de 1 500 pi.

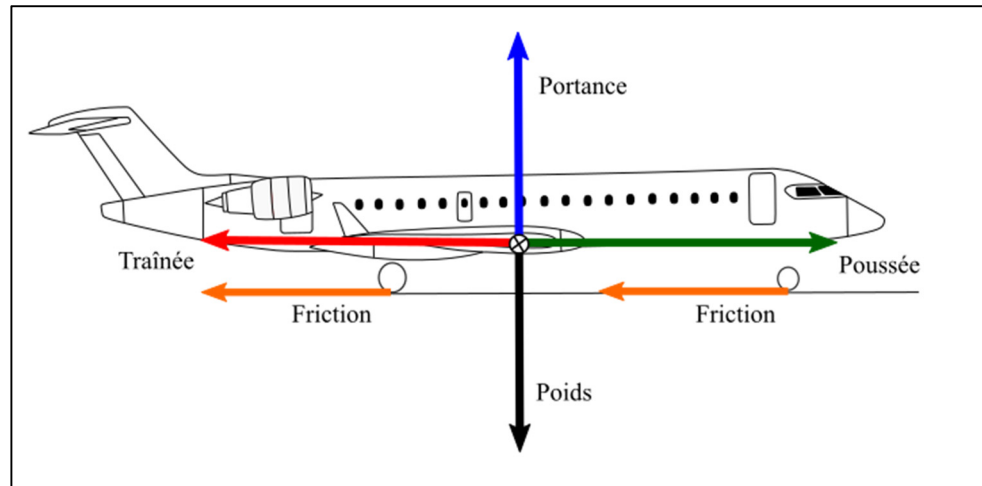


Figure 2.7 Forces agissant sur l'avion lors du roulage au sol

On ne développera pas en détails les équations du mouvement, étant donné que ce n'est pas l'objectif de ce mémoire. On donnera juste au lecteur une idée de la physique qui décrit le mouvement de l'avion lors des différentes phases de vol.

Le mouvement de l'avion peut être décrit par quatre forces dont la Portance L , le Poids W , la Traînée D et la Poussée T , auxquelles s'ajoutent les forces de friction lors des phases de roulage au sol. On considère que l'avion est un corps matériel rigide et que les forces sont appliquées dans son centre de gravité. La Figure 2.7 illustre les forces qui agissent sur l'avion lors de la phase d'accélération au sol. La résultante F_a de toutes ces forces est alors définie comme suit :

$$F_a = T - D - \mu(W - L) \quad (2.1)$$

où μ est le coefficient de friction entre les roues et la piste et il est généralement compris entre 0.02 et 0.05.

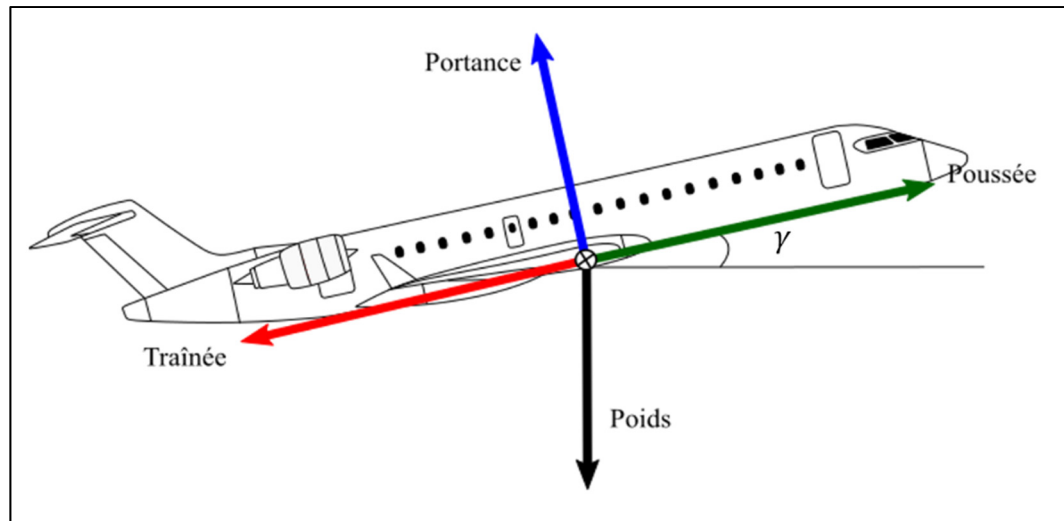


Figure 2.8 Forces agissant sur l'avion lors de la montée

La Figure 2.8 illustre les forces qui agissent sur l'avion lorsqu'il est en vol lors de la phase de montée. Durant la phase en vol du décollage, l'avion est en accélération. Son mouvement peut alors être décrit par les équations suivantes :

$$\begin{aligned}
 T - D - W \sin \gamma &= \frac{W}{g} \frac{dV}{dt} \\
 L - W \cos \gamma &= \frac{W}{g} V \frac{d\gamma}{dt}
 \end{aligned}
 \tag{2.2}$$

où γ est l'angle de pente et V est la vitesse réelle de l'avion (en anglais *True Air Speed* ou *TAS*). Le poids W correspond au produit de la masse avec l'accélération g défini comme suit :

$$W = mg \tag{2.3}$$

où m est la masse de l'avion et $g = 32.17 \text{ pi/s}^2$ (9.81 m/s^2) est l'accélération gravitationnelle. La portance L et la traînée D sont les forces aérodynamiques et peuvent être calculées comme suit :

$$\begin{aligned} L &= C_L q S \\ D &= C_D q S \\ q &= \frac{1}{2} \rho V^2 \end{aligned} \tag{2.4}$$

où C_L et C_D sont respectivement les coefficients de portance et de traînée, S est la surface de référence, q est la pression dynamique et ρ est la densité de l'air.

Un poids initial entre 50 000 lb et 70 000 lb a été choisi pour tous les tests de décollage. Toutefois, afin de déterminer l'impact de l'altitude-pression sur les performances du moteur, les tests ont été effectués à partir de différents aéroports avec des altitudes de départ entre 28 et 5 500 pi au-dessus de la mer.

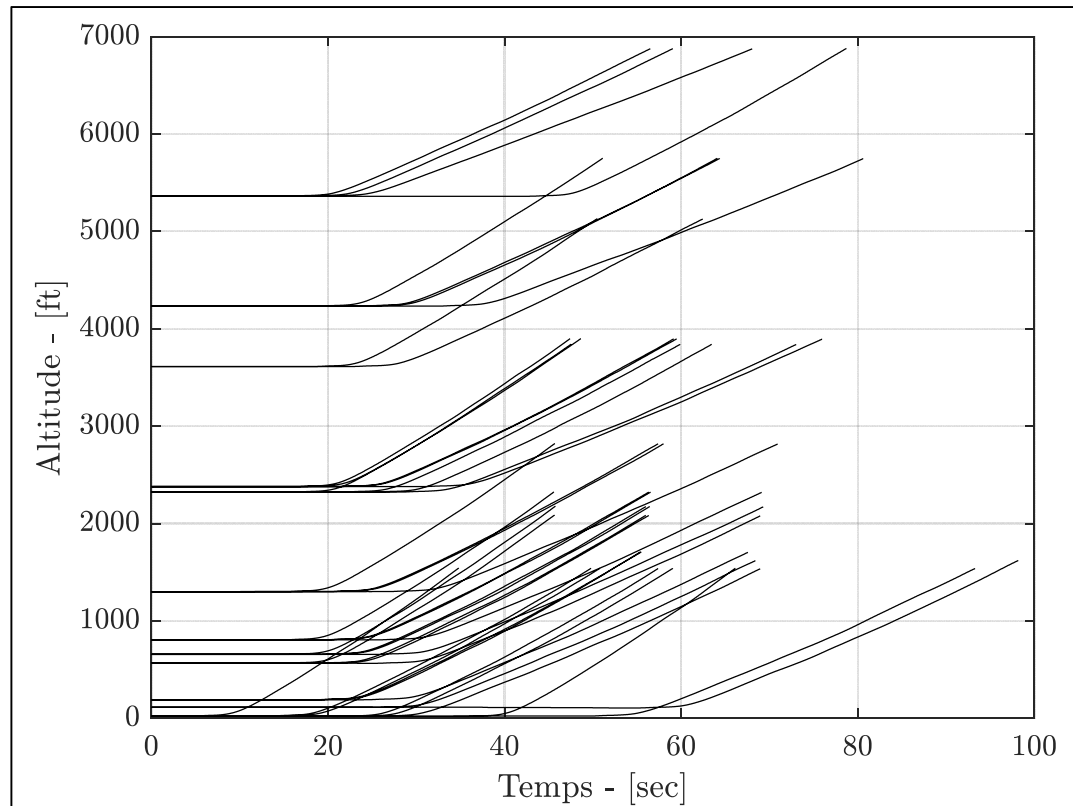


Figure 2.9 Ensemble des profils de décollage

Pour chaque test, les paramètres du moteur et les données de vol comme l'altitude, la vitesse, la température ambiante et la pression ambiante ont été collectés à partir du simulateur avec une fréquence de $\frac{1}{4}$ Hz (soit 1 donnée toutes les 4 secondes). Il faudrait noter que les données sont collectées à partir de 5 secondes après que le pilote ait placé la manette des gaz sur la position *Maximum TakeOff* ou MTO. Ce temps d'attente a été imposé afin de stabiliser le moteur et d'éviter son comportement dynamique au début de la phase d'accélération. En effet, étant donné que seul le modèle de performance statique nous intéresse, la dynamique du moteur en régime transitoire n'est pas prise en compte. Finalement, la collecte de données est arrêtée une fois que l'avion atteint une altitude de 1 500 pi au-dessus du sol. La Figure 2.9 illustre l'ensemble des tests effectués pour la phase de décollage : 48 tests ont été réalisés au total, dont 6 ont été sélectionnés pour l'identification du modèle, tandis que 42 ont été sélectionnés pour la validation du modèle.

Dans notre cas, le pilote a été remplacé par un pilote automatique afin de réaliser des scénarios de décollage uniformes. Le nombre de décollage effectué est arbitraire. Il dépend des différentes combinaisons possibles entre vitesse de décollage et altitude de l'aéroport afin de varier la base de données. Les profils de vol qui n'ont pas réussis ont ensuite été éliminés de afin de ne pas biaiser la base de données.

2.2.2 Phase de montée

La phase de montée correspond à la phase qui suit la phase de décollage comme schématisé sur la Figure 2.5. Les forces agissant sur l'avion sont illustrées sur la Figure 2.8. La manette des gaz est placée sur la position « montée maximale », en anglais *Maximum Climb* ou MCLB, où les trains d'atterrissage, les becs de bord d'attaque et les volets sont complètement rentrés. Une fois sur la position MCLB, l'avion réagit selon deux possibilités : 1) soit l'avion reste à la même altitude et accélère, 2) soit l'avion acquiert de l'altitude avec une vitesse constante, et une accélération nulle. La phase de montée correspond au second cas, dans lequel le mouvement de l'avion est uniforme et peut être décrit par les équations suivantes :

$$\begin{aligned} L - W \cos \gamma &= 0 \\ T - D - W \sin \gamma &= 0 \end{aligned} \tag{2.5}$$

où γ est l'angle de pente, V est la vitesse réelle de l'avion, L est la portance, T est la poussée, W est le poids et D est la traînée.

En général la montée est réalisée en quelques étapes avec des pauses qu'on appelle des « paliers », comme le montre la Figure 2.5. Dans le cadre de cette recherche, les tests de montée sont réalisés d'une façon continue avec une vitesse calibrée constante (en anglais *Calibrated Airspeed* ou CAS) et un nombre de Mach M initial. Un poids initial entre 50 000 lb et 70 000 lb a été choisi pour tous les tests de montée. Afin d'obtenir un nombre de données suffisant

couvrant toute l'enveloppe de vol de l'avion, chaque test de montée a été effectué avec différentes combinaisons de vitesses CAS et de nombre de Mach. Ces vitesses ont été choisies en variant le CAS entre 180 et 320 kts, et le nombre de Mach entre 0.30 et 0.82. Étant donné que les avions ne sont pas autorisés à voler au-dessus de 250 kts à une altitude inférieure à 10 000 pi, une accélération a été effectuée à 10 000 pi pour tous les scénarios de montées. Aussi, les tests de montées ont été réalisés sur deux intervalles d'altitudes. Les premiers tests ont été réalisés pour une altitude variant de 1 500 pi à 35 000 pi, tandis que les seconds tests ont été réalisés pour une altitude variant de 25 000 pi à 40 000 pi. Le choix de ces intervalles d'altitude a été défini grâce aux précédentes recherches effectuées par Bardela et Botez en 2019, et qui ont démontrées que la dynamique du moteur est différente en haute et basse altitude.

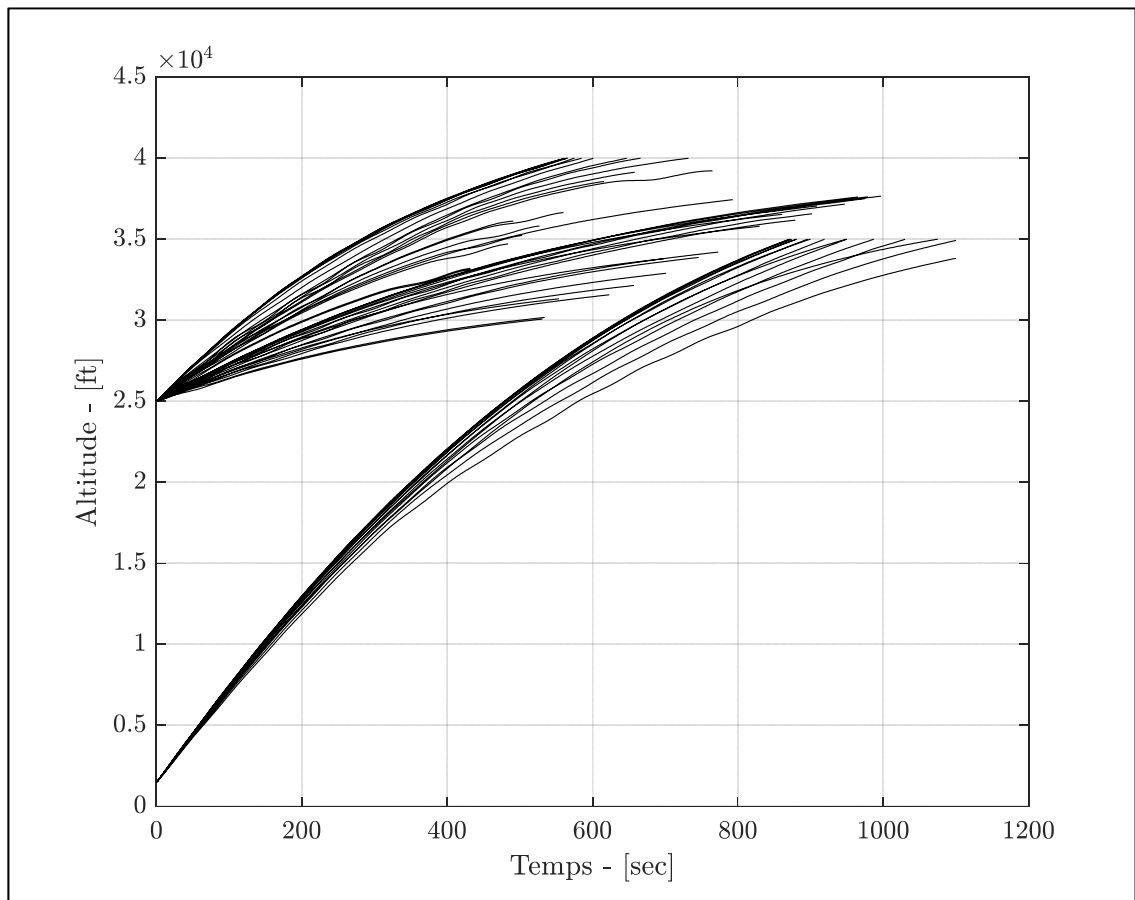


Figure 2.10 Ensemble des profils de montée

Finalement, pour tous les tests de montées, les données ont été collectées avec une fréquence de 1 Hz (soit 1 donnée par seconde). La collecte de données est arrêtée lorsque l'avion atteint l'altitude finale ou lorsque la vitesse verticale de l'avion devient nulle ou négative (dans ce cas, l'avion n'est plus en montée). Les données ont ensuite été réarrangées de sorte qu'on ait une donnée à chaque 1 000 pi, afin de réduire la taille de la base de données. La Figure 2.10 illustre l'ensemble des tests effectués pour la phase de montée. Un nombre de 55 tests a été réalisé au total, dont 8 ont été sélectionnés pour l'identification du modèle, tandis que 47 ont été sélectionnés pour sa validation.

2.2.3 Phase de croisière

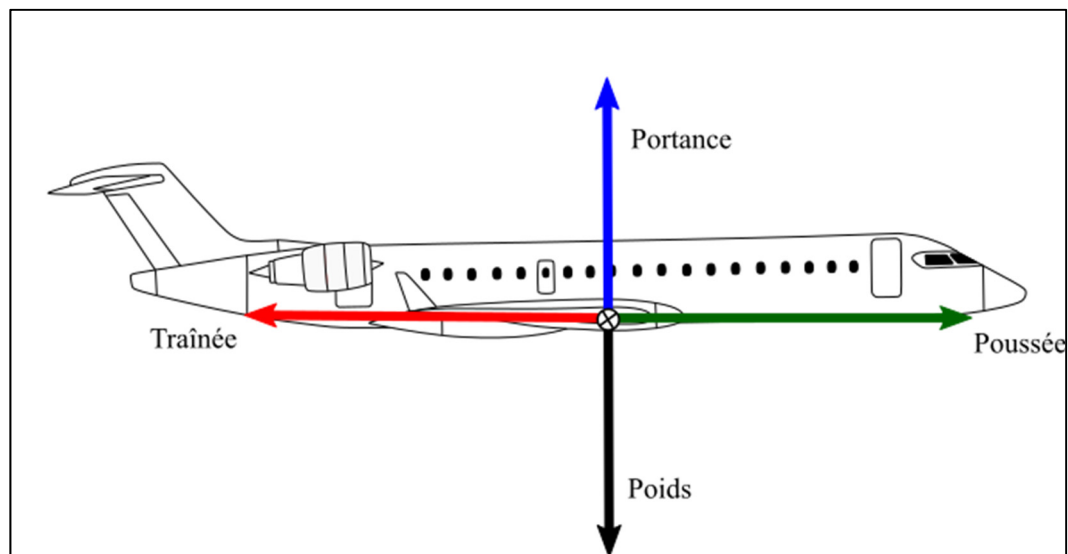


Figure 2.11 Forces agissant sur l'avion lors de la croisière

La phase de croisière correspond à la phase qui suit la phase de montée. C'est la phase la plus longue lors du vol de l'avion. Les forces agissant sur l'avion sont illustrées sur la Figure 2.11.

L'altitude est constante et la pente est nulle, soit $\dot{h} = 0$ et $\gamma = 0$. Le mouvement de l'avion est uniforme et il peut être décrit par les équations suivantes :

$$\begin{aligned} L - W &= 0 \\ T - D &= 0 \end{aligned} \quad (2.6)$$

Si la poussée du moteur est supérieure à la poussée nécessaire pour son mouvement rectiligne uniforme, alors l'avion se trouve en régime de montée ou en accélération. Si la poussée du moteur est inférieure à la poussée nécessaire pour son mouvement rectiligne uniforme, alors l'avion se trouve en régime de descente ou en perte de vitesse.

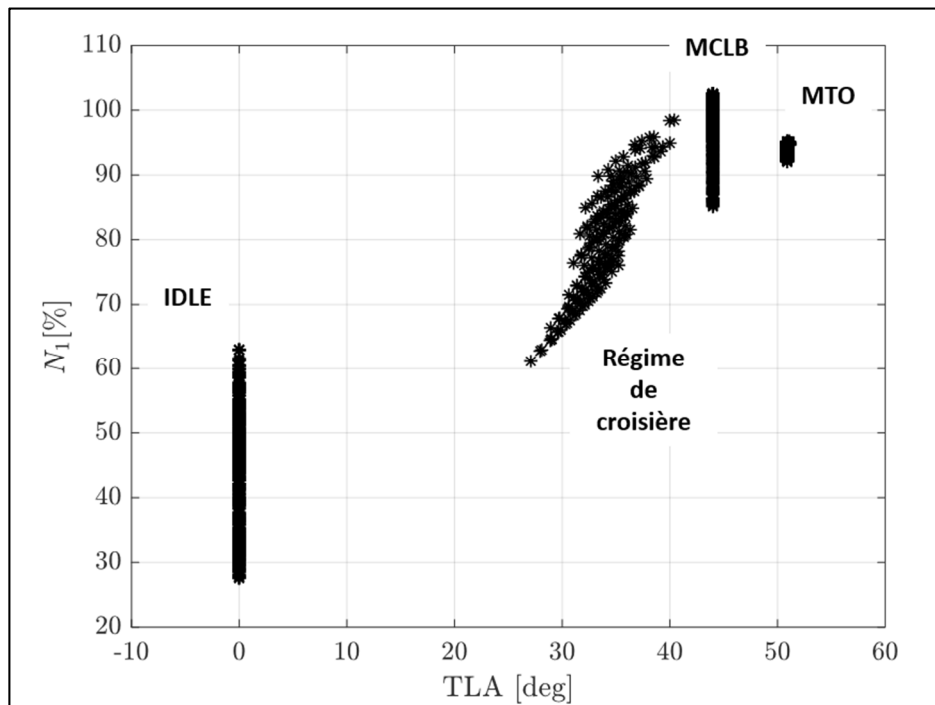


Figure 2.12 Variation du TLA en fonction du mode de poussée

Les tests de croisière sont généralement statiques, avec un nombre de Mach constant et une altitude constante. La position de la manette des gaz varie en mode continu entre les positions MCLB et IDLE comme illustré sur la Figure 2.12. Cependant, il est possible de modéliser les niveaux de poussée, en anglais *thrust ratings*, qui correspondent à des positions spécifiques du TLA. En effet, sur la Figure 2.12, on remarque que la position du TLA et la vitesse de rotation de la soufflante N_1 ont une corrélation linéaire pour chaque altitude. Ainsi, les tests de croisière ont été effectués pour des altitudes fixes entre 5000 pi et 40 000 pi, et des nombres de Mach fixes entre 0.54 et 0.84. Pour ce faire, l'autopilote a été activé pour garder une altitude constante et un nombre de Mach constant. Deux poids entre 50 000 lb et 70 000 lb ont été initialisés pour chaque altitude.

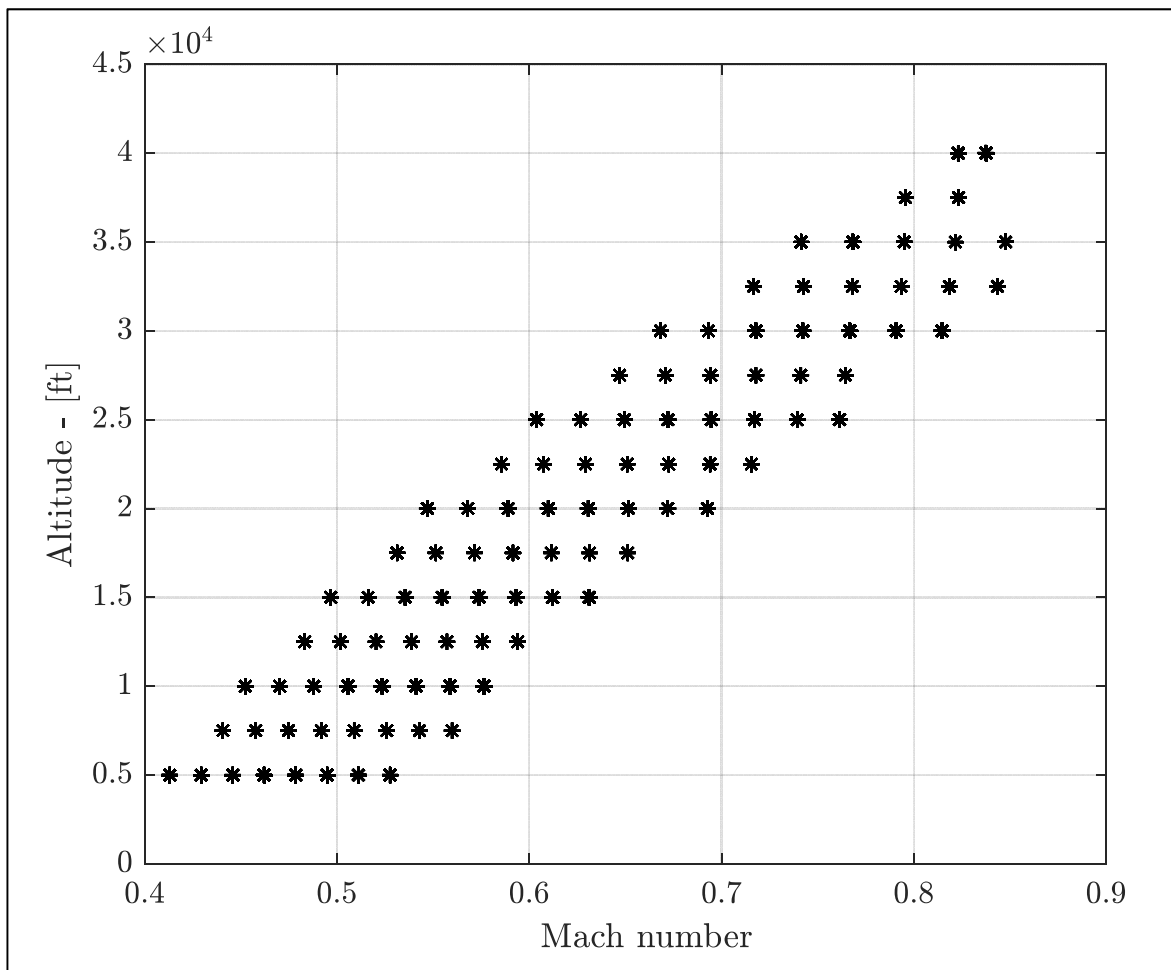


Figure 2.13 Ensemble des tests de croisière

Une fois que l'avion a été trimé sur une position spécifique (altitude et nombre de Mach), les paramètres du moteur ainsi que les données de vol ont été collectés pour une durée de 60 secondes, avec une fréquence de 1/10 Hz (soit 1 donnée toutes les 10 secondes). La moyenne de chaque paramètre a ensuite été calculée pour chaque test.

La Figure 2.13 illustre l'ensemble des tests statiques effectués pour la phase de croisière. Chaque point correspond à un test de 60 secondes. Au total 193 tests ont été réalisés, dont 39 ont été sélectionnés pour l'identification du modèle, tandis que 154 ont été sélectionnés pour sa validation.

2.2.4 Phase de descente

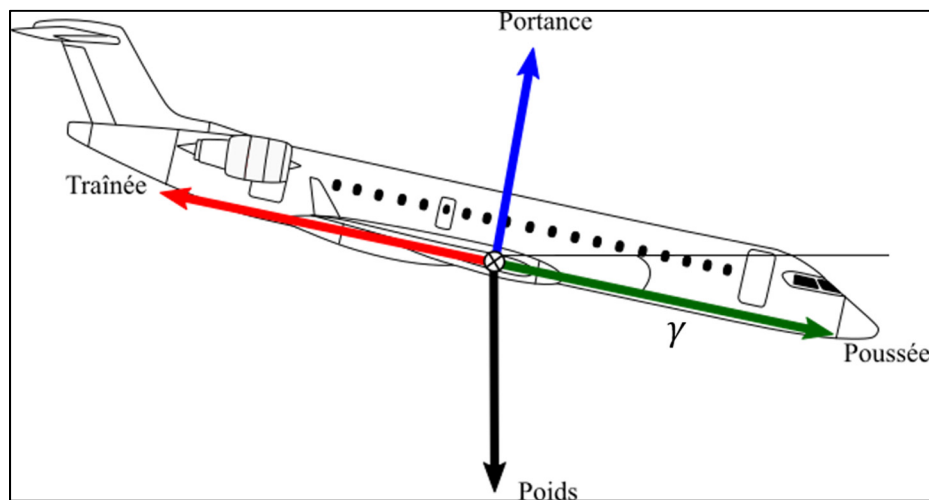


Figure 2.14 Forces agissant sur l'avion lors de la descente

La phase de descente correspond à la phase qui suit la phase de croisière. La phase de descente est parfois assimilée à un vol plané lors de laquelle la poussée est minimale, voire nulle. La manette des gaz est placée sur la position « poussée minimale », en anglais noté par IDLE. Une

fois sur la position IDLE, l'avion réagit selon deux cas :1) l'avion reste à la même altitude et décélère, 2) ou l'avion perd de l'altitude avec une vitesse constante, et une accélération nulle. La phase de descente correspond au second cas, où le mouvement de l'avion est uniforme et peut être décrit par les équations suivantes :

$$\begin{aligned} D + W \sin \gamma &= 0 \\ L - W \cos \gamma &= 0 \end{aligned} \tag{2.7}$$

La phase de descente est similaire à la phase de montée avec une pente négative, soit $\gamma < 0$. La poussée T est également négligeable par rapport aux autres forces aérodynamiques.

De la même manière que pour la phase de montée, la phase de descente est réalisée en « paliers », comme le montre la Figure 2.5. Dans le cadre de cette recherche, les tests de descente ont été réalisés d'une façon continue. Afin d'obtenir un nombre de données suffisant couvrant toute l'enveloppe de vol de l'avion, chaque test de descente a été effectué avec différentes combinaisons de vitesses CAS et de nombres de Mach. Ces vitesses ont été choisies en variant le CAS entre 180 et 320 kts, et le nombre de Mach entre 0.55 et 0.82.

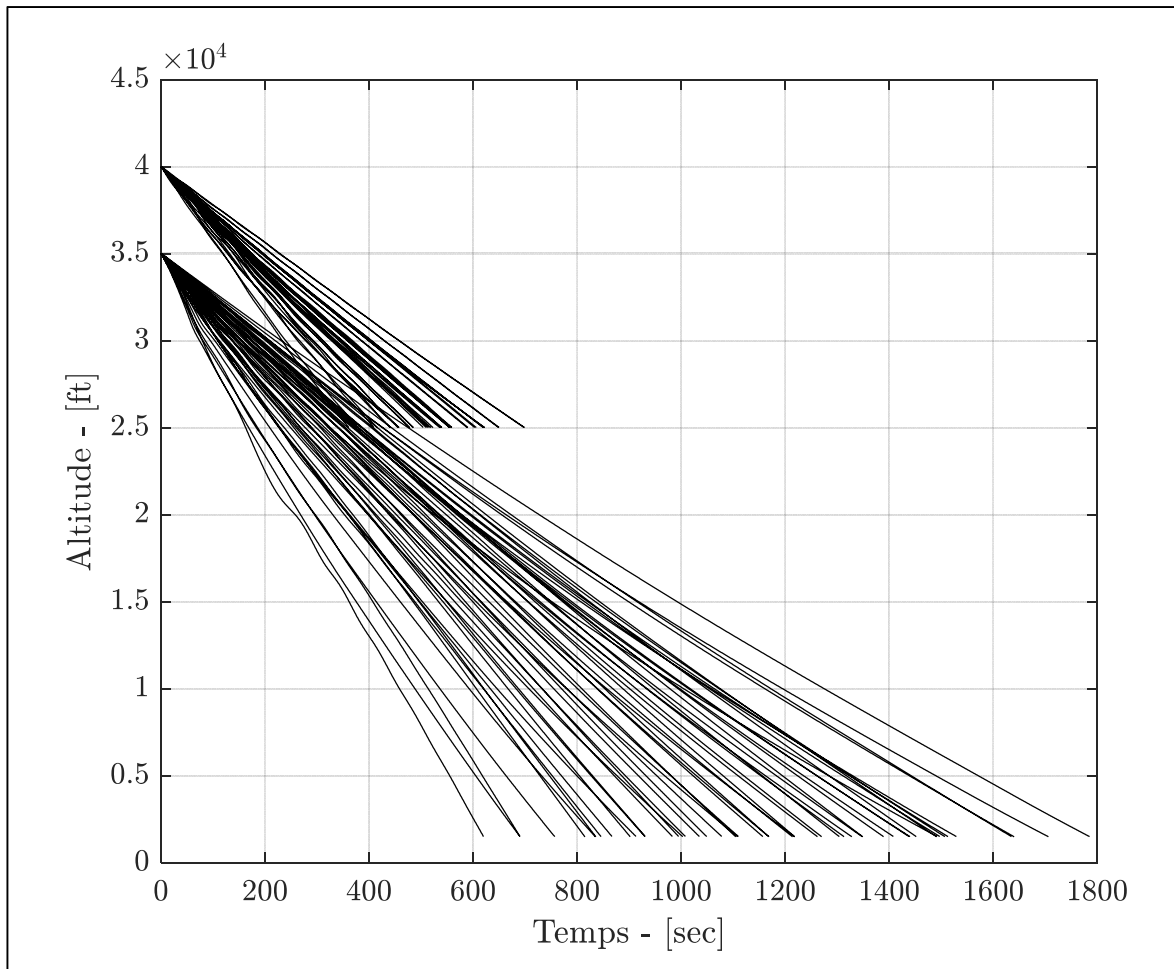


Figure 2.15 Ensemble des profils de descente

Un poids initial entre 50 000 lb et 70 000 lb a été choisi pour tous les tests de descente et ils ont été réalisés sur deux intervalles d'altitudes. Les premiers ensembles de tests ont été réalisés pour une altitude variant de 40 000 pi à 25 000 pi, tandis que les seconds tests ont été réalisés pour une altitude variant de 35 000 pi à 1 500 pi. Finalement, pour tous les tests de descente, les données ont été collectées avec une fréquence de 1 Hz (soit 1 donnée par seconde). La collecte de données a été arrêtée lorsque l'avion atteint l'altitude finale. Les données ont ensuite été réarrangées de sorte qu'on ait une donnée à chaque 1 000 pi afin de réduire la taille de la base de données.

La Figure 2.15 illustre l'ensemble des tests effectués pour la phase de descente. 93 tests ont été réalisés au total, dont 16 ont été sélectionnés pour l'identification du modèle, tandis que 77 ont été sélectionnés pour la validation du modèle.

2.2.5 Base de données complète du modèle de performance

Les données collectées lors des phases de décollage, de montée, de croisière et de descente ont été exportées du simulateur de recherche sous le format *.CSV, et ensuite elles ont été restructurées sur le logiciel Matlab afin d'établir une base initiale de données. Le nombre des tests de vol pour l'identification et la validation du modèle est montré dans le Tableau 2.3.

Tableau 2.3 Nombre des tests en vol pour chaque régime de vol

	Identification	Validation	Totaux
Régime de décollage	6	42	48
Régime de montée	8	47	55
Régime de croisière	39	154	193
Régime de descente	16	77	93
Totaux	69	320	389

Les variables collectées sont les conditions de vol, dont l'altitude pression h , la position de la manette des gaz TLA, le nombre de Mach M , la vitesse de vol calibrée CAS , les pressions P_{amb} , P_{01} , P_{02} , P_{03} , les températures T_{amb} , T_{04} , la vitesse de rotation de la soufflante N_1 , la vitesse de rotation de la turbine N_2 , le débit de carburant W_F et la poussée nette F_N .

La base de données a ensuite été complétée en y ajoutant les paramètres corrigés (dont la poussée corrigée $F_{N,C}$, la consommation de carburant corrigée $W_{F,C}$, la vitesse de rotation de la soufflante corrigée $N_{1,C}$ et la vitesse de rotation de la turbine corrigée $N_{2,C}$), ainsi que d'autres paramètres de design du turboréacteur dont le ratio de pressions du moteur EPR , le ratio de pressions de la soufflante FPR , le ratio de pressions totale OPR , la température

d'entrée de la turbine *ITT*. Les paramètres corrigés ($N_{1,C}$, $N_{2,C}$, $F_{N,C}$ et $W_{F,C}$) et les paramètres de design (*EPR*, *FPR*, *OPR* et *ITT*) sont calculés à partir des équations (2.8) à (2.15) et représentent une synthèse des équations déjà présentées dans les sections 1.7.4 et 1.7.5 :

$$EPR = \frac{P_{05}}{P_{01}} \quad (2.8)$$

$$FPR = \frac{P_{02}}{P_{01}} \quad (2.9)$$

$$ITT = T_{04} \quad (2.10)$$

$$OPR = \frac{P_{03}}{P_{01}} \quad (2.11)$$

$$N_{1,C} = \frac{N_1}{\sqrt{\theta}} \quad (2.12)$$

$$N_{2,C} = \frac{N_2}{\sqrt{\theta}} \quad (2.13)$$

$$F_{N,C} = \frac{F_N}{\delta} \quad (2.14)$$

$$W_{F,C} = \frac{W_F}{\delta\sqrt{\theta}} \quad (2.15)$$

Les paramètres δ et θ sont les ratios de pressions et de températures calculés à partir du modèle de l'atmosphère standard présenté à la section 1.7.2.

Enfin, chaque profil de vol a été discrétisé en « points de vol », en utilisant en moyenne 1 point tous les 20 à 30 secondes. Ainsi, la base de données compte environ 3 400 points pour l'ensemble des phases de vol. Les Figure 2.16 à Figure 2.22 présentent l'ensemble des données

pour chaque paramètre et pour chaque phase de vol en fonction de l'altitude et du nombre de Mach.

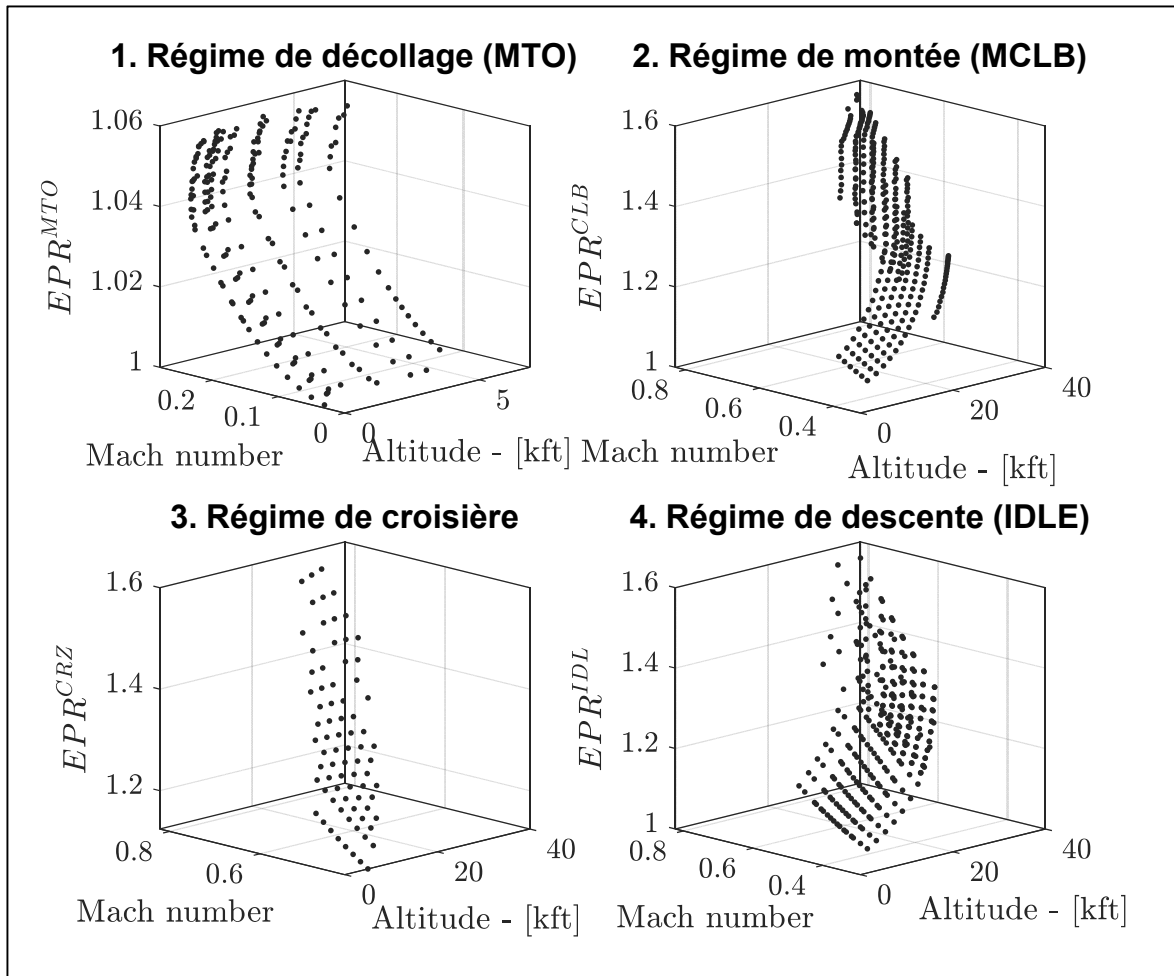


Figure 2.16 Ratio de pression du moteur EPR en fonction de l'altitude et du nombre de Mach pour les régimes de décollage, de montée, de croisière et de descente

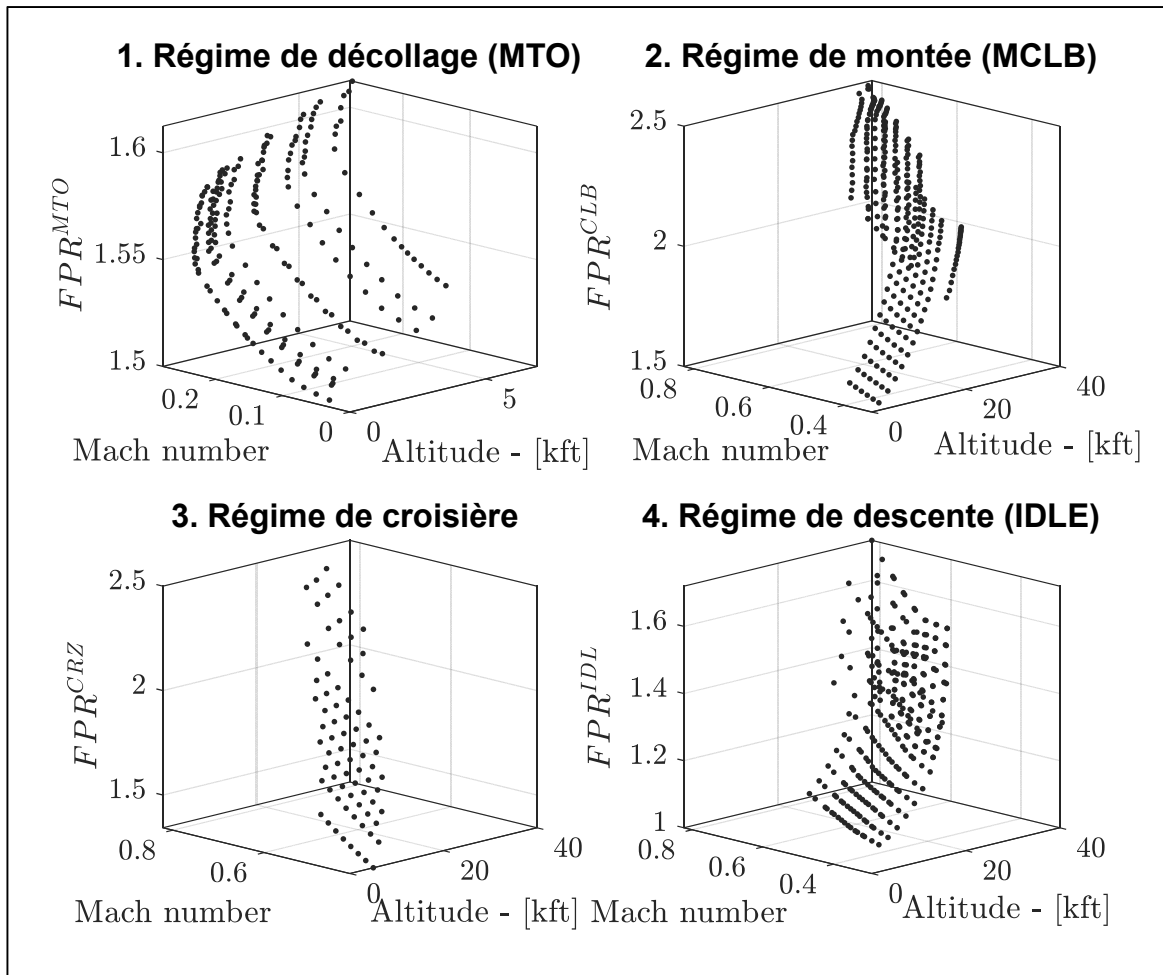


Figure 2.17 Ratio de pression de la soufflante FPR en fonction de l'altitude et du nombre de Mach pour les régimes de décollage, de montée, de croisière et de descente

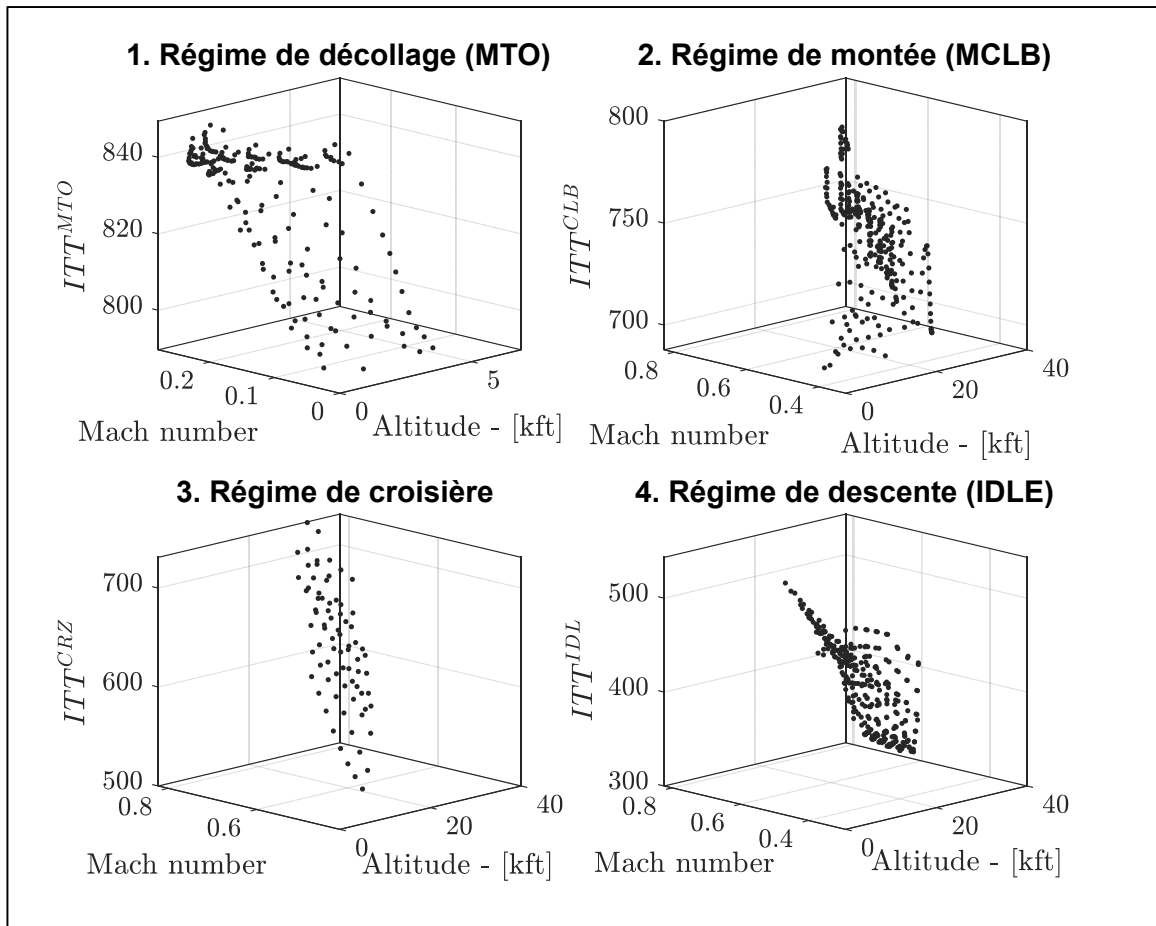


Figure 2.18 Température d'entrée de la turbine ITT en °K en fonction de l'altitude et du nombre de Mach pour les régimes de décollage, de montée, de croisière et de descente

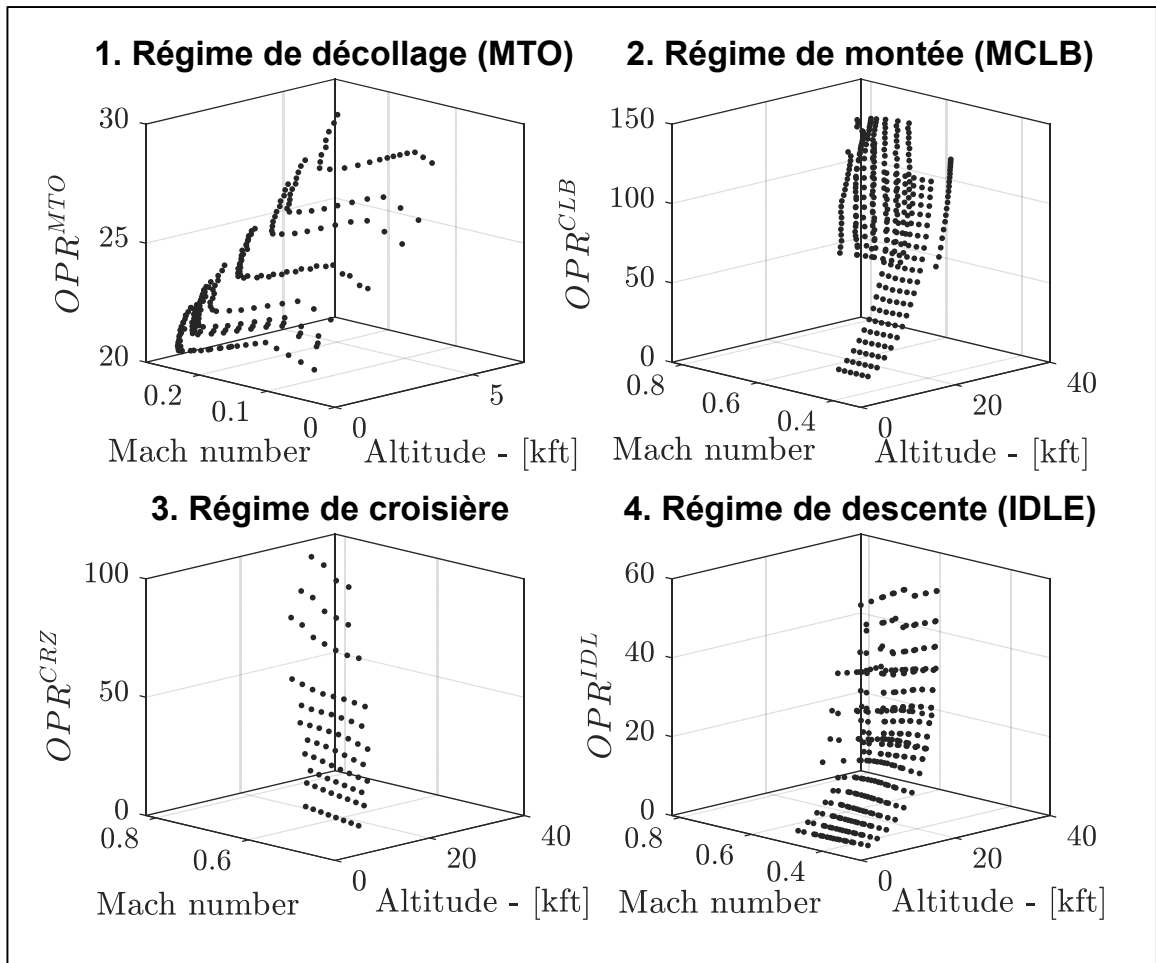


Figure 2.19 Ratio de pression totale OPR en fonction de l'altitude et du nombre de Mach pour les régimes de décollage, de montée, de croisière et de descente

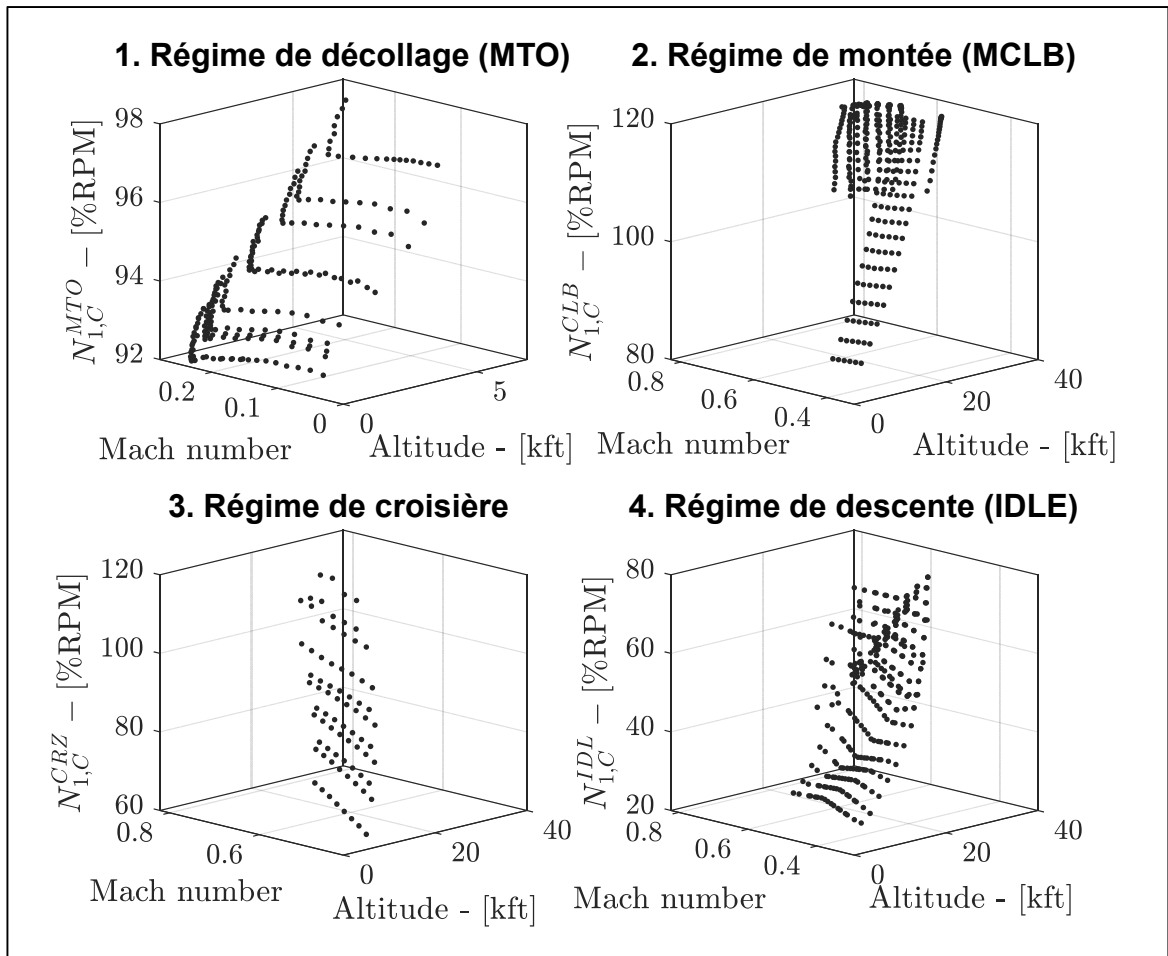


Figure 2.20 Vitesse de rotation de la soufflante corrigée $N_{1,C}$ en fonction de l'altitude et du nombre de Mach pour les régimes de décollage, de montée, de croisière et de descente

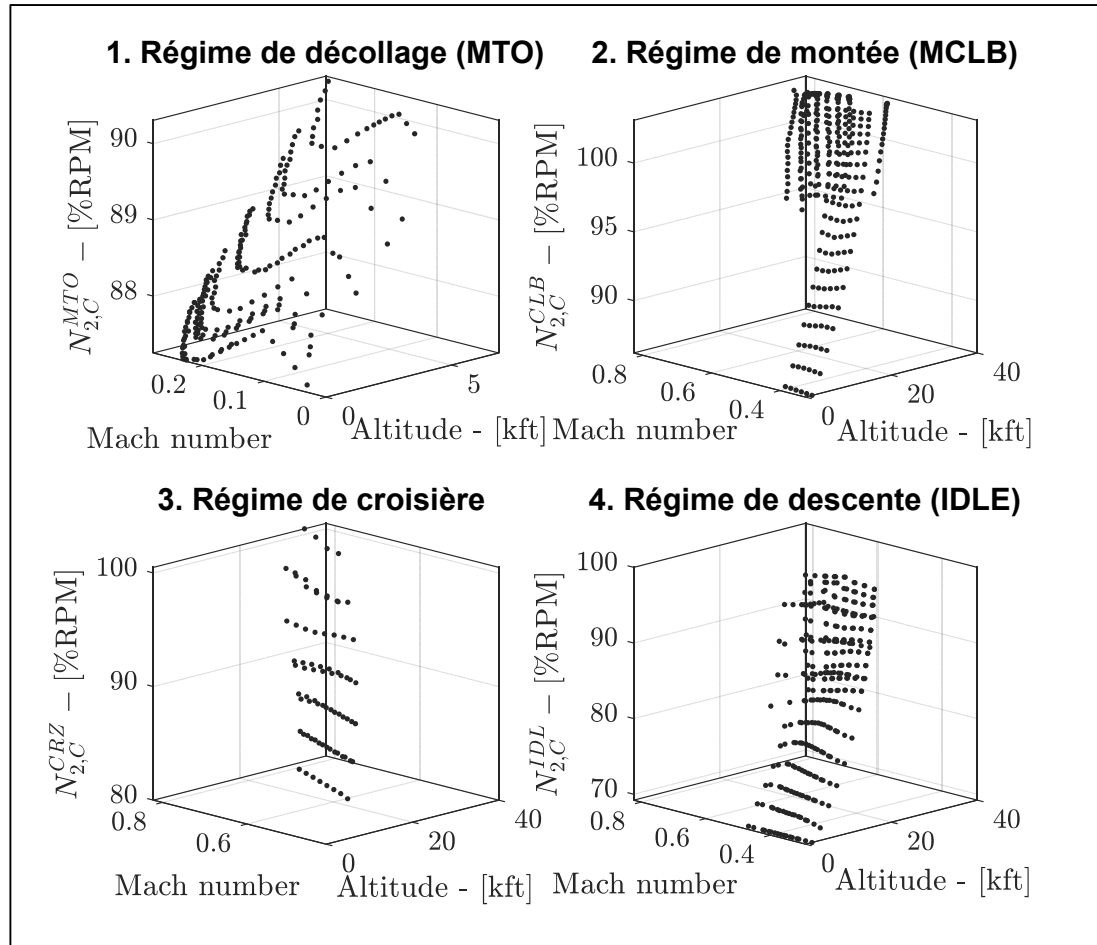


Figure 2.21 Vitesse de rotation de la turbine corrigée $N_{2,C}$ en fonction de l'altitude et du nombre de Mach pour les régimes de décollage, de montée, de croisière et de descente

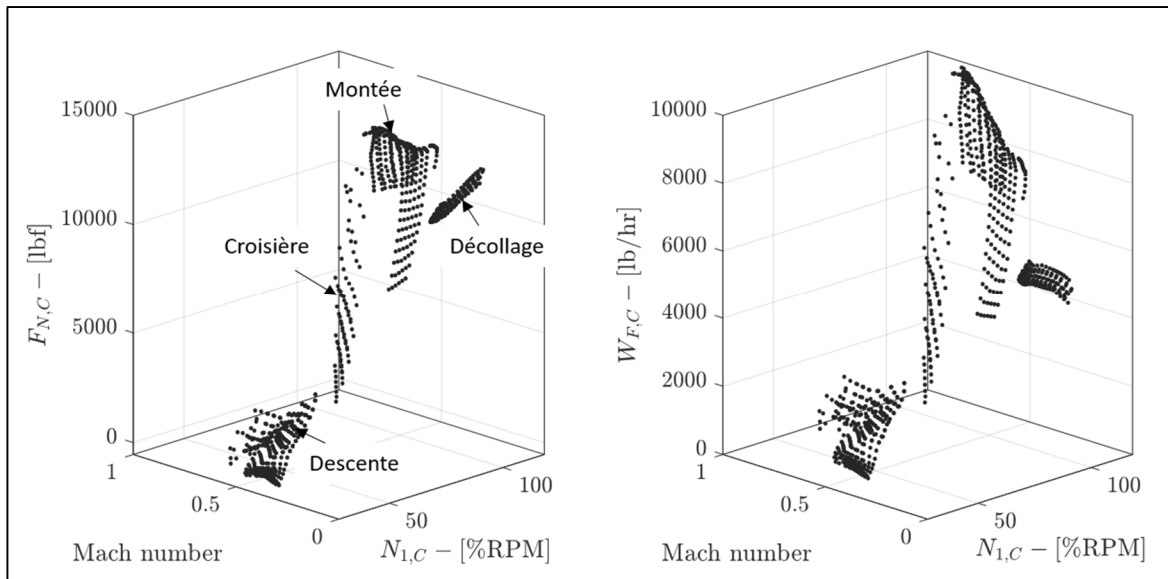


Figure 2.22 Poussée corrigée $F_{N,C}$ et débit de carburant corrigée $W_{F,C}$ en fonction du nombre de Mach et de la vitesse de rotation de la soufflante corrigée $N_{1,C}$ pour les régimes de décollage, de montée, de croisière et de descente

2.3 Identification du modèle de performance

Une fois que l'identification de la base de données a été effectuée, la prochaine étape de la méthodologie consiste à identifier le modèle de performance à partir des réseaux de neurones. Pour ce faire, il est nécessaire de définir la structure du réseau de neurones afin de reproduire la structure du moteur. En tout, 500 points (soit 15 % des données) ont été utilisés pour l'apprentissage du réseau de neurones et l'identification du modèle, tandis que 2 900 points (soit 85 % des données) ont été utilisés pour la validation du modèle.

2.3.1 Réseau de neurones de type perceptron multicouche

Il existe différents types de réseaux de neurones, comme ceux présentés à la section 1.4.2.1 (Hagan et *al.*, 2002). Toutefois, étant donné que l'objectif de cette étude consiste à concevoir un modèle statique, c'est-à-dire indépendant par rapport au temps, il a été décidé d'utiliser des réseaux de neurones de type perceptrons multicouches avec une propagation directe (en anglais

FeedForward Neural Network ou FFNN). Ce type de structure est approprié aux problèmes d'apprentissage supervisé, dont leur premier objectif consiste à établir des liens entre les données.

La Figure 2.23 représente le schéma d'un neurone artificiel. La description du neurone a été donnée dans la section 1.4.2.1. Le réseau de neurones FFNN de la Figure 2.24 consiste à une chaîne de structure mathématique qui imite le fonctionnement du cerveau humain afin d'établir des liens entre les paramètres dans un ensemble de données. Le FFNN est principalement composé de neurones artificiels interconnectés entre eux, et dont le rôle consiste à transporter et à transformer l'information à partir d'une ou plusieurs entrées, vers une ou plusieurs sorties.

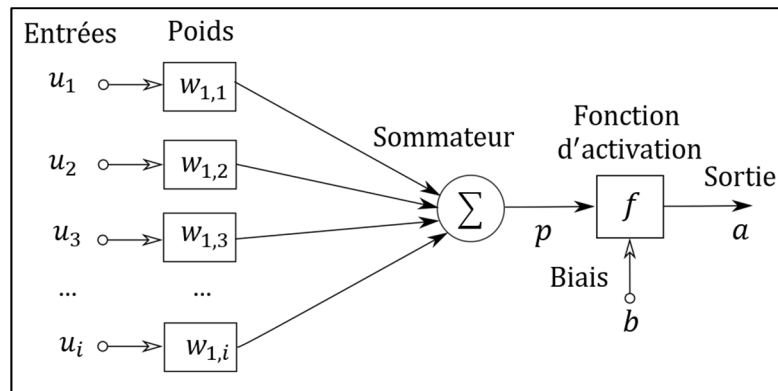


Figure 2.23 Représentation du neurone artificiel

Comme illustré sur la Figure 2.23 et par l'analogie avec des neurones biologiques de la Figure 1.3, un neurone artificiel peut être considéré comme un récepteur qui reçoit un flux d'information venant, soit d'autres neurones, soit directement des entrées du modèle. L'information est tout d'abord multipliée par des poids w , où chaque poids est une constante qui définit la proportion de l'entrée associée. Ainsi, plus le poids est large, plus l'entrée a un impact sur la sortie du neurone.

Les entrées multipliées par des poids sont ensuite pondérées et envoyées vers une fonction mathématique, que l'on nomme « fonction d'activation ». La fonction d'activation reproduit le comportement du neurone et peut varier dépendamment du problème à résoudre. Pour les problèmes de régression, le choix usuel est la fonction linéaire, la fonction hyperbolique tangente ou encore la fonction sigmoïde. Mathématiquement, les sorties du réseau de neurones artificiel peuvent être écrites sous la forme d'équation, comme suit :

$$a = f(\mathbf{u}, \mathbf{w}, b) = f(\mathbf{u} \cdot \mathbf{w}' + b) \quad (2.16)$$

où a est la sortie du neurone artificiel, $\mathbf{u} = [u_1, \dots, u_n]$ est le vecteur d'entrées, $\mathbf{w} = [w_1, \dots, w_n]$ est le vecteur de poids associé à chaque entrée et b est le biais. Le biais est un coefficient constant qui peut être utilisé pour ajuster la forme de la fonction d'activation f . À partir de cette représentation, le perceptron multicouche avec une propagation directe peut être défini comme étant une structure mathématique composée d'un ensemble de neurones, organisés en plusieurs couches, comme présentée sur la Figure 2.24.

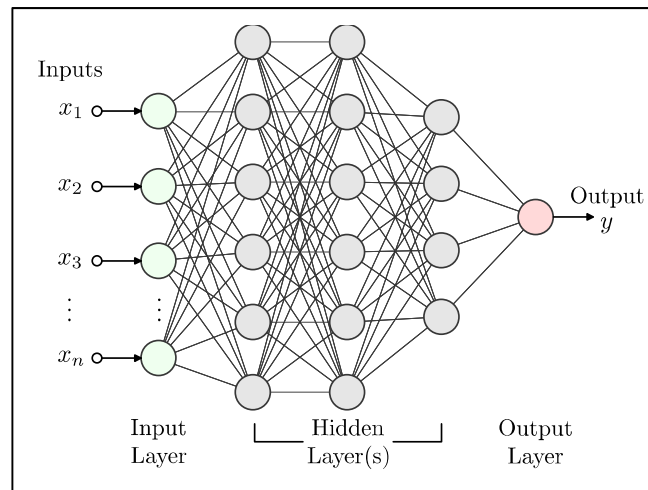


Figure 2.24 Représentation d'un réseau de neurones de type perceptron multicouche avec une propagation directe FFNN

Le nombre de couches, la fonction d'activation, ainsi que le nombre de neurones par couche sont définis par l'utilisateur, et déterminent la performance du réseau de neurones. Ainsi, il est important de bien choisir minutieusement ces paramètres afin de ne pas avoir de larges erreurs par rapport au modèle de référence.

2.3.2 Structure du modèle de moteur.

En se basant sur la description donnée dans les sections précédentes, et sur la théorie donnée dans la section 1.4.2.1, il a été décidé d'utiliser deux réseaux de neurones en cascade de type perceptron multicouche avec une propagation directe FFNN, tel que montré dans la Figure 2.25.

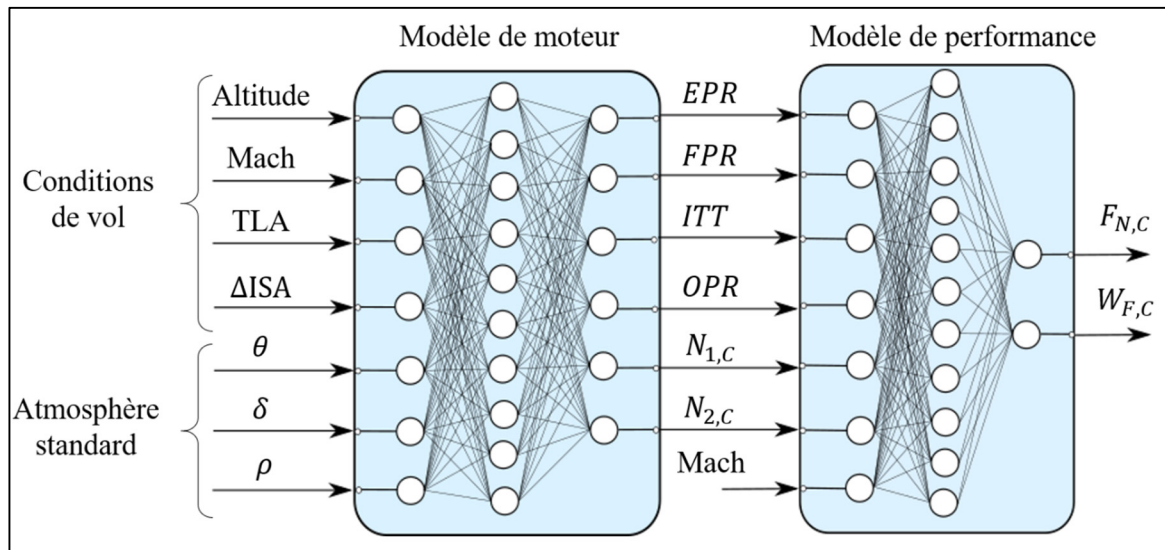


Figure 2.25 Structure complète du modèle de performance

Le premier réseau de neurones modélise le système de contrôle du moteur et il est dénommé par *Engine Control System Network* ou ECS-NET. Il consiste à modéliser les paramètres du moteur en fonction de l'altitude, du nombre de Mach, de la position de la manette des gaz TLA et des conditions de l'atmosphère standards. Les paramètres modélisés sont notamment les

ratios de pressions EPR , FPR et OPR , la température à l'entrée de la turbine ITT , la vitesse de rotation de la soufflante corrigée $N_{1,C}$ et la vitesse de rotation de la turbine corrigée $N_{2,C}$.

Le second réseau de neurones consiste à modéliser les performances du moteur et il est dénommé par *Engine Performance Network* ou ENG-NET. Il consiste à calculer la poussée corrigée $F_{N,C}$ et le débit de carburant corrigé $W_{F,C}$ à partir du nombre de Mach et des paramètres modélisés à partir du premier réseau de neurones ECS-NET. Une fois que les paramètres de performances ont été calculés, la valeur réelle de la vitesse de rotation de la soufflante N_1 , de la vitesse de rotation de la turbine N_2 , de la poussée F_N et du débit de carburant W_F sont obtenues comme suit :

$$N_1 = N_{1,C} \sqrt{\theta} \quad (2.17)$$

$$N_2 = N_{2,C} \sqrt{\theta} \quad (2.18)$$

$$F_N = F_{N,C} \delta \quad (2.19)$$

$$W_F = W_{F,C} \delta \sqrt{\theta} \quad (2.20)$$

L'avantage principal de représenter le moteur sous cette forme (Figure 2.25) est que celle-ci permet de représenter explicitement les éléments principaux du moteur : le système de contrôle du moteur d'une part, et le modèle de performance d'autre part.

Dans le cas de la modélisation à partir de la vitesse de rotation de la soufflante corrigée $N_{1,C}$, les données de l'atmosphère ne sont pas nécessaires (voir Annexe I) (Andrianantara et al., 2021). Par contre, au cours de la recherche, on a pu remarquer qu'il est nécessaire de spécifier explicitement les paramètres de l'atmosphère pour la modélisation des paramètres de design.

2.3.3 Apprentissage du réseau de neurones.

L'entraînement consiste à trouver les poids w et les biais b du réseau de neurones, qui permettent d'obtenir une performance optimale. Cette étape est réalisée en appliquant un algorithme d'optimisation qui tend à obtenir une combinaison optimale des poids w et des biais b , afin de minimiser une fonction de coût. Pour les problèmes de régressions, la fonction de coût la plus utilisée est la moyenne des erreurs quadratiques (en anglais *Mean Square Error* ou MSE). Cette fonction permet de quantifier la précision du réseau de neurones en comparant les valeurs estimées avec les valeurs de références comme montré dans l'équation (2.21):

$$MSE(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{2N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (2.21)$$

où N est le nombre de données, $\mathbf{y} = [y_1, \dots, y_N]$ est le vecteur correspondant aux valeurs de références, et $\hat{\mathbf{y}} = [\hat{y}_1, \dots, \hat{y}_N]$ est le vecteur correspondant aux valeurs estimées à partir du réseau de neurones.

La Figure 2.26 illustre le réseau de neurones lors de l'apprentissage sur Matlab. L'algorithme d'optimisation considéré est « l'algorithme de régularisation bayésienne », disponible dans la boîte d'outils dédiés aux réseaux de neurones du logiciel Matlab. L'algorithme de régularisation bayésienne est présenté dans la section 1.5.5. Il permet une excellente généralisation et assure la convergence des solutions en utilisant un minimum de données. De plus, afin de simplifier le processus d'entraînement, le nombre de couches a été réduit à une seule couche cachée. Le seul paramètre à déterminer sera alors le nombre de neurones. Cette stratégie a été évaluée, en se basant sur la propriété fondamentale des réseaux de neurones, qui dit qu'une fonction peut être approximée par une seule couche cachée, du moment que le nombre de neurones est suffisant. Enfin, après avoir testé plusieurs fonctions d'activations, on a décidé que la fonction hyperbolique tangente sera utilisée pour la couche cachée tandis que la fonction linéaire sera utilisée pour modéliser les sorties du réseau de neurones.

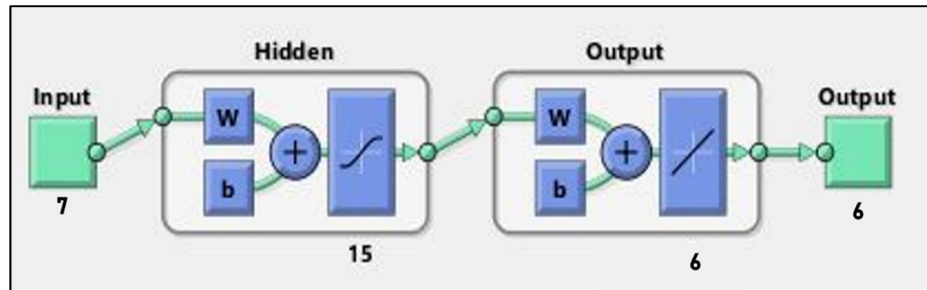
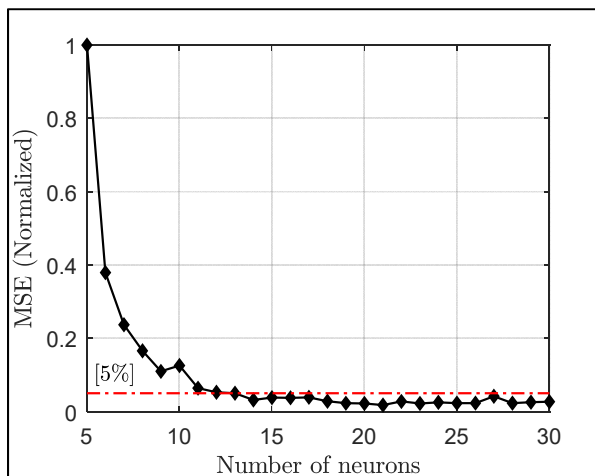
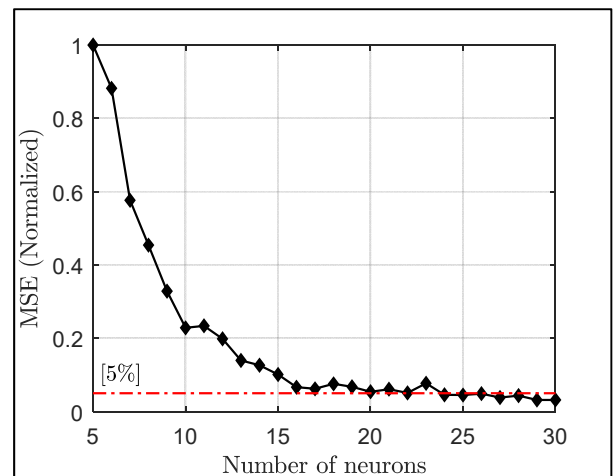


Figure 2.26 Identification du réseau de neurones sur Matlab

La Figure 2.27 montre la variation du MSE en fonction du nombre de neurones pour chaque réseau de neurones. On peut noter que les valeurs du MSE ont été normalisées entre 0 et 1 pour une meilleure analyse des résultats. Le nombre de neurones est fixé lorsque la valeur du MSE atteint 5 % de la valeur maximale. Ce choix permet d'éviter les situations de surapprentissage (en anglais *overfitting*) ou de sousapprentissage (en anglais *underfitting*) du réseau de neurones.



a) ECS-NET



b) ENG-NET

Figure 2.27 Variations de MSE pour chaque réseau de neurones (ECS-NET and ENG-NET) en fonction du nombre de neurones dans la couche cachée

2.3.4 Résultats de l'identification du réseau de neurones

Les résultats de la Figure 2.27a montrent que le MSE diminue lorsque le nombre de neurones augmente. Toutefois, on remarque que lorsque le nombre de neurones est supérieur à 13, le MSE est inférieur à 5 % de sa valeur maximale et qu'au-delà de ce point, la valeur du MSE converge lentement. De plus, en choisissant un nombre de neurones supérieur à 14, cela pourrait amener à un surentraînement. Dans ce cas, le réseau de neurones se contente d'apprendre « par cœur » sans comprendre la tendance des données ni de généraliser les résultats pour d'autres conditions de vol. Ainsi, à partir des observations, un compromis peut être établi pour un nombre de neurones égal à 15, ce qui évite un surentraînement et assure une valeur optimale de MSE. En appliquant le même raisonnement, pour le second réseau ENG-NET, la Figure 2.27b montre que le nombre de neurones optimal est de 20.

Tableau 2.4 Résultats de l'entraînement des réseaux de neurones

	ECS-NET	ENG-NET	
Nombre de neurones	15	20	
Nombre d'itérations	924	1697	
MSE (Normalisée)	0.031	0.035	
Paramètres du moteur	<i>EPR, FPR, OPR, ITT, N_{1,C}, N_{2,C}</i>	<i>F_N [lbf]</i>	<i>W_F [lb/hr]</i>

2.3.5 Résultats préliminaires

Les résultats de l'entraînement des réseaux de neurones sont présentés dans le Tableau 2.4. Les Figure 2.28 à Figure 2.35 montrent les distributions gaussiennes des erreurs résiduelles de chaque paramètre, calculées pour l'ensemble des points de la base de données. L'erreur résiduelle *RES* correspond à la différence entre la valeur mesurée y et la valeur estimée $\hat{y}$, soit :

$$RES_i = y_i - \hat{y}_i \quad (2.22)$$

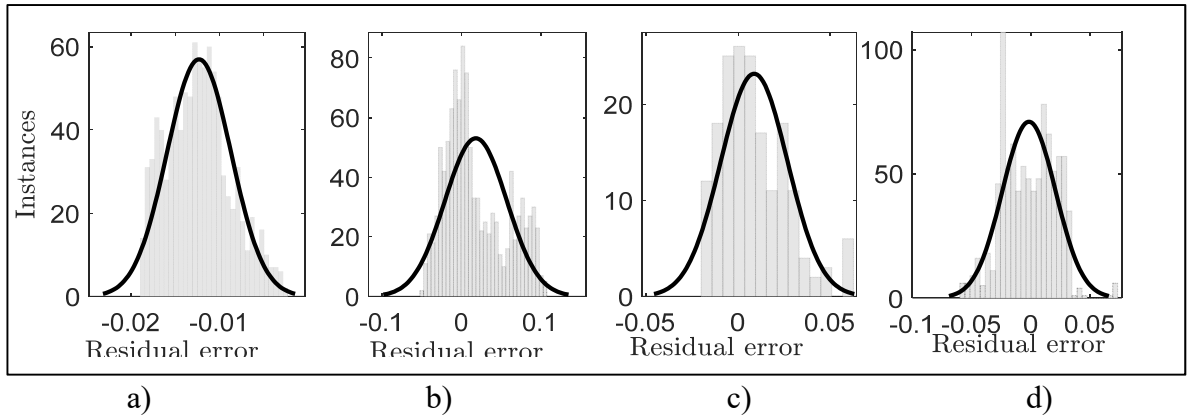


Figure 2.28 Erreurs résiduelles du ratio de pressions du moteur EPR pour les régimes de (a) décollage, (b) montée, (c) croisière et (d) descente

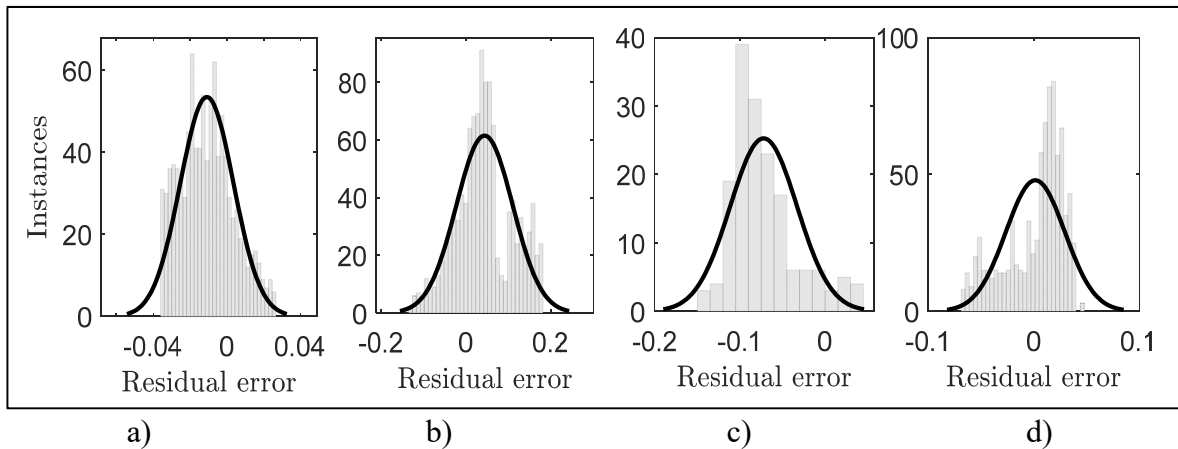


Figure 2.29 Erreurs résiduelles du ratio de pressions de la soufflante FPR pour les régimes de (a) décollage, (b) montée, (c) croisière et (d) descente

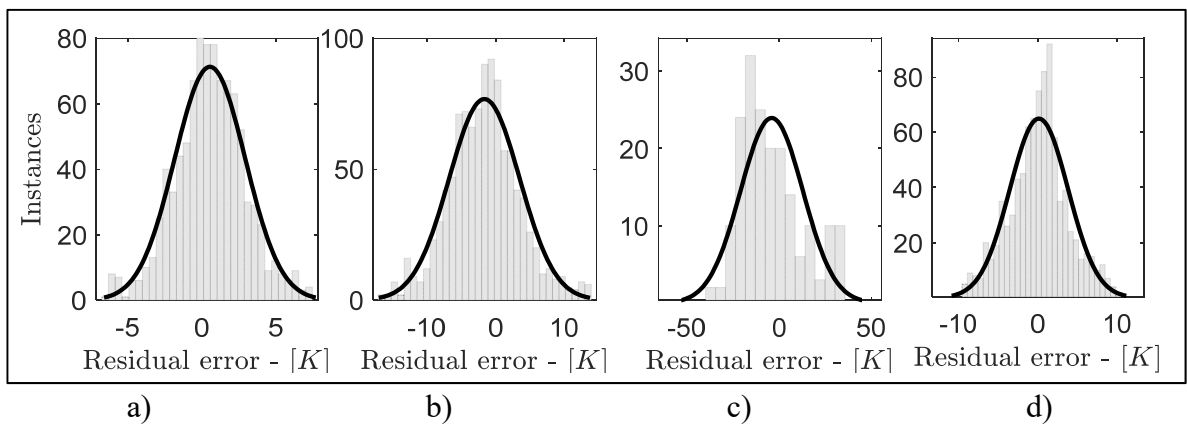


Figure 2.30 Erreurs résiduelles de la températures d'entrée de la turbine ITT pour les régimes de (a) décollage, (b) montée, (c) croisière et (d) descente

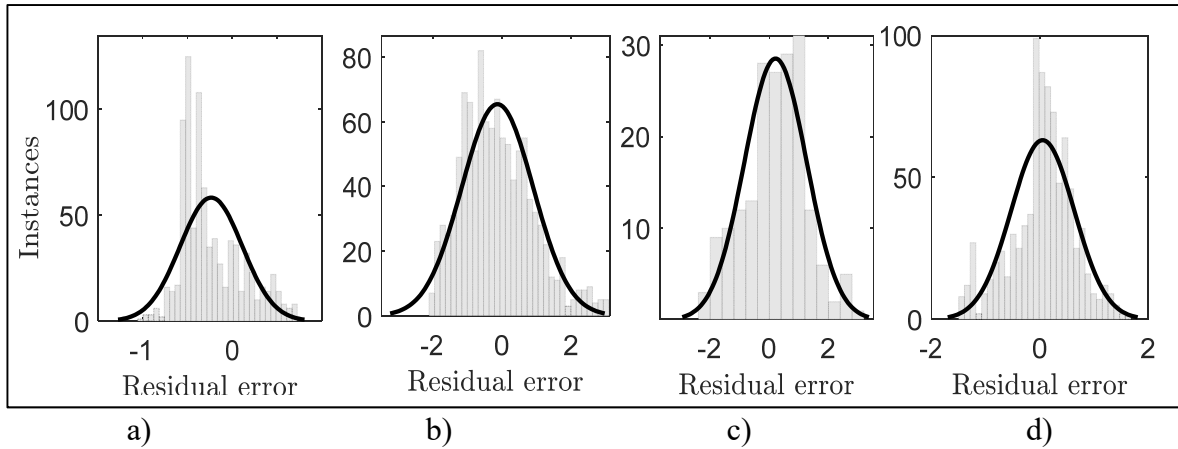


Figure 2.31 Erreurs résiduelles du ratio de pressions total OPR pour les régimes de (a) décollage, (b) montée, (c) croisière et (d) descente

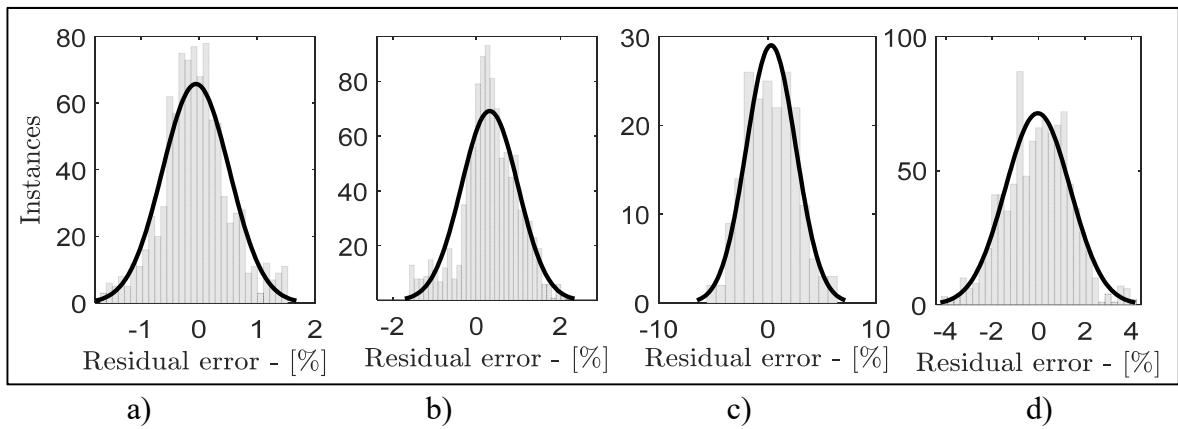


Figure 2.32 Erreurs résiduelles de la vitesse de rotation de la soufflante corrigée $N_{1,c}$ pour les régimes de (a) décollage, (b) montée, (c) croisière et (d) descente

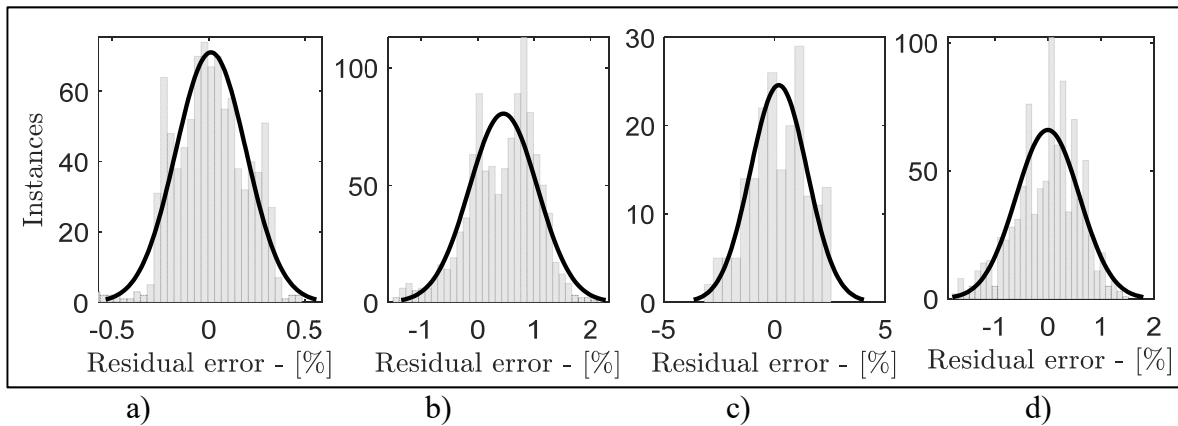


Figure 2.33 Erreurs résiduelles de la vitesse de rotation de la turbine corrigée $N_{2,c}$ pour les régimes de (a) décollage, (b) montée, (c) croisière et (d) descente

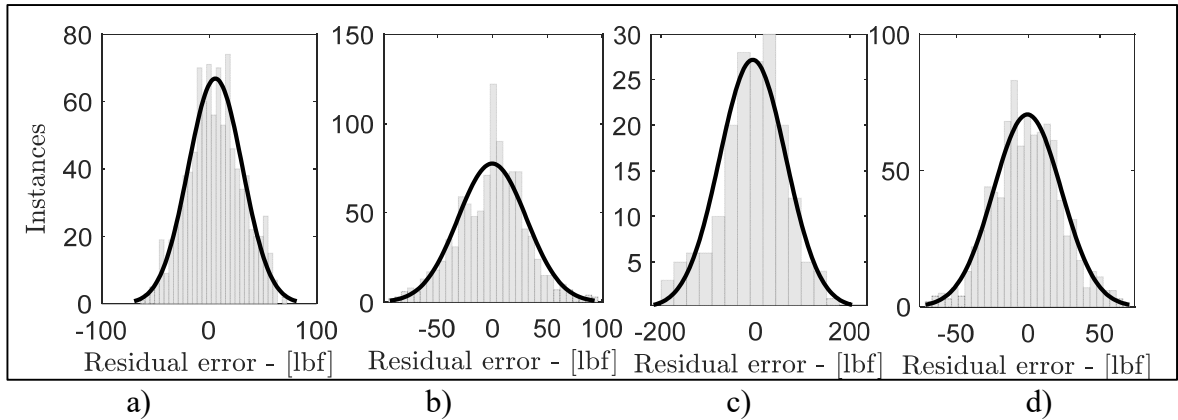


Figure 2.34 Erreurs résiduelles de la poussée corrigée $F_{N,C}$ pour les régimes de (a) décollage, (b) montée, (c) croisière et (d) descente

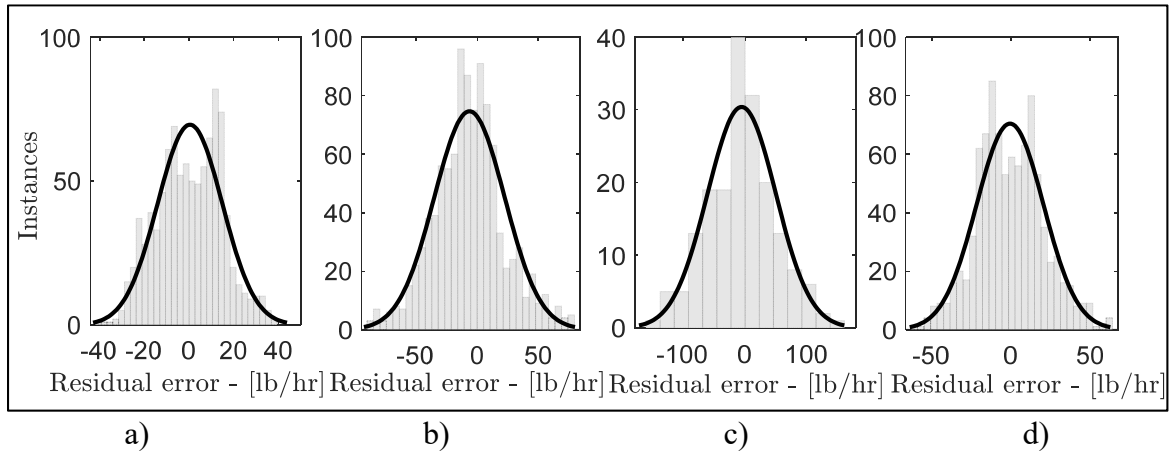


Figure 2.35 Erreurs résiduelles du débit de carburant corrigé $W_{F,C}$ pour les régimes de (a) décollage, (b) montée, (c) croisière et (d) descente

En effet, le calcul de l'erreur résiduelle est une étape préliminaire pour la validation du modèle et permet d'émettre une première conclusion. D'un point de vue général, les résultats sont excellents et montrent que le modèle est capable de prédire les paramètres du moteur grâce au réseau de neurones ECS-NET et les performances du moteur grâce au réseau de neurones ENG-NET. Les erreurs résiduelles sont notamment petites et centrées sur 0 pour chaque sortie. Par exemple, la poussée $F_{N,C}$ et le débit de carburant $W_{F,C}$ ont des erreurs résiduelles maximales de 200 lbf et 100 lb/hr respectivement, ce qui est relativement négligeable par rapport à la valeur maximale de la poussée produite par le moteur (13 790 lbf), et par rapport au débit de carburant maximal mesuré (10 000 lb/hr).

Les réseaux de neurones obtenus ECS-NET et ENG-NET sont donc capables de satisfaire les critères de performances du modèle et peuvent être étudiés pour une validation plus approfondie du modèle. Le prochain chapitre développera en détail les résultats grâce à une analyse statistique et des exemples d'application seront proposés. Une comparaison sera également réalisée en considérant différentes combinaisons pour les sorties du modèle de moteur ECS-NET.

CHAPITRE 3

VALIDATION DU MODÈLE DE RÉSEAUX DE NEURONES

Ce chapitre présente la validation des résultats pour les modèles de réseaux de neurones, modélisés dans la section 2.3. Les tests de vol dédiés à cette validation sont résumés dans le Tableau 2.3. Un nombre de 320 tests a été réalisé sur le simulateur de vol du CRJ700, dont 42 tests de décollage avec la manette des gaz placée sur la position MTO, 47 tests de montée avec la manette des gaz placée sur la position MCLB, 77 tests de descente, avec la manette des gaz placée sur la position IDLE et 154 tests statiques de croisière. Les tests de vol ont été réalisés suivant les mêmes procédures que celles décrites dans la section 2.2. La validation du modèle de réseaux de neurones a été réalisée en comparant les paramètres réels du simulateur de vol avec ceux prédits par le modèle. Le critère établi pour la validation du modèle consiste en une tolérance de $\pm 5\%$ sur les sorties du simulateur de vol selon le *Federal Aviation Administration* (FAA).

3.1 Combinaisons du modèle de moteur ECS-ENG

Dans cette section, nous proposons différentes combinaisons de sorties pour le modèle de moteur ECS-ENG présenté dans la section 2.3.2. En effet, selon la théorie décrite dans la section 1.7, la modélisation des performances du moteur peut être réalisée à partir de différents paramètres. Il existe l'approche par composant (section 1.7.4), qui s'intéresse seulement aux paramètres de conception du moteur dont les EPR , FPR , OPR et ITT , mais il existe également d'autres modèles comme le modèle simplifié qui prend seulement en compte la vitesse de rotation du moteur N_1 (section 1.7.5). Dans tous les cas, les paramètres de performances à modéliser sont la poussée F_N et le débit de carburant W_F .

Ainsi, afin d'obtenir différentes combinaisons de sorties du modèle de réseau de neurones ECS-ENG, il est nécessaire de numéroter chaque sortie comme dans le Tableau 3.1.

Tableau 3.1 Numérotation des sorties du bloc «Modèle de moteur»

Paramètre du moteur	<i>EPR</i>	<i>FPR</i>	<i>ITT</i>	<i>OPR</i>	N_1	N_2
Numéro de sortie	1	2	3	4	5	6

Un nom spécifique est ensuite attribué à chaque modèle selon la combinaison de ses sorties. Par exemple, le modèle qui comprend les sorties *EPR* (sortie 1) et *FPR* (sortie 2) sera renommé « modèle E12 ». La méthode de modélisation du moteur ECS-ENG est la même que celle décrite dans le chapitre précédent, à partir des réseaux de neurones. Ainsi, dans le chapitre précédent, le modèle qui comprend toutes les sorties *EPR*, *FPR*, *ITT*, *OPR*, N_1 et N_2 sera renommé par « modèle E123456 ». Le Tableau 3.2 montre toutes les combinaisons de sorties qui ont été considérées avec les noms attribués à chaque modèle.

Tableau 3.2 Dénomination des modèles de réseaux de neurones

Nom du modèle	Sorties du bloc « Modèle de moteur » ECS-ENG					
	1	2	3	4	5	6
E1	<i>EPR</i>					
E12	<i>EPR</i>	<i>FPR</i>				
E12345	<i>EPR</i>	<i>FPR</i>	<i>ITT</i>	<i>OPR</i>	N_1	
E14	<i>EPR</i>			<i>OPR</i>		
E15	<i>EPR</i>				N_1	
E5					N_1	
E56					N_1	N_2
E1234	<i>EPR</i>	<i>FPR</i>	<i>ITT</i>	<i>OPR</i>		
E123456	<i>EPR</i>	<i>FPR</i>	<i>ITT</i>	<i>OPR</i>	N_1	N_2
S3	Aucune sortie					
E13	<i>EPR</i>		<i>ITT</i>			
E2		<i>FPR</i>				
E4				<i>OPR</i>		

Pour le modèle S3 du Tableau 3.2, on peut noter que le moteur ECS-ENG n'est pas modélisé. Dans ce cas, les paramètres de performance F_N et W_F sont directement calculés à partir du nombre de Mach, de l'altitude et des conditions atmosphériques, toujours grâce aux réseaux de neurones. Le modèle S3 est un modèle de performance plus facile à modéliser, étant donné qu'il ne comporte qu'un seul réseau de neurones avec une seule couche cachée. Il permet de démontrer la capacité des réseaux de neurones à s'adapter à différents types de données en entrée et en sortie.

3.2 Validation des modèles de performances

Une analyse statistique a été réalisée dans cette section pour tous les modèles de moteur montrés dans le Tableau 3.2. Les paramètres de statistique utilisés sont l'erreur relative moyenne μ , la déviation standard σ , et l'erreur relative maximale pour chaque phase de vol et pour chaque paramètre. Enfin, une méthode de sélection du meilleur modèle est proposée dans la section 3.3.

3.2.1 Critères d'analyse statistique

Pour chaque observation y_i et chaque estimation $\hat{y}_i$ correspondante à y_i , l'erreur relative $\bar{\varepsilon}_i$ en [%] est calculée comme suit :

$$\bar{\varepsilon}_i = \frac{y_i - \hat{y}_i}{y_i} \quad (3.1)$$

La moyenne des erreurs relatives μ et la déviation standard des erreurs relative σ sont ensuite calculées comme suit :

$$\mu = \frac{1}{N} \sum_{i=1}^N \bar{\varepsilon}_i \quad (3.2)$$

$$\sigma = \sqrt{\frac{1}{N-1} \sum_{i=1}^N |\bar{\varepsilon}_i - \mu|^2} \quad (3.3)$$

où $\bar{\varepsilon}$ est l'erreur relative et N est le nombre d'observations. Par définition, la déviation standard σ correspond à la racine carrée de la variance et elle décrit la dispersion des erreurs relatives $\bar{\varepsilon}_i$ autour d'une valeur moyenne μ .

Pour un modèle idéal, l'erreur relative moyenne μ et la déviation standard sont nulles (Jategaonkar, 2015). Cependant, pour un modèle réaliste comme dans notre cas, on considérera une tolérance de 5 % définie par le FAA sur l'erreur relative moyenne μ pour chaque sortie du modèle. Cette approche statistique permet de généraliser la précision du modèle sur toute l'enveloppe de vol de l'avion et pour toutes les phases de vol.

3.2.2 Résultats de la validation

Les Tableau 3.3 à Tableau 3.9 présentent les erreurs relatives moyennes μ , les déviations standards σ et les erreurs maximales max. en valeurs absolues en [%] pour chaque paramètre de sortie, modèle et régime de vol.

L'erreur maximale max. correspond à l'erreur relative maximale parmi les N observations et elle représente moins de 1 % du nombre de données. Sur les distributions gaussiennes des erreurs présentées dans les Figure 2.28 à Figure 2.35, l'erreur maximale correspond à la valeur la plus éloignée de la moyenne μ et elle est exprimée en [%].

3.2.2.1 Ratio de pression du moteur *EPR*

Pour le paramètre *EPR* du Tableau 3.3 on remarque que l'erreur moyenne μ est inférieure à 1 % tandis que la déviation standard σ est inférieure à 4.5 % sauf pour le modèle E15. En effet, pour le modèle E15 on remarque une plus grande dispersion de *EPR*; la moyenne relative μ est de 2 % pour les phases de montée et de croisière. Cependant, les déviations standards σ sont inférieures à la tolérance de 5 %. Par conséquent le paramètre *EPR* est validé pour tous les modèles et pour toutes les phases de vol..

Tableau 3.3 Erreurs moyennes, déviations standards et erreurs maximales du ratio de pression du moteur *EPR* en [%] pour les régimes de décollage, de montée, de croisière et de descente

	Régime de décollage			Régime de montée			Régime de croisière			Régime de descente		
Modèle	μ	σ	max	μ	σ	max	μ	σ	max	μ	σ	max
E1	0.01	0.03	0.07	0.01	0.05	0.15	0.00	0.01	0.04	0.01	0.04	0.10
E12	0.01	0.03	0.13	0.01	0.06	0.21	0.01	0.03	0.08	0.02	0.04	0.16
E12345	0.50	0.16	0.00	0.76	3.19	8.00	0.73	1.85	6.92	0.32	1.78	6.58
E14	0.02	0.12	0.46	0.03	0.50	0.96	0.10	0.51	1.46	0.03	0.49	1.41
E15	0.83	1.21	3.92	1.98	4.31	11.64	1.95	3.01	5.20	0.12	1.76	7.09
E1234	0.93	0.44	0.13	0.69	2.36	6.81	0.88	0.97	3.95	0.10	1.28	5.24
E123456	0.49	0.55	1.35	0.42	3.09	7.13	0.02	2.77	6.80	0.41	1.19	4.24
E13	0.13	0.20	0.39	0.25	1.37	4.43	0.45	1.23	3.78	0.04	1.16	4.42

3.2.2.2 Ratio de pression de la soufflante *FPR*

Pour le paramètre *FPR* du Tableau 3.4 on remarque que l'erreur moyenne μ est inférieure à 1 % tandis que la déviation standard σ est inférieure à 2 % sauf pour le modèle E1234 et E123456 pour la phase de croisière, où la moyenne est de 2.5 %. Cependant, les erreurs moyennes μ et les déviations standards σ sont inférieures à la tolérance de 5 % en croisière. Et les erreurs maximales avoisinent également 7 %. Par conséquent, le paramètre *FPR* est validé pour tous les modèles et pour toutes les phases de vol.

Tableau 3.4 Erreurs moyennes, déviations standards et erreurs maximales du ratio de pression de la soufflante *FPR* en [%] pour les régimes de décollage, de montée, de croisière et de descente

	Régime de décollage			Régime de montée			Régime de croisière			Régime de descente		
Modèle	μ	σ	max	μ	σ	max	μ	σ	max	μ	σ	max
E12	0.01	0.05	0.16	0.01	0.10	0.38	0.04	0.49	1.38	0.01	0.16	0.57
E12345	0.13	0.50	1.86	0.87	2.52	7.12	0.52	2.29	7.35	0.22	1.63	5.88
E1234	0.39	1.21	4.41	0.93	1.58	6.13	2.51	1.97	3.67	0.00	1.75	5.13
E123456	0.21	1.23	3.86	1.01	2.26	6.48	2.31	4.56	6.11	0.19	2.11	7.05
E2	0.01	0.10	0.19	0.01	0.22	0.73	0.00	0.47	1.28	0.01	0.53	1.79

3.2.2.3 Température d'entrée de la turbine *ITT*

Pour le paramètre *ITT* du Tableau 3.5, on remarque que l'erreur moyenne μ est inférieure à 1 % tandis que la déviation standard σ est inférieure à 5 % pour tous les modèles, sauf pour le modèle E12345 où elle est de 6 %. Par contre l'erreur maximale est assez élevée et atteint 23 % pour le modèle E12345 et 17 % pour le modèle E1234 pour la phase de descente.

Tableau 3.5 Erreurs moyennes, déviations standards et erreurs maximales de la température d'entrée de la turbine *ITT* en [%] pour les régimes de décollage, de montée, de croisière et de descente

	Régime de décollage			Régime de montée			Régime de croisière			Régime de descente		
Modèle	μ	σ	max	μ	σ	max	μ	σ	max	μ	σ	max
E12345	0.40	1.17	2.53	0.13	1.63	6.39	0.48	2.70	4.65	0.04	5.83	23.29
E1234	0.42	1.48	4.05	0.23	1.55	5.12	0.42	2.87	6.58	0.23	4.22	16.92
E123456	0.16	1.43	2.97	0.16	0.94	3.55	0.84	4.10	9.86	0.07	2.88	8.60
E13	0.08	0.34	1.42	0.06	0.61	2.35	0.00	0.66	2.00	0.05	0.82	2.92

Comme il est dit dans la section 1.13, cette large erreur maximale peut s'expliquer du fait que la température de la chambre de combustion est plus difficile à modéliser, étant donné que les réactions chimiques sont très petites au niveau moléculaire. Cependant, les erreurs moyennes

μ et les déviations standards σ sont inférieures à la tolérance de 5 %. Le modèle est donc capable de prédire le paramètre *ITT* pour tous les modèles et pour toutes les phases de vol.

3.2.2.4 Ratio de pression totale *OPR*

Pour le paramètre *OPR* du Tableau 3.6 on remarque que l'erreur moyenne μ est inférieure à 0.5 % tandis que la déviation standard σ est inférieure à 3 % pour tous les modèles et phases de vol. Les erreurs maximales sont également inférieures à 8 %. Par conséquent le paramètre *OPR* est validé pour tous les modèles et phases de vol.

Tableau 3.6 Erreurs moyennes, déviations standards et erreurs maximales du ratio de pression totale *OPR* en [%] pour les régimes de décollage, de montée, de croisière, et de descente

	Régime de décollage			Régime de montée			Régime de croisière			Régime de descente		
Modèle	μ	σ	max	μ	σ	max	μ	σ	max	μ	σ	max
E12345	0.12	0.35	1.34	0.10	0.98	3.78	0.44	2.71	6.12	0.00	0.99	3.89
E14	0.01	0.24	0.77	0.04	0.30	1.20	0.09	1.64	4.42	0.04	0.74	2.73
E1234	0.10	0.31	1.17	0.10	0.89	3.41	0.19	2.74	8.09	0.07	0.80	3.14
E123456	0.12	0.37	1.60	0.15	0.73	2.79	0.48	2.74	6.48	0.03	1.07	4.02
E4	0.02	0.42	1.67	0.08	0.54	2.20	0.01	1.37	4.64	0.04	1.34	4.98

3.2.2.5 Vitesse de rotation de la soufflante N_1

Pour le paramètre N_1 du Tableau 3.7, on remarque que l'erreur moyenne μ est inférieure à 1 % tandis que la déviation standard σ est inférieure à 3 % pour tous les modèles et phases de vol, tandis que les erreurs maximales *max* sont inférieures à 6 %. Par conséquent le paramètre N_1 est validé pour tous les modèles et phases de vol.

Tableau 3.7 Erreurs moyennes, déviations standards et erreurs maximales de la vitesse de rotation de la soufflante N_1 en [%] pour les régimes de décollage, de montée, de croisière et de descente

	Régime de décollage			Régime de montée			Régime de croisière			Régime de descente		
Modèle	μ	σ	max	μ	σ	max	μ	σ	max	μ	σ	max
E12345	0.02	0.64	2.03	0.08	0.82	3.22	0.73	1.96	5.54	0.13	2.77	4.99
E15	0.01	0.08	0.21	0.00	0.27	1.07	0.16	1.09	2.17	0.11	0.88	3.18
E5	0.01	0.09	0.26	0.06	0.21	0.91	0.06	1.05	2.19	0.07	0.63	2.01
E56	0.00	0.07	0.29	0.03	0.24	0.96	0.26	1.07	2.63	0.08	0.75	2.74
E123456	0.02	0.60	0.87	0.20	0.67	2.18	0.37	1.82	4.02	0.25	3.18	7.15

3.2.2.6 Vitesse de rotation de la turbine N_2

Pour le paramètre N_2 du Tableau 3.8, on remarque que l'erreur moyenne μ est inférieure à 0.5 % et la déviation standard σ est inférieure à 1 %, tandis que l'erreur maximale est de l'ordre de 8 % pour le régime de décollage. Par conséquent le paramètre N_2 est validé pour tous les modèles et phases de vol.

Tableau 3.8 Erreurs moyennes, déviations standards et erreurs maximales de la vitesse de rotation de la turbine N_2 en [%] pour les régimes de décollage, de montée, de croisière et de descente

	Régime de Décollage			Régime de montée			Régime de croisière			Régime de descente		
Modèle	μ	σ	max	μ	σ	max	μ	σ	max	μ	σ	max
E56	0.001	0.12	5.39	0.10	0.19	0.84	0.26	0.90	1.40	0.01	0.23	0.80
E123456	0.09	0.39	7.77	0.11	0.52	1.88	0.29	1.01	1.95	0.14	1.01	2.27

3.2.2.7 Poussée nette F_N

Pour le paramètre de poussée F_N du Tableau 3.9, on remarque que pour la plupart des modèles (autre que E1, E2, E4 et E13), l'erreur moyenne μ est inférieure à 1 % et la déviation standard

σ est inférieure à 5 % pour les phases de décollage, de montée, et de croisière. Pour les modèles E1, E2, E4 et E13, la déviation standard σ lors de la descente est 30 fois supérieure à la tolérance de 5 % pour le modèle E1, tandis que pour les modèles E1 et E4, la moyenne μ est largement supérieure de la limite de 5 %.

Tableau 3.9 Erreurs moyennes, déviations standards et erreurs maximales de la poussée F_N en [%] pour les régimes de décollage, de montée de croisière et de descente

	Régime de décollage			Régime de montée			Régime de croisière			Régime de descente		
Modèle	μ	σ	max	μ	σ	max	μ	σ	max	μ	σ	max
E1	0.66	2.14	8.53	0.84	5.77	23.8	0.64	11.2	33.1	27.1	154	550
E12	0.05	0.22	0.88	0.07	0.38	1.58	0.32	3.44	9.78	0.72	7.47	30.5
E12345	0.06	0.18	0.70	0.17	0.61	2.26	0.03	1.41	3.78	0.18	7.45	28.5
E14	0.10	0.24	0.88	0.10	0.55	2.27	0.23	3.40	11.7	1.64	12.2	49.1
E15	0.07	0.25	0.85	0.08	0.53	2.04	0.15	2.67	5.51	1.73	8.03	32.6
E5	0.05	0.29	0.59	0.25	0.69	2.50	0.22	4.17	12.3	1.51	10.8	42.2
E56	0.03	0.26	1.03	0.20	0.78	2.79	0.13	3.77	10.1	0.67	9.59	38.4
E1234	0.06	0.22	0.89	0.08	0.47	1.83	0.00	1.18	3.96	1.51	13.8	56.0
E123456	0.07	0.18	0.52	0.04	0.45	1.75	0.01	0.99	2.59	1.46	8.74	36.5
S3	0.04	0.22	0.71	0.02	0.61	1.54	0.20	1.55	5.66	1.32	13.5	51.8
E13	0.09	0.82	3.11	2.88	5.56	12.3	1.23	10.9	23.5	2.19	88.9	344
E2	0.03	0.30	0.70	0.12	1.14	4.43	0.78	3.19	8.32	3.68	23.9	83.2
E4	0.06	0.83	3.22	3.46	7.14	32.3	41.9	28.2	83.8	54.6	176	604

Il faudrait noter que dans le cas de notre modélisation, afin de calculer la poussée F_N , le modèle E1 utilise seulement le paramètre EPR , le modèle E2 utilise seulement le paramètre FPR , le modèle E4 utilise seulement le paramètre OPR , et le modèle E13 utilise seulement les paramètres EPR et ITT . Ces résultats démontrent que les paramètres de design du moteur (EPR , FPR , OPR et ITT) ne peuvent pas être utilisés séparément lors du calcul de la poussée, et qu'ils doivent être combinés avec au moins deux autres paramètres, comme le témoigne le modèle E1234. Cependant, le paramètre N_1 peut être utilisé seul pour prédire la poussée comme il a été mentionné dans la section 1.7.5, étant donné qu'il existe une corrélation linéaire entre la poussée corrigée $F_{N,C}$ et la vitesse de rotation de la soufflante corrigée $N_{1,C}$.

Pour les autres modèles, autres que E1, E2, E4 et E13, pour la phase de descente, on peut remarquer que l'erreur moyenne μ est près de 2 % et la déviation standard σ atteint les 15 %. Ces valeurs peuvent s'expliquer par le fait que lors de la descente, la poussée peut diminuer jusqu'à 0 lbf et changer de signe. Cependant, comme il est montré sur la Figure 2.34, l'erreur résiduelle est près de 50 lbf, soit 0.4 % de la poussée maximale du moteur (13 790 lbf), ce qui est relativement négligeable. Par conséquent, la poussée est validée pour tous les modèles autres que E1, E2, E4 et E13 et pour toutes les phases de vol.

3.2.2.8 Débit de carburant W_F

Pour le débit de carburant W_F du Tableau 3.10, on remarque que pour la plupart des modèles (autre que E1, E2, E4 et E13), l'erreur moyenne μ est inférieure à 1 % et la déviation standard σ est inférieure à 5 % pour les phases de décollage, de montée, et de croisière et de descente.

Tableau 3.10 Erreurs moyennes, déviations standards et erreurs maximales du débit de carburant W_F en [%] pour les régimes de décollage, de montée de croisière et de descente

	Régime de décollage			Régime de montée			Régime de croisière			Régime de descente		
Modèle	μ	σ	max	μ	σ	max	μ	σ	max	μ	σ	max
E1	0.35	2.77	9.56	1.25	4.86	20.5	2.15	8.65	24.6	4.70	29.0	117
E12	0.01	0.23	0.90	0.12	0.57	2.18	0.29	2.70	7.37	0.09	1.27	5.04
E12345	0.01	0.20	0.77	0.23	0.79	2.95	0.01	1.15	3.52	0.05	1.64	6.57
E14	0.02	0.24	0.93	0.36	0.99	4.34	0.04	2.66	7.76	0.08	1.94	7.59
E15	0.03	0.24	0.92	0.03	0.60	2.38	0.03	1.95	6.08	0.07	1.86	6.43
E5	0.01	0.31	1.16	0.36	1.02	3.68	0.14	4.10	12.4	0.23	1.77	6.78
E56	0.03	0.33	1.15	0.11	0.90	3.23	0.14	3.15	7.43	0.04	2.02	7.28
E1234	0.00	0.28	0.97	0.00	0.56	2.15	0.09	1.28	3.70	0.07	1.97	7.76
E123456	0.00	0.21	0.85	0.08	0.48	1.81	0.02	1.09	3.12	0.03	1.18	4.37
S3	0.03	0.24	0.76	0.08	0.67	2.73	0.01	1.45	4.71	0.01	1.85	6.50
E13	0.48	2.05	4.87	2.40	4.79	15.7	1.22	8.08	18.6	0.04	8.80	33.6
E2	0.02	0.28	0.82	0.21	1.19	4.19	0.62	2.46	6.42	0.52	3.65	12.0
E4	0.28	1.35	2.30	2.63	5.52	24.3	32.5	21.7	66.4	20.0	28.7	39.9

Pour le modèle E1, la déviation standard σ est 30 fois supérieure à la tolérance de 5 % lors de la descente, tandis que pour les modèles E1 et E4, la moyenne est 6 fois supérieure à la limite de 5 %. De la même manière que pour le paramètre de poussée, ces résultats démontrent que les paramètres de design du moteur (EPR , FPR , OPR et ITT) ne peuvent pas être utilisés séparément lors du calcul du débit de carburant, et ils doivent être combinés avec au moins deux autres paramètres, comme le témoigne le modèle E1234.

Le paramètre N_1 peut être utilisé seul pour prédire le débit de carburant comme il a été mentionné dans la section 1.7.5, étant donné qu'il existe une corrélation linéaire entre le débit de carburant corrigé $W_{F,C}$ et la vitesse de rotation de la soufflante corrigée $N_{1,C}$.

Pour la phase de descente, la déviation standard σ est près de 2 %. Comme il est montré sur la Figure 2.35, l'erreur résiduelle est près de 50 lb/hr lors de la descente, soit 0.5 % du débit de carburant maximal mesuré (10 000 lb/hr), ce qui est relativement négligeable. Par conséquent, le débit de carburant W_F est validé pour tous les modèles autres que E1, E2, E4 et E13 et pour toutes les phases de vol.

3.3 Critères de sélection du modèle

Avant de procéder à la simulation, il est nécessaire de sélectionner les modèles de performances. La section précédente a donné au lecteur une idée des modèles adaptés pour la simulation. Cependant, il est possible d'évaluer chaque modèle en quantifiant sa qualité. Pour ce faire, 3 critères ont été suggérés dont 1) la racine des sommes quadratiques (en anglais *Root Sum Square* ou *RSS*), 2) le critère d'information de Akaike *AIC* et 3) le critère d'information Bayésien *BIC*.

La **racine des sommes quadratiques** RSS peut être déterminée selon la formule suivante :

$$RSS = \sqrt{\sum_1^{n_y} \sum_1^{n_{phase}} \sigma^2} \quad (3.4)$$

où σ_i est la déviation standard, n_y est le nombre de sorties et n_{phase} est le nombre de phases de vol. Dans notre cas, le nombre de phases de vol est $n_{phase} = 4$, et on ne considèrera que les paramètres de performances F_N et W_F donc $n_y = 2$.

La limite RSS_{FAA} est calculée selon le critère de 5 % de la FAA sur la déviation standard pour toutes les phases de vol et pour les paramètres F_N et W_F . Par contre, on considèrera une tolérance de 15 % sur la poussée F_N lors de la descente, étant donné que lors de la descente, la poussée est négligeable, voire nulle, et que l'erreur de prédiction est élevée. Par conséquent, la limite à ne pas dépasser est $RSS_{FAA} = 20$.

Critère d'information de Akaike et Critère d'information Bayésien

Dans la théorie de l'information, le critère d'information de Akaike (en anglais *Akaike's Information Criterion* ou *AIC*) permet de quantifier la qualité d'un modèle afin de le comparer par rapport à un autre.

Pour deux modèles établis à partir d'une même base de données, l'estimation est toujours meilleure lorsqu'on augmente l'ordre de grandeur du modèle, c'est-à-dire lorsqu'on augmente le nombre de paramètres et la flexibilité du modèle en termes de structure. Selon la théorie de Akaike, le meilleur modèle est celui avec la plus petite valeur de *AIC*. Le paramètre *AIC* est un paramètre de logistique, et il peut être calculé comme suit :

$$AIC = N * \log\left(\frac{1}{N} \sum_{i=1}^N \hat{L}(t_i|\theta)\right) + 2n_p + N * (n_y * (\log(2\pi) + 1)) \quad (3.5)$$

où N est le nombre d'observations dans la base de données pour l'entraînement du réseau de neurones, $\hat{L}$ est l'estimation de la fonction de coût, θ est le vecteur des paramètres estimés, n_p est le nombre total de paramètres et il correspond à la somme du nombre d'éléments des matrices de poids w et de biais b et n_y est le nombre de sorties du modèle.

Dans notre cas, la fonction de coût $\hat{L}$ pour l'entraînement des réseaux de neurones est la somme des erreurs quadratiques (en anglais *Sum Squared Error*). Le terme $\frac{1}{N} \sum_{i=1}^N \hat{L}(t_i|\theta)$ correspond donc à la moyenne des erreurs quadratiques MSE pour toutes les observations N . On peut alors réécrire le critère AIC comme suit :

$$AIC = N * \log(MSE) + 2n_p + N * (n_y * (\log(2\pi) + 1)) \quad (3.6)$$

où MSE est la moyenne des erreurs quadratiques calculée lors de l'apprentissage du réseau de neurones (voir section 2.3.3).

Un troisième critère peut être utilisé pour évaluer la qualité du modèle. Il s'agit du critère d'information Bayésien ou BIC . Le critère BIC est similaire à celui de AIC et il est défini comme suit :

$$BIC = N * \log\left(\frac{1}{N} \sum_{i=1}^N \hat{L}(t_i|\theta)\right) + 2n_p + N * (n_y * (\log(2\pi) + 1)) + n_p * \log(N) \quad (3.7)$$

La différence entre les deux critères AIC et BIC est que le critère BIC accorde plus d'importance au nombre d'observations N que le critère AIC . On peut remarquer le terme $n_p * \log(N)$ qui n'est pas présent dans AIC . Cependant, l'interprétation du critère est la même que celle de AIC , c'est-à-dire que le meilleur modèle est celui avec la plus petite valeur de BIC .

Tableau 3.11 Critères de sélection RSS ,
et critères d'information AIC et BIC de
chaque modèle de réseau de neurones

Modèles	RSS	AIC	BIC
E12345	7.91	548	1 715
E12	8.78	413	1 236
E123456	8.97	601	1 875
E15	8.92	314	980
E56	11.0	359	1 120
E5	12.5	326	1 021
E14	13.2	455	1 402
S3	13.8	201	643
E1234	14.1	443	1 395
Critère de la FAA : $RSS_{FAA} = 20$			
E2	24.6	241	729
E13	90.7	356	1 142
E1	158	355	1 071
E4	182	264	851

Dans le cadre de ce mémoire, les trois critères RSS , AIC et BIC seront appliqués pour évaluer et comparer les modèles de réseaux de neurones. Le Tableau 3.11 montre le classement des modèles selon les valeurs de RSS , AIC et BIC . Comme prévu dans la section précédente, les modèles E1, E2, E4 et E13 ne satisfont pas les critères de la FAA et ne seront donc pas considérés pour la simulation.

On remarque également que plus le nombre de sorties est élevé, plus les valeurs de AIC et BIC sont plus grandes. Ceci s'explique du fait que le nombre de paramètres (poids w et biais b) augmente lorsque le nombre de sorties augmente. Par conséquent, selon la théorie de Akaike, le meilleur modèle est le modèle S3 qui n'inclue pas le modèle de moteur ECS-ENG, mais calcule directement la poussée et le débit de carburant à partir du nombre de Mach, du TLA et des conditions atmosphériques. Ensuite, les modèles les meilleurs modèles sont, en ordre, les modèles E1234, E12345 et E123456.

Ainsi, les modèles S3, E12, E5, E56, E1234, E12345 et E123456 sont capables de satisfaire les critères de la FAA, et seront donc considérés pour les simulations dans la prochaine section. Les modèles sélectionnés ont été simulés pour différentes conditions de vol, dont le décollage, la montée, la croisière avec une panne du moteur, la descente et les changements de régime comme présenté dans la section 3.4.

3.4 Simulation du modèle complet pour différentes applications

Les sections 3.4.2 à 3.4.4 montrent les différents tests avec d'excellents résultats pour les phases de décollage, de montée, de croisière et de descente. Les paramètres estimés par les modèles sont comparés avec ceux mesurés par le simulateur de vol. La tolérance de $\pm 5\%$ est représentée par deux lignes pointillées en noir, ce qui permet de conclure sur la précision de chaque modèle de performance.

3.4.1 Résultats de la validation du modèle complet

Le Tableau 3.12 récapitule les résultats obtenus pour le modèle complet modélisé dans le CHAPITRE 2 et comprenant toutes les sorties $EPR, FPR, OPR, ITT, N_1, N_2$ ainsi que les paramètres de performance F_N et W_F . Les résultats sont très bons, étant donné que la moyenne des paramètres est centrée sur 0, tandis que les déviations standards sont inférieures à 5 %, sauf pour la poussée F_N lors de la descente, où la déviation standard est de 8.74 %.

Tableau 3.12 Erreurs moyennes, déviations standards et erreurs maximales en [%] des sorties du modèle complet E123456 pour les régimes de décollage, de montée, de croisière et de descente

	Régime de décollage			Régime de montée			Régime de croisière			Régime de descente		
Sortie	μ	σ	max	μ	σ	$max.$	μ	σ	max	μ	σ	$max.$
EPR	0.00	0.21	0.85	0.08	0.48	1.81	0.02	1.09	3.12	0.03	1.18	4.37
FPR	0.21	1.23	3.86	1.01	2.26	6.48	2.31	4.56	6.11	0.19	2.11	7.05
ITT	0.16	1.43	2.97	0.16	0.94	3.55	0.84	4.10	9.86	0.07	2.88	8.60
OPR	0.12	0.37	1.60	0.15	0.73	2.79	0.48	2.74	6.48	0.03	1.07	4.02
N_1	0.02	0.60	0.87	0.20	0.67	2.18	0.37	1.82	4.02	0.25	3.18	7.15
N_2	0.09	0.39	7.77	0.11	0.52	1.88	0.29	1.01	1.95	0.14	1.01	2.27
W_F	0.00	0.21	0.85	0.08	0.48	1.81	0.02	1.09	3.12	0.03	1.18	4.37
F_N	0.07	0.18	0.52	0.04	0.45	1.75	0.01	0.99	2.59	1.46	8.74	36.5

3.4.2 Simulation en phase de décollage

La Figure 3.1 montre un exemple de simulation pour la phase de décollage. La simulation dure 70 secondes durant lesquels l'avion monte jusqu'à 1 500 pi au-dessus du sol. La phase de décollage est divisée en deux segments : la phase d'accélération au sol jusqu'à 30 secondes, et la partie en vol à partir de 30 secondes jusqu'à 70 secondes.

Les Figure 3.2 et Figure 3.3 les résultats de la simulation pour les sorties des modèles sélectionnés. Comme on peut le voir, chaque modèle est capable de prédire les paramètres du moteur, dont les ratios de pressions, EPR , FPR et OPR , la température à l'entrée de la turbine ITT , la vitesse de rotation de la soufflante N_1 , la poussée F_N et le débit de carburant W_F avec une erreur inférieure à 5 %. Les courbes des modèles se superposent avec celles du simulateur de vol pour les paramètres EPR , F_N et W_F . Cependant, la vitesse de rotation de la turbine N_2 est légèrement supérieure aux limites de 5 %. Ceci peut s'expliquer par le fait que la poussée est essentiellement délivrée par la soufflante (80 %) lors du décollage. La vitesse de rotation de la turbine est quant à elle responsable de 20 % de la poussée. Donc une faible corrélation sur la vitesse de rotation de la turbine N_2 est négligeable. Les résultats de la simulation

correspondent aux résultats de statistique obtenus dans la section précédente pour les 42 tests de décollage. Les modèles sont donc validés pour la phase de décollage.

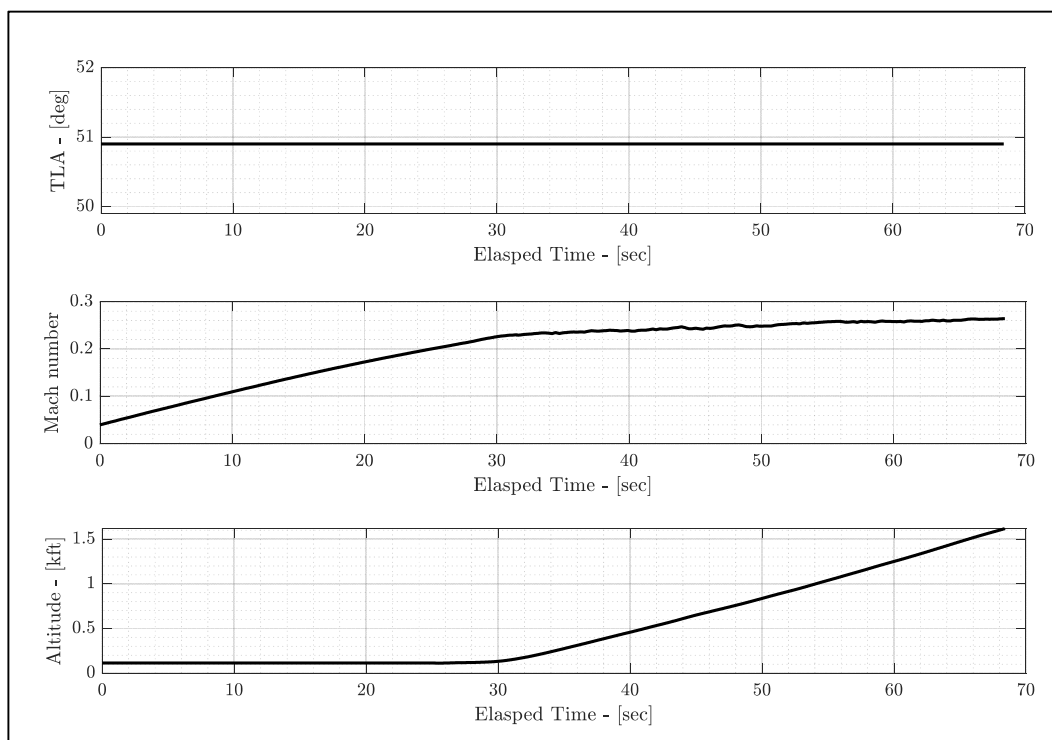


Figure 3.1 Conditions de vol lors de la simulation en phase de décollage

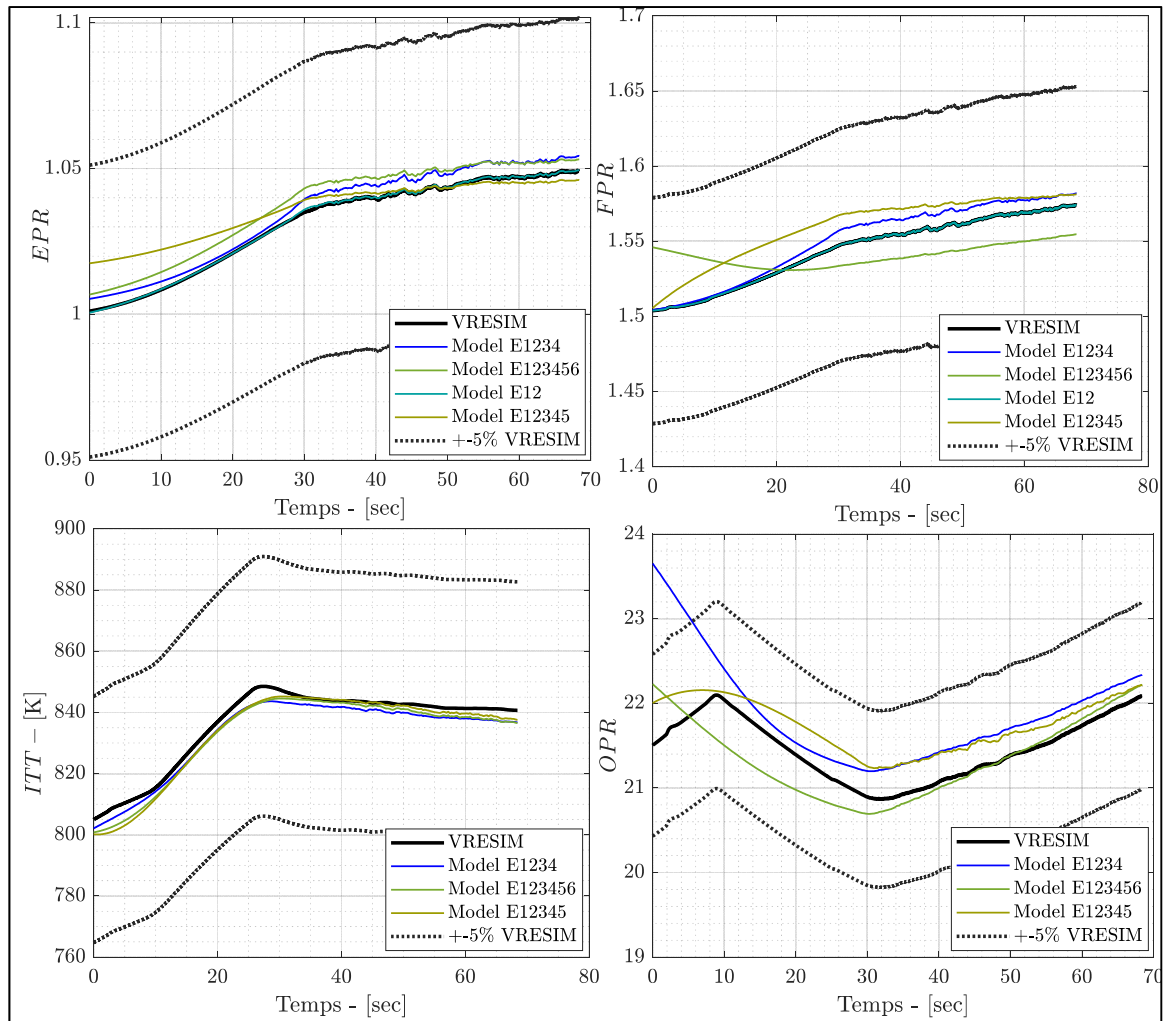


Figure 3.2 Sorties des modèles EPR , FPR , ITT et OPR lors de la simulation en phase de décollage

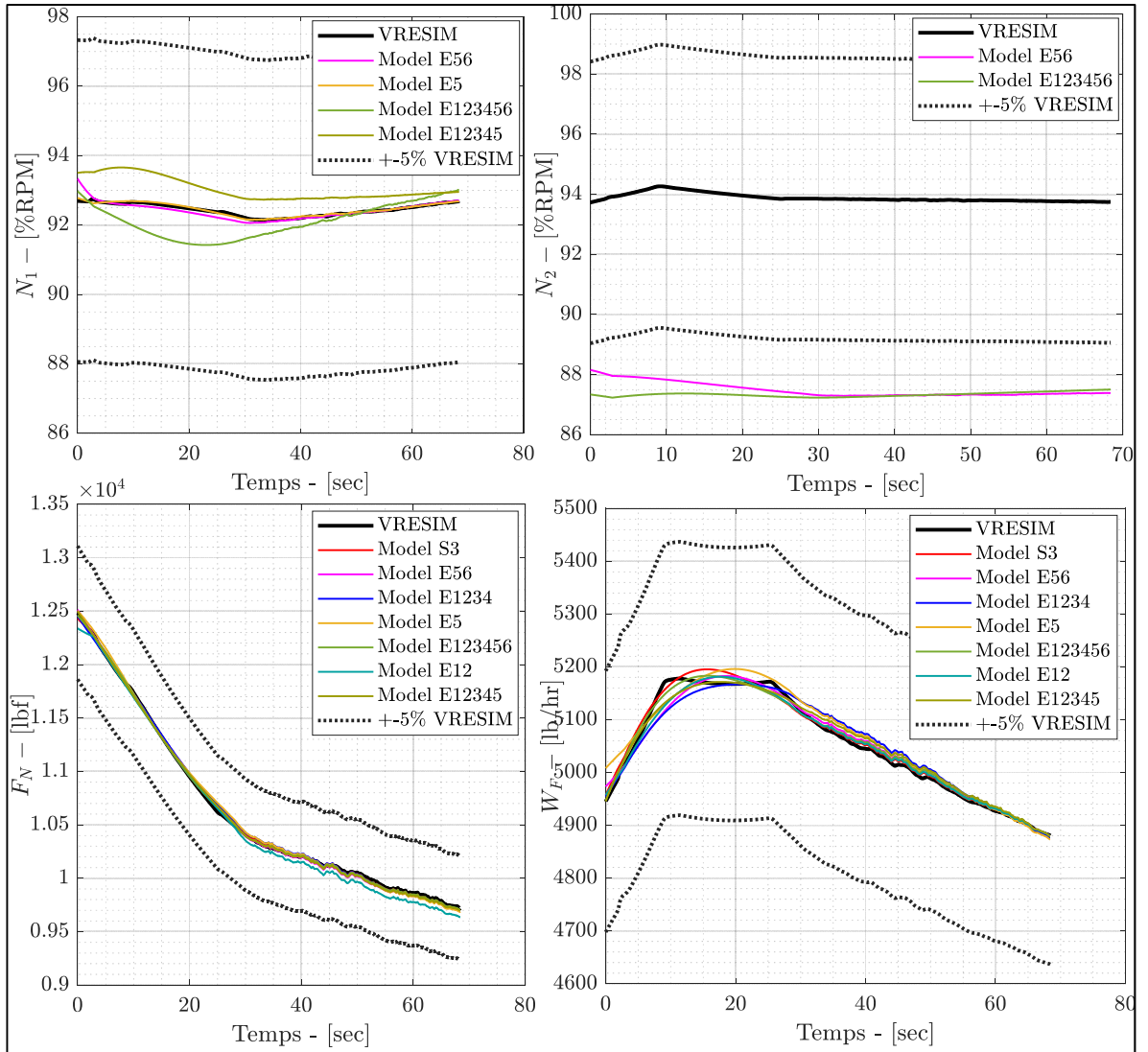


Figure 3.3 Sorties des modèles N_1 , N_2 , F_N et W_F lors de la simulation en phase de décollage

3.4.3 Simulation en phase de montée

Les Figure 3.4 à Figure 3.6 montrent un exemple de simulation pour la phase de montée. La manette des gaz est placée sur la position MCLB (45°) et l'altitude de l'avion varie entre 25 000 pi et 40 000 pi. La phase de montée est réalisée en continu (sans palier) pendant environ 8 minutes.

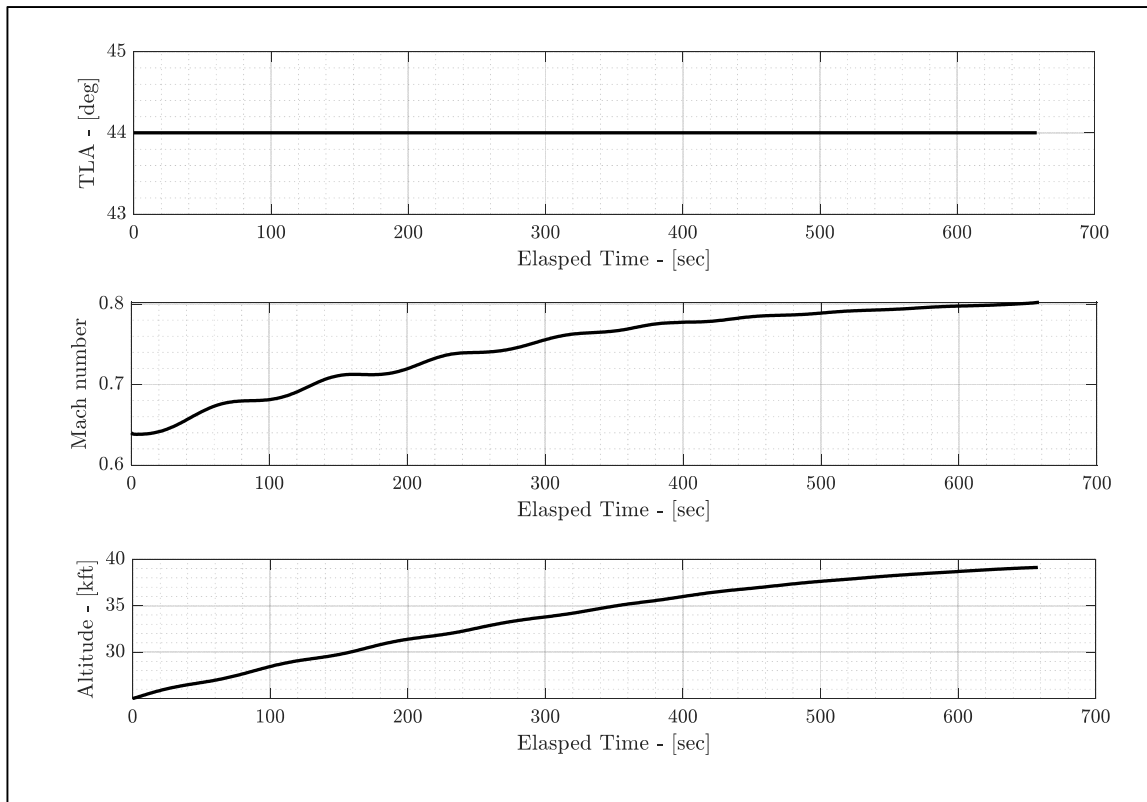


Figure 3.4 Conditions de vol lors de la simulation en phase de montée

De la même manière que pour la phase de décollage, chaque modèle est capable de prédire les paramètres du moteur, dont les ratios de pressions EPR , FPR et OPR , la température à l'entrée de la turbine ITT , la vitesse de rotation de la soufflante N_1 , la poussée F_N et le débit de carburant W_F avec une erreur inférieure à 5 %. Contrairement à la phase de décollage, la prédiction de la vitesse de rotation de la turbine N_2 est comprise dans la limite de 5 %. Aussi, les courbes des modèles se superposent avec celles du simulateur pour les paramètres OPR , F_N et W_F . Les résultats de la simulation correspondent aux résultats de statistiques obtenus dans la section précédente pour les 47 tests de montée. Par conséquent, les modèles sont validés pour la phase de montée.

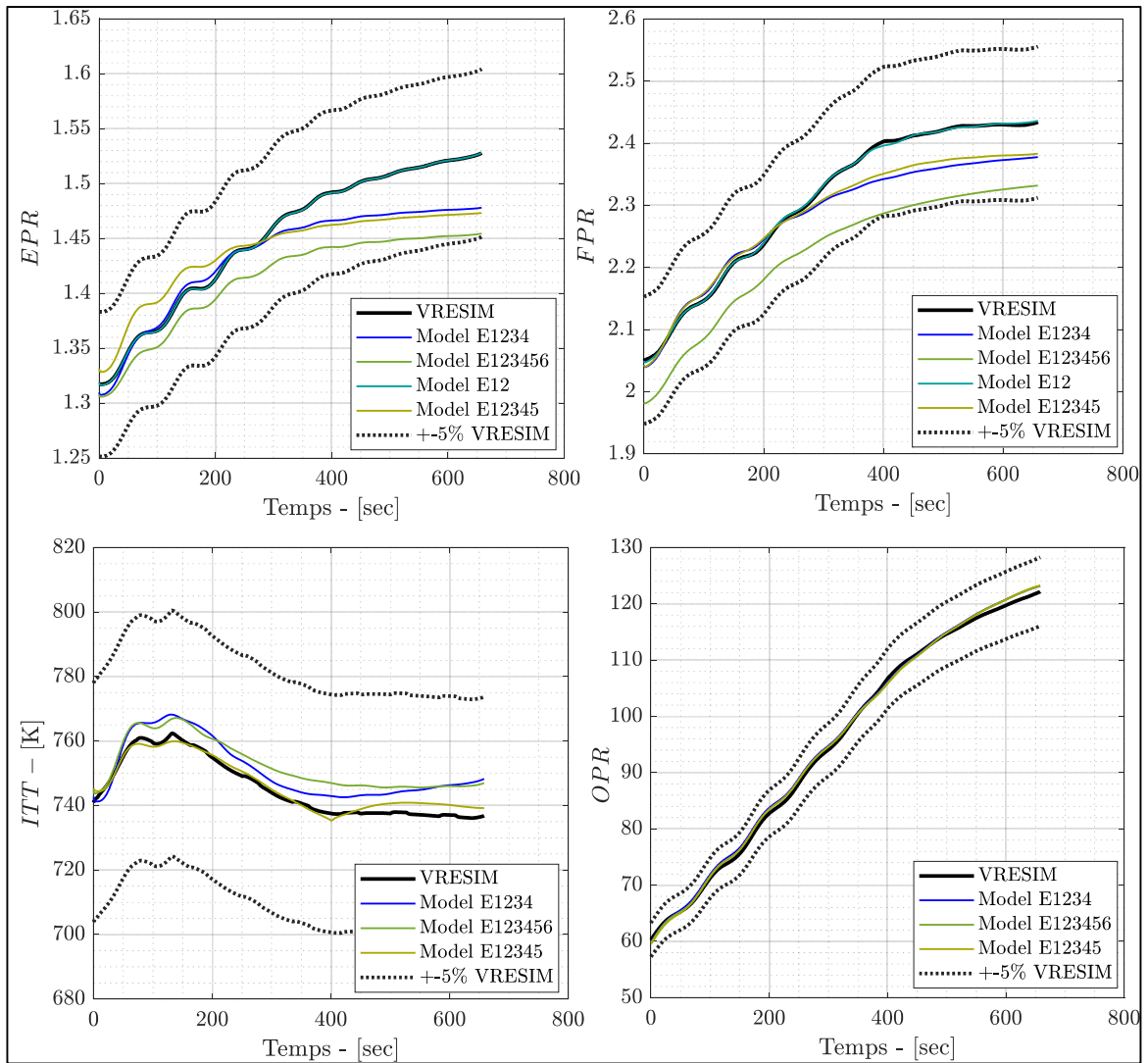


Figure 3.5 Sorties des modèles EPR , FPR , ITT et OPR lors de la simulation en phase de montée

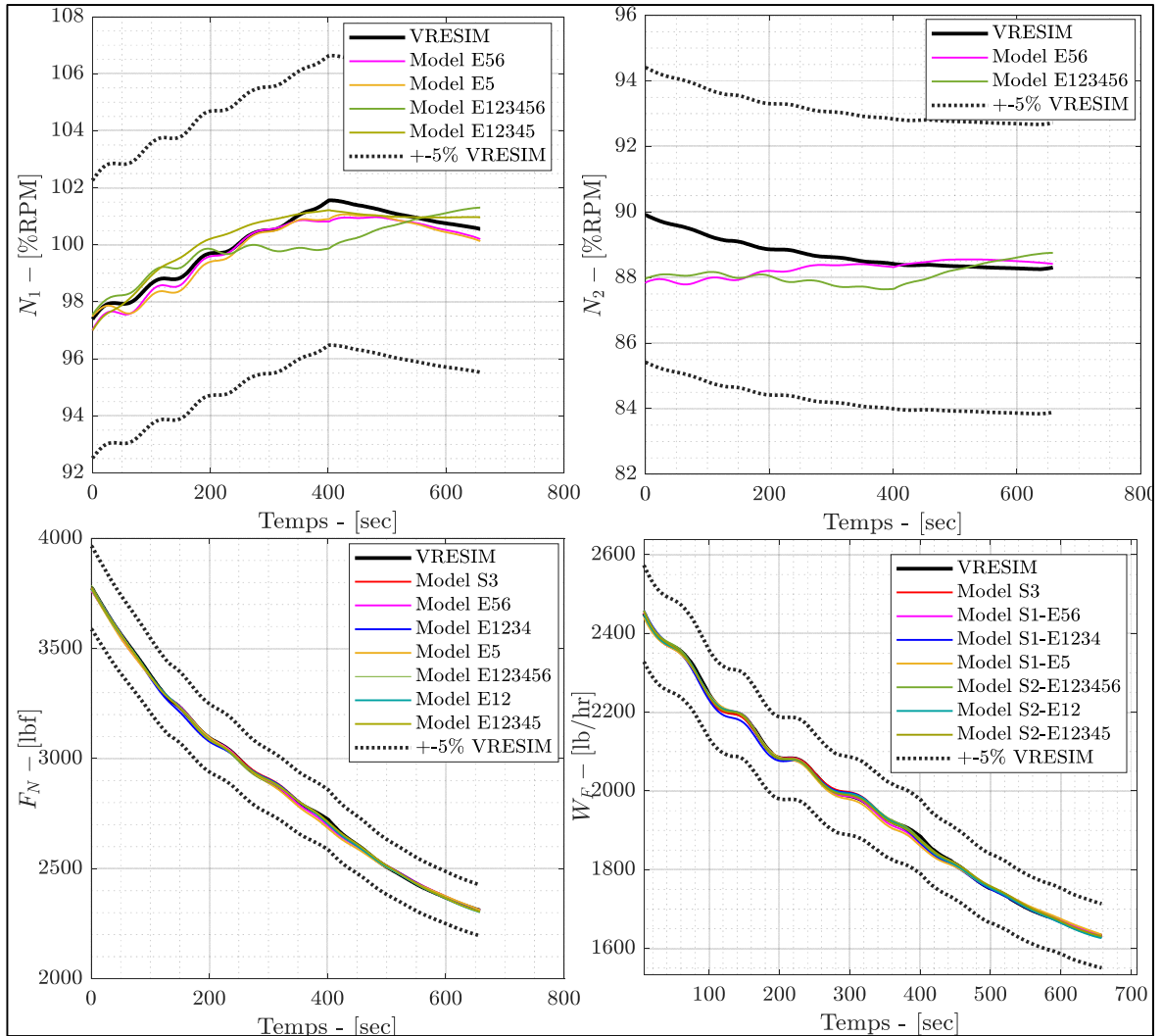


Figure 3.6 Sorties des modèles N_1 , N_2 , F_N et W_F lors de la simulation en phase de montée

3.4.4 Simulation du modèle en phase de descente

Les Figure 3.7 à Figure 3.9 montrent un exemple de simulation pour la phase de descente. La manette des gaz est placée sur la position IDLE (00°) et l'altitude de l'avion varie entre 25 000 pi et 1 500 pi. La phase de descente est réalisée pendant environ 8 minutes.

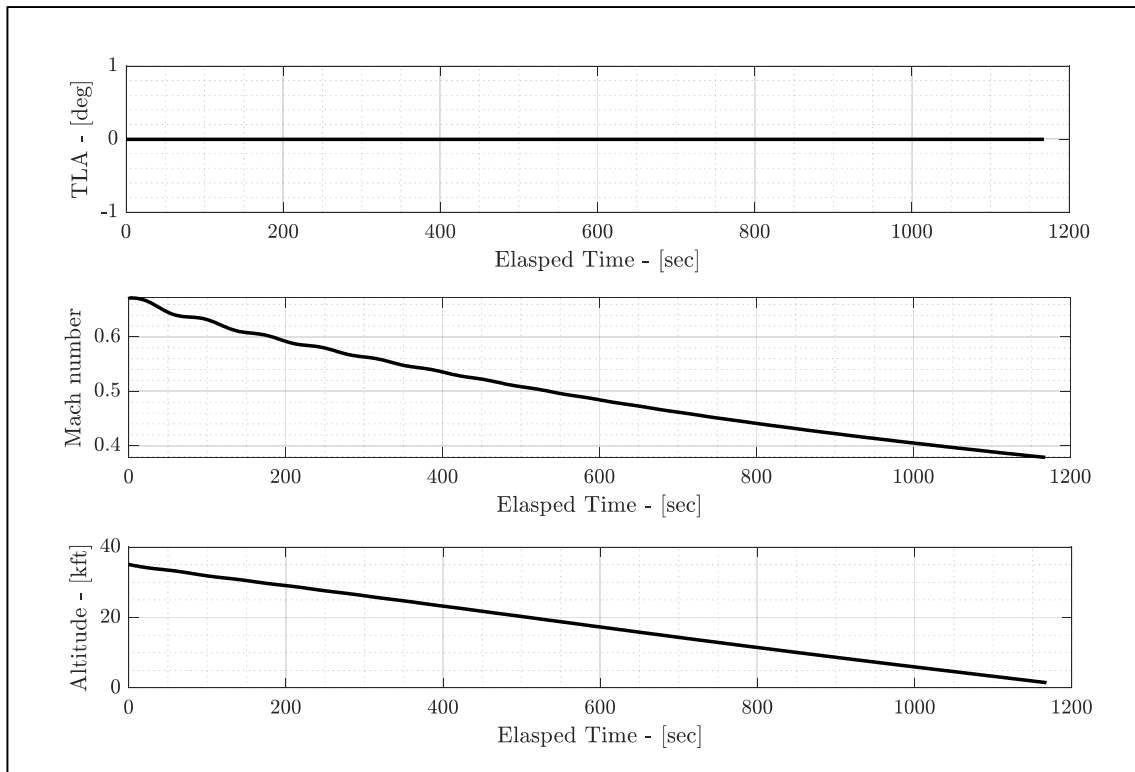


Figure 3.7 Conditions de vol lors de la simulation en phase de descente

D'après les Figure 3.8 et Figure 3.9, chaque modèle est capable de prédire les paramètres du moteur, dont les ratios de pression, EPR , FPR et OPR , la température à l'entrée de la turbine ITT , la vitesse de rotation de la soufflante N_1 , la vitesse de rotation de la turbine N_2 , la poussée F_N et le débit de carburant W_F avec une erreur inférieure à 5 %. Les courbes des modèles se superposent avec celles du simulateur pour les paramètres OPR et ITT . Tel qu'il est dit précédemment dans la section 3.2, l'erreur entre les modèles et les données du simulateur de vol sont plus ou moins élevée pour la poussée F_N , et le débit de carburant W_F . Les courbes ne se superposent pas avec celles du simulateur et l'erreur est supérieure à 5 %. Ceci est dû au fait que les valeurs de poussée sont petites par rapport à la valeur maximale de poussée (13 790 lb) et du débit de carburant (10 000 lb/hr). Aussi, l'erreur résiduelle est de l'ordre de 50 lbf (soit 0.36 % de la poussée maximale) ce qui est négligeable lors de la descente, comme il est montré sur la Figure 2.34. Dans tous les cas, les modèles estiment correctement la tendance générale des données du simulateur de vol. Les résultats de la simulation confirment les résultats de

statistiques obtenus dans la section précédente pour les 77 tests de descente. Par conséquent, les modèles sont validés pour la phase de descente.

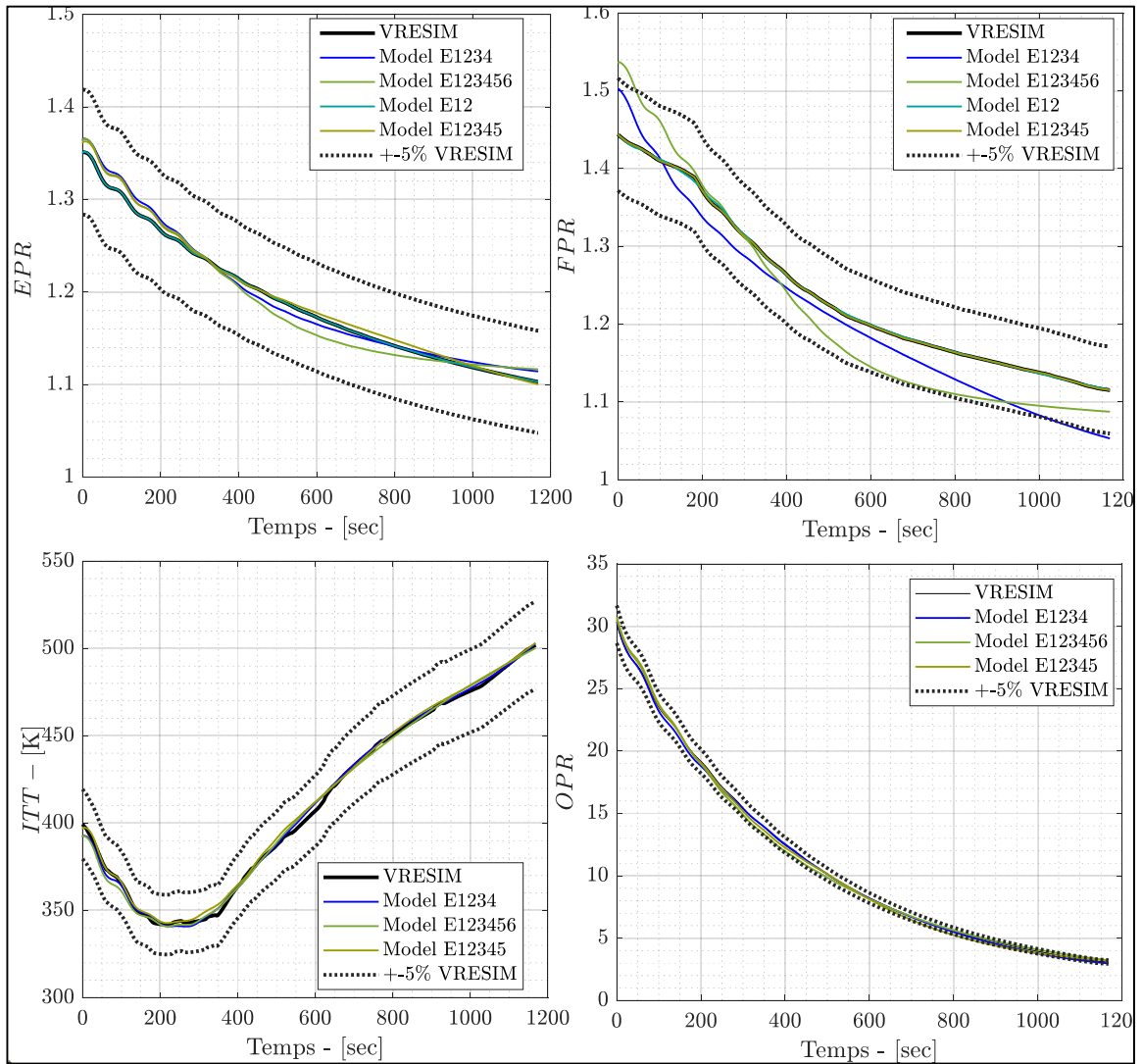


Figure 3.8 Sorties des modèles EPR , FPR , ITT et OPR lors de la simulation en phase de descente

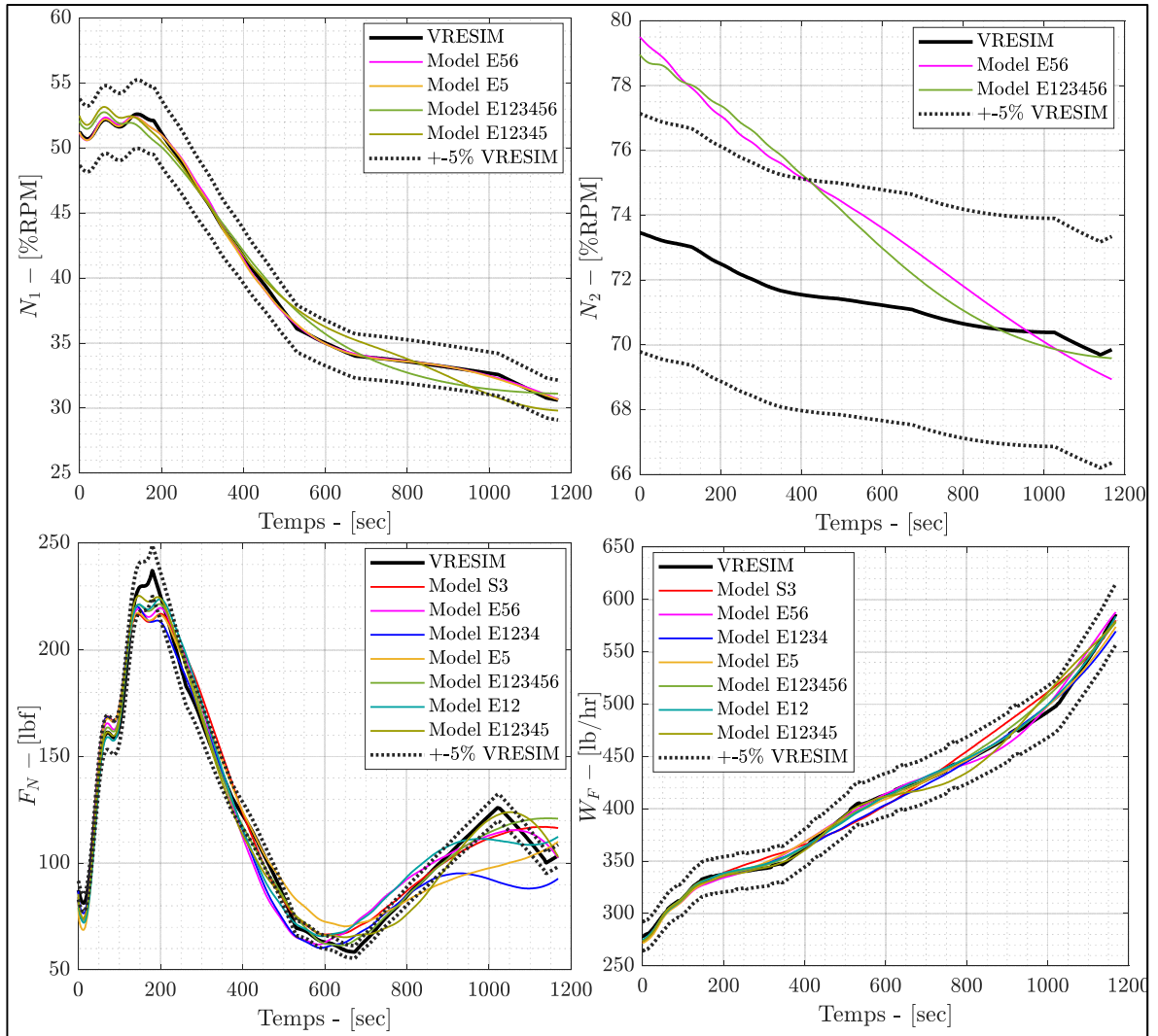


Figure 3.9 Sorties des modèles N_1 , N_2 , F_N et W_F lors de la simulation en phase de descente

3.4.5 Simulation en phase de croisière lors d'une panne

La validation du modèle pour la phase de croisière a été réalisée pour 154 tests de vol avec une altitude fixe entre 5 000 et 41 000 pieds et un nombre de Mach fixe entre 0.4 et 0.82. Les erreurs relatives du modèle sont résumées dans les Tableau 3.3 à Tableau 3.9. Comme on peut le remarquer, la déviation standard est moins de 5 % pour tous les paramètres des modèles sélectionnés (à l'exception des modèles E1, E2, E4 et E13).

La Figure 3.10 montre un exemple de simulation pour la phase de croisière, durant laquelle une panne a été appliquée. La simulation a pour objectif de démontrer la nécessité et l'utilité du modèle lors d'une panne. Dans notre cas la panne a été appliquée sur le compresseur à haute pression (HPC) à partir de 65 secondes. La manette des gaz est placée à 30° et l'avion vole à une altitude de 10 000 pi. La phase de croisière est réalisée pendant 100 secondes, et les simulations sont réalisées pour le modèle complet, dont les résultats sont présentés dans le Tableau 3.12.

Comme on peut le remarquer, entre 0 et 65 secondes, le modèle complet E123456 est capable de prédire tous les paramètres du moteur. Les courbes du modèle se superposent avec celles du simulateur pour les paramètres ITT , N_1 , F_N et W_F , tandis qu'on remarque une légère erreur pour les autres paramètres. Les erreurs entre 0 et 65 secondes sont des erreurs de modélisation, étant donné que le modèle de performance est statique.

Après l'application de la panne à partir de 65 secondes, on remarque une différence significative entre les sorties du modèle et celles du simulateur de vol. Cet écart est dû à la panne, et elle permet de diagnostiquer le moteur. Il est à noter qu'il est important d'avoir un modèle complet, comme le modèle E123456 capable de prédire tous les paramètres du moteur, étant donné qu'un modèle partiel, comme les modèles E15 ou E56, ne pourrait pas évaluer ou détecter correctement la panne.

Dans l'exemple de la Figure 3.10, les mesures de N_1 , FPR et F_N seuls ne peuvent pas confirmer la présence d'une panne, étant donné que ΔN_1 , ΔFPR et ΔF_N sont quasiment nulles, et que les erreurs s'apparentent à des bruits et ne sont pas faciles à détecter. Par contre les différences au niveau des paramètres OPR , ITT , N_2 et W_F témoignent la présence d'une panne, étant donné que les erreurs ΔOPR , ΔITT , ΔN_2 et ΔW_F sont statiquement stables. Des exemples similaires sont disponibles dans la littérature, avec des études plus approfondies sur la localisation de la panne dans le moteur.

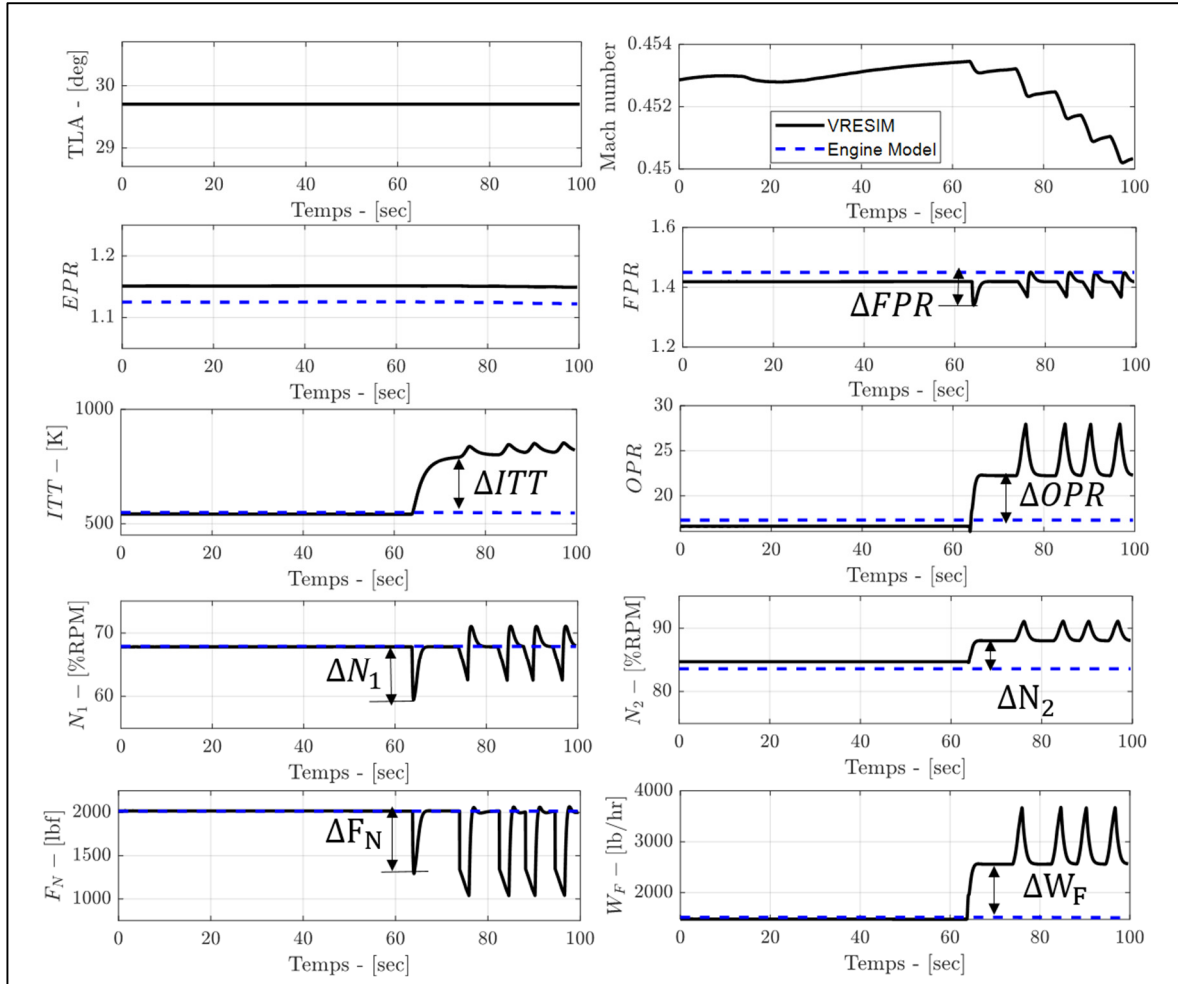


Figure 3.10 Résultats de la simulation du modèle complet lors de la phase de croisière et dans une situation de panne

3.4.6 Changement de modes de poussée

Deux simulations ont été réalisées, qui consistent à varier la manette des gaz en changeant sa position d'une position à une autre. La première simulation consiste à varier le TLA à partir de la position en régime de croisière dans la position en régime de montée (MCLB). La deuxième simulation consiste à varier le TLA à partir de la position en régime de croisière dans la position en régime de descente (IDLE).

3.4.6.1 Variation de la manette des gaz : Cas 1

La première simulation consiste à la variation de TLA à partir du régime de croisière dans le régime de montée, soit un changement de TLA de 30° à 44° (Figure 3.11). Cette simulation est réalisée à partir de 10 000 pi et dure 250 secondes. Le changement de TLA est effectué à partir de 80 secondes. Comme on peut le voir sur les Figure 3.12 et Figure 3.13, les modèles sont capables de prédire les paramètres du moteur pour le régime de croisière, mais également pour le régime de montée, avec une erreur inférieure à la limite de tolérance de 5 %.

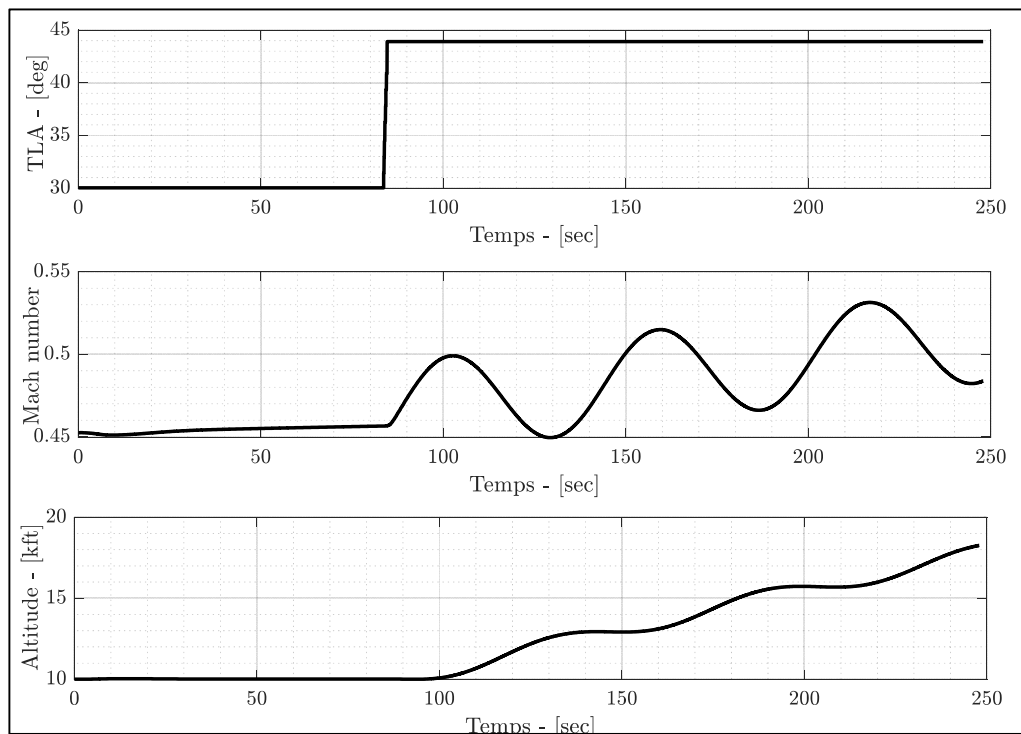


Figure 3.11 Conditions de vol lors de la variation de la manette des gaz de 30° à 44°

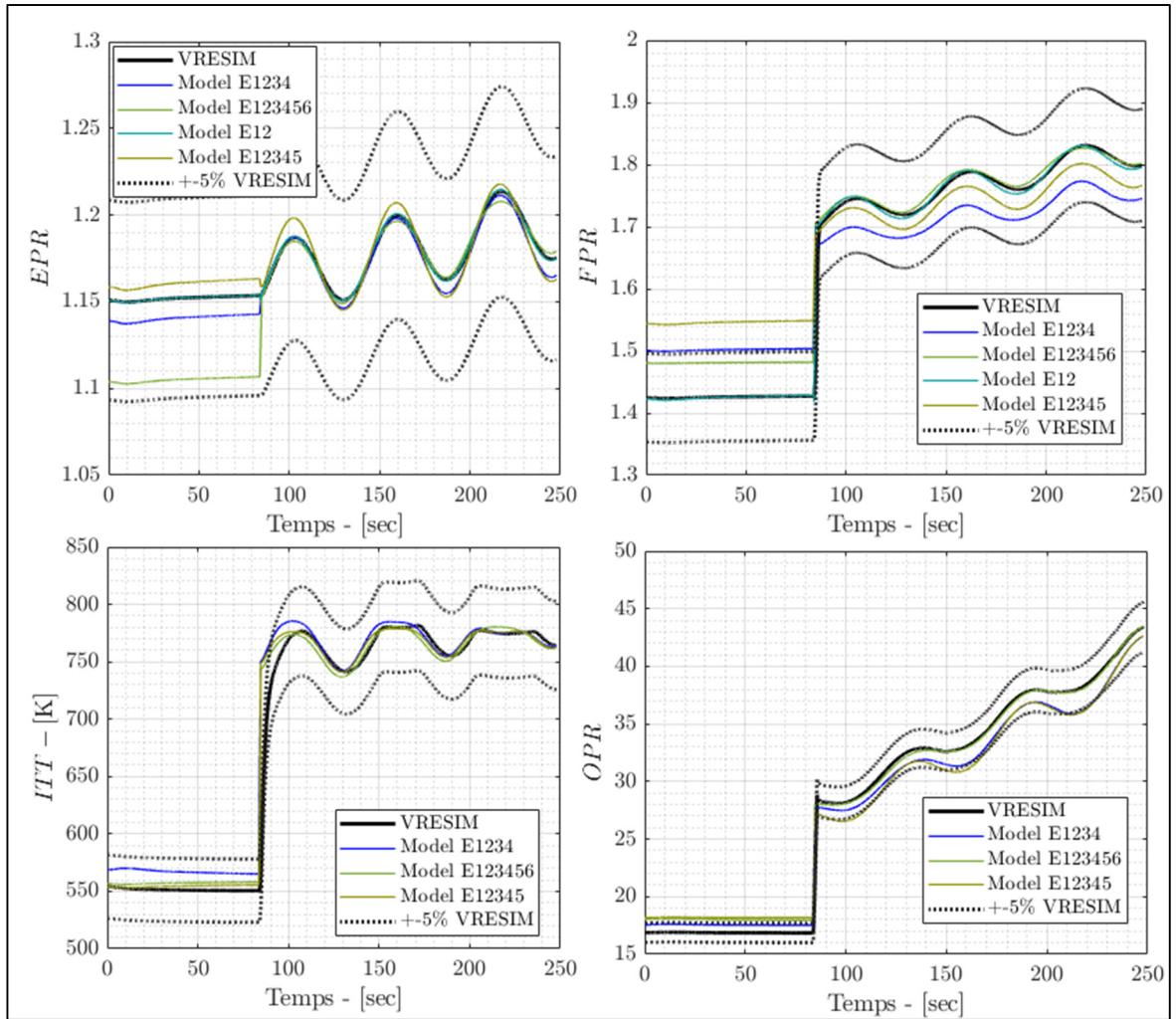


Figure 3.12 Sorties des modèles EPR , FPR , ITT et OPR lors de la variation de TLA de 30 à 44°

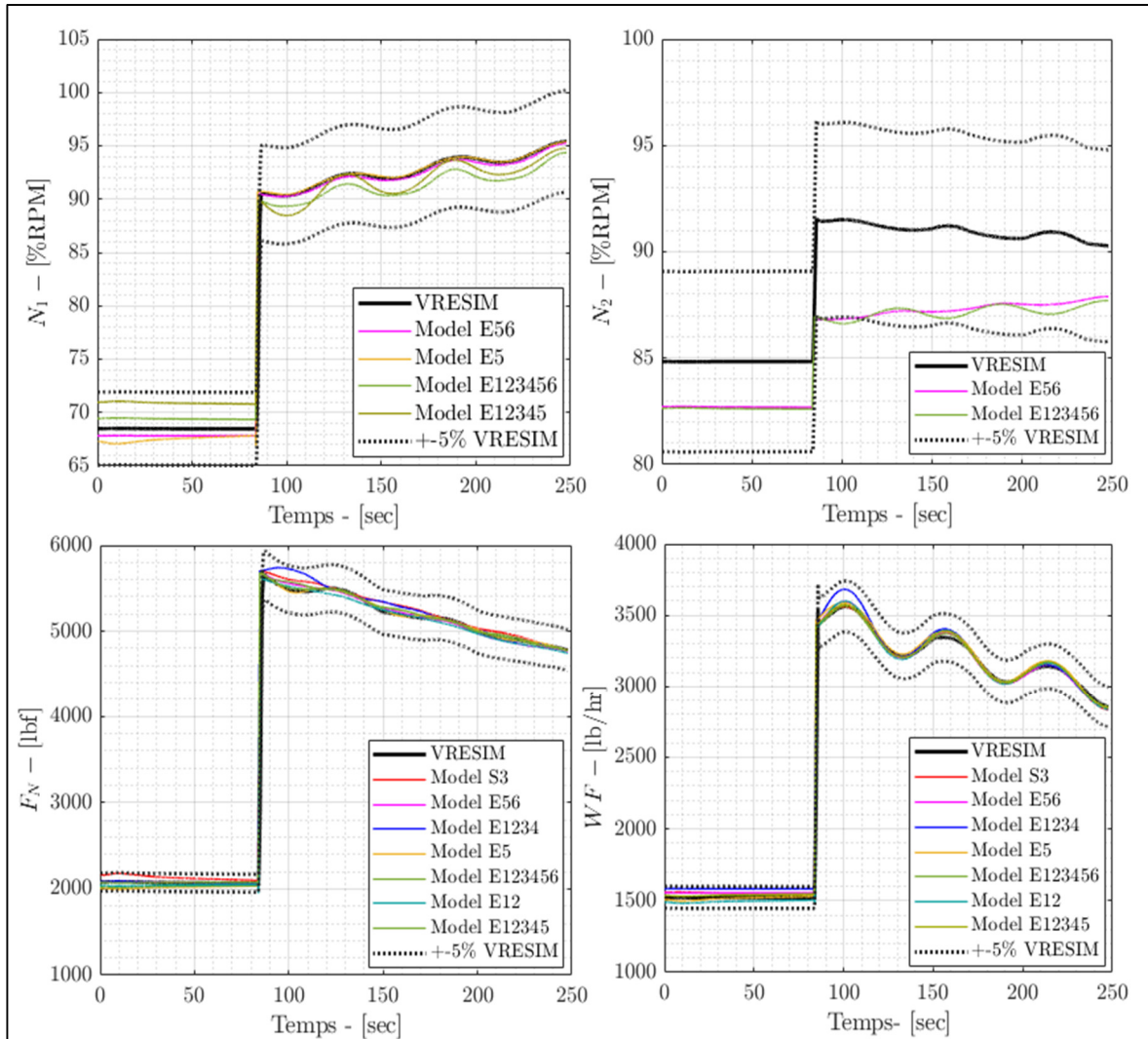


Figure 3.13 Sorties des modèles N_1 , N_2 , F_N et W_F lors de la variation de TLA de 30 à 44°

3.4.6.2 Variation de la manette des gaz : Cas 2

La deuxième simulation consiste à la variation de TLA à partir du régime de croisière dans le régime de descente, soit un changement de TLA de 30° à 0°. Cette simulation est réalisée à partir de 10 000 pi et dure 140 secondes. Le changement de TLA est effectué à partir de 40 secondes avec une lente variation sous forme de rampe (Figure 3.14). Comme on peut le voir sur les Figure 3.15 et Figure 3.16, les modèles sont capables de prédire les paramètres du moteur pour le régime de croisière, mais également pour le régime de descente, avec une erreur

inférieure à la limite de tolérance de 5 %. Par contre, les modèles ne sont pas capables de prédire la transition entre 40 et 80 secondes. Ceci s'explique par le fait que les modèles sont statiques et ne sont pas capables de modéliser la dynamique du moteur. Cependant, le modèle complet E123456 montre les meilleures caractéristiques, et il est capable de prédire la tendance générale des paramètres en régime transitoire.

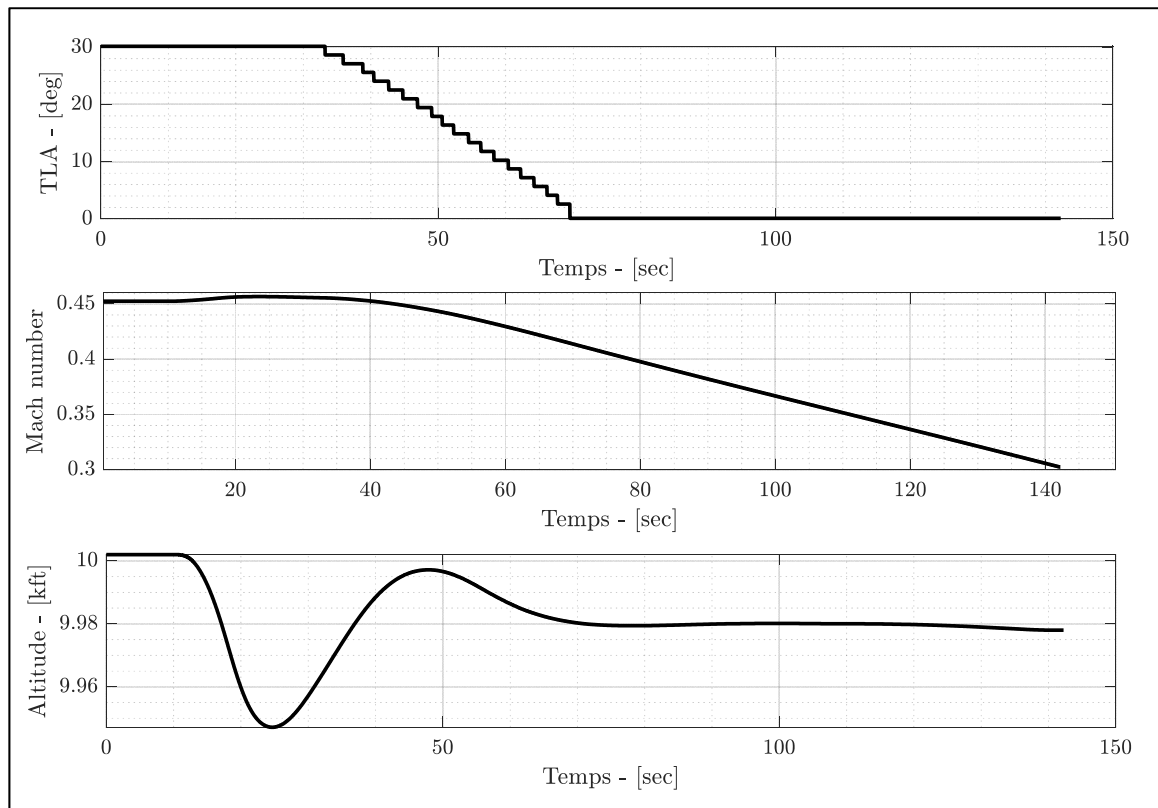


Figure 3.14 Conditions de vol lors de la variation de la manette des gaz de 30° à 0°

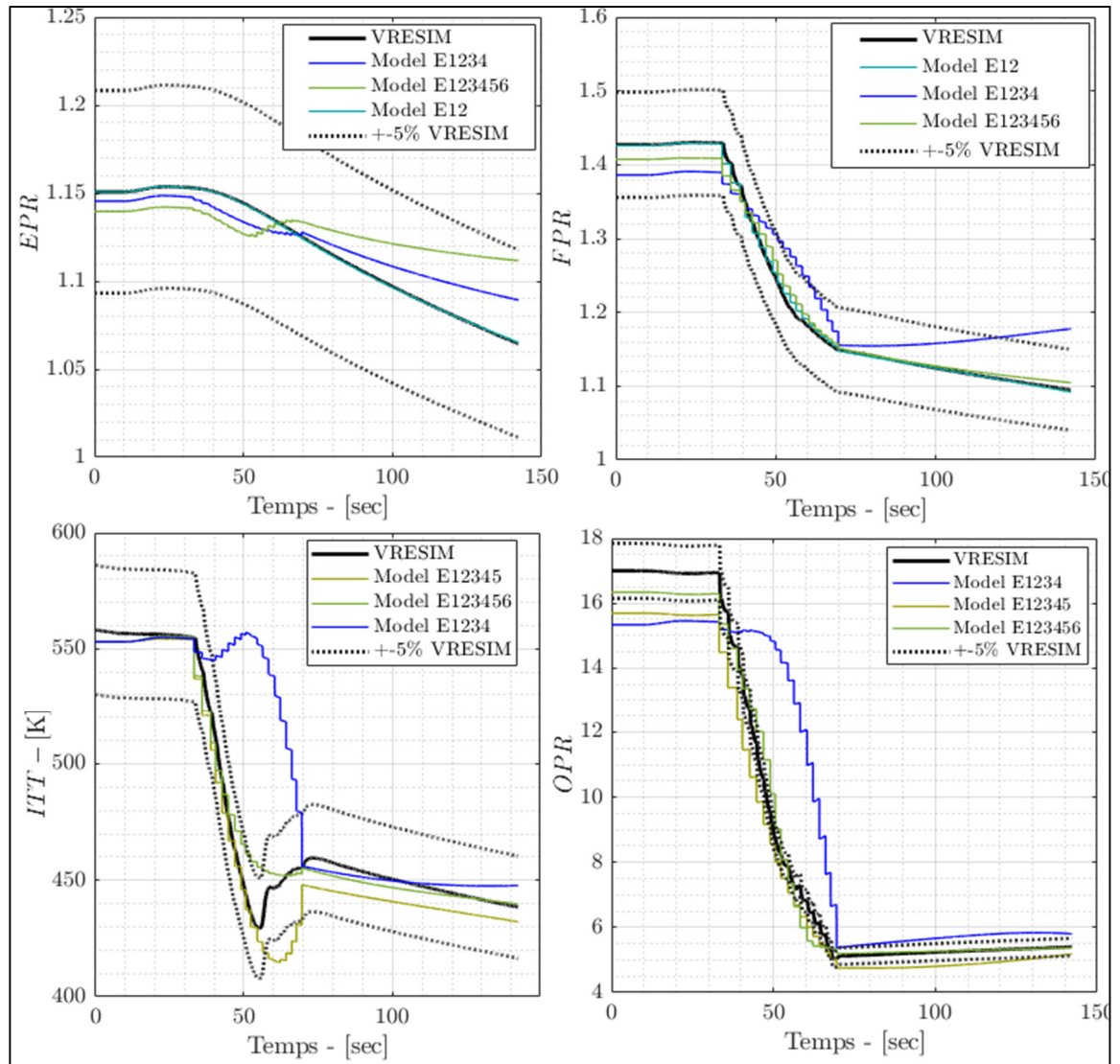


Figure 3.15 Sorties des modèles EPR , FPR , ITT , OPR lors de la variation de TLA de 30 à 0°

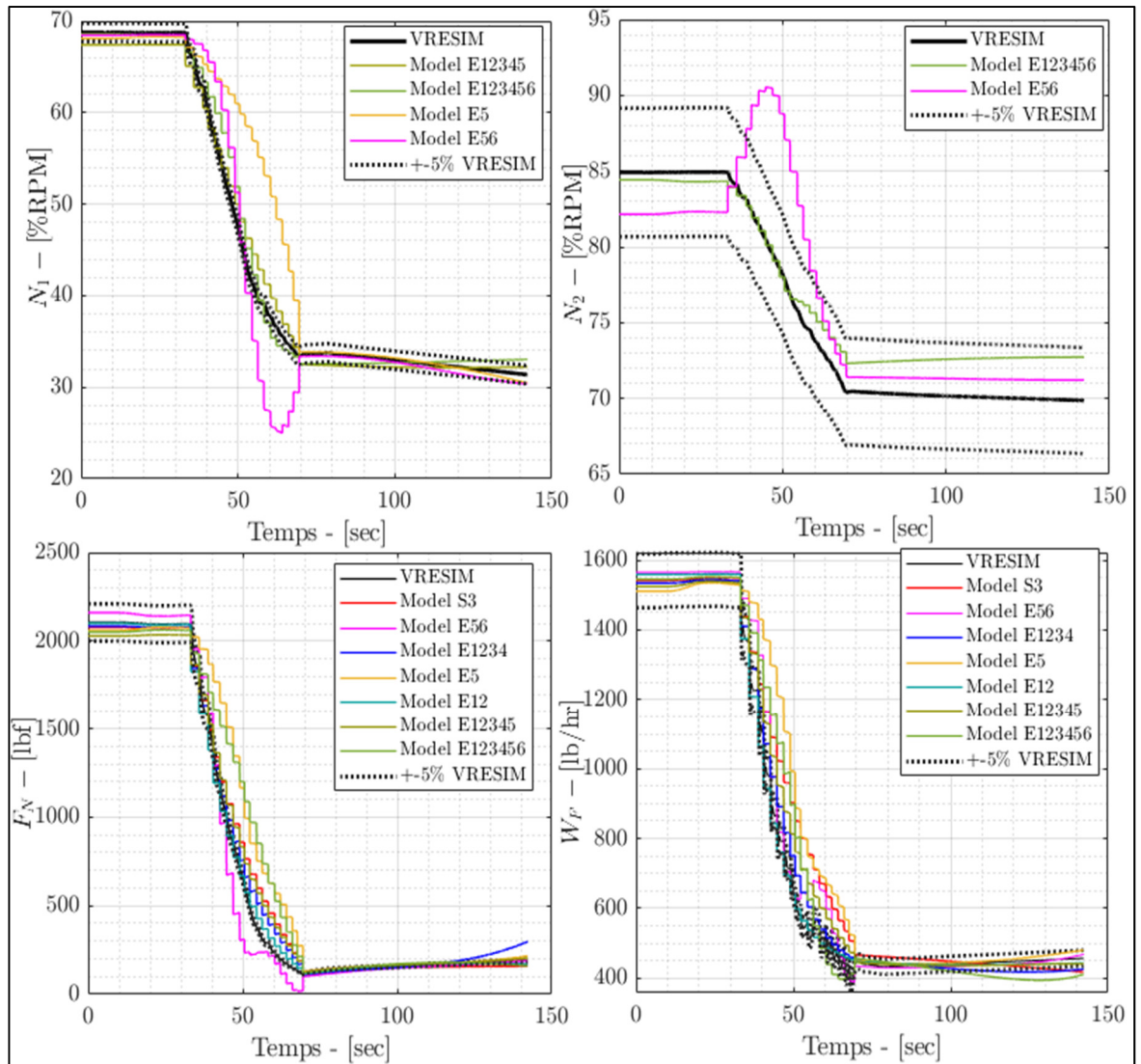


Figure 3.16 Sorties des modèles N_1 , N_2 , F_N et W_F lors de la variation de T_{LA} de 30 à 0°

CHAPITRE 4

DISCUSSIONS, REMARQUES ET OBSERVATIONS

Pour une turbosoufflante, le cycle de Brayton dépend essentiellement de paramètres de design dont : le ratio de pression du moteur EPR , le ratio de pression de la soufflante FPR , le ratio de pression totale OPR , la température à l'entrée de la turbine ITT , la vitesse de rotation de la soufflante N_1 et la vitesse de rotation de la turbine N_2 . Les paramètres de performances sont notamment le débit de carburant W_F et la poussée du moteur F_N , qui sont des performances importantes lors du choix d'un moteur par rapport à un autre.

Un modèle de « boîte noire » a été appliqué, qui consiste à identifier un modèle de performances à partir des réseaux de neurones. Le modèle de « boîte noire » ne s'intéresse pas à la physique du système, mais il cherche à reproduire ses signaux de sorties. Pour ce faire, il tente de trouver des liens sans interprétation physique entre ses entrées et ses sorties. Dans le cas des réseaux de neurones, les liens que l'utilisateur doit établir sont les poids w et les biais b , les fonctions d'activation et le nombre de neurones, qui sont établis lors de la phase d'apprentissage.

4.1 Méthodologie

Une nouvelle méthodologie a été développée afin d'identifier le modèle statique des performances de la turbosoufflante General Electric CF34-8C5B1 en utilisant deux réseaux de neurones en cascade de types perceptron simple couche avec une propagation directe (en anglais *FeedForward Neural Network*) et à l'aide des tests en vol. Cette méthodologie a été réalisée en deux étapes : 1) la première étape a consisté en l'estimation des paramètres du moteur dont les EPR , FPR , OPR , ITT , N_1 et N_2 à partir d'un premier réseau de neurones ECS-NET. Cette étape était nécessaire pour reproduire le système de contrôle du moteur en modélisant les niveaux de poussées, ainsi que les paramètres de chaque composant du moteur, 2) la deuxième étape a consisté à l'estimation des paramètres de performances du moteur, dont

le débit de carburant W_F et la poussée F_N à partir d'un deuxième réseau de neurones ENG-NET. Par conséquent, la méthodologie développée résulte implicitement d'une combinaison de l'approche par composant avec les réseaux de neurones.

4.2 Réalisation des tests en vol

Une base de données a été mise en place pour l'apprentissage des réseaux de neurones. Cette base de données a été créée à partir des tests réalisés sur le simulateur de vol de l'avion CRJ700.

Une méthodologie pour la réalisation des tests en vol a été développée afin de collecter tous les paramètres du moteur. L'élaboration de la base de données a occupé la plus grande partie du temps consacré à ce mémoire, étant donné le temps nécessaire pour se familiariser avec l'outil de recherche. Les efforts alloués ont principalement été consacrés à la planification des tests, à l'apprentissage au langage CTS, qui est le langage de programmation de CAE Inc. et à la réalisation des tests en vol.

Ces tests ont été réalisés sur toute l'enveloppe de vol de l'avion pour quatre phases de vol, dont le décollage, la montée, la croisière et la descente. Les données de pressions et de températures, les données de vol ainsi que celles des conditions atmosphériques ont ensuite été réarrangées sous Matlab pour l'identification du modèle de performances. Les paramètres corrigés (vitesse de rotation de la soufflante corrigée $N_{1,C}$, vitesse de rotation de la turbine corrigée $N_{2,C}$, poussée corrigée $F_{N,C}$ et débit de carburant corrigé $W_{F,C}$) et les paramètres de design (EPR, FPR, ITT, OPR) ont enfin été ajoutés à partir des équations (2.8) à (2.15). La base de données consiste ainsi en un ensemble de « points de vol » espacés dans le temps de 20 à 30 secondes pour chaque test de vol.

4.3 Identification du modèle

La modélisation a été réalisée sur Matlab/Simulink, qui est un logiciel adapté pour l'identification de systèmes. Le logiciel Matlab comprend une boîte d'outils destinée aux réseaux de neurones (*Neural Network Toolbox*). Ainsi, les réseaux de neurones ont été entraînés grâce à l'algorithme de régularisation bayésienne, qui utilise l'algorithme de Levenberg-Marquardt (LM). L'algorithme LM utilise l'algorithme de Gauss – Newton, ainsi que l'algorithme du gradient. L'algorithme de la régularisation bayésienne tente d'optimiser une fonction de coût qui calcule la moyenne des erreurs quadratiques (en anglais *Mean Square Error* ou MSE). Cet algorithme a été choisi suite à son excellente propriété de généralisation, et il nécessite un minimum de données pour l'identification du modèle.

Le premier réseau de neurones ECS-NET a consisté en l'estimation des paramètres EPR, OPR, FPR, ITT, N_1 et N_2 . Les données en entrées étaient les conditions de vol (dont l'altitude et le nombre de Mach) et les conditions atmosphériques, dont les données de pressions et de températures. Le deuxième réseau de neurones ENG-NET a consisté en la modélisation des paramètres de performances, dont le débit de carburant W_F et la poussée nette F_N . Les données d'entrées étaient notamment les sorties du réseau ECS-NET et le nombre de Mach. Le nombre de couches cachées pour chaque réseau a été limitée à un (1). La fonction hyperbolique tangente sigmoïde a été utilisée pour modéliser la couche cachée tandis que la fonction linéaire a été utilisée pour modéliser la couche de sortie. Le nombre de neurones a ensuite été déterminé en minimisant l'erreur moyenne quadratique (MSE). Pour ce faire, le réseau de neurones a été identifié lorsque le MSE était inférieur à 5 % de la valeur du MSE maximale. La méthodologie de recherche a ensuite été répétée pour différentes combinaisons de sorties pour le réseau ECS-NET.

4.4 Validation des modèles de performances

La phase de validation a été réalisée à partir d'une analyse comparative entre les sorties des modèles et celles de l'avion de référence. Pour ce faire, une analyse statistique a été effectuée et un critère de tolérance de 5 % défini par la FAA a été considéré pour les erreurs sur chaque

sortie. Les erreurs résiduelles ont montré que les modèles étaient parfaitement capables de prédire tous les paramètres du moteur. En effet, les erreurs étaient négligeables, surtout pour la poussée et le débit de carburant, dont les erreurs étaient respectivement inférieures à 200 lbf et 100 lb/hr, ce qui est relativement petit par rapport à la poussée maximale du moteur (13 790 lbf) et par rapport au débit de carburant maximal mesuré (10 000 lb/hr).

Différentes simulations ont été réalisées pour valider les résultats pour chaque régime et scénario de vol. Ces simulations comprennent une phase de décollage, une phase de montée, une phase de descente, une phase de croisière avec une panne du moteur, une phase de descente et deux cas caractérisant un changement du régime de poussée. Les sorties des modèles ont été comparées avec celles du simulateur. Les résultats de simulations ont été excellents, étant donné que pour les modèles et pour toutes les phases de vol, l'erreur de prédiction a été inférieure aux limites de tolérance de 5 %.

L'analyse statistique a permis de généraliser la précision de chaque modèle pour toute l'enveloppe de vol de l'avion et pour toutes les phases de vol. Pour chaque modèle de performance sélectionné, l'erreur a été proche de 0, la moyenne des erreurs μ a été inférieure à 1 %, et la déviation standard a été inférieure à 5 %, voire à 3 % pour certains paramètres. L'identification du modèle a été réalisée avec succès.

4.5 Observations et remarques

Les entrées du modèle ont été l'altitude, le nombre de Mach et la déviation de température par rapport à la température standard ISA, comme il est illustré sur la Figure 2.25. Les données de l'atmosphère ont été obtenues à partir d'un modèle standard de l'atmosphère, ce qui a facilité la modélisation.

Comme on peut le constater, si les ratios de pression EPR, FPR et OPR sont choisis pour la modélisation du réseau ECS-NET, alors au moins 2 de ces trois paramètres sont requis. Utiliser les paramètres séparément lors du calcul des performances F_N et W_F n'est pas recommandé,

puisque la prédiction des paramètres serait biaisée, comme dans le cas des modèles E1, E2 et E4 (voir section 3.2.2).

La prédiction des performances lors de la descente est meilleure lorsque les paramètres N_2 , ITT et OPR sont inclus dans le calcul de ECS-NET, tandis que pour les autres phases, la prédiction des performances est plus précise lorsque les paramètres N_1 et FPR sont considérés. Ceci est dû au fait que la soufflante produit 80 % de la poussée, tandis que le corps principal produit 20 %. Les régimes de poussées lors des phases de montée, de décollage et de croisière dépendent donc directement des paramètres de la soufflante, puisque lors de ces phases de vol, la poussée est élevée. Cependant, lors de la descente, la poussée est généralement petite et elle est très sensible aux paramètres du corps principal (compresseur à haute pression, chambre de combustion et turbine). Ainsi, la prédiction des paramètres du corps principal est plus importante lors de la descente.

La simulation du modèle complet lors de la panne du moteur est un exemple concret d'application. Bien qu'il existe une erreur statique de modélisation, le modèle est tout à fait capable d'évaluer le niveau de dégradation du moteur en calculant les différences entre les sorties estimées du modèle et celles du simulateur. Un axe de recherche pourrait être proposé, qui consisterait à identifier un modèle de performance capable de localiser la panne dans le moteur et de prendre des décisions intelligentes pour sa restauration.

Le modèle de performance identifié dans ce mémoire est statique, étant donné que les paramètres sont invariants par rapport au temps. Une extension de ces recherches consisterait à identifier un modèle dynamique capable de prédire les performances du moteur en régime transitoire, comme lors des accélérations et décélérations. Dans ce cas, les paramètres du moteur dépendraient du temps.

Les tests sur le simulateur de vol ont pu représenter tous les états du moteur pour toute l'enveloppe de vol de l'avion et tous les régimes de poussée. De ce fait, les barrières de protections du moteur ont implicitement été prises en compte dans les tests et n'ont pas requis

de modélisation additionnelle. De la même manière, les ratios de températures du compresseur et de la soufflante n'ont pas été nécessaires à la modélisation puisqu'ils sont indirectement calculés à partir des ratios de pressions et des équations d'hydrostatiques présentées dans la section 1.7.4.

Les réseaux de neurones ont permis d'identifier un modèle de performances capable de compenser les non-linéarités du système et le manque de paramètres comme les rendements. En effet, les rendements de chaque composant du moteur ont été implicitement pris en compte lors de l'apprentissage du réseau de neurones. Par conséquent, l'absence des valeurs de rendements ne constitue pas un obstacle pour l'identification. La modélisation à partir des réseaux de neurones est plus facile à réaliser puisqu'il évite l'utilisation d'équations complexes comme celles des équations de thermodynamique qui doivent prendre en compte les conditions de chocs dans les calculs (voir section 1.7.4).

Enfin, l'identification du modèle de performances a été réalisée avec succès, car d'excellents résultats ont été obtenus. Les résultats ont montré que les modèles identifiés ont été capables de prédire les performances du moteur avec erreur relative inférieure à 5 % pour tous les régimes de vol. La seule exception consistait à la prédiction de la poussée lors de la phase de descente pour laquelle la déviation standard a été d'environ 8.74 %. Ceci a été dû au fait que lors du régime de descente, lorsque la manette des gaz a été placée sur la position IDLE, la poussée est quasiment nulle, voire négative. Dans ce cas, la prédiction a été plus difficile à réaliser. Cependant, comme dit précédemment, l'erreur résiduelle est négligeable pour la poussée lors de la descente et les modèles sont est capable de prédire la tendance générale de la poussée. Chaque modèle de performance a donc été identifié et validé pour chaque régime de vol.

CONCLUSION

Une nouvelle méthodologie a été développée et validée afin d'identifier un modèle de performances pour la turbosoufflante General Electric CF34-8C5B1 à partir des réseaux de neurones et des tests en vol. Les tests en vol ont été réalisés sur le simulateur de vol certifié niveau D du CRJ700. Il est à savoir que le niveau D est la plus haute qualification décernée par la FAA pour la dynamique du vol et la modélisation des moteurs. Ainsi, les tests en vol ont couvert toute l'enveloppe de vol de l'avion pour quatre phases de vol, dont le décollage, la montée, la croisière et la descente.

L'identification du modèle a été réalisée avec succès en obtenant d'excellentes performances. Les résultats ont montré que les modèles ont une excellente capacité de prédiction, avec une erreur relative inférieure à la tolérance de 5 % définie par la FAA.

Ce projet de recherche a permis entre autres d'élargir l'application des réseaux de neurones. Les différentes combinaisons de sorties du modèle ECS-NET ont démontré l'efficacité et l'adaptabilité des réseaux de neurones à prédire les performances du moteur. En effet, chaque modèle de performances a été identifié puis validé pour chaque régime de poussée.

Les réseaux de neurones ont permis de prédire les performances du moteur à partir d'un minimum de données. Les modèles identifiés ont pu prédire les performances du moteur lors d'un changement de régime de poussée. Une application pour l'évaluation de la panne du moteur a également été proposée. En effet, la détection, la localisation, l'évaluation, et la réparation des défaillances du moteur pourraient être une extension de ces recherches. Enfin, étant donné que le modèle identifié est statique, il serait intéressant de concevoir un modèle dynamique, capable de modéliser les performances du moteur en régime transitoire.

ANNEXE I

AIAA Propulsion & Energy Forum 2021

Aircraft Engine Performance Model Identification using Artificial Neural Networks

Rojo Princy Andrianantara², Georges Ghazi³ and Ruxandra Mihaela Botez⁴

University of Québec, École de Technologie Supérieure,

*Laboratory of Applied Research in Active Controls, Avionics and AeroServoElasticity (LARCASE),
Montreal Quebec, H3C 1K3*

Abstract

This paper presents a methodology developed at the Laboratory of Applied Research in Active Controls, Avionics and AeroServoElasticity (LARCASE) to identify a performance model of the engine powering the CRJ-700 regional jet aircraft from flight data using neural networks. To this end, a qualified virtual research simulator (VRESIM) was used to conduct several categories of flight tests and collect engine data under a wide range of operating conditions. The collected data were then used to create a comprehensive database for the training process. This process was performed using the Bayesian regularization algorithm available in the Matlab Neural Networks Toolbox, and a study was carried out to estimate the optimal number of neurons in the network structure. Validation of the methodology was accomplished by comparing the prediction model with a series of flight data collected with the flight simulator for different flight conditions and different flight phases including takeoff, climb, cruise and descent. The results showed that the model was able to predict the engine performance in terms of fan speed, thrust and fuel flow with good accuracy.

I. Introduction

MATHEMATICAL modeling consists of establishing a set of relationships in order to predict the behavior of a system. Today, with the increase in computing power and the advances in numerical methods, manufacturers are more confident than in the past in their models to simulate engine behaviors during off-design operations and thus, in reducing the costs allocated to ground tests. Numerical models have become practical, especially for monitoring engines in order to evaluate their performance, diagnosing potential faults, assessing component efficiency, optimizing maintenance process, or estimating the engine's life cycle.

²Master Student, LARCASE, ETS, 1100 Notre Dame West, Montreal, QC, H3C-1K3, Canada.

³Assistant Professor, LARCASE, ETS, 1100 Notre Dame West, Montreal, QC, H3C-1K3, Canada.

⁴Full Professor, AIAA Associate Fellow, Canada Research Chair Holder Level 1 in Aircraft Modeling and Simulation, Head of the LARCASE, ETS, 1100 Notre Dame West, Montreal, QC, H3C-1K3, Canada.

Several modeling techniques have been explored by researchers in the literature. Among these methods, the most used one seems to be the Component Level Modeling (CLM). This method is usually considered for the design of engine simulation platforms as explained by Mattingly [1], Chapman [2, 3] and Saravanamuttoo [4, 5]. The CLM method consists of a physics-based approach, in which the conservation laws of the Brayton cycle are applied, and where each component of the engine is described by thermodynamic equations [6]-[8]. Consequently, the accuracy of the overall engine model directly depends on the accuracy of the models of each component.

Unfortunately, the characteristics of engine components are sensitive information and are often considered strictly confidential by their manufacturers. Thus, designers must rely on generic engine data or on fitting techniques [9], which complicates the identification process, and sometimes reduces the prediction accuracy. Furthermore, as the engine is functioning at various operating conditions, the CLM technique requires a large number of calculations and interpolations of component maps, that are expensive in terms of computational power and time.

Another alternative is to use Artificial Neural Networks (ANN). Indeed, systems based on machine learning, such as ANNs, can reproduce the behavior of complex nonlinear systems with fewer iterations and therefore, are faster than analytical methods [10]. Although neural network systems do not necessarily consider the internal physics and interactions between components, their efficiency leads researchers to choose this type of model rather than analytical models.

The use of ANNs for gas turbine modeling began in the early 2000s, notably with the work carried by Stamatis [11], Kanelopoulos [12], Mathioudakis [13] and Kobayashi [14], in which ANNs were applied for engine health monitoring and performance estimation. Similarly, Lazaretto [10] designed an hybrid engine model by combining the CLM method with an ANN to compensate the lack of knowledge of component maps. Most recently, Zhou [15] and Zhao [16] use sophisticated ANN structures for fault diagnostics and decision-making process.

The main objective of this paper is to investigate the possibility of using neural networks to model the steady-state performance of an aircraft engine. For this purpose, the methodology was developed and tested on a CRJ-700 regional jet aircraft, powered by two high-bypass turbofan engines. All data required to train the neural networks defining the engine model structure, and then to validate the model were obtained on a Virtual Research Simulator (VRESIM) designed and manufactured by CAE Inc. The VRESIM is a research aircraft flight simulator that meets all Level D qualification criteria for flight dynamics and engine propulsion model. Level D corresponds to the highest level of qualification according to the criteria of the Federal Aviation Administration (FAA), which implies that the VRESIM can be considered as a very precise model, and by extrapolation, as a “test aircraft”.

The paper is organized as follows: **Section II** briefly presents the engine to be modeled, as well as the main mathematical relationships that will be used to design the model. **Section III** deals with the methodology of the neural network. **Section IV** presents various simulation results and the validation of the proposed methodology. Finally, the paper ends with conclusions and remarks.

II. Aircraft Description and Engine Mathematical Model

A. Aircraft and Engine Description

The aircraft considered in this study to develop and to validate the proposed method is the well-known Bombardier CRJ-700, originally designed and manufactured by Bombardier Aerospace⁵. The CRJ-700 is a “T-Tail” regional jet designed to operate at a ceiling altitude of 41,000 ft, and at a maximum operating speed of Mach 0.82. The latest generation of CRJ-700 is equipped with two General Electric CF34-8C5B1 high-bypass turbofan engines, mounted on each side of the fuselage and

⁵ The group was acquired by Mitsubishi Aviation Group in 2020.

at the rear of the aircraft. Each engine can produce a maximum takeoff thrust of 13,790 lbf (61.3 kN) at the sea level and under standard temperature conditions (i.e., 15°C).

A schematic representation detailing the main components of the engine is given in Fig. 1.

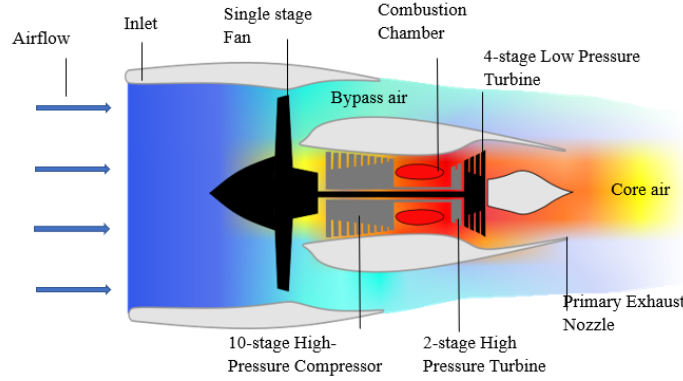


Fig. 1 Schematic Representation of the CF34-8C5B1

As shown in Fig. 1, the CF34-8C5B1 is composed of a two-spools axial flow with separated flow, and an internal combustion chamber. The engine core consists of a two-stage high-pressure turbine that drives the high-pressure shaft and the ten-stages high-pressure compressor (represented by gray color), and a four-stage low-pressure turbine that drives the low-pressure shaft and the single-stage fan (represented by black color).

B. Engine Functional Relationships

1. Thrust and Fuel Flow Functional Relationships

In general, the thrust and fuel flow of an engine are two complex parameters that depend mainly on the engine rotation speed (i.e., fan speed) N_1 , various flight parameters such as the aircraft airspeed V_T , the ambient temperature T , the ambient pressure P , and the specific air constant R_{air} , and different physical characteristics of the engine $\lambda = \{\lambda_1, \dots, \lambda_n\}$. Mathematically, the two parameters can therefore be described by functional relationships as follows:

$$F_N = f(N_1, V_T, T, P, R_{\text{air}}, \lambda) \quad (1)$$

$$W_F = f(N_1, V_T, T, P, R_{\text{air}}, \lambda) \quad (2)$$

where F_N is the thrust, and W_F is the fuel flow. Note that the expression “ $f(x, y, \dots, z)$ ” in the above equations is used to simplify the general notation “a function of $(x, y, \dots, z)$ ”.

The functional relationships in Eqs. (1) and (2) are governed by mathematical functions that depend on a minimum of 6 parameters (or more depending on the size of λ). Clearly, identifying such functions can be very complex, and require a large amount of data to be able to determine the effect of each parameter on the engine performance. However, using dimensional analysis methods such as the one presented by Ghazi and Botez [18], it is possible to reduce the number of parameters, and consequently to express the engine performance in a more compact form, as follows:

$$F_N/\delta = f(N_1/\sqrt{\theta}, M) \quad (3)$$

$$W_F/\delta\sqrt{\theta} = f(N_1/\sqrt{\theta}, M) \quad (4)$$

where $F_{N,C} \equiv F_N/\delta$ is the corrected thrust, $W_{F,C} \equiv W_F/(\delta\sqrt{\theta})$ is the corrected fuel flow, δ is the static air pressure ratio, and θ is the static air temperature ratio.

A significant advantage of representing the engine performance in the above form is that it results in a simplified functional relationship that depends on only two variables: the corrected fan speed $N_{1,C} \equiv N_1/\sqrt{\theta}$ and the Mach number M . This relationship reduces considerably the complexity of functions to be modeled and, indirectly, the amount of data required for the identification process.

2. Engine Thrust Ratings Functional Relationships

The parameter $N_1/\sqrt{\theta}$ in Eqs. (3) and (4) is the corrected fan speed and reflects the engine power. This parameter is controlled by the pilot using the thrust levers (i.e., throttles) to vary the engine power depending on the flight phase. When the pilot advances [*reduces*] the throttles, a signal is sent to the Engine Control System (ECS) to increase [*decrease*] the fan speed, resulting in greater [*lower*] thrust. However, to prevent engine components from operating beyond their design limits, the ECS includes a set of thrust limits. These limits, also called “thrust ratings”, define the maximum fan speed that can be controlled under given operating conditions.

For most commercial aircraft, there are three typical thrust ratings that pilots are expected to use depending on the flight phase. These ratings are: (1) Maximum Takeoff (MTO), (2) Maximum Climb (MCLB), and (3) Idle (IDLE).

Each of these thrust settings can be selected by placing the thrust levers in a specific position, and should be used during takeoff, climb and descent phases, respectively. The cruise, on the other hand, is the only flight phase in which the throttles should be constantly adjusted somewhere between the idle position and the maximum climb position according to the flight conditions. In this case, the corresponding N_1 is calculated by the ECS by assuming a variation between the IDLE and MCLB values.

As a result, the corrected engine fan speed can be described by the following functional relationship:

$$N_{1,C} = f(h, M, \Delta ISA, TLA) \quad (5)$$

where h is the altitude (i.e., pressure altitude), M is the Mach number, ΔISA is the standard temperature deviation, and TLA is the thrust levers angle (or throttles position).

III. Methodology and Model Identification

A. Engine Performance Database Generation

In order to identify an engine performance model, it was first necessary to develop a detailed and comprehensive database. In this study, the database was created using the VRESIM available at the LARCASE. As mentioned in **Section I**, the VRESIM is a research aircraft flight simulator designed by CAE Inc. that has a level-D qualification for its flight dynamics and engine propulsion model. It was therefore considered that the data collected from the VRESIM was accurate to represent the actual performance of the engines powering the CRJ-700 regional jet aircraft.

In order to obtain enough data that adequately reflects the engine performance over the entire aircraft operating envelope, four categories of flight tests were considered. The database generation for each case is detailed in next sections III.A.1 to III.A.4, and the total engine performance database is described in section III.A.5.

1. Normal Takeoff from an Airport

The first category of tests consisted of takeoffs under normal operating conditions. To this end, a total of 6 takeoff tests were simulated with the VRESIM. For the sake of simplicity, a medium takeoff weight was used for all these tests. However, in order to determine the impact of pressure-altitude on the engine performance, these tests were conducted at various airports with altitudes ranging from 28 to 8000 ft above sea level.

For each test, the engine parameters and flight data such as altitude, airspeed and outside air temperature were recorded from the VRESIM with a sampling rate of 1/4 Hz (i.e., 1 sample every 4 s). It should be noted that the data recording of the data was always initiated 5 seconds after the time when the “test pilots” placed the throttles in the MTO position. This delay was imposed in order to avoid the transient dynamics of the engines at the beginning of the acceleration phase. Indeed, since the study presented in this paper only focused on the identification of a model to predict the static performance of the engine, its transient dynamics was not considered.

Finally, the recording of the data was stopped once the aircraft reached an altitude of 1,500 ft above the airport reference altitude.

2. *Continuous Climb at Maximum Climb Thrust*

The second category of tests consisted of continuous climb tests at constant Calibrated Airspeed (CAS) and constant Mach number, and with both engines set at maximum climb thrust. For this purpose, a total of 10 climb scenarios were simulated with the VRESIM. In order to obtain sufficient data to cover the entire aircraft flight envelope, different climb speed profiles (i.e., a CAS / Mach combinations) were considered for each test. These speed profiles were chosen by varying the CAS from 180 to 320 kt, and the Mach number from 0.60 to 0.80. It is worth mentioning that since aircraft are not allowed to fly above 250 kt below 10,000 ft, an acceleration was performed at 10,000 ft for all scenarios with a climb speed greater than 250 kt.

Finally, for all climb tests, the engine parameters and flight data were collected from 2000 to 40,000 ft, and at a sampling rate of 1 Hz. The data was then resampled every 1000 ft in order to limit the database size.

3. *Continuous Descent at Idle Thrust*

In a complementary way, 10 descent scenarios were also considered in order to obtain engine data for the idle regime. The descent scenarios were determined in the same way as for the climb scenarios, i.e. by selecting different CAS/Mach speed profiles in order to cover the aircraft flight envelope in descent. In addition, when necessary, a level-off deceleration was performed before reaching 10,000 ft to reduce the aircraft’s CAS below the 250 kt limit.

Finally, as for the climb phase, the engine parameters and flight data were collected from 40,000 to 2000 ft, at a sampling rate of 1 Hz, and then resampled every 1000 ft to limit the size of the database.

4. *Static Cruise Performance*

The last category of tests considered in this study consisted of static cruise performance tests. These tests were mainly used to understand and model the logic of variation of engine parameters (in particular the fan speed) according to the throttle positions. For this purpose, a total of 25 cruise scenarios were determined by selecting different flight conditions in a sub-region of the aircraft flight envelope delimited by altitudes between 5000 and 40,000 ft, and Mach numbers between 0.45 and 0.83. For each flight condition, the aircraft was trimmed by using the autopilot to stabilize the altitude, and by manually adjusting the throttles to maintain the Mach number.

Once the aircraft was trimmed, the engine parameters and flight data were collected during 60 seconds at a sampling rate of 1/10 Hz (i.e., 1 sample every 10 seconds). The average of the sampled data was then used to determine the actual engines performance for the selected flight conditions.

5. Engine Performance Database

All data collected during the takeoff, climb, cruise and descent tests were next exported from the VRESIM in the form of *.CSV files, and then restructured into Matlab® in order to build an initial engine performance database. The database was then completed by adding the engine corrected parameters (i.e., thrust, fuel flow and fan speed) and all the other parameters required for the identification process. Finally, the database was randomly divided into three sub-databases: training (50%), cross-validation (25%) and test (25%), which were prepared for the supervised learning process.

The results obtained are shown in Fig. 17 and Fig. 18. Note that for confidentiality reasons, the values of the engine parameters in each graph have been normalized between -1 and 1.

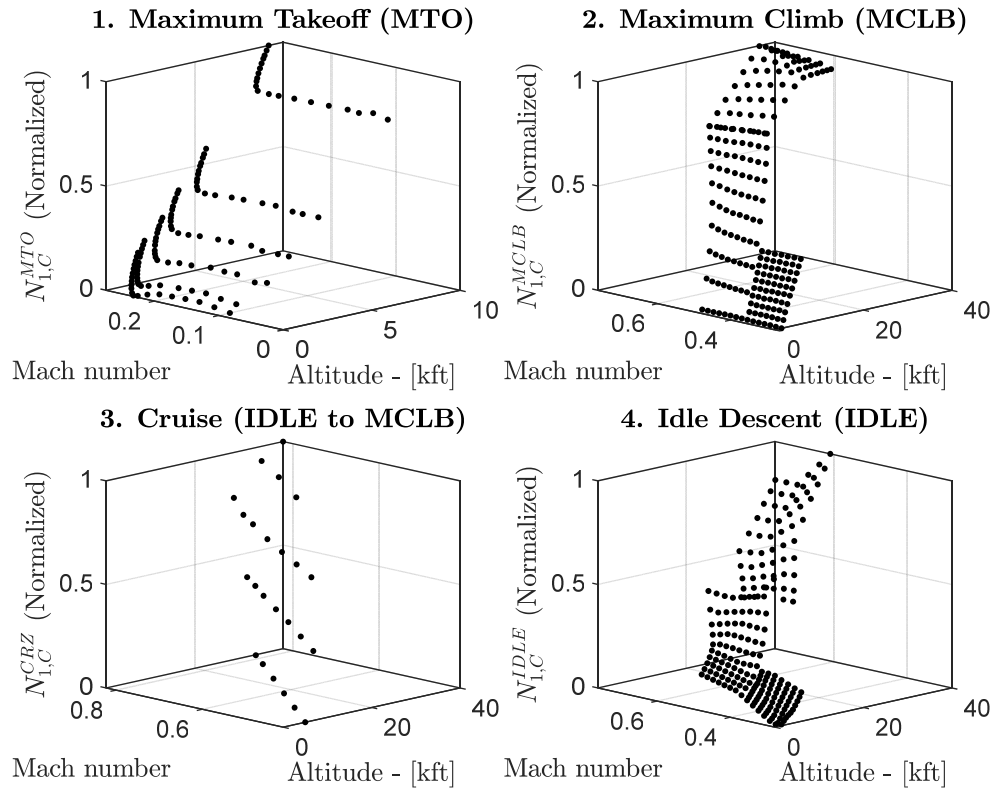


Fig. 17 Engine Corrected Fan Speed Database for all Flight Phases (function of Altitude and Mach number)

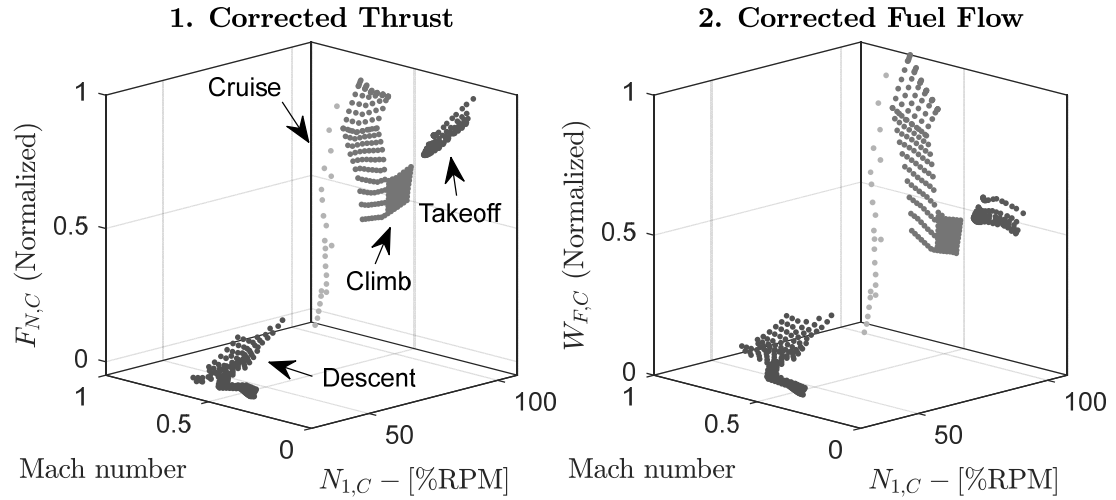


Fig. 18 Engine Corrected Thrust and Corrected Fuel Flow Database (function of Corrected Fan Speed and Mach number)

B. Engine Neural Network Structure

Once the identification database was created, the next step in the methodology was to define a neural network topology to mimic the engine structure. In general, there are different types of neural networks that can be considered depending on the characteristics of the system to be modeled. However, since the study presented in this paper is not intended to model a time-dependent system (only the static performance of the engine is of interest), it has been decided to use a feedforward type structure. This network structure is well suited to supervised learning problems, where the primary objective is to find relationships within a data set.

1. Feedforward Neural Network Structure

Basically, a feedforward neural network is a mathematical structure that aims to mimic the behavior of the human brain in order to understand and learn relationships between parameters within a data set. It is mainly composed of artificial neurons interconnected with each other, and whose role is to process and transport information from a set of inputs to one or multiple outputs.

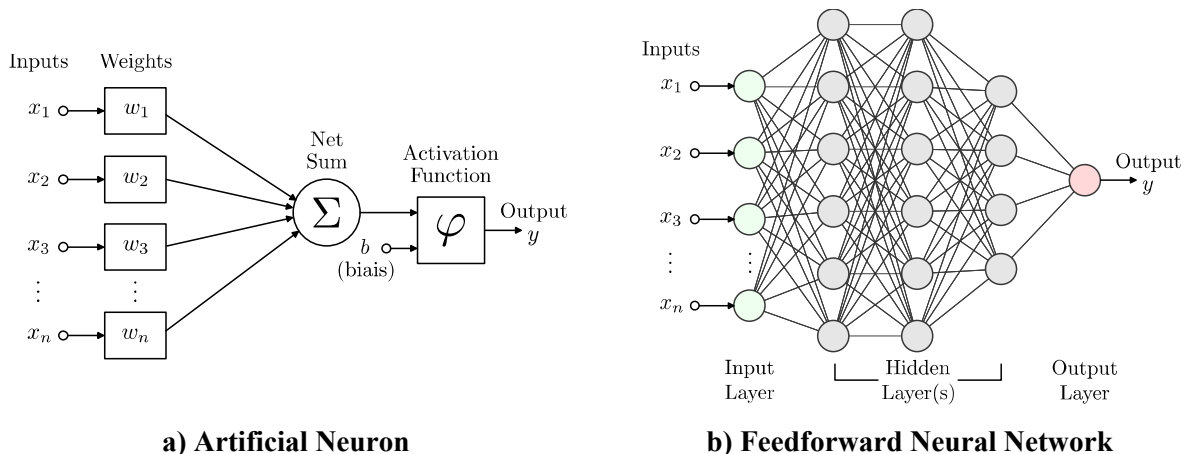


Fig. 19 Representation of an Artificial Neuron and of a Feedforward Neural Network

As illustrated in Fig. 19a, and by analogy to a biological neuron, an artificial neuron can be seen as a processing element that receives a flow of information either from other artificial neurons that precede it, or directly from inputs of the model. The information is first multiplied by weighs, where each weight is a constant coefficient that somehow quantifies the relevance of the input to which it is associated. Thus, the larger the weight, the greater is the impact of the corresponding input on the output of the artificial neuron (or of the model).

The weighted inputs are then summed and sent to a mathematical function, called activation function. This function reproduces the behavior of the neuron and can vary depending on the problem considered. For regression problems, typical choices include for instance the linear function, the sigmoid function or the tangent hyperbolic function. Mathematically, the output of an artificial neuron can be therefore written using the next equation:

$$y = \varphi(\mathbf{x}, \mathbf{w}, b) = \varphi(\mathbf{x} \cdot \mathbf{w}' + b) \quad (6)$$

where y is the output of the artificial neuron, $\mathbf{x} = [x_1, \dots, x_n]$ is a vector containing the inputs, $\mathbf{w} = [w_1, \dots, w_n]$ is a vector containing the weights associated to each input, and b is the bias. The latter is a constant coefficient which can be used to adjust the local shape of the activation function φ .

Starting from this representation, a feedforward neural network can be defined using a mathematical structure composed of a set of artificial neurons organized and grouped by layers, as shown in Fig. 19b.

The number of layers, as well as the number of neurons per layer are user-defined parameters that determine the efficiency of the network. Similarly, the choice of the activation function of neurons for each hidden layers is another parameter that affects the accuracy of the network. It is therefore important to correctly choose these three parameters in order to avoid extreme cases of underfitting and overfitting. These extreme cases occur when the network structure is too simple (underfitting) or conversely when is too complex (overfitting) compared to the system to be modeled.

2. Engine Neural Network Structure

Based on the technical descriptions in **Section II.B**, and on the theory in **Section III.B.1**, it was decided to design the engine performance model using two feedforward neural networks in cascade, as shown in Fig. 20.

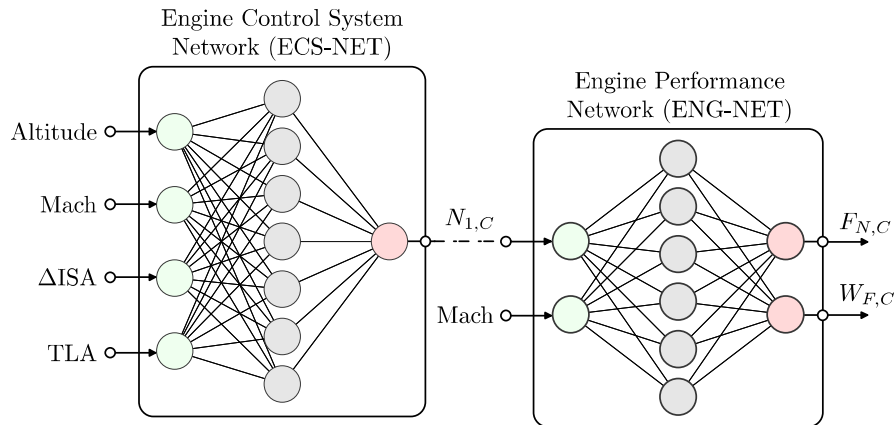


Fig. 20 Engine Static Performance Model Structure using two Feedforward Artificial Neural Networks in Cascade

The first network, referred to as “Engine Control System Neural Network” (ECS-NET), was used to determine the engine corrected fan speed ($N_1/\sqrt{\theta}$) as function of the flight conditions (i.e., altitude, temperature, and Mach number), and the thrust lever positions. The second network, referred to as “Engine Neural Network” (ENG-NET), was used to compute the engine corrected thrust (F_N/δ) and corrected fuel flow ($W_F/\delta\sqrt{\theta}$) from the knowledge of the Mach number and corrected fan speed ($N_1/\sqrt{\theta}$). Once all corrected parameters have been calculated, they can then be “de-corrected” to obtain the engine performance values corresponding to the operating conditions and flight phase.

The main advantage of designing the model under this form is that it allows the main elements of the engine to be represented explicitly: the engine control system on one side and the engine performance on the other side. By this way, the structure of the overall model reflects the actual structure of the engine.

C. Neural Network Training Process

The training process consists of finding the weights and biases of the neural network that offer very good performance. This process is typically achieved by using an optimization algorithm that iteratively searches in a space of possible values for a set of weights and biases that minimize a given cost function. For regression problems, a very practical cost function candidate is the Mean Squared Error (MSE). This function quantifies the accuracy of the neural network by comparing the estimated values with the reference values as follows:

$$MSE(\mathbf{y}, \hat{\mathbf{y}}) = \frac{1}{2N} \sum_{i=1}^N (y_i - \hat{y}_i)^2 \quad (7)$$

where N is the number of data, $\mathbf{y} = [y_1, \dots, y_N]$ is a vector containing the reference values, and $\hat{\mathbf{y}} = [\hat{y}_1, \dots, \hat{y}_N]$ is a vector containing the outputs calculated by the neural network.

The optimization algorithm considered in this paper to train both neural networks (ECS-NET and ENG-NET) was the Bayesian Regularization algorithm available from the Matlab Neural Networks Toolbox. In addition, in order to simplify the training process, it was decided to limit the two neural network structures to a single hidden layer, and to vary only the number of neurons in this layer. This strategy was considered, based on a fundamental property of neural networks which states that any continuous function can be approximated by a single-layer neural network, as long as there are enough neurons. Finally, by testing and comparing the efficiency of different activation functions, it was decided to use a hyperbolic tangent function for the neurons of the hidden layer because it offered better results in terms of generalization than linear or any other function. A linear activation function was applied for the neurons of the output layer.

Figure 6 shows the variation of the MSE as function of the number of neurons for each network. Note that, the values of the MSE in this figure were normalized between 0 and 1 for a better analysis of the results.

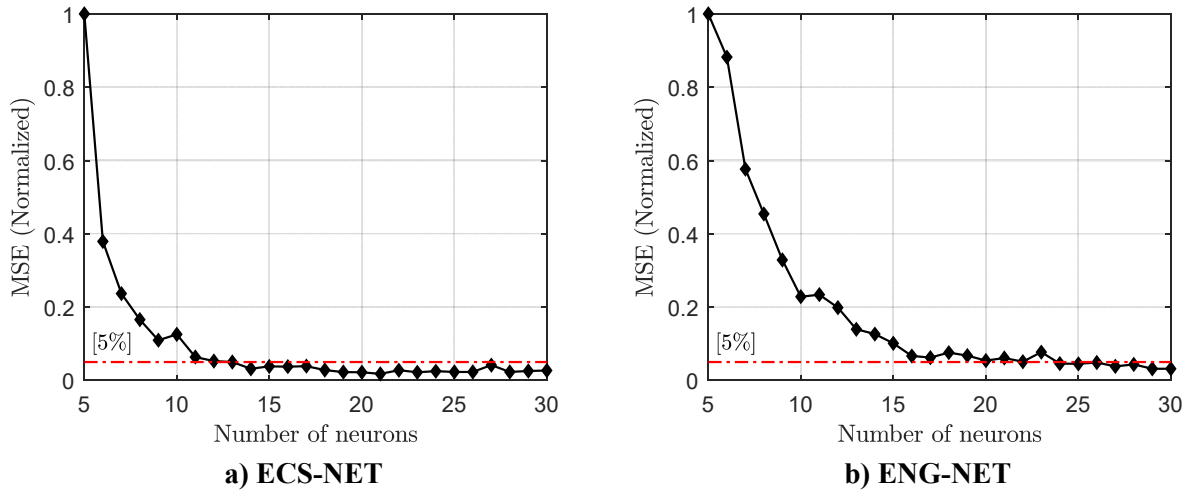


Fig. 21 Variation of the MSE for each Network (ECS-NET and ENG-NET) as function of the number of Neurons on the Hidden Layer

The results in Fig. 21a clearly show that the MSE decreases as the number of neurons increases. However, it can be seen that for more than 11 neurons, the MSE decreases below 5% of its maximum value, and that above this threshold, an increase in the number of neurons no longer allows the MSE to be significantly reduced. Moreover, choosing a number of neurons much higher than 12 could lead to overtraining. In this case, the network would learn “by heart” without necessarily being able to understand the trend of the data, and therefore to generalize it for other operating conditions. Thus, based on these observations, it was concluded that a good compromise would be a number of neurons equal to 15, which avoids overtraining while ensuring a very good MSE value.

Applying the same reasoning to the data shown in Fig. 21b, it was found that 20 neurons was a good choice for the second engine networks, i.e., ENG-NET.

The results obtained for the two trained neural networks are summarized in Table 1, while Fig. 22 shows the distribution of residual errors calculated over the entire data set of the identification database.

Table 1 Neural Networks Training Results

	ECS-NET	ENG-NET	
Number of Neurons	15	20	
Number of Epochs	924	1697	
MSE (Normalized)	0.031	0.035	
Engine Parameter(s)	N_1 [%RPM]	F_N [lbf]	W_F [lb/hr]
Mean Residual Errors	0.004 ± 0.13	0.35 ± 13.95	0.05 ± 8.82

From a general point of view, we can conclude that the results obtained are very good. Indeed, the results show that the ECS-NET can model the engine fan speed with a residual error of less than 0.75% RPM, and within a confidence interval of the order of 0.004 ± 0.13 %RPM. Regarding the engine performance, the results are also very good. The ENG-NET was able to model the engine thrust and fuel flow with residual errors of less than 60 lbf and 45 lb/hr, respectively. In comparison to the average magnitude for the two parameters during a typical flight, these differences are clearly negligible. It is

also interesting to note that the errors distributions for both parameters are bell-shaped around 0 and have relatively small standard deviations.

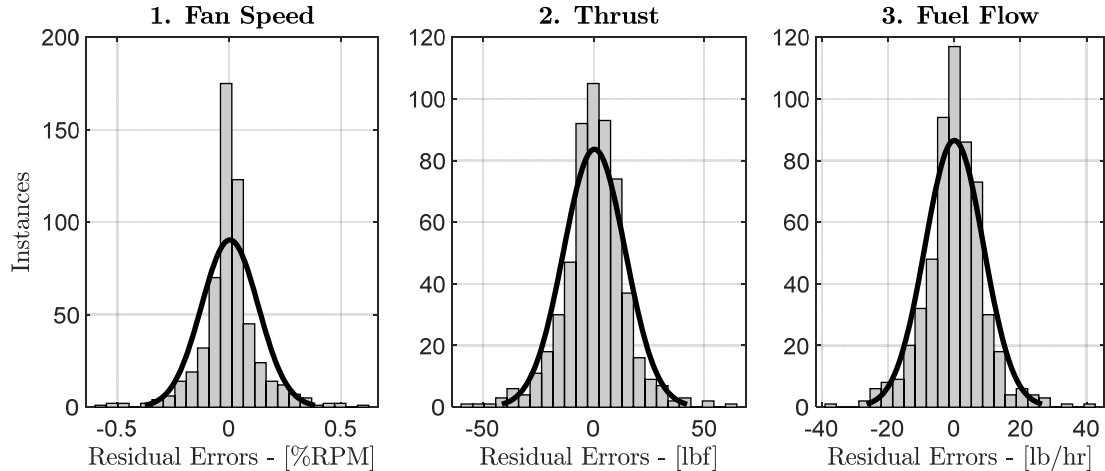


Fig. 22 Residual Errors for the Engine Fan Speed, Thrust and Fuel Flow obtained for the Training Process

Both neural networks ECS-NET and ENG-NET satisfy the required criteria in terms of performance and can be studied further for validation.

IV. Simulation and Validation Results

This section presents the validation results for the neural network model of the engine. To this end, 400 additional flight tests were conducted with the VRESIM, including 28 takeoffs at MTO thrust, 127 climbs at MCLB thrust, 123 descents at IDLE confidence, and 122 static cruises. The flight tests were carried out following the same procedures as those for the identification flight tests (see **Section III.A**). This process yielded a set of data describing engine parameters over more than 3600 operating points. Validation of the model was thus realized by comparing the sampled engine parameters (fan speed, thrust and fuel flow) with those predicted by the model. The criterion established for validating the model was the agreement between the sampled and predicted data within 5% tolerance (relative error).

Fig. 23 to Fig. 25 show three examples of successful results comparisons for the take-off, climb and descent phases. In these figures, the outputs calculated by the engine neural network model are represented in blue, while the reference outputs measured from the VRESIM are represented in black. The tolerance limits of $\pm 5\%$ represented by two dotted red lines allows to conclude on the accuracy of the results. In addition, results expressed in terms of mean errors, standard deviations, and maximum errors obtained for each flight phase are shown in Table 2 to Table 5. This statistic approach allows us to generalize the accuracy of prediction for the entire aircraft flight envelope.

A. Model validation in the Takeoff Phase

Fig. 23 shows an example of results comparison for a normal takeoff scenario. The simulation segment takes 60 seconds, during which the aircraft climbs from 0 to 1500 ft above sea level. The takeoff is divided into two sub-segments: a ground segment phase from 0 to 30 seconds, and an airborne segment from 30 to 60 seconds. As seen on Fig. 23, the engine model predicted very well the fan speed, the thrust and the fuel flow. Although there is a small difference in the beginning of the simulation of these 3 parameters, which is mainly due to the engine transient behavior, the predictions fall within the

5% tolerance limit after 5 seconds. Above this threshold, the predicted and measured data almost overlap, and the differences are close to zero.

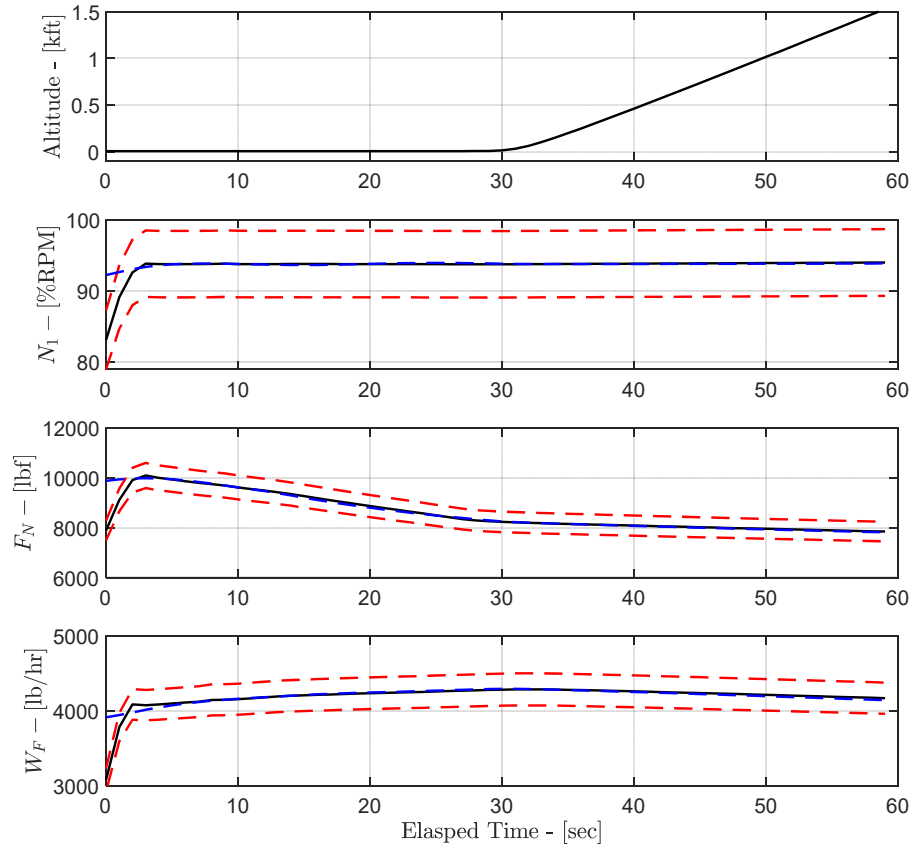


Fig. 23 Example of Comparison Results for a Takeoff Scenario.

These observations are validated by the results presented in Table 2. Indeed, for all operating points of the 28 takeoff tests, the average relative errors are close to zero. In addition, the standard deviations, as well as the maximum relative errors, are below 5%. These results therefore lead to the conclusion that the engine model is validated for the takeoff phase.

Table 2 Relative errors on validation for Takeoff phase

Engine Parameter	Mean error	Standard Deviation	Maximum
Fan speed N_1	0.05%	0.03%	0.40%
Thrust F_N	0.022%	0.17 %	1.016%
Fuel flow W_F	0.002%	0.17%	0.81%

B. Model validation in the Climb Phase

Fig. 71 shows an example of results comparison for a climb phase. The simulation takes about 8 minutes, and the climb is realized from 2000 to 40,000 ft. As for the takeoff phase, the engine model is well predicting the fan speed, the net thrust and the fuel flow, as the simulated results are included

within 5% tolerance limit. However, we can see a difference for the modeling of the fan speed N_1 at 300 seconds where the aircraft approaches its maximum ceiling altitude at 40,000 ft and prepares for a cruise flight. Although the neural networks model is not able to predict the dynamic change of the fan speed at 300 seconds, it can still predict its overall behavior with less than 5% error.

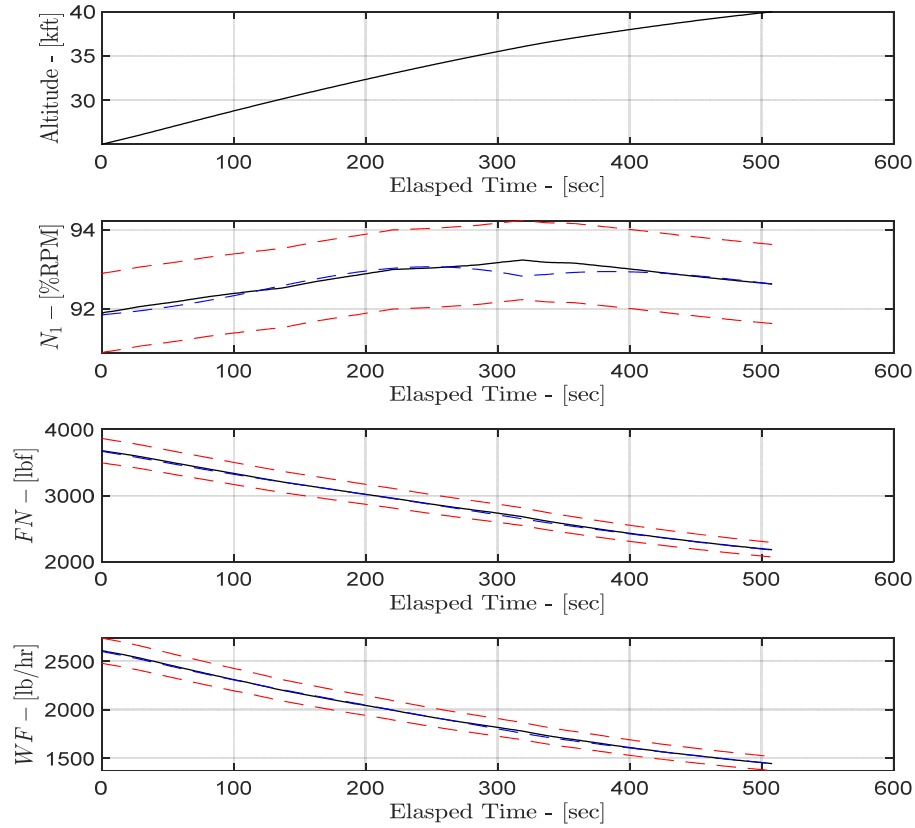


Fig. 71 Example of Comparison Results for a Climb Scenario.

From a statistical point of view, Table 3 shows that the engine modeling errors are less than 5%, where the maximum error is 1.05% for the fuel flow and for all the 127 climb tests. Thus, the engine model is validated for the climb phase.

Table 3 Relative errors on validation for climb phase

Engine Parameter	Mean error	Standard Deviation	Maximum
Fan speed N_1	0.01%	0.07%	0.19%
Thrust F_N	0.051%	0.20%	0.40%
Fuel flow W_F	0.004%	0.22%	1.05%

C. Model validation in the Descent Phase

On Fig. 25, the comparison is made for the descent phase results. The simulation takes 8 minutes, and the descent is realized for altitude from 25,000 ft to 1500 ft. The model is well predicting the fan speed N_1 and the fuel flow W_F within the 5% tolerance bound, although as for the climb phase, the model does not accurately estimate the dynamic change in N_1 at 50, 200 and 450 seconds. However,

the mean errors and standard deviations for N_1 and W_F on Table 4 are still less than 5% for the overall 123 descent tests.

As noticed in Fig. 25 and Table 4, the engine model is not able to fully predict the thrust within the 5% tolerance. But since the thrust may decrease below 0 lbf. and change its sign in IDLE thrust setting, the thrust value is relatively small (about 100 lbf. versus nearly over 10,000 lbf. for cruise, climb and takeoff). Consequently, the thrust modeling errors are negligible during descent, and as we can see on Fig. 25, the general trend of the thrust is still well estimated.

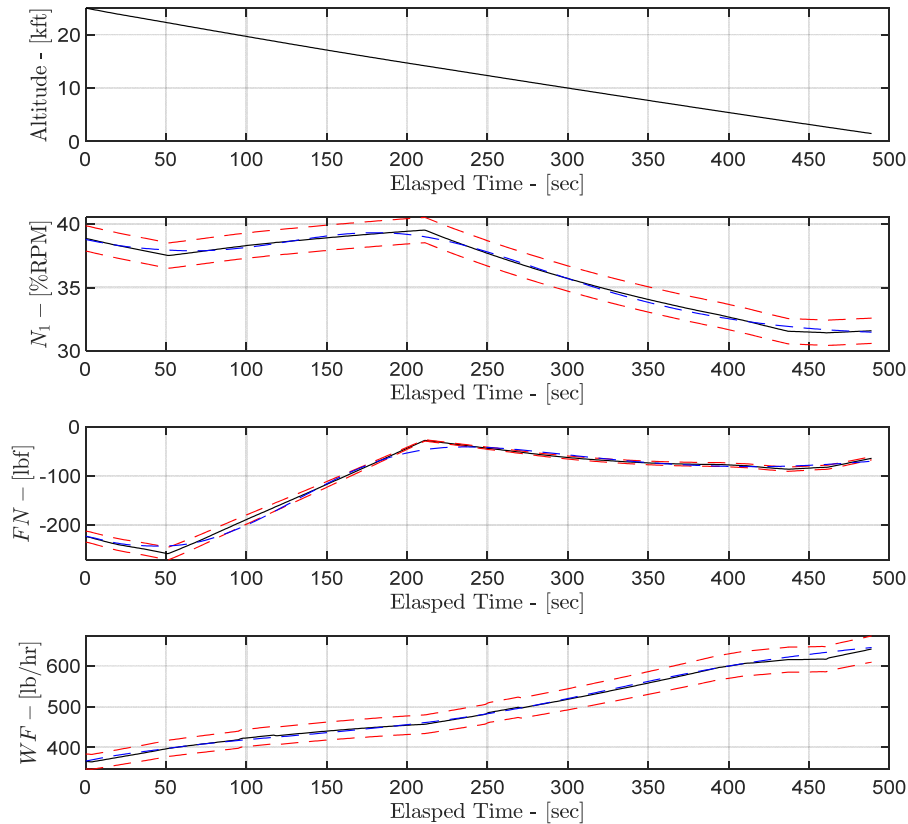


Fig. 25 Comparison of the engine model outputs with the VRESIM during descent phase.

A statistic approach on Table 4 confirms that the mean error is still less than 0.1% for the thrust, and the worse cases of simulation represent only a small percentage 0.67% of the operating points used for descent tests (9 operating points out of 1330). In that case the standard deviations and the maximum errors cannot be used to conclude on the engine model accuracy, and it is better to consider another parameter as the residual error, which is small for the thrust prediction (56 lbf). Thus, we can conclude that the engine model is validated for the descent phase.

Table 4 Relative errors on validation for descent phase

Engine Parameter	Mean error	Standard Deviation	Maximum	Residual error
Fan speed N_1	0.02%	0.50%	1.69%	2.3 %RPM
Thrust F_N	0.054%	47.9%	151%	56 lbf
Fuel flow W_F	0.034%	1.61%	9.53%	75 b/hr

D. Model validation in the Cruise Phase

The model validation for the cruise phase is performed on 122 flight tests designed for fixed cruise altitude (between 0 and 41,000 ft) and Mach number (between 0.4 and 0.82). The relative errors of the engine model are summarized in Table 5. As shown in this table, the standard deviations for the engine parameters are less than 5%, even though maximum errors reach 6.88% for few cases during fuel flow prediction. However, the overall level of prediction is still very good for all parameters, and we can conclude that the engine model is validated for cruise phase.

Table 5 Relative errors on validation for cruise phase

Engine Parameter	Mean error	Standard Deviation	Maximum
Fan speed N_1	0.25%	1.23%	3.45%
Thrust F_N	0.036%	0.82%	2.82%
Fuel flow W_F	0.24%	0.82%	6.88%

V. Conclusion

A new methodology was developed to identify the engine performance model of the General Electric CF34-8C5B1 high-bypass turbofan, using two cascading Feedforward Neural Networks and flight tests. First, the Engine Control System is modeled (ECS-NET), which calculates the corrected fan speed for given altitude, Mach number, throttle lever angle and temperature deviation. Then the corrected fuel flow and the corrected net thrust are modeled for given fan speed and Mach number (ENG-NET).

The flight tests were realized on a virtual research flight simulator (VRESIM) of the Bombardier CRJ-700 aircraft, developed by CAE Inc. The VRESIM has a Level-D qualification, which is the highest level for flight dynamics and engine propulsion modeling according to the FAA. A number of 451 flight tests covered the aircraft envelope, including takeoff, climb, cruise and descent phases.

The identification of the engine model has been performed successfully with particularly good performance. The fan speed, the fuel flow and the net thrust were predicted accurately with less than 5% error, and even 2% error on the overall flight envelope.

Single cases (less than 1%) were found during the model validation in descent phases, for which the maximum error for fuel flow and net thrust were high. In these cases, as already explained, the net thrust decreased below 0 lbf and changed its sign during Idle thrust setting. However, the residual errors were less than 75 lb/hr for the fuel flow and 60 lbf for the net thrust and were relatively small.

Further work on engine transient behavior could complete this work, and a validation for a full mission profile, including weather hazards and accelerations. One could consider another engine parameter such as the Engine Pressure Ratio (EPR), which is also well known to use along with the fan speed for engine control.

Acknowledgments

This research was performed at the Laboratory of Applied Research in Active Controls, Avionics and AeroServoElasticity research (LARCASE). The Virtual Research Simulator (VRESIM) for the Bombardier CRJ-700 was obtained by Dr Ruxandra Botez, Full Professor, thanks to the research grants that were approved by the Canadian Foundation of Innovation (CFI) and Ministère du Développement Économique, de l'Innovation et de l'Exportation (MDEIE) and the contribution of CAE Inc. For more information related on this research, please visit the LARCASE website at <http://larcase.etsmtl.ca>. The authors would like to thank CAE Inc. team, and Mr. Oscar Carranza Moyao for their support in the development of the VRESIM at the LARCASE laboratory. Thanks are also due to Mrs Odette Lacasse at ETS for her support.

BIBLIOGRAPHIE

- Andrianantara, R. P., Ghazi, G., & Botez, R. M. (2021). Aircraft Engine Performance Model Identification using Artificial Neural Networks. *AIAA Propulsion and Energy 2021 Forum*, 3247.
- Andrianantara, R. P., Ghazi, G., & Botez, R. M. (n.d.). *General Electric CF34-8C5B1 Turbofan Parameters Estimation using Artificial Neural Networks and Flight Tests*.
- Austin, A., Nelessen, A., Strauss, B., Ravich, J., Lobbia, M., Venkatapathy, E., Wercinski, P., Yount, B., Wilder, M., Cassell, A., & others. (2019). *SmallSat aerocapture: Breaking the rocket equation to enable a new class of planetary missions*.
- Bagwan, L., Raibhole, V., & Meshram, C. (2017). Full Authority Digital Engine Control: Overall System Function and its Application. *IOSR J Mech Civ Eng*, 17, 01–04.
- Bardela, P. A., & Botez, R. M. (2017). Identification and validation of the Cessna Citation X engine component level modeling with flight tests. *AIAA Modeling and Simulation Technologies Conference*, 1942.
- Bardela, P. A., Botez, R. M., Bournisien, T., & Rusovici, R. (2018). Identification and validation of the cessna citation X turbofan modelling with flight tests. *2018 AIAA Modeling and Simulation Technologies Conference*, 1920.
- Bardela, P.-A. (2017). *Identification et validation du modèle mathématique du moteur de l'avion d'affaire Cessna Citation X par essais en vol* [M.Sc.A Thesis]. École de technologie supérieure.
- Bartel, M., & Young, T. M. (2008). Simplified thrust and fuel consumption models for modern two-shaft turbofan engines. *Journal of Aircraft*, 45(4), 1450–1456.
- Bazazzade, M., Shahriari, A., & Badihi, H. (2009). Improved turbine engine hierarchical modeling and simulation based on engine fuel control system. *45th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, 5525.
- Ben Mosbah, A. (2017). *New methodologies for calculation of flight parameters on reduced scale wings models in wind tunnel* [PhD Thesis]. École de technologie supérieure.
- Benson, T. (1995). An interactive educational tool for turbojet engines. *31st Joint Propulsion Conference and Exhibit*, 3055.

- Benyounes, A., Hafaifa, A., & Guemana, M. (2016). Gas turbine modeling based on fuzzy clustering algorithm using experimental data. *Applied Artificial Intelligence*, 30(1), 29–51.
- Boely, N., Botez, R. M., & Kouba, G. (2011). Identification of a non-linear F/A-18 model by the use of fuzzy logic and neural network methods. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 225(5), 559–574.
- Botez, R. (2006). GPA-745 Introduction à l'Avionique. Notes de Cours.
- Botez, R. (2018). Morphing wing, UAV and aircraft multidisciplinary studies at the Laboratory of Applied Research in Active Controls, Avionics and AeroServoElasticity LARCASE. *Aerospace Lab*, 14, 1–11.
- Botez, R. M., Bardela, P.-A., & Bournisien, T. (2019). Cessna Citation X simulation turbofan modelling: Identification and identified model validation using simulated flight tests. *The Aeronautical Journal*, 123(1262), 433–463.
- Chapman, J. W., Lavelle, T. M., & Litt, J. S. (2016). Practical techniques for modeling gas turbine engine performance. *52nd AIAA/SAE/ASEE Joint Propulsion Conference*, 4527.
- Chapman, J. W., & Litt, J. S. (2017). Control design for an advanced geared turbofan engine. *53rd AIAA/SAE/ASEE Joint Propulsion Conference*, 4820.
- Cluzeau, J., Henriquel, X., Rebender, G., Soudain, G., van Dijk, L., Gronskiy, A., Haber, D., Perret-Gentil, C., & Polak, R. (2020). Concepts of design assurance for neural networks (CoDANN). *Public Report Extract Version 1.0*.
- Csank, J., May, R., Litt, J., & Guo, T.-H. (2010). Control design for a generic commercial aircraft engine. *46th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, 6629.
- Dufour, G., García Rosa, N., & Duplaa, S. (2015). Validation and flow structure analysis in a turbofan stage at windmill. *Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy*, 229(6), 571–583.
- Eastbourn, S. M. (2012). *Modeling and Simulation of a dynamic turbofan engine using MATLAB/SIMULINK*.

- Eastbourn, S., & Roberts, R. (2011). Modeling and Simulation of a Dynamic Turbofan Engine Using Simulink. *47th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, 6100.
- Force, R. C. A. (n.d.). *Avro Canada CF-105 Arrow*.
- Ghazi, G., Bosne, M., Sammartano, Q., & Botez, R. M. (2017). Cessna citation X stall characteristics identification from flight data using neural networks. *AIAA Atmospheric Flight Mechanics Conference*, 0937.
- Ghazi, G., & Botez, R. M. (2019). Identification and validation of an engine performance database model for the flight management system. *Journal of Aerospace Information Systems*, 16(8), 307–326.
- Ghazi, G., Botez, R. M., & Tudor, M. (2015). Performance database creation for cessna citation x aircraft in climb regime using an aero-propulsive model developed from flight tests. *AHS Sustainability*.
- Ghazi, G., Tudor, M., & Botez, R. (2015). Identification of a Cessna Citation X aero-propulsive model in climb regime from flight tests. *International Conference on Air Transport INAIR*.
- Gishvarov, A., & Raherinjatovo, J. (2017). Parametric Diagnostic State of Gas Turbine Power Plant. *Procedia Engineering*, 206, 1819–1824.
- Hagan, M. T., Demuth, H. B., & Beale, M. H. (2002). *Neural network design*. University of Colorado at Boulder; WorldCat.org.
- Hill, P. G., & Peterson, C. R. (1992). Mechanics and thermodynamics of propulsion. *Reading*.
- Ibrahim, I. M., Akhrif, O., Moustapha, H., & Staniszewski, M. (2019). Neural Networks Modelling of Aero-derivative Gas Turbine Engine: A Comparison Study. *ICINCO (1)*, 738–745.
- Ismail, I. H., & Bhinder, F. (1991). *Simulation of aircraft gas turbine engines*.
- Jategaonkar, R. V. (2015). *Flight vehicle system identification: A time-domain methodology*. American Institute of Aeronautics and Astronautics, Inc.
- Jaw, L. C., & Mattingly, J. D. (2009). Aircraft engine controls. *AIAA, Reston, VA*, 37–65.

- Joksimovic, A., Duplaa, S., Bousquet, Y., & Tantot, N. (2018). Performance prediction methodology and analysis of a variable pitch fan turbofan engine. *Aeronautics and Aerospace Open Access Journal (AAOAJ)*, 2(6), 394–402.
- Jones, S. M. (2007). *An introduction to thermodynamic performance analysis of aircraft gas turbine engine cycles using the numerical propulsion system simulation code*.
- Kanelopoulos, K., Stamatis, A., & Mathioudakis, K. (1997). Incorporating neural networks into gas turbine performance diagnostics. *Turbo Expo: Power for Land, Sea, and Air*, 78712, V004T15A011.
- 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.
- Kouba, G., Botez, R. M., & Boely, N. (2010). Fuzzy logic method use in F/A-18 aircraft model identification. *Journal of Aircraft*, 47(1), 10–17.
- Lazzaretto, A., & Toffolo, A. (2001). Analytical and neural network models for gas turbine design and off-design simulation. *International Journal of Thermodynamics*, 4(4), 173–182.
- Martucci, A., & Volponi, A. (2003). Fuzzy fuel flow selection logic for a real time embedded full authority digital engine control. *J. Eng. Gas Turbines Power*, 125(4), 909–916.
- Mathioudakis, K., & Romessis, C. (2004). Probabilistic neural networks for validation of on-board jet engine data. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 218(1), 59–72.
- Mattingly, J. D. (1996). *Elements of gas turbine propulsion* (Vol. 1). McGraw-Hill New York.
- Mattingly, J. D. (2006). *Elements of propulsion: Gas turbines and rockets*. American Institute of Aeronautics and Astronautics.
- Mattingly, J. D., Heiser, W. H., & Pratt, D. T. (2002). *Aircraft engine design*. American Institute of Aeronautics and Astronautics.
- NASA. (n.d.-a). *Gas Turbine Schematic and Station Numbers*. NASA. [https://www.grc.nasa.gov/www/k-12/airplane/turbdraw.html.].
- NASA. (n.d.-b). *Turbine Engine Thermodynamic Cycle—Brayton Cycle* [https://www.grc.nasa.gov/www/k-12/airplane/brayton.html.].

- Qiu, H., Eklund, N., Yan, W., Bonissone, P., Xue, F., & Goebel, K. (2007). Estimating deterioration level of aircraft engines. *ASME Turbo Expo 2007: Power for Land, Sea, and Air*, 661–667.
- Roberts, R. A., & Eastbourn, S. M. (2014). Modeling techniques for a computational efficient dynamic turbofan engine model. *International Journal of Aerospace Engineering*, 2014.
- Ruichao, L., Yingqing, G., Nguang, S. K., & Yifeng, C. (2018). Takagi-Sugeno fuzzy model identification for turbofan aero-engines with guaranteed stability. *Chinese Journal of Aeronautics*, 31(6), 1206–1214.
- Saravanamuttoo, H. I., Rogers, G. F. C., & Cohen, H. (2001). *Gas turbine theory*. Pearson Education.
- Segui, M., Bezin, S., & Botez, R. M. (2018). Cessna Citation X performances improvement by an adaptive winglet during the cruise flight. *International Journal of Aerospace and Mechanical Engineering*, 12(4), 423–430.
- Stamatis, A., Mathioudakis, K., & Papailiou, K. (1990). *Adaptive simulation of gas turbine performance*.
- Wilkinson, C., Lynch, J., Bharadwaj, R., & Woodham, K. (2016). Verification of adaptive systems. *Federal Aviation Administration, DOT/FAA/TC-16/4*.
- Zaag, M. (2017). *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* [M.Sc.AThesis]. École de technologie supérieure.
- 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. *AIAA Modeling and Simulation Technologies Conference*, 1941.
- Zaag, M., Botez, R. M., & Wong, T. (2019). CESSNA citation X engine model identification using neural networks and extended great deluge algorithms. *INCAS Bulletin*, 11(2), 195–207.
- Zhao, L., Mo, C., Sun, T., & Huang, W. (2020). Aero engine gas-path fault diagnose based on multimodal deep neural networks. *Wireless Communications and Mobile Computing*, 2020.

- Zhao, W., Hou, C., & Wang, Q. (2014). Diagnosis of aircraft engine performance deterioration based on support vector machines. *2014 10th International Conference on Reliability, Maintainability and Safety (ICRMS)*, 44–48.
- Zhou, D., Yao, Q., Wu, H., Ma, S., & Zhang, H. (2020). Fault diagnosis of gas turbine based on partly interpretable convolutional neural networks. *Energy*, 200, 117467.
- Zhu, P., & Saravanamuttoo, H. (1992). *Simulation of an advanced twin-spool industrial gas turbine*.