



Titre: Optimisation du routage des avions sous contraintes de
Title: maintenance par génération de colonnes

Auteur: Olfa Mezlini
Author:

Date: 2026

Type: Mémoire ou thèse / Dissertation or Thesis

Référence: Mezlini, O. (2026). Optimisation du routage des avions sous contraintes de
Citation: maintenance par génération de colonnes [Mémoire de maîtrise, Polytechnique
Montréal]. PolyPublie. <https://publications.polymtl.ca/76597/>

 **Document en libre accès dans PolyPublie**
Open Access document in PolyPublie

URL de PolyPublie: <https://publications.polymtl.ca/76597/>
PolyPublie URL:

**Directeurs de
recherche:** Guy Desaulniers, & Frédéric Quesnel
Advisors:

Programme: Maîtrise recherche en génie industriel
Program:

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

**Optimisation du routage des avions sous contraintes de maintenance par
génération de colonnes**

OLFA MEZLINI

Département de mathématiques et de génie industriel

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
Génie industriel

Mai 2026

POLYTECHNIQUE MONTRÉAL

affiliée à l'Université de Montréal

Ce mémoire intitulé :

**Optimisation du routage des avions sous contraintes de maintenance par
génération de colonnes**

présenté par **Olfa MEZLINI**

en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
a été dûment accepté par le jury d'examen constitué de :

Issmaïl EL HALLAOUI, président

Guy DESAULNIERS, membre et directeur de recherche

Frédéric QUESNEL, membre et codirecteur de recherche

Sanjay DOMINIK JENA, membre

DÉDICACE

À mes parents, piliers de ma vie, à qui je dois tout.

À mon frère, qui a su illuminer mes journées à sa manière.

À mon fiancé, pour ses encouragements, sa présence et pour avoir toujours cru en moi.

REMERCIEMENTS

Je tiens à exprimer ma profonde reconnaissance à François Soumis, que son âme repose en paix, pour m'avoir offert l'opportunité de travailler sur ce projet de recherche et pour m'avoir accompagnée avec bienveillance durant ma première année de maîtrise.

J'adresse mes sincères remerciements à mes directeurs de recherche. Je suis particulièrement reconnaissante envers Guy Desaulniers pour sa disponibilité, son encadrement rigoureux et la qualité de ses conseils tout au long de ce projet, ainsi que pour son soutien précieux dans la réalisation de ce travail. Je remercie également Frédéric Quesnel pour sa compréhension, son soutien constant et ses précieux conseils.

Je remercie vivement Issmail El Hallaoui et Sanjay Dominik Jena pour avoir accepté de faire partie de mon jury de maîtrise et pour l'intérêt porté à ce travail.

Je souhaite également remercier l'équipe du GERAD, en particulier François Lessard et Benoît Rochefort, pour leur aide et pour m'avoir permis de disposer des outils nécessaires à la réalisation de ce projet.

Je remercie l'équipe IBS pour son soutien financier, et tout particulièrement Arun Sreekumar pour son accompagnement tout au long de ce projet.

Enfin, je tiens à remercier sincèrement mes amis qui m'ont accompagnée tout au long de ces deux années à Montréal. Partager cette expérience à vos côtés restera toujours inoubliable.

RÉSUMÉ

La maintenance des avions constitue un enjeu à la fois économique et stratégique dans le secteur aéronautique, influençant directement la sécurité, la fiabilité ainsi que la rentabilité des opérations. Par conséquent, les compagnies aériennes se concentrent sur l'optimisation des coûts afin de concilier les normes de sécurité avec la disponibilité des appareils. Pour assurer une navigabilité continue, la planification de la maintenance doit être adaptée aux spécifications des avions tout en tenant compte des contraintes logistiques des installations de maintenance.

Une caractéristique clé du problème de routage de maintenance des avions sur le niveau opérationnel (*OAMRP*) est la gestion des fenêtres de maintenance flexibles. Chaque contrôle de maintenance doit être réalisé dans un intervalle de tolérance défini par un moment le plus précoce autorisé, un moment idéal pour l'utilisation optimale des ressources et une limite réglementaire absolue. L'exécution de la maintenance en dehors de cet intervalle entraîne des pénalités. Celles-ci sanctionnent soit une maintenance trop précoce, entraînant une sous-utilisation de l'appareil, soit une maintenance trop tardive, augmentant le risque de perturbations opérationnelles.

De nombreux travaux ont étudié l'*OAMRP*, proposant diverses formulations et approches. À notre connaissance, aucun modèle existant ne prend en compte simultanément, dans une formulation unique, l'ensemble des exigences opérationnelles, les intervalles de tolérance, la capacité de la main-d'œuvre et les horaires d'ouverture des stations. De plus, la plupart des modèles ne traitent qu'une flotte unique, alors qu'en pratique, plusieurs sous-flottes interagissent en partageant les mêmes ressources de maintenance.

Dans cette étude, nous abordons l'*OAMRP*, qui consiste à construire des routes pour une flotte hétérogène afin de couvrir un ensemble prédéfini de vols sur un horizon de planification donné (par exemple, deux semaines). Les routes doivent satisfaire les exigences de maintenance en termes d'heures de vol, de nombre d'atterrissages et de jours écoulés depuis la dernière maintenance, respecter les contraintes de capacité des stations de maintenance et minimiser la somme des coûts opérationnels et des pénalités liées à la maintenance. Pour résoudre ce problème, nous proposons une heuristique basée sur la génération de colonnes, reposant sur des sous-problèmes de plus court chemin avec des contraintes de ressources. Les sous-problèmes sont résolus à l'aide d'un algorithme d'étiquetage intégrant explicitement des pénalités bilatérales pour les écarts par rapport aux cibles de maintenance prédéfinies. De plus, une heuristique de réduction de taille de réseau est appliquée aux sous-problèmes afin

de réduire le temps de calcul.

L'approche proposée est validée sur un ensemble d'instances issues de cas réels, comprenant de 10 à 84 avions, 228 à 1 598 vols et jusqu'à 3 millions de connexions entre vols. Le jeu de données est divisé en trois catégories : petites, moyennes et grandes instances. Les expériences montrent que la limitation du nombre d'étiquettes par nœud et l'application de l'heuristique de réduction de taille de réseau, soit séparément, soit dans le cadre d'une stratégie hybride, ont un impact significatif. Ces techniques d'accélération permettent de réduire le temps de calcul de 13 % à 94 %, tout en maintenant un écart par rapport à la solution optimale inférieur à 1,5 %. Pour les grandes instances, la méthode a permis une réduction de 76 % du temps de calcul avec un écart de seulement 0,04 %.

ABSTRACT

Aircraft maintenance carries economic and strategic weight in aviation, directly impacting safety, reliability, and profitability. Consequently, airlines focus on cost optimization to reconcile safety standards with aircraft uptime. To ensure continuous airworthiness, maintenance scheduling is tailored to aircraft specifications while accounting for logistical constraints at maintenance facilities.

A key feature of the Operational Aircraft Maintenance Routing Problem (*OARMP*) is managing flexible maintenance windows. Each task must be performed within a tolerance interval defined by an earliest eligible time, an ideal timing for resource utilization, and an absolute regulatory limit. Executing maintenance outside this interval incurs penalties. These penalties account for scheduling maintenance either too early, leading to under-utilization of aircraft, or too late, increasing the risk of operational disruptions

Many works have studied the *OAMRP*. Different formulations and approaches have been proposed and discussed. To the best of our knowledge, no existing model simultaneously considers all operational maintenance requirements, tolerance intervals, workforce capacity, and maintenance station working hours in a single formulation. Moreover, most models address only a single fleet, whereas in practice, multiple sub-fleets interact through shared maintenance resources. Integrating these timing, capacity, and fleet interaction constraints is therefore critical to realistic and efficient aircraft maintenance planning.

In this study, we address the *OAMRP*, which consists of constructing routes for a heterogeneous fleet of aircraft to cover a predefined set of flights over a given planning horizon (e.g., two weeks). The routes must satisfy maintenance requirements on flying hours, number of landings, and elapsed days since the last maintenance, respect maintenance station capacity constraints, and minimize the sum of operational costs and maintenance related penalties. To solve this problem, we propose a column generation based heuristic that relies on shortest path subproblems with resource constraints, solved using a labeling algorithm that explicitly incorporates two-sided penalties for deviations from predefined maintenance targets. In addition, a network-reduction heuristic is applied to the pricing subproblems to reduce the computational time.

The proposed approach is validated on a set of real-world instances, including 10 to 84 aircraft, 228 to 1598 flights, and up to 3 million flight connections. The dataset is divided into three categories: small, medium, and large instances. The experiments show that limiting the number of labels per node and applying the network-reduction heuristic, either separately

or as part of a hybrid strategy, has a distinct impact on each category. These acceleration techniques reduce computational time by 13 % to 94 %, while maintaining a gap to the optimal solution below 1.5 %. For the large instances, the solution method achieved a computational time reduction of 76 % with a gap of only 0.04 %.

TABLE DES MATIÈRES

DÉDICACE	iii
REMERCIEMENTS	iv
RÉSUMÉ	v
ABSTRACT	vii
LISTE DES TABLEAUX	xi
LISTE DES FIGURES	xii
LISTE DES SIGLES ET ABRÉVIATIONS	xiii
LISTE DES ANNEXES	xiv
CHAPITRE 1 INTRODUCTION	1
1.1 Objectifs de la recherche	2
1.2 Organisation du mémoire	5
CHAPITRE 2 REVUE DE LITTÉRATURE	6
2.1 Contexte logistique de la planification aérienne	6
2.1.1 Le cadre hiérarchique de la planification aérienne	6
2.1.2 La recherche opérationnelle dans l'écosystème du transport aérien	9
2.2 Problème de routage d'avions avec contraintes de maintenance	10
2.2.1 Contraintes réglementaire et opérationnelle	11
2.2.2 Description du problème et ses variantes	14
2.2.3 Fenêtres de flexibilité de maintenance	20
2.3 Résolution par génération de colonnes	22
2.3.1 Fondement théorique de l'algorithme de génération de colonnes	22
2.3.2 Applications de la génération de colonnes au routage aérien	24
2.3.3 Intégration des contraintes de maintenance dans les sous-problèmes	25
CHAPITRE 3 DESCRIPTION ET MODÉLISATION DU PROBLÈME	28
3.1 Description du problème	28
3.1.1 Les éléments du problème	28
3.1.2 Contraintes de maintenance et pénalités associées	34
3.2 Formulation mathématique	36

CHAPITRE 4	MÉTHODE DE RÉOLUTION	39
4.1	Résolution par une heuristique de <i>branch-and-price</i>	39
4.2	Problème maître restreint	40
4.3	Sous-problèmes	40
4.3.1	Réseaux des sous-problèmes	41
4.3.2	Ressources	44
4.3.3	Résolution des sous-problèmes	45
4.4	Recherche d'une solution entière	48
4.5	Stratégies d'accélération	49
4.5.1	Stratégies d'arrêt de <i>GC</i>	49
4.5.2	Stratégies d'élimination d'arcs de connexion	50
CHAPITRE 5	RÉSULTATS NUMÉRIQUES	52
5.1	Présentation des instances	52
5.2	Implémentation et résolution	55
5.2.1	Outil de résolution	55
5.2.2	Résultats préliminaires et configuration des paramètres	56
5.3	Comparaison des résultats	58
5.3.1	Variation de $\mathbf{Lab}_{max}$	58
5.3.2	Activation de l'heuristique de réduction de la taille de réseau	62
5.3.3	Approche de résolution hybride	66
5.3.4	Recommandations des paramètres pour chaque catégorie d'instances	68
CHAPITRE 6	CONCLUSION	69
6.1	Synthèse des travaux	69
6.2	Limitations de la solution proposée et améliorations futurs	70
RÉFÉRENCES		71
ANNEXES		75

LISTE DES TABLEAUX

Tableau 3.1	Ensembles	37
Tableau 3.2	Paramètres	37
Tableau 3.3	Variables de décision	37
Tableau 4.1	Définition de l'ensemble des ressources	45
Tableau 5.1	Caractéristiques des instances	53
Tableau 5.2	Activités et lieux de maintenance	54
Tableau 5.3	Définition de l'ensemble des paramètres liés à la modélisation du problème	55
Tableau 5.4	Résultats préliminaires	56
Tableau 5.5	Performances computationnelles des résultats préliminaires	57
Tableau 5.6	Définition de l'ensemble des paramètres liés à la configuration d'algorithme	58
Tableau 5.7	Impact de Lab_{max} sur le temps et le coût total pour les petites instances	59
Tableau 5.8	Impact de Lab_{max} sur le temps et le coût total pour les moyennes instances	60
Tableau 5.9	Impact de Lab_{max} sur le temps et le coût total pour les grandes instances	62
Tableau 5.10	Performances de l'approche hybride	67
Tableau 5.11	Impact de l'arrêt de GC pour les grandes instances	68
Tableau 5.12	Synthèse des paramètres recommandés par catégorie d'instances . . .	68
Tableau A.1	Performances pour l'instance I1-A	75
Tableau A.2	Performances pour l'instance I2-A	76
Tableau A.3	Performances pour l'instance I2-C	77
Tableau A.4	Performances pour l'instance I1-C	78
Tableau A.5	Performances pour l'instance I1-B	79
Tableau A.6	Performances pour l'instance I2-B	80

LISTE DES FIGURES

Figure 2.1	Diagramme hiérarchique de la planification aérienne. Adapté de Lagos et al. [1]	9
Figure 3.1	Planification de maintenance	32
Figure 3.2	Variation de la pénalité en fonction des critères de maintenance . . .	35
Figure 4.1	Réseau pour un avion k	43
Figure 5.1	Analyse de sensibilité au Lab_{max} sur les petites instances	60
Figure 5.2	Analyse de sensibilité au Lab_{max} sur les moyennes instances	61
Figure 5.3	Analyse de sensibilité au Lab_{max} sur les grandes instances	62
Figure 5.4	Front de Pareto pour les petites instances	64
Figure 5.5	Front de Pareto pour la moyenne instance	65
Figure 5.6	Front de Pareto pour les grandes instances	66

LISTE DES SIGLES ET ABRÉVIATIONS

AMRP	Aircraft Maintenance Routing Problem
ARP	Aircraft Routing Problem
FAA	Federal Aviation Administration
GC	Génération de colonnes
LOF	Line Of Flight
MPD	Maintenance Planning Document
MRO	Maintenance, Repair and Overhaul
OAMRP	Operational Aircraft Maintenance Routing Problem
PM	Problème Maître
PMR	Problème Maître Restreint
SP	Sous-problème
SPPRC	Shortest Path Problem with Resource Constraints
TAP	Tail Assignment Problem
TSN	Time-Space Network

LISTE DES ANNEXES

Annexe A	Performances de l’heuristique de réduction de taille de réseau	75
----------	--	----

CHAPITRE 1 INTRODUCTION

L'industrie du transport aérien constitue l'un des systèmes logistiques les plus complexes et interconnectés au monde. Chaque journée d'opérations implique la coordination de milliers de vols, de flottes hétérogènes, d'équipages soumis à des règles strictes en matière de sécurité et de temps de travail, le tout dans un environnement marqué par la congestion aéroportuaire et un ensemble rigoureux de réglementations nationales et internationales.

Selon les données publiées par l'association internationale de transport aérien en 2023, la flotte mondiale comptait environ 33 271 avions, dont près de 85 % étaient en service actif [2]. Cette ampleur témoigne de la complexité croissante, de la gestion des opérations aériennes et des besoins de maintenance à l'échelle mondiale. Pour gérer cette complexité, la planification globale des opérations est généralement décomposée en une série de sous-problèmes d'optimisation interdépendants, à résoudre de manière séquentielle ou itérative.

La première étape consiste à définir le réseau de vols, quelles destinations seront desservies et selon quel horaire. Cette étape représente une décision stratégique guidée par la demande, la rentabilité des lignes et les droits d'exploitation.

Une fois le programme de vols établi, il faut déterminer quel type d'avion couvrira chaque vol, en tenant compte de la capacité requise, des caractéristiques techniques, de la disponibilité des avions et des coûts d'exploitation. Cette étape vise à utiliser la capacité de la flotte de façon optimale tout en limitant les vols sous-remplis ou surchargés.

Ensuite vient le problème de routage des avions qui vise à établir le parcours détaillé de chaque avion sur l'horizon de planification. Ce problème consiste à déterminer la séquence exacte de vols effectuée par chaque avion ; cette séquence doit être temporellement et géographiquement cohérente. Au-delà de ces contraintes classiques, une dimension essentielle s'ajoute à ce problème, qui est la maintenance des avions. En effet, pour des raisons de sécurité, de conformité réglementaire et de performance technique, chaque avion doit être soumis à des inspections et opérations de maintenance périodiques. La maintenance aéronautique se divise généralement en maintenance planifiée et non planifiée, les premières étant définies dans le document de planification des maintenances (Maintenance Planning Document, MPD) fourni par le constructeur de l'avion.

Les maintenances planifiées peuvent être classées en maintenance légère, de base et lourde. La maintenance légère regroupe les interventions rapides effectuées sur l'aire de trafic, telles que les contrôles hebdomadaires, permettant de maintenir l'avion opérationnel entre deux vols. La maintenance de base correspond aux contrôles intermédiaires, souvent effectués en hangar avec des outils spécifiques, incluant des travaux sur la cabine, le moteur ou des modifications. La maintenance lourde concerne les vérifications structurelles importantes et les grandes révisions, nécessitant des contrôles avant et après intervention pour préparer et finaliser les travaux. Certaines interventions dites non-MPD proviennent des fabricants d'équipement d'origine, de la politique de l'opérateur ou de la compagnie aérienne, par exemple pour la cabine, le moteur ou des modifications spécifiques [3]. Les maintenances non planifiées désignent l'ensemble des interventions techniques effectuées de manière imprévue, en dehors du calendrier de maintenance établi. Elle est déclenchée par la détection d'une anomalie, d'une panne ou d'un dommage survenu lors de l'exploitation ou d'une inspection.

Les maintenances planifiées peuvent être programmées selon différents critères pour garantir la sécurité et la disponibilité des avions : le nombre de jours calendaires écoulés depuis la dernière intervention, le nombre de cycles de vol (décollages–atterrissages) effectués ou encore le temps de vol accumulé. Cette planification permet d'optimiser l'utilisation des avions tout en respectant les exigences réglementaires et les contraintes opérationnelles. Dans un contexte opérationnel contraint, les compagnies aériennes doivent veiller à ce que chaque avion soit dirigé vers une station de maintenance avant d'atteindre ses limites réglementaires tout en préservant la cohérence du programme de vols et la disponibilité de la flotte. C'est dans ce contexte que s'inscrit notre travail de recherche, qui se concentre sur l'optimisation du routage d'avions tout en intégrant la planification des contrôles de maintenance.

Enfin, la dernière étape de planification des opérations consiste à élaborer l'horaire des membres d'équipage, pilotes, copilotes et personnel de cabine, en respectant les règles de temps de vol, de repos, de qualification linguistique, ainsi que les conventions collectives en vigueur.

1.1 Objectifs de la recherche

Ce projet de recherche s'inscrit dans le cadre d'un partenariat industriel avec l'entreprise *IBS*, un leader mondial dans le développement de solutions logicielles pour l'industrie du transport aérien. L'objectif principal de cette maîtrise est d'explorer une approche et de développer un outil d'aide à la décision pour optimiser le routage des avions afin de desservir autant que possible les vols programmés en tenant en compte des contraintes de maintenance, dans un environnement opérationnel soumis à de nombreuses contraintes techniques, réglementaires

et économiques.

Selon le rapport publié par Oliver Wyman [4], un cabinet international de conseil en stratégie et en management, le marché mondial de la maintenance, réparation et révision (Maintenance, Repair and Overhaul, MRO) a connu une croissance significative de 18 % en 2022, atteignant une valeur d'environ 79 milliards de dollars. Cette progression s'explique principalement par l'expansion de la flotte mondiale, l'augmentation du nombre d'heures de vol et la reprise soutenue de la demande en transport aérien après la période de ralentissement due à la pandémie. En 2023, les compagnies aériennes ont dépensé environ 94 milliards de dollars en MRO. Ce montant représente près de 11 % des coûts opérationnels totaux et environ 10 % des revenus globaux du secteur aérien [2].

Ces chiffres mettent en évidence le poids économique considérable de la maintenance dans l'industrie aéronautique et son rôle stratégique dans la sécurité, la fiabilité et la rentabilité des opérations. Dans ce contexte, la maîtrise et l'optimisation des coûts de maintenance sont devenues un enjeu central pour les compagnies aériennes, qui cherchent à concilier exigences de sécurité, disponibilité des appareils et réduction des dépenses opérationnelles.

En pratique, plusieurs types de maintenances sont requis pour assurer la navigabilité des avions, chacun ayant ses propres exigences de fréquence, de durée et de conditions de réalisation. Par conséquent, selon le type d'avion, ses caractéristiques, les heures et cycles de vol et plusieurs autres facteurs liés aux bases de maintenance, tels que la main-d'œuvre et la surface disponible du hangar, le programme de maintenance est défini. Cependant, la conversion des différents paramètres d'usage (temps de vol, nombre de cycles, etc.) en dates ou moments de maintenance standardisés, sur la base d'une utilisation moyenne estimée pour l'ensemble de la flotte, introduit des imprécisions. Cette approche conduit à planifier certaines opérations de maintenance trop tôt pour certains avions et trop tard pour d'autres, en raison de la variabilité réelle de leur utilisation [5].

Une particularité importante du contexte industriel dans lequel s'inscrit ce projet est la gestion des fenêtres de maintenance flexibles. En effet, pour chaque avion, la planification des maintenances repose non seulement sur des limites réglementaires strictes, mais également sur une valeur cible optimale, une borne inférieure et une borne supérieure définissant une zone de tolérance autour de cette cible.

- La borne inférieure définit le moment le plus précoce à partir duquel une tâche de maintenance peut être exécutée. Bien que la maintenance prématurée soit généralement évitée pour des raisons économiques, une flexibilité est introduite [5].
- La valeur cible correspond au moment idéal ou nominal où la compagnie aérienne souhaite effectuer la tâche de maintenance. Elle vise à atteindre une utilisation optimale

de l'avion ou du composant. Elle est souvent déterminée en fonction de l'âge économique d'un composant ou des intervalles standards prédéterminés pour les visites périodiques.

- La borne supérieure constitue la contrainte absolue et non négociable imposée par les autorités de navigabilité. Elle correspond au délai maximal admissible mesuré par des paramètres d'usage cumulés. Le franchissement de cette borne conduit à l'immobilisation immédiate de l'avion et rend le programme de vols irréalisable.

Dans les problèmes où les tâches ont des échéances prédéfinies, toute exécution en dehors de l'idéal entraîne un coût dans la fonction objectif. Par conséquent, l'un des aspects clés de cette recherche est donc l'intégration de doubles pénalités dans le modèle d'optimisation. Concrètement, cela se traduit par :

- Une pénalité de maintenance anticipée, appliquée lorsque l'entretien est effectué avant la borne inférieure, afin de limiter la sous-utilisation des ressources aériennes et la réduction de l'efficacité opérationnelle en immobilisant l'avion.
- Une pénalité de maintenance retardée, appliquée lorsque la maintenance est réalisée après la date cible mais avant la limite maximale, afin d'éviter un report excessif susceptible de compromettre la sécurité ou de provoquer une non-conformité réglementaire ou un risque accru de défaillance d'un composant.

Outre les coûts, les contrôles de maintenance peuvent avoir un impact négatif sur le niveau de service client en retardant les routes, voire en annulant des vols. Les compagnies aériennes doivent alors s'assurer que chaque avion soit acheminé vers une station de maintenance appropriée avant d'atteindre ses seuils réglementaires, tout en planifiant les interventions au moment le plus opportun. L'objectif est de maximiser la couverture du programme de vols et de réduire au minimum le nombre de vols annulés, afin de garantir une utilisation optimale de la flotte et une continuité du service.

Afin de résoudre ce problème complexe, nous adopterons une approche par génération de colonnes (*GC*), couplée à des heuristiques, dans le but de traiter efficacement des instances de grande taille et de proposer des solutions robustes dans un temps raisonnable. L'objectif est de développer un modèle d'optimisation capable de générer des routes d'avions viables, minimisant à la fois les coûts opérationnels et les pénalités associées aux maintenances précoces ou tardives, tout en limitant les annulations dans les programmes de vol.

1.2 Organisation du mémoire

Ce mémoire est structuré en six chapitres. Suite à ce premier chapitre d'introduction, nous proposons, dans le deuxième chapitre, une revue de la littérature sur la planification et la gestion des opérations dans le domaine aérien, et plus particulièrement sur le problème de routage des avions avec des contraintes de maintenance. Puis nous rappelons le principe de l'algorithme de *GC*, en mentionnant les types de problèmes auxquels il a déjà été appliqué. Le troisième chapitre présente une description détaillée du problème étudié, d'abord de manière textuelle, puis à travers une modélisation mathématique. Ensuite le quatrième chapitre expose la méthodologie de résolution retenue, reposant sur une approche de *GC*. Dans le cinquième chapitre, nous présentons les instances de test utilisées et les résultats expérimentaux obtenus. Finalement, au dernier chapitre, nous concluons avec un résumé des principales contributions et des pistes d'amélioration envisageables pour de futurs travaux.

CHAPITRE 2 REVUE DE LITTÉRATURE

Ce chapitre présente l'état de l'art sur notre sujet de recherche. La section 2.1 fait un tour d'horizon sur le contexte du système logistique aérien, dans lequel nous explorons les différents processus. Ainsi nous élaborons l'application de la recherche opérationnelle dans cet écosystème. Ensuite, dans la section 2.2, nous nous intéressons au problème de routage d'avion et on y détaille les différents types de maintenance. Les variantes et la classification des problèmes de planification de maintenance sont explorées dans cette section aussi, ainsi que les différentes approches de modélisation et de résolution utilisées dans la littérature. Enfin, dans la section 2.3, puisque nous avons choisi de résoudre notre problème à l'aide de la méthode de *GC*, nous retraçons l'historique de cette approche, en rappelant son principe ainsi que ses différentes composantes.

2.1 Contexte logistique de la planification aérienne

L'industrie du transport aérien est intrinsèquement caractérisée par la gestion de processus opérationnels à plusieurs niveaux. Sur le plan logistique, ces processus exigent l'exécution d'une multitude de décisions interdépendantes impliquant des actifs de très grande valeur (avions, équipages, etc.) et des ressources limitées (créniaux aéroportuaires, temps de vol, etc.).

2.1.1 Le cadre hiérarchique de la planification aérienne

Les décisions logistiques dans le transport aérien s'organisent selon leur portée et leur horizon de planification, formant une structure hiérarchique allant du stratégique à l'opérationnel. Trois niveaux principaux se distinguent alors : la planification stratégique, la planification tactique et la planification opérationnelle, interconnectés de manière séquentielle et interdépendante.

Planification stratégique (Horizon à long terme)

La planification stratégique concerne l'élaboration pluriannuelle des plans de vol et de la stratégie financière de la compagnie. Elle vise à définir la viabilité économique de l'entreprise, les grandes orientations de développement et la structure globale du réseau aérien à long terme. Les principales décisions prises à ce niveau englobent :

- La gestion des revenus et la stratégie de distribution, orientées vers la rentabilité et la compétitivité de la compagnie sur ses marchés.

- La stratégie de réseau et de flotte, incluant le choix du modèle de desserte, par exemple, *hub-and-spoke* (les passagers voyagent depuis de plus petites villes appelées *spokes* vers un aéroport central appelé *hub*, où ils changent d’avion pour atteindre leur destination finale *spoke*) ou *point-à-point* (les compagnies aériennes exploitent des vols directs entre la plupart des villes de leur réseau, sans passer par un *hub*) [6]. Ainsi que la détermination du volume global et des types d’avions nécessaires à long terme.
- La planification des horaires qui s’amorce par la génération du programme de vols. Celui-ci détermine les liaisons exactes à opérer pour la saison en fonction des objectifs stratégiques, des prévisions de trafic et des variations saisonnières. Cette première étape, souvent fondée sur les horaires des saisons précédentes, vise à ajuster le réseau pour répondre aux nouvelles conditions du marché. Elle est suivie par la prévision et l’allocation du trafic, où la demande entre différentes villes est estimée, puis répartie entre les compagnies concurrentes et les routes disponibles [7]. La conception du programme de vols aboutit à l’établissement d’un réseau initial, en détaillant les paires de vols de type origine-destination, les horaires de départ et d’arrivée ainsi que les aéroports.
- Une fois le réseau de vols fixé, les modèles d’affectation de flotte sont utilisés pour déterminer le type d’avion le plus approprié pour chaque liaison, optimisant l’utilisation des ressources définies stratégiquement. L’objectif est d’attribuer le bon type d’avion à chaque vol, maximisant la rentabilité en fonction de la demande et des contraintes opérationnelles. Ce processus est soumis à des contraintes strictes, notamment la disponibilité physique et la taille réelle de la flotte, tout en assurant la couverture complète de tous les vols.
- La planification de la maintenance lourde, qui est programmée plusieurs mois, voire des années à l’avance. En raison de l’immobilisation prolongée des appareils, ces interventions sont généralement planifiées durant les périodes creuses afin de minimiser les pertes de revenus.

Ainsi, la planification stratégique établit le cadre décisionnel et les ressources qui guideront les étapes tactiques et opérationnelles du processus global de planification aérienne.

Planification tactique (Horizon à moyen terme)

La planification tactique constitue le lien entre la vision stratégique et les opérations quotidiennes. Elle vise à transformer les orientations à long terme en programmes concrets en définissant, plusieurs mois à l’avance, les horaires détaillés et les routes de vol tout en intégrant les contraintes de maintenance à moyen terme.

Une fois le programme de vol établi et les flottes affectées aux différents segments de vols, la

problématique du routage des avions doit être résolue. Ce problème consiste à déterminer les routes complètes de chaque avion, en respectant un ensemble de contraintes de maintenances de type léger, notamment la disponibilité des services. Ces exigences de maintenance seront expliquées en détail dans la section 2.2. L'étape suivante, menée en parallèle avec le routage des avions, consiste à planifier le personnel navigant (pilotes et agents de bord), qui est la ressource la plus coûteuse après le carburant et la flotte [8].

Planification opérationnelle (Horizon à court terme)

La planification opérationnelle vise à exécuter le programme de vols au jour le jour et à gérer les imprévus. Elle constitue le point où la planification théorique rencontre les contraintes du terrain. Les principales tâches comprennent :

- La mise en œuvre de l'horaire, assurant la réalisation effective des vols planifiés.
- La gestion des perturbations, telles que les retards, les conditions météorologiques défavorables ou les pannes techniques, nécessitant des ajustements rapides pour minimiser les impacts opérationnels.

Un défi souvent englobé par l'*OAMRP* (*Operational Aircraft Maintenance Routing Problem*) ou le *TAP* (*Tail Assignment Problem*) consiste à ajuster le plan face aux difficultés opérationnelles imprévues, assurant la transition entre le plan théorique et la réalité des opérations quotidiennes. Ces deux problèmes définissent les tournées des avions en prenant en considération les conditions réelles et les contraintes opérationnelles. Le *TAP* et l'*OAMRP* seront expliqués plus en détail dans la section 2.2.

Afin de résumer ce qui précède, le schéma 2.1 illustre la hiérarchie des décisions de planification au sein d'une compagnie aérienne, organisée en trois niveaux. Au niveau stratégique, la planification des vols et l'affectation de la flotte définissent le cadre général, fortement contraint par la maintenance lourde qui détermine la disponibilité structurelle des avions. Ces décisions alimentent le niveau tactique, où la planification des équipages et la construction des routes sont initiées.

L'ensemble des défis liés à l'affectation des avions aux vols peut en effet être regroupé sous une appellation commune, le Problème de Routage d'Avions ou *ARP* (*Aircraft Routing Problem*) en anglais où sa définition varie d'un auteur à un autre. L'*AMRP* (*Aircraft Maintenance Routing Problem*) est la variante au niveau tactique, visant à établir les routes détaillées de la flotte pour satisfaire les besoins de vol tout en intégrant les exigences de maintenance légère. Le processus de routage débute ainsi au niveau tactique et s'affine progressivement, intégrant de plus en plus de contraintes à mesure que l'on se rapproche des dates de vol. Enfin, le niveau opérationnel est chargé de l'exécution finale et en temps réel, procédant à

l'assignation précise d'une immatriculation d'avion à un vol donné et gérant les imprévus.

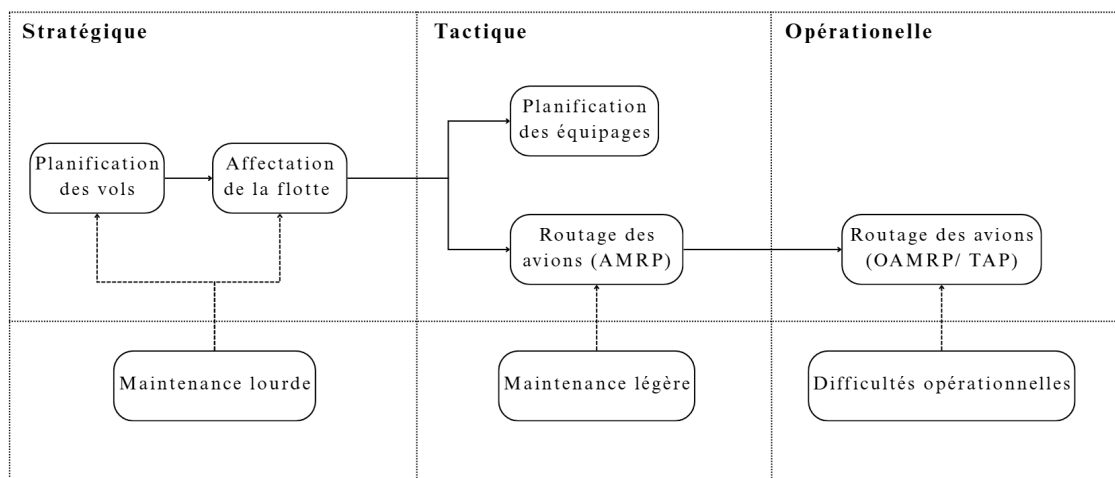


FIGURE 2.1 Diagramme hiérarchique de la planification aérienne. Adapté de Lagos et al. [1]

Cette dualité entre un processus séquentiel idéalisé et une réalité opérationnelle cyclique constitue le défi central de la recherche opérationnelle dans l'aviation.

2.1.2 La recherche opérationnelle dans l'écosystème du transport aérien

La volatilité du secteur, combinée à des coûts fixes élevés et des marges d'exploitation serrées, a historiquement contraint les compagnies aériennes à rechercher des méthodes rigoureuses d'optimisation. Face à cette complexité, elles recourent aux méthodes de recherche opérationnelle pour résoudre les problèmes associés à ces processus depuis 1950. L'origine de la recherche opérationnelle moderne remonte aux besoins militaires de la seconde guerre mondiale, où la nécessité de déplacer rapidement de vastes quantités de personnel et d'équipement a donné naissance à des algorithmes mathématiques pour trouver des solutions optimales dans des contextes logistiques complexes.

Plusieurs domaines de la recherche opérationnelle sont directement liés aux défis auxquels font face les compagnies aériennes. Parmi eux, on distingue :

- La planification, qui vise à assurer la disponibilité des ressources (avions, équipages et équipements) au bon endroit et au bon moment.
- L'affectation et l'allocation consistent à associer les personnels, les avions et les contrats les plus adaptés à chaque tâche, dans le but d'améliorer l'efficacité globale et de réduire les dépenses d'exploitation.
- L'optimisation, dont l'objectif est d'utiliser ces ressources de manière efficiente afin de minimiser les gaspillages et les coûts, tout en maximisant la rentabilité et la perfor-

mance opérationnelle.

Les problématiques qui concernent les décisions stratégiques et tactiques sont largement étudiées dans la littérature en recherche opérationnelle. Jusqu'en 1970, la recherche sur la planification des transports aériens s'est principalement concentrée sur la prévision de la demande, la planification de la flotte et le routage des avions. Les progrès durant cette période étaient limités par l'intensité de calcul requise pour trouver des solutions optimales, la programmation dynamique étant la méthode principale. Un progrès majeur a été réalisé par Levin [9] en 1971 en formulant le problème de l'affectation de la flotte comme un programme linéaire en nombres entiers, une amélioration par rapport aux programmes dynamiques. Il a cherché à minimiser la taille de la flotte en considérant des fenêtres de temps discrètes. Levin visait une solution optimale aux problèmes réels, sans heuristique, et utilisait des méthodes de *branch-and-bound* pour y parvenir. Des modèles et algorithmes pour l'affectation de la flotte sont également présentés par Abara [10] et Hane et al. [11]. Plus tard, Desaulniers et al. [12] ont présenté deux modèles d'affectation de flottes et de routage d'avions pour un horaire quotidien de vols avec fenêtres temporelles. L'un est une formulation de type partitionnement d'ensemble, et l'autre est une formulation de flux dans un réseau multi-commodités avec contraintes temporelles. Les auteurs démontrent que le premier modèle peut s'obtenir du deuxième en appliquant la décomposition de Dantzig-Wolfe. Une technique de *GC* est utilisée pour résoudre la relaxation linéaire du premier modèle qui s'imbrique dans une méthode de *branch-and-bound* pour obtenir des solutions entières. Cette méthode exploite des règles de branchement déduites du deuxième modèle.

Xu et al. [13] font une synthèse de différents travaux en mettant l'accent sur la variété des méthodes utilisées. Ils montrent que certains modèles s'appuient sur des réseaux espace-temps, d'autres sur des réseaux de connexions ou encore sur des réseaux de routes. Les études recensées couvrent également plusieurs horizons de planification, allant de l'hebdomadaire au journalier et jusqu'à l'opérationnel. Ils montrent aussi que certains travaux adoptent des modèles intégrés combinant plusieurs sous-problèmes (comme l'affectation de flotte, le routage et la planification des équipages), tandis que d'autres s'appuient sur des formulations plus spécialisées.

2.2 Problème de routage d'avions avec contraintes de maintenance

Comme expliqué dans la section 2.1, la définition de l'*ARP* varie d'un auteur à l'autre, chacun intégrant ses propres contraintes et hypothèses. Cette section présente donc les principaux travaux de recherche consacrés à l'*ARP*, ses variantes ainsi que les méthodes de résolution associées. Avant d'introduire la définition de ces différentes catégories de problèmes, nous

décrivons d'abord, dans la section 2.2.1, les contraintes réglementaires et opérationnelles auxquelles les compagnies aériennes sont confrontées. La section 2.2.2 présente ensuite l'*ARP* avec contraintes de maintenance, suivie d'un survol de ses principales variantes, puis des méthodes de résolution. Finalement, la section 2.2.3 expose la spécificité et la contribution de notre travail.

2.2.1 Contraintes réglementaire et opérationnelle

Différentes contraintes encadrent la planification aérienne, incluant les normes de sécurité, les contraintes de maintenance, les restrictions de temps et de ressources, ainsi que les particularités liées aux opérations quotidiennes des compagnies aériennes. Ces contraintes sont principalement catégorisées en deux groupes : les contraintes de conformité réglementaire et les contraintes de faisabilité opérationnelle.

Contraintes réglementaires et types de maintenance en aviation

La maintenance des avions joue un rôle essentiel pour garantir la sécurité, la fiabilité et la conformité réglementaire des opérations aériennes. Elle permet de détecter et de corriger les anomalies avant qu'elles ne compromettent la sécurité des passagers et de l'équipage. L'administration fédérale de l'aviation (FAA) aux États-Unis impose quatre types d'inspections désignés par les lettres *A*, *B*, *C* et *D* [14]. Ces quatre principaux types de maintenance diffèrent par leur étendue, leur durée et leur fréquence.

- Type *A* : c'est le premier type de maintenance obligatoire selon la FAA, qui doit être effectué généralement toutes les 65 heures de vol, soit environ une fois par semaine. Il consiste à inspecter les principaux systèmes de l'appareil, tels que le train d'atterrissage, les moteurs et les surfaces de contrôle. Il dure en moyenne 4 heures [15].
- Type *B* : elle est réalisée toutes les 300 à 600 heures de vol, elle comprend une inspection visuelle approfondie accompagnée de la lubrification de toutes les pièces mobiles. Sa durée varie entre 10 et 15 heures [15].
- Types *C* et *D* : ces types de maintenance sont effectués à intervalles relativement longs. La maintenance de type *C* est une inspection approfondie effectuée environ une fois par an et nécessite l'immobilisation de l'avion pendant une à deux semaines. La maintenance de type *D* est considérée comme l'inspection la plus complète et la plus exigeante ; elle comprend notamment des opérations de démontage, de décapage, de peinture et de rénovation de la cabine. Elle entraîne une immobilisation pouvant varier de trois semaines à deux mois et est généralement réalisée tous les quatre ans [16].

Les compagnies aériennes doivent impérativement se conformer aux réglementations émises par la FAA. Cependant, au-delà des visites de maintenance périodiques (*A*, *B*, *C* et *D*), d'autres autorités de contrôle aérien imposent également leurs propres exigences réglementaires. Ces inspections supplémentaires peuvent être requises avant d'autoriser le survol ou l'atterrissage dans une zone donnée en fonction des règles de maintenance internes de la compagnie aérienne et des exigences propres aux zones d'opération. Chaque transporteur élabore son programme de maintenance approuvé à partir du rapport du comité d'évaluation de la maintenance, du MPD du constructeur et des exigences réglementaires en vigueur, ce qui peut conduire à l'ajout de tâches de maintenance renforcées en fonction des conditions opérationnelles [17] [18]. Ces exigences sont généralement formulées sous forme de limites en temps de vol cumulé, en nombre de cycles ou en nombre de jours calendaires, au-delà desquelles une inspection ou une action de maintenance devient obligatoire.

- **Restrictions sur le temps de vol cumulé** : le temps total de vol entre deux opérations de maintenance successives ne doit pas dépasser une limite maximale autorisée.
- **Restrictions sur le nombre de cycles** : le nombre de décollages/atterrissages effectués entre deux maintenances successives est également limité par une valeur maximale.
- **Restrictions liées au nombre de jours calendaires** : l'avion doit obligatoirement être soumis à une visite de maintenance avant de dépasser l'intervalle maximal fixé en jours.

Dans la littérature, différents termes sont utilisés pour désigner un même type de maintenance, sans qu'il n'existe de terminologie uniforme. Les maintenances de types *A* à *D* sont largement employées, mais leur classification n'est pas uniforme : certains auteurs distinguent les maintenances de court et de long terme, tandis que d'autres les classent selon la complexité et la durée des interventions. D'autres notions, telles que la maintenance planifiée, préventive, corrective ou en ligne, sont également fréquemment utilisées, souvent sans frontières clairement définies.

Afin de clarifier ces définitions, Van den Bergh et al. [19] proposent un cadre de référence basé sur les classifications de maintenance les plus couramment employées. Ce cadre distingue les maintenances selon l'intensité de la charge de travail, allant des interventions à court terme, fréquentes et légères, aux opérations à long terme, plus rares et plus lourdes. Il permet également de différencier les maintenances selon leur niveau d'incertitude, en séparant les maintenances planifiées des interventions non planifiées. La taxonomie distingue également les maintenances en fonction de leur lieu d'exécution.

La maintenance en ligne regroupe les interventions pouvant être réalisées directement à la porte d'embarquement ou sur l'aire de trafic. Elle inclut l'ensemble des tâches de court terme ainsi qu'une partie des maintenances de moyen terme, qu'elles soient planifiées ou non. En revanche, la maintenance lourde, qui comprend les maintenances C et D , est beaucoup plus complexe : elle nécessite du temps, du personnel spécialisé et des équipements lourds. C'est pour cette raison que les maintenances C et D ne peuvent être réalisées que dans des hangars spécialisés.

On peut conclure que d'une façon générale, la maintenance des avions s'effectue selon une série de vérifications périodiques, de plus en plus approfondies, à l'exception des réparations imprévues. La fréquence de ces contrôles dépend du nombre d'heures de vol, du nombre de cycles de et/ou du nombre de jours calendaires, du type d'avion ainsi que des politiques internes des compagnies aériennes. Elle peut être réalisée sur tout site disposant des équipements adéquats.

Contraintes opérationnelles

Une connexion correspond à la transition entre deux vols consécutifs opérés par le même avion. Pour qu'elle soit considérée comme réalisable, cette transition doit satisfaire un ensemble de contraintes opérationnelles qui constituent la condition de faisabilité. Cette condition s'articule autour de deux critères principaux, comme suit :

1. Condition de temps minimum (Minimum Ground Time) : cette contrainte garantit qu'un délai au sol suffisant est respecté entre l'arrivée du premier vol et le départ du second. Ce temps minimum est essentiel pour l'exécution des tâches opérationnelles critiques : le nettoyage, le changement et le repos des équipages, l'embarquement des passagers et la préparation de l'avion. Lorsqu'une maintenance est programmée durant cette transition, la durée de la connexion doit être étendue afin d'intégrer le temps d'intervention requis, garantissant ainsi la bonne exécution des opérations de maintenance avant le prochain départ.
2. Condition de continuité logistique (ou d'aéroport) : elle impose que l'aéroport d'arrivée du premier vol soit identique à l'aéroport de départ du second vol. Cette règle assure la continuité physique de la route d'avion sur le réseau.

Ces contraintes de base sont complétées par les conditions initiales, qui spécifient l'aéroport de départ assigné à chaque avion au début de l'horizon de planification et à quel moment l'avion y devient disponible.

La planification de la maintenance doit également prendre en compte la capacité des stations

de maintenance. Celles-ci sont souvent limitées en termes d'heures-main-d'œuvre disponibles et de places de stationnement. Ainsi, le nombre d'avions pouvant être maintenus simultanément dans une station ne peut dépasser la capacité de sa main-d'œuvre. Négliger cette contrainte peut entraîner des temps d'attente prolongés et générer des coûts supplémentaires. Aussi, chaque station n'est pas nécessairement équipée pour réaliser tous les types de maintenance : certaines sont spécialisées pour des visites ou opérations spécifiques. De plus, les stations de maintenance disposent de plages horaires de fonctionnement spécifiques. Les avions doivent arriver durant ces horaires afin d'éviter des temps d'attente excessifs et de potentielles annulations de vols.

2.2.2 Description du problème et ses variantes

Une compagnie aérienne exploite une flotte composée de plusieurs sous-flottes, chacune correspondant à un type d'avion donné (par exemple, Airbus A320 ou Boeing 737-800). Chaque sous-flotte regroupe plusieurs avions, chacun étant identifié par un numéro de queue ou une immatriculation unique. Les avions appartenant à une même sous-flotte partagent des caractéristiques techniques communes, notamment en termes de capacité, de performances opérationnelles et de programme de maintenance. Toutefois, malgré cette homogénéité au niveau du type d'avion, des différences individuelles peuvent exister entre les appareils, telles que l'âge, l'historique de maintenance ou l'état technique, ce qui peut influencer leur planification opérationnelle.

Par conséquent, l'*ARP* consiste à créer l'ensemble optimal de routes individualisées couvrant la totalité ou la majorité des vols requis sur un horizon défini. Dans le contexte d'*ARP* avec intégration des contraintes de maintenance, dans les systèmes réels, les compagnies aériennes doivent gérer beaucoup de règles pour satisfaire aux contraintes réglementaires et opérationnelles. Bien que les logiciels d'optimisation commerciaux doivent intégrer la complexité totale des exigences logistiques, les modèles académiques de routage d'avion se concentrent principalement sur un sous-ensemble de contraintes fondamentales partagées par l'industrie. Cela permet de mieux comprendre la structure d'*ARP* et d'éviter des modèles trop complexes à résoudre.

En effet, les modèles académiques du problème de routage avec maintenance utilisent une version simplifiée car ils sont relativement faciles à implémenter. Ils se concentrent surtout sur la continuité géographique et temporelle entre les vols. Dans les premiers travaux, les auteurs se focalisaient surtout sur la résolution du problème de routage lui-même, souvent traité comme un problème de faisabilité où ils n'autorisaient les inspections que durant la nuit afin de simplifier la planification [20]. D'autres travaux considèrent en plus un seul

type d'inspection principale, réalisée de façon cyclique avec une durée fixe. Par exemple, Papadakos [21] étudie seulement les inspections de maintenance de type *A*, qui doivent être effectuées tous les 4 jours et durent environ 8 heures. Aussi Liang et al. [22] ont proposé que lorsqu'un avion doit effectuer une vérification de maintenance tous les *D* jours, le programme de vols quotidien peut être répliqué sur une période de *D* jours, et les maintenances ne sont effectuées que durant la nuit. Finalement, certaines versions du problème incluent également des restrictions sur le temps total en vol entre deux maintenances successives. Pour optimiser l'utilisation de l'appareil, des coûts de pénalité sont introduits afin de cibler le moment idéal de l'intervention [23].

De manière générale, l'*ARP* consiste à construire des enchaînements de vols cohérents pour chaque appareil tout en minimisant les coûts opérationnels, qu'il s'agisse des coûts de correspondance, des pénalités liées aux perturbations ou des surcoûts induits par le non-respect des exigences de maintenance. Lorsqu'on y intègre les contraintes de maintenance, le problème peut être formulé selon différents horizons de planification. La littérature distingue principalement deux niveaux ; tactique et opérationnel, chacun se caractérisant par des objectifs, des contraintes et des conditions initiales propres.

Planification des maintenances sur le plan tactique

Le problème de la planification de la maintenance aérienne au niveau tactique, ou l'*AMRP* dans la littérature, vise à établir des routes et des calendriers pour les avions, souvent organisés selon un modèle cyclique où les mêmes séquences de vols se répètent régulièrement [21] [22]. L'*AMRP* est généralement résolu plusieurs mois avant le début des opérations, une flotte à la fois. Cet horizon de planification étendu permet d'intégrer les inspections de type *A* et *B*, réalisées à des fréquences hebdomadaires ou mensuelles. En revanche, les contraintes opérationnelles fines ne peuvent pas encore être prises en compte à ce stade, puisque la disponibilité de la main-d'œuvre ainsi que les capacités des ressources ne sont pas connues avec précision. Par conséquent, les inspections de maintenance planifiées sont fréquemment ajustées par les planificateurs opérationnels afin de répondre aux exigences détaillées du jour d'exploitation [24]. L'objectif principal de cette planification est d'optimiser l'utilisation de la flotte tout en minimisant les coûts et en respectant les exigences réglementaires à moyen terme. Toutefois, cette approche présente certaines limites : sa rigidité la rend peu adaptée aux perturbations opérationnelles fréquentes dans l'industrie aérienne, susceptibles d'entraîner des re-calculs coûteux lorsque les plans initiaux doivent être ajustés, voire des annulations de vols lorsque des contrôles de maintenance critiques doivent être réalisés.

Planification des maintenance sur le plan opérationnel

Les décisions prises plusieurs mois à l'avance reposent sur des informations limitées et ne peuvent pas intégrer toutes les caractéristiques réelles et individuelles de chaque avion, telles que son historique spécifique de maintenance ou encore son état opérationnel actuel. À l'opposé de la planification sur le plan tactique, la planification opérationnelle se concentre sur le court terme, avec un horizon temporel allant de quelques jours à quelques semaines.

L'*OAMRP*, également appelé *TAP*, correspond à un problème de planification à court terme ; il vise à générer des routes pour chaque avion de la flotte, tout en assurant la couverture de l'ensemble des vols planifiés sur l'horizon considéré. L'*OAMRP* intervient à un niveau opérationnel lorsque des perturbations telles que des retards, des pannes ou des conditions météorologiques imprévues modifient l'historique de chaque avion. Par conséquent, les plans initiaux ne peuvent plus être réalisables. Il ne s'agit alors plus de rechercher une optimisation économique à long terme, mais d'adapter rapidement les routes des avions afin de rétablir la faisabilité tout en respectant les contraintes de maintenance.

Il est important de distinguer l'*OAMRP* du problème de la ré-optimisation en temps réel. Ce dernier intervient immédiatement lors de perturbations opérationnelles, telles que des retards, des pannes ou des conditions météorologiques défavorables. Il se concentre généralement sur une portion locale du problème et vise principalement à assurer la continuité des opérations jusqu'à la fin de la journée, souvent en coordination avec l'ajustement des horaires de vols et des équipages. En revanche, l'*OAMRP* est résolu de manière périodique, typiquement en fin de journée, afin de prendre en compte les impacts de l'ensemble des perturbations survenues au cours de la journée. Il permet ainsi de reconstruire un plan global cohérent pour les jours suivants, en tenant compte de l'état actualisé des avions et des contraintes de maintenance.

Dans ce travail, nous nous intéressons à ce niveau de planification, où le principal enjeu consiste à intégrer simultanément l'ensemble des contraintes opérationnelles et réglementaires, les maintenances déjà planifiées au niveau de l'*AMRP*, les contrôles de maintenance requis et la nécessité de couvrir un maximum de vols avec un coût total minimal, afin de préserver la continuité des opérations.

Approches de résolution

En 1989, Feo et Bard [25] figurent parmi les premiers à intégrer des contraintes de maintenance dans la résolution de l'*ARP*. Ils formulent le problème comme un problème de flux dans un réseau à coût minimum. Le modèle vise à déterminer le nombre optimal de stations de maintenance ainsi que leur localisation pour desservir l'ensemble des vols programmés.

La résolution est effectuée à l'aide d'une heuristique en deux phases. Le modèle dans le travail de Feo et Bard met principalement l'accent sur la détermination des stations de maintenance plutôt que sur la construction des routes individuelles pour chaque avion. Une autre catégorie de travaux présente dans la littérature qui, au contraire, se concentrent sur la génération des routes de chaque appareil en s'appuyant sur le concept de ligne de vol (*Line Of Flight, LOF*)¹. La résolution du problème repose alors sur la génération de l'ensemble des *LOFs* réalisables, parmi lesquels le modèle sélectionne la combinaison optimale. Kabbani et Patty [26] figurent parmi les premiers à utiliser ce concept en 1992 ; ils ont proposé un modèle de partitionnement d'ensemble visant à identifier les routes d'avions permettant d'assurer une maintenance nocturne tous les trois jours. Leur approche de résolution s'effectue en deux étapes, la construction des *LOFs*, puis la connexion de ces routes afin de générer les tournées complètes. Par la suite en 1998, Gopalan et Talluri [20] étendent ce principe pour traiter un horizon de k jours. Ils développent un algorithme polynomial permettant de déterminer des routes réalisables du point de vue de la maintenance. Cet algorithme est appliqué à la fois à une version statique du problème, où les routes sont prédéterminées, et à une version dynamique où elles peuvent être modifiées. Les auteurs montrent aussi que, lorsque les routes peuvent être réajustées, la version à deux jours devient *NP-difficile*.

D'autres travaux ont adapté la modélisation en considérant un réseau de connexions, où les nœuds représentent les vols avec un départ et une arrivée à un instant précis. Les arcs représentent les connexions réalisables entre une arrivée et un départ ultérieur au sein d'une même station, sous réserve du respect du temps minimum d'escale. Abara [10] fut l'un des premiers chercheurs à proposer une formulation en réseau de connexions pour résoudre un problème réaliste d'affectation de flotte. Clarke et al. [27] proposent un modèle avec cette formulation ; les contraintes de maintenance sont formulées explicitement au moyen de variables supplémentaires. L'objectif de leur travail était de déterminer des *rotations*² réalisables avec maintenance, en maximisant le profit généré, selon une formulation inspirée du problème du voyageur de commerce asymétrique.

La modélisation la plus couramment utilisée pour formuler le problème d'affectation, ainsi que les problèmes de planification aérienne en général, est le réseau espace-temps (*Time-Space Network, TSN*). Dans ce type de réseau, les nœuds représentent les villes de départ et d'arrivée à différents instants, tandis que les arcs correspondent aux vols programmés. Plusieurs travaux utilisent cette représentation, notamment [28]. Cependant, un réseau espace-temps présente certaines limites, en particulier l'impossibilité d'identifier explicitement un avion

1. Un LOF correspond à une suite de vols pouvant être effectués par un avion durant une journée complète.

2. Une rotation désigne la séquence de vols effectuée par un même avion sur une période donnée, en respectant la continuité dans le temps et dans l'espace.

spécifique lorsqu'il est au sol. Cette limitation restreint son utilisation lors de l'étape suivante de la planification, à savoir le routage des avions. Par conséquent, le problème est souvent formulé comme un problème de flot multi-produits en nombres entiers mixtes. La formulation de Liang et al. [22] est sur la base d'une structure compacte du réseau *TSN* avec une simplification des exigences de maintenance ; seul le nombre maximal de jours calendaires accumulés est pris en compte.

Sriram et Haghani [15] ont étudié l'*OAMRRP* comme un problème de routage hebdomadaire intégrant les maintenances de type *A* et *B*. Ils ont proposé un modèle de programmation linéaire en nombres entiers intégrant deux types de maintenance, une visite de maintenance tous les quatre jours, une visite de maintenance toutes les huit semaines et une contrainte de capacité de la main-d'œuvre. Dans leur approche, les connexions entre vols sont supposées connues à l'avance et les opérations de maintenance sont réalisées exclusivement durant la nuit. L'objectif du modèle était d'affecter chaque avion à des routes quotidiennes tout en minimisant à la fois les coûts de maintenance et les coûts liés à une éventuelle réaffectation. Une heuristique a été proposée pour résoudre le modèle. Les auteurs ont également envisagé une extension incluant les heures de vol cumulées, mais celle-ci n'a pas été résolue en raison de sa complexité élevée.

Temucin et al. [29] ont examiné la littérature sur les opérations aériennes effectuées entre 2009 et 2019, en mettant l'accent sur le sujet de la planification des maintenances. Les auteurs ont montré que bien que les contraintes opérationnelles permettent une planification des interventions de maintenance tout au long de la journée, il y a une prédominance des modèles privilégiant la planification nocturne. Les premiers travaux de recherche se sont principalement concentrés sur la sélection de connexions entre les vols afin de construire des routes complexes, tout en planifiant l'intégralité des contrôles de maintenance durant la nuit. Cette approche visait à maximiser l'utilisation des avions durant la journée tout en respectant les contraintes réglementaires de maintenance pendant les périodes où les avions sont inactifs.

Sarac et al. [24] ont adapté le concept de *high-time aircraft*, lorsqu'un avion atteint un nombre d'heures de vol accumulées proche d'un certain seuil depuis sa dernière maintenance, son temps de vol restant devient insuffisant pour poursuivre ses opérations sans risque. Il doit être obligatoirement acheminé vers une station de maintenance durant l'horizon de planification en cours en priorité. Ils ont formalisé ce problème en intégrant explicitement la disponibilité des ressources de maintenance. Leur objectif principal était de minimiser le temps de vol inutilisé restant pour les avions sur un horizon de planification journalier. Pour ce faire, ils ont modélisé le problème comme un problème de partitionnement d'ensemble basé sur un réseau de connexions en ne tenant compte que des heures de vol maximales et de la capacité

de la main-d'œuvre comme contraintes de maintenance, tout en négligeant les autres. Comme ce type de formulation génère un nombre exponentiel de routes réalisables, les auteurs ont eu recours à la *GC* comme méthode de résolution.

Başdere et Bilge [23] ont proposé un modèle en programme linéaire en nombres entiers pour un horizon de planification hebdomadaire afin d'optimiser l'utilisation du temps de vol restant, en appliquant un algorithme de *branch-and-bound* et une méthode d'heuristique.

D'autres travaux d'*OAMRP* s'orientent vers une modélisation explicite des tâches de maintenance plutôt que de contrôles périodiques. Dans ce contexte, Ruther et al. [30] proposent un modèle intégré qui considère simultanément le routage des avions, les exigences de maintenance spécifiques à chaque appareil ainsi que la planification des équipages. Le problème est résolu à l'aide d'une approche par *GC*. Plus récemment, Lagos et al. [1] développent un cadre intégré dans lequel les décisions d'affectation des avions aux vols et de planification de la maintenance sont prises conjointement à partir d'un ensemble de vols planifiés. Dans leur approche, les tâches de maintenance sont révélées dynamiquement au cours du temps et sont supposées être effectuées durant la nuit, conformément aux pratiques traditionnelles de l'industrie aérienne.

Ruan et al. [31] ont pris en compte trois contraintes de maintenance : le nombre maximal d'heures de vol, la limite de nombre de cycles et la capacité de la main-d'œuvre. Les auteurs proposent un modèle de programmation en nombres entiers basé sur les flux dans un réseau ainsi qu'un algorithme fondé sur l'apprentissage par renforcement.

Wen et al. [32] construisent des réseaux de vols spécialisés pour les avions de type *non high-time aircraft* et *high-time aircraft* en s'appuyant sur le concept introduit par Sarac et al. [24], afin de garantir l'existence d'opportunités de maintenance réalisables. Le problème est résolu à l'aide d'une approche par *GC*, dans laquelle le problème maître (*PM*) sélectionne les routes avions, tandis que les sous-problèmes (*SPs*) sont formulés comme des problèmes de plus court chemin sous contraintes de ressources, résolus par un algorithme d'étiquetage. Les auteurs autorisent le recours à une stratégie de maintenance avec des ressources supplémentaires, consistant à déployer temporairement du personnel et des équipements de maintenance dans des aéroports ne disposant pas de station dédiée, moyennant un coût additionnel. Par ailleurs, le modèle permet la reprogrammation des vols, notamment par leur annulation assortie de pénalités élevées, tout en minimisant l'impact sur le plan initial.

Es Yurek [33] a adapté le cadre défini par Sarac et al. [24] en supposant que les avions nécessitant une maintenance sont préalablement identifiés. Le modèle impose que les avions non prioritaires (*non high-time aircraft*) ne subissent aucune maintenance durant la période de planification courante. Pour garantir la faisabilité opérationnelle, il suppose que ces avions

disposent d'un temps de vol restant suffisant pour parcourir la route la plus longue possible sans requérir de maintenance. L'objectif principal de ce travail consiste à optimiser l'utilisation du temps de vol restant avant la maintenance pour les avions prioritaires. Afin de résoudre ce problème d'optimisation, il a proposé une méthode de décomposition combinatoire de Benders, en appliquant l'algorithme de *branch-and-Benders-cut*.

D'après la littérature présentée ci-dessus, la modélisation du routage des avions avec contraintes de maintenance ne considère que les types de maintenance les plus fréquents et quelques contraintes de capacité. Les objectifs d'optimisation varient selon les modèles, certains visent la réduction des coûts ou l'optimisation du temps restant de vol avant d'atteindre le seuil de maintenance, afin de maximiser l'utilisation des avions. D'autres travaux se concentrent seulement sur la faisabilité opérationnelle. Pour résoudre le problème de routage d'avions avec contraintes de maintenance, des méthodes exactes, des méthodes heuristiques ou des méthodes hybrides ont été adaptées selon les approches de modélisation. L'analyse de Temucin et al. [29] démontre une adoption limitée des méthodes exactes de résolution. Cette préférence s'explique par la complexité *NP-difficile* du problème [34].

2.2.3 Fenêtres de flexibilité de maintenance

Dans la littérature existante, la fonction objectif des modèles de routage d'avions varie selon les priorités de chaque étude et n'intègre pas toujours l'ensemble des coûts ou pénalités liés au respect des fenêtres de maintenance. Certains travaux montrent que, pour une flotte donnée, les coûts de maintenance totaux restent relativement stables [35]. Toutefois, le coût effectif d'un contrôle de maintenance dépend fortement de l'avion concerné et de l'aéroport où l'intervention peut être réalisée, en raison des variations locales de capacité, de main-d'œuvre et d'organisation. D'autres auteurs associent un coût spécifique à chaque type d'inspection. Par exemple, Sriram et Haghani [15] considèrent dans leur étude les maintenances de types *A* et *B* : la première doit être effectuée tous les quatre jours, tandis que la seconde intervient environ tous les deux mois. Comme leur horizon de planification est d'une semaine, ils attribuent le coût complet pour les inspections de type *A*, mais répartissent celui de la maintenance de type *B* sur la période considérée. L'objectif n'est donc pas d'imposer la réalisation de l'inspection *B* à chaque cycle hebdomadaire, mais plutôt d'en lisser le coût réel sur l'horizon. Ainsi, lorsque le programme hebdomadaire est répété sur huit semaines, un huitième du coût total de l'inspection *B* est intégré dans la fonction objectif. À l'opposé de ces approches, notre travail prend explicitement en compte les pénalités associées aux écarts par rapport aux cibles de maintenance, ainsi que des seuils flexibles permettant d'avoir des fenêtres d'intervention. En effet, la planification de la maintenance trop tôt conduit à une utilisation

inefficace des avions et des ressources de maintenance. Cette situation est particulièrement indésirable dans l'industrie aéronautique, qui supporte des coûts considérables par rapport à de nombreux autres secteurs, tant en termes d'investissement, de main-d'œuvre, d'exploitation que de maintenance. À l'inverse, un avion qui viole les exigences de maintenance n'est pas autorisé à voler, ce qui entraîne des pénalités financières sévères.

Safaei et Jardine [36] indiquent que l'approche optimale serait de réaliser simultanément la rotation de chaque avion tout en surveillant en même temps l'évolution des besoins de maintenance à l'aide de divers attributs de vol (heures de vol, cycles, temps calendaires). Une telle approche garantirait la disponibilité de créneaux de maintenance suffisants et réalisables pour couvrir les fenêtres de maintenance prévues, mais elle est impraticable en raison de sa complexité combinatoire élevée. Pour répondre à cette difficulté, les auteurs proposent une approche décisionnelle itérative visant à simuler une interaction optimale entre le routage des avions et la planification de la maintenance. Sur un horizon hebdomadaire, des rotations sont construites pour chaque avion et les fenêtres de maintenance correspondantes sont projetées afin de réduire le désalignement entre les opportunités de maintenance et les fenêtres de maintenance dues. Par ailleurs, les auteurs introduisent un modèle mathématique fondé sur un réseau de connexions, dans lequel l'horizon de planification est discrétisé et des contraintes spécifiques sont ajoutées afin de garantir un nombre suffisant d'opportunités de maintenance pour chaque avion.

D'après les travaux présentés ci-dessus et à notre connaissance, il n'existe aucun modèle qui prenne en compte, dans une seule formulation, l'ensemble des exigences opérationnelles de maintenance tout en intégrant la flexibilité offerte par des intervalles de tolérance, en plus de la capacité de la main-d'œuvre et des heures de travail des stations de maintenance. De plus, les modèles existants considèrent généralement une seule flotte à la fois, alors qu'en pratique plusieurs sous-flottes sont planifiées simultanément et interagissent à travers des ressources communes, notamment les capacités des stations de maintenance. Les intervalles de tolérance sont définis par une borne inférieure (maintenance anticipée possible), une borne optimale (moment idéal de maintenance) et une borne supérieure (seuil maximal à ne pas dépasser). Ce travail vise à combler cette lacune en proposant un modèle intégrant l'ensemble de ces contraintes dans un cadre unifié, tout en tenant compte de la planification simultanée de plusieurs sous-flottes. Un algorithme de résolution efficace est également développé afin de traiter des instances de taille moyenne et grande. Cette modélisation plus fine vise à refléter de manière réaliste les pratiques opérationnelles.

2.3 Résolution par génération de colonnes

L'*OAMRP* est un problème d'optimisation combinatoire de grande taille, dont la résolution efficace repose sur la combinaison de techniques de programmation linéaire et de méthodes de décomposition. À cet égard, Desrosiers et al. [37] ont démontré la pertinence des algorithmes de *branch-and-price* pour une vaste classe de problèmes de transport et de confection d'horaires, incluant le routage d'avions. Ces approches s'appuient sur la *GC*, une méthodologie puissante permettant de traiter des problèmes pour lesquels l'ensemble de variables de décision ne peut être énuméré explicitement [38]. Cette section présente les fondements théoriques de la *GC*, le *SP* et le calcul des coûts réduits, ainsi que ses principales applications au routage aérien. Une attention particulière est enfin portée à l'intégration des contraintes de maintenance dans le *SP*.

2.3.1 Fondement théorique de l'algorithme de génération de colonnes

La méthode de *GC* a été introduite en 1960 par Dantzig-Wolfe [39] et a été appliquée pour la première fois par Gilmore et Gomory en 1961 [40] dans le cadre de la résolution d'un problème de découpe. Elle repose sur le principe de la décomposition de Dantzig-Wolfe, qui consiste à reformuler un problème d'optimisation en un *PM* et un *SP*. Le *SP* peut se séparer en plusieurs *SPs* comme dans le cas de l'*OAMRP* où il y a un *SP* par avion.

Ce principe de décomposition permet, dans un premier temps, de reformuler un modèle initial, qu'il soit linéaire ou non linéaire, et de grande dimension ou de taille polynomiale par rapport aux données d'entrée, en un programme linéaire équivalent. Dans un second temps, la reformulation donne un programme linéaire qui contient un très grand nombre de variables, appelé *PM*, et un ensemble de *SPs* qui permettent d'identifier les variables du *PM*.

Dans ce cadre, le problème original est reformulé de manière à ce que chaque variable du *PM* représente une solution élémentaire faisable d'un *SP*, communément appelée colonne. Étant donné que l'énumération explicite de toutes les colonnes réalisables est généralement impossible en pratique, la *GC* consiste à ne considérer qu'un sous-ensemble restreint de ces variables dans le *PM*, qui s'appelle alors le problème maître restreint (*PMR*).

Problème maître restreint

Le *PMR* est initialement formulé à partir d'un ensemble limité de colonnes. Il est ensuite résolu sous forme de relaxation linéaire. La résolution du *PMR* fournit non seulement une solution primale, mais également des valeurs duales associées aux contraintes, lesquelles jouent un rôle central dans la suite de l'algorithme.

La procédure de résolution est décrite par l’Algorithme 1. Elle consiste en une alternance entre le *PMR* et les *SPs*. À chaque fois que le *PMR* est résolu, les variables duales pour chaque contrainte sont mises à jour et utilisées pour modifier les coûts des *SPs*. Si la valeur optimale de chacun des *SPs* est non négative, alors la solution courante du *PMR* est optimale pour le *PM*. Dans le cas contraire, c’est-à-dire si au moins un *SP* a une valeur optimale négative, une ou plusieurs variables correspondant aux chemins obtenus sont ajoutées au *PMR*. Ce processus est répété de manière itérative jusqu’à ce qu’aucune colonne à coût réduit négatif ne puisse être générée, ce qui garantit l’optimalité de la solution du *PM*.

Algorithme 1 : Algorithme de génération de colonnes

```

1 Initialiser le PMR avec un sous-ensemble de colonnes
2 continue := vrai
3 tant que continue faire
4   Résoudre le PMR et obtenir les valeurs des variables duales
5   Actualiser les coûts ajustés des variables des SPs à l’aide des valeurs duales
   associées aux contraintes pertinentes
6   Résoudre les SPs pour identifier des variables du PM de coûts réduits négatifs
7   si toutes les solutions des SPs ont un coût non-négatif alors
8     La solution du PMR est optimale pour le PM
9     continue := faux
10  sinon
11    Ajouter les variables de coût réduit négatif à l’ensemble des variables
    considérées dans le PMR

```

L’algorithme de *GC* permet ainsi de résoudre efficacement des problèmes linéaires de très grande taille en exploitant leur structure particulière. Il est particulièrement bien adapté pour la résolution des relaxations linéaires de problèmes d’horaires et de routage, où les décisions peuvent être naturellement représentées sous forme de chemins, de tournées ou de séquences temporelles.

Sous-problèmes et calcul des coûts réduits

Les *SPs* constituent le cœur algorithmique de la *GC*. Leur objectif est d’identifier, parmi l’ensemble des colonnes potentiellement admissibles, celles qui peuvent améliorer la solution du *PMR*. De manière générale, le coût réduit d’une colonne est défini comme la différence entre son coût réel et la somme pondérée des valeurs duales associées aux contraintes auxquelles elle contribue. Une colonne présentant un coût réduit négatif indique qu’en l’introduisant dans le *PMR*, il est possible de diminuer la valeur de la fonction objectif du nouveau *PMR*. Ainsi, la recherche de colonnes à coût réduit négatif constitue le critère fondamental guidant

la génération de nouvelles variables.

La structure du *SP* dépend étroitement de la nature du problème étudié. Dans de nombreux problèmes de routage et d'horaire, les *SPs* prennent la forme d'un problème de plus court chemin, parfois enrichi de contraintes supplémentaires, telles que des contraintes de ressources, de temps ou de capacité. Ces contraintes sont intégrées directement dans le processus de génération des colonnes, ce qui permet de ne produire que des solutions réalisables au regard du problème original.

2.3.2 Applications de la génération de colonnes au routage aérien

Les problèmes de routage d'avions sont généralement formulés comme des programmes linéaires en nombres entiers, où les variables de décision représentent des routes d'avions admissibles respectant l'ensemble des contraintes opérationnelles et de maintenance. Pour des instances de petite taille, l'ensemble complet des routes peut être énuméré explicitement, ce qui permet de résoudre le problème à l'aide de méthodes standard de programmation linéaire, telles que l'algorithme du simplexe [41].

Pour des instances de taille réelle, impliquant des flottes hétérogènes, la dimension du modèle croît rapidement avec le nombre de vols, rendant impossible l'énumération exhaustive de l'ensemble des routes. Dans ce contexte, la méthode de *GC* est utilisée pour résoudre le programme linéaire en ne considérant qu'un sous-ensemble de variables, tout en générant dynamiquement de nouvelles variables pertinentes via les *SPs*.

Lorsque des contraintes d'intégrité doivent être respectées pour obtenir une solution entière, la *GC* est intégrée dans un algorithme *branch-and-bound*, formant ainsi la technique dite *branch-and-price*. Alternativement, elle peut être combinée avec des heuristiques de recherche de solutions entières, telles que le *diving* ou la résolution d'un *PMR* sujet à des contraintes d'intégrité, afin d'obtenir des solutions quasi-optimales rapidement pour les très grandes instances.

Dans le cadre des problèmes de routage aérien, les *SPs* sont généralement formulés comme des problèmes de plus court chemin avec contraintes de ressources. La structure du réseau utilisé dans chaque *SP* varie selon le coût de la route ainsi que selon les contraintes prises en considération. Selon Saltzman et Stern [42] les réseaux peuvent être classés en trois grandes catégories : les réseaux ville-jour, les réseaux temps-espace et les réseaux de connexions.

Un réseau ville-jour est une représentation bidimensionnelle dans laquelle les nœuds sont définis selon deux axes distincts : l'axe spatial, correspondant aux aéroports, et l'axe temporel discret, correspondant aux jours de l'horizon de planification. Chaque nœud représente ainsi

la présence d'un avion dans un aéroport donné à un jour donné. Les arcs du réseau modélisent les segments de vol, reliant un nœud associé à l'aéroport de départ à un jour donné à un nœud associé à l'aéroport d'arrivée au jour correspondant à l'exécution du vol. Toutefois, ce type de représentation ne permet pas de prendre en compte explicitement les horaires de départ et d'arrivée, ce qui constitue une limitation majeure. Cette insuffisance est levée par les réseaux espace-temps, dans lesquels une ligne temporelle est associée à chaque aéroport sur l'horizon de planification. Les nœuds positionnés sur ces lignes temporelles représentent les événements de départ et d'arrivée des avions, reliés par des arcs correspondant aux segments de vol. Bien que les réseaux espace-temps présentent l'avantage de comporter un nombre réduit d'arcs, ils rendent l'identification et le suivi des routes complets des avions plus complexes, dans la mesure où les connexions entre vols successifs ne sont pas explicitement représentées.

À l'inverse, les réseaux de connexions modélisent directement les segments de vol comme des nœuds, tandis que les arcs représentent les connexions réalisables entre vols consécutifs. Cette structure facilite la construction et l'analyse des routes d'avion, ce qui en fait une représentation particulièrement adaptée aux problèmes de routage avec des contraintes opérationnelles et de maintenance. Néanmoins, peu de travaux décrivent en détail la construction effective de ces réseaux, et encore moins les méthodes visant à en réduire la taille afin d'améliorer l'efficacité computationnelle.

Dans le présent travail, la représentation sous forme de réseau de connexions se révèle particulièrement adaptée puisqu'il y a des coûts de connexion spécifiques à chaque connexion et à chaque avion.

2.3.3 Intégration des contraintes de maintenance dans les sous-problèmes

L'intégration des contraintes de maintenance constitue l'un des principaux défis dans l'application de la *GC* au routage d'avions. Contrairement aux problèmes de routage classiques, les routes doivent non seulement couvrir tous les vols, mais aussi respecter des exigences réglementaires strictes, comme les limites sur les indicateurs de maintenance et la réalisation de certaines opérations dans des stations autorisées. Pour maintenir l'efficacité de l'algorithme, ces contraintes sont prises en compte au niveau des *SPs* lors de la génération des routes.

Dans le réseau de connexions, des fenêtres de ressources sont associées à chaque nœud. Les ressources considérées peuvent inclure, entre autres, le temps de vol cumulé, le nombre de cycles, le nombre de jours calendaires cumulés depuis la dernière maintenance, ainsi que d'autres contraintes opérationnelles et de maintenance pertinentes. L'objectif du *SP* consiste à identifier un chemin de coût réduit minimal, qui est la colonne à transmettre au *PMR*, en intégrant les valeurs duales issues du *PM*.

D'un point de vue algorithmique, les *SPs* associés au routage d'avions avec maintenance sont fréquemment modélisés comme un problème de plus court chemin avec contrainte des ressources (*Shortest Path Problem with Resource Constraints, SPPRC*). Afin de résoudre le *SPPRC*, les *SPs* sont généralement résolus à l'aide d'un algorithme d'étiquetage (*labeling algorithm*) fondé sur la programmation dynamique. Un algorithme d'étiquetage pour le *SPPRC* est introduit par Desrochers et Soumis en 1988 [43]. Il repose sur une modélisation du réseau composée de nœuds et d'arcs, où chaque chemin partiel amenant à un nœud à partir de tous les nœuds prédécesseurs possibles est représenté par une étiquette. Une étiquette pour un nœud donné regroupe l'ensemble des informations nécessaires à la description du chemin parcouru : le coût cumulé et la consommation des ressources (par exemple, capacité ou temps), ainsi que le dernier nœud visité. À partir de cette structure, l'algorithme procède par extension progressive des étiquettes, en créant de nouvelles étiquettes pour les arcs sortants tant que les contraintes de ressources ne sont pas violées. Afin de limiter cet ensemble, les règles de dominance introduites aussi par Desrochers et Soumis [43], qui permettent d'écarter les étiquettes dominées, c'est-à-dire celles qui ne sont pas meilleures en termes de coût et de consommation de ressources, peuvent être appliquées. Ainsi, seules les étiquettes non dominées sont sauvegardées. Irnich et Desaulniers [44] proposent une description détaillée des règles de dominance et d'un algorithme de dominance appliqué au problème de routage de véhicules. Cet algorithme sera expliqué plus en détail et son application au *OAMRP* dans le chapitre 4. Dans la littérature, la performance globale d'un algorithme de *GC* dépend fortement de l'efficacité du *SP*, tant en termes de qualité des colonnes générées que de temps de calcul. Par conséquent, une attention particulière est accordée à la conception du *SP*.

Cette méthodologie d'algorithme de *GC* s'avère adaptée au *OAMRP*, qui présente une complexité combinatoire élevée due aux milliers de routes possibles combinant les vols et les contrôles de maintenance pour une flotte hétérogène. Face à l'impossibilité pratique d'énumérer toutes les solutions, le mécanisme de *GC* résout en alternance le *PMR* et des *SPs* de type *SPPRC*, cherchant ainsi les meilleures routes pour chaque avion tout en respectant les contraintes de maintenance réglementaires et les capacités limitées des stations. Cette approche offre un cadre computationnel robuste pour traiter des instances à grande échelle.

Recherche d'une solution entière

L'algorithme de *GC* converge lorsqu'aucun *SP* ne permet d'identifier de colonne à coût réduit négatif. À ce stade, la solution du *PMR* correspond à la solution optimale de la relaxation linéaire du problème original. Les solutions retournées par l'algorithme de *GC* sont généralement fractionnaires. Dans ce cas, un algorithme de branchement est utilisé

afin d'obtenir une solution entière. La méthode *branch-and-price* permet de résoudre un problème linéaire en nombres entiers en faisant appel à un arbre de branchement comme dans l'algorithme de *branch-and-bound* en résolvant à chaque nœud de l'arbre la relaxation linéaire par *GC* [38]. Un branchement exact est cependant très coûteux en temps de calcul. Ainsi, une exploration partielle de l'arbre de recherche, souvent une seule descente en profondeur dans une branche de l'arbre (*pure diving*), est couramment employée. Ce type de méthodes heuristiques repose sur la fixation progressive de variables sélectionnées à partir des solutions fractionnaires du *PM*, sans recourir à un schéma complet de *branch-and-price* [45] [46].

CHAPITRE 3 DESCRIPTION ET MODÉLISATION DU PROBLÈME

Dans ce chapitre, la modélisation d'*OAMRP* proposée est détaillée. Une description du problème est présentée dans la section 3.1. Pour structurer la description du problème, nous introduisons d'abord ses éléments constitutifs. Nous décrivons ensuite les contraintes de maintenance imposées aux avions. Enfin, nous formulons les pénalités associées aux dates d'exécution des maintenances. La formulation mathématique du problème est présentée dans la section 3.2.

3.1 Description du problème

Dans ce travail, nous nous intéressons à l'optimisation du routage des avions sous contraintes de maintenance, dans un contexte où l'affectation de la flotte est déjà fixée et où un sous-ensemble des connexions est fourni pour chaque avion. Les horaires de vols sont définis sur un horizon de deux semaines. Afin de garantir la faisabilité des opérations de maintenance requises, chaque avion doit être affecté à la route la plus appropriée. L'objectif du routage est ainsi d'intégrer les opérations de maintenance tout en respectant l'ensemble des contraintes opérationnelles.

3.1.1 Les éléments du problème

Dans cette section, nous présentons une description des éléments constitutifs d'*OAMRP*.

- Aéroport : un aéroport représente l'origine ou la destination d'un vol. Chaque aéroport est associé à un identifiant unique ;
- Flotte d'avions : un ensemble d'avions soumis aux mêmes exigences de maintenance et pouvant être entretenus dans les mêmes lieux. Une flotte peut être subdivisée en plusieurs sous-flottes (différant, par exemple, par leur configuration de sièges) ;
- Maintenance pré-assignée : une maintenance pré-assignée pour un avion donné est caractérisée par un identifiant unique, un aéroport principal où elle doit être effectuée, ainsi qu'un ensemble d'aéroports alternatifs vers lesquels elle peut éventuellement être réalisée. La maintenance pré-assignée est préprogrammée et ne peut être modifiée ;
- Contrôle de maintenance : un contrôle de maintenance pour un avion donné est défini par un identifiant unique, un aéroport principal où il doit être réalisé, ainsi qu'un ensemble de lieux alternatifs où il peut éventuellement être déplacé. Chaque contrôle de maintenance appartient à un type précis ;

- Type de contrôle de maintenance : les avions doivent être entretenus régulièrement. Ils sont soumis à différents types de contrôle de maintenance, chacun étant régi par des règles distinctes : temps de vol maximal, nombre maximal de cycles et/ou nombre maximal de jours depuis la dernière maintenance ;
- Lieu de maintenance : certains aéroports disposent de la capacité de réaliser les maintenances spécifiques à certains types et pour certaines sous-flottes. Pour ces aéroports, une capacité maximale, définissant le nombre d'avions pouvant être entretenus sur un intervalle de temps donné, doit être prise en compte. Ainsi, un lieu de maintenance est associé à un ensemble de quarts de travail disjoints pour chaque type de maintenance. Un quart de travail correspond à un poste de travail et est associé à une capacité donnée en fonction du type de maintenance et de la flotte concernée ;
- Vol : un vol est identifié de manière unique par un identifiant spécifique. Il est caractérisé par son heure de départ, son heure d'arrivée, son aéroport de départ (origine) et son aéroport d'arrivée (destination). D'autres attributs incluent la durée du vol et le nombre de cycles, représentant le nombre de cycles effectués. Chaque vol est en outre associé à un coût d'annulation ;
- Connexion : la succession de deux vols consécutifs dans une même route définit une connexion. Une connexion est valide si l'aéroport d'arrivée du premier vol est identique à l'aéroport de départ du second, et si l'intervalle entre l'arrivée et le départ est suffisant pour préparer l'avion au vol suivant, incluant éventuellement la durée d'une maintenance planifiée. La liste des connexions possibles est donnée en entrée, accompagnée d'un coût de connexion dépendant de l'avion. La notion de connexion est étendue en considérant soit le début de l'horizon comme premier vol, soit la fin de l'horizon comme second vol ;
- Route : une route d'avion correspond à la séquence de vols affectés à un appareil donné sur l'horizon de planification, pouvant inclure les maintenances dans des lieux spécifiques.

L'objectif d'*OAMRP* est de minimiser le coût total des routes d'avion tout en respectant les règles de maintenance propres à chaque appareil, sous les contraintes de couverture de l'ensemble des vols et de capacité des lieux de maintenance. Les coûts pris en compte comprennent les coûts d'annulation de vols, les coûts de connexion ainsi que les pénalités liées aux objectifs de maintenance. Ainsi, la solution obtenue fournit une route (incluant des vols et des maintenances) pour chaque avion disponible. Pour résumer, l'objectif de ce problème est de minimiser les coûts suivants :

1. Coûts de connexion entre les vols d'une route ;
2. Coûts d'annulation de vols ;

3. Coûts liés aux écarts par rapport aux objectifs de maintenance.

Ces coûts sont agrégés dans une fonction objectif unique sous diverses contraintes opérationnelles et hypothèses afin de garantir la faisabilité et le respect des exigences opérationnelles et réglementaires :

- Tous les vols doivent être couverts par un avion valide autant que possible ;
- Toutes les routes doivent être composées de connexions valides ;
- Les conditions initiales de chaque avion doivent être respectées, notamment la date et le lieu de disponibilité, ainsi que l'état des différents indicateurs pour la prochaine maintenance, le cas échéant ;
- Chaque avion doit respecter un temps minimal au sol lors de la connexion entre deux vols ;
- Chaque avion doit effectuer les maintenances nécessaires dans un lieu de maintenance valide et respecter les bornes spécifiques sur le temps de vol, le nombre de cycles et/ou le nombre de jours depuis la dernière maintenance ;
- La capacité des sites de maintenance doit être respectée ;
- Les opérations de maintenance pré-assignées, le cas échéant, doivent être exécutées pour chaque avion.

Connexions de vols et gestion du temps minimal au sol

Considérons K l'ensemble des avions disponibles au début de l'horizon de planification et V l'ensemble des vols à planifier. Nous définissons également les paramètres suivants :

- τ_i^{arr} : heure d'arrivée du vol $i \in V$;
- τ_j^{dep} : heure de départ du vol $j \in V$;
- g_{ijk} : temps minimal requis au sol excluant le temps de maintenance entre les vols $i \in V$ et $j \in V$ pour l'avion $k \in K$, excluant toute durée de maintenance ;
- c_{ijk} : coût de la connexion entre les vols $i \in V$ et $j \in V$ pour l'avion $k \in K$. C_{ijk} se généralise aux cas où i représente le début de l'horizon et j représente la fin de l'horizon.

Une connexion (i, j) pour un avion k est dite réalisable lorsque l'aéroport d'arrivée du vol i est le même que l'aéroport d'arrivée du vol j et que le temps minimal au sol est respecté, c'est-à-dire lorsque

$$\tau_j^{dep} - \tau_i^{arr} \geq g_{ijk}.$$

Afin d'introduire une certaine flexibilité dans la construction des routes, une marge de tolérance Δ^G est autorisée. Ainsi, une connexion peut être considérée malgré une violation du

temps minimal au sol lorsque

$$\tau_j^{dep} - \tau_i^{arr} \geq g_{ijk} - \Delta^G.$$

Lorsque $\tau_j^{dep} - \tau_i^{arr} < g_{ijk}$, une violation du temps minimal au sol est observée. Le temps de violation est alors donné par

$$g_{ijk}^{viol} = g_{ijk} - (\tau_j^{dep} - \tau_i^{arr}).$$

Cette violation entraîne l'application de deux types de pénalités : une pénalité fixe c_{fix}^G ainsi qu'une pénalité proportionnelle au temps de violation en minutes c_{min}^G .

Le coût total de pénalité associé à la connexion (i, j) est donc calculé comme suit :

$$c_{ijk}^G = \begin{cases} c_{fix}^G + c_{min}^G \cdot g_{ijk}^{viol} & \text{si } g_{ijk} - \Delta^G \leq \tau_j^{dep} - \tau_i^{arr} < g_{ijk}, \\ 0 & \text{sinon .} \end{cases}$$

Options de planification de maintenance

Considérons W_{lm} , l'ensemble de quarts de travail dans le lieu de maintenance $l \in L$ pour le type de maintenance $m \in M$. L'horizon de planification, d'une durée de 14 jours, est discrétisé en périodes correspondant aux quarts de travail. Chaque journée est divisée en deux quarts consécutifs de durée identique (720 minutes), représentant les quarts de jour et de soir. Cette structure est reproduite sur l'ensemble de l'horizon, qui totalise 20 160 minutes. Pour chaque station de maintenance, une capacité maximale L_{lmw} est définie pour chaque lieu de maintenance l , de type de maintenance m et de quart de travail w . Cette capacité correspond au nombre maximal d'avions pouvant être entretenus simultanément durant la période considérée.

La figure 3.1 illustre la structure temporelle de l'horizon de planification. Chaque ligne représente un aéroport où le temps est discrétisé en journées ($J0, J1, J2$) divisées en deux quarts : le Quart 1 et le Quart 2. Pour chaque connexion entre deux vols, les crochets noirs délimitent l'intervalle durant lequel une opération de maintenance peut être planifiée. Cet intervalle est obtenu en réservant un temps minimal au sol requis (g_{ijk}), réparti symétriquement avec une première moitié ($0.5 g_{ijk}$) après l'arrivée du vol i et une seconde moitié avant le départ du vol j . Cette marge assure le respect du temps au sol minimal. La durée d'une opération de maintenance n'est pas nécessairement bornée par celle d'un quart de travail, aussi elle peut débuter dans un quart et se terminer dans le suivant, voire s'étendre sur plusieurs quarts consécutifs. Dans ce cas, une maintenance contribue à un quart de travail si sa durée

programmée dans ce quart de travail dépasse une durée de ξ^{min} en minutes. En supposant que l'intervalle à l'Aéroport B entre les Vols 1 et 2 est suffisamment grand par rapport à la durée d'une maintenance à planifier, il est possible d'effectuer cette maintenance soit durant le Quart 1 uniquement, soit durant le Quart 2 uniquement ou soit durant ces quarts en supposant que la maintenance chevauche la frontière entre les deux quarts. Ces différents cas de figure sont appelés des options de planification de maintenance. Dans le modèle d'optimisation, il est suffisant de considérer seulement les deux premières options car la troisième option est dominée puisqu'elle consomme une unité de capacité pour chacun des deux quarts.

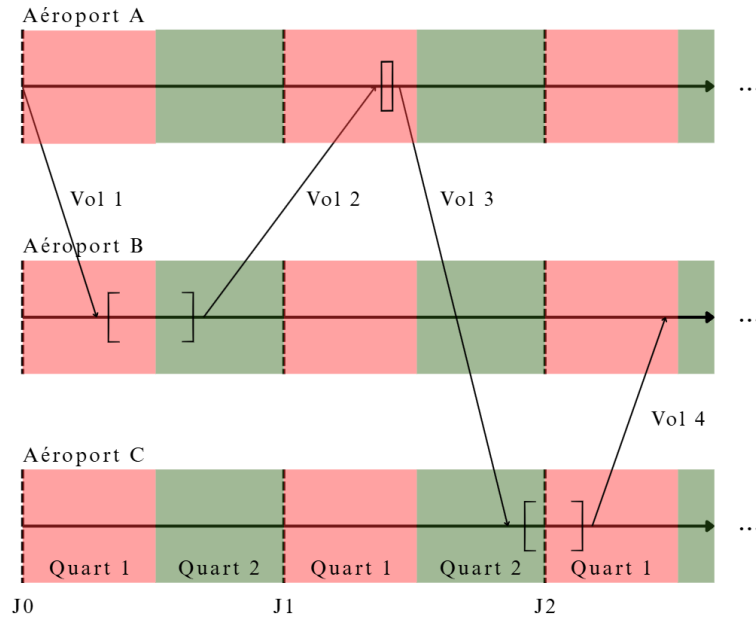


FIGURE 3.1 Planification de maintenance

Pour générer les options de planification entre deux vols consécutifs, nous proposons la procédure décrite dans l'algorithme 2, où :

- δ_m^k : durée de la maintenance $m \in M$ pour l'avion $k \in K$;
- μ_w^- et μ_w^+ : heures de début et de fin du quart de travail $w \in W_{lm}$.

On suppose que les quarts de travail dans W_{lm} sont numérotés chronologiquement de 1 à $|W_{lm}|$, c'est-à-dire $W_{lm} = \{1, 2, \dots, |W_{lm}|\}$. L'ensemble des options de planification de maintenance est noté P . Chaque option indique uniquement les quarts de travail dans lesquels la maintenance aura lieu pendant au moins ξ^{min} en unités de temps. Une telle option est représentée par la paire de quarts de travail (w^-, w^+) auxquels la maintenance contribue pour la contrainte de capacité (avec $w^- = w^+$ possible). La maintenance débute dans w^- et se termine dans w^+ .

Algorithme 2 : Génération des options de planification de maintenance

```

1  $P := \emptyset$ 
2  $h := \tau_i^{arr} + 0.5 g_{ijk}$ 
3 tant que  $h + \delta_m^k + 0.5 g_{ij,k} \leq \tau_j^{dep}$  faire
4    $w^- := \text{getSlot}(W_{lm}, h + \xi^{min})$ 
5    $w^+ := \text{getSlot}(W_{lm}, h + \delta_m^k - \xi^{min} + 1)$ 
6    $P := P \cup \{(w^-, w^+)\}$ 
7    $h := \mu_{w^-}^+ - \xi^{min} + 1$ 
8 retourner  $P$ 

```

L'algorithme débute par l'initialisation de l'ensemble des options de planification P à vide. Il fixe ensuite un instant initial h , correspondant au début potentiel de la maintenance, en ajoutant à l'heure d'arrivée du vol i la moitié du temps minimal au sol requis entre les vols i et j . Cette répartition symétrique du temps minimal au sol permet de réserver une marge avant et après la maintenance.

La condition de la boucle à la ligne 3 vérifie alors que la durée de la maintenance, augmentée de la seconde moitié du temps minimal au sol, demeure compatible avec l'heure de départ du vol j . Tant que cette condition est satisfaite, une nouvelle option de planification est générée.

Aux lignes 4 et 5, la fonction $\text{getSlot}(W, h)$ renvoie le quart de travail $w \in W$ contenant le temps h , c'est-à-dire $h \in]\mu_w^-, \mu_w^+]$. Le premier quart de travail contribuant à la contrainte de capacité est déterminé comme celui contenant l'instant $h + \xi^{min}$, ce qui garantit que la maintenance y demeure pendant au moins la durée minimale requise pour être comptabilisée. De manière analogue, le dernier quart de travail impacté est identifié comme celui contenant l'instant correspondant à la fin de la maintenance diminuée de ξ^{min} (ajusté d'une unité afin de respecter la discrétisation), assurant ainsi que la contribution au dernier quart de travail satisfait également le seuil minimal. La paire des quarts de travail ainsi obtenue est ajoutée à l'ensemble P .

Enfin, à la ligne 7, l'heure de début de la maintenance h est mise à jour pour correspondre au plus tôt au quart de travail suivant w^- , ajustée de façon à préserver une contribution minimale à ce quart de travail. Cette mise à jour permet de déplacer progressivement la maintenance dans la fenêtre admissible et d'explorer toutes les options de planification. Lorsque la condition de faisabilité n'est plus vérifiée, l'algorithme s'arrête et retourne l'ensemble des options générées.

Lorsqu'une maintenance est insérée entre deux vols i et j et que le temps au sol restant est insuffisant, soit $(\tau_j^{dep} - \tau_i^{arr}) - \delta_m^k < g_{ijk}$, il est possible de violer légèrement le temps

minimum au sol g_{ijk} (excluant la durée de la maintenance δ_m^k) entre $i \in V$ et $j \in V$ par une tolérance d'au plus Δ^G , donnant plus de flexibilité pour insérer une maintenance pendant une connexion. Le temps de violation est alors donné par

$$g_{ijk}^{Mviol} = g_{ijk} - (\tau_j^{dep} - \tau_i^{arr}) + \delta_m^k.$$

On dénote alors le coût de violation associé à la connexion (i, j) , avec une maintenance, par c_{ijk}^{GM} . Il est donc calculé comme suit :

$$c_{ijk}^{GM} = \begin{cases} c_{fix}^G + c_{min}^G \cdot g_{ijk}^{Mviol} & \text{si } g_{ijk} - \Delta^G \leq (\tau_j^{dep} - \tau_i^{arr}) - \delta_m^k < g_{ijk}, \\ 0 & \text{sinon.} \end{cases}$$

3.1.2 Contraintes de maintenance et pénalités associées

Nous considérons deux classes de maintenance : une maintenance pré-assignée, dont la date d'exécution et la durée sont connues à l'avance, et des contrôles de maintenance requis, définis par des seuils portant sur un ou plusieurs indicateurs. Un avion ne peut avoir plus d'un contrôle de maintenance requis dans l'horizon, mais plusieurs maintenances pré-assignées. Pour chaque contrôle de maintenance requis, des bornes inférieures et supérieures, ainsi qu'une valeur cible, sont spécifiées pour les critères considérés, à savoir le temps de vol, le nombre de cycles et/ou le nombre de jours écoulés depuis la dernière maintenance. Pour un avion donné, tous les critères s'appliquent ou seulement un sous-ensemble de ceux-ci lorsque certains ne peuvent pas influencer la date de la maintenance dans l'horizon de planification.

Considérons un avion k nécessitant une maintenance de type m , soumise aux trois règles de maintenance :

- fT : temps de vol cumulé depuis la dernière maintenance ;
- nC : nombre de cycles cumulés depuis la dernière maintenance ;
- nD : nombre de jours calendaires cumulés depuis la dernière maintenance.

L'horizon de planification débute à l'instant 0. Les contrôles de maintenance doivent être réalisés selon les règles suivantes :

1. Pour chaque critère $\ell \in \{fT, nC, nD\}$, il existe un intervalle faisable $[\underline{M}_{k,m}^\ell, \overline{M}_{k,m}^\ell]$ ainsi qu'une valeur cible $O_{k,m}^\ell \in [\underline{M}_{k,m}^\ell, \overline{M}_{k,m}^\ell]$. Sans perte de généralité, on peut poser $\underline{M}_{k,m}^\ell = 0$ en l'absence de borne inférieure. On suppose en revanche qu'une borne supérieure $\overline{M}_{k,m}^\ell$ existe toujours ;
2. La maintenance doit être planifiée avant qu'un critère $\ell \in \{fT, nC, nD\}$ atteigne sa borne supérieure $\overline{M}_{k,m}^\ell$ au cours de l'horizon de planification.

Des pénalités sont associées aux écarts par rapport aux bornes inférieure et supérieure pour chaque critère $\ell \in \{fT, nC, nD\}$, afin d'inciter la planification à respecter au mieux ces intervalles de tolérance.

Pénalité pour maintenance anticipée : une pénalité linéaire est appliquée lorsqu'une maintenance est effectuée avant la borne inférieure, afin de limiter la sous-utilisation de la capacité opérationnelle de l'avion.

Pénalité pour maintenance tardive : une autre pénalité linéaire est appliquée lorsqu'une maintenance est réalisée après la date, le temps de vol ou le nombre de cycles cibles, car l'avion risque de dépasser les limites légales de sécurité opérationnelle en cas d'imprévu.

Au début de chaque horizon de planification, l'état initial de chaque avion est connu en termes d'heures de vol cumulées, de cycles effectués et de jours calendaires écoulés depuis la dernière maintenance. Cet état de référence permet de calculer les valeurs cibles actualisées, $O_{k,m}^\ell$, pour chaque critère $\ell \in \{fT, nC, nD\}$, ainsi que les fenêtres d'intervention correspondantes pour chaque indicateur de maintenance $[\underline{M}_{k,m}^\ell, \overline{M}_{k,m}^\ell]$.

La figure 3.2 illustre la variation de la pénalité de maintenance en fonction du nombre de jours, du temps de vol ou du nombre de cycles.

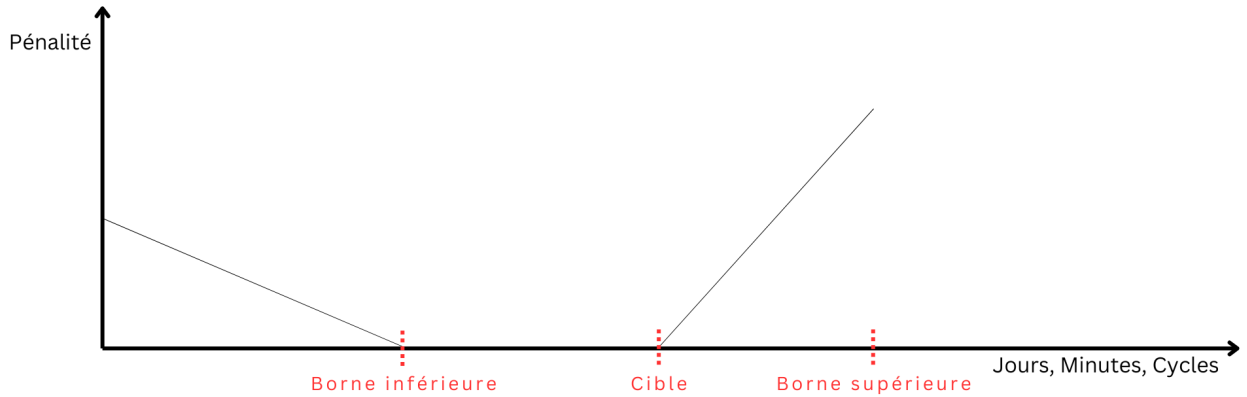


FIGURE 3.2 Variation de la pénalité en fonction des critères de maintenance

Considérons un avion k nécessitant une maintenance de type m , soumise aux trois règles de maintenance. Si des pénalités de maintenance doivent être prises en compte, nous faisons les hypothèses suivantes :

1. Une pénalité indépendante de l'avion $\gamma_m^{E,\ell}$ par unité peut être appliquée en cas de maintenance anticipée (effectuée en dessous de $\underline{M}_{k,m}^\ell$), et/ou une pénalité $\gamma_m^{L,\ell}$ par unité en cas de maintenance tardive (effectuée après la cible $O_{k,m}^\ell$ mais avant $\overline{M}_{k,m}^\ell$) ;
2. Aucune pénalité n'est appliquée lorsque la maintenance est effectuée dans l'intervalle $[\underline{M}_{k,m}^\ell, O_{k,m}^\ell]$;
3. Les pénalités pour maintenance anticipée ne s'appliquent que si une maintenance est effectivement réalisée en dessous de $\underline{M}_{k,m}^\ell$. Si la maintenance est reportée (c'est-à-dire non planifiée sur l'horizon), aucune pénalité anticipée n'est encourue, même si l'utilisation reste inférieure à $\underline{M}_{k,m}^\ell$.

Plus précisément, la pénalité C_{klm}^P à appliquer pour une maintenance de type $m \in M$ à un avion $k \in K$ selon le critère $\ell \in \{fT, nC, nD\}$ est calculée comme suit :

$$C_{klm}^P = \begin{cases} \gamma_m^{E,\ell} \cdot (\underline{M}_{k,m}^\ell - \Delta_{k,m}^\ell) & \text{si } \Delta_{k,m}^\ell < \underline{M}_{k,m}^\ell, \\ \gamma_m^{L,\ell} \cdot (\Delta_{k,m}^\ell - \overline{M}_{k,m}^\ell) & \text{si } \Delta_{k,m}^\ell > \overline{M}_{k,m}^\ell, \\ 0 & \text{sinon.} \end{cases} \quad (3.1)$$

où $\Delta_{k,m}^\ell$ est la valeur cumulative du critère ℓ au début de la maintenance. Ainsi, la pénalité totale pour cette maintenance si elle est sujette aux trois critères est égale à $\sum_{\ell \in \{fT, nC, nD\}} C_{klm}^P$.

3.2 Formulation mathématique

Afin de formaliser l'*OAMRP*, nous adoptons une formulation de type partitionnement d'ensemble, à laquelle nous intégrons des contraintes supplémentaires permettant de représenter les exigences de maintenance ainsi que de garantir la faisabilité opérationnelle des routes générés pour chaque avion. Pour une formulation de base de type partitionnement d'ensemble, on considère un ensemble de vols qui doivent être exécutés par des avions. Notons K l'ensemble des avions considérés dont les variables sont associées à des routes d'avion.

Pour formaliser la représentation du modèle, nous définissons d'abord la notation avant de présenter la formulation détaillée. Dans les tableaux 3.1 - 3.3, nous listons les ensembles, les paramètres et les variables de décision utilisés dans le modèle.

TABLEAU 3.1 Ensembles

Ensemble	Description
K	Ensemble des avions.
R_k	Ensemble des routes réalisables pour l'avion k .
V	Ensemble des vols.
A	Ensemble des aéroports.
L	Ensemble des lieux de maintenance.
M_l	Ensemble des types de maintenance pouvant être réalisés au lieu de maintenance l .
$W_{l,m}$	Ensemble des quarts de travail horaires dans le lieu de maintenance l pour le type de maintenance m .

TABLEAU 3.2 Paramètres

Paramètre	Description
c_{rk}	Coût de la route $r \in R_k$ pour l'avion $k \in K$.
c_i	Coût d'annulation du vol $i \in V$.
α_{ri}	1 si la route $r \in R_k$ couvre le vol $i \in V$, 0 sinon.
β_{rlmw}	1 si la route $r \in R_k$ inclut une maintenance de type m au lieu $l \in R_k$ dans le quart de travail $w \in L_{l,m}$, 0 sinon.
L_{lmw}	Nombre maximal d'avions pouvant recevoir une maintenance de type $m \in M_l$ dans le quart de travail $w \in L_{l,m}$ au lieu $l \in L$.

TABLEAU 3.3 Variables de décision

Variable	Description
$x_{rk} \in \{0, 1\}$	1 si la route r est assignée à l'avion k , 0 sinon.
$s_i \in \{0, 1\}$	1 si le vol i est annulé, 0 sinon.

Sur la base de ces notations, le modèle mathématique du PM peut être formulé comme suit :

$$\min \quad \sum_{k \in K} \sum_{r \in R_k} c_{rk} x_{rk} + \sum_{i \in V} c_i s_i \quad (3.2)$$

sujet à

$$\sum_{r \in R_k} x_{rk} = 1, \quad \forall k \in K, \quad (3.3)$$

$$\sum_{k \in K} \sum_{r \in R_k} \alpha_{ri} x_{rk} + s_i = 1, \quad \forall i \in V, \quad (3.4)$$

$$\sum_{k \in K} \sum_{r \in R_k} \beta_{rlmw} x_{rk} \leq L_{lmw}, \quad \forall l \in L, m \in M_l, w \in W_{l,m}, \quad (3.5)$$

$$x_{rk} \in \{0, 1\}, \quad \forall k \in K, r \in R_k, \quad (3.6)$$

$$s_i \in \{0, 1\}, \quad \forall i \in V. \quad (3.7)$$

La fonction objectif (3.2) vise à minimiser le coût total des routes sélectionnées ainsi que l'ensemble des pénalités liées à l'annulation des vols. Les contraintes (3.3) garantissent que chaque avion est affecté à une seule route, tandis que les contraintes de couverture des vols (3.4) imposent que chaque vol soit effectué par un avion ou annulé. Ensuite, les contraintes (3.5) assurent que la capacité de chaque quart de travail horaire dans les lieux de maintenance est respectée. Enfin, les contraintes (3.6) - (3.7) limitent le domaine des variables de décision.

CHAPITRE 4 MÉTHODE DE RÉOLUTION

Dans ce chapitre, nous abordons la résolution du modèle (3.2) - (3.7). Tout d'abord, nous décrivons à haut niveau l'heuristique de *GC* proposée dans la section 4.1. Ensuite, nous présentons le *PMR* dans la section 4.2 et les *SPs* dans la section 4.3. Puis, nous discutons les stratégies de branchement utilisées pour trouver une solution entière dans la section 4.4 avant de proposer les techniques d'accélération dans la section 4.5.

4.1 Résolution par une heuristique de *branch-and-price*

Le modèle (3.2) - (3.7) comporte un très grand nombre de variables binaires. Sa relaxation linéaire peut être résolue par un algorithme de *GC* qui ne nécessite pas de connaître toutes les variables (colonnes) à l'avance. Néanmoins, la solution optimale obtenue pour la relaxation linéaire peut être fractionnaire. Une telle solution n'est pas admissible d'un point de vue opérationnel, puisqu'un avion ne peut exécuter qu'une seule route complète. Afin de pallier cette difficulté, la *GC* est intégrée dans un schéma de *branch-and-bound*, communément appelé *branch-and-price*. Nous adoptons une approche heuristique par descente en profondeur, dans laquelle une seule branche de l'arbre est explorée.

À chaque itération, une méthode de *GC* résout d'abord un *PMR*, soit la relaxation linéaire restreinte à un sous-ensemble des variables, et ensuite, des *SPs* qui permettent d'identifier des variables de coût réduit négatif à ajouter au *PMR*. Dans notre cas, il y a un *SP* par avion qui correspond à un problème de plus court chemin avec contraintes de ressource et qui permet de générer des routes pour cet avion. Ce processus itératif se poursuit jusqu'à ce qu'aucune colonne (route) améliorante ne puisse être générée, garantissant ainsi l'optimalité de la relaxation linéaire avant l'exploration éventuelle des nœuds de branchement. Cette approche permet de résoudre efficacement des instances de grande taille caractérisées par un nombre exponentiel de variables.

Afin de détailler le fonctionnement de cette approche, nous présentons dans un premier temps, la relaxation linéaire du *PMR*, dont la résolution fournit les variables duales nécessaires au calcul des coûts réduits des routes. Nous décrivons ensuite la structure des graphes associés aux *SPs*, qui modélisent les routes réalisables intégrant les contraintes opérationnelles et de maintenance.

4.2 Problème maître restreint

La méthode de *GC* résout en alternance un *PMR* et un ou plusieurs *SPs*. Pour l'*OAMRP*, le *PMR* à l'itération t est la relaxation linéaire du modèle (3.2) - (3.7) restreint à un sous-ensemble des variables R_k^t . Il se formule comme suit :

$$\min \sum_{k \in K} \sum_{r \in R_k^t} c_{rk} x_{rk} + \sum_{i \in V} c_i s_i \quad (4.1)$$

sujet à

$$\sum_{r \in R_k^t} x_{rk} = 1, \quad \forall k \in K \quad (4.2)$$

$$\sum_{k \in K} \sum_{r \in R_k^t} \alpha_{ri} x_{rk} + s_i = 1, \quad \forall i \in V \quad (4.3)$$

$$\sum_{k \in K} \sum_{r \in R_k^t} \beta_{rlmw} x_{rk} \leq L_{lmw}, \quad \forall l \in L, m \in M_l, w \in W_{l,m} \quad (4.4)$$

$$x_{rk} \geq 0, \quad \forall k \in K, r \in R_k^t \quad (4.5)$$

$$s_i \geq 0, \quad \forall i \in V. \quad (4.6)$$

À la première itération de la *GC*, le *PMR* ne contient aucune route réalisable. Afin d'assurer sa faisabilité, une base initiale artificielle est introduite. Celle-ci comprend des variables de non-couverture des vols s_i , ainsi que des variables artificielles, associées à un coût très élevé, permettant de satisfaire les contraintes d'affectation des avions. Ces pénalités rendent ces variables fortement désavantageuses. Ainsi, lors des itérations suivantes, l'algorithme génère de nouvelles colonnes (routes valides) présentant des coûts réduits négatifs, ce qui a pour effet d'éliminer naturellement les variables artificielles de la solution courante.

La solution du *PMR* à l'itération t est optimale si aucune route à coût réduit négatif n'est trouvée. La résolution du *SP* consiste alors à trouver des colonnes (des variables) x_{rk} de coûts réduits négatifs $\bar{c}_{rk}$, ce coût réduit étant calculé à partir des variables duales $(\sigma_k)_{k \in K}$, $(\pi_i)_{i \in V}$ et $(\xi_{lmw})_{l \in L, m \in M_l, w \in W_{lm}}$ associées respectivement aux contraintes (4.2), (4.3) et (4.4) dans le *PMR*.

4.3 Sous-problèmes

Dans l'algorithme *GC*, les *SPs* sont responsables de la génération de routes d'avions à ajouter au *PMR*. Une route est ajoutée si son coût réduit est négatif. Étant donné que les avions

appartiennent à différentes sous-flottes et présentent des conditions initiales distinctes, il existe un SP par avion.

4.3.1 Réseaux des sous-problèmes

Le SP pour un avion $k \in K$ est formulé comme un $SPPRC$ sur un réseau de connexions acyclique $\mathcal{G}^k = (\mathcal{N}^k, \mathcal{A}^k)$, avec $\mathcal{N}^k$ l'ensemble des nœuds et $\mathcal{A}^k$ l'ensemble des arcs. Le SP intègre simultanément les contraintes opérationnelles et de maintenance et détermine à la fois la séquence des vols à effectuer et le moment ainsi que le lieu où l'avion associé doit subir une opération de maintenance. La représentation sous forme de réseau de connexions se révèle particulièrement adaptée. Les nœuds du réseau correspondent aux vols, auxquels s'ajoutent deux nœuds fictifs représentant respectivement le début et la fin de l'horizon de planification. Les arcs définissent explicitement les connexions entre vols, ce qui permet d'interdire certaines connexions jugées irréalisables par la compagnie aérienne et d'associer un coût aux connexions afin de refléter l'intérêt de faire suivre deux vols consécutifs.

Le réseau $\mathcal{G}^k$ comprend les nœuds suivants :

- Un nœud *source* o^k correspondant à la position initiale et à l'horaire initial de l'avion k ;
- Pour chaque vol $i \in V$, un nœud *vol* i qui représente ce vol ;
- Un nœud *puits* d^k qui indique la fin de l'horizon de planification.

Le réseau $\mathcal{G}^k$ inclut les arcs sans maintenance suivants :

- Pour chaque vol $i \in V$ tel que la connexion entre o^k et i est réalisable, il y a un arc (o^k, i) de *début de route sans maintenance* ayant un coût $c_{o^k i k} + c_{o^k i k}^G$;
- Pour chaque paire de vols $i, j \in V$ telle que la connexion entre i et j est réalisable, il y a un arc (i, j) de *connexion sans maintenance* ayant un coût $c_{ijk} + c_{ijk}^G$;
- Pour chaque vol $i \in V$, il y a un arc (i, d^k) de *fin de route sans maintenance* ayant un coût $c_{id^k k}$.

Lorsque l'avion k est sujet à un contrôle de maintenance, le réseau $\mathcal{G}^k$ inclut aussi des arcs avec un contrôle de maintenance requis :

- Pour chaque vol $i \in V$ tel que la connexion entre o^k et i est réalisable et chaque option de planification de maintenance durant cette connexion, il y a un arc (o^k, i) de *début de route avec un contrôle de maintenance requis* ayant un coût $c_{o^k i k} + c_{o^k i k}^{GM}$;
- Pour chaque paire de vols $i, j \in V$ telle que la connexion entre i et j est réalisable et chaque option de planification de maintenance durant cette connexion, il y a un arc (i, j) de *connexion avec un contrôle de maintenance requis* ayant un coût $c_{ijk} + c_{ijk}^{GM}$;
- Pour chaque vol $i \in V$ et chaque option de planification de maintenance après le vol

i , il y a un arc (i, d^k) de *fin de route avec un contrôle de maintenance requis* ayant un coût $c_{id^k k}$.

Il peut donc y avoir plusieurs arcs qui relient la même paire de noeuds : un arc sans maintenance et des arcs avec maintenance selon le nombre d'options réalisables.

Finalement, si des maintenances pré-assignées sont prévues pour l'avion k , le réseau $\mathcal{G}^k$ contient les arcs avec maintenance pré-assignée suivants :

- Pour chaque vol $i \in V$ tel que la connexion entre o^k et i est réalisable et chaque maintenance pré-assignée qui peut être réalisée durant cette connexion, il y a un arc (o^k, i) de *début de route avec maintenance pré-assignée* ayant un coût $c_{o^k i k} + c_{o^k i k}^{GM}$;
- Pour chaque paire de vols $i, j \in V$ telle que la connexion entre i et j est réalisable et chaque maintenance pré-assignée qui peut être réalisée durant cette connexion, il y a un arc (i, j) de *connexion avec maintenance pré-assignée* ayant un coût $c_{ij k} + c_{ij k}^{GM}$;
- Pour chaque vol $i \in V$ et chaque maintenance pré-assignée qui peut être réalisée après le vol i , il y a un arc (i, d^k) de *fin de route avec maintenance pré-assignée* ayant un coût $c_{id^k k}$.

Tous les arcs en conflit avec une maintenance pré-assignée qui ne la contient pas sont supprimés, garantissant ainsi le respect de la maintenance pré-assignée.

Soit A_{rk} le sous-ensemble des arcs du chemin dans $\mathcal{G}^k$ représentant la route r . Afin de garantir que le coût de tout chemin faisable du nœud *source* o^k au nœud *puits* d^k dans $\mathcal{G}^k$ soit égal au coût réduit de la variable correspondante, les coûts ajustés des arcs peuvent être définis comme suit :

$$\bar{c}_{ak} = \begin{cases} c_{ak} - \sigma_k - \sum_{l \in L} \sum_{m \in M_l} \sum_{w \in W_{lm}} \beta_{lmw}^a \xi_{lmw}^t & \text{si l'arc } a = (o^k, i) \text{ avec } i \in V, \\ c_{ak} - \pi_i - \sum_{l \in L} \sum_{m \in M_l} \sum_{w \in W_{lm}} \beta_{lmw}^a \xi_{lmw}^t & \text{si l'arc } a = (i, j) \text{ avec } i \in V \text{ et } j \in V. \end{cases} \quad (4.7)$$

où $\bar{c}_{ak}$ est le coût ajusté de l'arc a , et β_{lmw}^a est un paramètre binaire égal à 1 si l'arc a contient une maintenance de type m à la station l couvrant le créneau temporel w .

Le coût réduit $\bar{c}_{rk}$ d'une variable x_{rk} est donné par :

$$\bar{c}_{rk} = c_{rk} - \sigma_k - \sum_{i \in V} \alpha_{ri} \pi_i - \sum_{l \in L} \sum_{m \in M_l} \sum_{w \in W_{lm}} \beta_{rlmw} \xi_{lmw} \quad (4.8)$$

$$= \sum_{a \in A_{rk}} \bar{c}_{ak} + C_{rk}^P, \quad (4.9)$$

où C_{rk}^P est égal aux pénalités de maintenance qui s'appliquent à cette route. Comme ces

pénalités dépendent possiblement des nombres de jours, de cycles et d'heures volées depuis la dernière maintenance, elles ne peuvent être associées directement à un arc comprenant cette maintenance. Elles seront calculées en cours de création de la route.

Un exemple simple de réseau $\mathcal{G}^k$ est illustré à la figure 4.1. Les lignes pleines représentent les arcs de connexion entre les vols, ainsi que les arcs de début et de fin de route. La position initiale de l'avion est l'aéroport représenté par le nœud *source* o^k , à partir duquel les vols i_1 , i_2 et i_3 peuvent être effectués en début d'horizon de planification. Ces vols sont compatibles avec les contraintes opérationnelles, car ils partent de l'aéroport initial de l'avion et débutent après son heure de disponibilité. Ensuite, certaines paires de vols peuvent être réalisées consécutivement par ce même avion (i_2, i_1) ou (i_1, i_3), les temps de connexion entre ces vols respectant les contraintes opérationnelles.

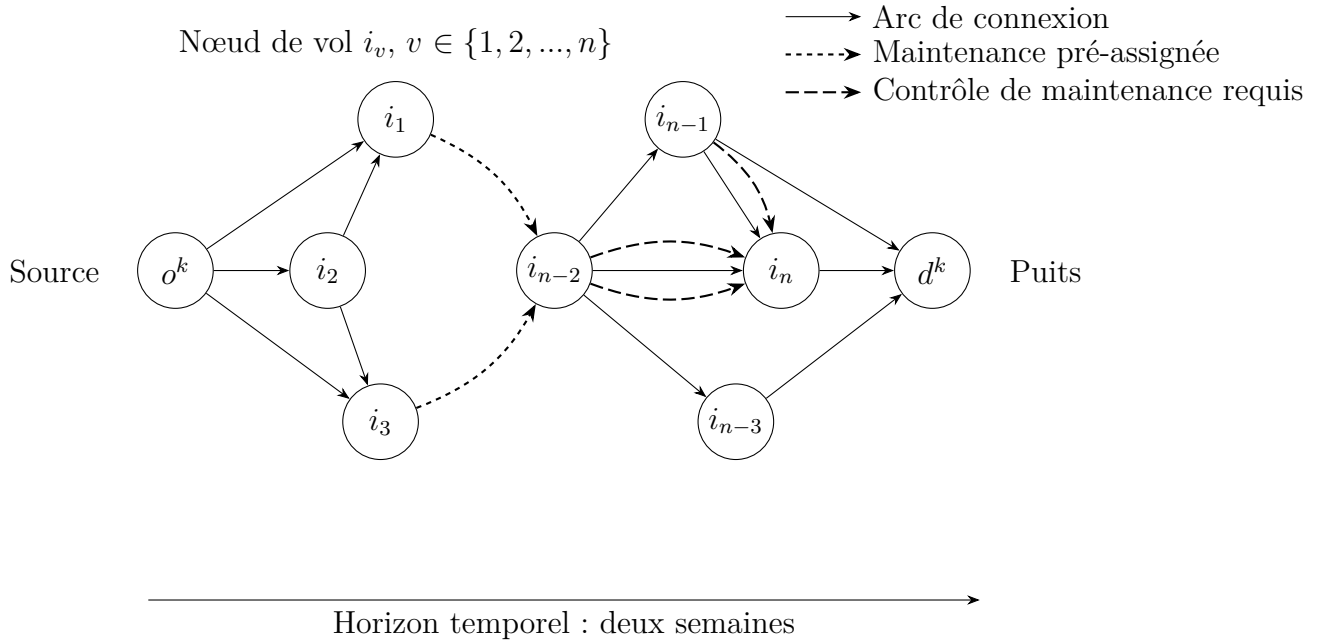


FIGURE 4.1 Réseau pour un avion k

Puis, l'avion subit une maintenance pré-assignée ; elle est représentée par les arcs pointillés (i_1, i_{n-2}) ou (i_3, i_{n-2}). Les arcs en conflit sont supprimés et seuls les arcs de maintenance pré-assignée sont conservés. Par conséquent, l'avion sera obligatoirement acheminé vers une station de maintenance située à l'aéroport de départ du vol i_{n-2} . Les arcs pointillés pour les contrôles de maintenance requis (i_{n-2}, i_n) et (i_{n-1}, i_n) présentent les opportunités et les critères de différents contrôles de maintenance pour cet avion.

Enfin, l'avion peut terminer sa rotation après l'exécution de n'importe quel vol, ce qui permet

de relier chaque vol au nœud *puits* d^k , indépendamment du fait que l'aéroport d'arrivée soit ou non une station de maintenance.

Pour un avion k , l'ensemble des routes réalisables correspond aux chemins reliant le nœud *source* o^k au nœud *puits* d^k dans le réseau acyclique $\mathcal{G}^k$. Toutefois, tous les chemins $o^k - d^k$ ne représentent pas nécessairement des routes faisables lorsque les règles de maintenance de l'avion imposent la réalisation d'une maintenance. Afin de prendre en compte ces exigences et de forcer l'exécution des contrôles de maintenance requis au bon moment, il est nécessaire d'introduire des contraintes de ressources.

4.3.2 Ressources

Afin de garantir qu'une route soit réalisable, il faut que celle-ci respecte les contraintes de ressources introduites. Formellement, une ressource est une valeur prise à chaque nœud dans un chemin qui est fonction de la valeur de la consommation de la ressource sur l'arc entrant dans le nœud et de la valeur de la ressource au dernier nœud dans le chemin. Pour un avion donné k , plusieurs cas d'exigences de maintenance doivent être considérés (en plus des maintenances pré-assignées) :

- Aucune maintenance n'est requise ;
- Une maintenance doit être planifiée avant d'atteindre un temps de vol maximal ;
- Une maintenance doit être planifiée avant d'atteindre un nombre maximal de cycles ;
- Une maintenance doit être planifiée avant d'atteindre un nombre maximal de jours ;
- Toute combinaison des trois règles précédentes.

En plus de ces règles, lorsqu'une contrainte maximale est considérée, une borne minimale sur le même critère peut également être imposée. Nous supposons qu'il y a au plus une maintenance à planifier par avion, à l'exclusion des maintenances pré-assignées.

On a, par conséquent, une fenêtre de ressources pour chaque nœud, indiquant la faisabilité d'un chemin p . Ainsi qu'une consommation de cette ressource sur chaque arc. L'état de l'avion k est suivi par sept ressources distinctes, comme indiqué dans le tableau 4.1. La ressource Tr^C accumule le coût réduit du chemin. Les ressources de consommation T_p^ℓ , $\ell \in \{fT, nC, nD\}$, suivent l'utilisation de l'avion depuis sa dernière maintenance : elles augmentent selon le temps de vol b_j^{fT} , le nombre de cycles b_j^{nC} associés au vol j , ou d'une unité pour la ressource temporelle T_p^{nD} lors d'un changement de jour.

Ces compteurs sont contraints par les bornes supérieures $\overline{M}_{k,m}^\ell$ associées au type de maintenance m . La valeur -1 agit comme un indicateur signifiant qu'une maintenance a déjà été réalisée sur le chemin, neutralisant ainsi la ressource. Enfin, les ressources de seuil $T_p^{n\ell}$,

$\ell \in \{fT, nC, nD\}$, permettent de vérifier l'éligibilité de l'avion à une maintenance. Elles garantissent que l'appareil a atteint les seuils minimaux requis $\underline{M}_{k,m}^\ell$ avant d'autoriser l'emprunt d'un arc de maintenance, évitant ainsi des interventions prématurées.

TABLEAU 4.1 Définition de l'ensemble des ressources

Ressource	Description	Consommation sur arc (i, j)	Bornes
T^{rC}	Coût réduit	$\bar{c}_{ijk}$	$[-\infty, +\infty]$
<i>Ressources de consommation (Bornes Supérieures)</i>			
T^{fT}	Temps de vol	$+b_j^{fT}$ (ou (-1) si maint.)	$\{-1\} \cup [0, \bar{M}_{k,m}^{fT}]$
T^{nC}	Nb. de cycles	$+b_j^{nC}$ (ou (-1) si maint.)	$\{-1\} \cup [0, \bar{M}_{k,m}^{nC}]$
T^{nD}	Nb. de jours	$+1$ (si jour change) ou $((-1)$ si maint.)	$\{-1\} \cup [0, \bar{M}_{k,m}^{nD}]$
<i>Ressources de seuil (Bornes Minimales)</i>			
T^{nfT}	Seuil vol	$-b_j^{nC}$ (si $T^{fT} < \underline{M}_{k,m}^{fT}$)	$[-\underline{M}_{k,m}^{fT}, 0]$
T^{nnC}	Seuil cycles	$-b_j^{nC}$ (si $T^{nC} < \underline{M}_{k,m}^{nC}$)	$[-\underline{M}_{k,m}^{nC}, 0]$
T^{nnD}	Seuil jours	-1 (si jour change) et $(T^{nD} < \underline{M}_{k,m}^{nD})$	$[-\underline{M}_{k,m}^{nD}, 0]$

4.3.3 Résolution des sous-problèmes

Pour résoudre ce type de *SPPRC*, on utilise la programmation dynamique [37] [44]. L'idée est d'étendre itérativement le chemin initial qui comporte juste le nœud *source* en des chemins réalisables et Pareto-optimaux jusqu'à atteindre le nœud *puits*. Les différents chemins partiels sont stockés à l'aide d'étiquettes. Cet algorithme est appelé un algorithme d'étiquetage. À chaque itération, on regarde les derniers nœuds atteints par les différents chemins et on étend les chemins aux nœuds réalisables suivants. On utilise des règles de dominance pour éliminer des chemins partiels non Pareto-optimaux.

Dans le cas le plus général (toutes les règles s'appliquent), une étiquette

$$E_p = (T_p^{rC}, T_p^{fT}, T_p^{nC}, T_p^{nD}, T_p^{nfT}, T_p^{nnC}, T_p^{nnD}),$$

est associée à un chemin partiel p est définie par les composantes de ressources suivantes :

T_p^{rC} : le coût réduit de p ;

T_p^{fT} : le temps de vol cumulé avant la maintenance (égal à -1 après la maintenance) ;

T_p^{nC} : le nombre total de cycles avant la maintenance (égal à -1 après la maintenance) ;

- T_p^{nD} : le nombre total de jours avant la maintenance (égal à -1 après la maintenance) ;
- T_p^{nfT} : l'opposé de T_p^{fT} si le temps de vol minimal n'est pas encore atteint ;
- T_p^{nnC} : l'opposé de T_p^{nC} si le nombre minimal de cycles n'est pas encore atteint ;
- T_p^{nnD} : l'opposé de T_p^{nD} si le nombre minimal de jours n'est pas encore atteint.

Une étiquette E_p peut être étendue le long d'un arc (i, j) sauf si (i, j) contient un contrôle de maintenance requis et $T_p^{fT} = T_p^{nC} = T_p^{nD} = -1$.

Lorsque l'extension est possible, elle génère une nouvelle étiquette $E_{p'}$ calculée à l'aide des fonctions d'extension de ressources (FER), où $\bar{c}_{i,j}$ représente le coût ajusté de l'arc (i, j) :

$$T_{p'}^{rC} = \begin{cases} T_p^{rC} + \bar{c}_{ij} + \sum_{\ell \in \{fT, nC, nD\}} \gamma_m^{E, \ell} (\underline{M}_{k,m}^\ell - \min\{\underline{M}_{k,m}^\ell, T_p^\ell\}) & \text{si } (i, j) \text{ contient une maintenance,} \\ T_p^{rC} + \bar{c}_{ij} + \sum_{\ell \in \{fT, nC, nD\}} \gamma_m^{L, \ell} (\max\{O_{k,m}^\ell, T_{p'}^\ell\} - \max\{O_{k,m}^\ell, T_p^\ell\}) & \text{sinon.} \end{cases} \quad (4.10)$$

$$T_{p'}^{fT} = \begin{cases} -1 & \text{si } T_p^{fT} = -1 \text{ ou si } (i, j) \text{ contient une maintenance,} \\ T_p^{fT} + b_j^{fT} & \text{sinon.} \end{cases} \quad (4.11)$$

$$T_{p'}^{nC} = \begin{cases} -1 & \text{si } T_p^{nC} = -1 \text{ ou si } (i, j) \text{ contient une maintenance,} \\ T_p^{nC} + b_j^{nC} & \text{sinon.} \end{cases} \quad (4.12)$$

$$T_{p'}^{nD} = \begin{cases} -1 & \text{si } T_p^{nD} = -1 \text{ ou si } (i, j) \text{ contient une maintenance,} \\ T_p^{nD} + 1 & \text{si } (i, j) \text{ implique un changement de jour,} \\ T_p^{nD} & \text{sinon.} \end{cases} \quad (4.13)$$

$$T_{p'}^{nfT} = \begin{cases} -\underline{M}_{k,m}^{fT} & \text{si } T_{p'}^{fT} = -1 \text{ ou } T_{p'}^{fT} \geq \underline{M}_{k,m}^{fT}, \\ -T_{p'}^{fT} & \text{sinon.} \end{cases} \quad (4.14)$$

$$T_{p'}^{nnC} = \begin{cases} -\underline{M}_{k,m}^{nC} & \text{si } T_{p'}^{nC} = -1 \text{ ou } T_{p'}^{nC} \geq \underline{M}_{k,m}^{nC}, \\ -T_{p'}^{nC} & \text{sinon.} \end{cases} \quad (4.15)$$

$$T_{p'}^{nnD} = \begin{cases} -\underline{M}_{k,m}^{nD} & \text{si } T_{p'}^{nD} = -1 \text{ ou } T_{p'}^{nD} \geq \underline{M}_{k,m}^{nD}, \\ -T_{p'}^{nD} & \text{sinon.} \end{cases} \quad (4.16)$$

Afin d'affiner le calcul du coût réduit T_p^{rC} et d'orienter l'algorithme vers des choix de planification plus efficaces, nous intégrons au modèle des fonctions de pénalité associées aux écarts

par rapport aux valeurs cibles de maintenance dans la FER (4.10).

Il est à noter que la FER (4.10) intègre les pénalités de maintenance C_{klm}^P (3.1). Le premier cas (avec maintenance) ajoute une pénalité si la maintenance est hâtive par rapport à chacun des critères applicables. Le second cas (sans maintenance) ajoute une portion de la pénalité pour une maintenance tardive. Il est à noter qu'aucun ajout n'est appliqué dans ce cas si la maintenance a déjà eu lieu ($T_p^\ell = T_{p'}^\ell = -1$). Il est aussi à noter que les FER (4.14)–(4.16) garantissent que la ressource correspondante reste constante une fois la borne minimale atteinte, ce qui permet d'éliminer plus d'étiquettes par dominance.

L'étiquette résultante $E_{p'}$ est dite réalisable si $T_{p'}^{fT} \leq \overline{M}_{k,m}^{fT}$, $T_{p'}^{nC} \leq \overline{M}_{k,m}^{nC}$ et $T_{p'}^{nD} \leq \overline{M}_{k,m}^{nD}$.

Notons qu'au nœud *puits*, il n'est pas obligatoire de satisfaire les bornes minimales, puisqu'il n'est pas requis d'inclure une maintenance si les bornes supérieures ne sont pas violées.

Pour initialiser l'algorithme d'étiquetage, une étiquette initiale,

$$E_0 = (0, z^{fT}, z^{nC}, z^{nD}, z^{nfT}, z^{nnC}, z^{nnD}),$$

est associée au nœud *source* du réseau, où z^ℓ , $\ell \in \{fT, nC, nD\}$, vaut 0 si la règle de maintenance sur ℓ s'applique et -1 sinon, et z^ℓ , $\ell \in \{nfT, nnC, nnD\}$, vaut 0 si la règle s'applique et $-\underline{M}_{k,m}^\ell$ sinon.

Enfin, afin de limiter le nombre d'étiquettes, les règles de dominance suivantes sont utilisées pour éliminer les étiquettes non optimales. L'étiquette E_p domine $E_{p'}$ si :

$$T_p^{rC} \leq T_{p'}^{rC} - \sum_{\ell \in \{fT, nC, nD\}} \gamma_m^{E,\ell} f^{E,\ell}(T_p^\ell, T_{p'}^\ell), \quad (4.17)$$

$$T_p^{fT} \leq T_{p'}^{fT}, \quad (4.18)$$

$$T_p^{nC} \leq T_{p'}^{nC}, \quad (4.19)$$

$$T_p^{nD} \leq T_{p'}^{nD}, \quad (4.20)$$

$$T_p^{nfT} \leq T_{p'}^{nfT}, \quad (4.21)$$

$$T_p^{nnC} \leq T_{p'}^{nnC}, \quad (4.22)$$

$$T_p^{nnD} \leq T_{p'}^{nnD}. \quad (4.23)$$

où

$$f^{E,\ell}(T_p^\ell, T_{p'}^\ell) = \begin{cases} \min\{O_{k,m}^\ell, T_{p'}^\ell\} - T_p^\ell & \text{si } T_p^\ell < O_{k,m}^\ell, \\ 0 & \text{sinon.} \end{cases} \quad (4.24)$$

représente la plus grande différence possible entre les quantités (minutes, cycles ou jours) susceptibles d'être soumises à une pénalité de maintenance anticipée pour les deux étiquettes. Notons que, d'après les conditions (4.18) - (4.20), on a $T_p^\ell \leq T_{p'}^\ell$, ce qui implique $f^{E,\ell}(T_p^\ell, T_{p'}^\ell) \geq 0$.

Il est à noter que la composante T_p^{nnD} pourrait être omise en supprimant du réseau tous les arcs incluant une maintenance violant la contrainte sur le nombre minimal de jours. Toutefois, son maintien n'est pas pénalisant, car toutes les étiquettes associées à un même nœud partagent la même valeur de T_p^{nnD} . De même, les arcs violant la contrainte sur le nombre maximal de jours pourraient être retirés du réseau. Même si la composante T_p^{nD} prend la même valeur pour toutes les étiquettes associées à un même nœud, elle demeure utile pour calculer les pénalités associées à une maintenance anticipée ou tardive.

En conclusion, l'algorithme va naturellement chercher un compromis entre effectuer la maintenance avant la borne inférieure et la faire après la cible pour chaque critère. Il va donc chercher à faire rentrer l'avion en maintenance le plus souvent possible dans la zone gratuite $[\underline{M}_{k,m}^\ell, O_{k,m}^\ell]$.

4.4 Recherche d'une solution entière

La résolution du *PM* par *GC* fournit une solution optimale qui peut être fractionnaire. Une telle solution n'est pas admissible d'un point de vue opérationnel, puisqu'un avion ne peut exécuter qu'une seule route complète. Pour obtenir une solution entière dans ce cas, on explore un arbre de branchement de façon heuristique, plus précisément, on fait une descente en profondeur dans une seule branche de cet arbre. Après avoir résolu chaque relaxation linéaire, on applique la technique de fixation de colonnes suivante.

Fixation de colonnes (*cfix*) : La technique de fixation de colonnes consiste à fixer les valeurs d'un sous-ensemble de variables du *PM* (les routes d'avions) à chaque nœud. Lorsqu'une solution fractionnaire est obtenue, l'ensemble des variables est analysé afin d'identifier un ensemble de candidates présentant des valeurs fractionnaires. Ce score correspond à la partie fractionnaire de la valeur de la variable dans la solution courante.

Le score associé à chaque variable correspond à sa valeur dans la solution courante. Soit $\tilde{x}_{rk}$ la valeur de la variable x_{rk} dans la solution courante. Le sous-ensemble de variables à fixer

est ensuite construit en sélectionnant, parmi les candidates, celles dont $\tilde{x}_{rk}$ est fractionnaire et supérieure ou égale à un seuil prédéfini.

La taille de ce sous-ensemble est contrôlée soit par une borne supérieure sur le nombre de variables sélectionnées, soit par une borne sur la somme de leurs valeurs complémentaires, définies par $1 - \tilde{x}_{rk}$. Les variables ayant les scores les plus faibles sont progressivement éliminées jusqu'à satisfaction de la borne imposée. Il est également possible d'imposer une borne inférieure sur la cardinalité du sous-ensemble ; dans ce cas, certaines variables dont le score est inférieur au seuil peuvent être retenues afin de satisfaire cette contrainte. S'il n'y a aucune variable avec une valeur fractionnaire excédant le seuil, la variable avec la plus grande valeur fractionnaire est choisie.

Dans la stratégie *cfx*, une seule branche est créée dans l'arbre de recherche, au sein de laquelle chacune des variables sélectionnées est forcée à prendre la valeur 1. Les valeurs correspondantes des colonnes fixées sont ensuite retranchées des membres de droite des contraintes du *PM* et les variables fixées à 1 sont mises de côté.

4.5 Stratégies d'accélération

Malgré le recours à la *GC* et à une stratégie de branchement heuristique, la résolution des grandes instances demeure difficile. Cette difficulté s'explique en partie par le phénomène de dégénérescence. La dégénérescence apparaît lorsqu'une même solution optimale correspond à plusieurs bases admissibles. Dans ce cas, la méthode du simplexe peut effectuer un grand nombre d'itérations au cours d'une même itération de *GC*. De plus, les nouvelles colonnes générées peuvent entrer dans la base sans modifier la solution courante, en conservant une valeur nulle. Il en résulte un alourdissement de la résolution du *PM*. Par ailleurs, pour les instances de grande taille, la résolution des *SPs* devient également plus complexe. Plusieurs mécanismes peuvent être mis en œuvre afin d'obtenir plus rapidement une solution quasi-optimale.

4.5.1 Stratégies d'arrêt de *GC*

Premièrement, la *GC* peut être tronquée lors de la résolution du *PM* lorsque l'amélioration du coût de la solution du *PMR* courant devient inférieure à un seuil donné pendant un nombre fixé d'itérations. Cette stratégie limite les itérations peu productives et contribue à réduire les effets liés à la dégénérescence.

Deuxièmement, on peut sélectionner un certain nombre de colonnes de coût réduit négatif et les ajouter à R_k^t pour former R_k^{t+1} . En effet, on peut choisir d'ajouter une seule colonne

ou bien un nombre maximal de colonnes ou bien toutes les colonnes. Le nombre de colonnes ajoutées à chaque itération influence le temps par itération et le nombre d'itérations pour atteindre la solution optimale. Ensuite, une nouvelle itération $t + 1$ de GC est faite.

Enfin, pour les instances de grande taille, la résolution des sous-problèmes devient elle-même coûteuse. En particulier, l'application des règles de dominance sur l'ensemble complet des ressources peut entraîner un temps de calcul important, en raison du nombre élevé de comparaisons nécessaires entre les étiquettes. Afin de réduire cette complexité, il est possible de restreindre le test de dominance à un sous-ensemble de ressources. Cette approche permet d'accélérer le filtrage des étiquettes en réduisant le coût des comparaisons, tout en éliminant efficacement celles qui ont peu de chances d'être prolongées de manière optimale. Bien que cette dominance partielle soit moins stricte que la dominance complète, elle constitue un compromis pertinent entre la qualité de la solution et l'efficacité computationnelle. En complément, le nombre d'étiquettes par nœud peut être limité à un seuil prédéfini pour en préserver au plus un nombre fixé à chaque nœud.

4.5.2 Stratégies d'élimination d'arcs de connexion

Malgré ces améliorations, la résolution des grandes instances peut rester lente. L'approche développée dans cette étude consiste alors à restreindre le graphe initial à un sous-ensemble d'arcs sélectionnés, afin de réduire la taille effective du problème et d'en accélérer la résolution.

Afin d'améliorer l'efficacité du processus de GC , nous avons développé une stratégie d'accélération fondée sur une heuristique de réduction de taille de réseau des SPs , consistant à réduire temporairement la taille des réseaux d'avions considérés dans les SPs .

Pour chaque avion $k \in K$, on considère le réseau acyclique $\mathcal{G}^k = (\mathcal{N}^k, \mathcal{A}^k)$. Lorsque l'heuristique est activée à une itération donnée, une procédure d'élimination d'arcs est appliquée à chaque vol $i \in \mathcal{N}^k$. On note N_i le nombre total d'arcs de connexion sans maintenance sortant de i .

Pour chaque arc $(i, j) \in \mathcal{A}^k$ ne correspondant pas à une maintenance, on définit deux rangs : R_{ij}^{cost} , obtenu en classant les arcs sortants de i par coût croissant ($c_{ijk} + c_{ijk}^G$), et R_{ij}^{dur} , obtenu en les classant par durée de connexion croissante. L'arc (i, j) est supprimé du réseau si la condition suivante est satisfaite :

$$\left(\min\{R_{ij}^{\text{cost}}, R_{ij}^{\text{dur}}\} > \text{MinConn} \right) \quad \text{et} \quad \left(\frac{\min\{R_{ij}^{\text{cost}}, R_{ij}^{\text{dur}}\}}{N_i} > \text{RankThres} \right),$$

où MinConn et RankThres sont les paramètres de l'heuristique.

Cette règle garantit qu'un arc est éliminé uniquement s'il est simultanément mal classé selon les deux critères considérés et si un nombre minimal de connexions plus attractives demeure disponible pour le vol i . Le réseau réduit conserve ainsi les arcs les plus prometteurs tout en limitant l'espace de recherche des SPs .

Pour les très grandes instances où la résolution des SPs devient elle aussi coûteuse, une approche en deux phases peut être adoptée. Dans un premier temps, chaque nœud de l'arbre de branchement est traité en appliquant l'heuristique de réduction de la taille, jusqu'à ce que l'amélioration de la solution du PMR courant devienne insuffisante. Dans un second temps, l'heuristique est désactivée afin de chercher une meilleure solution. Des expérimentations montrent que cette stratégie heuristique permet d'accélérer la résolution des SPs . Dans chacune de ces deux phases, un ajustement des paramètres est fait pour trouver la meilleure combinaison de paramètres.

CHAPITRE 5 RÉSULTATS NUMÉRIQUES

Dans ce chapitre, nous présentons les résultats des expérimentations numériques que nous avons menées afin de comparer les performances de la méthode de résolution décrite au chapitre 4. Tout d’abord, les instances d’*OAMRP* sont présentées dans la section 5.1. Puis l’implémentation de la méthode de résolution, les résultats préliminaires et les paramètres utilisés sont discutés dans la section 5.2. Enfin, la section 5.3 présente les principaux résultats comparatifs obtenus.

5.1 Présentation des instances

Nous avons travaillé sur des jeux de données réelles fournis par notre partenaire industriel, *IBS*. Ces données se composent de deux ensembles distincts, chacun regroupant plusieurs instances. Chaque instance correspond à une flotte donnée, elle-même constituée d’une ou de plusieurs sous-flottes. Notre objectif étant d’optimiser le routage des avions ainsi que la planification des maintenances dans un contexte d’exploitation bimensuelle, toutes les instances considérées dans notre étude couvrent un horizon de planification de 14 jours. Chaque instance inclut notamment les informations relatives aux vols (origines, destinations, horaires), aux connexions réalisables entre vols, aux stations de maintenance disponibles ainsi qu’aux opérations de maintenance pré-assignées et aux contrôles de maintenance requis.

Les instances considérées, présentées dans le tableau 5.1, se caractérisent par une grande hétérogénéité en termes de taille. Le nombre d’avions varie entre 10 et 158, tandis que le nombre de vols s’échelonne de 228 à 1 598 sur un horizon de 14 jours. Le nombre de connexions réalisables entre vols est particulièrement élevé, atteignant plusieurs millions pour les plus grandes instances, ce qui illustre la forte combinatoire du problème. Par ailleurs, le nombre d’aéroports varie entre 11 et 99, dont 2 stations de maintenance, et les instances comprennent plusieurs dizaines de maintenances pré-assignées et de contrôles de maintenance requis.

Durant chaque quart de travail, chaque station dispose d’une capacité par type de contrôle de maintenance, reflétant les ressources humaines et équipements disponibles (voir tableau 5.2). Cette capacité correspond au nombre d’avions pouvant être entretenus simultanément selon la sous-flotte, la classe de maintenance et le type d’intervention. Chaque semaine comprend sept jours (1 à 7), chacun divisé en deux quarts. La capacité liée à la maintenance pré-assignée est réservée à l’avance et n’est donc pas contraignante.

TABLEAU 5.1 Caractéristiques des instances

Instance	Sous-flottes	Nb. avions	Nb. vols	Nb. connexions	Nb. aéro-ports	Nb. stations maint.	Nb. maint. pré-assignées	Nb. cont-rôles requis	Horizon (jours)
I1-A	77LJ	20	228	122 068	18	1	7	5	14
I1-B	77WP 77WN	89	907	3 249 146	82	1	19	15	14
I1-C	388H 388J 388L 388P 388U 388Y	158	1131	997 831	51	2	139	12	14
I2-A	77LJ	10	228	60 726	18	1	7	5	14
I2-B	77WP 77WR 77WN	84	1598	3 372 925	99	1	62	24	14
I2-C	388H 388J 388L 388P 388U 388Y	73	1131	449 502	51	2	140	12	14

TABLEAU 5.2 Activités et lieux de maintenance

Instance	Sous-flottes	Classe de maintenance	Contrôle de maintenance	Aéroport	Quart 1	Quart 2	Capacité
I1-A	77LJ	Préassigné	-	DXB	1234567	1234567	∞
		Contrôle requis	SV	DXB	1234567	1234567	2
			25DY	DXB	1234567	1234567	2
I1-B	77WP 77WN	Préassigné	-	DXB	1234567	1234567	∞
		Contrôle requis	SV	DXB	1234567	1234567	3
			25DY	DXB	1234567	1234567	3
I1-C	388H 388J 388L 388P 388U 388Y	Préassigné	-	DXB	1234567	1234567	∞
				KHB	1234567	1234567	∞
		Contrôle requis	SV	DXB	1234567	1234567	5
I2-A	77LJ	Préassigné	-	DXB	1234567	1234567	∞
		Contrôle requis	SV	DXB	1234567	1234567	2
			25DY	DXB	1234567	1234567	2
I2-B	77WP 77WR 77WN	Préassigné	-	DXB	1234567	1234567	∞
		Contrôle requis	SV	DXB	1234567	1234567	5
			25DY	DXB	1234567	1234567	5
I2-C	388H 388J 388L 388P 388U 388Y	Préassigné	-	DXB	1234567	1234567	∞
				KHB	1234567	1234567	∞
		Contrôle requis	SV	DXB	1234567	1234567	5

Les valeurs des paramètres associés à la modélisation du problème sont présentées dans le tableau 5.3. Ces paramètres ont été maintenus constants pour l'ensemble des instances.

TABLEAU 5.3 Définition de l'ensemble des paramètres liés à la modélisation du problème

Paramètre	Description	Valeur
<i>Pénalité pour maintenance anticipée</i>		
$\gamma_m^{E,FT}$	Pénalité encourue par minute de vol lorsque la maintenance est effectuée avant $\underline{M}^{fT}$	20
$\gamma_m^{E,nC}$	Pénalité encourue par cycle lorsque la maintenance est effectuée avant $\underline{M}^{nC}$	20
$\gamma_m^{E,nD}$	Pénalité encourue par jour lorsque la maintenance est effectuée avant $\underline{M}^{nD}$	20
<i>Pénalité pour maintenance tardive</i>		
$\gamma_m^{L,FT}$	Pénalité encourue par minute de vol lorsque la maintenance est effectuée entre O^{fT} et $\overline{M}^{fT}$	60
$\gamma_m^{L,nC}$	Pénalité encourue par cycle lorsque la maintenance est effectuée entre O^{nC} et $\overline{M}^{nC}$	60
$\gamma_m^{L,nD}$	Pénalité encourue par jour lorsque la maintenance est effectuée entre O^{nD} et $\overline{M}^{nD}$	60
<i>Contribution de maintenance au quart de travail</i>		
ξ^{min}	Durée minimale requise en minutes	30
<i>Violation maximale de temps minimal au sol entre deux vols consécutifs</i>		
Δ^G	Valeur maximale de violation autorisée en minutes	20
<i>Pénalité pour violation de temps minimal au sol</i>		
c_{fix}^G	Pénalité fixe	500
c_{var}^G	Pénalité variable par minute	10

5.2 Implémentation et résolution

5.2.1 Outil de résolution

Les instances décrites précédemment sont résolues avec le logiciel GENCOL, un programme d'optimisation spécialement conçu pour résoudre une vaste classe de problèmes de routage et

de planification à grande échelle. Les fondements théoriques du programme ont été établis par Desrosiers et Soumis en 1981 [47], dont les travaux pionniers sur la *GC* ont constitué la base de sa conception. Depuis sa création, GENCOL n’a cessé d’évoluer grâce aux contributions de nombreux étudiants aux cycles supérieurs et analystes en informatique, donnant lieu à d’importantes avancées méthodologiques et computationnelles.

Les expériences ont été réalisées sur un ordinateur Linux équipé d’un processeur Intel Core i7-12700 cadencé jusqu’à 4,9 GHz et de 64 Go de mémoire RAM. Les algorithmes ont été implémentés en C et C++ à l’aide de la bibliothèque commerciale GENCOL, version 4.5.

5.2.2 Résultats préliminaires et configuration des paramètres

Afin d’évaluer les temps de calcul, des tests préliminaires ont été réalisés et présentés dans le tableau 5.4. Pour ces tests, aucun paramètre d’accélération n’était contrôlé. Sur la base des résultats computationnels obtenus, les instances ont été classées en trois catégories selon leur taille : petites, moyennes et grandes. Les petites instances (I1-A1, I2-A, I2-B) nécessitent un temps de calcul inférieur à une minute, les instances de taille moyenne (I1-C) présentent un temps de calcul de l’ordre de 15 minutes, tandis que les grandes instances (I1-B et I2-B) dépassent 1 heure de temps de calcul.

TABLEAU 5.4 Résultats préliminaires

Instance	Coût total	Temps (min)	Couverture de vols (%)	Coût de maintenance
I1-A	1 638 010	< 1	98,68	0
I1-B	5 607 090	75.23	99,78	0
I1-C	7 619 130	14.94	99,47	2 560
I2-A	5 350 220	< 1	82,02	92 220
I2-B	26 594 100	54.3	89,05	33 900
I2-C	16 520 200	< 1	92,40	479 520

Pour les instances A, B et C des deux ensembles de données, les sous-flottes sont globalement similaires. Néanmoins, l’instance B de l’ensemble 2 inclut une sous-flotte supplémentaire. De plus, l’ensemble 1 comporte un plus grand nombre de connexions possibles ainsi qu’un nombre plus élevé d’avions disponibles. Ces différences, associées au nettoyage préalable des données, expliquent le taux de couverture des vols plus élevé atteignant le 99 % observé pour l’ensemble

1 tandis que celui de l'ensemble 2 varie entre 82 % et 89 %.

Afin d'identifier où le temps de calcul est principalement investi lors de la résolution des différentes instances, le tableau 5.5 fournit des statistiques supplémentaires pour les performances computationnelles des différentes instances. Il présente les temps consacrés à la résolution du *PM* et à la résolution des *SPs* (la somme de ces deux temps peut être inférieure au temps total lorsque les réseaux sont de grande taille car ils sont reconstruits à chaque itération). Le temps total est dans le tableau 5.4. Le pourcentage dans la colonne de temps pour les *SPs* indique la proportion du temps passé pour les résoudre par rapport au temps total de la *GC*. L'objectif de cette analyse est de comprendre les goulots d'étranglement de l'algorithme et d'orienter d'éventuelles améliorations. Enfin, le nombre d'itérations de génération de colonnes ainsi que le gap d'intégrité sont également rapportés afin de mesurer à la fois l'efficacité de la convergence et la qualité des solutions obtenues. Le gap d'intégrité correspond à la différence relative, exprimée en pourcentage, entre la valeur de la solution finale entière et la borne inférieure obtenue à la racine (nœud 0) de l'arbre de branchement.

TABLEAU 5.5 Performances computationnelles des résultats préliminaires

Instance	Temps (s)		#Itérations de <i>GC</i>	Gap d'intégrité (%)
	<i>PM</i>	<i>SPs</i>		
I1-A	1,1	3,8 (23,75 %)	494	$1,83 \times 10^{-6}$
I1-B	85,2	3385,1 (77,29 %)	2627	$6,42 \times 10^{-7}$
I1-C	21,6	844,8 (97,08 %)	1280	$8,66 \times 10^{-6}$
I2-A	0,6	3,8 (25,50 %)	463	$1,87 \times 10^{-7}$
I2-B	212,1	1894,3 (63,40 %)	2354	$5,15 \times 10^{-7}$
I2-C	6,7	20,6 (72,28 %)	522	$8,09 \times 10^{-4}$

Pour la majorité des instances (notamment I1-C avec 97,08 % et I1-B avec 77,29 %), les sous-problèmes constituent le goulot d'étranglement majeur. Les gaps d'intégrité sont extrêmement faibles, allant de $1,87 \times 10^{-7}$ % à $8,09 \times 10^{-4}$ %. Même pour les instances les plus lourdes à résoudre (comme I1-B), le gap reste infime ($6,42 \times 10^{-7}$ %), ce qui démontre la robustesse de la méthode employée. Afin de réduire les temps de calcul élevés, des méthodes d'accélération ont été mises en œuvre sur les *SPs*, notamment l'heuristique de réduction de taille, la limitation du nombre d'étiquettes par nœud ainsi que les méthodes d'arrêt de *GC*, tout en analysant leur impact sur la fonction objectif, représentée par le coût total. Le tableau 5.6 présente l'ensemble des paramètres utilisés pour la configuration de l'algorithme, regroupés selon leur rôle dans la résolution des sous-problèmes, le contrôle du nombre d'itérations et l'heuristique de réduction de taille du réseau.

TABLEAU 5.6 Définition de l'ensemble des paramètres liés à la configuration d'algorithme

Paramètre	Description
<i>Paramètres de résolution des SPs</i>	
Dominance	Chaîne de caractères définissant les ressources spécifiques (temps de vol, cycles, jours écoulés ainsi que leurs bornes supérieure et inférieure) prises en compte dans le test de dominance.
Lab_{max}	Le nombre maximal d'étiquettes conservées par nœud lors de la résolution de <i>SPs</i> .
<i>Paramètres d'itération</i>	
LogITR	Le nombre d'itérations du processus de <i>GC</i> qui doivent être effectuées avant que le solveur ne vérifie la diminution de la fonction objectif.
CostDec	Diminution minimale requise de la fonction objectif sur LogITR pour poursuivre la génération. À défaut, le processus est interrompu prématurément.
<i>Paramètres de l'heuristique de réduction de taille de réseau</i>	
MinConn	Un seuil de classement (exprimé en pourcentage du nombre total de connexions hors maintenance) utilisé pour identifier les arcs à supprimer.
RankThres	Le nombre minimal de connexions hors maintenance à préserver pour chaque vol donné lors de la construction des routes dans le <i>SP</i> .

5.3 Comparaison des résultats

Dans cette étude, les catégories d'instance ont été testées de manière isolée afin d'évaluer précisément l'impact des différentes stratégies. La **dominance** est appliquée sur l'ensemble complet des ressources, ce qui restreignait le nombre d'étiquettes dominées et engendrait un nombre élevé d'étiquettes à traiter. Pour réduire les temps de calcul, une limitation du nombre d'étiquettes conservées par nœud (**Lab_{max}**) a été testée, tout en continuant à exploiter les règles de dominance. Puis l'heuristique de réduction de taille de réseau est activée. Enfin les deux méthodes sont combinées dans une approche hybride.

5.3.1 Variation de **Lab_{max}**

Le **Lab_{max}** a été évalué pour différentes valeurs (5, 20, 50 et 200), tandis qu'une exécution sans limitation est considérée comme cas de référence.

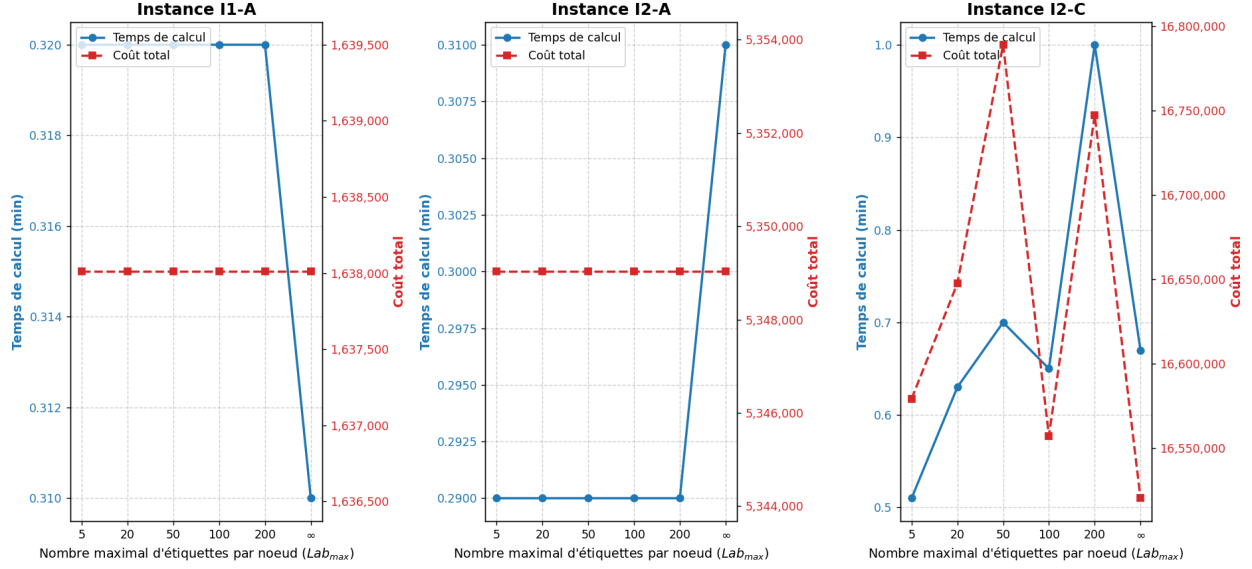
Petites instances

Les résultats sur les petites instances de l'analyse de sensibilité de l'algorithme au paramètre Lab_{max} sont présentés dans le tableau 5.7 et la figure 5.1.

TABLEAU 5.7 Impact de Lab_{max} sur le temps et le coût total pour les petites instances

Instance	Lab_{max}	Temps (min)	Coût total	Gain Temps (%)	Gap Coût (%)
I1-A	-	0,31	1 638 010	-	-
	5	0,32	1 638 010	-3,2	0
	20	0,32	1 638 010	-3,2	0
	50	0,32	1 638 010	-3,2	0
	100	0,32	1 638 010	-3,2	0
	200	0,32	1 638 010	-3,2	0
I2-A	-	0,31	5 349 030	-	-
	5	0,29	5 349 030	6,5	0
	20	0,29	5 349 030	6,5	0
	50	0,29	5 349 030	6,5	0
	100	0,29	5 349 030	6,5	0
	200	0,29	5 349 030	6,5	0
I2-C	-	0,67	16 520 200	-	-
	5	0,51	16 579 000	23.9	0.3559
	20	0,63	16 647 500	6,0	0,7706
	50	0,70	16 788 900	-4,5	1,6265
	100	0,65	16 557 200	3,0	0,2240
	200	1,00	16 747 200	-49,2	1,3741

Pour les deux plus petites instances IA-1 et IA-2, elles montrent que cette restriction n'affecte en rien l'optimalité, le coût total restant inchangé. Pour l'instance I2-C, la configuration $Lab_{max} = 5$ s'avère la plus performante, permettant un gain de temps de 23,9% au prix d'une dégradation marginale de la fonction objectif de seulement 0,35%. En revanche, l'augmentation du seuil au-delà de 50 étiquettes tend à dégrader les performances temporelles sans apporter de gain significatif sur le coût total.

FIGURE 5.1 Analyse de sensibilité au Lab_{max} sur les petites instances

Moyennes instances

Pour l'instance de taille moyenne I1-C, l'analyse de sensibilité du paramètre Lab_{max} montre que la réduction du nombre maximal d'étiquettes par nœud permet un gain de temps très important tout en maintenant la quasi-optimalité de la solution (voir tableau 5.8 et figure 5.2).

TABLEAU 5.8 Impact de Lab_{max} sur le temps et le coût total pour les moyennes instances

Instance	Lab_{max}	Temps (min)	Coût total	Gain Temps (%)	Gap Coût (%)
I1-C	-	14,94	7 619 130	-	-
	5	1,52	7 619 390	89,8	0,0034
	20	1,65	7 619 290	88,9	0,0021
	50	2,03	7 619 430	86,4	0,0039
	100	2,76	7 619 290	81,5	0,0021
	200	3,63	7 619 070	75,7	-0,0008

En particulier, la configuration $Lab_{max} = 5$ réduit le temps de calcul de 14,94 minutes à seulement 1,52 minute, soit un gain de 89,8 %, avec une dégradation négligeable du coût total de 0,0034 %. Des valeurs plus élevées de Lab_{max} entraînent une augmentation progressive du temps de calcul, tandis que l'impact sur le coût reste quasiment nul (écart inférieur à 0,004 %), confirmant que la sélection rigoureuse des étiquettes les plus prometteuses permet

de diminuer l'effort computationnel tout en préservant la qualité de la solution.

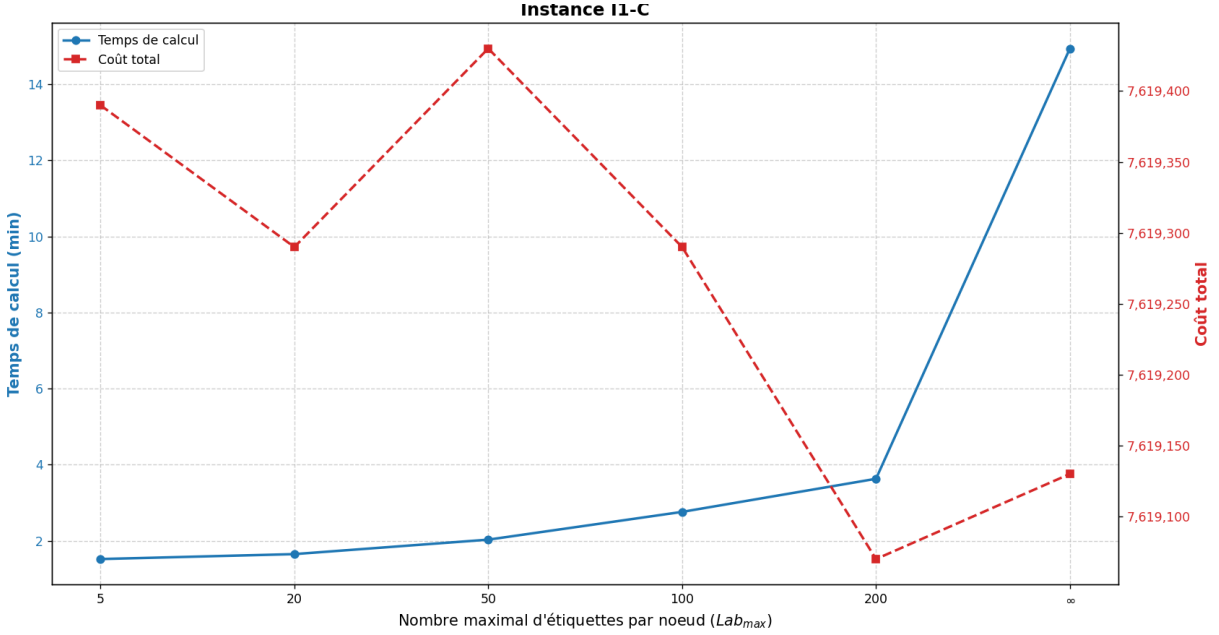


FIGURE 5.2 Analyse de sensibilité au Lab_{max} sur les moyennes instances

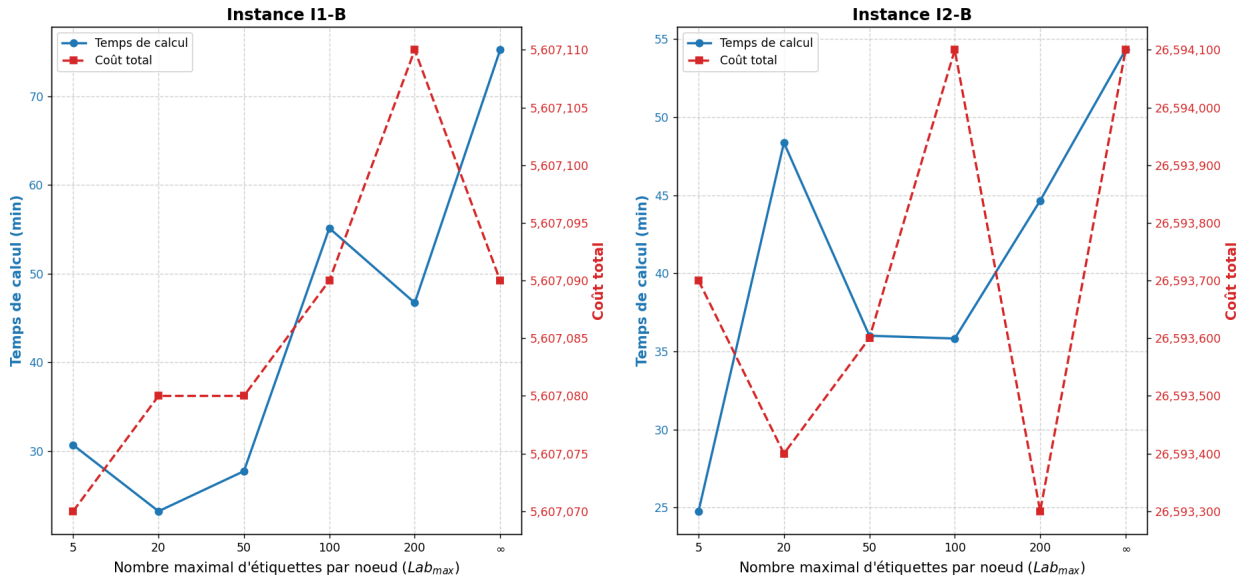
Grandes instances

Pour les grandes instances I1-B et I2-B, l'impact du paramètre Lab_{max} sur les temps de calcul et la qualité des solutions est plus marqué (voir tableau 5.9 et figure 5.3). Pour I1-B, le choix $Lab_{max} = 20$ s'avère optimal, réduisant le temps de calcul de 75,23 minutes à 23,21 minutes, soit un gain de 69,1 %, avec une variation quasi-nulle du coût total (-0,0002 %). Des valeurs plus faibles (5) ou plus élevées (50 et au-delà) conduisent soit à un gain de temps légèrement moindre, soit à une augmentation significative du temps de calcul, sans amélioration du coût.

Pour I2-B, la configuration $Lab_{max} = 5$ offre le meilleur compromis, permettant de réduire le temps de calcul de 54,30 minutes à 24,76 minutes (gain de 54,4 %) avec une dégradation du coût total (-0,0015 %). Des valeurs plus élevées de Lab_{max} entraînent une augmentation des temps de calcul et un impact négligeable sur la qualité de la solution, confirmant l'importance d'une sélection rigoureuse des étiquettes par nœud pour limiter l'effort computationnel tout en conservant la quasi-optimalité.

TABLEAU 5.9 Impact de Lab_{max} sur le temps et le coût total pour les grandes instances

Instance	Lab_{max}	Temps (min)	Coût total	Gain Temps (%)	Gap Coût (%)
I1-B	-	75,23	5 607 090	-	-
	5	30,68	5 607 070	59,2	-0,0004
	20	23,21	5 607 080	69,1	-0,0002
	50	27,73	5 607 080	63,1	-0,0002
	100	55,13	5 607 090	26,7	0,0000
	200	46,71	5 607 110	37,9	0,0004
I2-B	-	54,30	26 594 100	-	-
	5	24,76	26 593 700	54,4	-0,0015
	20	48,36	26 593 400	10,9	-0,0026
	50	36,00	26 593 600	33,7	-0,0019
	100	35,82	26 594 100	34,0	0,0000
	200	44,65	26 593 300	17,8	-0,0030

FIGURE 5.3 Analyse de sensibilité au Lab_{max} sur les grandes instances

5.3.2 Activation de l'heuristique de réduction de la taille de réseau

Afin d'analyser le compromis entre le temps de calcul et le coût total pour l'heuristique de réduction de taille de réseau, une approche bi-objectif est adoptée basée sur la construction d'un front de Pareto empirique. Les différentes configurations expérimentales sont générées à partir de la combinaison de deux paramètres, *MinConn* et *RankThres*, chacun prenant quatre valeurs distinctes, ce qui conduit à un ensemble de 16 scénarios évalués par instance.

Chaque scénario est représenté par un point dans l'espace des solutions, défini par ses performances en termes de temps de calcul et de coût total, deux objectifs à minimiser. Le front de Pareto est ensuite déterminé en identifiant les solutions non dominées, c'est-à-dire celles pour lesquelles aucune autre configuration ne présente simultanément un temps de calcul inférieur et un coût plus faible. En effet, les solutions sont d'abord triées par ordre croissant de temps de calcul, puis un filtrage itératif est appliqué afin de conserver uniquement celles qui améliorent strictement le coût total. Les solutions appartenant à ce front sont mises en évidence graphiquement et reliées afin de visualiser les meilleurs compromis observés entre les deux objectifs. Cette approche permet ainsi de distinguer les configurations efficaces des solutions dominées et de faciliter l'identification de points de compromis pertinents, représentant un équilibre intéressant entre performance et coût computationnel. Il est à noter que lors de l'activation de cette heuristique, aucune limitation sur **Lab_{max}** n'a été imposée. Les détails sur les résultats obtenus sont fournis en annexe A.

Petites instances

Pour les instances I1-A et I2-A, plusieurs points de données sont superposés sur le graphique dans la figure 5.4, car différentes combinaisons de paramètres aboutissent à des résultats identiques en termes de coût et de temps de calcul. L'analyse des résultats pour les instances de type A (I1-A et I2-A) présente une forte stabilité. La quasi-totalité des points se trouve sur une ligne de coût horizontale. Cela signifie que même avec des ressources très limitées (un **MinConn** faible ou un **RankThres** élevé), l'heuristique parvient à trouver la solution optimale ou quasi-optimale. Le coût total converge vers une valeur minimale stable (1 638 010 pour I1-A et 5 343 070 pour I2-A). À l'inverse, l'instance I2-C se caractérise par une variabilité plus marquée des coûts. Le temps de calcul minimal est observé pour une configuration associant une valeur de **MinConn** égale à 5 et un **RankThres** de 0,1, mais celle-ci conduit également au coût le plus élevé (16 595 800). Ce comportement suggère que, pour des instances de petite taille, une réduction trop agressive peut altérer la qualité des solutions en limitant la diversité des chemins explorés. Le meilleur compromis est obtenu pour une faible valeur de **MinConn** égale à 5 combinée à un seuil de **RankThres** égal à 0,75 (coût de 16 518 900).

Globalement, le paramétrage **MinConn** allant de 5 à 10 associé à un **RankThres** de 0,25 à 0,75 apparaît comme un choix robuste, permettant de stabiliser les coûts tout en garantissant une efficacité computationnelle élevée sur l'ensemble des petites instances.

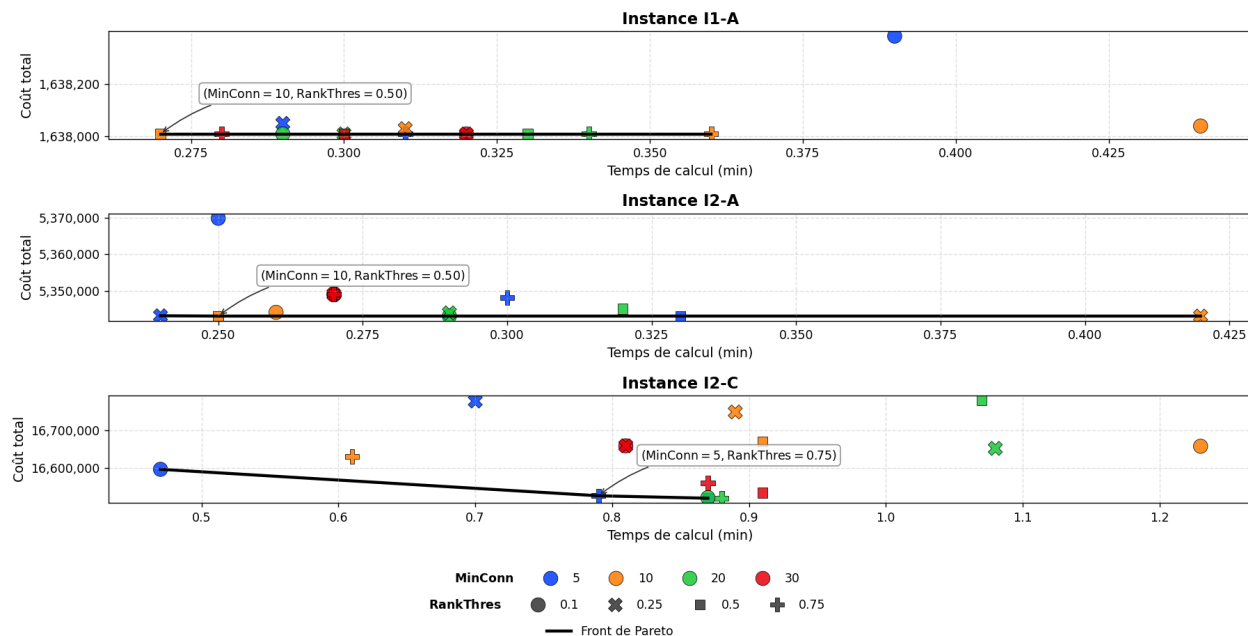


FIGURE 5.4 Front de Pareto pour les petites instances

Moyennes instances

Contrairement aux petites instances (I1-A et I2-A), le front de Pareto sur la figure 5.5 est ici incliné. Réduire le réseau (en allant vers la gauche du graphique) permet de gagner du temps, mais au prix d'une augmentation du coût. L'analyse montre que les performances de l'heuristique dépendent des paramètres *MinConn* et *RankThres*. Trois points distinguant clairement :

- La configuration (*MinConn* = 5 ; *RankThres* = 0,1) résout l'instance en seulement 5 minutes mais avec un coût plus élevé (7 620 390).
- La configuration (*MinConn* = 20 ; *RankThres* = 0,5) atteint le coût minimal (7 618 970) pour un temps de calcul d'environ 13 minutes.
- La configuration (*MinConn* = 10 ; *RankThres* = 0,5) offre un coût quasi-optimal (7 619 420) avec un temps raisonnable (≈ 9 minutes), formant le meilleur compromis entre rapidité et qualité.

Ainsi la zone *MinConn* [10;20] couplée à un *RankThres* de 0,50 semble être la zone offrant le meilleur compromis.

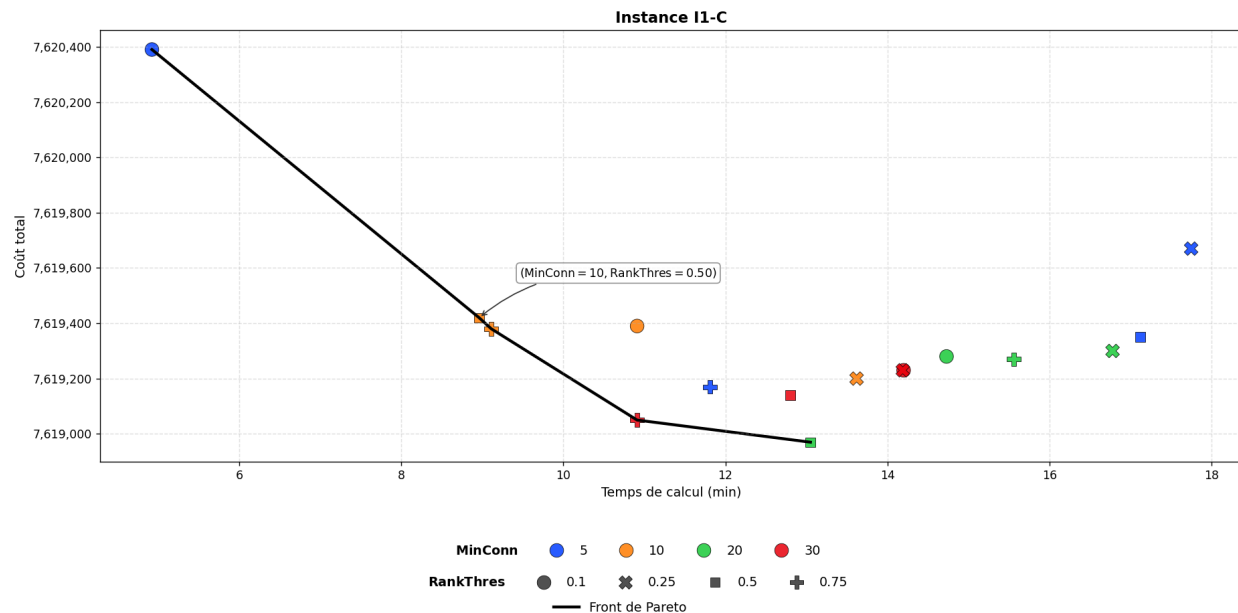


FIGURE 5.5 Front de Pareto pour la moyenne instance

Grandes instances

Pour l'instance I1-B, un paramétrage agressif ($\text{Min Conn} = 10$; $\text{Rank Thres} = 0,1$) permet d'obtenir le temps de résolution minimal (22,97 minutes), au prix d'un coût total légèrement sous-optimal (26 789 200). Par ailleurs, un paramétrage encore plus restrictif met en évidence un point isolé, correspondant à $\text{Min Conn.} = 5$, caractérisé par un coût significativement plus élevé pour un temps de calcul comparable à celui des meilleures solutions (voir Figure 5.6). Ce résultat suggère qu'un nombre insuffisant de connexions limite excessivement l'espace de recherche, conduisant ainsi à des solutions de moindre qualité sans gain notable en termes de temps de calcul. Pour l'instance I2-B, un filtrage moins restrictif des arcs, obtenu en passant d'un $\text{RankThres} = 0,1$ à un $\text{RankThres} = 0,5$, favorise une amélioration de la qualité des solutions.

La configuration ($\text{MinConn} = 10$; $\text{RankThres} = 0,25$) atteint ainsi le temps de calcul (environ 28 minutes), au prix d'un coût minimal (26 595 800), ce qui présente le meilleur compromis. Au-delà de ce seuil, l'augmentation de MinConn n'apporte plus d'amélioration de la fonction objectif et s'accompagne d'une dégradation significative et instable des temps de calcul (jusqu'à environ 72 minutes).

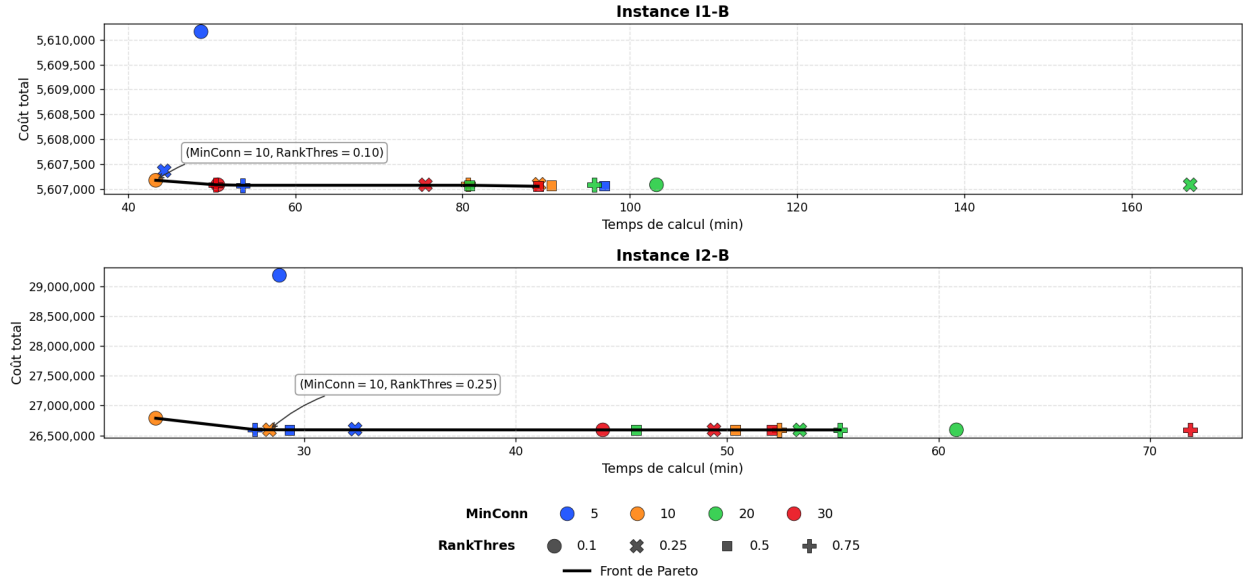


FIGURE 5.6 Front de Pareto pour les grandes instances

L'analyse de sensibilité menée sur l'ensemble des instances met en évidence un compromis entre l'efficacité computationnelle et la qualité des solutions, structuré par les paramètres *MinConn* et *RankThres*. Pour les petites instances, la convergence est quasi instantanée et peu sensible au paramétrage. Le front de Pareto est plat, ce qui indique que l'optimum global est atteint rapidement. En revanche, sur les grandes instances (I1-B et I2-B), le front de Pareto révèle qu'un paramétrage trop restrictif, notamment avec *MinConn* = 5, fragilise la recherche en piégeant l'algorithme dans des optima locaux de mauvaise qualité sans gain de temps significatif. À l'opposé, l'augmentation excessive de ces paramètres au-delà de *MinConn* = 20 sature le temps de calcul pour des gains de coût marginaux, voire nuls. En conclusion, la configuration *MinConn* = 10 associée à un *RankThres* compris entre 0, 10 et 0, 50 apparaît comme une configuration de référence : elle garantit une exploration suffisante de l'espace de recherche pour assurer la stabilité du coût total, tout en maintenant un temps de résolution minimal, offrant ainsi le meilleur équilibre pour la résolution de problèmes d'optimisation de différentes tailles.

5.3.3 Approche de résolution hybride

Nous proposons une stratégie de résolution hybride consistant à initialiser le processus avec un réseau réduit afin d'accélérer la convergence lors des premières itérations, puis à réintroduire progressivement le modèle complet pour devenir moins heuristique et s'approcher de la valeur optimale de la relaxation linéaire, jusqu'à satisfaction des critères d'arrêt. Plus précisément,

une réduction de la taille du réseau est appliquée au niveau du nœud racine, en utilisant des valeurs des paramètres de l’heuristique déterminées à partir de l’analyse de sensibilité préalable. Dans une seconde phase, l’heuristique de réduction est désactivée et une limite sur le nombre maximal de labels $\mathbf{Lab}_{\max}$ est réintroduite, conformément aux conclusions de l’étude précédente. Les résultats obtenus, présentés au tableau 5.10, démontrent l’efficacité de l’approche hybride, en particulier pour les instances I2-C et I1-C. En revanche, les instances I1-A et I2-A n’ont pas montré d’amélioration significative. En comparaison avec les résultats préliminaires, des gains significatifs en temps de calcul sont observés, atteignant une réduction de 47,8 % pour l’instance I2-C et de 93,7 % pour l’instance I1-C, tout en maintenant une qualité de solution très proche de la solution préliminaire (écarts inférieurs à 0,02 %).

Pour les instances de grande taille, des améliorations du temps de résolution sont également observées (jusqu’à 76,1 %), bien qu’elles s’accompagnent d’une légère dégradation du coût, qui reste toutefois limitée et acceptable au regard des gains computationnels obtenus. Ces résultats confirment la pertinence de l’approche hybride, qui permet de concilier efficacement la rapidité de convergence et la qualité des solutions.

TABLEAU 5.10 Performances de l’approche hybride

Instance	$\mathbf{Lab}_{\max}$	$\mathbf{MinConn}$	$\mathbf{RankThres}$	Temps (min)	Coût
<i>Petites Instances</i>					
I2-C	5	10	0,5	0,35 (-47,8 %)	16 523 300 (+0,018 %)
<i>Moyennes Instances</i>					
I1-C	5	20	0,5	0,94 (-93,7 %)	7 619 160 (+0,004 %)
<i>Grandes Instances</i>					
I1-B	20	10	0,1	17,98 (-76,1 %)	5 607 270 (+0,003 %)
I2-B	5	10	0,1	14,61 (-73,1 %)	26 979 300 (+1,45 %)

Afin de réduire les temps de calcul pour les grandes instances, une technique d’arrêt anticipé de la GC a été explorée avec l’approche hybride. Le tableau 5.11 présente l’impact des critères d’arrêt de la GC , définis par $\mathbf{LogItr}$ et $\mathbf{CostDec}$, sur les performances de résolution.

Pour l’instance I1-B, un arrêt après 5 itérations sans amélioration permet d’obtenir un coût total de 5 607 290 en seulement 16,85 minutes. De manière similaire, pour l’instance I2-B, un seuil plus permissif ($\mathbf{LogItr} = 10$) conduit à une solution de coût 26 970 600 en 12,96 minutes. Ces résultats suggèrent que la limitation du nombre d’itérations permet d’éviter des explorations inutiles de l’espace de recherche, contribuant ainsi à une amélioration signi-

ficative de l'efficacité computationnelle. Toutefois, ce mécanisme repose sur un compromis : un arrêt trop précoce pourrait empêcher l'identification de colonnes potentiellement améliorantes, tandis que l'absence de limite pourrait engendrer des temps de résolution excessifs. Globalement, les paramètres retenus offrent un équilibre entre la rapidité d'exécution et la qualité des solutions.

TABLEAU 5.11 Impact de l'arrêt de *GC* pour les grandes instances

Instance	Arrêt de GC		Temps (min)	Coût total
	<i>LogItr</i>	<i>CostDec</i>		
I1-B	5	1	16,85 (-77,6 %)	5 607 290 (+0,04 %)
I2-B	10	1	12,96 (-76,1 %)	26 970 600 (+1,41 %)

5.3.4 Recommandations des paramètres pour chaque catégorie d'instances

Sur la base de l'ensemble des analyses précédentes (effets de ***Lab_{max}***, heuristique de réduction de taille de réseau et critères d'arrêt de la *GC*), il est possible de proposer des recommandations de paramétrage adaptées à chaque catégorie d'instance. Le tableau 5.12 résume cette suggestion de paramétrage. Pour les petites instances, un ***MinConn*** faible combiné à un ***RankThres*** modéré permet de maintenir un équilibre optimal entre temps de calcul et qualité de solution. Pour les instances moyennes, des valeurs intermédiaires de ***MinConn*** et ***RankThres*** ainsi qu'un paramétrage de ***Lab_{max}*** limité à 100 favorisent une exploration suffisante de l'espace de recherche tout en limitant les coûts de calcul. Enfin, pour les grandes instances, l'utilisation d'une approche hybride associée à un arrêt anticipé de la *GC* et à un paramétrage de ***Lab_{max}*** limité à 50 est recommandée afin de réduire les temps de calcul tout en garantissant des solutions proches de l'optimum.

TABLEAU 5.12 Synthèse des paramètres recommandés par catégorie d'instances

Catégorie	Approche hybride	<i>Lab_{max}</i>	<i>MinConn</i>	<i>RankThres</i>	<i>LogItr</i>	<i>CostDec</i>
Petite	✗	✗	[5; 10]	[0, 25; 0, 75]	✗	✗
Moyenne	✓	[5; 100]	[10; 20]	[0, 5; 0, 75]	✗	✗
Grande	✓	[5; 50]	[10; 20]	[0, 1; 0, 5]	[5; 10]	[0; 1]

CHAPITRE 6 CONCLUSION

6.1 Synthèse des travaux

Au cours de cette maîtrise, nous avons eu l’opportunité de proposer une méthode de résolution pour l’*OAMRP*, intégrant explicitement les contraintes de maintenance ainsi qu’une certaine flexibilité dans leur planification. Cette approche permet notamment de prendre en compte des pénalités associées aux maintenances précoces et tardives, afin de mieux refléter les compromis entre exigences opérationnelles et coûts.

Dans un premier temps, le problème a été formulé sous la forme d’un programme linéaire en nombres entiers de type partitionnement d’ensemble. À cette formulation, nous avons intégré des contraintes liées aux exigences de maintenance ainsi que des contraintes de faisabilité opérationnelle, afin de mieux représenter les conditions réelles d’exploitation.

Par la suite, une méthode de résolution basée sur une heuristique de *branch-and-price* a été proposée, dans laquelle la *GC* est intégrée au sein d’un schéma de *branch-and-bound*. Afin de réduire la complexité de l’exploration, une seule branche de l’arbre est explorée à l’aide d’une heuristique de descente en profondeur. Les *SPs* sont formulés comme des *SPPRCs* sur des réseaux de connexions acycliques. La génération de nouvelles routes, à intégrer dans le *PMR* en fonction de leurs coûts réduits, repose sur l’intégration des contraintes de maintenance et des pénalités associées dans les ressources des nœuds ainsi que dans les coûts des arcs. La résolution des *SPs* s’appuie sur une approche de programmation dynamique via un algorithme d’étiquetage, combinée à des règles de dominance visant à limiter le nombre d’étiquettes générées. Après la résolution de la relaxation linéaire, et afin d’obtenir une solution entière, une technique de fixation de colonnes (*fix*) est appliquée pour fixer des routes d’avions et identifier les meilleures variables selon le seuil défini parmi les solutions fractionnaires.

Puis, les expérimentations réalisées sur des instances réelles, fournies par notre partenaire industriel *IBS*, ont permis de valider l’approche proposée. En termes de performance, les petites instances sont résolues en moins d’une minute, les instances de taille moyenne nécessitent environ 15 minutes de calcul, tandis que les grandes instances requièrent entre une et deux heures de résolution. Afin de maîtriser ces temps de calcul, plusieurs techniques d’accélération ont été mises en œuvre, notamment des stratégies de réduction de la taille du réseau, des règles de limitation du nombre d’étiquettes par nœud ainsi que des mécanismes d’arrêt anticipé de *GC*.

Enfin, les techniques d'accélération proposées ont permis de réduire les temps de calcul de manière significative. Ainsi, les petites instances peuvent désormais être résolues en quelques secondes, les instances de taille moyenne en environ une minute, et les grandes instances en près de 15 minutes tout en conservant une qualité de solution satisfaisante avec un écart à la valeur optimale inférieur à 1,5 %.

6.2 Limitations de la solution proposée et améliorations futurs

Le présent travail pourrait être amélioré en cherchant à optimiser davantage les performances computationnelles sur les grandes instances, notamment en intégrant des techniques basées sur l'intelligence artificielle, telles que les techniques d'apprentissage, pour guider la *GC* ou l'exploration de l'arbre de recherche.

De plus, l'*OAMRP* proposé est de nature déterministe. Toutefois, dans un contexte réel, plusieurs éléments sont soumis à des aléas, notamment les horaires des vols, pouvant être affectés par des retards, ainsi que l'annulation éventuelle de certains vols. L'intégration de ces incertitudes dans un cadre stochastique constituerait ainsi une piste de recherche prometteuse. Une telle approche permettrait de concevoir des plans plus robustes et flexibles, capables de mieux s'adapter aux perturbations. Dans un secteur comme l'industrie aérienne, fortement exposé aux imprévus, le développement de modèles stochastiques ou robustes représente une perspective pertinente pour améliorer la fiabilité des solutions proposées.

Ce travail s'est concentré sur l'optimisation de la maintenance opérationnelle à court terme, visant à garantir la navigabilité de la flotte. Cependant, ce cadre pourrait être élargi : une extension intéressante consisterait à enrichir le modèle en intégrant la planification de maintenances préventives sur des avions non critiques. En effet, les solutions obtenues peuvent révéler des périodes de capacité résiduelle au niveau des stations de maintenance. L'exploitation de cette capacité disponible permettrait d'anticiper certaines opérations de maintenance, contribuant ainsi à une meilleure utilisation des ressources et à une réduction des risques de surcharge future. Une telle approche offrirait une vision plus proactive de la gestion de la maintenance en améliorant l'efficacité opérationnelle du système.

RÉFÉRENCES

- [1] C. Lagos, F. Delgado et M. A. Klapp, “Dynamic optimization for airline maintenance operations,” *Transportation Science*, vol. 54, n°. 4, p. 998–1015, 2020.
- [2] IATA, “Airline maintenance cost executive commentary fy2023 data.” 2023. [En ligne]. Disponible : https://www.iata.org/contentassets/8437020db31a4717b70677d9b06b1a45/fy2024-mcx-report_public.pdf
- [3] IATA, “Aircraft operational availability 2nd edition - 2022,” 2022. [En ligne]. Disponible : <https://www.iata.org/contentassets/bf8ca67c8bcd4358b3d004b0d6d0916f/aoa-2nded-2022.pdf>
- [4] Oliver Wyman, “Global fleet and mro market forecast 2023-2033.” 2021. [En ligne]. Disponible : <https://www.oliverwyman.com/our-expertise/insights/2023/feb/global-fleet-and-mro-market-forecast-2023-2033.html>
- [5] C. Senturk et I. Ozkol, “The effects of the use of single task-oriented maintenance concept and more accurate letter check alternatives on the reduction of scheduled maintenance downtime of aircraft,” *International Journal of Mechanical Engineering and Robotics Research*, vol. 7, n°. 2, p. 189–196, 2018.
- [6] G. N. Cook et J. Goodwin, “Airline networks : A comparison of hub-and-spoke and point-to-point systems,” *Journal of Aviation/Aerospace Education & Research*, vol. 17, n°. 2, p. 1, 2008.
- [7] C. Barnhart et B. Smith, *Quantitative Problem Solving Methods in the Airline Industry*, ser. International Series in Operations Research & Management Science. Springer, 2012, vol. 169.
- [8] IATA, “Unveiling the biggest airline costs,” <https://www.iata.org/en/publications/newsletters/iata-knowledge-hub/unveiling-the-biggest-airline-costs/>, June 2024.
- [9] A. Levin, “Scheduling and fleet routing models for transportation systems,” *Transportation Science*, vol. 5, n°. 3, p. 232–255, 1971.
- [10] J. Abara, “Applying integer linear programming to the fleet assignment problem,” *Interfaces*, vol. 19, n°. 4, p. 20–28, 1989.
- [11] C. A. Hane, C. Barnhart, E. L. Johnson, R. E. Marsten, G. L. Nemhauser et G. Sigismondi, “The fleet assignment problem : Solving a large-scale integer program,” *Mathematical Programming*, vol. 70, n°. 1, p. 211–232, 1995.
- [12] G. Desaulniers, J. Desrosiers, Y. Dumas, M. M. Solomon et F. Soumis, “Daily aircraft routing and scheduling,” *Management Science*, vol. 43, n°. 6, p. 841–855, 1997.

- [13] Y. Xu, S. Wandelt et X. Sun, “Airline scheduling optimization : Literature review and a discussion of modelling methodologies,” *Intelligent Transportation Infrastructure*, vol. 3, 2024.
- [14] Federal Aviation Administration, “FAA regulations & policies,” 2023. [En ligne]. Disponible : https://www.faa.gov/regulations_policies/faa_regulations
- [15] C. Sriram et A. Haghani, “An optimization model for aircraft maintenance scheduling and re-assignment,” *Transportation Research Part A : Policy and Practice*, vol. 37, n°. 1, p. 29–48, 2003.
- [16] J. Beliën, B. Cardoen et E. Demeulemeester, “Improving workforce scheduling of aircraft line maintenance at Sabena Technics,” *Interfaces*, vol. 42, n°. 4, p. 352–364, 2012.
- [17] EASA. Regulations. EASA Document Library. [En ligne]. Disponible : <https://www.faa.gov/documentLibrary/>
- [18] U.S. Department of Transportation, Federal Aviation Administration, *Advisory Circular - Air Carrier Maintenance Programs*, FAA, 2012, available from FAA Document Library. [En ligne]. Disponible : https://www.faa.gov/documentlibrary/media/advisory_circular/ac%20120-16f.pdf
- [19] J. Van den Bergh, P. De Bruecker, J. Beliën et J. Peeters, “Aircraft maintenance operations : state of the art,” *HUB Research Paper 2013/09*, 2013.
- [20] R. Gopalan et K. T. Talluri, “The aircraft maintenance routing problem,” *Operations Research*, vol. 46, n°. 2, p. 260–271, 1998.
- [21] N. Papadakos, “Integrated airline scheduling,” *Computers & Operations Research*, vol. 36, n°. 1, p. 176–195, 2009.
- [22] Z. Liang, W. A. Chaovalitwongse, H. C. Huang et E. L. Johnson, “On a new rotation tour network model for aircraft maintenance routing problem,” *Transportation Science*, vol. 45, n°. 1, p. 109–120, 2011.
- [23] M. Başdere et Ü. Bilge, “Operational aircraft maintenance routing problem with remaining time consideration,” *European Journal of Operational Research*, vol. 235, n°. 1, p. 315–328, 2014.
- [24] A. Sarac, R. Batta et C. M. Rump, “A branch-and-price approach for operational aircraft maintenance routing,” *European Journal of Operational Research*, vol. 175, n°. 3, p. 1850–1869, 2006.
- [25] T. A. Feo et J. F. Bard, “Flight scheduling and maintenance base planning,” *Management Science*, vol. 35, n°. 12, p. 1415–1432, 1989.
- [26] N. M. Kabbani et B. W. Patty, “Aircraft routing at american airlines.” dans *proceedings of the agifors symposium*, 1992.

- [27] L. Clarke, E. Johnson, G. Nemhauser et Z. Zhu, “The aircraft rotation problem,” *Annals of Operations Research*, vol. 69, n^o. 0, p. 33–46, 1997.
- [28] M. Haouari, H. D. Sherali, F. Z. Mansour et N. Aissaoui, “Exact approaches for integrated aircraft fleet and routing at tunisair,” *Computational Optimization and Applications*, vol. 49, n^o. 2, p. 213–239, 2011.
- [29] T. Temucin, G. Tuzkaya et O. Vayvay, “Aircraft maintenance routing problem—a literature survey,” *Promet-Traffic&Transportation*, vol. 33, n^o. 4, p. 491–503, 2021.
- [30] S. Ruther, N. Boland, F. G. Engineer et I. Evans, “Integrated aircraft routing, crew pairing, and tail assignment : Branch-and-price with many pricing problems,” *Transportation Science*, vol. 51, n^o. 1, p. 177–195, 2017.
- [31] J. Ruan, Z. Wang, F. T. Chan, S. Patnaik et M. K. Tiwari, “A reinforcement learning-based algorithm for the aircraft maintenance routing problem,” *Expert Systems with Applications*, vol. 169, p. 114399, 2021.
- [32] X. Wen, X. Sun, H.-L. Ma et Y. Sun, “A column generation approach for operational flight scheduling and aircraft maintenance routing,” *Journal of Air Transport Management*, vol. 105, p. 102270, 2022.
- [33] E. E. Yurek, “Combinatorial benders decomposition for the operational aircraft maintenance routing problem,” *Computers & Operations Research*, vol. 164, p. 106545, 2024.
- [34] N. A. Al-Thani, M. B. Ahmed et M. Haouari, “A model and optimization-based heuristic for the operational aircraft maintenance routing problem,” *Transportation Research Part C : Emerging Technologies*, vol. 72, p. 29–44, 2016.
- [35] M. Haouari, S. Shao et H. D. Sherali, “A lifted compact formulation for the daily aircraft maintenance routing problem,” *Transportation Science*, vol. 47, n^o. 4, p. 508–525, 2013.
- [36] N. Safaei et A. K. Jardine, “Aircraft routing with generalized maintenance constraints,” *Omega*, vol. 80, p. 111–122, 2018.
- [37] J. Desrosiers, Y. Dumas, M. M. Solomon et F. Soumis, “Time constrained routing and scheduling,” *Handbooks in operations research and management science*, vol. 8, p. 35–139, 1995.
- [38] G. Desaulniers, J. Desrosiers et M. M. Solomon, *Column generation*. Springer Science & Business Media, 2006, vol. 5.
- [39] G. B. Dantzig et P. Wolfe, “Decomposition principle for linear programs,” *Operations research*, vol. 8, n^o. 1, p. 101–111, 1960.
- [40] P. C. Gilmore et R. E. Gomory, “A linear programming approach to the cutting-stock problem,” *Operations research*, vol. 9, n^o. 6, p. 849–859, 1961.

- [41] J. Hu et E. L. Johnson, “Computational results with a primal–dual subproblem simplex method,” *Operations Research Letters*, vol. 25, n^o. 4, p. 149–157, 1999.
- [42] R. M. Saltzman et H. I. Stern, “The multi-day aircraft maintenance routing problem,” *Journal of Air Transport Management*, vol. 102, p. 102224, 2022. [En ligne]. Disponible : <https://www.sciencedirect.com/science/article/pii/S096969972200045X>
- [43] M. Desrochers et F. Soumis, “A generalized permanent labelling algorithm for the shortest path problem with time windows,” *INFOR : Information Systems and Operational Research*, vol. 26, n^o. 3, p. 191–212, 1988.
- [44] S. Irnich et G. Desaulniers, “Shortest path problems with resource constraints,” dans *Column Generation*, G. Desaulniers, J. Desrosiers et M. M. Solomon, édit. New York, NY : Springer, 2005, ch. 2, p. 33–76.
- [45] C. Joncour, S. Michel, R. Sadykov, D. Sverdlov et F. Vanderbeck, “Column generation based primal heuristics,” *Electronic Notes in Discrete Mathematics*, vol. 36, p. 695–702, 2010.
- [46] R. Sadykov, F. Vanderbeck, A. Pessoa, I. Tahiri et E. Uchoa, “Primal heuristics for branch and price : The assets of diving methods,” *INFORMS Journal on Computing*, vol. 31, n^o. 2, p. 251–267, 2019.
- [47] J. Desrosiers, *GENCOL : Une équipe et un logiciel d’optimisation*. Groupe d’études et de recherche en analyse des décisions, 2010.

ANNEXE A PERFORMANCES DE L'HEURISTIQUE DE RÉDUCTION DE TAILLE DE RÉSEAU

Cette annexe présente en détail les résultats de l'analyse de sensibilité des valeurs des paramètres *MinConn* et *RankThres* sur le coût total et le temps de calcul. Les valeurs mises en gras correspondent à la meilleure combinaison, soit en termes de coût, de temps de calcul ou d'un compromis optimal entre les deux.

TABLEAU A.1 Performances pour l'instance I1-A

Heuristique de réduction		Coût total	Temps (min)
<i>MinConn</i>	<i>RankThres</i>		
5	0,10	1 638 380	0,39
	0,25	1 638 050	0,29
	0,50	1 638 010	0,30
	0,75	1 638 010	0,31
10	0,10	1 638 040	0,44
	0,25	1 638 030	0,31
	0,50	1 638 010	0,27
	0,75	1 638 010	0,36
20	0,10	1 638 010	0,29
	0,25	1 638 010	0,30
	0,50	1 638 010	0,33
	0,75	1 638 010	0,34
30	0,10	1 638 010	0,32
	0,25	1 638 010	0,32
	0,50	1 638 010	0,30
	0,75	1 638 010	0,28

TABLEAU A.2 Performances pour l'instance I2-A

Heuristique de réduction		Coût total	Temps (min)
<i>MinConn</i>	<i>RankThres</i>		
5	0,10	5 369 820	0,25
	0,25	5 343 150	0,24
	0,50	5 343 070	0,33
	0,75	5 348 150	0,30
10	0,10	5 344 100	0,26
	0,25	5 343 070	0,42
	0,50	5 343 070	0,25
	0,75	5 349 030	0,27
20	0,10	5 343 070	0,29
	0,25	5 343 940	0,29
	0,50	5 345 070	0,32
	0,75	5 349 030	0,27
30	0,10	5 349 030	0,27
	0,25	5 349 030	0,27
	0,50	5 349 030	0,27
	0,75	5 349 030	0,27

TABLEAU A.3 Performances pour l'instance I2-C

Heuristique de réduction		Coût total	Temps (min)
<i>MinConn</i>	<i>RankThres</i>		
5	0,10	16 595 800	0,47
	0,25	16 777 300	0,70
	0,50	16 518 900	0,87
	0,75	16 525 400	0,79
10	0,10	16 657 200	1,23
	0,25	16 748 300	0,89
	0,50	16 670 100	0,91
	0,75	16 629 600	0,61
20	0,10	16 520 400	0,87
	0,25	16 651 500	1,08
	0,50	16 779 000	1,07
	0,75	16 519 200	0,88
30	0,10	16 658 300	0,81
	0,25	16 658 300	0,81
	0,50	16 533 100	0,91
	0,75	16 559 100	0,87

TABLEAU A.4 Performances pour l'instance I1-C

Heuristique de réduction de taille		Coût total	Temps (min)
<i>MinConn</i>	<i>RankThres</i>		
5	0,10	7 620 390	4,92
	0,25	7 619 670	17,75
	0,50	7 619 350	17,12
	0,75	7 619 170	11,81
10	0,10	7 619 390	10,91
	0,25	7 619 200	13,62
	0,50	7 619 420	8,96
	0,75	7 619 380	9,11
20	0,10	7 619 280	14,73
	0,25	7 619 300	16,78
	0,50	7 618 970	13,05
	0,75	7 619 270	15,56
30	0,10	7 619 230	14,20
	0,25	7 619 230	14,19
	0,50	7 619 140	12,80
	0,75	7 619 050	10,91

TABLEAU A.5 Performances pour l'instance I1-B

Heuristique de réduction		Coût total	Temps (min)
<i>MinConn</i>	<i>RankThres</i>		
5	0,10	5 610 160	48,72
	0,25	5 607 370	44,32
	0,50	5 607 070	96,96
	0,75	5 607 080	53,72
10	0,10	5 607 180	43,31
	0,25	5 607 100	89,15
	0,50	5 607 080	90,63
	0,75	5 607 100	80,63
20	0,10	5 607 090	103,19
	0,25	5 607 090	167,01
	0,50	5 607 080	80,83
	0,75	5 607 090	95,73
30	0,10	5 607 090	50,73
	0,25	5 607 090	75,56
	0,50	5 607 060	89,03
	0,75	5 607 090	50,48

TABLEAU A.6 Performances pour l'instance I2-B

Heuristique de réduction		Coût total	Temps (min)
<i>MinConn</i>	<i>RankThres</i>		
5	0,10	29 184 200	28,82
	0,25	26 606 200	32,40
	0,50	26 595 100	29,30
	0,75	26 598 100	27,66
10	0,10	26 789 200	22,97
	0,25	26 595 800	28,35
	0,50	26 594 900	50,38
	0,75	26 594 500	52,47
20	0,10	26 594 600	60,85
	0,25	26 593 400	53,44
	0,50	26 593 500	45,70
	0,75	26 593 400	55,35
30	0,10	26 593 800	44,12
	0,25	26 594 000	49,38
	0,50	26 593 500	52,12
	0,75	26 596 200	71,91