

Mise en Place d'un Système d'aide à la Décision Face Aux Opérations Irrégulières¹ Cas de Tassili Airlines

BELALMI Maroua¹ et HAMIDOUCHE Sara¹
Encadrée par : BADJARA Mohamed el Amine¹

Résumé

Dans cet article, nous avons développé un système d'aide à la décision en utilisant une méthodologie d'optimisation basée sur des approches multi-objectifs, la recherche comprend la modélisation mathématique du problème de gestion des opérations irrégulières, ainsi que la mise en œuvre de méthodes heuristique et métaheuristique notamment l'algorithme génétique NSGA-IIⁱ. Notre approche vise à améliorer la gestion des vols imprévus, en augmentant leur nombre dans le programme initial avec le minimum de perturbations des vols déjà programmés. Ce système est conçu pour aider la compagnie aérienne Tassili Airlines à passer d'une gestion empirique à une solution informatisée et optimisée.

Mots clés : *Optimisation multi objectif, l'algorithme de NSGA2, irrégularités, transport aérien, outil d'aide à la décision.*

I. Introduction

Dans le monde du transport aérien, la gestion des opérations irrégulières est un défi où les compagnies aériennes ont un intérêt vital à atténuer les effets des événements perturbateurs et à accélérer la récupération des opérations irrégulières. Il est de plus en plus reconnu dans le domaine aérien que la planification de la robustesse et de la fiabilité des horaires dans l'environnement complexe pourrait être tout aussi importante que la planification pour la minimisation des coûts. Par conséquent, ce travail de recherche se concentre sur l'intégration des vols imprévus dans un programme de vols initial avec un minimum de perturbations afin d'aider la compagnie à passer d'une gestion empirique à une gestion rapide et informatisée. Pour se faire, nous utilisons la modélisation mathématique, la métaheuristique NSGA-II et le langage de programmation Python avec ses différentes bibliothèques pour implémenter une interface simple et facile à utiliser pour les utilisateurs de la compagnie aérienne Tassili Airlines.

II. Algorithme génétique

Les algorithmes génétiques appartiennent à la famille des algorithmes évolutionnaires. Leur but est d'obtenir une approximation de la solution à un problème NP-complet par un mécanisme d'optimisation. Ils utilisent la notion de sélection naturelle développée au XIX^e siècle par Darwin et l'appliquent à une population de solutions potentielles au problème donné.



Figure 1 : Chromosome DNA

(Source: <https://www.istockphoto.com/>)

¹ Département de Génie Logistique et transport, Ecole Nationale Supérieure des Technologies Avancées (ENSTA), Dergana, Algérie.

Les algorithmes génétiques utilisent des structures appelées chromosomes pour représenter des solutions potentielles, ou chaque chromosome est composé de gènes. Les gènes codent les caractéristiques de la solution, et les algorithmes génétiques appliquent des opérations comme la sélection, le croisement et la mutation. [1]

Pour résoudre ces problèmes, nous utilisons des métaheuristiques et des heuristiques afin de trouver les solutions. Sur la figure 2, un exemple graphique où l'heuristique explore l'espace des solutions afin de converger vers la meilleure.

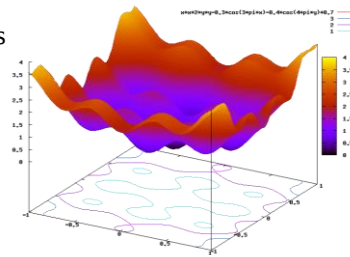


Figure 2 : exemple : heuristique où la meilleure solution est la valeur la plus proche de 4

(source : <https://igm.univ-gustave-eiffel.fr/>)

III. Evolution :

Dans les années 1960, John Holland de l'université du Michigan a initié des recherches sur ce qui deviendrait les algorithmes génétiques, culminant en 1975 avec la publication d'*Adaptation in Natural and Artificial Systems* [2]. Holland visait à mieux comprendre les processus naturels d'adaptation et à créer des systèmes artificiels aux propriétés similaires. Il postulait que le pool génétique d'une population contenait potentiellement de meilleures solutions à un problème adaptatif, solutions qui pouvaient s'exprimer qu'à travers la recombinaison génétique durant la reproduction. Plutôt que de se focaliser uniquement sur les mutations comme source d'évaluation, Holland a mis en avant l'importance de croisement (crossover) génétiques, permettant ainsi de rapprocher les solutions potentielles existantes de l'optimum. [3]

IV. Quelques travaux en relation avec NSGA-II et le domaine aérien

Michał Weiszer et al (2013) [4] ont proposé un modèle d'optimisation, combinant la planification des pistes et le problème des mouvements au sol afin de minimiser le temps et le carburant. Pour le résoudre, ils ont utilisé l'algorithme évolutif NSGA-II qui est basé sur une distance d'encombrement modifiée et une relation de surclassement qui prend en compte le coût du retard et le prix du carburant.

Morteza Khakzar Bafreui et al (2016) [5] ont supposé que le temps de traitement de compression/expansion possible d'un vol peut être contrôlé en continu. Sur la base de cette hypothèse ils ont proposé un modèle qui simultanément (1) minimise le coût total du retard, de l'avance, du retard et de la compression ainsi que les coûts d'expansion du temps de traitement des tâches, et (2) minimise la surpopulation des passagers aux portes d'embarquement et en raison de la complexité du problème ils ont appliqué NSGA-II avec un autre algorithme de résolution. D'après les résultats ils ont constaté qu'il donne une meilleure convergence près du véritable front Pareto-optimal.

Jia huan CHEN et al (2019) [6] ont proposé un modèle d'optimisation du temps de vol à court terme multi-objectifs basé sur les conditions météorologiques, et ils ont utilisé l'algorithme NSGA-II pour le résoudre. Le modèle prend en compte l'impact des conditions météorologiques et combine des stratégies d'attente au sol et dans les airs pour assurer une gestion tactique du flux de trafic aérien et réduire le temps de retard des vols et les coûts engendrés.

Baradaran et al. (2019) [7] ont proposé un modèle mathématique multi-objectif pour le problème de planification de l'équipage polyvalent des compagnies aériennes. Le modèle contient les deux fonctions objectives qui sont : maximiser le nombre de jours de congés prévus en fonction des jours annoncés par l'équipage de conduite, et minimiser les coûts de pénalité associés à la violation des heures de travail. Ils ont utilisé 2 méta-

heuristiques, NSGA-II et MODEⁱⁱ. Ces algorithmes sont comparés sur la base de plusieurs mesures de performance multi-objectifs.

Zhang et al. (2021) [8] ont proposé un nouvel algorithme d'optimisation multi objectif, MOGWOⁱⁱⁱ inspiré de du comportement social des loups gris, et l'ont comparé à NSGA-II pour la gestion des problèmes d'atterrissage d'avions. L'objectif est de minimiser le temps total d'attente des avions et le nombre de croisement de trajectoires.

Jiang et al. (2022) [9] se sont concentrés sur l'optimisation des affectations de portes d'embarquement. Ils ont proposé une approche basée sur NSGA-II afin d'atteindre 2 fonctions objectifs : minimisation des couts et maximisation de l'équité. Les solutions du problème AGAP^{iv} ont été représentées par des chromosomes. Chaque chromosome est une affectation spécifique.

Ming Wei et al (2022) [10] ont travaillé sur l'optimisation de la planification des véhicules de transport de bagages dans les aéroports. Ils ont proposé DEGA^v qui utilise la sélection, le croisement et la mutation, avec une évaluation qui consiste à minimiser le temps de trajet et minimiser le temps d'attente des passagers. Les problèmes rencontrés ont été dans la collecte des données et le tuning des paramètres de la NSGA2.

Shufan et al. (2023) [11] ont comparé l'efficacité de NSGA-II à d'autres méthodes pour la planification des créneaux horaires d'aéroport. Ils ont comparé entre la programmation par contraintes (CP) et l'algorithme génétique mono objectif. La NSGA-II à utiliser 2 fonctions objectifs : minimisation du nombre de conflits de créneaux horaire et la maximisation de l'utilisation des ressources. Les résultats ont démontré que NSGA-II surpassait les autres méthodes en termes de qualité de solution.

Wanru Gao et al (2023) [12] se sont concentrés sur la planification des chariots à bagages dans les aéroports. L'objectif est de créer un modelé qui optimise l'utilisation des chariots a bagages en tenant compte des besoins en transport des bagages et des horaires de vols. Dans leur article ils ont proposé un algorithme d'optimisation multi-objectif NSGAII combiné avec une métrique d'hypervolume de contribution. Cet algorithme permet aux décideurs d'ajuster les poids des fonctions objectives pendant le processus d'optimisation afin de le guider pour trouver des solutions adaptées au monde réel.

V. Opération irrégulières : récupération et robuste

D'après Barnhart, les plans de récupération d'horaires visent à atteindre un ou plusieurs objectifs possibles, tels que : minimiser les couts de récupération des passagers et minimiser le temps nécessaire pour répandre le plan initial. Quel que soit l'objectif, le problème de récupération doit souvent être résolu en quelques minutes, si non la solution peut devenir irréalisable. Les options de récupération d'horaires en cas d'opérations irrégulières :

- Retarder les départs des vols.
- Réacheminer ou échanger des avions.
- Annuler les segments de vols.
- Affecter un équipage nouveau ou réaffecter l'équipage existant.
- Reporter les heures de départ des segments de vol pour éviter que les passagers en correspondance ne manquent leurs connexions.

Après des couts économiques directs et indirects de plusieurs milliards de dollars par an, l'industrie aérienne doit atténuer les effets des événements perturbateurs et accélérer la récupération des opérations irrégulières. Ce travail est encore à ses débuts, malgré des progrès en recherche et en soutien matériel des données, Les outils d'aide à la décision basé sur l'optimisation pour une récupération rapide ne sont qu'à un stade précoce de mise en œuvre dans les grandes compagnies aériennes, représentant un défi futur mais crucial. [13]

VI. Etude de cas

Un cas réel de la compagnie aérienne Tassili Airlines, qui a un problème de récupération de programme initial, dans le cas d'intégration des vols imprévus, elle a adapté une approche empirique basé sur l'expérience des gestionnaires pour programmer ces vols. Tassili choisit d'annuler les vols initialement programmés et les remplacer avec les nouveaux vols, ou bien reporter la programmation des vols imprévus au J+1, J+2 voire J+3.[14]

a) Problématique

Dans le cadre de la mise en place d'un programme de vols qui satisfait à la fois les exigences de la compagnie aérienne et les attentes des clients, il est crucial de repenser les méthodes traditionnelles de gestion des vols imprévus. Actuellement, ces vols représentent des opérations irrégulières, gérées de manière empirique et souvent manuellement, ce qui entraîne des inefficacités et des délais dans la prise de décision. Une dépendance excessive à l'égard de l'expérience individuelle des gestionnaires peut limiter la capacité de la compagnie à mettre en œuvre des solutions innovantes et à tirer parti des meilleures pratiques de l'industrie. Pour répondre à ces défis, nous envisageons de développer une solution digitale innovante capable d'optimiser la gestion des vols imprévus tout en récupérant les vols initialement programmés.

b) Modélisation mathématique

- Notation des indices :

Table 1: Indices du modèle et leurs définitions

Indices	Définitions
i	Indice des vols
j	Indice des périodes dans une journée
k	Nombre de vols imprévus à programmer pour le jour J+1
n	Nombre de vols imprévus à programmer pour le jour J+1
m	Nombre de vols déjà programmés pour le jour J+1

- Notation des paramètres :

Table 2 : Paramètres du modèle et leurs définitions

Paramètre	Définition
T_i	Tolérance maximale d'un vol i donné
$Y_{ij} =$	$\begin{cases} 1 & \text{si le vol } i \text{ déjà programmé demmare a l'instant } j \\ 0 & \text{si non} \end{cases}$

- Notation des ensembles :

Table 3: Ensembles du modèle et leurs définitions

Ensemble	Définition
I_I	Ensemble des périodes interdites pour le décollage du vol i, $\forall i \in \{1, \dots, n+m\}$
J	Ensemble des instants dans une journée $J \in \{1, \dots, 1440\}$
I	Ensemble des vols $I \in \{1, 2, \dots, n+1, \dots, n+m\}$

- Définition de variable de décision :

$$X_{ij} = \begin{cases} 1 & \text{si le vol imprévu ou déjà programmé } i \text{ est programmé à l'instant } j \text{ pour le jour } J + 1 \\ 0 & \text{sinon} \end{cases}$$

$$\forall i = \overline{1, (m + n)}, \quad j = \overline{1, 1440}$$

e) Définition des hypothèses

- Pour chaque vol imprévu à programmer, on considère que l'avion et le personnel navigant sont disponibles
- Le modèle mathématique est fixé pour une journée et pour un seul aéroport de départ
- Le temps de préparation d'un vol au départ est fixé à 30 minutes pour les avions de type Q400 et Q200 et à 1 heure pour les avions de type B780.

f) Définition du modèle mathématique :

$$\text{Max } Z_1 = \sum_{i=1}^n \sum_{j=1}^{1440} x_{ij} \quad (1)$$

$$\text{Min } Z_2 = \sum_{i=1}^m \sum_{j=1}^{1440} |x_{ij} - y_{ij}| \quad (2)$$

S. c :

$$\sum_{j=1}^{1440} x_{ij} = 1 ; \quad \forall i \in \{n + 1, \dots, n + m\} \quad (3)$$

$$\sum_{j=1}^{1440} x_{ij} \leq 1 ; \quad \forall i \in \{1, \dots, n\} \quad (4)$$

$$\sum_{j \in I_i} x_{ij} = 0 ; \quad \forall i \in \{n + 1, \dots, n + m\} \quad (5)$$

$$y_{ik} \cdot k \cdot x_{ij} - k y_{ik} \leq T_i ; \quad \forall i \in \{n + 1, \dots, n + m\} ; \forall j \geq \sum_{k=1}^{1440} k y_{ik} \quad (6)$$

$$x_{ij} = 0 ; \quad \forall i \in \{n + 1, \dots, n + m\} ; \forall j < \sum_{k=1}^{1440} k y_{ik} \quad (7)$$

Dans la formulation ci-dessus, la première fonction objective (1) vise à maximiser le nombre de vols imprévus à programmer tandis que la deuxième (2) vise à minimiser les perturbations des vols programmés. La contrainte (3) s'assure que tous les vols déjà programmés doivent être reprogrammés, la contrainte (4) confirme tous les vols imprévus sont au plus programmé une fois, la contrainte (5) s'assure que pour chaque vol a un ensemble de départ interdits pour son décollage, la contrainte (6) confirme qu'on ne doit pas dépasser la Tolérance d'un vol déjà programmé s'il est perturbé, et la contrainte (7) confirme qu'on ne doit pas programmer un vol déjà programmé par rapport à son instant déjà programmé.

g) Description de l'approche proposée :

Notre problème est un problème d'optimisation multi objectif, donc nous avons opté pour l'utilisation d'une méthode de résolution approchée basée sur l'emploi de la méta heuristique NSGA-II qui se caractérise par ses opérateurs de variations (la sélection par tournoi, le croisement binaire simulé et la mutation polynomiale)

De ce fait nous avons implémenté une heuristique pour initialiser la population. Et puis, nous avons appliqué les instructions de la méta-heuristique NSGA-II pour générer les solutions possibles de notre problème.

L'heuristique constructive de l'initialisation fonctionne en demandant des slots de la part de l'aéroport pour l'ensemble des vols. Si des vols imprévus surviennent, ils sont intégrés dans les créneaux existants. Dans le cas où les créneaux sont insuffisants pour accueillir ces derniers, l'heuristique consiste à perturber les vols déjà programmés afin de les accommoder

Algorithm 1: Initialisation de la population

Input: Liste des vols programmés VP , Liste des vols imprévus VI , Liste des slots libres SL

Output: Population initiale P

```

P ← liste vide;
for Vi in VI do
    slot_trouve ← faux;
    for slot in SL do
        if intervalle de départ de Vi est compatible avec intervalle de slot then
            Vi.heure_depart ← choisir.heure_depart(Vi, slot);
            P.append(VP + [Vi]);
            slot_trouve ← vrai;
            break;
        end
    end
end
if non slot_trouve then
    for Vp in VP do
        if Vp.type_avion == "ATR" then
            nouvelle.heure_depart ← Vp.heure_depart + 30 minutes;
        end
        else if Vp.type_avion == "Boeing737" then
            nouvelle.heure_depart ← Vp.heure_depart + 60 minutes;
        end
        if nouvelle.heure_depart est dans l'intervalle de Vp.slot_programme then
            solution ← copie.solution_actuelle(VP);
            solution[Vp].heure_depart ← nouvelle.heure_depart;
            solution[Vi].heure_depart ← nouvelle.heure_depart;
            P.append(solution);
        end
    end
end
end
return P;

```

h) Quelques es vols programmés et imprévus avec leurs intervalles de départ :

ID	Intervalle.Départ	Avion	Periode interdite
10001	[1030-1140]	B738	[(0,959) , (1081 ,1439)]
10002	[1080-1220]	B738	[(0, 999) , (1101, 1439)]
10003	[1095-1170]	B738	[(0, 1094) , (1216,1439)]
10004	[1140-1280]	B738	[(0,1139) , (1261, 1439)]
10005	[960,1005]	Q400	[(0,959),(1125,1439)]
10006	[1080,122]	B738	[(0,1097),(1340,1439)]
10007	[960,1080]	B738	[(0,1064),(1066,1439)]
10008	[1000,1080]	B738	[(0,1074),(1076,1439)]
10009	[1140,1260]	Q400	[0,1139),(1380,1439)]
10010	[480,690]	B738	[(0,479),(810,1439)]
10011	[480,645]	B738	[(0,479),(765,1439)]

Vol	ID	Instant	Avion	Intervalle de départ
ALG - HME	1	310	B738	[250, 505]
ALG - HRM	2	480	Q400	[450, 675]
ALG - CDG	3	495	B738	[435, 825]
ALG - HME	4	495	B738	[435, 690]
ALG - HME	5	525	B738	[465, 720]
ALG - AZR	6	550	B738	[490, 780]
ALG - DAEN	7	705	Q400	[675, 925]
ALG - CZL	8	780	B738	[720, 945]
ALG - ORN	9	945	B738	[885, 1125]
ALG - CDG	10	955	B738	[895, 1285]
ALG - IAM	11	960	Q200	[930, 1245]

VI. Présentation du logiciel

« OPT-Flight » est un logiciel destiné à la branche exploitation sol de la compagnie aérienne «TAL^{vi}», afin de planifier son programme de vol en cas de vols imprévus. La planification suggérée par ce logiciel est basée sur l'approche de résolution présentée ci-dessus. Dans le but de :

- Minimiser les perturbations des vols programmés.
- Maximiser le nombre de vols imprévus à programmés.
- Optimiser le programme de vols final.

Fenêtre d'accueil :

Après le lancement du logiciel, une fenêtre d'accueil Va apparaître, elle contient un fond correspondant à Notre thème et un bouton « commencer » Pour le démarrage du logiciel.

Fenêtre d'identification :

L'utilisateur doit entrer ses données (nom, email, et mot de passe) pour des raisons de sécurité et traçabilité.

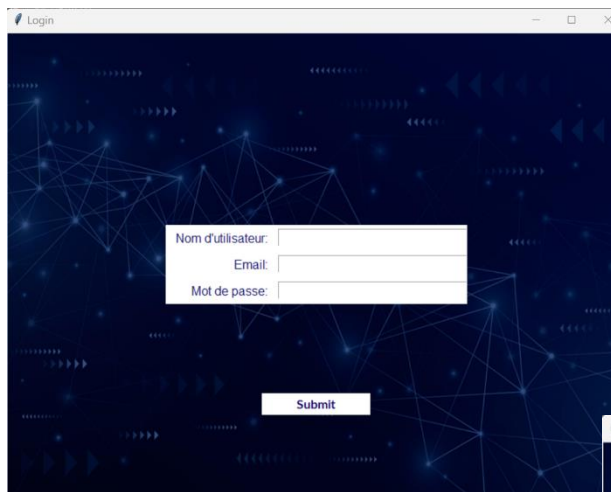


Figure 4: Fenêtre d'identification

Fenêtre de sélection du fichier Excel :

L'utilisateur doit entrer le fichier Excel qui contient les informations sur les vols programmés et les vols imprévus ainsi que les données sur les slots demandés

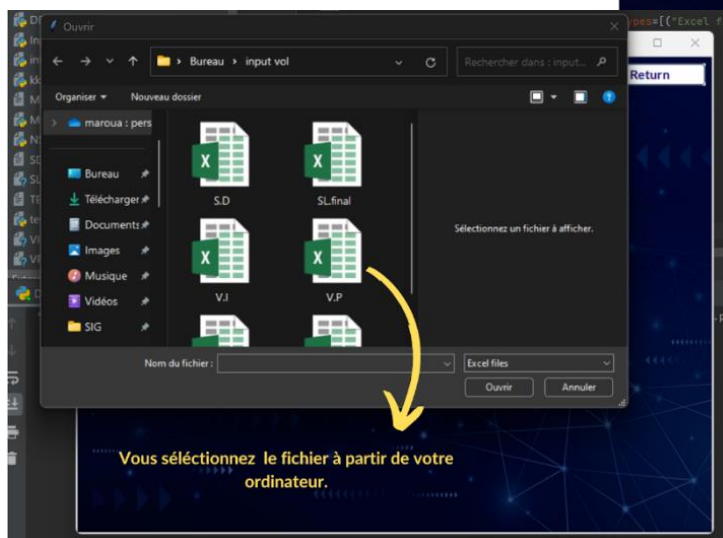


Figure 6 : Sélection des données

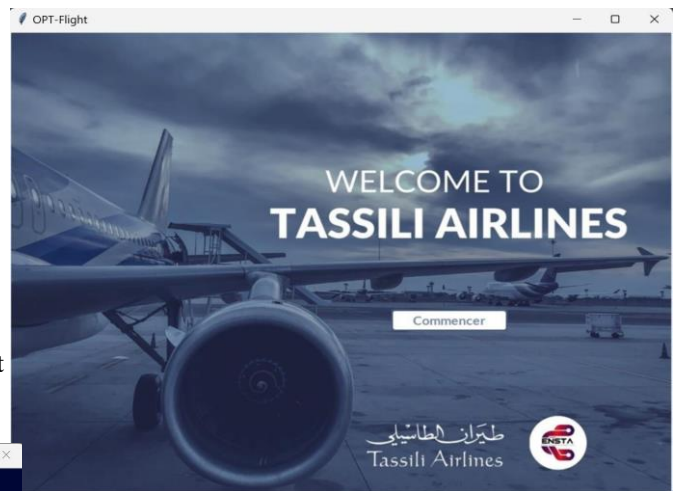


Figure 3 : Fenêtre d'accueil

Fenêtre d'opérations :

En cliquant sur le bouton « submit » on y aperçus deux boutons : «Return» qui permet de retourner à la fenêtre précédente et « chercher » qui permet de saisir le fichier Excel qui contient les informations nécessaires pour le fonctionnement du logiciel.

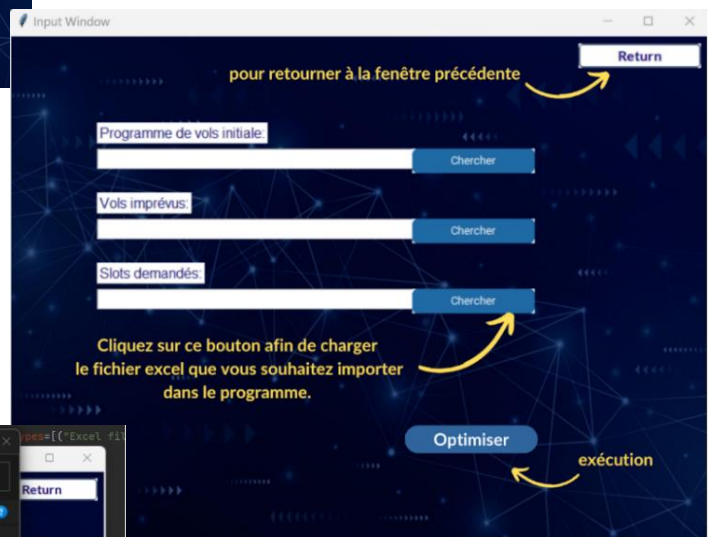


Figure 5 : Fenêtre d'opérations

VI. Exécution de l'algorithme NSGA-II :

Évaluation de la population : l'évaluation se fait par l'emploi de la procédure de ranking et de la distance de crowding, tel que le ranking représente le tri non dominé et permet de classer les individus d'une population selon la notion de front tandis que la distance de crowding permet de comparer les individus qui ont le même rang et choisit la solution à la distance la plus élevée.

Algorithm : Évaluation de la population dans NSGA-II

Entrées: P de taille N , n , m , x_{ij} et y_{ij}

Sortie : P avec les valeurs $Z1$ et $Z2$

```

1 for chaque individu  $p$  dans  $P$  do
2   Calculer  $Z1(p) = \sum_{i=1}^n \sum_{j=1}^{1440} x_{ij} \cdot p[i, j]$ 
3   Calculer  $Z2(p) = \sum_{i=1}^m \sum_{j=1}^{1440} |x_{ij} - y_{ij}| \cdot p[i, j]$ 
4   Stocker  $Z1(p)$  et  $Z2(p)$  comme attributs de l'individu  $p$ 
5 end
```

Sélection des parents par tournoi :

Algorithm : Sélection des parents par tournoi

Input: population de taille N , taille_tournoi T

Output: parents sélectionnés

```

1  $parents \leftarrow []$ ;
2  $N \leftarrow$  taille de la population;
3 while longueur( $parents$ ) <  $N$  do
4    $candidates\_tournoi \leftarrow$  échantillon aléatoire de  $P$  de taille  $T$ ;
5    $meilleur\_candidat \leftarrow None$ ;
6   for candidat dans  $candidates\_tournoi$  do
7     if  $meilleur\_candidat = None$  ou
       ( $EvaluerZ1(candidat) < EvaluerZ1(meilleur\_candidat)$ )
8     then
9        $meilleur\_candidat \leftarrow candidat$ ;
10    end
11  else if  $EvaluerZ1(candidat) = EvaluerZ1(meilleur\_candidat)$ 
12    then
13      if  $EvaluerZ2(candidat) > EvaluerZ2(meilleur\_candidat)$ 
14        then
15           $meilleur\_candidat \leftarrow candidat$ ;
16      end
17    end
18  end
19  Ajouter  $meilleur\_candidat$  à  $parents$ ;
20 end
21 return  $parents$ ;
```

Croisement et mutation : Ces opérations sont appliquées à la population sélectionnée pour créer une nouvelle génération de solutions. De plus, ils introduisent de la variation et explorent de nouvelles régions de l'espace de solutions.

Croisement binaire simulé :

Dans le croisement binaire simulé les individus parents sont échangés selon une probabilité p_c pour former deux nouveaux chromosomes ou solutions enfants. Ces enfants sont calculés à l'aide d'une formule qui prend en compte la distribution des valeurs des parents.

Les solutions sont sélectionnées pour former une nouvelle population en utilisant la sélection dite par tournoi qui n'utilise que des comparaisons entre les individus.

Les individus sont comparés deux à deux et celui qui a la valeur de fitness la plus élevée emporte le tournoi

Algorithm : Croisement SBX

Require: $P1, P2$, $P_c = 0.6$, $\eta_c = 2$

Output : $Y1, Y2$

```

1  $Y1 \leftarrow$  copie de  $P1$ ;
2  $Y2 \leftarrow$  copie de  $P2$ ;
3 for chaque variable  $i$  dans  $P1$  et  $P2$  do
4    $x1, x2 \leftarrow \min(P1[i], P2[i]), \max(P1[i], P2[i])$ ;
5    $\beta \leftarrow 1.0 + (2.0 \cdot (x1 - 0.0) / (x2 - x1))$ ;
6    $\alpha \leftarrow 2.0 - (\beta \cdot (\eta_c + 1))$ ;
7    $rand \leftarrow \text{random}()$ ;
8   if  $rand \leq 1.0 - \alpha$  then
9      $\beta_q \leftarrow (rand \cdot \alpha)^{\frac{1.0}{\eta_c + 1}}$ ;
10  end
11  else
12     $\beta_q \leftarrow (1.0 / (2.0 - rand \cdot \alpha))^{\frac{1.0}{\eta_c + 1}}$ ;
13  end
14   $c1 \leftarrow 0.5 \cdot ((x1 + x2) - \beta_q \cdot (x2 - x1))$ ;
15   $c2 \leftarrow 0.5 \cdot ((x1 + x2) + \beta_q \cdot (x2 - x1))$ ;
16   $Y1[i] \leftarrow \min(\max(c1, 0.0), 1.0)$ ;
17   $Y2[i] \leftarrow \min(\max(c2, 0.0), 1.0)$ ;
18 end
19 return  $Y1, Y2$ ;
```

Algorithm Mutation polynomiale

Require: $Y'_1, Y'_2, P_m = 20, n_m = 20$

Ensure: Y'_1, Y'_2

```

1: Pour chaque enfant  $Y$  obtenus après le croisement SBX
2: choisir un paramètre de mutation  $n_m = 20$ 
3: for chaque dimension (instant de départ de chaque vol) do
4:   Générer une perturbation aléatoire  $\delta$  selon une distribution normale
     centrée en zéro et d'écart type  $n_m = 20$ 
5:   Appliquer la mutation  $Y'_i = Y_i + \delta$ 
6:   Assurer que la nouvelle valeur générée reste dans les limites permises
     pour les instants de départ des vols
7: end for
8: return  $Y'_1, Y'_2$ 
```

Mutation polynomiale :

Dans la mutation, les solutions sont perturbées selon une distribution normale pour obtenir des valeurs de départ de vol

Sélection pour la nouvelle génération :

La nouvelle génération de solutions est formée en combinant la population actuelle et la nouvelle population générée, suivie d'une nouvelle évaluation et tri non dominé.

Algorithm	Sélection des individus pour la nouvelle génération
Require:	P1, P2: Les parents sélectionnés
Require:	Y1, Y2: Les enfants générés après croisement et mutation
Ensure:	new_population
1:	Combine P1, P2, Y1, Y2 pour former une population combinée
2:	combined_population $\leftarrow$ [P1, P2, Y1, Y2]
3:	Trier combined_population selon Z1 (décroissant) et Z2 (croissant) pour attribuer les rangs
4:	Calculer les distances de crowding
5:	Sélectionner les meilleurs individus
6:	Retourner new_population

Génération du fichier final : Le programme génère un fichier Excel qui contient les informations nécessaires pour la mise en œuvre du plan de vol, y compris les horaires des vols et les créneaux affectés

ID Vol	Heure de Départ	Avion	ID Vol.Imprévu	Heure de Départ	Avion
1	08:00	B737	24	09:00	B737
2	09:30	Q400	25	10:45	Q400
3	11:15	B737	26	12:30	B737
4	13:00	Q200	27	14:15	Q200
5	14:45	Q400	28	16:00	B737
6	16:30	Q400	29	17:45	Q400
7	18:15	B737	30	19:30	B737
8	20:00	Q400			
9	21:45	B737			
10	23:30	Q200			
11	01:15	B737			
12	03:00	Q400			
13	04:45	B737			
14	06:30	Q200			
15	08:15	B737			
16	10:00	Q400			
17	11:45	B737			
18	13:30	B737			
19	15:15	B737			
20	17:00	Q400			
21	18:45	B737			
22	20:30	Q200			
23	22:15	B737			

Figure 7 : Output Flight-Opt

Les vols de 1 jusqu'à 23 se sont des vols déjà programmés, les vols : **3, 6, 7, 17 et 18** ont été perturbés (indiqués en gras) pour libérer de l'espace afin d'intégrer les vols imprévus au programme initiale. Les vols de 24 jusqu'à 30 ce sont des vols imprévus, ils ont été efficacement intégrés avec des heures de départs.

VII. Analyse des résultats :

On a simulé pour une journée de vols afin de voir l'optimalité à petite échelle. Le même principe s'applique à une simulation de plusieurs journées avec des petits changements de paramétrage. Les vols sont en fonction de leurs heures de départ et la durée des vols.

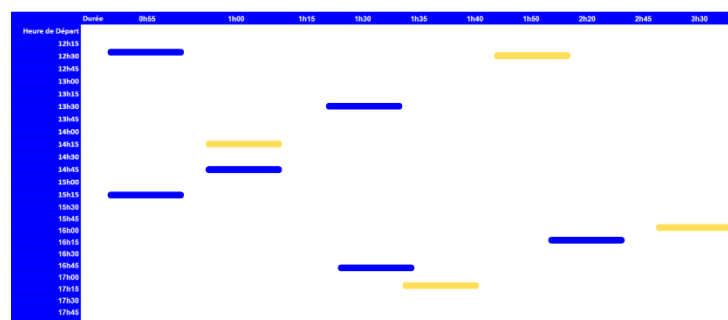


Figure 9 : Graph d'intervalle des vols de l'après midi

Evaluation des fonctions objectives :

Au départ, il y avait 23 vols prévus pour la journée au départ de l'aéroport d'Alger. Chaque vol avait un horaire de départ dans le programme initial des vols. Après l'ajout d'une nouvelle liste des vols imprévus, une nouvelle planification a été établie. L'une des fonctions objectives de notre modèle mathématique est de maximiser le nombre de vols imprévus à programmés. Ce graphique illustre la différence entre le nombre de vols avant et après l'intégration des nouveaux vols programmés :

On remarque une augmentation de 7 nouveaux vols intégrés au programme initial dans la même journée, ce qui révèle une récupération très bénéfique pour la compagnie aérienne. Ce résultat démontre la capacité de la compagnie à s'adapter rapidement aux changements et à saisir les opportunités pour répondre efficacement à la demande croissante au marché aérien.

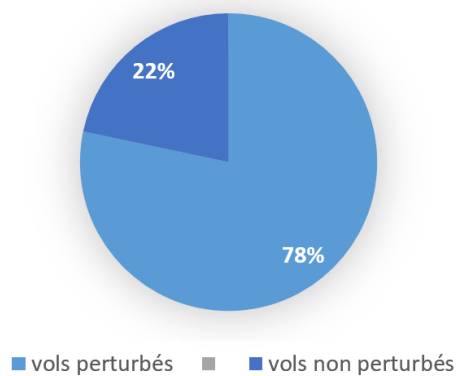


Figure 11 : Proportion de vols perturbés VS non perturbés

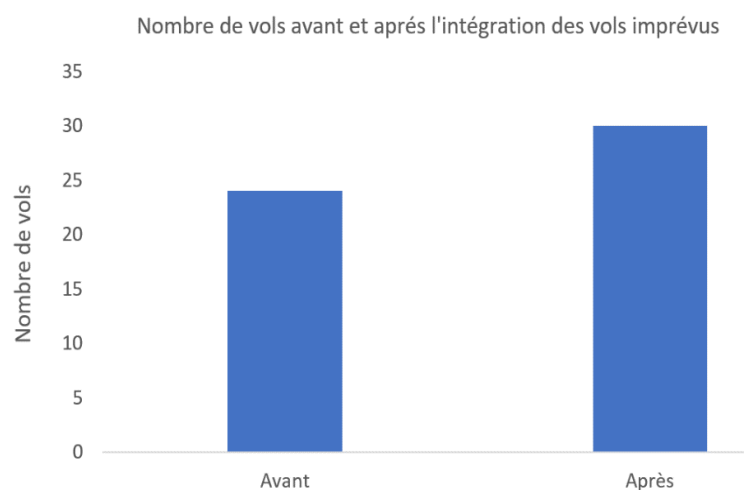


Figure 10 : Graphique du nombre de vols imprévus avant et après l'intégration des vols imprévus

Par rapport à la 2eme fonction objectif, qui consiste à minimiser les perturbations au niveau du programme de vol initial. On remarque dans ce diagramme en secteur que la proportion de vols perturbés est de 21.7% qui représente 5 vols, qui est très minimal par rapport au proportion des vols déjà programmés qui ont gardé leurs départs initiaux.

VIII. Conclusion

En suivant les étapes de l'algorithme NSGA-II pour résoudre le problème des irrégularités dans la compagnie aérienne Tassili Airlines et en utilisant le modèle mathématique qu'on a formulé, nous avons pu proposer un outil d'aide à la décision qui optimise le programme de vols de la compagnie. Notre recherche a permis de démontrer que l'approche multi-objectifs que nous avons mis en œuvre est efficace pour le problème traité. Et le choix de ces techniques repose sur une base scientifique solide.

VI. Références :

[1] Université Gustave Eiffel, Institut d'électronique et d'Informatique Gaspard Monge. (s.d). *Fonctionnement – Algorithmes génétiques*. Consulté le 25 juin 2024, à l'adresse

https://igm.univ-mlv.fr/~dr/XPOSE2013/tleroux_genetic_algorithm/fonctionnement.html

[2] Holland, J. H. (1975). *Adaptation in natural and artificial systems*. Ann Arbor: The University of Michigan Press.

- [3] Rennard, J.-P. (2000, avril). Genetic Algorithm Viewer : Démonstration d'un algorithme génétique.
- [4] Weiszer, M., Chen, J., & Stewart, P. (2013). Preference-based evolutionary algorithm for airport runway scheduling and ground movement optimization.
- [5] Khakzar, M. Bafrueia, S., Khatibi, K., Rahmanic, M. (2016). A Bi-Objective Airport Gate Scheduling with Controllable Processing Times Using Harmony Search and NSGA-II Algorithm 11(1)77_90.
- [6] Jun Bi ,Cong Ding ,Dongfan Xie , Yanhua Li .(2022). Multiobjective Optimization of Airport Ferry Vehicle à l'adresse: <https://doi.org/10.1155/2022/7388406>
- [7] Baradaran, V., & Hosseini, A. H. (2021). A multi-objective mathematical formulation for the airline crew scheduling problem: MODE and NSGA-II solution approaches. *Journal of Industrial Management Perspective*, 11(1, Spring 2021), 247-269.
- [8] Zhang, Y., Lui, X., & Wang, J. (2021). Un algorithme d'optimisation multi-objectif du loup gris pour le problème d'atterrissage d'avions. *Recherche et applications en transport*, 60(3),57-74.
- [9] Jiang, Y., Liu, X., & Sun, X. (2022). Optimisation d'un problème d'assignation de portes d'aéroport multi-objectif : prise en compte de l'équité entre les compagnies aériennes. *Recherche et applications en transport*, 61(2), 29-45.
- [10] Wei, M., Yang, C., & Liu, T. (2022).An Integrated Multi-Objective Optimization for Dynamic Airport Shuttle Bus Location, Route Design and Departure Frequency Setting Problem,19, 14469.a l'adresse :<https://doi.org/10.3390/ijerph192114469>
- [11] Li, S., Liu, Y., & Liu, X. (2023). Optimisation du problème de planification des créneaux horaires d'aéroport & l'aide d'algorithmiques d'optimisation. *Recherche et applications en transport*, 61(2),29-25.
- [12] Gao, W., Li, J., Xu, M. (2023). Interactive Multi objective Optimization of Airport Baggage Trolley Scheduling Based on NSGAII.
- [13] Belobaba, P., Barnhart, C., & Odani, O. (Éd.). (2021). *Opérations irrégulières : récupération et robustesse*. Dans *The Global Airline Industry* (pp. 233-259). Wiley.
- [14] Document de la compagnie, MANEX A de Tassili Airlines (GÉNÉRALITÉS ET FONDEMENTS). (EDITION 2)2009,(révision 37)2023.

ⁱ Non Dominated Sorting Genetic Algorithm.

ⁱⁱ MultiObjectif Differential Evolution.

ⁱⁱⁱ Multi-objective Grey Wolf Optimizer.

^{iv} Algorithme génétique à Archivage de pareto.

^v Diversity Enhancement Genetic Algorithm.

^{vi} Tassili Airlines.