Optimisation multi-objectif et algorithmes génétiques pour établir un plan blanc

Table des matières

1

Plan blanc

① Introduction

L’exercice d’une fonction dans le domaine médical requiert de nombreuses compétences et de l’expertise de la part de l’individu étant donné que l’on mise sur la vie et la mort des patients. Ceci aboutit donc souvent à une rareté des chirurgiens dans les centres de santé pourtant confrontés à un nombre toujours croissant et excessif de malades par jour. Ceci est d’autant plus vrai lorsque des catastrophes naturels (inondation, séismes, pandémies, ...) se déclenchent à l’improviste et causent ainsi de nombreux blessés et au pire des cas des morts, augmentant donc l’afflux des patients dans les centres de santé, et la pression sur l’ensemble des équipes médicaux qui doivent s’organiser pour attribuer les soins les plus immédiats et les opérations nécessaires.

② Position du problème & Origine de référence

Le problème consiste à proposer un modèle de plan blanc dans le cas d’une catastrophe survenue. Nous devons entre autres connaître le nombre, l’état, la date d’arrivée des patients qui doivent se faire opérés; ainsi que la date d’arrivée des équipes médicales. Dans ce contexte, on parle seulement de chirurgien pour se référer à une équipe médicale. Une équipe médicale étant constituée de plusieurs chirurgiens, elle commencera par opérer un patient à condition que le dernier membre nécessaire pour l’opération soit présent. Ainsi, la date d’arrivée d’un chirurgien dans notre modèle se réfère implicitement à la date d’arrivée du dernier membre de l’équipe qui garantit que cette dernière est au complet et que l’opération peut commencer.

Par ailleurs, chaque chirurgien doit opérer un patient fictif. Le patient fictif est en général un personne qui simule des symptômes d’une maladie dans le but de permettre aux médecins de mieux comprendre la maladie et de distinguer les vrais malades des faux malades. Bien évidemment le temps pour traiter un patient fictif est zéro. L’interet du patient fictif est qu’il permet de réduire le nombre le variable de décisions dans le modèle mathématique, sinon on devrait ajouter une variable permettant de savoir le premier patient opéré par un chirurgien donné.

On remarque aussi que dans le domaine médical, chaque chirurgien dispose d’une « charge maximale de travail » qui est le temps maximum au cours duquel un chirurgien peut travailler sans compromettre son état de santé, ainsi que la qualité des soins qu’il fournit à ses patients.

La catastrophe a lieu à un temps T = 0 min. Le début de la catastrophe est notre référence dans le modèle. Toutes les dates (dates d’arrivée des chirurgiens, des patients, ...) seront prises en compte à l’instant où la catastrophe a lieu. Si le patient arrive alors à l’hôpital à t = 5, cela veut dire que sa date d'arrivée est 5 minutes après l’avènement de la catastrophe Si la date d’opération au plus tard d'un patient est t = 120, cela veut dire qu'il doit se faire traiter au plus tard 2 heures (120 minutes = 2 h) après le début de la catastrophe. Il en est de même pour la date d’opération du patient. De plus si on considère que le chirurgien arrive au centre de santé t minutes après le début de la catastrophe , et que sa charge maximale (temps qui lui reste à travailler avant d'être déchargé de ses fonctions) est Q, alors sa date d'arrêt de travail est Q+t. C'est-à-dire qu'au delà de Q+t après le début de la catastrophe, le chirurgien ne peut plus opérer de patients.

③ **Modélisation mathématique**

Données du problème bi-objectif :

£ Ensemble des patients $V = \{1,...,N\}$

£ Nombre de patients N

£ Nombre de chirurgiens (bloc de chirurgiens) H

£ Un patient factice qui est opéré par tous les chirurgiens $\{0\}$

£ Durée d'intervention (ou d'opération) du patient $i : d_i$

£ Date d'arrivée du chirurgien $k : rs_k$

£ Date d'arrivée du patient $i : rv_i$

£ Date au plus tard d'opération d'un patient $i : dl_i$

£ Charge maximale de travail d'un chirurgien Q^k

£ Un ensemble $V^* = V \bigcup \{0\}$

Variables de décisions :

Pour tout chirurgien $k \in \{1,...H\}$ et tout patient $i, j \in V^*$ on pose :

- $x_{i,j}^k = \begin{cases} 1 \text{ si le chirurgien } k \text{ opère le patient } j \text{ après avoir opéré le patient } i \\ 0 \text{ sinon} \end{cases}$

Remarque : Pour savoir si un chirurgien a effectué une opération, il suffit de calculer la somme $S = \sum_{j=1}^{V} x_{0,j}^k$. Cette somme doit être inférieure ou égale à 1 car tout chirurgien ne peut qu'opérer un seul patient à la fois. Ainsi si par exemple, $x_{0,1}^1 = 1$, alors cela montre que le chirurgien 1 a opéré en première position le patient 1.

- w_i le début de l'opération du patient i

Contraintes

● *La première opération d'un chirurgien ne peut se faire que sur un seul patient au plus, autrement tous les chirurgiens ne sont pas obligés de faire des opérations*

$$\sum_{j=1}^{N} x_{0,j}^{k} \leq 1, \forall k \in \{1,\ldots,H\}$$

● *Un patient ne peut subir une opération que chez un seul chirurgien. Il a donc 2 cas pour un patient : soit il est le premier dans la liste des patients opéré par un chirurgien donné, ou bien il est le successeur d'un autre patient*

$$\begin{cases} \sum_{k=1}^{H} x_{0,j}^{k} = 1 \ et \ \sum_{k=1}^{H} \sum_{i \in V \setminus \{j\}} x_{i,j}^{k} = 0 \ \ \forall j \in V \\ \qquad\qquad ou \\ \sum_{k=1}^{H} \sum_{i \in V \setminus \{j\}} x_{i,j}^{k} = 1 \ et \ \sum_{k=1}^{H} x_{0,j}^{k} = 0 \ \ \forall j \in V \end{cases} \quad □ \quad \sum_{k=1}^{H} \sum_{i \in V^{*} \setminus \{j\}} x_{i,j}^{k} = 1 \ \ \forall j \in V$$

● *Le respect de la fenêtre de temps, le début de l'opération d'un patient doit se faire entre sa date d'arrivée et sa date au plus tard*

$$rv_i \leq w_i \leq dl_i \ \ \forall i \in V$$

● **S***i le patient j est opéré après le patient i par le chirurgien k dans cet ordre, alors le début de l'opération pour le patient j doit être au moins égal à la fin de l'opération pour le patient i*

$$w_j \geq (w_i + d_i)\, x_{i,j}^{k}, \forall k \in \{1,\ldots,H\}, \forall (i,j) \in V \times V$$

Cette contrainte est non linéaire puisque deux variables de décision sont multipliées entre elles, mais on peut la linéariser. On pose alors :

$$w_j \geq (w_i + d_i) - (dl_i + d_i)(1 - x_{i,j}^{k}), \forall k \in \{1,\ldots,H\}, \forall (i,j) \in V \times V$$

Nous avons deux cas :

** Si j est effectivement opéré après i par le chirurgien k, le membre de droite donne :*

$$(w_i + d_i) - (dl_i + d_i)(1 - x_{i,j}^{k}) = w_i + d_i \ \ (avec \ x_{i,j}^{k} = 1)$$

L'inéquation est donc vérifiée.

** Si ce n'est pas le cas, le membre de droite donne :*

$$(w_i + d_i) - (dl_i + d_i)(1 - x_{i,j}^{k}) = w_i - dl_i \leq 0$$

L'inéquation est encore vérifiée (les variables de décision étant positives)

La contrainte qui garantit la succession des opérations sans que ces dernières ne se chevauchent se résume donc ainsi :

$$w_j \geq (w_i + d_i) - (dl_i + d_i)(1 - x_{i,j}^k), \forall k \in \{1,\ldots,H\}, \forall (i,j) \in V \times V$$

● **S***i le patient j est le premier à être opéré par le chirurgien k, alors le début de l'opération pour le patient j doit être au moins égal à la date d'arrivée du chirurgien k à l'hôpital*

$$w_j \geq rs_k x_{0,j}^k, \forall k \in \{1,\ldots,H\}, \forall j \in V$$

● *Le patient fictif doit être traité avant tous les autres* :

$$w_i \geq w_0 \quad \forall i \in V$$

● *Tous les patients doivent être opérés*

Le nombre de patients opérés par un chirurgien est : $\sum_{j=1}^{N} x_{0,j}^k + \sum_{i=1}^{N} \sum_{j \in V\{i\}} x_{i,j}^k \quad \forall k \in \{1,\ldots,H\}$

La première somme $\sum_{j=1}^{N} x_{0,j}^k \; \forall k \in \{1,\ldots,H\}$ *indique si le chirurgien a opéré au moins un patient. Si oui alors elle indique aussi sur lequel parmi les patients il a fait sa première opération*

La deuxième somme $\sum_{i=1}^{N} \sum_{j \in V\{i\}} x_{i,j}^k \quad \forall k \in \{1,\ldots,H\}$ *indique si le chirurgien a effectué au moins une opération après sa première (s'il en a fait)*

Alors on a

$$\sum_{k=1}^{H} \left(\sum_{j=1}^{N} x_{0,j}^k + \sum_{i=1}^{N} \sum_{j \in V\{i\}} x_{i,j}^k \right) = N$$

● *Si le chirurgien n'a pas effectué de première opération sur un patient donné, alors il ne peut avoir un patient qui soit opéré par lui après un autre (autre que le patient fictif). En se référant à la contrainte précédente on a deux cas :*

First case : si $\sum_{j=1}^{N} x_{0,j}^k = 0$ *alors* $\sum_{i=1}^{N} \sum_{j \in V\{i\}} x_{i,j}^k = 0 \;\; \forall k \in \{1,\ldots,H\}$

Second case : si $\sum_{j=1}^{N} x_{0,j}^k = 1$ *alors* $\sum_{i=1}^{N} \sum_{j \in V\{i\}} x_{i,j}^k \geq 0 \; \forall k \in \{1,\ldots,H\}$

On peut résumer ces deux conditions en une seule. Comme on sait que le nombre d'opérations effectuées par un chirurgien ne peut pas dépasser le nombre total de patients, on a donc :

$$\sum_{i=1}^{N} \sum_{j \in V\{i\}} x_{i,j}^k \leq N \times \sum_{j=1}^{N} x_{0,j}^k \quad \forall k \in \{1,\ldots,H\}$$

● *La charge de travail maximale de chaque chirurgien ne doit pas être excédé*

Pour chaque chirurgien arrivant à une date rs_k *et ayant une charge maximale de travail* Q^k*, il doit arrêter de travailler à* $rs_k + Q^k$*. Il faut alors que la date de la dernière opération effectuée par le chirurgien plus la durée de cette opération n'excède pas la valeur* $rs_k + Q^k$.

$$\forall k \in \{1,\dots,H\} \begin{cases} \sum_{j=1}^{N} x_{0,j}^{k}\left(w_j + d_j\right) \le rs_k + Q^k \\ et \\ \sum_{j \in V \setminus \{i\}} x_{i,j}^{k}\left(w_j + d_j\right) \le rs_k + Q^k \quad \forall i \in V = \{1,\dots,N\} \end{cases}$$

Cette contrainte est malheureusement non linéaire car il un produit de deux variables de décision.

<u>Fonctions objectifs</u> *:*

● *Minimiser de façon globale la somme de la différence des écarts entre le début de l'opération d'un patient et sa date d'arrivée*

$$Min \sum_{j=1}^{N}\left(w_j - rv_j\right)$$

● *Minimiser l'ensemble des chirurgiens qui interviennent pour les opérations*

$$Min \sum_{k=1}^{H}\sum_{j=1}^{N} x_{0,j}^{k}$$

2

④ Codage d'un individu (codage réel)

Un individu est un plan blanc réalisable. Il est représenté sous forme d'une matrice à deux dimensions. Le nombre de lignes étant égal au nombre de patients et le nombre de colonnes au nombre de chirurgiens. A l'intersection de la ligne *i* et de la colonne *j*, nous avons la date d'opération du patient *i* par le chirurgien *j*. Si *P* représente un plan blanc constitué de *N* patients et *H* chirurgiens, on a donc :

$$P(i,j) = \begin{cases} 0 \; si \; le \; patient \; i \; n'est \; pas \; opéré \; par \; le \; chirurgien \; j \\ a \; > \; 0 \; sinon \end{cases}$$

Étant donné qu'un patient ne peut se faire opérer que par un seul chirurgien, on a donc pour tout patient $i \in \{1,...,N\}$:

$$\begin{cases} \exists \; un \; unique \; q \in \{1,\ldots,H\} \; tel \; que \; P(i,q) \; > \; 0 \\ \forall \; h \in \{1,\ldots,H\} \backslash \{q\} \; P(i,h) \; = 0 \end{cases}$$

Exemple de représentation d'un individu :

t_{11}	0	0	0	0
0	t_{22}	0	0	0
0	0	0	0	t_{35}
0	0	0	0	t_{45}
0	0	t_{53}	0	0
0	0	t_{63}	0	0
0	0	0	0	t_{75}
0	t_{82}	0	0	0
0	t_{92}	0	0	0
t_{101}	0	0	0	0

La matrice *P* ci-dessus est un plan blanc constitué de 5 chirurgiens et 10 patients. Comme on le remarque, chaque patient n'est opéré que par un seul chirurgien. Nous allons voir comment on peut vérifier que cette matrice respecte les contraintes de notre modèle. On s'intéresse au premier chirurgien i.e la colonne 1 (les autres cas étant pareils). Il opère deux patients : les patients 1 & 10. On rappelle que l'origine des références des dates est le début de la catastrophe à T = 0 min. Ainsi le patient 1 se fait opérer t_{11} minutes après le début de la catastrophe, tandis que le patient 2 se fait opérer t_{101} minutes après.

* Soit s_1 la date d'arrivée du chirurgien à l'hôpital et Q_1sa charge maximale à l'instant où il arrive

* Soient e_1, e_2 les dates d'arrivée respectives des patients 1 & 10; l_1, l_2 les dates au plus tard respectives des deux patients; et d_1, d_2 les durées respectives de leurs opérations

$$\text{On a alors : } \begin{cases} max(\, e_1, \, s_1) \; \leq \; t_{11} \; \leq \; l_1 \\ max(t_{11} + d_1, e_2) \; \; \leq \; t_{101} \; \leq \; l_2 \\ t_{101} + d_2 \; \leq \; s_1 + Q_1 \end{cases}$$

Il en est de même pour toutes les autres colonnes de la matrice *P*.

⑤ Ordre de passage

Supposons que nous avons un ensemble de patients après la catastrophe. On peut toujours les classer par ordre de priorité, c'est-à-dire qui parmi les patients aura plus de privilège à se faire opérer avant les autres. Ceci peut se faire grâce à la date au plus tard des patients. De deux patients A & B, si A a une date au plus tard plus courte que B, A sera privilégié puisque son cas est plus critique que celui de B. Il a donc du bon sens à vouloir d'abord opérer A pour le mettre hors de danger, puis ensuite B.

Ainsi, l'ensemble des patients est trié par ordre croissant de leur date au plus tard.

⑥ Croisement & Mutation[3]

Crossing

Table 2.4: Parent P_1

0	t_{12}	0	0
0	0	t_{23}	0
t_{31}	0	0	0
t_{41}	0	0	0
0	t_{12}	0	0
0	0	t_{63}	0

Table 2.5: Parent P_2

0	0	t_{13}	0
0	0	0	t_{24}
t_{31}	0	0	0
0	0	t_{43}	0
0	t_{52}	0	0
0	0	0	t_{64}

Table 2.6: Offspring E_1

0	t_{12}	t_{13}	0
0	0	0	t_{24}
t_{31}	0	0	0
0	0	0	0
0	t_{52}	0	0
0	0	t_{63}	0

Table 2.7: Offspring E_2

0	0	0	0
0	0	t_{23}	0
t_{31}	0	0	0
t_{41}	0	t_{43}	0
0	t_{52}	0	0
0	0	0	t_{64}

Mutation

Table 2.8: Parent P_3

0	t_{12}	0	0
0	0	t_{23}	0
t_{31}	0	0	0
t_{41}	0	0	0
0	t_{52}	0	0
0	0	t_{63}	0

Table 2.9: Offspring E_3

0	t_{12}	0	0
0	0	t_{23}	0
t_{31}	0	0	0
t_{41}	0	0	0
0	0	0	t_{52}
0	0	t_{63}	0

⑦ Evaluation d'un Individu

La phase d'évaluation est capitale dans les algorithmes génétiques car elle permet généralement de définir les individus à sélectionner pour le croisement et la mutation. Dans notre cas, un individu est évalué en calculant la valeur des deux fonctions objectifs à minimiser. On rappelle qu'on cherche à minimiser à la fois le « nombre de chirurgiens nécessaires pour tout le plan blanc » & « le retard global dans les interventions chirurgicaux ». On reprend l'exemple précédent. Soit l'individu P suivant :

t_{11}	0	0	0	0
0	t_{22}	0	0	0
0	0	0	0	t_{35}
0	0	0	0	t_{45}
0	0	t_{53}	0	0
0	0	t_{63}	0	0
0	0	0	0	t_{75}
0	t_{82}	0	0	0
0	t_{92}	0	0	0
t_{101}	0	0	0	0

On désigne par $E(P) = (X, Y)$ la performance ou l'évaluation du plan blanc P. Soit e_j la date d'arrivée d'un patient quelque soit $j \in \{1, \ldots, 10\}$.

* Le plan blanc ci-dessus utilise 4 sur 5 chirurgiens pour opérer tous les patients donc on a : $X = 4$

* Le retard global dans les interventions vaut :

$$Y = (t_{11} - e_1) + (t_{101} - e_{10}) + (t_{22} - e_2) + (t_{82} - e_8) + (t_{92} - e_9) + (t_{53} - e_5) + (t_{63} - e_6) + (t_{35} - e_3) + (t_{45} - e_4) + (t_{75} - e_7)$$

Par conséquent, on a $E(P) = (4, Y)$ (la valeur de Y étant à calculer)

⑧ Front de Pareto

Le front de Pareto est constitué de l'ensemble des individus non dominés. Soient deux individus A & B. On désigne respectivement par E(A) & E(B) l'évaluation de A & B. On pose conformément à ce qui précède :

$$\begin{cases} E(A) = (X_1, Y_1) \\ E(B) = (X_2, Y_2) \end{cases}$$

On dit que A domine B si et seulement si on a :

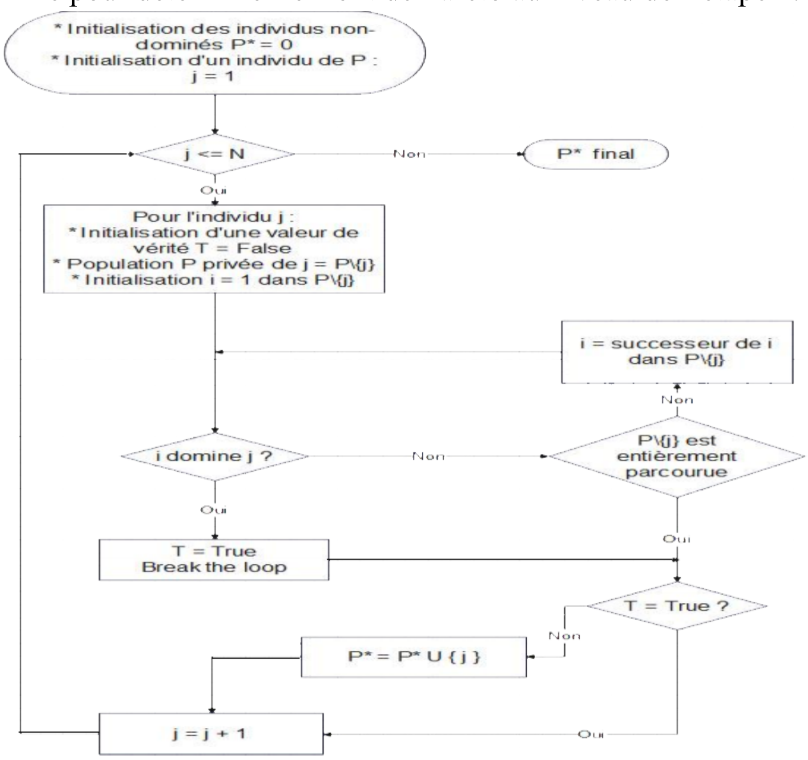
$$\begin{cases} X_1 \leq X_2 \\ Y_1 < Y_2 \end{cases} \text{ ou } \begin{cases} X_1 < X_2 \\ Y_1 \leq Y_2 \end{cases}$$

Si l'une des deux conditions n'est pas vérifiée, alors B est non dominé par A.
En supposant que nous avons au départ une population {G} de *N* individus, on cherche à la subdiviser en plusieurs fronts de Pareto. Voici un exemple de pseudo code pour diviser {G} en plusieurs fronts de Pareto.

Pseudo Code

**** [Étape 0] Créer une liste vide L***
**** [Étape 1]***
1) Déterminer le front de Pareto {FP} de {G}
2) Ajouter {FP} à L : L = L + {FP}
3) Retrancher {FP} de {G} : {G} = {G} - {FP}
**** [Étape 2]***
Tant que {G} non vide :
Retourner à [Étape 1]
**** [Étape 3] Retourner L***

Ci dessous l'organigramme pour déterminer le front de Pareto au niveau de l'étape 1.



Supposons que nous obtenons à la fin du pseudo code une liste de longueur $q \geq 2$ telle que : [4]

$$L = \{FP_j \ tel\ que\ j \in \{1, \dots, q\}\}\ avec \begin{cases} \bigcup_{1 \leq j \leq q} FP_j = \{G\} \\ FP_i \cap FP_j = \emptyset\ \forall\ i \neq j \end{cases}$$

Alors on a : $\forall i \in \{1, \dots, q-1\}$, l'ensemble FP_i domine les ensembles $\{FP_{i+1}, \dots, FP_q\}$
Les individus d'un rang inférieur sont meilleurs que ceux du rang supérieur.

⑨ Crowding-Distance

De ce qui précède, on sait que les individus du rang inférieur sont meilleurs que ceux du rang supérieur. Cependant nous ne savons pas comment comparer les individus d'un même rang puisque deux individus du même rang ne se dominent pas. C'est à ce niveau qu'intervient la métrique « crowding distance » ou distance de surpeuplement qui permet de classer les individus provenant d'un même rang de non domination. Cette métrique a pour rôle d'assurer la diversité de la population lors de la phase de sélection en vue de ne pas converger vers un minimum local et d'explorer davantage l'espace des solutions. Voici un pseudo code.

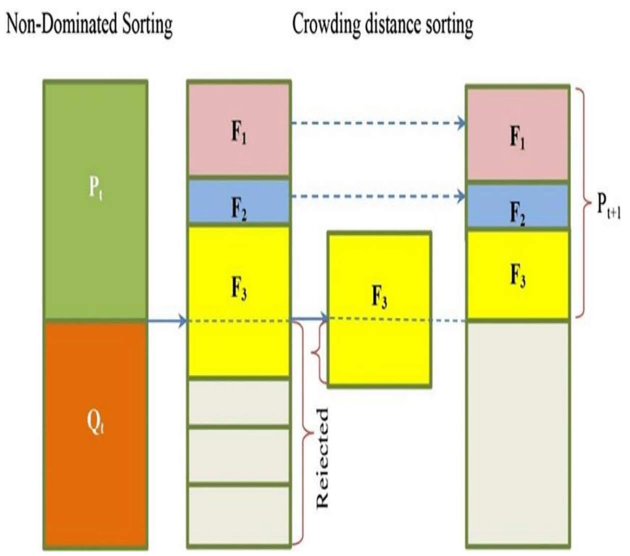
Pseudo code

**** F = Ensemble des individus d'un niveau de non domination***
**** Initialiser la distance de tous les individus de F à 0***
**** Pour chaque fonction objectif m*** $\in \{1, 2\}$ ***:***
**** Trier dans l'ordre croissant les individus de F selon la valeur de la fonction objectif m et obtenir une nouvelle liste F'***
**** Attribuer une distance infinie aux deux individus qui détiennent la valeur minimale et maximale de la fonction objectif m dans F'***
**** Pour tous les autres individus :***
**** Mettre à jour la distance de l'individu selon la formule suivante***
$d[i] = d[i] + (F'[i+1]_m - F'[i-1]_m) / (max_m - min_m)$
// d[i] : distance de l'individu
// $F'[i+1]_m$ ***: valeur de la fonction objectif m du successeur de i***
// $F'[i-1]_m$ ***: valeur de la fonction objectif m du prédécesseur de i***
// max_m ***: maximum de la fonction objectif m dans F'***
// min_m ***: minimum de la fonction objectif m dans F'***

Les individus seront sélectionnés dans l'ordre décroissant de leur distance à la fin du pseudo code de telle manière que ceux ayant une distance infinie seront toujours choisis en premier lieu.

⑩ Sélection

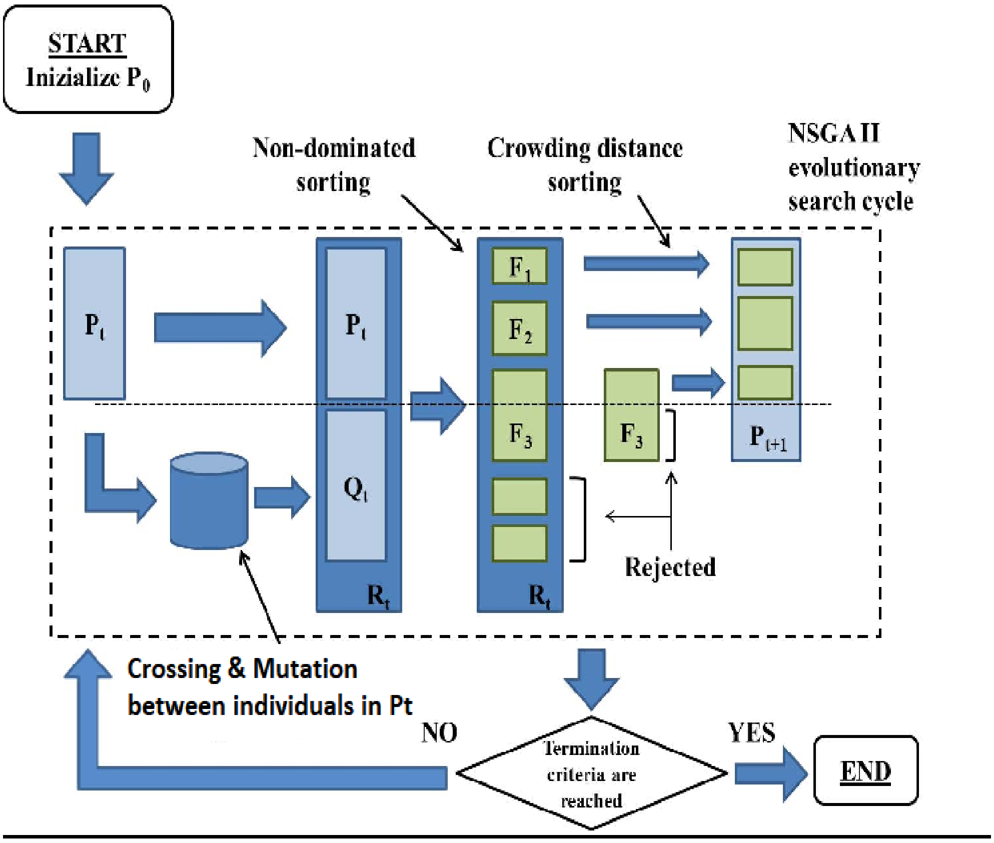
La phase de sélection consiste à choisir parmi les individus ceux qui seront utilisés dans les opérations de croisement et de mutation. Comme vu précédemment, avant de faire la sélection, l'ensemble {G} des individus est subdivisé en plusieurs fronts de non domination. Le front de rang inférieur est toujours choisi en premier. Vient ensuite les autres fronts. Si à un front donné, on remarque que tous les individus n'ont pas besoin d'être pris (cela dépend du nombre total d'individus à sélectionner), alors on se base sur la distance de crowding pour déterminer lequel des individus du même front sera choisi.



* Pt représente l'ensemble des individus qui seront mutés et croisés
* Qt est la génération issue du croisement et de la mutation des individus de Pt

L'ensemble {Qt U Pt} est d'abord subdivisé en plusieurs niveaux de non domination, les individus de rang inférieur étant toujours sélectionnés. La sélection des individus de même rang se fait en utilisant la distance de crowding.

⑪ Global Structure of NSGA



⑫ <u>Inventaire des fonctions implémentées</u>

Le langage de programmation utilisé est « python » qui est assez prisé pour sa flexibilité et le grand nombres de bibliothèques et fonctions déjà prédéfinies.[5]

L'implémentation de l'algorithme NSGA s'est fait en définissant au préalable des blocs de fonction pour rendre la fonction finale plus lisible et plus compréhensible. Ci-dessous un inventaire complet des blocs de fonctions définies.

Fonctions	Arguments principaux	Rôle & Retour de la fonction
Order_passage_privilege( )	*Nombre de patients : N *Date d'arrivée des patients *Date au plus tard des patients *Durée des opérations de chaque patient	* Renvoie l'ensemble des patients triés dans l'ordre croissant de leur date au plus tard; la liste triée sera passée en argument à la fonction d'initialisation d'un individu
Init_Individu_White_Plan ()	*Nombre de patients : N *Nombre de chirurgiens : H *Date d'arrivée des patients *Date au plus tard des patients *Date d'arrivée des chirurgiens *Durée des opérations de chaque patient *Charge maximale de travail des chirurgiens	* Générer un individu réalisable s'il existe selon les données passées en argument * Renvoie : $\begin{cases} un\ individu\ s'il\ existe \\ Failure\ of\ White\ Plan\ si \end{cases}$
Init_Population_White_Plan()	* Nombre d'individus à créer dans la population + arguments de la fonction « Init_Individu_White_Plan() »	* Renvoie une population d'individus (Il est conseillé de faire spécifier un nombre d'individus ≥ 50)
Check_out_Population()	*Population d'individus crées + arguments de la fonction « Init_Population_White_Plan () »	* Vérifie que la population d'individus crées respecte toutes les contraintes * Renvoie True or False N.B. La fonction prend son importance dans le croisement et la mutation pour vérifier si la progéniture est réalisable.
performance_population()	*Population d'individus crées *Date d'arrivée des patients	* Évalue les individus de la population selon les deux fonctions objectifs

	*Durée des opérations de chaque patient	* Renvoie une matrice où chaque ligne représente l'évaluation d'un individu
Front_Pareto()	Arguments de la fonction « performance_population() »	* Détermine le front de Pareto de la population * Renvoie l'ensemble des individus non dominés
crossing()	*Deux individus A & B à croiser *Une probabilité au dessus de laquelle les deux individus peuvent être croisés	* Renvoie les deux enfants issus du croisement
heuristique_crossing()	*Un enfant issu du croisement	* Corrige l'enfant s'il n'est pas réalisable * Renvoie l'enfant corrigé
mutate()	* L'individu à muter *Une probabilité au dessus de laquelle l'individu peut être croisés	* Renvoie l'individu issu de la mutation
heuristique_mutate()	*L'individu issu de la mutation	* Corrige l'individu muté s'il n'est pas réalisable * Renvoie l'individu corrigé
divide_non_dominated()	Arguments de la fonction « performance_population() »	* Divise la population en plusieurs niveaux de non domination * Renvoie les différents fronts de non domination
crowding_distance()	* Les différents niveaux de non domination * Les performances des individus à chaque niveau de non domination * Un argument spécifiant le niveau où l'on veut trier les individus	* Trie les individus d'un même front de non domination * Renvoie les individus triés d'un front de non domination

⑬ Simulation & Résultats

Données $\begin{cases} 50\ patients \\ 5\ chirurgiens \end{cases}$

La charge maximale de chaque chirurgien à leur arrivée à l'hôpital après le début de la catastrophe à T = 0 est fixée à 11,5 h soit 690 minutes.

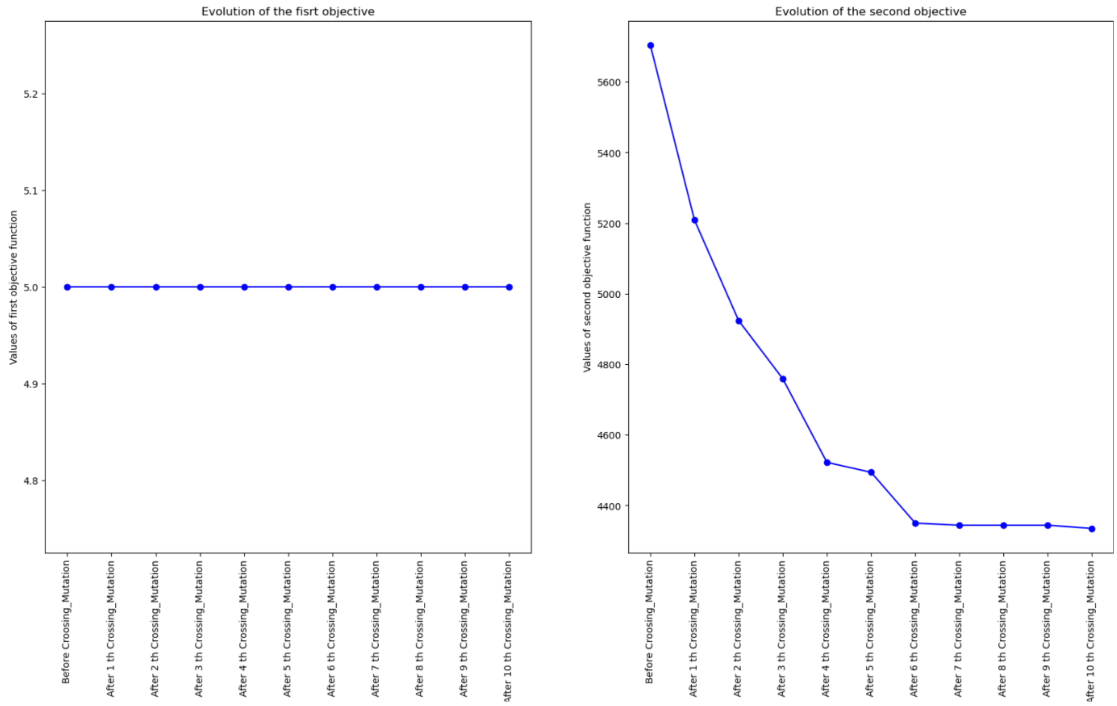
Les chirurgiens ont une date d'arrivée suivant une suite arithmétique de raison 3 (si t_j représente la date d'arrivée d'un chirurgien alors $t_{j+1} = t_j + 3 \ \forall j \in \{1, \ldots, 4\}$ avec $t_0 = 0$)

Une vue globale des données des patients :

	0	1	2	3	4	5	6	7	8	9	...	41	42	43	44	45	46	47	48	49	50
0	Duration operations in mn	60	30	30	30	90	30	30	30	60	...	60	30	30	60	30	90	60	30	90	30
1	Late date patients in mn	510	270	420	150	630	750	300	630	450	...	600	450	600	900	660	630	480	630	600	510
2	Early date patients in mn	0	0	240	30	120	420	120	450	60	...	420	360	270	450	240	480	150	180	270	210

Une population de 100 individus à été initialisée avec un nombre maximal d'itérations fixé à 10.

Par ailleurs, l'évolution des deux fonctions objectifs au cours des itérations a été tracé pour montrer une amélioration de la performance des individus après les opérations de croisement et de mutation.



On remarque d'après la courbe de gauche, que tous les chirurgiens sont utilisés sont pour le plan blanc, ce qui semble logique étant donné le grand nombre de patients et le nombre restreint de chirurgiens.

Cependant on constate clairement au niveau de la courbe de droite une amélioration de la valeur de la seconde fonction objectif suite aux opérations de croisement et de mutation.

⑭ Représentations graphiques du plan Blanc

Pour évaluer l'efficacité de notre code, nous avons procédé à 10 itérations, impliquant un total de 50 patients pris en charge par 5 chirurgiens distincts. Les résultats obtenus ont été rigoureusement analysés et présentés à travers des graphiques dédiés à chaque chirurgien.

-

Le premier graphique pour chaque chirurgien offre une représentation visuelle du temps nécessaire pour chaque opération en fonction de chaque patient. Sur l'axe des ordonnées, nous avons minutieusement indiqué le numéro du patient. En examinant le bloc correspondant à un patient donné, situé sur la ligne correspondante, nous sommes en mesure d'observer clairement le début et la fin de l'opération.

Il est crucial de souligner que les opérations de chaque chirurgien se déroulent dans la plage temporelle comprise entre sa date d'arrivée et la somme de sa date d'arrivée et de sa charge maximale de travail.

Le deuxième graphique fournit une perspective détaillée du plan blanc en présentant, pour chaque chirurgien, la date d'opération du patient en abscisse et, en ordonnées, la date des opérations. Cette représentation permet d'illustrer le parcours suivi par chaque chirurgien pendant le plan blanc. Il est essentiel de souligner à nouveau que les opérations de chaque chirurgien sont planifiées dans la plage temporelle définie par sa date d'arrivée et la somme de sa date d'arrivée et de sa charge maximale de travail. Cette considération temporelle précise contribue à mettre en lumière les interactions et les séquences d'opérations, offrant ainsi une compréhension approfondie du déroulement du plan blanc pour chaque praticien.

En combinant ces deux graphiques, nous sommes en mesure de discerner les moments spécifiques où les chirurgiens ont intervenu, sur quels patients, et d'analyser le temps dédié à chaque opération. Ces visualisations enrichissent notre compréhension des performances individuelles, apportant ainsi des informations cruciales pour optimiser le plan blanc et l'efficacité globale des chirurgiens.

Et en fin de document, nous présentons deux diagrammes en bâtons. Le premier illustre, pour chaque chirurgien, le nombre d'opérations qu'il a effectuées, tandis que le deuxième représente la durée totale des opérations réalisées par chaque chirurgien. Une observation importante réside dans le fait qu'un chirurgien peut avoir pris en charge moins de patients mais avoir une durée de travail supérieure à celle des autres, et vice versa. Ces diagrammes offrent une vision comparative des performances des chirurgiens, mettant en évidence des nuances significatives dans leur productivité et leur gestion du temps opératoire. Cela souligne l'importance de considérer à la fois la quantité et la durée des opérations pour évaluer de manière holistique les contributions individuelles de chaque chirurgien au sein du plan blanc.

⑮ Bibliographie

(S. Makboul, A. El Hilali Alaoui) Rapport de These, *Developing multiple approaches based on robust and multi-objective optimization to promote health care systems.*