

# 1 Introduction

Les heuristiques sont des méthodes spécifiques qui exploitent au mieux la structure du problème dans le but de trouver une solution raisonnable (non nécessairement optimale) en un temps réduit. L'utilisation de ce type d'algorithmes s'impose car les méthodes de résolution exactes sont de complexité exponentielle, et échouent à trouver la solution pour des instances de tailles moyennes voir petites, comme on la constater lors du chapitre précédant . L'usage des heuristiques est donc pertinent pour surmonter ces limites.

Dans ce chapitre, nous allons présenter la conception détaillée des heuristiques sur lesquelles notre choix d'implémentation s'est porté et qui sont:

1. Next Fit (NF)
2. Next Fit Decreasing (NFD)
3. First Fit (FF)
4. First Fit Decreasing (FFD)
5. Best Fit (BF)
6. Best Fit Decreasing (BFD)

Dans le but d'explorer ces méthodes, comparer leurs performances, montrer leurs avantages et découvrir leurs limites , nous effectuerons des tests empiriques et comparatifs sur les mêmes benchmarks utilisés pour les tests des méthodes exactes.( Benchmark Scholl)

## 2 Next Fit (NF)

### 2.1 principe

Si l'article tient dans la même boîte que l'article précédent, il est placé avec ce dernier. Sinon, on ouvre une nouvelle boîte et le mettre là-dedans.

- NF est un algorithme simple d'une complexité de  $O(n)$ .

### 2.2 Pseudo-Code

[H] Next Fit Tous les articles  $i = 1, 2, \dots, n$  l'article  $i$  s'inscrit dans la boîte actuelle Ranger l'article  $i$  dans la boîte actuelle Créer une nouvelle boîte, en faire la boîte actuelle et ranger l'article  $i$  dedans.

## 3 Next Fit Decreasing (NFD)

### 3.1 principe

Le NFD est une amélioration de l'algorithme Next-Fit. Cet algorithme ordonne es articles par ordre décroissant des poids, puis applique l'algorithme NF.

### 3.2 Pseudo-Code

Next Fit Decreasing Triez les articles par ordre décroissant Appliquer Next-Fit à la liste triée

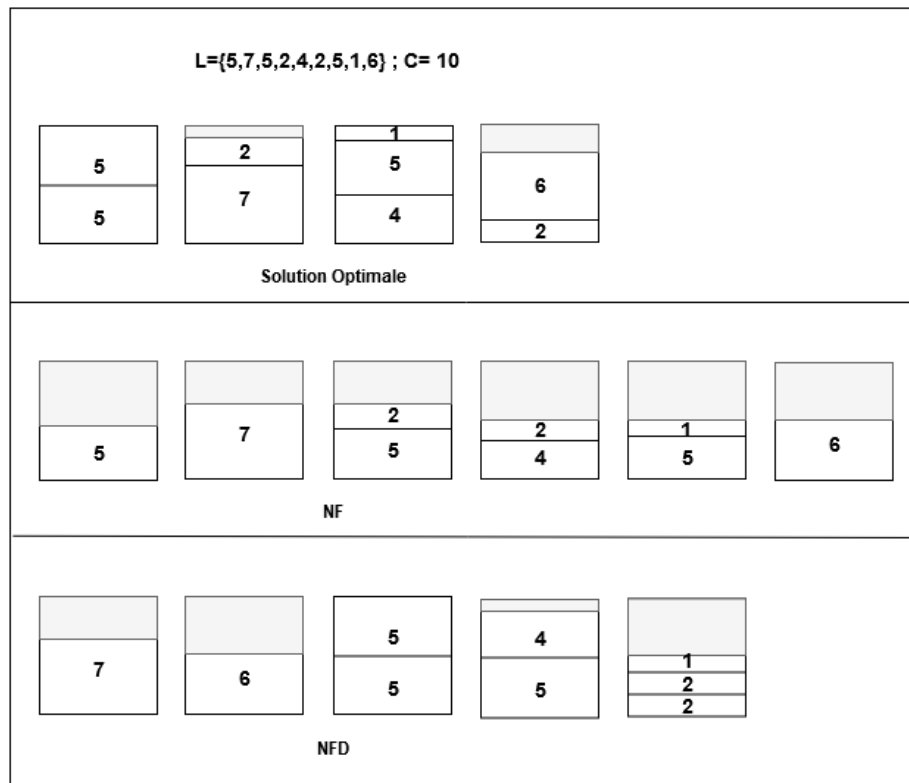


Figure 1: Exemple NF et NFD

## 4 First Fit (FF)

### 4.1 principe

Ranger chaque article courant dans la première boîte, entre celles déjà ouvertes, qui peut le contenir sinon ouvrir une nouvelle boîte et on le range dedans.

- L'algorithme First Fit peut être implémenté en  $O(n \log n)$  en utilisant les arbres de recherche binaires

### 4.2 Pseudo-Code

[H] First Fit Tous les articles  $i = 1, 2, \dots, n$  Tous les boîtes  $j = 1, 2, \dots, m$   
 m l'article  $i$  s'inscrit dans la boîte  $j$  Ranger l'article  $i$  dans la boîte  $j$  Quitter la boucle ( passer à l'article suivant) l'article  $i$  ne rentre dans aucune boîte disponible Créer une nouvelle boîte et ranger l'article  $i$  dedans

## 5 First Fit Decreasing (FFD)

### 5.1 principe

Le FFD est une amélioration de l'algorithme First-Fit . Cet algorithme ordonne les poids dans le sens décroissant puis lui applique l'algorithme FF.

- L'algorithme First Fit peut être implémenté en  $O(n \log n)$  en utilisant les arbres de recherche binaires

### 5.2 Pseudo-Code

First Fit Decreasing Triez les articles par ordre décroissant Appliquer First-Fit à la liste triée

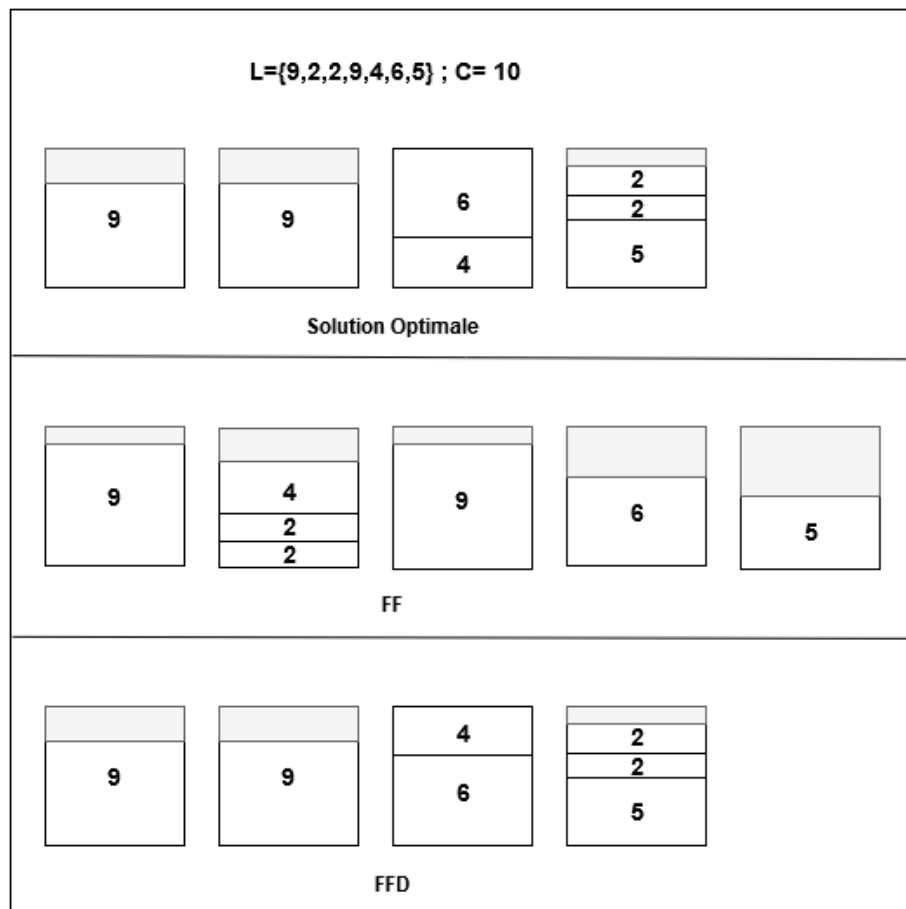


Figure 2: Exemple FF et FFD

## 6 Best Fit (BF)

### 6.1 principe

Ranger chaque article courant dans la boîte la mieux remplie, entre celles déjà ouvertes, qui peut le contenir sinon ouvrir une nouvelle boîte et on le range dedans.

- L'algorithme Best Fit peut être implémenté en  $O(n \log n)$  en utilisant les arbres de recherche binaires

### 6.2 Pseudo-Code

[H] Best Fit Tous les articles  $i = 1, 2, \dots, n$  Tous les boîtes  $j = 1, 2, \dots$   
m l'article  $i$  s'inscrit dans la boîte  $j$  Calculer la capacité restante dans la boîte  
 $j$  une fois l'article Ranger l'article  $i$  dans la boîte  $j$ , où  $j$  est la boîte ayant la  
capacité restante minimale après avoir ajouté l'article (c'est-à-dire que "l'article  
convient le mieux"). une telle boîte n'existe pas ( l'article ne peut être rangé  
dans aucune boîte) Créer une nouvelle boîte et ranger l'article  $i$  dedans

## 7 Best Fit Decreasing (BFD)

### 7.1 principe

Le BFD est une amélioration de l'algorithme Best-Fit . Cet algorithme ordonne les poids dans le sens décroissant puis applique l'algorithme BF.

### 7.2 Pseudo-Code

Best Fit Decreasing Triez les articles par ordre décroissant Appliquer Best-Fit à la liste triée

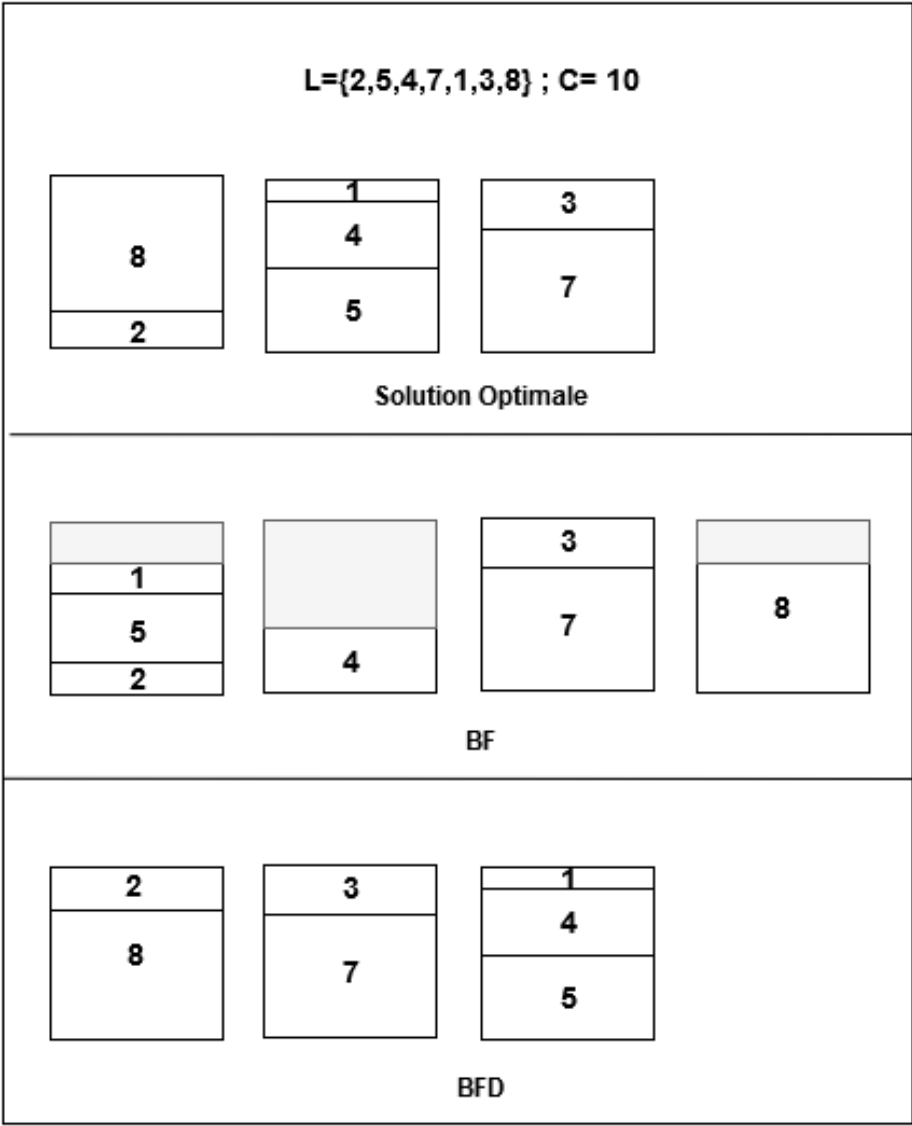


Figure 3: Exemple BF et BFD

## 8 Résultats des tests