



Projet TER

Recuit Simulé adapté au problème du tri du courrier de La Poste



YADEL Gaya - CANNAVACCIUOLO Nello

GUREVSKY Evgeny

Table des matières

I.	Introduction	3
II.	Tri de courrier chez La Poste	4
II..1	Phase 1 :	4
II..2	Phase 2 :	6
III.	Présentation de la problématique et de l'objectif	8
III..1	Problématique :	8
III..2	Objectif :	8
IV.	Modélisation du problème	9
V.	Résolution du problème :	12
V..1	Heuristique : Procédure de rééquilibrage <i>i,j shift</i> :	12
V..2	Métaheuristique : Algorithme du <i>Recuit Simulé</i> :	16
VI.	Résultats Recuit Simulé et comparaison avec ij_Shift	19
VII.	Graphiques	20
VIII.	Conclusion	22

I. Introduction

La Poste est une société anonyme française aux racines historiques profondes, fondée en 1477 par Louis XI pour le transport des messages royaux. Aujourd'hui, elle s'est diversifiée en un groupe multiservices majeur, offrant une gamme étendue de prestations. Ses activités principales englobent :

- les services postaux traditionnels (courrier, colis),
- les services bancaires et assurantiels via *La Banque Postale*,
- la téléphonie mobile,
- des solutions de commerce en ligne incluant le marketing et la logistique.

Cette évolution reflète l'adaptation de La Poste aux besoins contemporains, tout en conservant son rôle historique. Son actionnariat est réparti entre la *Caisse des dépôts et consignations* (66 %) et l'État français (34 %), soulignant son importance stratégique nationale.

Chaque jour, La Poste traite plusieurs millions de courriers à travers la France. En 2008, le volume annuel avoisinait 18 milliards de lettres, soit environ 49 millions de courriers par jour. Cependant, ce chiffre a considérablement diminué au fil des années. En 2020, La Poste ne traitait plus que 7,5 milliards de lettres, correspondant à environ 20 millions par jour. Cette baisse s'est accentuée avec la crise sanitaire, qui a entraîné une chute historique de 18 % en une seule année. En 2023, l'activité courrier ne représentait plus que 16 % du chiffre d'affaires de La Poste, mettant en lumière la transition vers des moyens de communication numériques et une réduction des envois physiques par les entreprises et les particuliers.

La transition vers les moyens de communication numériques, tels que les emails et les services en ligne, est la principale raison de la diminution du volume de courriers traités par La Poste. Cette évolution a eu des conséquences importantes sur l'organisation et l'économie du service postal. Pour faire face à la baisse drastique des volumes – passant de 18 milliards de lettres en 2008 à 7,5 milliards en 2020 –, La Poste a dû fermer plusieurs centres de traitement du courrier à travers la France, réduisant ainsi sa capacité opérationnelle. Parallèlement, le coût d'envoi d'une facture papier a considérablement augmenté, passant de 1,70 € à plus de 3 €, afin de compenser la baisse des revenus issus du courrier.

Ces mesures, bien que nécessaires pour maintenir l'équilibre financier, ont suscité des critiques, notamment de la part des entreprises et des usagers, qui doivent s'adapter à cette transformation. [1]

II. Tri de courrier chez La Poste

Le tri du courrier constitue une étape essentielle dans le processus logistique des services postaux, permettant d'assurer une distribution rapide et précise des envois. Avec des millions de lettres et colis traités quotidiennement, une organisation rigoureuse et des procédures bien définies sont nécessaires pour gérer efficacement ce flux massif de correspondances. Ce processus repose sur une série d'étapes méthodiques qui visent à regrouper et classer les courriers en fonction de leurs destinations finales.

Dans un premier temps, les courriers mélangés sont triés de manière globale en fonction des adresses, sans prendre en compte les tournées spécifiques. Ce premier tri constitue la base pour affiner ensuite la répartition des courriers dans les étapes suivantes. Une fois cette première organisation réalisée, un second tri permet de séparer les courriers selon les tournées des facteurs, tout en maintenant un ordre cohérent pour faciliter la distribution. Ce système en plusieurs phases garantit une gestion optimale des envois, tout en minimisant les erreurs et les délais.

Les sections suivantes détaillent les deux principales phases du tri postal, chacune ayant un objectif spécifique et complémentaire pour préparer les courriers à leur distribution finale.[2]

II.1 Phase 1 :

Dans la première phase du tri postal, les courriers mélangés provenant de différents points d'entrée arrivent dans un centre de tri. L'objectif principal de cette étape est de regrouper tous les courriers destinés à une même adresse dans un bac spécifique, indépendamment des tournées auxquelles ils appartiennent. Par exemple, le Bac 1 contiendra l'ensemble des courriers destinés à l'Adresse 1, le Bac 2 regroupera tous ceux pour l'Adresse 2, et ainsi de suite pour toutes les adresses. Ce processus garantit que chaque bac contient exclusivement les courriers destinés à une adresse unique, ce qui facilite les étapes ultérieures du traitement postal. Cependant, il est important de noter qu'à ce stade, les courriers regroupés dans un bac peuvent appartenir à des tournées différentes. Cette organisation initiale permet de structurer efficacement le flux des courriers en amont des opérations de tri plus spécifiques.[2]

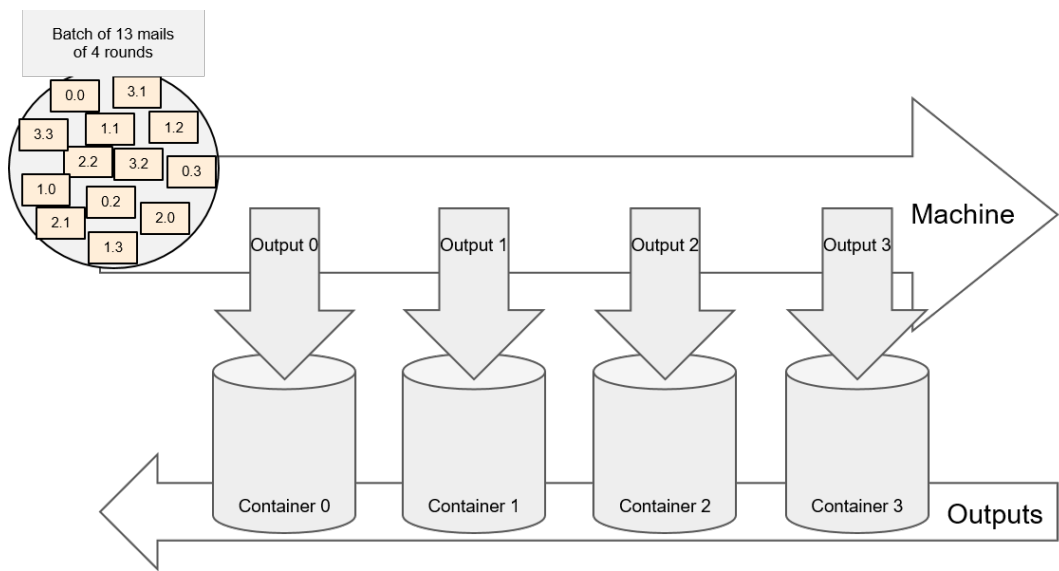


FIGURE 2 – Début de la phase 1

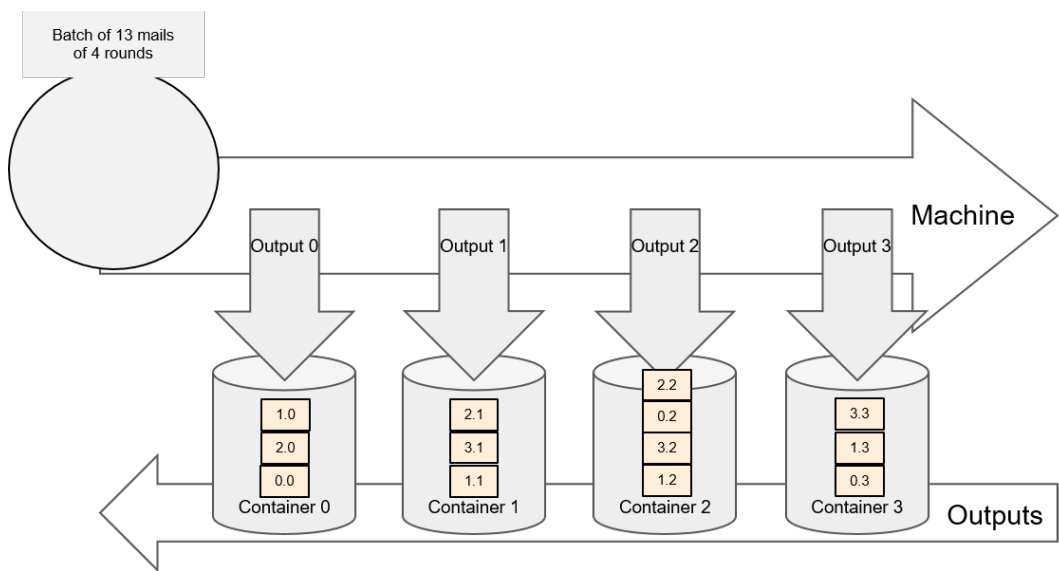


FIGURE 3 – Fin de la phase 1

II..2 Phase 2 :

Dans la deuxième phase du tri postal, chaque bac traité lors de la Phase 1 est repris individuellement afin d'effectuer un tri plus spécifique. L'objectif principal de cette étape est de répartir les courriers d'un même bac, qui correspondent à une même adresse, en fonction des tournées de distribution. Par exemple, pour le Bac 1 contenant les courriers destinés à l'Adresse 1, on identifie les courriers qui doivent être distribués dans différentes tournées, telles que la Tournée A, la Tournée B, etc. Ce tri permet de scinder chaque bac en plusieurs sous-groupes correspondant aux tournées spécifiques des facteurs. À la fin de cette phase, chaque bac initial est décomposé en plusieurs sous-ensembles triés par tournée, tout en conservant l'ordre des adresses dans les courriers. Cette organisation prépare le courrier pour la distribution finale, optimisant ainsi le processus logistique.[2]

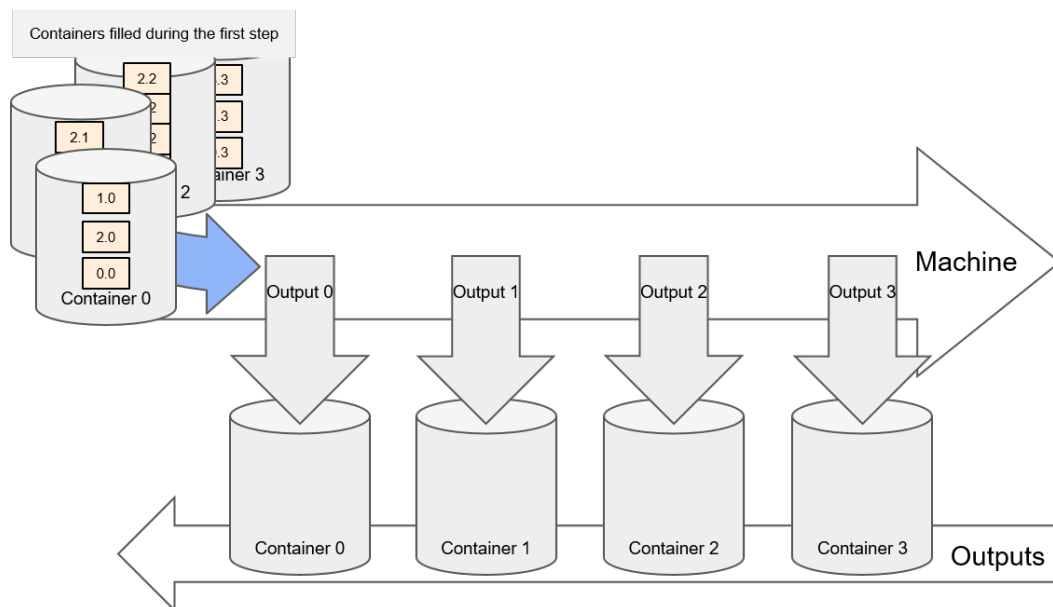


FIGURE 4 – Début de la phase 2

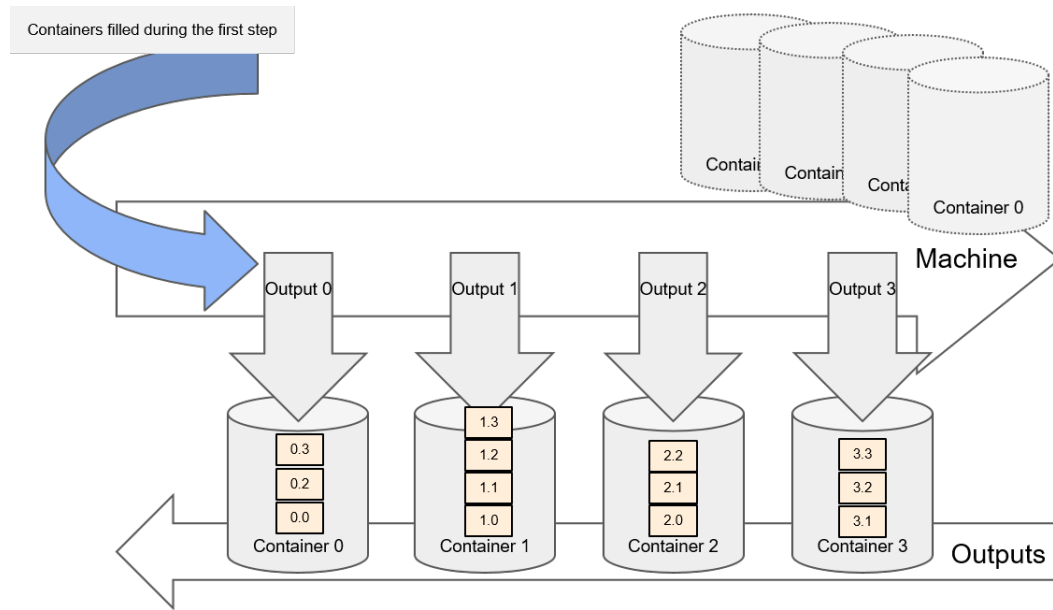


FIGURE 5 – Fin de la phase 2

Commentaires :

- À la fin de la phase 1 les courriers sont bien triés par ordre d'adresses.
- On prend les bacs triés lors de la phase 1 pour les retrier selon l'ordre des tournées
- À la fin de la phase 2 les bacs sont triés par ordre de tournées et chaque tournée est triée par ordre d'adresses

III. Présentation de la problématique et de l'objectif

III..1 Problématique :

Dans le processus de tri du courrier à La Poste, une problématique majeure réside dans l'équilibrage des charges entre les différentes sorties de la machine de tri dès la Phase 1, où les courriers sont regroupés par adresse. À cette étape, chaque sortie de la machine reçoit un volume total de courriers correspondant à plusieurs adresses, mais sans prendre en compte d'éventuels déséquilibres entre les charges des sorties. Cette situation engendre des disparités importantes : certaines sorties peuvent être surchargées, avec des volumes significativement supérieurs à d'autres. De telles inégalités compliquent les manipulations des bacs, réduisent l'ergonomie des transferts et augmentent la difficulté opérationnelle pour les opérateurs. Pour pallier ces déséquilibres dès la première phase, il devient essentiel d'optimiser la répartition des courriers par adresse entre les sorties de la machine, de manière à minimiser les écarts de charge tout en respectant les contraintes de regroupement par adresse. Ce défi d'optimisation constitue l'un des enjeux principaux abordés dans cette étude.

III..2 Objectif :

L'objectif de cette étude est de concevoir un modèle d'optimisation permettant de minimiser l'épaisseur maximale des bacs utilisés pour le tri du courrier à La Poste, tout en respectant les contraintes de regroupement des courriers par adresse. En d'autres termes, il s'agit de répartir les courriers de manière à équilibrer les charges entre les bacs tout en limitant le bac le plus chargé, afin d'éviter les surcharges qui augmentent la difficulté des manipulations et nuisent à l'ergonomie des transferts. Ce modèle doit ainsi garantir que l'épaisseur des bacs, c'est-à-dire le volume de courriers regroupés dans chaque bac, ne dépasse pas un seuil prédéfini. L'optimisation de cette répartition permet non seulement d'améliorer l'efficacité des opérations de tri et de transfert, mais aussi de faciliter le travail des opérateurs en réduisant les déséquilibres de charges et en améliorant la gestion des volumes de courriers dans chaque bac. Ce défi d'optimisation est crucial pour garantir un processus de tri du courrier plus fluide, rapide et moins sujet aux erreurs humaines.

IV. Modélisation du problème

Soit $R = \{1, \dots, |R|\}$ l'ensemble des tournées, où chaque tournée $r \in R$ est associée à un ensemble de points de distribution $s^{(r)} = (s_1^{(r)}, s_2^{(r)}, \dots, s_{|s^{(r)}|}^{(r)})$.

Chaque point de distribution $s_j^{(r)}$ dans la tournée r a une épaisseur $t_j^{(r)}$.

- $P = \{1, 2, \dots, e_{\max}\}$: l'ensemble des positions disponibles dans les bacs lors de la Passe 1.
- $P_j^{(r)} = \{j, j+1, \dots, e_{\max} - |s^{(r)}| + j\}$: l'ensemble des positions disponibles pour le $j^{\text{ème}}$ point de distribution de la tournée r .
- $x_{jk}^{(r)}$: variable binaire qui vaut 1 si le $j^{\text{ème}}$ point de la tournée r est affecté à la position k dans le bac, et 0 sinon.
- E_{\max} : l'épaisseur maximale parmi les bacs.
- T_{\max} : la capacité maximale de chaque bac.

Fonction Objectif

L'objectif est de minimiser l'épaisseur maximale parmi tous les bacs. Cela revient à minimiser E_{\max} , qui représente l'épaisseur totale du bac le plus rempli. Ainsi, l'objectif est :

$$\min E_{\max}$$

Contraintes

1. Affectation des Points de Distribution

Chaque point de distribution $s_j^{(r)}$ de la tournée r doit être affecté à **exactement une position** parmi les positions disponibles $P_j^{(r)}$. Cela se formalise par la contrainte suivante :

$$\sum_{k \in P_j^{(r)}} x_{jk}^{(r)} = 1 \quad \forall j \in \{1, 2, \dots, |s^{(r)}|\}, \forall r \in R$$

2. Une Position ne Peut Être Occupée Qu'une Seule Fois

Chaque position k dans les bacs ne peut être occupée que par **un seul point de distribution**. Cette contrainte est exprimée par :

$$\sum_{j \in U_k} x_{jk}^{(r)} \leq 1 \quad \forall k \in P, \forall r \in R$$

3. Respect des Contraintes de Précédence (Ordre des Points de Distribution)

Les points de distribution doivent respecter l'ordre dans lequel ils apparaissent dans la tournée. Si $s_j^{(r)}$ précède $s_{j+1}^{(r)}$, alors $s_j^{(r)}$ doit être affecté à une position avant $s_{j+1}^{(r)}$. Cette contrainte se formalise ainsi :

$$\sum_{k \in P_j^{(r)}} k \cdot x_{jk}^{(r)} \leq \sum_{k \in P_{j+1}^{(r)}} k \cdot x_{(j+1),k}^{(r)} \quad \forall j \in \{1, 2, \dots, |s^{(r)}| - 1\}, \forall r \in R$$

4. Capacité Maximale de Chaque Bac

L'épaisseur totale des points de distribution affectés à chaque bac ne doit pas dépasser la capacité maximale T_{\max} . Cela se formalise par la contrainte :

$$\sum_{j \in U_k} t_j^{(r)} \cdot x_{jk}^{(r)} \leq T_{\max} \quad \forall k \in P, \forall r \in R$$

5. Épaisseur Maximale parmi les Bacs

L'épaisseur totale des points de distribution affectés à chaque bac ne doit pas dépasser une valeur maximale E_{\max} . Cette contrainte est exprimée par :

$$\sum_{k \in P} \sum_{j \in U_k} t_j^{(r)} \cdot x_{jk}^{(r)} \leq E_{\max} \quad \forall r \in R$$

Variables Binaires

Les variables $x_{jk}^{(r)}$ sont définies comme suit :

$$x_{jk}^{(r)} \in \{0, 1\} \quad \forall j \in \{1, 2, \dots, |s^{(r)}|\}, \forall k \in P_j^{(r)}, \forall r \in R$$

Conclusion

L'objectif est de minimiser l'épaisseur maximale E_{\max} parmi tous les bacs, tout en respectant les contraintes suivantes :

- Chaque point de distribution est affecté à une position unique.
- Les points de distribution respectent l'ordre de la tournée.
- La capacité maximale de chaque bac est respectée.
- L'épaisseur totale de chaque bac est inférieure ou égale à E_{\max} .

Modèle Complet

Le problème peut être formalisé comme suit :

$$\min E_{\max}$$

sous les contraintes :

$$\sum_{k \in P_j^{(r)}} x_{jk}^{(r)} = 1 \quad \forall j \in \{1, 2, \dots, |s^{(r)}|\}, \forall r \in R$$

$$\sum_{j \in U_k} x_{jk}^{(r)} \leq 1 \quad \forall k \in P, \forall r \in R$$

$$\sum_{k \in P_j^{(r)}} k \cdot x_{jk}^{(r)} \leq \sum_{k \in P_{j+1}^{(r)}} k \cdot x_{(j+1),k}^{(r)} \quad \forall j \in \{1, 2, \dots, |s^{(r)}| - 1\}, \forall r \in R$$

$$\sum_{j \in U_k} t_j^{(r)} \cdot x_{jk}^{(r)} \leq T_{\max} \quad \forall k \in P, \forall r \in R$$

$$\sum_{k \in P} \sum_{j \in U_k} t_j^{(r)} \cdot x_{jk}^{(r)} \leq E_{\max} \quad \forall r \in R$$

avec les variables binaires :

$$x_{jk}^{(r)} \in \{0, 1\} \quad \forall j \in \{1, 2, \dots, |s^{(r)}|\}, \forall k \in P_j^{(r)}, \forall r \in R$$

V. Résolution du problème :

V..1 Heuristique : Procédure de rééquilibrage *i,j shift* :

Présentation de la procédure *i,j shift* :

Dans le cadre du tri du courrier, il est essentiel de répartir de manière équilibrée la charge des bacs afin d'éviter les surcharges et de garantir une distribution efficace. Pour cela, nous avons mis en place une procédure appelée **i,j Shift**. Cette méthode permet de détecter les cases (colonnes) où une surcharge est présente et de les permuter avec des cases vides ou moins chargées, tout en respectant l'ordre des adresses, cette procédure se fait ligne par ligne.

Le processus s'effectue en trois étapes :

1. **Détection des surcharges** : Pour chaque tournée (ligne de la matrice), on identifie les colonnes les plus chargées et les colonnes les moins chargées (ou vides).
2. **Déplacement des valeurs** : Si des déplacements sont possibles (c'est-à-dire qu'il existe à la fois des colonnes pleines et des colonnes vides), on choisit :
 - La colonne vide la moins chargée (cible du déplacement),
 - La colonne pleine la plus chargée (source du déplacement).On tente ensuite un échange entre ces deux positions dans la ligne en cours (tout en respectant l'ordre des adresses).
3. **Équilibrage final** : On évalue si le déplacement permet de mieux équilibrer les charges entre les colonnes, si c'est le cas, ou si l'amélioration est suffisante selon un seuil τ , le déplacement est accepté, on met à jour la matrice et on passe à la ligne suivante.

Cette procédure garantit que chaque facteur reçoit une charge mieux répartie, facilitant ainsi la collecte et la distribution du courrier. Elle constitue une étape essentielle pour améliorer l'efficacité logistique du tri postal.

Variantes de la procédure *i,j shift*

La procédure *i,j shift* peut être implémentée de différentes manières afin d'optimiser le rééquilibrage des charges dans la matrice représentant le tri du courrier. Nous avons étudié trois variantes principales :

1. **Méthode séquentielle** : Cette approche applique l'algorithme *i,j shift* ligne par ligne, en parcourant la matrice dans un ordre fixe. Pour chaque tournée, on détecte les surcharges et on les redistribue vers les cases disponibles, tout en respectant l'ordre des adresses.

2. **Méthode aléatoire sur les lignes** : Contrairement à la première méthode, celle-ci ne traite pas les lignes dans un ordre déterministe. À chaque itération, une ligne est choisie aléatoirement pour y appliquer la procédure *i,j shift*. Cette approche permet d'éviter d'introduire des biais liés à l'ordre de traitement et peut améliorer la répartition globale.
3. **Méthode totalement aléatoire** : Dans cette variante, à chaque itération, une ligne est sélectionnée aléatoirement, mais également les colonnes concernées par l'échange. Ainsi, la redistribution de la charge ne suit pas nécessairement un schéma fixe, ce qui peut conduire à une meilleure répartition de la charge sur l'ensemble de la matrice tout en conservant les contraintes d'ordre des adresses.

Ces trois stratégies offrent différentes façons de rééquilibrer les charges en fonction du niveau de contrainte et de la flexibilité souhaitée. L'efficacité de chaque approche peut être évaluée en fonction des performances obtenues sur les instances de test.

Pseudo-code de l'algorithme i,j shift (version 1) :

Algorithm 1 *i_j_shift*

Input: M : matrice courante, τ : tolérance, *historique_cout_meilleur* : vecteur historique des coûts

Output: M : matrice modifiée, *historique_cout_meilleur* : historique mis à jour

Function *i_j_shift*($M, \tau, \text{historique_cout_meilleur}$) :

```

/* Obtenir dimensions de la matrice */
(lignes, colonnes)  $\leftarrow$  taille de  $M$ 
/* Calcul des sommes par colonnes */
sommes_colonnes  $\leftarrow$  somme des éléments de  $M$  par colonne
for chaque ligne  $r$  de 1 à lignes do
    /* Identifier colonnes vides et pleines pour la ligne  $r$  */
     $E \leftarrow$  indices  $j$  tels que  $M[r, j] = 0$   $F \leftarrow$  indices  $j$  tels que  $M[r, j] \neq 0$ 
    /* Si aucun déplacement possible, passer à la prochaine ligne */
    if  $E$  est vide ou  $F$  est vide then
        | quitter la boucle for sur cette ligne
    end
    /* Choisir les colonnes à échanger */
     $q \leftarrow$  indice de  $E$  correspondant à la plus petite somme de colonne
     $k \leftarrow$  indice de  $F$  correspondant à la plus grande somme de colonne
    /* Créer une copie temporaire de  $M$  et effectuer le déplacement */
     $s_T \leftarrow$  copie de  $M$   $s_T \leftarrow$  déplacement de la valeur de la colonne  $k$  vers  $q$  sur la ligne  $r$ 
    /* Mettre à jour les sommes temporaires */
    cols  $\leftarrow$  indices entre  $q$  et  $k$  sommes_temporaire  $\leftarrow$  copie de sommes_colonnes
    for chaque colonne  $j$  dans cols do
        | sommes_temporaire[ $j$ ]  $\leftarrow$  sommes_colonnes[ $j$ ] +  $s_T[r, j] - M[r, j]$ 
    end
    /* Accepter ou non la nouvelle solution */
    if écart-type(sommes_temporaire) < écart-type(sommes_colonnes) +  $\tau$  then
        |  $M \leftarrow s_T$  sommes_colonnes  $\leftarrow$  sommes_temporaire
    end
    /* Mise à jour de l'historique des coûts */
    Ajouter écart-type(sommes_colonnes) à historique_cout_meilleur
end
return  $M, \text{historique\_cout\_meilleur}$ 

```

Affichage des résultats :

Afin d'évaluer les performances des trois stratégies proposées, nous utiliserons une fonction de fitness basée sur l'écart-type de la somme des colonnes obtenues en sortie.

Ici, nous avons effectué une série de tests pour comparer les résultats, dans un premier temps, entre les trois stratégies ij-shift : chaque stratégie est testée sur n itérations, le n étant prédéfini pour chaque instance afin d'obtenir une différenciation suffisante des trois stratégies.

Les instances sont sous la forme ($\text{nb_colonnes} \times \text{nb_lignes}$)

Nom d'instance	Écart type initial	Écart type 1	Écart type 2	Écart type 3
Instance 1 (20 \times 200)	111.4292	1.4653	2.4580	2.3885
Instance 2 (40 \times 200)	110.4119	93.8742	4.5358	11.2641
Instance 3 (80 \times 200)	107.4257	11.7104	13.2996	19.6609
Instance 4 (100 \times 200)	112.7318	16.8047	17.4372	41.6293
Instance 5 (150 \times 200)	111.9466	20.0443	24.7726	39.3213
Instance 6 (200 \times 200)	109.2833	24.9739	25.9802	50.9930

TABLE 1 – Comparaison des écarts types pour différentes stratégies sur les 6 instances fournies

V..2 Métaheuristique : Algorithme du *Recuit Simulé* :

Présentation de l'algorithme du *Recuit Simulé* :

L'algorithme du Recuit Simulé (Simulated Annealing, SA) est une méthode d'optimisation inspirée du processus physique de refroidissement lent d'un matériau chauffé. Il s'agit d'un algorithme stochastique dont l'objectif est de trouver une bonne solution dans un espace de recherche potentiellement complexe, en évitant de rester bloqué dans des minima locaux. Pour cela, l'algorithme explore des solutions voisines de la solution courante, et accepte parfois des solutions moins bonnes avec une certaine probabilité, dépendante d'un paramètre appelé température. Cette probabilité décroît au fil du temps selon une politique de refroidissement. Ainsi, le recuit simulé équilibre progressivement l'exploration de l'espace des solutions et l'exploitation des zones prometteuses, ce qui le rend particulièrement efficace pour des problèmes combinatoires.[3]

Paramètres de l'algorithme :

- **M_initiale** : la matrice initiale représentant la solution de départ (format entier).
- **T_initiale** : température initiale du recuit simulé.
- **T_finale** : température finale, seuil à partir duquel le processus s'arrête.
- **taux_refroidissement** : facteur de refroidissement, typiquement un nombre dans $(0, 1)$.
- **type_i_j_shift** : type de mouvement utilisé pour générer une solution voisine :
 - 1 : balayage séquentiel (*i_j* shift par ligne croissante).
 - 2 : balayage aléatoire sur les lignes (*i_j* shift avec **randperm**).
 - 3 : *i_j* shift totalement aléatoire (*i_j* shift avec la fonction **i_j_rand_full**).
 - * correspond aux 3 stratégies *i_j* shift

Étapes clés de l'algorithme du Recuit Simulé :

L'algorithme de recuit simulé suit les étapes suivantes :

1. Initialisation :

- Soit M_{courante} la solution initiale (matrice).
- Calculer le coût de la solution courante $\text{cout}_{\text{courant}}$.
- Initialiser la température T à une valeur T_{initiale} .
- Définir la température finale T_{finale} et le taux de refroidissement α .
- Initialiser la meilleure solution $M_{\text{meilleure}}$ avec M_{courante} et son coût $\text{cout}_{\text{meilleur}}$.

2. Boucle principale :

- Tant que $T > T_{\text{finale}}$, répéter les étapes suivantes :
 - (a) **Proposer une nouvelle solution voisine :**
 - Choisir une nouvelle solution voisine M_{voisine} à partir de la solution courante M_{courante} , par un mouvement aléatoire ou structuré (i,j Shift).
 - (b) **Calcul du coût de la solution voisine :**
 - Calculer le coût $\text{cout}_{\text{voisin}}$ de la solution voisine M_{voisine} .
 - (c) **Décision d'acceptation :**
 - Si $\text{cout}_{\text{voisin}} < \text{cout}_{\text{courant}}$ (meilleure solution), accepter M_{voisine} comme nouvelle solution courante.
 - Sinon, accepter M_{voisine} avec une probabilité $P_{\text{acceptation}} = \exp\left(-\frac{\Delta\text{cout}}{T}\right)$, où $\Delta\text{cout} = \text{cout}_{\text{voisin}} - \text{cout}_{\text{courant}}$.
 - (d) **Mise à jour de la meilleure solution globale :**
 - Si $\text{cout}_{\text{voisin}} < \text{cout}_{\text{meilleur}}$, mettre à jour $M_{\text{meilleure}}$ et $\text{cout}_{\text{meilleur}}$.
 - (e) **Réduction de la température :**
 - Refroidir la température : $T \leftarrow T \times \alpha$, où α est le taux de refroidissement.

3. Fin de l'algorithme :

- Retourner la meilleure solution trouvée $M_{\text{meilleure}}$ et son coût $\text{cout}_{\text{meilleur}}$.

Recuit simulé en pseudo code :

Algorithm 2 Recuit Simulé

Input: $M_{initiale}$, $T_{initiale}$, T_{finale} , α (taux de refroidissement), $type_i_j_shift$
Output: $sommes_colonnes$, $coût_{meilleur}$, $historique_cout_courant$, $historique_cout_meilleur$

$M_{courante} \leftarrow copy(M_{initiale})$ $sommes_colonnes \leftarrow sum(M_{courante}, axis = 1)$ $coût_{courant} \leftarrow std(sommes_colonnes)$ $M_{meilleure} \leftarrow M_{courante}$ $coût_{meilleur} \leftarrow coût_{courant}$ $T \leftarrow T_{initiale}$
 $historique_cout_meilleur \leftarrow []$ $historique_cout_courant \leftarrow []$

while $T > T_{finale}$ **do**
 if $type_i_j_shift = 1$ **then**
 for $j \leftarrow 1$ **to** $lignes$ **do**
 if $T \leq T_{finale}$ **then**
 break ; /* Sortie anticipée de la boucle */
 end
 /* Génération du voisin (type 1) */
 $M_{voisine}, sommes_colonnes \leftarrow i_j_minmax(M_{courante}, sommes_colonnes, j)$
 $coût_{voisin} \leftarrow std(sommes_colonnes)$ $\Delta coût \leftarrow coût_{voisin} - coût_{courant}$
 if $\Delta coût < 0$ **then**
 $M_{courante} \leftarrow M_{voisine}$ $coût_{courant} \leftarrow coût_{voisin}$ **if** $coût_{courant} < coût_{meilleur}$
 then
 $M_{meilleure} \leftarrow copy(M_{courante})$ $coût_{meilleur} \leftarrow coût_{courant}$
 end
 else
 $p \leftarrow exp(\Delta coût / T)$ **if** $random() < p$ **then**
 $M_{courante} \leftarrow M_{voisine}$ $coût_{courant} \leftarrow coût_{voisin}$
 end
 end
 $T \leftarrow T \times \alpha$ /* Mise à jour des historiques */
 $historique_cout_meilleur.push(coût_{meilleur})$
 $historique_cout_courant.push(coût_{courant})$
 end
 else
 if $type_i_j_shift = 2$ **then**
 /* Version aléatoire des lignes (type 2) */
 for $j \in randperm(lignes)$ **do**
 /* Même logique que le type 1, avec ordre aléatoire */
 $M_{voisine}, sommes_colonnes \leftarrow i_j_minmax(M_{courante}, sommes_colonnes, j)$
 [...] /* Partie identique à la boucle type 1 */
 end
 else
 /* Version totalement aléatoire (type 3) */
 for $j \in randperm(lignes)$ **do**
 $M_{voisine}, sommes_colonnes \leftarrow i_j_rand(M_{courante}, sommes_colonnes, j)$
 [...] /* Partie identique à la boucle type 1 */
 end
 end
 end
end
return $sommes_colonnes$, $coût_{meilleur}$, $historique_cout_courant$,
 $historique_cout_meilleur$; /* Optionnel */

VI. Résultats Recuit Simulé et comparaison avec ij_Shift

TABLE 2 – instance : "trafic_20_200_35_1.xlsx"

Itérations	Initiale	ijshift1	Recuit 1	ijshift2	Recuit 2	ijshift3	Recuit 3
456	111.4292	75.1675	75.6115	75.0154	74.8911	84.2161	87.1782
2298	111.4292	5.2656	6.5647	7.7690	6.9615	39.1184	46.3333
9206	111.4292	1.4654	1.0954	1.7290	0.9403	3.0505	22.8644

TABLE 3 – Instance : "trafic_40_200_35_1.xlsx"

Itérations	Initiale	ijshift1	Recuit 1	ijshift2	Recuit 2	ijshift3	Recuit 3
456	110.4119	94.7204	89.1128	89.4282	89.2807	95.1369	94.8781
2298	110.4119	93.8742	40.0315	42.0946	40.7193	62.9812	63.1210
9206	110.4119	93.8742	1.1140	6.8358	1.3436	23.6429	28.7172

TABLE 4 – Instance : "trafic_80_200_35_1.xlsx"

Itérations	Initiale	ijshift1	Recuit 1	ijshift2	Recuit 2	ijshift3	Recuit 3
456	107.4257	96.3865	96.1748	96.4955	96.3757	100.4207	100.5233
2298	107.4257	68.4536	67.5294	68.2897	67.3928	81.0008	80.1999
9206	107.4257	17.5916	9.3521	19.4043	9.9706	47.1815	45.0709

TABLE 5 – Instance : "trafic_100_200_35_1.xlsx"

Itérations	Initiale	ijshift1	Recuit 1	ijshift2	Recuit 2	ijshift3	Recuit 3
456	112.7318	103.2296	103.3033	103.4865	103.1550	106.7034	107.0999
2298	112.7318	76.2539	76.9842	77.9379	77.0656	89.0667	88.7956
9206	112.7318	25.7179	19.4821	27.1557	18.7869	58.2376	55.0625

TABLE 6 – Instance : "trafic_150_200_35_1.xlsx"

Itérations	Initiale	ijshift1	Recuit 1	ijshift2	Recuit 2	ijshift3	Recuit 3
456	111.9466	105.1882	105.1032	105.3199	105.2088	108.5398	107.9734
2298	111.9466	86.0562	85.2079	86.9088	85.3311	95.9437	94.6725
9206	111.9466	45.6268	35.4307	46.6529	35.6840	69.3452	65.1174

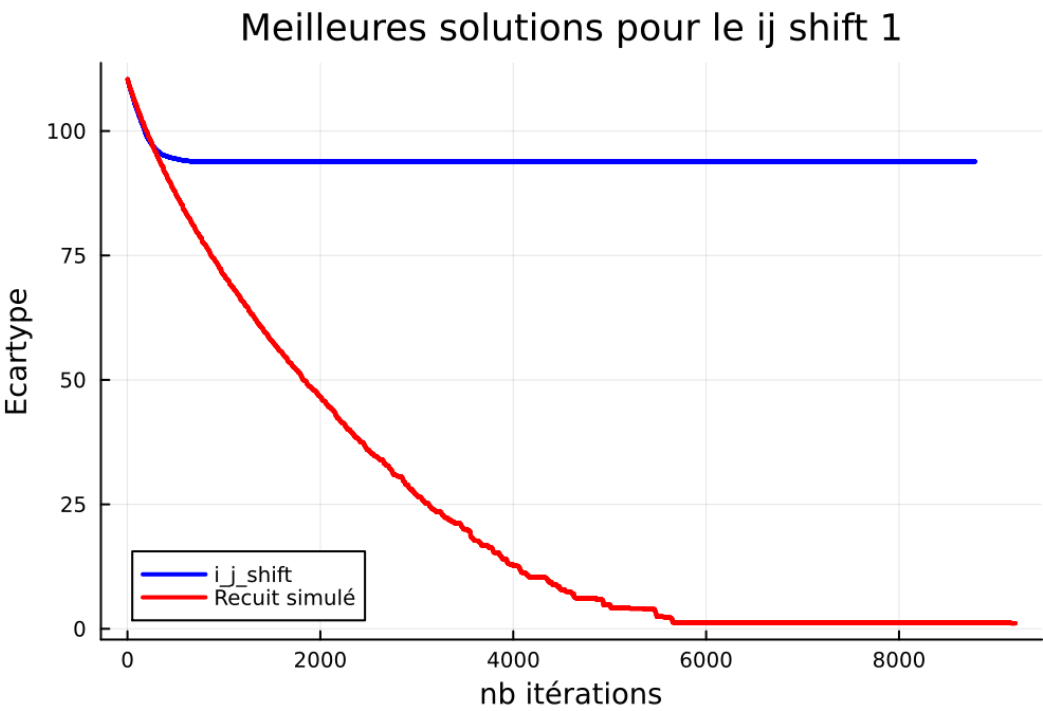
TABLE 7 – Instance : "trafic_200_200_35_1.xlsx"

Itérations	Initiale	ijshift1	Recuit 1	ijshift2	Recuit 2	ijshift3	Recuit 3
456	109.2833	104.6138	104.3791	104.4291	104.4316	106.6802	106.2189
2298	109.2833	90.2252	88.8891	90.4823	88.9980	97.6441	96.2733
9206	109.2833	59.1260	51.0635	59.8167	50.9330	76.7064	71.4645

Note : Ces expérimentations ont d’abord été réalisées sur le recuit simulé avec des paramétrages différents, pour ensuite extraire le nombre d’itérations réalisées, puis appliquer ce même nombre aux i-j shifts (afin d’avoir une base de comparaison identique).

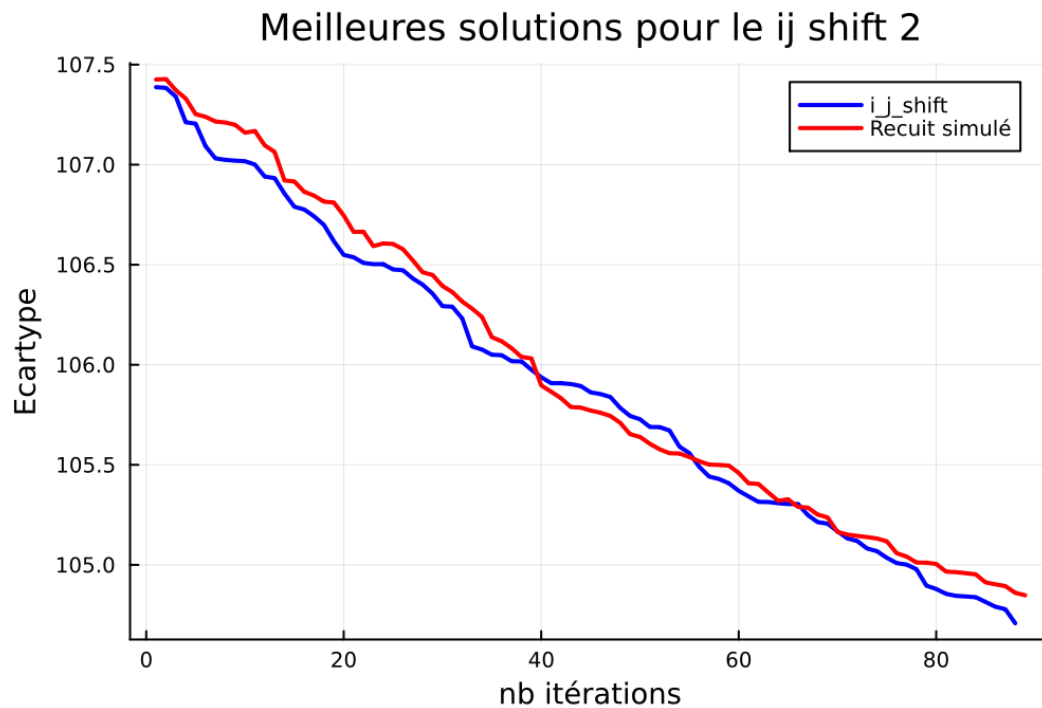
VII. Graphiques

Quelques graphiques, malheureusement le nombre d’itérations est tellement important qu’il est compliqué de visualiser correctement les changements de solution.

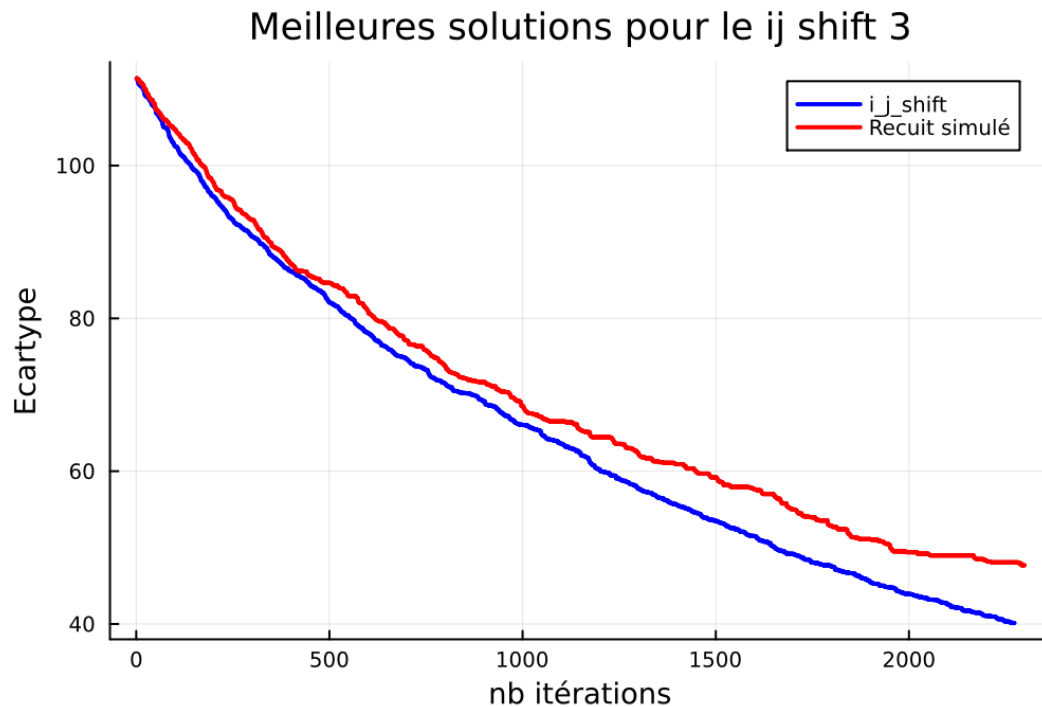


On remarque cependant sur ce premier graphique la nette amélioration du Recuit Simulé par rapport au ij-Shift (instance "trafic_40_200_35_1.xlsx").

En effet ij-Shift stagne très vite contrairement au Recuit Simulé qui continue de descendre.



On remarque également avec ce second graphique que sur toutes les instances la tendance est similaire entre ij-Shift et Recuit Simulé, qui après analyse est logique car au début l'écart type (donc le déséquilibre entre les bacs) est tellement important que quasiment toutes les solutions sont améliorantes. (instance "trafic_80_200_35_1.xlsx")



On remarque sur ce dernier graphique que recuit simulé n'est pas toujours améliorant, ici sur la troisième stratégie du ij-Shift on obtient une moins bonne solution sur le Recuit Simulé (surement dû au côté trop aléatoire de cette dernière). (instance "trafic_20_200_35_1.xlsx")

VIII. Conclusion

Comme le montrent les résultats précédents, le recuit simulé apporte une amélioration dans la majorité des cas (la troisième stratégie constituant la principale exception, produisant généralement les pires solutions parmi les i-j shifts). On en déduit que pour les stratégies 1 et 2, le recuit simulé apporte une réelle plus-value quant à l'équilibrage des bacs des employés. Il permet d'éviter de s'engouffrer dans des optimums locaux, avec une convergence peut-être légèrement plus lente initialement, mais qui s'avère finalement très utile. En résumé, notre TER prouve l'utilité du recuit simulé dans cette application concrète du problème de La Poste !

Remerciements

Nous tenons à exprimer notre profonde gratitude à notre professeur encadrant, Evgeny GUREVSKY, pour son accompagnement, ses conseils et sa disponibilité tout au long de ce projet. Son encadrement nous a permis d'avancer avec confiance et rigueur.

Bibliographie

- [1] *La Poste*, Wikipédia.
Disponible via : [https://fr.wikipedia.org/wiki/La_Poste_\(entreprise_fran%C3%A7aise\)](https://fr.wikipedia.org/wiki/La_Poste_(entreprise_fran%C3%A7aise))
- [2] *Groupe La Poste*, mai 2005.
Disponible via : <http://ekldata.com/ItuAuUWhg2DT0Nfwrw53DGMGvac.pdf>
- [3] S. Kirkpatrick, C. D. Gelatt, Jr., M. P. Vecchi
Optimization by Simulated Annealing
Disponible via : <https://www2.stat.duke.edu/~scs/Courses/Stat376/Papers/TemperAnneal/KirkpatrickAnnealScience1983.pdf>
- [4] Damien Lovato
Développement de critères d'optimisation pour la planification de tâches dans une chaîne d'assemblage aéronautique
Disponible via : <https://theses.hal.science/tel-04098033v1/document>
- [5] Siwar Arbi, Audrey Cerqueus, Evgeny Gurevsky, Alexandre Dolgui, Ali Siadat
Simulated annealing for a bi-objective-based scalable balancing of a reconfigurable assembly line under uncertainty and limited resources
Disponible via : <https://hal.science/EC-NANTES/hal-04199608v1>
- [6] Emmmanuelle Amann
Optimisation des plans de tri courrier de La Poste. Sorbonne Université, 2022.