

**République Algérienne Démocratique et Populaire**

**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**



**Université Abou bekr Belkaid – Tlemcen**

**Faculté de Technologie**

**Département de Génie Electrique et Electronique**

**Filière : Génie Industriel**

## **Projet de Fin d'Etudes**

**Master : Génie Industriel**

**Option : Productique**

**Intitulé :**

**Modélisation et optimisation de l'espace en utilisant le « Nesting »**

**Réalisé par :**

- AZZOUZ Hamza

- MLATA Ibrahim El Khalil

-YAHIAOUI Zakaria

**Devant le jury :**

**Président : GUEZZEN AMINE**

**MCB**

**UAABT**

**Encadrant : MALIKI Fouad**

**MAA**

**ESSAT**

**Examineur : BENNEKROUF Mohammed**

**MCB**

**ESSAT**

**Examineur : HADRI Abdelkader**

**MAA**

**UAABT**

**Année Universitaire : 2016/2017**

# *Remerciements*

*Nous tenons tout d'abord à remercier Dieu le tout puissant et miséricordieux, qui nous a donné la force et la patience d'accomplir ce modeste travail. En second lieu, nous remercions nos enseignants et notre encadreur depuis le primaire jusqu'au supérieur, car, si nous soutenons aujourd'hui, c'est grâce à leurs fonctions éminentes dans notre réussite, la transmission de leurs connaissances et leurs savoir-faire. Enfin, nous remercions aussi toute personne qui a participé de près ou de loin à la réalisation de ce travail Nous tenons à remercier notre encadreur :*

***Mr. MALIKI Fouad** pour leur précieux conseils durant toute la période du travail. Nos remerciements à **Mr GUEZZEN Amine** pour vouloir accepter de présider le jury. Nos vifs remerciements vont également à **Mr BENNEKROUF Mohamed** et **Mr HADRI Abdelkader** pour l'intérêt qu'ils ont porté à notre recherche en acceptant d'examiner notre travail.*

## *Dédicace*

*Je voudrais dédier ce modeste travail A mes très chers Parents qui m'ont tant soutenu et encouragé dans tous les domaines et surtout pour réaliser ce mémoire. A la mémoire de mes grands-parents et tous les chers qui nous ont quittés, allah 'yarhamhoum A mon frère jumeau Youcef, mon grand frère Mohammed, et à ma soeur Meriem. A toute la famille, grand et petit. A toute la famille du Génie Productique.*

### **Hamza**

*Je voudrais dédier ce travail à mes parents que j'aime tant, mon Père l'idole de ma vie, ma Mère la plus douce des mères. A mes chères frères Noureddine et Amine. A mes deux chères soeurs Soumia et Zahira et aussi a mes soeurs Ikram et Meriem . A toute la famille grande et petite, et à tous mes amis*

### **Ibrahim**

*Je voudrais dédier ce modeste travail A mes très chers Parents qui m'ont tant soutenu et encouragé dans tous les domaines et surtout pour réaliser ce mémoire . A mes chers frères Ilyes et Ishak. A mes chères soeurs Samia, Meriem et Khawla. A mes chers Oncles Y.Ahmed et B.Ismain. A toute la famille BEROUILA Et la famille AL KHAYAT et a tous mes amis*

### **Zakaria**

# Table des matières

INTRODUCTION GENERALE .....	1
<b>CHAPITRE 1 .....</b>	<b>2</b>
<b>Le monde de Nesting et Packing .....</b>	<b>2</b>
1.1 Introduction:.....	2
1.2. Problèmes de Découpe et Placement : .....	2
1.2.1 Définition : .....	2
1.2.2 Classification des problèmes de découpe /packing : .....	3
1.2.3. Autre Critère de classification : .....	3
1.3.Type de problème : .....	7
1.3 .1. Strip packing : (problème en forme de bande) .....	7
1.3 .2. Le problème du BIN PACKING : .....	7
1.4 Problèmes de Placement et NP-Complétude (complexité):.....	9
1.4.1. Heuristique basé sur le positionnement : .....	10
1.4.2. Problèmes de « Nesting » : .....	11
1.5. Conclusion: .....	12
<b>CHAPITRE 2 .....</b>	<b>13</b>
<b>Applications industrielles .....</b>	<b>13</b>
2.1. Introduction : .....	13
2.2. Classification des matériaux : .....	13
2.2.1. L'industrie des tôles : .....	13
2.2.2. L'industrie de textile : .....	15
2.2.3. L'industrie du cuir : .....	15
2.2.4. Industries avec des problèmes de coupe rectangulaires : .....	16
2.3. Mode de découpe : .....	17
2.3.1. Guillotine : .....	17
2.3.2. Non-Guillotine : .....	17
2.4. Le role de Nesting : .....	19
2.5. Logiciels de Nesting : .....	20

2.6. Avantages et efficacité :	21
2.7. Conclusion :	22
<b>CHAPITRE 3</b> .....	<b>24</b>
<b>Problème avec des pièces rectangulaires</b> .....	<b>24</b>
3.1. Introduction :	24
3.2. Revue de Littérature :	24
3.3. Définition du problème :	25
3.4. Résolution du problème :	27
3.4.1. Algorithme heuristique Least Wasted First :	27
3.4.2. Processus de LWF :	33
3.5. Vision du projet :	42
3.5.1. Création de logiciel :	42
3.5.2. Algorithme avancée (Machine learning):	43
3.5.3. Cloud computing et Big Data:	43
3.6. Conclusion:	43
CONCLUSION GENERALE .....	43
REFERENCES BIBLIOGRAPHIQUES .....	44

# Liste des figures

## Chapitre 1

Figure 1.1.Placement orthogonale, découpe guillotine

Figure 1.2.Placement orthogonale, découpe non-guillotine.

Figure 1.3.Placement non orthogonale.

Figure 1.4. Nesting des formes irrégulières

Figure 1.5.Classification de Hooper.b basé sur la classification de Dychoff.1990

Figure 1.6. Structure de classification pour les problèmes de SPP.

Figure 1.7. Exemple d'emballage en bande inclus des formes en rectangle.

Figure 1.8. Exemple de compréhension du probleme de BinPacking .

Figure 1.9. L'heuristique « Bottom Left » selon différent stratégies

Figure 1.10. Schéma explicatif pour le processus de Nesting

## Chapitre 2

Figure .2.1. exemple du layout dans L'industrie des tôles

Figure.2. 2. Consommation annuelle de matières premières

Figure .2.3. Exemple du layout dans L'industrie de textile

Figure .2.4. Exemple du layout dans L'industrie du cuir

Figure .2.5. Solution numérique dans l'industrie du cuir

Figure .2.6. Mode guillotine

Figure .2.7 . Mode non-guillotine

Figure .2.8. Exemple du layout guillotinable

Figure .2.9. Exemple du layout non-guillotinable

Figure .2.10. Découpe laser dans l'industrie des tôles

Figure .2.11. Découpe jet d'eau

Figure .2.12. Poinçonneuse multi-têtes

Figure .2.13. Le Nesting dans le domaine de fabrication

Figure .2.14. Interface d'un logiciel de nesting

## Chapitre 3

Figure .3.1. Le rectangle  $R_i$  dans le system cartésien

Figure .3.2.position faisable

Figure .3.3. Mauvaises position.

Figure .3.4. Changer l'enveloppe.

Figure .3.5. good pack

Figure .3.6 .Table de travail avec deux positions

Figure .3.7 .Liste des rectangles

Figure .3.8. Table de travail et la liste des rectangles

Figure .3.9 .Placement du premier rectangle

Figure .3.10 . Recherche de position

Figure .3.11 .Placement du deuxième rectangle

Figure .3.12.Recherche d'une nouvelle position

Figure .3.13.éliminé la mauvaise position

Figure .3.14.Placement du 3eme rectangle

Figure .3.15.Placement le 4eme rectangle

Figure .3.16.Plan final

## **Liste des tableaux**

Tableau 1.1. Exemple des formes géométrique et l'application industrielle associée

# INTRODUCTION GENERALE

Les problèmes de Packing 2D sont une variété de problèmes combinatoires avec un très grand espace de solutions qui ne peut normalement être exploré de manière exhaustive.

Une série d'approches hybrides a été développée pour résoudre les problèmes packing 2D pour des formes rectangulaires et irrégulières et les problèmes bin packing 2D en générale. Comme la littérature récente encourage l'utilisation d'algorithmes génétiques en particulier, ainsi que d'autres méthodes méta-heuristiques pour leur solution, l'objectif principal de ce projet consiste en l'application d'une heuristique différente pour les problèmes avec des pièces rectangulaire. Afin de bien définir le thème du mémoire, nous avons décidé de ramener un problème industriel, ce qui est le cas pour les problèmes appelé « Nesting », ces problème occupe la branche des problèmes de « Découpe et Placement ».

## **Objectif**

Modéliser le problème (pièces rectangulaires) et le résoudre avec une heuristique bien définie. Nous nous sommes focalisé sur l'identification des types de problèmes en étudiant le degré de complexité de chaque sous problèmes et le cadre scientifique mathématique ou informatique de chaque type de problèmes ainsi que les différentes applications industrielles existantes. Notre mémoire est organisé en trois chapitres comme suit :

### **Chapitre 1 : Le monde de Packing et Nesting**

Dans ce chapitre, on a définie en gros ce qui est les problèmes de packing 2D, un petit historique pour la compréhension de la démarche de classification, les types de problèmes existant ainsi que leurs complexités mathématiques. Enfin, nous avons présenté le problème de Nesting.

### **Chapitre 2 : Applications industrielles**

Ce chapitre présente les applications industrielles du problème de « Nesting » avec un œil sur les contraintes imposées du domaine de fabrication et découpe de pièce.

### **Chapitre 3 : Problème avec des pièces rectangulaires**

Le dernier chapitre définit le problème étudié avec ses contraintes, les méthodes de résolution proposées sont présentées avec la définition en trois étape d'une heuristique appropriée pour le problème du nesting de pièces rectangulaires.

Ce mémoire est terminé par une vision future du projet et une conclusion générale qui exposera les perspectives envisagées.

# CHAPITRE 1

## Le monde de Nesting et Packing

### 1.1 Introduction:

Un individu s'affronte régulièrement a des circonstances ou son talent est requis pour trouvé un bon arrangement des ses vêtement dans une valise ou trouver un bon emplacement pour les produits alimentaires au frigo. L'être humain a une capacité intuitive de résoudre les problèmes de placement et remplissage d'espace, toutefois dans un environnement industriel très active, les problèmes de placement des produits ou pièce sont parfois similaire et d'autre diverse. L'utilisation des manière manuel d'où la méthode visuel pour résoudre ces problèmes de placement (Packing) n'est pas efficace ou efficient dans le but de faire un bénéfice d'argent et de temps car les employé n'ont pas le même réflexe ou le degré de compétence a trouver des solutions satisfaisante, pour cela les problèmes de placement appelé « Nesting » dans le monde industriel sont automatisé en créant des solution logiciels qui facilite la tache au employé a trouvé le meilleur emplacement des pièce a découpe ou stocker en basant sur la puissance des algorithmes et les heuristiques. Ce type de problèmes est traité dans plusieurs disciplines tel que : science de management, Ingénierie, l'informatique, recherche opérationnelle, l'industrie.... La dénomination des problèmes peuvent varier ou changé entre les disciplines mais il faut savoir qu'il fait partie des problèmes combinatoire retrouvé dans la recherche opérationnelle. Le processus d'allocation est de maximiser l'utilisation et donc minimiser le matériel gaspillé. Le premier problème de ce type apparait dans l'industrie de papier en 1939 par l'économiste russe Kantorovitch. Gilmore et Gomory ont introduit en 1961 une technique de génération motifs (forme) pour résoudre un problème de découpe en une dimension en utilisant la programmation linéaire. [1.3]

### 1.2. Problèmes de Découpe et Placement :

#### 1.2.1 Définition :

Les problèmes de placement sont des problemes connu sous le nom « Cutting & Packing problems » (Découpe et placement), Les problèmes de Placement sont des problèmes d'optimisation qui cherche à trouver un bon arrangement de plusieurs articles (pièce) dans un espace (conteneur) plus grand.[1.8] Ces problèmes combinatoires sont généralement NP-Difficiles

ou NP-Complet [Garey & Johnson, 1979]. Ce type de problème est cité dans plusieurs domaines d'industrie (industrie de papier, Cuir .....).

### 1.2.2 Classification des problèmes de découpe /packing :

Dychoff.1990 a proposé une classification qui facilite le partage d'informations entre plusieurs domaines de recherche et plusieurs disciplines. Il a proposé une typologie de problèmes de packing.[1.4] . En générale les problèmes de packing appartiennent au domaine de la géométrie et l'informatique combinatoire. Dyckhoff et Finke ont développé un schéma de classification pour les problèmes de découpe et placement, il a identifié 4 caractères pour ses classifications :

1. le nombre de dimension de problème (1D, 2D,..., N Dimension)
2. Type de tâche : tous les objets et une sélection de bins, ou bien ; une sélection d'objet et plusieurs bins (conteneur).
3. Caractéristique des bins : plusieurs bins, des bins de taille identique, des bins de taille différente, un seul bins ...
4. Caractéristique des objets : objet (pièces) de forme identique, peu d'objet de forme identique, objet de forme différente, objet de forme relativement différente.

### 1.2.3. Autre Critère de classification :

L'objectif des problèmes de packing est de trouver le meilleur emplacement pour des objets dans un conteneur sans chevauchement, la difficulté de ces problèmes est fortement relié à la forme géométrique des objets à emballer ou remplir. En ce qui concerne la forme, deux types de problèmes sont distingués : **forme régulière** (cercle, rectangle....) , **forme irrégulière**. Dans le problème de placement en deux dimensions, la disposition suivante peut être distinguée sur la base de la géométrie des articles ajoutés, dans le cas des articles avec une forme régulière, l'emballage peut être orthogonal c'est-à-dire que la découpe serait parallèle au côté de la feuille (conteneur) régulière, si non (la figure 1.3) montre le placement non-orthogonal des articles.

#### 1.2.3.1 Type de découpe :

Il existe deux types de découpe à considérer dans la classification des problèmes, découpe guillotine et non-guillotine. Dans la découpe guillotine on coupe d'un côté de la feuille de travail vers l'autre côté, la découpe serait une droite, cette technique est très utilisée dans l'industrie du verre pour les caractéristiques de ce matériau à ne pas supporter des angles de découpe. Les outils

de découpe ont aussi un rôle dans le choix du type de découpe [1.4] par exemple si une entreprise veut découper des CardBoard, une lame tranchante peuvent être utile, mais si les forme a couper était bien placer l'une a coté de l'autre, une découpe guillotine serait nécessaire.

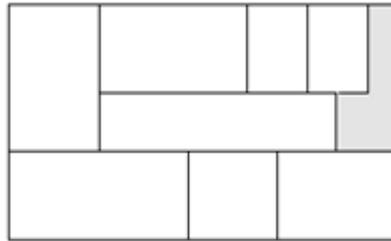


Figure 1.1.Placement orthogonale, découpe guillotine

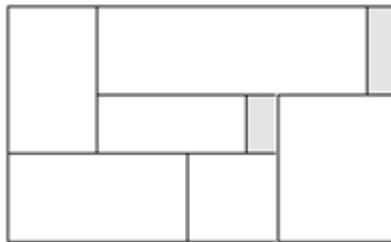


Figure 1.2.Placement orthogonale, découpe non-guillotine.

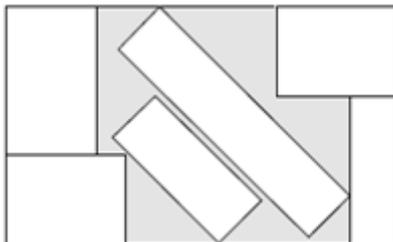


Figure 1.3.Placement non orthogonale.



Figure 1.4. Nesting des formes irrégulières

Exemple des formes géométrique aux problèmes de packing en 2D :

Forme géométrique	Exemple d'application
<b>Ellipse, cercle</b>	Chargement des conteneurs avec tuyaux
<b>Polygone : Convexe, Concave</b>	Formation de pièce irrégulière dans l'industrie métallurgique.
<b>Forme-libre : des objets composés de segments de ligne droite et courbe</b>	Mis en page des marqueurs dans l'industrie de textiles.
<b>Forme avec enceinte complète ou partielle</b>	Fabrication des bateaux.

Tableau 1.1. Exemple des formes géométrique et l'application industriel associé

Ces problèmes peuvent être séparés en deux catégories ; une catégorie avec une dimension spatiale et une autre catégorie pour les dimensions non-spatiales. La première catégorie est destinée pour les problèmes de placement et de chargement ou la complexité peut aller jusqu'à une 3ème dimension. La deuxième catégorie de dimension non-spatiale inclut des cas où la dimension est temporaire (temporel) ou le cas de la mémoire physique d'une mémoire d'un ordinateur. [1.2]

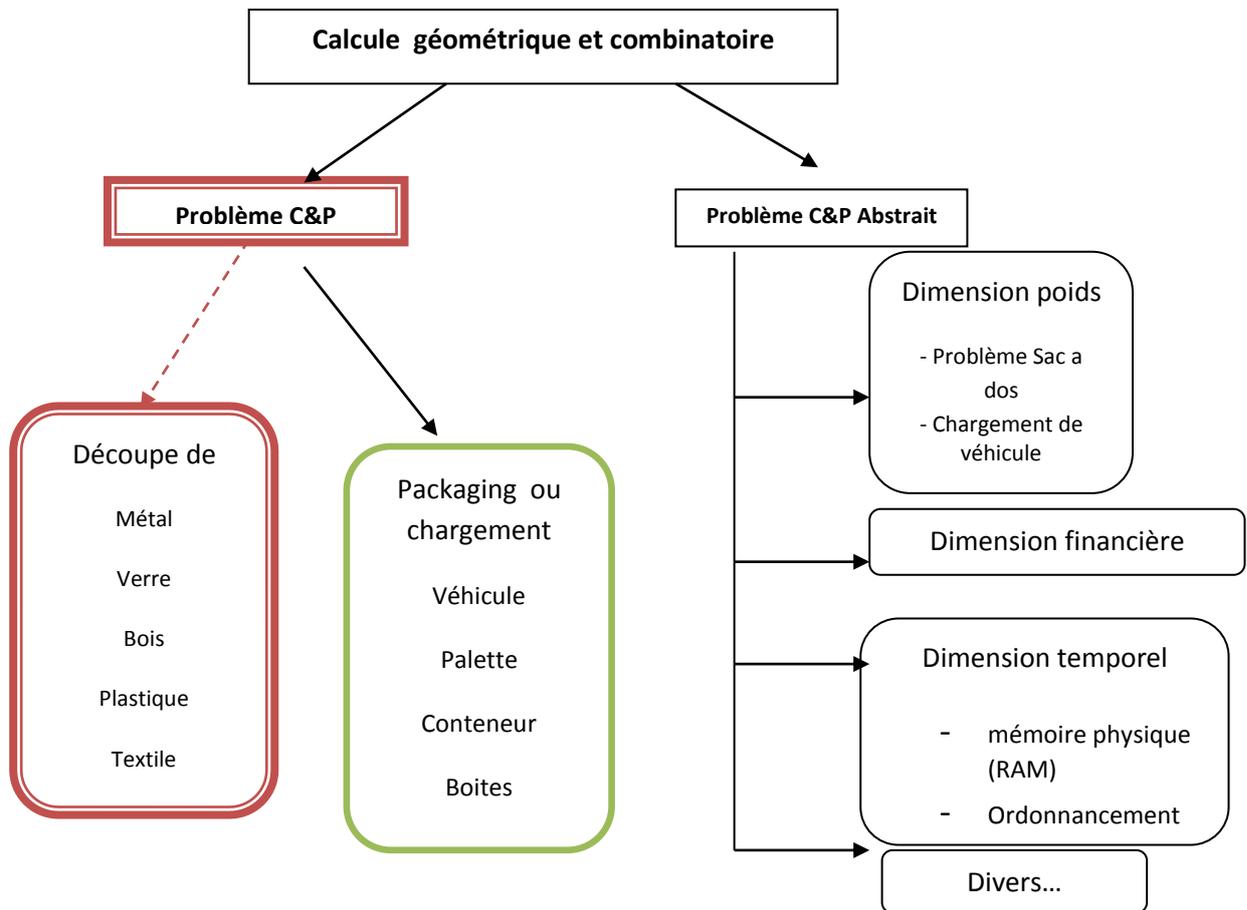


Figure 1.5. Classification de Hooper.b basé sur la classification de Dychoff.1990

Malgré que Dychoff a publié une autre classification en 2004 considérant d'autres critères de forme géométrique comme une réponse à des critiques par certains collaborateurs pédagogiques ou commerciaux à ne pas suivre le développement surtout que dans l'application plusieurs problèmes de découpe se distinguent par la difficulté de la forme géométrique ou par la technique avancée (ex : contrainte de rotation, résistance matérielle, ...) qui ne marche plus avec la pensée de résolution. Une autre classification de référence de Washer.2007 suit celle de Dychoff.1990 montrée dans le schéma ci-dessous [1.10]

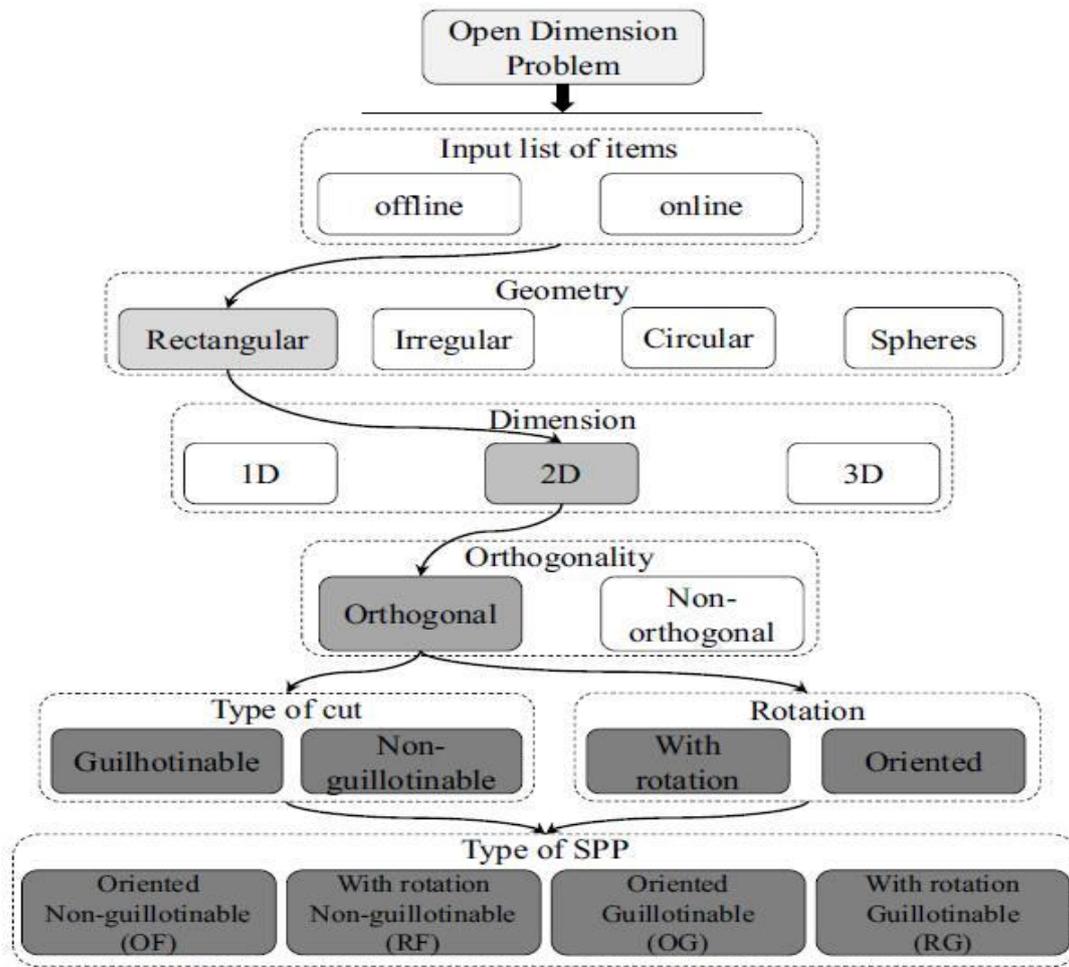


Figure 1.6. Structure de classification pour les problèmes de SPP.

**1.2.3.2. Point clé classification washer.2007 :**

Ce qu'on va retenir de ce schéma pour la suite est que le problème a traité de ce document est problème orthogonale avec des pièces rectangulaires en 2D, les entré sont offline, c'est-à-dire que le stock de pièces nécessaire sont connu d'avance, on n'a pas un système online qui délivre des pièces multi forme a chaque instant selon le type de découpe et la contrainte de rotation, les types de problème de découpe vont être différencié et nommé sur la base de la classification de washer.2007 qu'on va continuer vers les problèmes de packing en domaine de fabrication appelé « Nesting ».

### 1.3.Type de problème :

#### 1.3 .1. Strip packing : (problème en forme de bande)

Le problème d'emballage en bande qui persiste dans l'industrie de textile et papier vise à réduire la hauteur de la disposition.

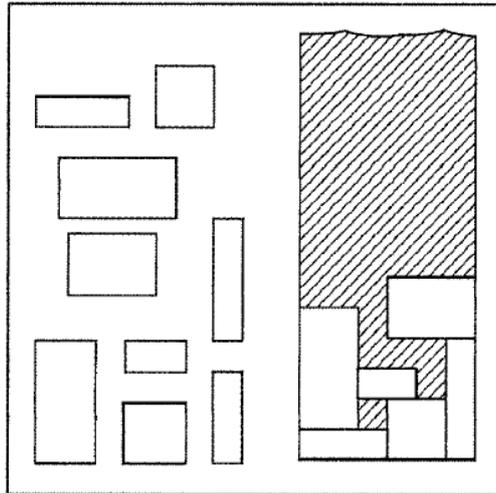


Figure 1.7. Exemple d'emballage en bande inclus des formes en rectangle.

#### 1.3 .2. Le problème du BIN PACKING :

Supposons que nous ayons  $n$  objets, chacun d'une taille donnée, et des boîtes de même capacité, en veut ranger les objets dans des boîtes, en utilisant le moins de boîtes possible. La contrainte à respecter bien évidemment est que la taille totale des objets affecté a une boîte ne doit pas dépasser ça capacité. Il existe plusieurs version du problème avec une seul dimension ou multidimensionnel.

Le problème de Bin packing est formulé dans le « chapitre 18 » dans le livre d'optimisation combinatoire 'théorie et algorithmes' comme preuve de ça complexité NP. Le problème apparait dans les branches d'applications : fabrication industrielle, chargement de véhicule, Ordonnancement, fabrication des circuits intégrés ...

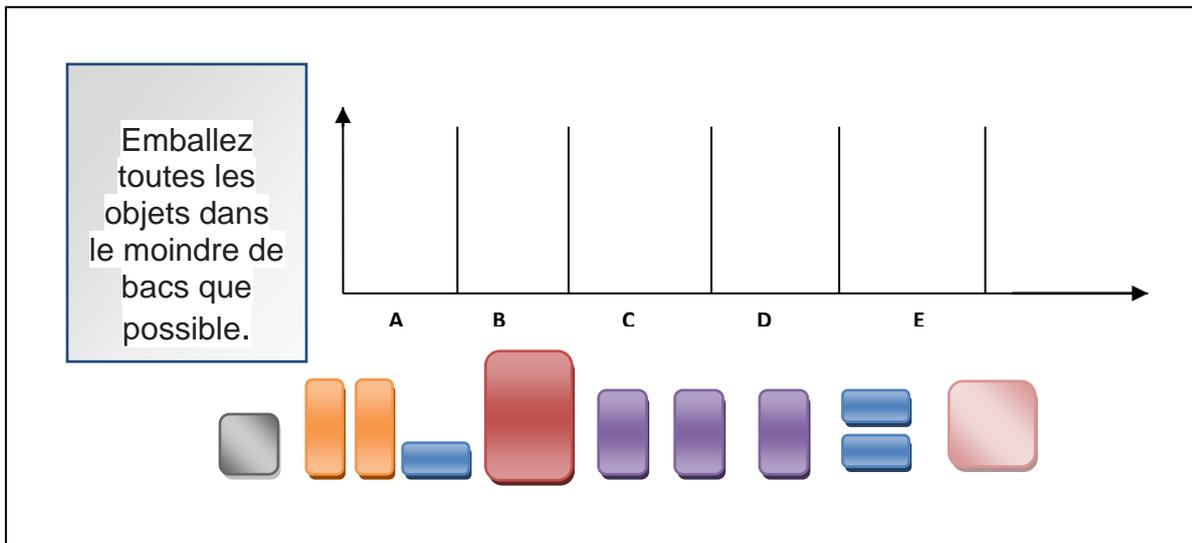


Figure 1.8. Exemple de compréhension du problème de BinPacking .

Dans l'application industrielle, le problème de BIN PACKING apparaît en plusieurs formes, selon l'objective, plusieurs type de problème sont distingué. Dans une échèles industriel plus grande, ils peuvent y'avoir un type de problème qui se compose de deux type combiné ou d'une multitude des sous problèmes basique :

#### **A-Problème de perte de matière en découpe :**

Ce qui concerne l'attribution de la liste de commande par apport a la plaque en stock disponible. Le « problème de perte de matière » (TLP) est l'un des problèmes les plus difficiles dans le contexte de la recherche d'optimisation. Il vise à déterminer le motif de coupe optimal d'un certain nombre d'éléments de différentes longueurs à partir d'un stock de matériaux de taille standard pour répondre aux exigences des clients selon lesquels le gaspillage dû à la perte de coupe est minimisé.[1.6]

Exemple très récurant dans l'industrie de papier ou un ensemble de bobines de papier produit doit être coupé à partir de bobines en papier brut. Le problème est par nature un problème combinatoire difficile. Peut-être le problème le plus difficile à côté de la combinatoire, c'est le fait que l'usine de conversion de papier doit s'adapter à la fois aux largeurs spécifiées par le client et aux largeurs de papier brut livrées à partir d'une autre usine qui fabrique le brut. Ce fait rend difficile l'évitement des pertes matérielles pendant le processus.[1.7]

**B-Problème d'assortiment :**

implique de déterminer lequel des ensembles possibles de tailles ou de qualités de certains produits devrait être stocké lorsqu'il n'est pas possible ou souhaitable de stocker tous et de la substitution dans un sens (plus grand pour les plus petites ou de meilleure qualité pour les bas-Qualité) est possible à un certain coût.[1.9]

**C- Problème de découpe dimensionnel (cutting stock problem):**

Découper des pièces avec un ordre défini depuis une feuille de travail. Ce problème peut être divisé en deux sous problème : problème de perte de matière (Détermination du motif de coupe pour minimiser les déchets) et problème d'assortiment (déterminer lesquels des 'plaques' (surface conteneur : drap, plaque, feuille....) a conserver en stock.

**D-Problème de chargement :**

Décris le processus d'ajustement d'un nombre maximal de boîtes sur une palette ou dans un récipient. Le problème de chargement de la palette peut être considéré du point de vue du fabricant où des boîtes identiques doivent être chargées sur une palette, ainsi que du côté du distributeur, où La palette doit être emballée avec des articles non identiques.

**E- Problème de sac a dos :**

Le problème consiste à remplir un sac a dos sans dépasser la contrainte de poids d'articles ajouté, et selon l'importance de chaque article, d'où l'objective d'ajouter le maximum d'article pour pouvoir atteindre une valeur max d'importance.

**F- Problème de placement orthogonal :**

Les rectangles à placer dans le conteneur doivent être parallèle à l'axe vertical ou horizontal.

**G-Problème de Nesting :**

Généralement des pièces de forme irrégulière, utilisé dans la fabrication navale.

**1.4 Problèmes de Placement et NP-Complétude (complexité):**

La complexité d'un problème est la complexité minimale dans le pire des cas d'un algorithme qui les résout. C'est souvent la complexité en temps qu'on considère mais on peut s'intéresser a d'autre mesure comme par exemple la complexité en espace. Les problèmes de

classe P sont des problèmes qui peuvent être résolus dans un temps polynomial, ces problèmes sont généralement faciles ou faisables, mais ça ne veut pas dire nécessairement que le problème doit être résolu dans un temps faible. En tant que problème d'optimisation combinatoire, plusieurs types de problèmes de Placement (packing) peuvent être résolus par des approches exactes (l'approche fournit une solution optimale garantie, généralement basée sur des modèles de programmation mathématique), des méthodes d'approximation (heuristiques et métaheuristiques) ou des méthodes hybrides (par exemple, les métaheuristiques, ...), En recourant à des éléments des deux mondes. Compte tenu de la complexité de ces problèmes, les méthodes heuristiques, offrant de bonnes solutions dans un assez petit temps, ont été plutôt populaires dans le domaine, en mesure de résoudre des problèmes avec de nombreux éléments.

#### 1.4.1. Heuristique basé sur le positionnement :

Les heuristiques basées sur le positionnement sont les plus anciennes de la littérature SPP et les plus fréquentes dans les premiers travaux. Ils sont assez flexibles, ce qui permet d'intégrer les contraintes les plus courantes du problème. Le mécanisme de base derrière les heuristiques basées sur le positionnement est l'identification de l'espace libre sur la bande qui convient le mieux à une pièce donnée, selon un critère également donné. L'heuristique « Bottom-left » (BL) proposée par [baker et al. 1980] est la seule heuristique qui a proposé le modèle basé sur la stratégie first-fit qui est la plus célèbre dans l'approche de résolution pour le problème. Le but de BL est de placer chaque rectangle le plus bas et le plus à gauche possible, comme illustré à la (figure 1.9) La loi de « BL » proposée par Baker est appelée « Bottom Left Fill » (surnommé BLF), un rectangle préserve la position de stabilité ssi il est placé dans la position la plus basse (en premier) et la plus à gauche possible. Jakobs a utilisé l'heuristique BL avec une autre loi comme suit ; Une position initiale et réalisable est attribuée au rectangle qui arrive en haut à droite, puis elle est déplacée vers le bas et vers la gauche. Cette stratégie est appelée « Bottom left » tout court. Liu et Tung ont développé une autre stratégie similaire à celle de Jakobs sauf que le mouvement vers le bas est prioritaire quand la pièce glisse à gauche. (c)

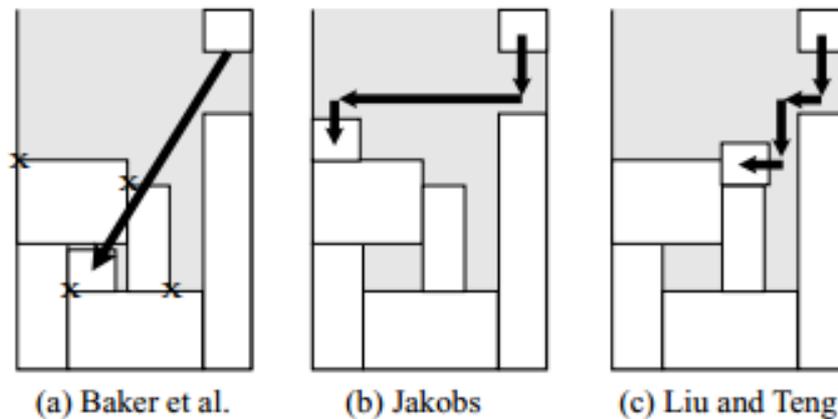


Figure 1.9. L'heuristique « Bottom Left » selon différentes stratégies

## 1.4.2. Problèmes de « Nesting » :

### 1.4.2.1. Définition :

Les problèmes de Nesting sont des problèmes ramenés de l'industrie plus précisément du domaine de fabrication de pièce aux niveaux des machines de découpe, le dilemme à résoudre est quand en a une quantité de matière première (surface de travail) vue en deux dimensions généralement, l'objectif d'un algorithme de Nesting est la création d'un nid de plusieurs pièces (en stock virtuel) d'une manière à minimiser les chutes à la fin de l'opération. Les problèmes de Nesting ont été introduits dans les années 80' mais depuis ils se sont développés avec la technologie de découpe en proposant des solutions logicielles de plus en plus précises et rapides. Ces logiciels de Nesting sont spécifiques pour chaque type d'industrie, pour chaque type de matière ou machine alors que maintenant on appelle plus une solution logicielle mais un système de Nesting intelligent intégré.

### 1.4.2.2. Dimensionnalité du problème :

Les problèmes de « Nesting » sont généralement 2-D (appelé : Blank nesting, nesting optimale, problème de découpe optimale). Dans les problèmes de bin Packing 1-D, l'objectif est de minimiser le nombre de bins (bacs) qui contiennent un ensemble de poids d'article, sous réserve de la limitation de poids pour chaque bin.

L'objectif des problèmes de Bin-Packing 2-D est de minimiser l'espace d'un seul bac orthogonale contenant plusieurs formes de dimension arbitraire-rectangulaires. Le chevauchement entre les articles n'est pas accepté, mais il n'y a aucune contrainte sur la diversité ou le

regroupement des articles. Dans le bin-packing 3-D l'objectif est de minimiser le volume d'un seul bac orthogonale avec des articles de forme en 3-D de dimension arbitraire. Dans les problèmes de NESTING 2-D l'objectif est de Maximiser le nombre d'éléments de forme arbitraire intégrés dans un seul bac orthogonal. C'est souvent autorisé de faire tourner l'élément durant le processus.[5]

La structure d'un Système de Nesting montre bien que la communication avec le système de conception et le système de fabrication est le point clé de l'intelligence d'un système de nesting avec l'utilisation des algorithmes géométrique et de placement juste à temps.

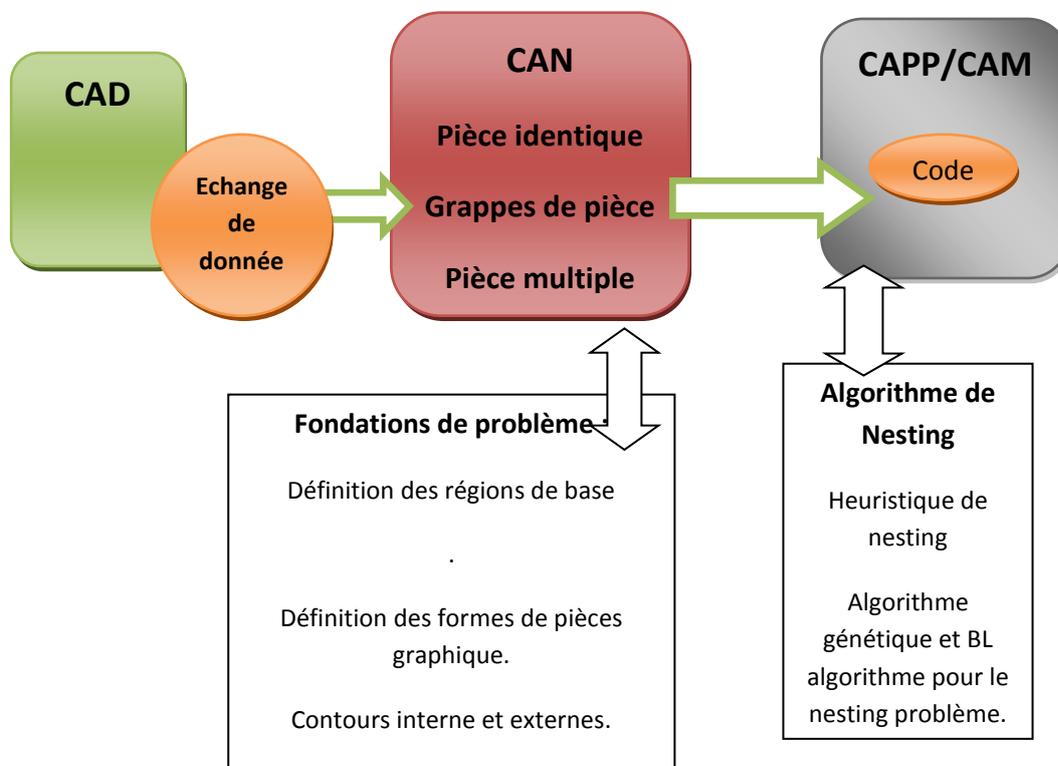


Figure 1.10. Schéma explicatif pour le processus de Nesting

**Mot clé :** *CAD* : CAO, *CAM*: FAO, *CAN* : imbrication assisté par ordinateur

### 1.5. Conclusion:

Dans ce chapitre, nous avons essayé de donner une image globale sur le monde de Packing et de Nesting , une introduction, un bref historique et une description de certains problèmes et de leur stratégie de solution.

# CHAPITRE 2

## Applications industrielles

### 2.1. Introduction :

La réduction des coûts de production est l'un des principaux problèmes dans l'industrie dans un monde toujours plus compétitif où les produits sont disponibles presque instantanément partout dans le monde. Une utilisation élevée des matériaux est particulièrement intéressante pour les industries ayant une production de masse, car de petits changements de disposition des pièces peuvent entraîner une réduction considérable des coûts de production et grandes économies de matériel. Les problèmes de coupe et d'emballage sont des problèmes d'optimisation qui concernent la recherche d'un bon arrangement de plusieurs éléments dans des régions contenant plus grandes. L'objectif habituel du processus d'attribution est de maximiser l'utilisation et, par conséquent, de minimiser le matériel «gaspillé». La complexité du problème et l'approche de la solution, dépendent de la géométrie des éléments à placer et des contraintes imposées comme le matériel utilisé et le mode de découpe.

### 2.2. Classification des matériaux :

#### 2.2.1. L'industrie des tôles :

L'industrie des tôles doit faire face à des problèmes de nesting réguliers et irréguliers. Certaines contraintes concernant les propriétés du matériau, le processus de coupe et les aspects de planification distinguent la tâche de nesting dans ce domaine par d'autres industries. En plus de réduire le gaspillage au minimum, il existe un certain nombre d'autres facteurs qui décident de la disposition finale des pièces.

Les matières premières sont disponibles sous forme de matériau en feuille avec des dimensions fixes ou en matériau enroulé d'une largeur fixe. Cela a un impact sur l'objectif de la tâche de nesting. Dans le premier cas, la forme d'un éventuel reste décide s'il peut être utilisé dans une future tâche de nesting ou doit être traité comme un gaspillage. Pour le matériau enroulé, ce n'est généralement pas une contrainte. Comme les matériaux ont des propriétés inhomogènes telles que l'orientation du grain, le nombre d'orientations possibles dans lesquelles les pièces peuvent

être imbriquées peut être limité. Cela dépend entièrement de l'application et du traitement ultérieur des formes. S'il n'y a pas d'opérations de pliage à suivre, les pièces peuvent être tournées dans Toute direction; Sinon un angle spécifié par rapport à l'orientation du grain doit être appliqué.[2.1]

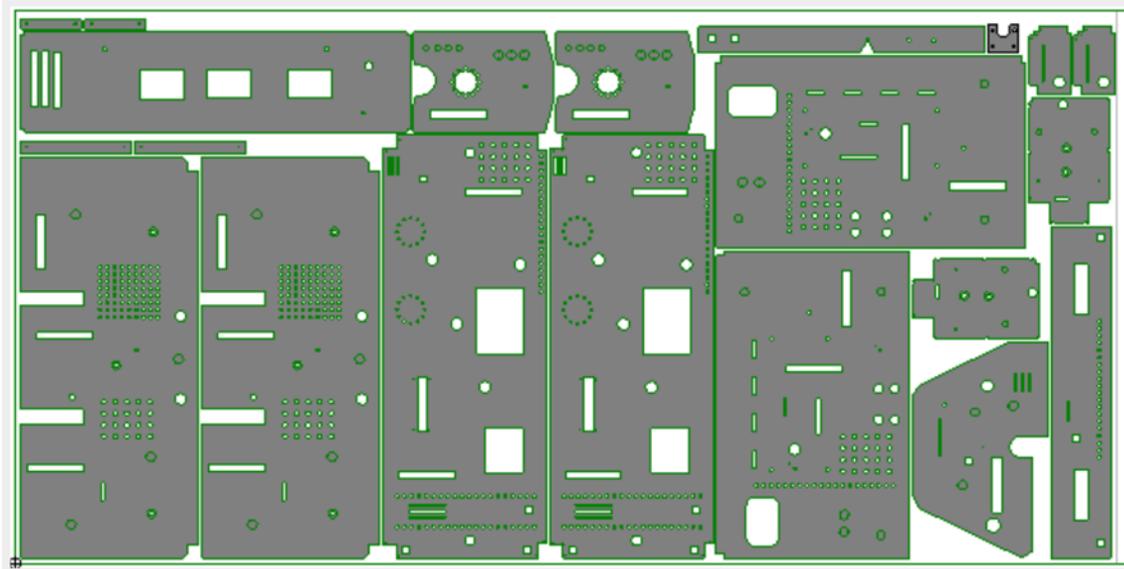


Figure.2.1.Exemple du layout dans L'industrie des tôles

Le logiciel Nesting offre une opportunité d'épargne. "Dans un magasin de fabrication de métaux, la matière première varie de 30 à 50 pour cent des coûts opérationnels. Par conséquent, la réduction des déchets, directement lié aux matières premières, permettra aux utilisateurs de réaliser d'énormes économies", Ben Terre Blanche président de Sigmatek Corp. «L'amélioration des nids de seulement 5 pour cent et la réduction de la quantité de matières premières nécessaires pour compléter un emploi peuvent avoir un impact majeur sur une organisation (figure 2).[2.2]

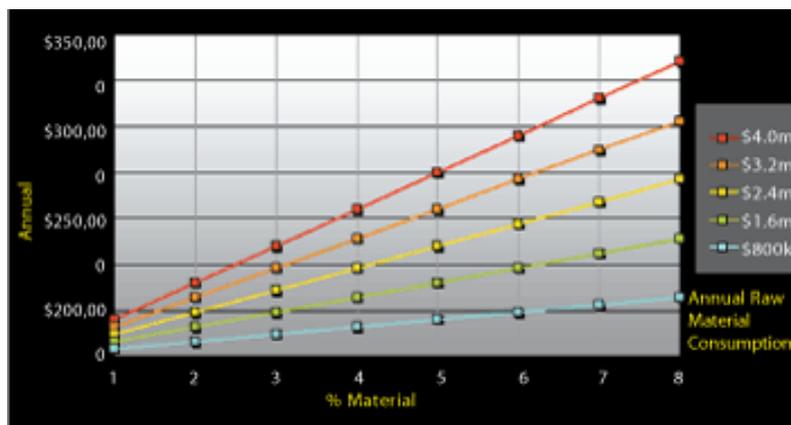


Figure .2.2.Consommation annuelle de matières premières



Le processus de nesting doit donc correspondre aux pièces avec leurs zones de qualité respectives sur la peau. Habituellement, le traitement de l'image a lieu avant le processus de nesting afin de déterminer la forme et la qualité des peaux. Comme cette tâche de nesting est très complexe et doit tenir compte de nombreuses exigences spécifiques, des paquets de nesting spécialement conçus sont disponibles pour l'industrie du cuir. [2.1]



Figure.2.5. Solution numérique dans l'industrie du cuir

#### 2.2.4. Industries avec des problèmes de coupe rectangulaires :

Le groupe de problèmes de nesting rectangulaire apparaît dans l'industrie du papier, du bois et du verre. La technique de coupe, qui implique des coupures de cisaillement, impose une contrainte caractéristique sur les mises en page dans cette zone. Les modèles de nesting doivent être guillotinables (Figure 6), de sorte que les pièces peuvent être obtenues par coupe droite uniquement dans la disposition restante. Le nesting dans l'industrie du verre et du bois doit également tenir compte de diverses gammes de qualité et des défauts de la matière première. Des tâches de nesting rectangulaires peuvent se produire Problèmes de nesting de bande et de bac. [2.1]

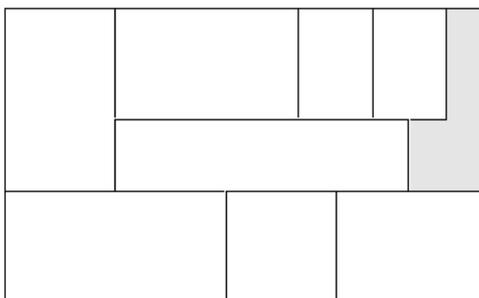


Figure .2.6. Mode guillotine

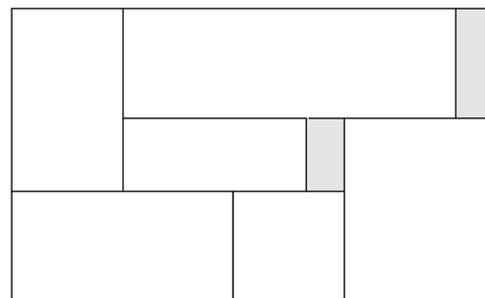


Figure .2.7. Mode non- guillotine

## 2.3. Mode de découpe :

### 2.3.1. Guillotine :

Une contrainte supplémentaire considérée dans ce chapitre est la coupe de la guillotine tous les éléments doivent être extraits par des coupures allant d'un bord au bord opposé. La figure (2.8) montre un motif de guillotine où tous les éléments peuvent être extraits par des coupures de guillotine qui divisent le récipient en deux parties Sans couper des objets.



Figure .2.8. Exemple du layout guillotinable

### 2.3.2. Non-Guillotine :

Le terme «non guillotine» se réfère au fait que les coupes ne se limitent pas aux coupures de guillotine imposées par certaines machines à couper (CNC) ou des environnements de nesting (une guillotine orthogonale coupée sur un rectangle) est une coupe d'un bord du rectangle au bord opposé, parallèle Au bord restant

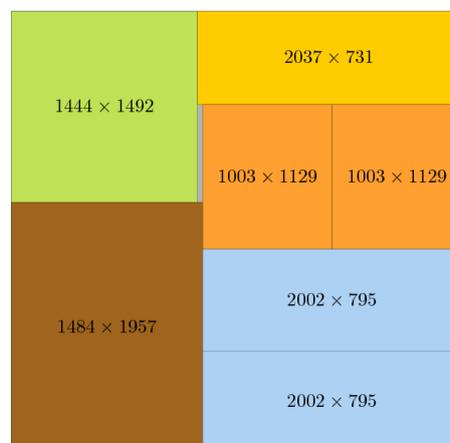


Figure .2.9. Exemple du layout non-guillotinable

### 2.3.2.1. La découpe laser :

La découpe laser est un procédé de fabrication qui consiste à découper la matière grâce à une grande quantité d'énergie générée par un laser et concentrée sur une très faible surface. Toute matière peut être découpée par un laser : métal, textile, papier, carton, céramique, composite, cuir, verre, etc. Des longueurs d'onde spécifiques ont été développées pour obtenir un rendu optimisé sur chaque matière. Certains matériaux possèdent toutefois des propriétés qui complexifient leur réaction au laser : par exemple l'argent et le cuivre en raison de leurs propriétés réfléchissantes. Il est toujours important de bien qualifier l'adéquation entre le laser et la matière.



Figure .2.10. Découpe laser dans l'industrie des tôles

### 2.3.2.2. La découpe Jet d'Eau :

Le découpage au jet d'eau est un procédé de fabrication qui utilise un jet d'eau hyperbare pour découper la matière (exemples : mousse, cuir, matériaux métalliques, matériaux composites, produits agroalimentaires, etc.). Les matériaux découpés sont souvent des plaques ou tôles. Celles-ci sont généralement posées sur un support composé de lames qui finissent par être découpées par le jet d'eau. Il est donc nécessaire de les remplacer régulièrement pour assurer un bon positionnement du matériau à découper. Un seul matériau ne peut être découpé au jet d'eau : le verre trempé, qui casse dès les premiers millimètres de découpe à cause des trop fortes contraintes présentes dans le matériau.



Figure .2.11. Découpe jet d'eau

### 2.3.2.3. Le poinçonnage :

La technologie de poinçonnage permet de réaliser sur une même machine des formes complexes dans la tôle, cuir et le textile. Le nesting pour le poinçonnage ne doit pas seulement tenir compte de la géométrie des pièces, mais aussi de la limite de l'outil. Certains logiciels de nesting peuvent respecter la limite de l'outil plutôt que la limite de la partie. Le logiciel respecte l'intégrité de la pièce perforée, du squelette et de l'outillage tout à la fois.

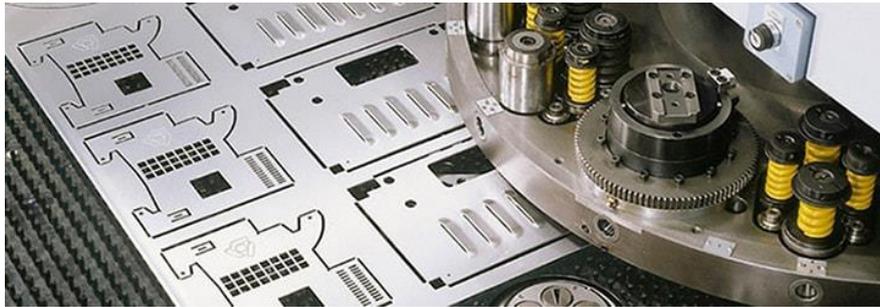


Figure .2.12.Poinçonneuse multi-têtes

## 2.4. Le role de Nesting :

Les logiciels de nesting sur place répondent aux besoins d'optimisation. Alors que certains ne servent qu'à le nesting rectangulaire, d'autres offrent un profil ou un nesting de forme où les pièces requises peuvent être n'importe quelle forme. Ces pièces irrégulières peuvent être créées à l'aide d'outils de conception assistée par ordinateur (CAO) populaires. La plupart des logiciels de nesting peuvent lire automatiquement les fichiers de profil IGES ou DXF, dont certains fonctionnent avec des convertisseurs intégrés. La (figure 14) résume les étapes généralement impliquées dans le processus de nesting industrielle.

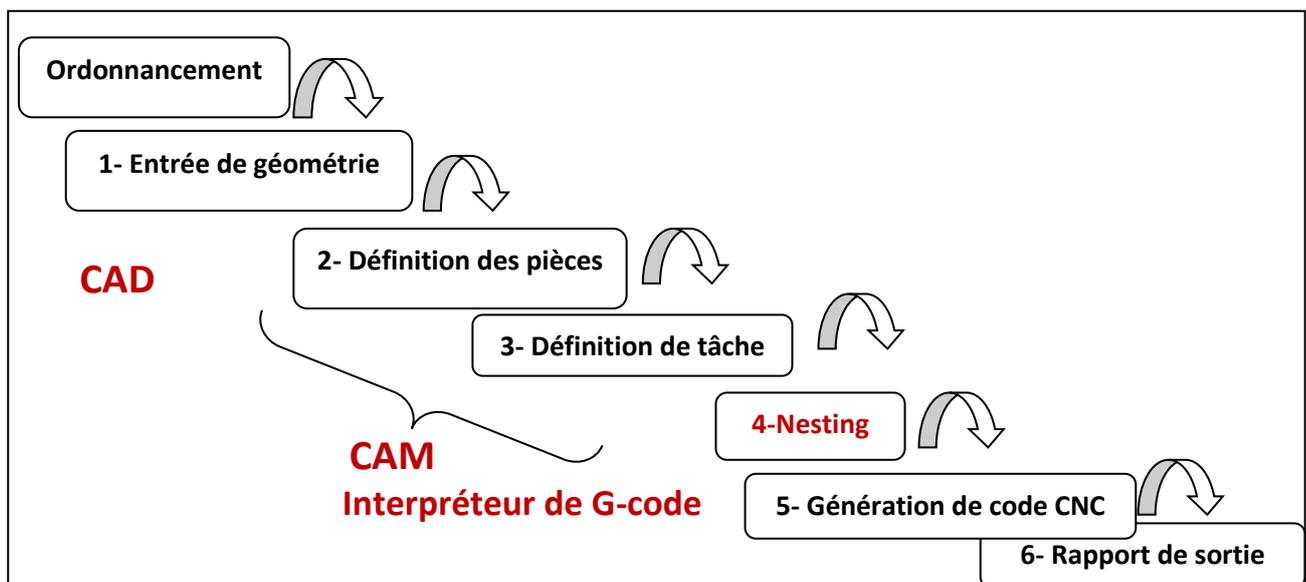


Figure .2.13. le Nesting dans le domaine de fabrication

- enregistrer la géométrie à partir d'autres systèmes de CAO via le format de fichier DXF, DWG etc. ou créer une géométrie à l'aide d'un module de CAO intégré
- définir les pièces: matériau, épaisseur, qualité de la pièce, etc.
- définir la tâche de nesting: nombre et taille de l'objet, dédoublement, processus de découpe, etc.
- mode de nesting : automatique, manuel ou interactif sur des objets simples ou multiples
- générer un ordre dans lequel les parties doivent être coupées; Code NC de sortie pour le processus de coupe
- générer un rapport comprenant des détails de mise en page, p.ex. Le pourcentage de déchets, la longueur du matériau en bande ou l'utilisation de la feuille; Réduction du temps et du coût

### **2.5. Logiciels de Nesting :**

Logiciels de nesting est utilisé pour adapter de manière optimale de nombreuses pièces de fabrication différentes à une seule feuille de matière première. Optimal signifie que vous obtenez les pièces que vous voulez en quantités exactes, au coût le plus bas possible; Le coût le plus bas comprend l'efficacité matérielle, l'efficacité de la machine, l'achèvement de l'ordre et toutes les autres considérations relatives aux coûts. En termes généraux, le logiciel de nesting organise automatiquement et efficacement les quantités requises de pièces individuelles à produire sur des feuilles ou des plaques de matière en stock. Il le fait en utilisant la géométrie partielle à partir de fichiers CAO pour produire un code CNC qui contrôle une machine à découper.

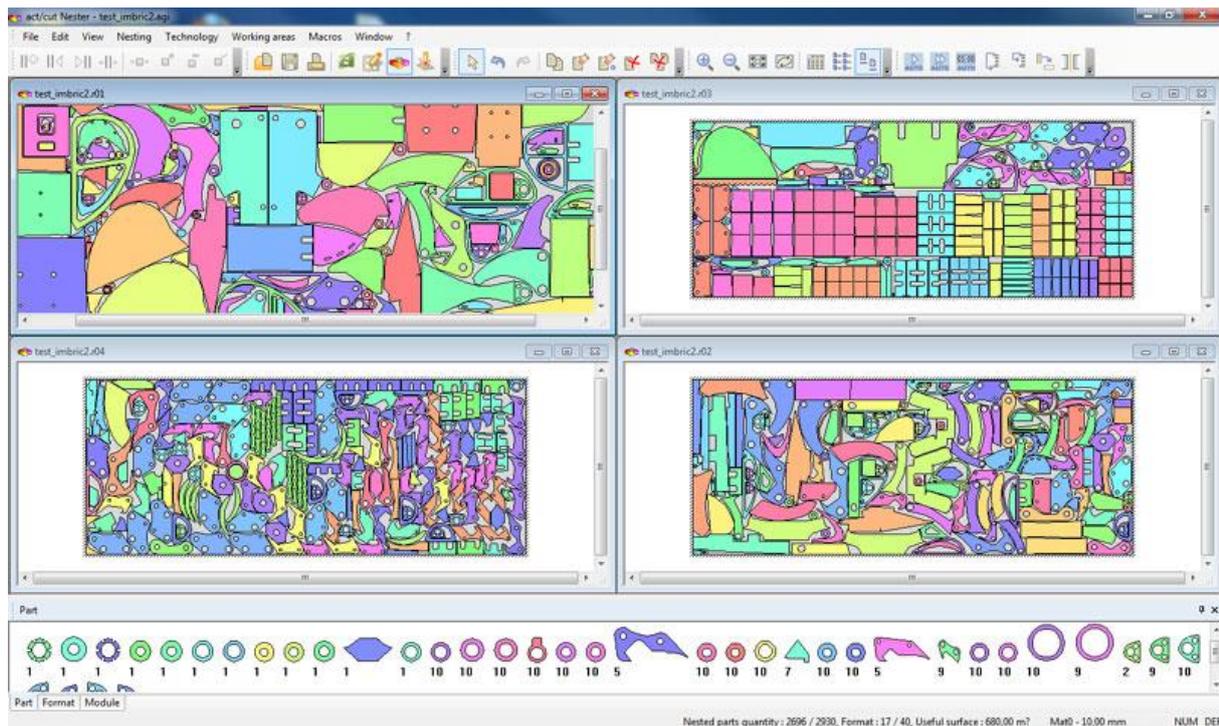


Figure .2.14.Interface d'un logiciel de nesting

## 2.6. Avantages et efficience :

Lorsque nous parlons d'efficacité, nous parlons d'économiser de l'argent et du temps, Nous prenons l'exemple d'ACME Company pour comprendre plus, ACME a une dépense matérielle mensuelle de 50 000 \$, Cependant 25% du coût est une perte de 12 500 \$ par mois en dépenses gaspillées. ACME a réussi à réduire le taux de chute de 10% ou 1 250 \$ avec le logiciel de nesting. Déduisant leur paiement de location de 500 \$ par mois, ils ont une réduction de coût net de 750 \$ par mois. [2.3]

- Voici les grands avantages de l'utilisation du logiciel de nesting:

- **Efficacité matérielle:** signifie économiser du matériel, L'approche exclusive du logiciel de nesting peut signifier une augmentation de 8 à 20% des économies de matériaux par rapport à l'approche actuelle.

- **Temps de programmation:** obtenez l'imbrication et la programmation de la machine en quelques minutes; Pas des heures. Créez de meilleurs nids qui ont également considérablement réduit la main-d'œuvre des ateliers! Laissez un logiciel intelligent vous faire économiser du temps, de l'argent, des efforts et des maux de tête.

- **Augmenter le débit:** fabrique plus de pièces avec moins d'erreurs en moins de temps. Vos machines de fabrication CNC (poinçon à tourelle, laser, plasma, etc.) peuvent être plus productives grâce à un logiciel de nesting avancé. On utilise généralement des machines à 90% des cycles de service et au-dessus; Doublant souvent le débit.
- **Contrôle de qualité:** les pièces endommagées, les pièces perdues, les pièces incorrectes et les quantités de pièces sont le mal de tête de chaque fabricant. Ils causent du chaos sur le sol et sont un drain de trésorerie dans le back office. Créez les bonnes pièces la première fois, à chaque fois.
- **Réduire les frais généraux:** économiser sur l'usure de la machine avec des nids plus efficaces. Réduisez considérablement le temps d'installation, le temps de manipulation et le temps de déchargement. Utilisez le logiciel pour piloter tous les équipements de fabrication CNC; Économisez encore plus.
- **Améliorer les flux de trésorerie: un logiciel ne coûte rien :** Chaque client obtient au moins 100% de rendement en première année. Lors de l'utilisation du logiciel, les fabricants réalisent généralement un rendement annuel supérieur à 100% sur leur investissement chaque année. Réduisez les dépenses mensuelles et créez des flux de trésorerie supplémentaires en économisant davantage sur les matières premières que vous dépensez sur un contrat de logiciel.

## 2.7. Conclusion :

Dans ce chapitre pour continuer notre concept, nous avons expliqué l'importance de nesting dans le monde industriel et les différentes contraintes puis nous avons parlé de logiciels et son efficacité, tout cela pour donner une idée de la nécessité de ce type de logiciel dans l'industrie.

## CHAPITRE 3

### Problème avec des pièces rectangulaires

#### 3.1. Introduction :

Il existe de nombreux problèmes d'emballage différents dans les industries. Comme l'importance de ces problèmes, divers algorithmes heuristiques basés sur différentes stratégies ont été présentés. Ces algorithmes peuvent être classés en deux catégories: les algorithmes heuristiques traditionnels et les algorithmes méta-heuristiques. Les algorithmes heuristiques traditionnels utilisent l'information heuristique pour guider le processus de recherche. Le chapitre présente d'abord le problème et son modèle mathématique avec une approche de solution qui est basée sur l'algorithme heuristique (a Least Wasted First – LWF) qui évalue les positions utilisées par les rectangles et une recherche locale aléatoire est introduite pour améliorer les résultats, on a aussi posé un processus de packing détaillé pour expliquer les étapes principales de l'heuristique (LWF) finalement on a terminé avec une vision de projet et conclusion.

#### 3.2. Revue de Littérature :

Tellement il y a un énorme nombre des articles qui ont étudié ce type de problème on a essayé de commencer avec les algorithmes heuristiques puis les métaheuristiques. Les algorithmes heuristiques traditionnels utilisent l'information heuristique pour guider le processus de recherche. Les méthodes de remplissage en bas à gauche (BL) et BL [3.1,3.2] sont les approches heuristiques les plus célèbres. Liu et al. A présenté un algorithme heuristique amélioré basé sur BL [3.3], une méthode bestfit (BF) a été suggérée par Burke et al. [3.4]. Le principe de moins de flexibilité a été introduit par Wu et al. Pour déterminer la règle d'emballage [3.5]. Zhang et al, a proposé un algorithme récursif heuristique [3.6], qui organise les rectangles à l'aide d'une structure récursive. Huang et al. A présenté un algorithme heuristique très efficace [3.7]. Cui et al. A présenté un nouvel algorithme récursif heuristique [3.8]. Les algorithmes méta-heuristiques utilisent les stratégies méta-heuristiques telles que le recuit simulé, l'algorithme génétique et les réseaux neuronaux artificiels pour améliorer les résultats de recherche. Hopper et al. A donné une enquête empirique sur les algorithmes méta-heuristiques et heuristiques pour le problème d'emballage 2D [3.9]. Zhang et al. A présenté un algorithme méta-heuristique basé sur la stratégie récursive et l'algorithme de recuit simulé [3.10]. Les réseaux de neurones artificiels ont été introduits par Dagli

et al. Pour résoudre le problème d'emballage [3.11]. Berthold [3.12] a présenté une approche génétique pour le problème de l'emballage de bacs guillotinables, un algorithme génétique sans codage des solutions a été présenté par Bortfeldt [3.13]. Beasley [3.14] a présenté une heuristique de population pour le problème contraint de coupe à deux dimensions non-guillotine. Les approches heuristiques pour les problèmes d'emballage des sacs à dos et à trois dimensions ont été proposées par Egeblad et al. [3.15]. Notre cas concerne le problème d'emballage rectangulaire 2D qui s'appelle également un problème de sac à dos rectangulaire dans le but de maximiser la surface couverte par les rectangles emballés ou le taux de remplissage. Notre travail est basé sur la littérature [3.16] qui présente d'abord une première stratégie (a Least Wasted First) qui évalue les positions utilisées par les rectangles, puis introduit une recherche locale aléatoire pour améliorer les résultats, développe finalement un premier algorithme heuristique (LWF) pour le problème d'emballage rectangulaire 2D. Les résultats de calcul montrent que LWF peut obtenir les solutions optimales en peu de temps pour ces instances.

**Mot clé :** BL : bottom left , LWF : Least Wasted First

### 3.3. Définition du problème :

Étant donné une feuille rectangulaire  $C$  avec largeur  $W$  et hauteur  $H$ , un ensemble de  $n$  rectangles  $R$  avec chaque rectangle  $R_i$  de largeur  $w_i$  et hauteur  $h_i$  ( $1 \leq i \leq n$ ), placez le BL (Bottom Left = Coin inférieur gauche) de la feuille à l'origine  $(0,0)$  de système de coordonnées cartésiennes et laissez ses quatre côtés parallèlement aux axes  $X$  et  $Y$ , respectivement (figure 1). Le but du problème nesting rectangulaire 2D est de trouver un emplacement qui maximise la surface totale des rectangles placés dans la feuille ou le taux de remplissage. L'emballage doit satisfaire aux contraintes suivantes:

- (1) chaque rectangle emballé doit être complètement emballé dans la feuille.
- (2) chaque rectangle peut être emballé horizontalement ou verticalement dans la feuille, ce qui signifie que les rectangles sont rotatifs.
- (3) chaque bord des rectangles emballés doit être parallèle à un bord de la feuille, qui est également appelé emballage orthogonal.
- (4) les deux rectangles emballés ne doivent pas se chevaucher.
- (5) L'emballage non guillotine est autorisé.

Soit  $f_i$  ( $1 \leq i \leq n$ ) indiquer si le rectangle  $R_i$  a été emballé dans la feuille ou non, si le rectangle  $R_i$  a été emballé dans la feuille, alors  $f_i = 1$ , sinon  $f_i = 0$ . Pour chaque rectangle  $R_i$  emballé dans la feuille, laissez  $(x_{li}, y_{li})$  indiquer les coordonnées de son coin inférieur gauche et  $(x_{ri}, y_{ri})$  indiquent les coordonnées de son coin supérieur droit figure (3.1)

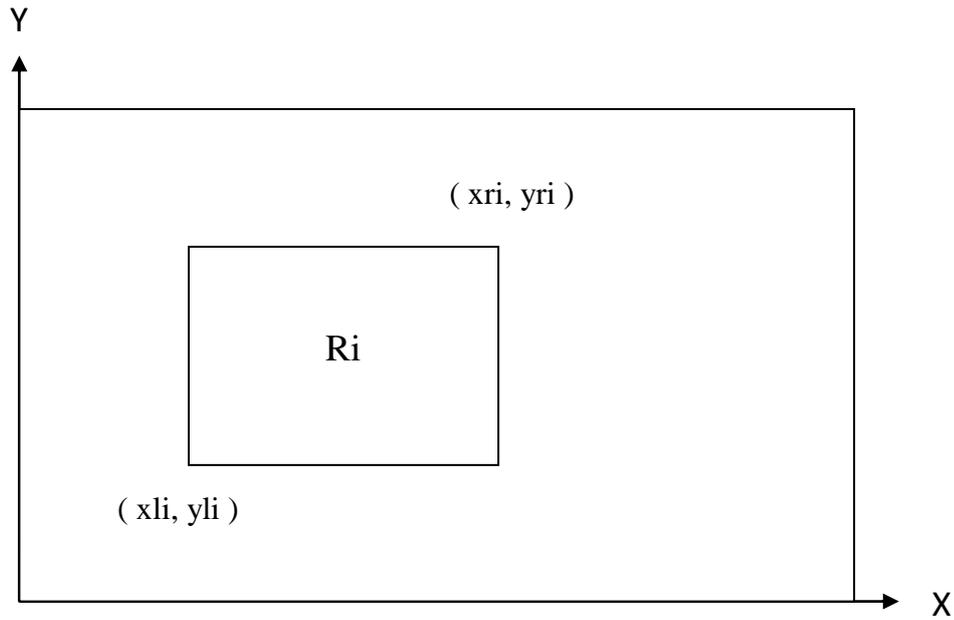


Figure .3.1. Le rectangle Ri dans le system cartésien

La formulation mathématique du problème peut être décrite comme suit:

$$\text{Max } \sum_{i=1}^n f_i w_i h_i$$

Contraintes :

$$(1) f_i = 0 \cup (0 \leq x_{li} < x_{ri} \leq w \cap 0 \leq y_{li} < y_{ri} \leq H).$$

$$i = 1, 2, \dots, n$$

$$(2) f_i = 0 \cup (x_{ri} - x_{li} = w_i \cap y_{ri} - y_{li} = h_i) \cup (x_{ri} - x_{li} = h_i \cap y_{ri} - y_{li} = w_i).$$

$$i = 1, 2, \dots, n$$

$$(3) f_i = 0 \cup f_j = 0 \cup (x_{li} \geq x_{rj} \cup x_{lj} \geq x_{ri} \cup y_{li} \geq y_{rj} \cup y_{lj} \geq y_{ri}).$$

$$i, j = 1, 2, \dots, n, i \neq j.$$

$$(4) f_i \in \{0, 1\}, i = 1, 2, \dots, n.$$

(1) Implique tous les rectangles emballés sont complètement dans la feuille.

(2) implique que les rectangles emballés sont rotatifs.

(3) implique que Les rectangles emballés ne peuvent pas se chevaucher.

(4) implique les rectangles peuvent être choisis pour emballer ou non.

### 3.4. Résolution du problème :

Même si c'est le 21ème siècle et avec toutes les avancées dans la théorie des mathématiques, il n'y a pas de solution mathématique complète pour ce problème.

#### 3.4.1. Algorithm heuristique Least Wasted First :

##### 3.4.1.1 La stratégie Least Wasted First :

La façon de trouver des positions pour placer les rectangles, qui est présenté par Martello et al. [3.16]. Le premier rectangle est emballé avec son angle **BL** à l'origine. Soit (**I**) l'ensemble des rectangles emballés dans la feuille, les rectangles restants ne peuvent être emballés que dans la position **P** sachent que tout rectangle dans **I** soit inférieur à **P** ou à gauche de **P** et ne puisse pas dépasser la bordure de la feuille, plus Formellement, les rectangles à emballer peuvent être emballés uniquement aux positions contenues dans l'ensemble:

$$S(I) = \{(x, y) : (\forall Ri \in I, x \geq xri \vee y \geq yri) \wedge (x \leq W \wedge y \leq H)\}.$$

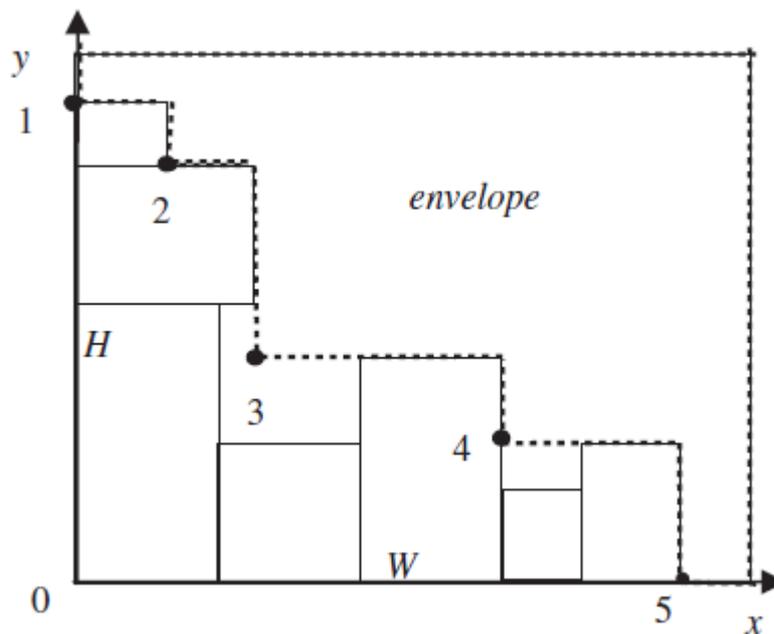


Figure .3.2.position faisable

Figure.3.2.montre les régions faisable (la zone entourée par la ligne brisée), les rectangles restants peuvent être emballés dans la région appelée **enveloppe** (le concept d'enveloppe est différent de celui de [3.16]). Il faut noter que nous considérons seulement les points où la pente de l'enveloppe passe de la verticale à l'horizontale (points noirs de la figure 3.2) [3.16], ces positions sont appelées des positions faisables. Pour vérifier si un rectangle peut être emballé à une position possible, il suffit de vérifier si le rectangle dépasse le bord de la feuille ou non. Il existe une

optimisation qui peut être utilisée pour réduire le nombre de positions faisables. Si l'écart entre une position faisable et la bordure de la feuille est inférieur au plus petit bord des rectangles non emballés, cette position est appelée *la mauvaise position* (voir la figure 3.3) car aucun des rectangles non emballés peut être emballé à cette position. Nous pouvons éliminer les mauvaises positions et apporter des modifications à l'enveloppe. Par exemple, dans la figure.3.3, si  $l_1$  et  $l_2$  sont plus petits que le plus petit bord des rectangles décompressés, nous éliminons les mauvaises positions 1 et 4, l'enveloppe modifiée est illustrée à la figure.3.4 (la région noire est rejetée). En outre, nous devons considérer quel rectangle et quelle position doit être sélectionnée lorsque le nombre de ces rectangles et positions est supérieur à un. Inspiré par l'expérience dans la vie quotidienne, nous devrions rendre l'enveloppe aussi douce que possible. Plus les changements de la pente de l'enveloppe de la verticale à l'horizontale sont, plus l'enveloppe est douce. Nous devrions donc développer une façon de mesurer le bon fonctionnement d'un placement. Nous pouvons voir que chaque position possible est formée par une ligne horizontale et une ligne verticale. Lorsqu'un rectangle est emballé dans une position, si sa largeur est égale à la ligne horizontale ou sa hauteur est égale à la ligne verticale, nous appelons ce pack *un bon paquet* (figure 3.5). Nous utilisons un nombre variable de valeur (GN = goodness) pour évaluer ce pack. Si la largeur est égale à la ligne horizontale et que la hauteur est égale à la ligne verticale, GN de ce paquet sera 2; Si seulement l'un d'eux est égal, GN sera 1; Sinon GN sera 0. Nous allons sélectionner le placement avec une plus grande valeur de GN lorsque la zone gaspillée est identique. Par conséquent, nous pouvons décrire la manière de sélectionner le rectangle et de positionner comme suit:

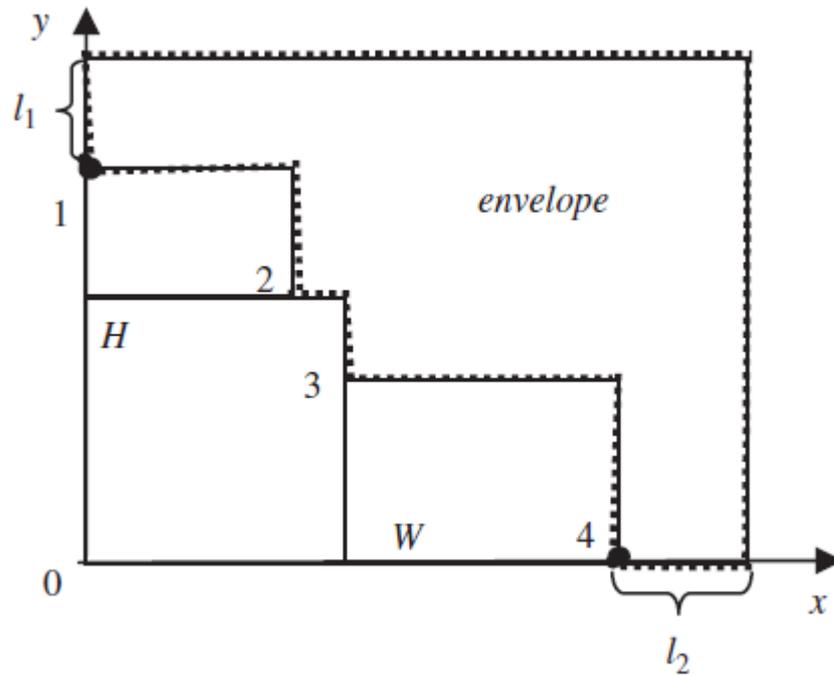


Figure .3.3. Mauvaises position.

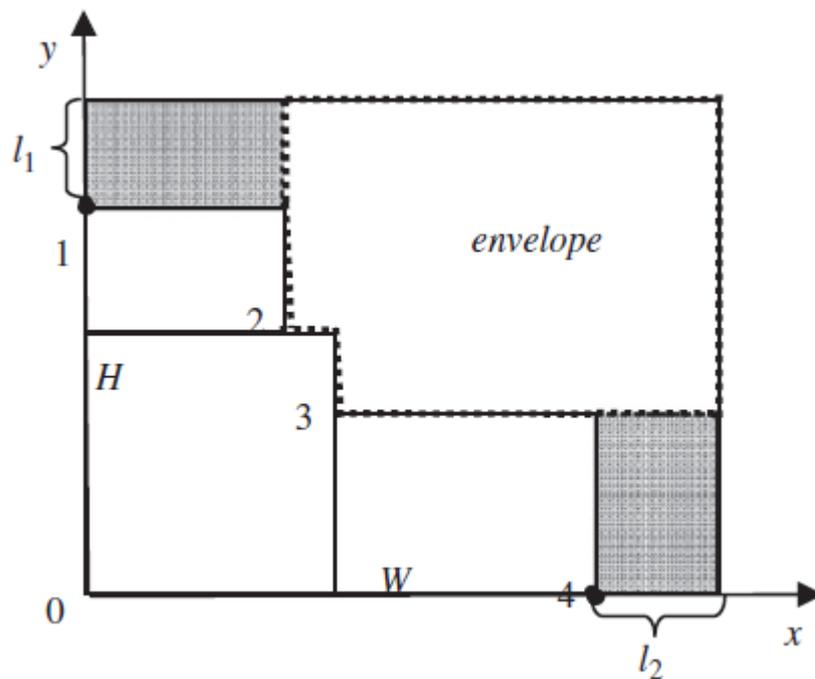


Figure .3.4. Changer l'enveloppe.

- (A) sélectionnez le rectangle et positionnez-le de telle sorte que la zone gaspillée soit minimale si le rectangle sélectionné est emballé à la position sélectionnée;
- (B) si le nombre de ces emplacements est supérieur à un, sélectionnez celui avec la valeur

maximale de GN;

(C) si le nombre de ces postes est supérieur à un, sélectionnez la première position (considérez les positions de gauche à droite); et

(D) si le nombre de ces rectangles est supérieur à un, sélectionnez le premier rectangle (considérons les rectangles dans l'ordre donné). En utilisant la stratégie d'emballage des rectangles décrits ci-dessus, nous obtenons

L'algorithme de base suivant:

2d-packing (R)

$I \leftarrow \emptyset$ ;

**while** R n'est pas vide **do**

Calculer l'ensemble P de tous les position faisables ;

Utilisez la façon décrite ci-dessus pour sélectionner le rectangle de R Et la position P;

**if** aucun des rectangles de R ne peut être emballé à l'un des

Position en P **then break** ;

**else**

Laissez le rectangle sélectionné r, paquet r dans la sélection

Position,  $I \leftarrow I \cup \{r\}$ ,

$R \leftarrow R \setminus \{r\}$ ;

**end if**

**end while**

**return** la surface totale des rectangles en I

**end .**

Où R est l'ensemble ordonné de rectangles, I est l'ensemble des rectangles Emballé jusqu'à présent, P désigne l'ensemble de tous les positions faisables. Dans la procédure 2d-packing (R), nous avons d'abord mis I pour être vide, puis exécuter le processus d'emballage jusqu'à ce que aucun des rectangles déballés peuvent être emballés dans la feuille. La surface totale des rectangles emballés est retournée à la fin.

### 3.4.1. 2. Recherche locale aléatoire :

Étant donné que le résultat de la procédure 2d-packing (R) dépend de l'Ordre des rectangles dans R, nous importons une recherche locale aléatoire pour Améliorer la qualité de la solution.

L'algorithme développé est présenté comme suit:

Rls-2d-packing (R, callmax, samemax)

Trier les rectangles dans R par leur zone;

```
for i = 1 to |R| do
  if  $w_i < h_i$  then
    Échangez ( $w_i$ ,  $h_i$ );
  end if
end for
```

Meilleur  $\leftarrow$  2d-packing (R);

Même  $\leftarrow$  0;

**for** i = 1 **to** callmax **do**

Sélectionnez le rectangle a et b de R de manière aléatoire, échangez le Ordre de a, b dans R ;

Zone  $\leftarrow$  2d-packing (R);

**if** zone > mieux **then**

Même  $\leftarrow$  0, la meilleure  $\leftarrow$  zone;

**if** mieux = W \* H ou best = totarea **then break**;

**end if**

**else**

Échangez la position de a, b dans R, même  $\leftarrow$  même + 1;

**end if**

**if** même > samemax **then break**;

**end if**

**end for**

**return** le mieux

**end .**

Où *callmax*, *samemax* sont les paramètres utilisés pour contrôler le nombre d'appels de la procédure 2d-packing (R), le meilleur est utilisé pour enregistrer le meilleur résultat trouvé jusqu'ici, *totarea* est la superficie totale des rectangles dans *R.sameis* pour comptait le nombre d'appels dans lesquels le résultat n'est pas amélioré. Dans le rls-2d-packing (*R*, *callmax*, *samemax*) Procédure, nous trions d'abord les rectangles par leur zone car les rectangles avec une zone maximale devraient avoir la priorité, puis nous échangeons la hauteur et la largeur des rectangles de sorte que la largeur soit pas plus court que la hauteur. Afin d'améliorer le résultat,

nous sélectionnons deux rectangles au hasard (la taille des rectangles sélectionnés ne devrait pas être égal), et échanger leur commande en R, appeler 2d-packing (R) pour vérifier si le résultat est amélioré ou non, si oui, nous allons accepter le changement, sinon l'annuler. Si le résultat ne peut pas être amélioré dans les appels *samemax* ou le nombre d'appels dépasse *Callmax* ou une solution optimale est trouvée, nous arrêterons le processus.

### 3.4.1. 3. algorithme heuristique Least Wasted First :

La procédure rls-2d-packing (*R*, *callmax*, *samemax*) est souvent Piégée dans les minimums locaux. Inspiré par l'idée de l'algorithme de Las Vegas [3.17], l'algorithme heuristique LWF est comme suit:

Lwf-rls-2d-packing (*R*, *lmax*, *callmax*, *samemax*)

Meilleur  $\leftarrow$  0;

**for** *i* = 1 **to** *lmax* **do**

Zone  $\leftarrow$  rls-2d-packing (*R*, *callmax*, *samemax*);

**if** zone > mieux **then**

Meilleure  $\leftarrow$  zone;

**if** mieux =  $W * H$  ou best = totarea **then break**;

**end if**

**end if**

**end for**

**return** le mieux

**end**

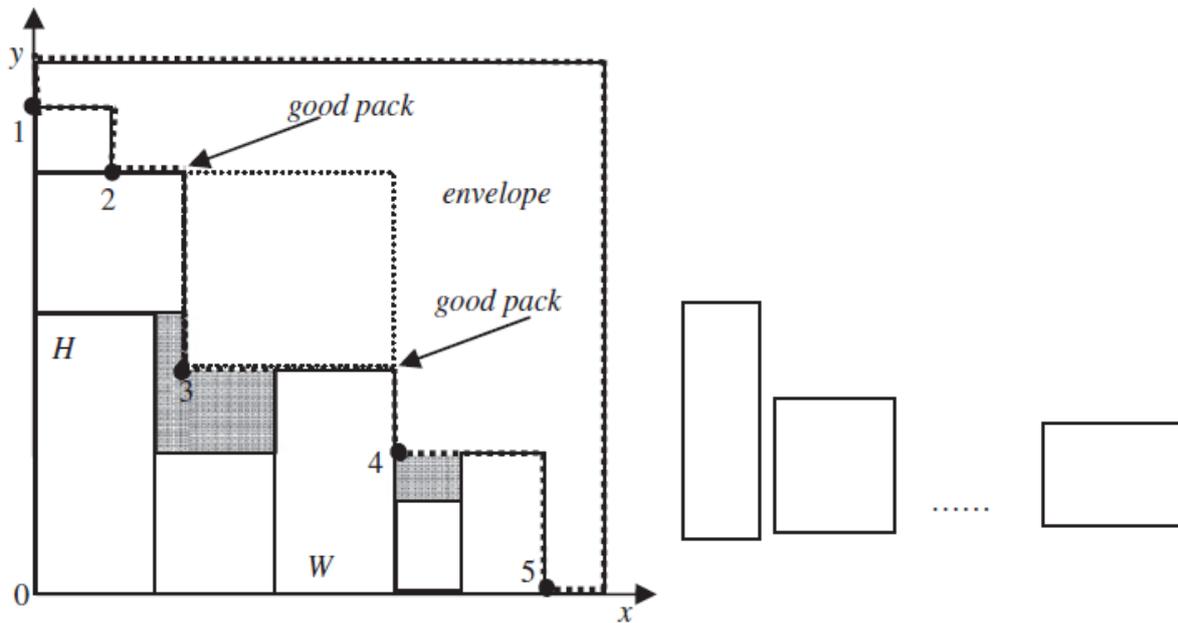


Figure .3.5. good pack

Où  $l_{max}$  est le nombre d'appels de la procédure `rls-2d-packing` ( $R, callmax, samemax$ ), **le meilleur** est utilisé pour enregistrer le meilleur résultat trouvé jusqu'à présent. **Totarea** est la surface totale des rectangles dans  $R$ . Dans la procédure `lwf-rls-2d-packing` ( $R, l_{max}, callmax, samemax$ ), nous appelons à plusieurs reprises la procédure `rls-2d-packing` ( $R, callmax, samemax$ ) et enregistrons le meilleur résultat trouvé jusqu'à ce que le nombre d'appels dépasse  $l_{max}$  ou une solution optimale soit trouvée.

### 3.4.2. Processus de LWF :

Dans ce processus on expliquera les étapes de fonctionnement de cette heuristique avec une petite amélioration

#### 3.4.2.1 Conditions :

Pour commencer le packing il faut, Une table  $T(H,W)$  et une séquence finale des rectangles  $R_n(l,w)$

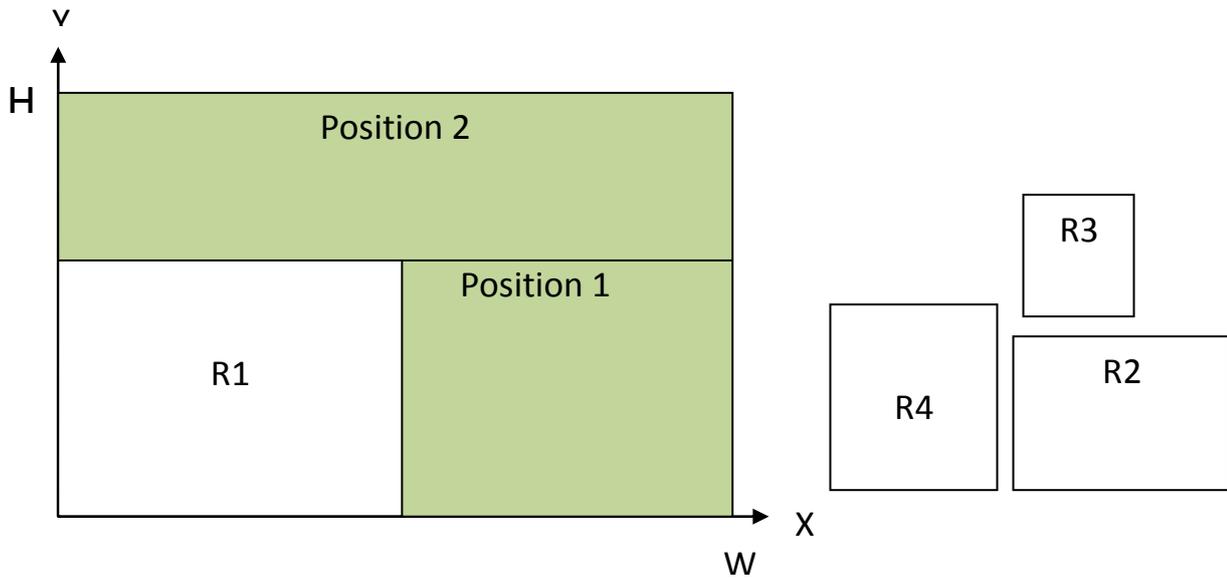


Figure .3.6 Table de travail avec deux positions

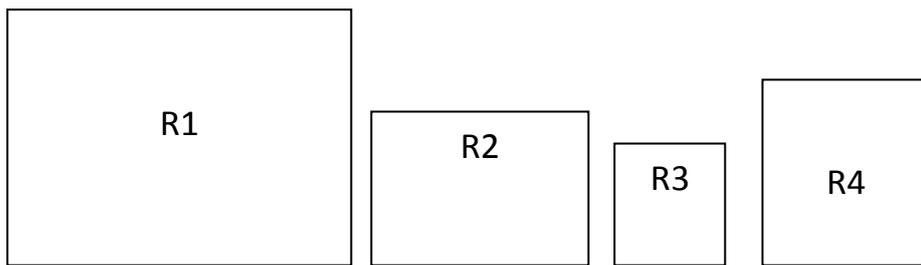


Figure .3.7 Liste de rectangles

La table et les rectangles sont caractérisés par sa longueur et largeur. Ces deux caractéristiques avec ses valeurs sont appelées dimensions.

$T(H, W), \{H > 0 \text{ et } W > 0\}$

$R(l, w), \{ \text{et } l > 0 \text{ et } l \leq H \}, \{ w > 0 \text{ et } w \leq W \}$

La longueur du rectangle ne peut pas être supérieure à la longueur du table et la largeur de l'élément ne peut pas être supérieure à la largeur de la table.

En raison de la précision, un processus d'emballage complet doit être observé dans le système de coordonnées orthogonales XY.

Lorsqu'il est placé à l'intérieur du système de coordonnées, table et rectangle obtient une troisième caractéristique orientation - position de rectangle, relatif à X ou Y. Compte tenu des nombreuses positions que les rectangles peuvent prendre, il existe deux distinctions pour lesquelles orientation obtient une valeur définie: *horizontale* et *verticale*.

- L'orientation horizontale est un cas lorsque le côté plus long de rectangle est parallèle à l'axe X.
- L'orientation verticale est le cas lorsque le bord plus long de rectangle est parallèle à l'axe Y.
- L'orientation et la disposition des rectangles avant l'emballage ne sont pas pertinentes
- Les rectangles peuvent être tournés et emballés horizontalement ou verticalement.
- Les rectangles d'emballage à la table peuvent être aléatoires ou en séquence le long de l'axe X ou Y à la surface libre de la table, appelée *position* pour l'emballage.
- La *position* c'est une surface libre de la table, indique une partie de la table qui n'est pas déjà occupée avec un autre rectangle emballé et où peut être placé l'un des rectangles restants.
- Il ne peut y avoir de chevauchement des rectangles à la surface de la table.
- La forme du *position* ne doit pas nécessairement être rectangulaire en tant que rectangle.
- La sélection de rectangle suivant qui va être emballé, dépend de la possibilité d'un rectangle à installer dans la position observée.
- Lorsqu'il y a plus d'un rectangle pouvant être placé à la position sélectionnée, le choix est aléatoire.
- Les rectangles d'emballage doivent être compacts côte à côte, assortir les dimensions des rectangles et des dimensions des positions afin que l'utilisation de la surface de la table soit maximale.
- Le processus d'emballage est terminé lorsque:
  - Il n'y a plus de rectangle à emballer, tous les rectangles sont emballés sur la table.
  - La table est complètement remplie, la somme des zones des rectangles emballés est égale à la surface de la table.
  - Il n'y a plus de positions sur la table pour les rectangles d'emballage, la surface libre restante de la table n'est pas possible d'adapter les rectangles restants.

L'algorithme heuristique LWF qui devrait être développé sur les observations ci-dessus est complexe, donc, pour une étude plus approfondie de ce problème, il est nécessaire d'introduire certaines restrictions.

- Avant le démarrage du processus d'emballage, le tableau doit être placé à l'origine du système de coordonnées avec orientation horizontale.
- Avant le démarrage du processus d'emballage, tous les rectangles doivent être orientés horizontalement et triés par ordre décroissant selon la taille de sa surface puis par la longueur puis par la largeur.
- Les rectangles ne peuvent pas tourner pendant le processus d'emballage.
- Les rectangles doivent être emballés en ordre suivant l'axe X de la table, avec une orientation horizontale exclusivement.
- Les coordonnées de la première position sont toujours des coordonnées de l'origine, car celle au début de la table des processus d'emballage est vide et toute sa surface est libre.
- Conditions pour trouver de nouvelles positions libres selon la demande d'un moyen d'emballage.
  - ❖ La position doit être de forme rectangulaire, avec des coordonnées déterminées du coin inférieur gauche et une dimension qui correspond au moins à l'un des rectangles restants. Au moins l'un des rectangles restants devrait être en position.
  - ❖ Les coordonnées de la nouvelle position ne peuvent être identiques que les coordonnées du coin inférieur ou supérieur gauche de l'élément préalablement emballé.
  - ❖ Si la position a une forme différente de rectangulaire, il faut l'ajuster. Cela se fait en ouvrant de nouvelles positions, qui doivent remplir auparavant deux conditions. Si les positions nouvellement créés ne remplissent pas les conditions, ils sont éliminés.
  - ❖ L'ordre des positions est déterminé en triant les valeurs des coordonnées de l'amant du coin gauche de la position, dans l'ordre croissant, d'abord par Y que par l'axe X. De cette façon, les éléments sont emballés dans la manière spécifiée.
  - ❖ Lorsqu'il n'y a plus de positions libres, le processus d'emballage se termine.
- Conditions pour sélectionner le rectangle suivant pour l'emballage.
  - ❖ La sélection de rectangle suivant qui va être emballé dépend de la prochaine caractéristique de rectangle emballé – *facteur de montage* et sa valeur numérique calculée pour la position sélectionnée, ce qui définit la possibilité d'installer un rectangle à l'intérieur de la position observée. La plus grande valeur de cette caractéristique signifie que ce rectangle correspond mieux à la position observée.
  - ❖ Le rectangle suivant qui va être emballé est sélectionné sur la valeur calculée pour *facteur de montage* pour la position sélectionnée.

- ❖ Le *facteur de montage* initial de tous les éléments pour la position observée est nul.
- ❖ Si le rectangle avec ses dimensions peut entrer dans la position intérieure, le *facteur de montage* est augmenté de un.
- ❖ Si le rectangle a la même longueur que la position, le *facteur de montage* est augmenté d'un, ce qui favorise le rectangle qui correspond exactement à toute la longueur de la position.
- ❖ Si le rectangle a une largeur égale à la position, le *facteur de montage* est augmenté d'un, ce qui favorise l'élément qui correspond exactement à la largeur de la position.
- ❖ Lorsque le *facteur de montage* est calculé pour tous les éléments, ils sont triés en ordre de tri descendant d'abord par valeur de *facteur de montage*, par valeur de surface, puis par valeur de longueur, puis par valeur de largeur.
- ❖ Le premier élément dans la séquence triée est sélectionné et si le *facteur de montage* est supérieur à zéro, il est placé en position sélectionnée de sorte que le coin inférieur gauche de rectangle correspond au coin inférieur gauche de la position.
- ❖ Si la valeur du *facteur de montage* du premier rectangle dans la séquence triée est égale à zéro, la position sélectionnée est rejetée en raison du manque rectangle de montage.
- ❖ Quand il n'y a plus de rectangle, le processus d'emballage se termine.

Cet algorithme n'a pas besoin de vérifier si la table est complètement remplie pour le processus d'arrêt.

#### 3.4.2.2. Présentation graphique :

Le processus d'emballage étape par étape, selon la restriction ci-dessus :

- La table est placée à l'origine du système de coordonnées et les rectangles sont triés par ordre décroissant selon la taille de la zone.

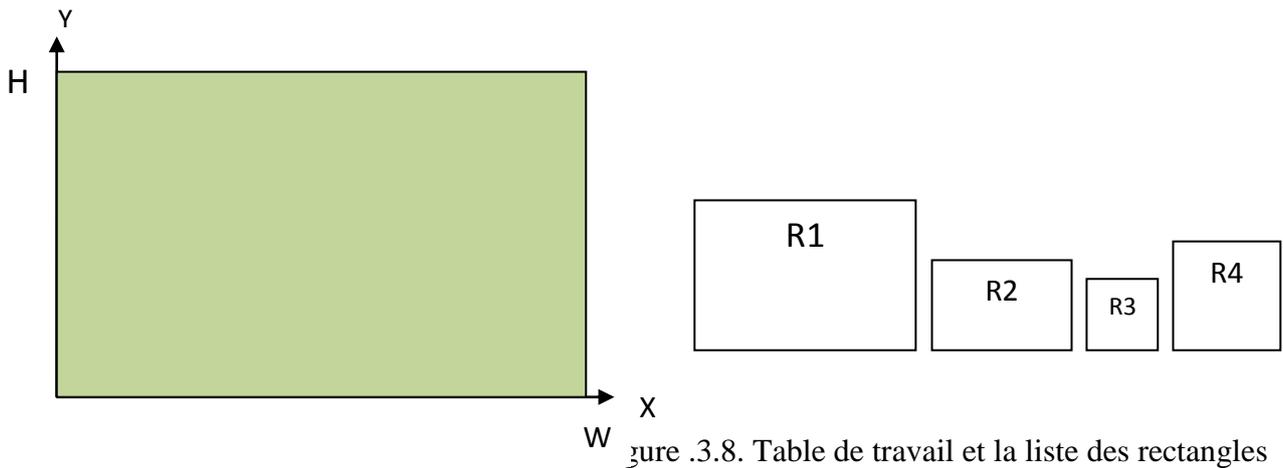


Figure 3.8. Table de travail et la liste des rectangles

### 3.4.2.2.1. Sélection de la première position et emballage d'un rectangle :

- Puisque, au début la table d'emballage, est vide, la première position libre, la Position 1, est toute la zone de table avec son coin inférieur gauche placé à l'origine.
- Les coordonnées de la position 1 sont des coordonnées de l'origine.
- La valeur calculée pour le facteur de montage pour les quatre rectangles est une, de sorte que celle avec la plus grande taille de la zone est sélectionnée pour être emballée, et c'est rectangle R1.
- Le rectangle sélectionné R1 est placé aux coordonnées de la Position 1 sélectionnée (Figure 3.9)

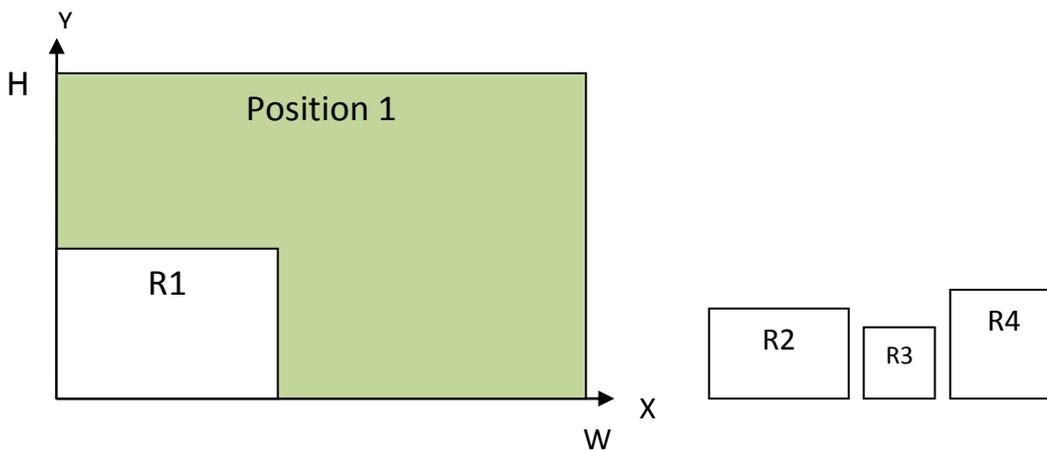


Figure 3.9 placement du premier rectangle

### 3.4.2.2.2. Trouver de nouvelles positions après le rectangle placée :

- La surface libre sur la table ne répond pas à l'exigence d'une nouvelle position en raison de sa forme et doit être ajustée. En conséquence, à la demande de direction d'emballage,

selon l'axe X de la table, avec l'orientation horizontale exclusivement, la position 1 est divisée en deux formes rectangulaires avec une ligne horizontale depuis le coin supérieur droit du dernier élément placé sur le bord droit de la position 1.

- Maintenant, il existe deux nouvelles positions (Figure .3.10), la Position 1 et la Position 2, avec ses coordonnées alignées en fonction de l'état des positions de tri. La position précédente 1 est effacée de la liste des positions.
- Les coordonnées de la nouvelle position 1 sont des coordonnées du coin inférieur droit de R1.
- Les coordonnées du nouveau poste 2 sont des coordonnées du coin supérieur gauche de R1.

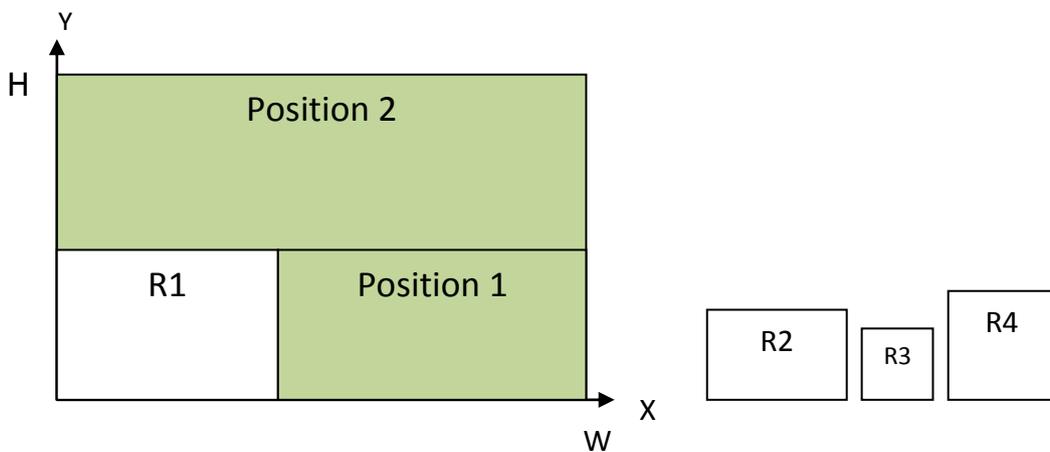


Figure .3.10 . Recherche de position

### 3.4.2.2.3. Emballage d'un nouveau rectangle :

- La position sélectionnée est la position 1.
- La valeur calculée pour le facteur de montage pour les trois rectangles est une, de sorte que celle avec la plus grande taille de la zone est sélectionnée pour être emballée, et c'est le rectangle R4.
- Le rectangle sélectionné R4 est placé aux coordonnées de la Position 1 sélectionnée.

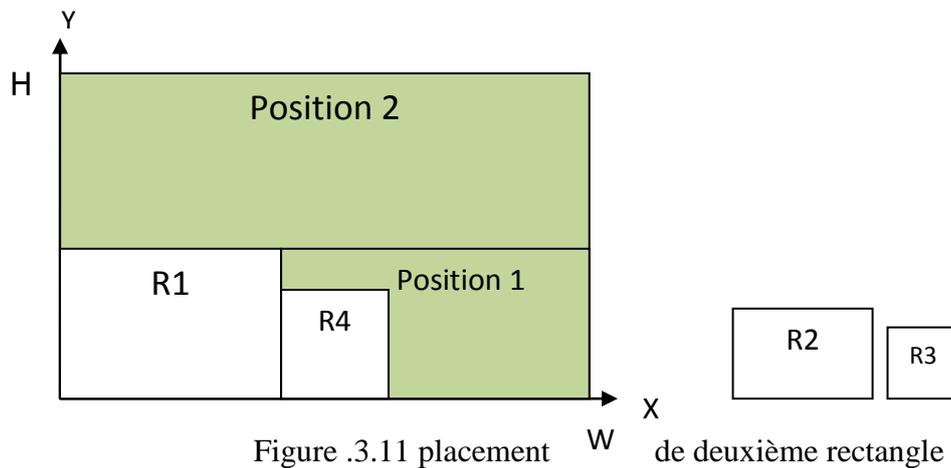


Figure .3.11 placement de deuxième rectangle

#### 3.4.2.2.4. Trouver de nouvelles positions :

- La situation sur la table est similaire à la situation après avoir placé le premier rectangle de sorte que les étapes sont également similaires. Diviser la position 1 en deux formes rectangulaires avec une ligne horizontale du coin supérieur droit du dernier élément placé sur le bord droit de la position 1.
- Il existe maintenant deux nouvelles positions et un ancien (Figure .3.12), avec ses coordonnées alignées en fonction de la condition pour les positions de tri.
- Position 1 avec coordonnées dans le coin inférieur droit de R4.
- Position 2 avec des coordonnées dans le coin supérieur gauche de R4.
- Position 3, ancienne position 2, avec des coordonnées dans le coin supérieur gauche de R1.

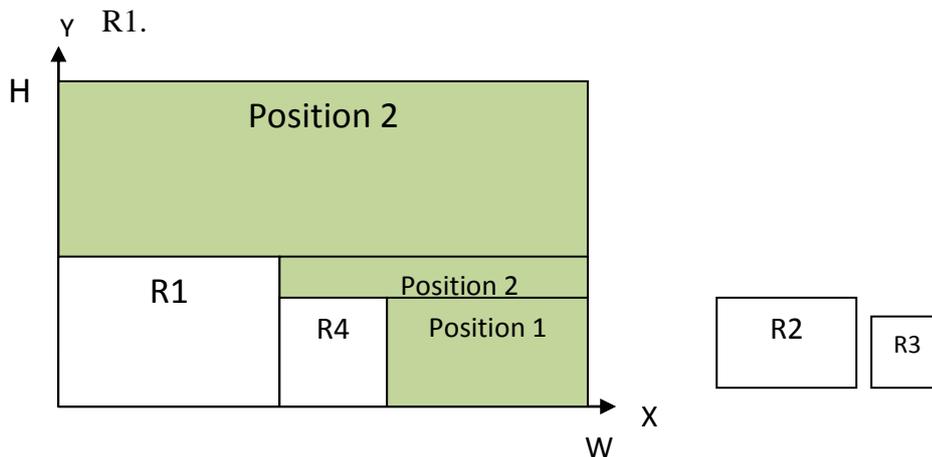


Figure .3.12 Recherche d'une nouvelle position

#### 3.4.2.2.5. Cas de position inadéquat :

- La position 2 est insuffisante en raison de sa largeur et doit être rejetée. Il ne reste plus aucun rectangle pouvant correspondre à la largeur de la position 2.

- Après avoir effacé la position 2, la position 1 doit être réglée comme indiqué ci-dessous (Figure .3.13).
- Coloré en blanc fait partie de la Position 2 rejetée et de l'espace de table inutilisé.
- Position 3 est maintenant position 2.

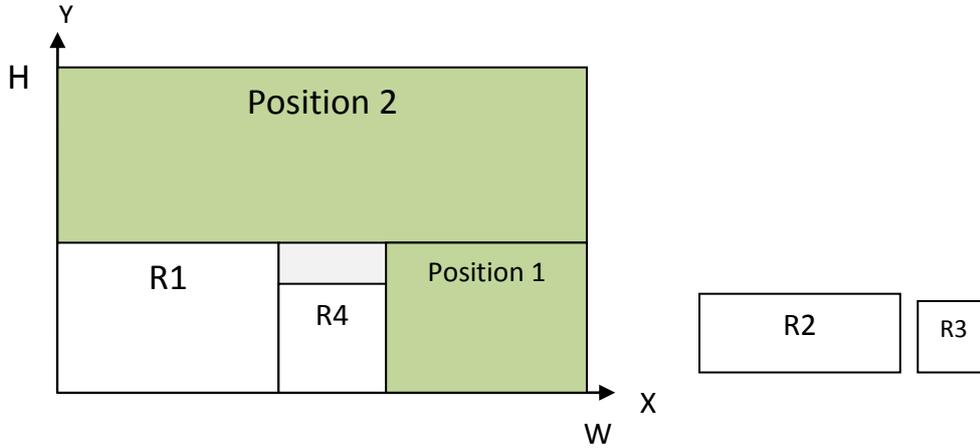


Figure .3.13 éliminé la mauvaise position

- Le processus se poursuit en emballant les deux derniers rectangles et en trouvant de nouvelles positions.

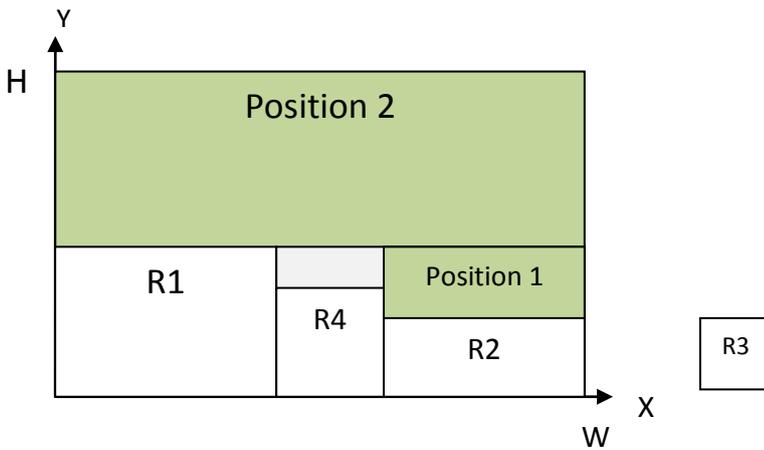


Figure .3.14 placement du 3eme rectangle

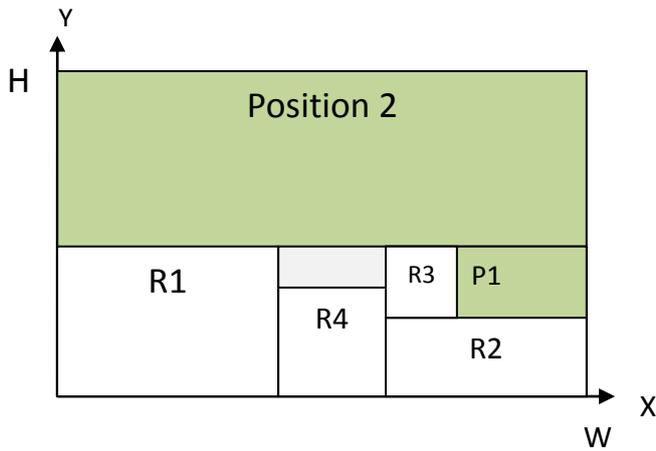


Figure .3.15 placement le 4eme rectangle

Le processus s'est terminé car il n'y a plus rectangle libre. Tous les rectangles sont emballés. Enfin, les rectangles emballés sur la table (Figure .3.16).

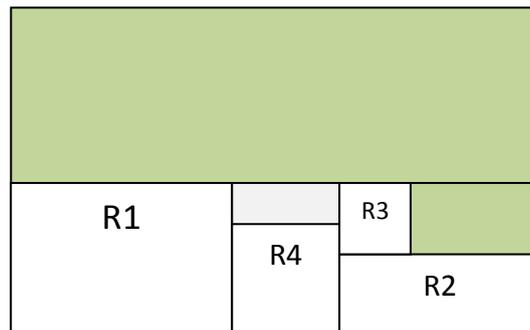


Figure .3.16 plan final

### 3.5. Vision du projet :

Après avoir travaillé dans ce projet, nous avons pensé à des idées pour développer le projet et créer d'autres horizons et opportunités pour nous bien sur et aussi pour inspirer les autres à continuer à travailler dans ce projet et à l'améliorer pour le meilleur .

#### 3.5.1. Création de logiciel :

Premières idées, il s'agit de créer un logiciel qui a la capacité de résoudre le problème de nesting 2D et de packing 3D pour tout type de forme géométrique avec un temps d'exécution minimal et un taux d'utilisation de matériel maximal .Puis ,essayer d'entrer sur le marché du logiciel de nesting et packing avec un nouveau business model, nouveau design ergonomique plein d'options et d'avantages et une nouvelle vision .

### **3.5.2. Algorithm avancée (Machine learning):**

Deuxième idée, il s'agit de développer un algorithme d'apprentissage automatique qui peut améliorer la solution de ce problème de classification et nous donner les meilleurs résultats dans le meilleur temps.

### **3.5.3. Cloud computing et Big Data:**

La troisième idée, il s'agit d'un grand projet, nous utilisons notre logiciel de Nesting avec notre puissant algorithme à pouvoir calculer le programme dans le Cloud et nous utilisons nos données pour les utiliser dans notre nouvel algorithme. Tout ça peut nous aider pour une intégration du logiciel de Nesting avec les systèmes ERP, MRP, PLM ou autres.

### **3.6. Conclusion:**

Dans ce chapitre, nous définissons le problème de packing avec les pièces rectangulaires et nous avons donné les articles les plus populaires dans ce domaine. Après nous avons donné une heuristique, que nous avons pensé qu'elle nous donne la meilleure solution, nous expliquons également comment cet algorithme fonctionne et on a donné toutes les étapes nécessaires.

## **CONCLUSION GENERALE**

Dans un problème de découpe et placement l'objectif est surtout maximiser le taux d'utilisation de la surface de travail pour moins de chutes, alors que plusieurs algorithmes et heuristiques se présentent, après plusieurs recherche et compréhension des types de problèmes, notre étude nous a conduis à choisir une heuristique présenter dans le chapitre 3 pour un problème de placement en deux dimensions.

Ce projet peut être développé en plusieurs sens quand à la méthode utilisée dans la résolution du problème ou le choix de l'outil informatique qui va avec, alors selon l'objectif de notre travail, plusieurs points on était abordé même si la définition du problème semble basique mais le contenu de la littérature académique fait preuve de diversité des pensé menée par le travail de recherche scientifique et celui du milieu de travail professionnel, par apport a ce point, dans ce projet nous avons bien voulu créé un lien de complétude entre le monde industriel et le monde pédagogique.

L'imbrication (nesting) est un outil indispensable à une entreprise de découpe et usinage CNC puisqu'elle offre un gain de temps et une optimisation du stock en générale. Les leaders de manufacture adaptent pour chaque type de matière et pour chaque type de machine un système automatisé de nesting dédié pour une meilleur planification et gestion dans un atelier opérationnel.

## REFERENCES BIBLIOGRAPHIQUES

- [1.1] Eva Hooper : two dimension utilizing evolutionary algorithm and other meta-heuristic algorithm)
- [1.2] An intelligent approach to stock cutting optimization, University of Louisville, Louisville KY USA; Cairo University, Giza Egypt)
- [1.3] Novel heuristic and Metaheuristic approach to cutting and packing, by Glenn Whitwell, BSc, School of computer science and information Technology, 2004)
- [1.4] Applying Meta- heuristic Algorithm to the Nesting Problems utilising the no-fit polygon, Graham Kendall, BSc(Hons)
- [1.5] ERIK D. Goodman, Alexander Y. Tetelbaum. and Victor M. Kureichik 1994, Michigan State University
- [1.6] Mathematical Problems in Engineering Musrrat Ali, Chang Wook Ahn, and Millie Pant
- [1.7] ELSEVIER : Computers & Chemical Engineering Volume 22, Supplement 1, 15 March 1998, Pages S1019-S1022)
- [1.8] E. Hooper and B.C.H Turton. Cardiff University, School of Engineering : artificial intelligence Review 16 : 257 :300, 2001
- [1.9] European Journal of Operational Research, Volume 190, Issue 2, 16 October 2008, Pages 295–309
- [1.10] A SURVEY ON HEURISTICS FOR THE TWO-DIMENSIONAL RECTANGULAR STRIP PACKING PROBLEM , José Fernando Oliveira, Alvaro Neuenfeldt Júnior , Elsa Silva, Maria Antónia Carravilla]
- [2.1] Eva Hopper : Two-dimensional Packing utilising Evolutionary Algorithms and other Meta-Heuristic Methods
- [2.2] Kevin Cole : the fabricator
- [2.3] Optinest blog
- [3.1] Brenda SB, Edward Jr GC, Ronald LR. Orthogonal packings in two dimensions. SIAM Journal on Computing 1980;9(4):808--26.
- [3.2] Bernard C. The bottom-left bin packing heuristic: an efficient implementation. IEEE Transactions on Computers 1983;32(8):697--707.
- [3.3] Liu D-Q, Teng H-F. An improved BL-algorithm for genetic algorithms of the orthogonal packing of rectangles. European Journal of Operational Research 1999;112(2):413--9.
- [3.4] Burke EK, Kendall G, Whitwell G. A new placement heuristic for the orthogonal stock-cutting problem. Operations Research 2004;52(4):655--71.

## Références bibliographiques

- [3.5] Wu Y-L, Huang W, Lau S-C, Wong CK, Young GH. An effective quasi-human based heuristic for solving the rectangle packing problem. *European Journal of Operational Research* 2002;141(2):341--58.
- [3.6] Zhang D, Kang Y, Deng A. A new heuristic recursive algorithm for the strip rectangular packing problem. *Computer & Operations Research* 2006;33(8): 2209--17.
- [3.7] Huang W, Chen D, Xu R. A new heuristic algorithm for rectangle packing. *Computers & Operations Research* 2007;34(11):3270--80.
- [3.8] Cui Y, Yang Y, Cheng X, Song P. A recursive branch-and-bound algorithm for the rectangular guillotine strip packing problem. *Computers & Operations Research* 2008;35(4):1281--91.
- [3.9] Hopper E, Turton B. An empirical investigation of meta-heuristic and heuristic algorithm for a 2D packing problem. *European Journal of Operational Research* 2001;128(1):34--57.
- [3.10] Zhang D, Liu Y, Chen S, Xie X. A meta-heuristic algorithm for the strip rectangular packing problem. *Lecture Notes in Computer Science* 2005;3612:1235--41.
- [3.11] Dagli CH, Poshyanonda P. New approaches to nesting rectangular patterns. *Journal of Intelligent Manufacturing* 1997;3(3):177--90.
- [3.12] Kroger B. Guillotineable bin packing: a genetic approach. *European Journal of Operational Research* 1995;84(3):645--61.
- [3.13] Bortfeldt A. A genetic algorithm for the two-dimensional strip packing problem with rectangular pieces. *European Journal of Operational Research* 2006;172(3):814--37.
- [3.14] Beasley JE. A population heuristic for constrained two-dimensional nonguillotine cutting. *European Journal of Operational Research* 2004;156(3): 601--7.
- [3.15] Egeblad J, Pisinger D. Heuristic approaches for the two- and three-dimensional knapsack packing problems 2006. Technical report 06-13, DIKU, University of Copenhagen, Denmark.
- [3.16] Martello S, Pisinger D, Vigo D. The three-dimensional bin packing problem. *Operations Research* 2000;48(2):256--67.
- [3.17] Zhang D. The design and analysis of advanced algorithms. National Defence Industry Press; 2007.

## **RESUME**

Les problèmes de Nesting et Packing sont des problèmes d'optimisation combinatoire avec de nombreuses applications dans divers domaines tels que l'industrie de l'acier et du vêtement et la conception des circuits intégrés. Depuis plusieurs décennies, le domaine de Nesting attire de nombreux chercheurs et praticiens. En fonction des applications, il faut résoudre différents types de problèmes de Nesting et Packing. Dans cette thèse, nous décrivons le Nesting de forme rectangulaire orthogonal, une liste de rectangles doit être emballée dans une zone désignée, de sorte qu'aucuns deux rectangles ne se chevauchent et Tous les rectangles sont emballés en parallèle d'axe.

**Mots clés:** Nesting, packing, rectangulaire, orthogonal

## **ABSTRACT**

Nesting and packing problems are combinatorial optimization problems with many applications in various areas such as steel and garment industry and Integrated circuit design. For several decades, the field of Nesting has been attracting many researchers and practitioners. Depending on applications, different types of Nesting and packing problems need to be solved. In this thesis we describe the Nesting of two dimensional rectangular shapes at orthogonal table, a list of rectangles has to be packed into a designated area such that no two rectangles overlap and all rectangles are packed axis-parallel.

**Key words:** Nesting, packing, rectangular, orthogonal

## **ملخص**

مشاكل التراكب والتعبئة هي مشاكل أمثلية اندماجية مع العديد من التطبيقات في مجالات مختلفة مثل صناعة الحديد والملابس وتصميم الدارات الكهربائية المدمجة. لعدة عقود، مجال التراكب والتعبئة جذب اهتمام العديد من الباحثين. اعتمادا على عدة تطبيقات، أنواع مختلفة من مشاكل التراكب والتعبئة احتاجت إلى حل. في هذه الأطروحة قدمنا شرحا لمشكل تراكب ثنائي الأبعاد خاص بالقطع المستطيلة الشكل بخاصية التوازي بحيث يكون لدينا قائمة من المستطيلات يجب أن تكون معبأة في منطقة معينة بحيث لا يوجد مستطيلين اثنين متداخلان و جميع المستطيلات معبأة بالتوازي

**الكلمات المفتاحية:** التراكب , التعبئة, المستطيلات, التوازي