

**République Algérienne Démocratique et Populaire**

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



**UNIVERSITÉ HADJ LAKHDAR « BATNA »**

**FACULTÉ DE TECHNOLOGIE**

**DÉPARTEMENT DE GÉNIE INDUSTRIEL**

**MEMOIRE DE MAGISTER**

**PRESENTÉ AU**

**LABORATOIRE D'AUTOMATIQUE ET PRODUCTIQUE**

**En vue de l'obtention du diplôme de**

**MAGISTER**

***Spécialité : Génie Industriel et Productique***

**Par**

**MEDJGHOU Aicha**

***Ingénieur d'état en Informatique***

**Thème**

***OPTIMISATION D'UN PROCESSUS DE SOUDAGE  
EFFECTUE PAR UN ROBOT BRAS  
MANIPULATEUR***

**Devant le JURY :**

<b>Dr . ABDELHAMID Samir</b>	<b>MC.A</b>	<b>Université de Batna</b>	<b>Président</b>
<b>Dr . KEZZAR Okba</b>	<b>MC.A</b>	<b>Université de Biskra</b>	<b>Examineur</b>
<b>Dr . MOUSS Mohamed Djamel</b>	<b>MC.A</b>	<b>Université de Batna</b>	<b>Rapporteur</b>
<b>Dr . BELLALA Djamel</b>	<b>MC.A</b>	<b>Université de Batna</b>	<b>Co-Rapporteur</b>
<b>Dr . SMADI Hacène</b>	<b>MC.A</b>	<b>Université de Batna</b>	<b>Examineur</b>

**ANNÉE UNIVERSITAIRE : 2013-2014**

# *Dédicace*

*Je dédie ce travail :*

*À mes très chers **parents**, à qui je dois tous et qui ont  
tout donné, que Dieu les guérisse*

*À **mes frères et mes sœurs***

*À **toute ma famille***

*À **mes professeurs***

*À **mes amis et mes collègues**, pour leurs  
encouragements*

*À toute **personne** que j'aime et  
qui m'aime*

*À **ceux et celles** qui m'ont aidé de loin ou de près*

*À **Riadh***

*Aïcha*

## REMERCIEMENTS

*Nos remerciements vont tout premièrement à Dieu tout puissant pour la volonté, la santé et la patience, qu'il nous a donné durant ces années d'études afin d'achever ce travail.*

*Je tiens à exprimer ici toutes mes reconnaissances à mon professeur Dr. Djamel BELLALA pour l'honneur qu'il m'a fait en assurant la direction et le suivi scientifique et technique du présent mémoire. Je le remercie non seulement pour sa grande contribution à l'aboutissement de ce travail et pour sa disponibilité malgré ses nombreuses activités, mais aussi pour sa grande générosité.*

*Mes remerciements s'adressent également à Dr. Med. Djamel Mouss, pour son aide, encouragement et conseils intéressants.*

*Je tiens, à exprimer ma profonde gratitude à mon enseignante Dr. Mouss N. K je la remercie pour son aide*

*Le travail présenté dans ce mémoire a été initié au sein du Laboratoire d'Automatique et Productique (LAP), dont je tiens à remercier vivement la directrice Mme L.H Mouss pour ses efforts considérables.*

*Mes remerciements vont également à l'ensemble des enseignants et des responsables du département de Génie Industriel qui ont contribué à notre formation par leurs conseils et leur aide.*

*Je remercie vivement le président et les membres de jury pour l'honneur qu'ils m'ont fait en acceptant de juger ce travail.*

*Bien évidemment, je remercie infiniment mes parents et toute ma famille.*

*Enfin, je remercie toutes les personnes de près ou de loin qui m'ont aidé à finir ce travail.*

## TABLE DES MATIERES

	<b>Page</b>
Introduction générale	1
<b>Chapitre I : Généralités sur la Robotique</b>	
Introduction	3
Partie 1 : Automatisation des systèmes	3
Définition	3
Les objectifs de l'automatisation	4
Système automatisé de production	4
La partie opérative (P.O)	4
La partie commande (P.C)	5
Le pupitre de commande	5
Localisation de la robotique dans un système de production automatisé	5
Partie 2 : Généralités sur les robots industriels	6
Définition d'un robot :(définition de l'ISO)	6
Brève historique sur la robotique	6
Domaine d'application des robots	7
Robotique industrielle (Automobile et chaîne de production)	8
Environnement hostile	9
Sécurité militaire	9
Médicale et Chirurgie	9
Autres domaines	10
Schéma fonctionnel des robots de 3 <sup>ème</sup> génération (adaptation à l'environnement)	10
Source d'énergie	11
Actionneurs	11
Transmission	11
Les capteurs	11
Le système de commande	12
Constituants mécaniques des robots	12
Les articulations	13
Autres types d'articulations	14
Notion de degré de liberté	15
Décomposition du déplacement d'un solide dans un repère	15
Degré de liberté d'un solide (ddl)	15

Degré de liberté d'une liaison –degré de liaison	15
Mécanismes	16
Espace articulaire et Espace opérationnel	17
Espace articulaire d'un robot (joint space)	17
Espace opérationnel (operational space)	17
Performances des robots industriels	17
Volume de travail ou volume atteignable	17
Charge	18
La vitesse	18
Précision	18
La fiabilité	19
Sensibilité à l'environnement	19
Conclusion	19

## **Chapitre II: Modélisation en Robotique**

Introduction	20
Modèle géométrique	20
Modèle géométrique directe	20
Calcul du MGD	20
La matrice de passage	22
Modèle géométrique inverse	23
Le modèle cinématique	25
Le modèle cinématique directe	25
Le modèle cinématique inverse	27
Le modèle dynamique	28
Le modèle dynamique directe	28
Le modèle dynamique inverse	28
Le modèle statique	29
Le modèle statique indirecte	29
Le modèle statique directe	29
Conclusion	30

## **Chapitre III: Planification de la Trajectoire d'un Bras Manipulateur**

Introduction	31
Définitions	31
Chemin (path)	31
Trajectoire	32

Les via points (Knot points)	32
Planification de trajectoire dans l'espace articulaire	32
Planification de trajectoire dans l'espace cartésien	32
Génération de trajectoire	32
Les étapes de planification de trajectoire	32
Description de la tâche	32
Sélection et utilisation d'une technique de planification de trajectoire	33
Calcul de la trajectoire	34
Obtention du modèle cinématique directe pour un mécanisme 2R	34
Les techniques de planification de trajectoire	37
Techniques articulaires pour la planification de trajectoire	37
Planification de trajectoire par la technique polynomiale	38
Utilisation du polynôme de degré trois	38
Utilisation de polynôme de degré cinq	44
Application de la stratégie 4-3-4trajectory	45
Résolution du problème appliqué au mouvement du robot	47
Segments linéaire combinés avec des paraboles	52
Segments linéaires combinés avec des courbes paraboliques et des points intermédiaires	55
Trajectoires d'ordre supérieur	56
Algorithme de la planification de trajectoire	57
Conclusion	58

## **Chapitre IV : Les Méthodes d'Optimisation**

Introduction	59
Le problème de voyageur de commerce TSP	59
Les méthodes d'optimisation	61
Les méthodes exactes	61
Principes de la méthode de branch and bound	62
Application de la méthode branch and bound au problème de TSP	62
Principe de la méthode	63
Première étape	64
Deuxième étape	67
Troisième étape	70
Circuit Optimal	71
Construction de l'arbre branch and bound	72
Les méthodes approchées	73

Métaheuristiques à base de solution unique	73
Algorithmes de recherche locale pour une optimisation locale	74
Algorithmes de recherche locale pour une optimisation globale	74
Recuit simulé	74
Recherche tabou	74
Méta heuristiques à base de population de solutions	75
Les algorithmes génétiques	75
Colonie de fourmis	76
Métaheuristiques avancées	76
Algorithme mémétique	77
Application des heuristiques au problème de TSP	78
L'heuristique Nearest Neighbor	78
L'heuristique de subtour reversal	80
Conclusion	82

## **Chapitre V : Optimisation du Processus de Soudage**

Introduction	83
Généralités sur le soudage	84
Définition	84
Types de soudage	85
Procédé de soudage par résistance électrique	85
Le soudage à l'arc électrique	86
Soudage MIG/MAG	87
Application des méthodes d'optimisation sur un système industriel de fabrication	88
Présentation de l'entreprise et situation géographique	88
Types de produits et services offerts	89
Processus de soudage au niveau de Tirsam	89
Dispersion des points de soudage sur le longeron	91
Les données nécessaires	93
Détermination de problème et l'environnement de travail	93
Application d'une heuristique de construction	95
La méthode nearest neighbor	95
Application d'une heuristique d'amélioration	96
La méthode de Subtour Reversal Heuristic	96
Proposition d'un robot de soudage	117
Motivation du choix de ce robot	118

Résultats obtenus lors de l'utilisation du robot proposé	118
Nearest neighbor	118
Sub tour reversal heuristic	119
Conclusion	120
Conclusion générale	121

## Liste des Tableaux

\*\*\*\*\* Tableaux du Chapitre I \*\*\*\*\*

**Tableau I.1 Les inventions en robotique**

\*\*\*\*\* Tableaux du Chapitre III \*\*\*\*\*

**Tableau III.1 Position, vitesse et accélération du mouvement du point B**

\*\*\*\*\* Tableaux du Chapitre IV \*\*\*\*\*

**Tableau IV.1 Coordonnées des villes**

**Tableau IV.2 Le Reversal Subtour**

**Tableau IV.3 Combination Nearest Neighbor & reversal subtour**

\*\*\*\*\* Tableaux du Chapitre V \*\*\*\*\*

**Tableau V.1 Coordonnées de points de soudage**

**Tableau V.2 variation des améliorations subtour reversal heuristic**

**Tableau V.3 Résultats NN pour les quatre parties du longeron**

**Tableau V.4 Résultats RSH pour les quatre parties du longeron**

**Tableau V.5 Chemins RSH des quatre parties du longeron**

## LISTE DES FIGURES

\*\*\*\*\* Figures du Chapitre I \*\*\*\*\*

- Fig. I.1** Les composants de l'automatisation (*adaptée*)
- Fig. I.2** Structure d'un système automatisé de production (SAP)
- Fig. I.3** Pupitre de commande d'une machine
- Fig. I.4** Automatisation, productique et robotique dans un système de production.
- Fig. I.5** Robots industriels de soudage et de peinture
- Fig. I.6** Robotique médicale
- Fig. I.7** Robot pour planter les melons(A), Robot aspirateur(B)
- Fig. I.8** Structure générale d'un robot
- Fig. I.9** Composants mécaniques d'un robot
- Fig. I.10** SMA d'un robot
- Fig. I.11** Articulation rotoïde
- Fig. I.12** Articulation prismatique
- Fig. I.13.** (a), (b), (c), (d) Articulations diverses
- Fig. I.14** ddl d'un solide
- Fig. I.15** Chaîne simple ouverte
- Fig. I.16** (a), (b).Chaines complexes
- Fig. I.17** Volume de travail du robot SCARA

\*\*\*\*\* Figures du Chapitre II \*\*\*\*\*

- Fig. II.1** structure simple ouverte
- Fig. II.2** structure de solides successifs dans une chaîne
- Fig. II.3** Passage entre MG ( $D \Leftrightarrow I$ )
- Fig. II.4** Vecteurs vitesse pour une liaison de deux solides

**Fig. II.5 Robot plan 2R1P**

\*\*\*\*\* Figures du Chapitre III \*\*\*\*\*

**Fig. III.1 Manipulateur 2R simple**

**Fig. III.2 Déplacement du point B de l'effecteur**

**Fig. III.3 Fonction lisse pour le mouvement vers une position finale**

**Fig. III.4 Application pick and place**

**Fig. III.5 Manipulateur à deux segments**

**Fig. III.6 Mouvement d'un robot manipulateur**

**Fig. III.7 Chemin d'un bras manipulateur**

**Fig. III.8 Schéma pour Segments linéaire combinés avec des sections paraboliques**

**Fig. III.9 Segments linéaires combinés avec des courbes paraboliques et des points intermédiaires**

**Fig. III.10 Technique cartésienne de planification de trajectoire**

\*\*\*\*\* Figures du Chapitre IV \*\*\*\*\*

**Fig. IV.1 voyageur de commerce**

**Fig. IV.2 le problème TSP pour cinq villes**

**Fig. IV.3 Matrice initiale**

**Fig. IV.4 Minimum de lignes RM**

**Fig. IV.5 Matrice réduite**

**Fig. IV.6 Matrice restante cas  $x_{13} = 1$**

**Fig. IV.7 Matrice réduite cas  $x_{14} = 1$**

**Fig. IV.8 Matrice réduite cas  $x_{15} = 1$**

**Fig. IV.9 Matrice réduite cas (1-5) et (2-1)**

**Fig. IV.10 Matrice réduite cas (1-5) et (2-3)**

**Fig. IV.11 Matrice réduite cas (1-5) et (2-4)**

**Fig. IV.12** Matrice réduite cas (1-3) et (2-1)

**Fig. IV.13** Matrice réduite cas (1-3) et (2-4)

**Fig. IV.14** Matrice réduite cas (1-3) et (2-5)

**Fig. IV.15** Matrice réduite cas (1-3), (2-4) et (3-2)

**Fig. IV.16** Matrice réduite cas (1-3), (5-2) et (3-5)

**Fig. IV.17** Matrice réduite cas (1-3), (2-5) et (3-2)

**Fig. IV.18** Matrice réduite cas (1-3), (2-5) et (3-4)

**Fig. IV.19** Arbre Branch and Bound (TSP à 5 villes)

**Fig. IV.20** Principes des algorithmes génétiques

**Fig. IV.21** Classification des méthodes approchées

**Fig. IV.22** Les figures (1 à 8) Application de l'algorithme du plus proche voisin

\*\*\*\*\* Figures du Chapitre V \*\*\*\*\*

**Fig. V.1** Les types de soudage

**Fig. V.2** Coupe transversale d'un assemblage

**Fig. V.3** Les phases du procédé de soudage par point

**Fig. V.4** (a) Soudure MMA. (b) Soudure TIG.

**Fig. V.5** Localisation géographique de TIRSAM

**Fig. V.6** Positionnement du Longeron dans la semi remorque

**Fig. V.7** Schéma du Longeron en 3D

**Fig. V.8** les points des soudages essentiels sur le longeron

**Fig. V.9** Schéma des heuristiques appliquées pour l'optimisation

**Fig. V.10** Nearest Neighbor appliquée au longeron

**Fig. V.11** Subtour reversal heuristic (inversion deux à la fois)

**Fig. V.12** Subtour reversal heuristic (inversion trois à la fois)

**Fig. V.13** Subtour reversal heuristic (inversion quatre à la fois)

**Fig. V.14** Subtour reversal heuristic (inversion cinq à la fois)

**Fig. V.15 Subtour reversal heuristic (inversion six à la fois)**

**Fig. V.16 Subtour reversal heuristic (inversion sept à la fois)**

**Fig. V.17 Subtour reversal heuristic (inversion neuf à la fois)**

**Fig. V.18 Subtour reversal heuristic (inversion onze à la fois)**

**Fig. V.19 Subtour reversal heuristic (inversion douze à la fois)**

**Fig. V.20 Subtour reversal heuristic (inversion treize à la fois)**

**Fig. V.21 Subtour reversal heuristic (inversion quinze à la fois)**

**Fig. V.22 Subtour reversal heuristic (inversion seize à la fois)**

**Fig. V.23 Subtour reversal heuristic (inversion dix sept à la fois)**

**Fig. V.24 Subtour reversal heuristic (inversion dix neuf à la fois)**

**Fig. V.25 Subtour reversal heuristic (inversion vingt à la fois)**

**Fig. V.26 Subtour reversal heuristic (inversion vingt sept à la fois)**

**Fig. V.27 Subtour reversal heuristic (inversion vingt huit à la fois)**

**Fig. V.28 Subtour reversal heuristic (inversion vingt neuf à la fois)**

**Fig. V.29 Subtour reversal heuristic (inversion trente trois à la fois)**

**Fig. V.30 Amélioration par la méthode Subtour Reversal Heuristic.**

# ABRÉVIATION

SAP : Système Automatisé de Production

PO : Partie Opérative

PC : Partie Contrôle

CIM : Computer Integrated Manufacturing

ISO : International Standardisation Organization

SMA : Système Mécanique Articulé

OT : Organe Terminal

DDL : Degré De Liberté

DDLN : Derge De Liaison

CAO : Conception Assistée par Ordinateur

MGD : Modèle Géométrique Directe

DH : Denavit-Hartenberg

MGI : Modèle Géométrique Inverse

MCD : Modèle Cinématique Direct

MCI : Modèle Cinématique Inverse

MDD : Le Modele Dynamique Directe

MDI : Le Modèle Dynamique Inverse

MSI : Le Modèle Statique Indirecte

MSD : Le Modèle Statique Directe

RO : Recherche Opérationnelle

TSP : Traveling Salesman Problème

IA : Intelligence Artificielle

NP: Non Polynomial

LB: Lower Bound

RM: Row Minima

SMAW: Shielded Metal Arc Welding

MMA: Manual Metal Arc

TIG : Tungsten Inert Gas

PAW: Plasma Arc Welding

MIG MAG: Metal Inert Gas Metal Active Gas

CPU: Central Processor Unit

NN: Nearest Neighbor

SRH: Subtour Reversal Heuristic

OPT :Optimal

## **Introduction générale**

L'optimisation est la science cherchant à analyser et à résoudre analytiquement ou numériquement les problèmes qui consistent à déterminer le meilleur élément d'un ensemble, au sens d'un critère quantitatif donné. De nombreux secteurs de l'industrie sont concernés tels que : les télécommunications, le transport, l'environnement, la mécanique, l'aéronautique, la robotique, ...etc.

L'optimisation combinatoire est une branche de l'optimisation en mathématiques appliquées et en informatique, également liée à la recherche opérationnelle, l'algorithmique et la théorie de la complexité. On parle également d'optimisation discrète.

Les problèmes d'optimisation sont définis selon plusieurs caractéristiques : mono ou multi objectifs, linéaire ou non linéaire, avec ou sans contraintes, stochastique ou non stochastique, etc. Cela implique qu'un outil d'optimisation peut être développé pour résoudre ces types de problèmes.

Dans sa forme la plus générale, un problème d'optimisation combinatoire consiste à trouver dans un ensemble discret une parmi les meilleures solutions réalisables, la notion de meilleure solution étant définie par une fonction objective.

Les questions qu'on peut poser lorsqu'on rencontre un problème d'optimisation sont :  
Ce problème possède-t-il une solution ?

Si ce problème possède une solution, on cherchera à la caractériser :  
Est-elle unique? Ou mieux, à la déterminer lorsque ce sera possible. On exploitera pour cela les conditions nécessaires d'optimalité.  
Enfin, on se posera la question, lorsque l'on ne sait pas déterminer explicitement les solutions du problème d'optimisation, du choix de méthodes numériques adaptées pour déterminer l'optimum.

Beaucoup de méthodes sont proposées pour la résolution des problèmes d'optimisation classées suivant deux grandes classes : ceux donnant une solution exacte appelées méthodes exactes ou déterministes d'un côté et ceux donnant des solutions approchées on parle de méthodes heuristiques de l'autre côté.

Un des problèmes le plus étudié en recherche opérationnelle est le problème du voyageur de commerce(TSP). C'est un problème d'optimisation combinatoire de type NP-difficile très complexe à résoudre, et dont nous disposons de différentes variantes.

Notre contribution est orientée vers un axe très important de la recherche scientifique à savoir la robotique. Le travail est effectué au sein d'une entreprise de fabrication et de

distribution semi-remorques ; on se basant sur un processus très important dans l'industrie qui est le processus de soudage. Les objectifs de cette étude sont:

Application de deux méthodes heuristiques d'optimisation à l'opération de soudage afin de trouver un plus court chemin le premier algorithme est constructif le deuxième est amélioratif, et l'automatisation de la tâche de soudage effectuée par l'homme on proposant un robot soudeur.

Le choix des méthodes heuristiques lors de l'optimisation est justifié comme suit :

Les méthodes exactes comme Branch and Bound nécessitent l'énumération de toutes les solutions ce qui prend trop de temps or, le temps de recherche de la solution optimale est un facteur très important et c'est à cause de lui que les problèmes d'optimisation combinatoire sont réputés si difficiles. La théorie de la complexité donne des outils pour mesurer ce temps de recherche. De plus, comme l'ensemble des solutions réalisables est défini de manière implicite, il est aussi parfois très difficile de trouver ne serait-ce qu'une solution réalisable. En pratique, on se contente très souvent d'avoir une solution approchée, obtenue par une heuristique ou une méta-heuristique. Car pour certains problèmes, l'utilisation de ces méthodes garantit la performance on matière de solution obtenue.

Notre mémoire est organisé en cinq chapitres qui peuvent être ainsi résumés :

***Le premier chapitre*** est une étude générale sur les robots industriels dont l'objectif est l'apprentissage de la terminologie de la robotique.

***Le deuxième chapitre*** est dédié à la présentation des différents modèles utilisés dans la robotique : modèles géométriques, modèles cinématique, les modèles dynamiques et les modèles statiques

***Le troisième chapitre*** est consacré à la l'étude de la planification de la trajectoire des robots lors de leur mouvement, plusieurs stratégies sont exposées dans ce chapitre.

***Le quatrième chapitre*** est consacré à la présentation des méthodes d'optimisation est ces deux variantes (exactes et approchées).

***Le dernier chapitre*** est une application des méthodes heuristiques d'optimisation: nearest neighbor en tant que méthode constructive de solution optimale et la deuxième méthode appliquée est subtour reversal heuristic pour améliorer les résultats de la première méthode.

Nous terminons, avec une conclusion générale relative à l'utilisation des méthodes d'optimisation puis on donnant des perspectives de recherche prometteuses.



**Chapitre01**  
*Chapitres*

**Généralités sur la Robotique**  
*Généralités sur la Robotique*

## Chapitre I

### Généralités sur la Robotique

#### I. 1. Introduction

La robotique est la science qui s'intéresse aux robots, ce terme est apparu en 1942. En fait, il s'agit d'un domaine multidisciplinaire, on y trouve des aspects concernant la mécanique, l'informatique, l'automatique, etc.

Les robots prennent actuellement une place importante dans notre vie, on les trouve dans toutes les entreprises pour accélérer la production ou pour agir là où l'homme ne peut travailler à cause du danger, à la maison pour aider à la cuisine et aux tâches ménagères, dans les services publics tels que les hôpitaux, la construction des casernes de pompiers, de police, assister les médecins lors de leurs opérations, etc.

La robotique est devenue en quelques années une science importante qui ne cesse d'évoluer grâce à l'automatisation.

#### I. 2. Partie 1 : Automatisation des systèmes

##### I. 2. 1. Définition

L'automatisation est considérée comme l'étape d'un progrès technique où apparaissent des dispositifs techniques susceptibles de seconder l'homme, non seulement dans ses efforts musculaires, mais également dans son travail intellectuel de surveillance et de contrôle (Voir figure I.1).

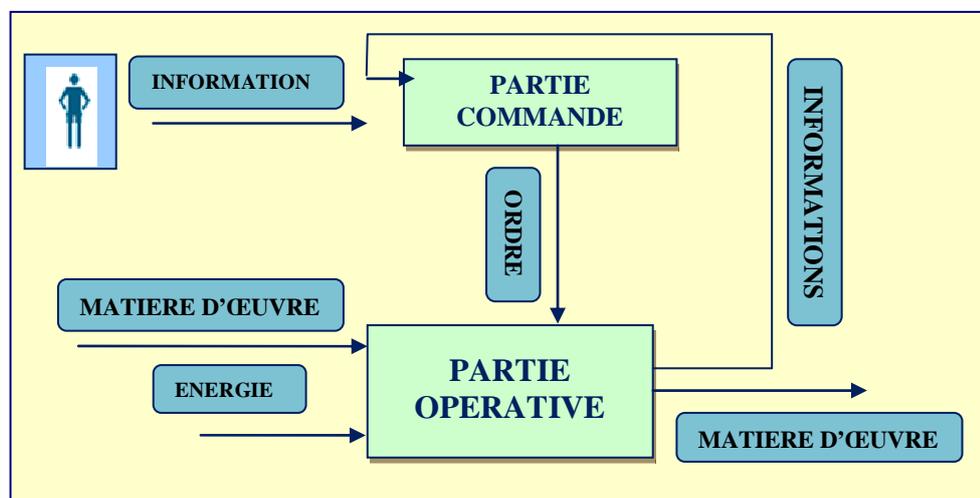


Fig. I.1 Les composants de l'automatisation (adaptée)

## I. 2. 2. Les objectifs de l'automatisation

- Simplification du travail de l'humain,
- Augmenter la productivité et la sécurité dans le système,
- Économiser les matières premières et l'énergie,
- Rendre le système flexible,
- Innovation et amélioration de la qualité [1].

## I. 2. 3. Système automatisé de production

Un système automatisé consiste en trois parties essentielles (voir figure I.2) :

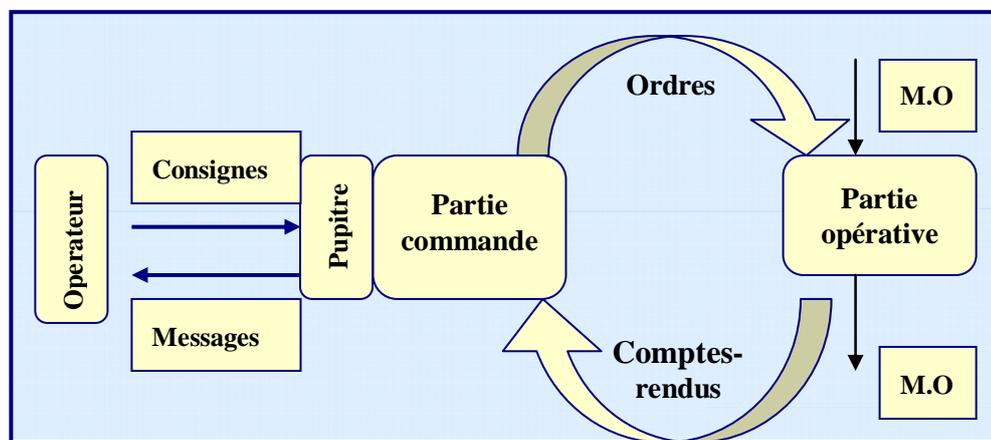


Fig. I.2 Structure d'un système automatisé de production (SAP)

### I. 2. 3. a. Partie opérative (P.O)

C'est le corps du système, il est constituée de :

- Éléments mécaniques
- Actionneurs : vérins, moteurs
- Pré-actionneurs : distributeurs, contacteurs
- Éléments de détection : capteurs, détecteurs.

La partie opérative agit sur la matière d'œuvre en entrée à partir d'ordres envoyés par la partie commande (marche d'une machine/arrêt d'une machine), et renvoie à cette dernière des informations sur son état ou sur l'environnement (par exemple le niveau de l'eau dans une station de pompage détecté par les différents capteurs). En générale cette partie permet d'apporter de la valeur ajoutée à la matière d'œuvre.

### I. 2. 3. b. Partie commande (P.C)

Le rôle principal de cette partie est le traitement de l'information issue des capteurs et le pilotage de la partie opérative (contrôle et gestion).

### I. 2. 3. c. Pupitre de commande

Permet à l'opérateur d'intervenir sur le système, donner des ordres (départ/arrêt), introduire les paramètres de fonctionnement, et à effectuer des opérations pour la maintenance du système (voir figure I.3) [2].



Fig. I.3 Pupitre de commande d'une machine

### I. 2. 3. d. Localisation de la robotique dans un système de production automatisé

Avant de robotiser un système de production, il doit s'automatiser d'abord en utilisant des outils modernes d'automatisation tels que : les robots et d'autres machines afin de garantir une certaine flexibilité.

La productique est la production des biens manufacturés tout en utilisant les technologies les plus récentes. C'est l'équivalent du CIM (computer integrated manufacturing) qui signifie la fabrication intégrée par ordinateur, en ce sens la robotique en tant que science des robots n'intervient que comme un sous ensemble. Le système de production peut maintenant se décomposer en un nombre réduit de classes de machine comme le montre la figure I.4.

- Machines de conception : elles dépendent du produit à fabriquer, son mode de fabrication, les machines de fabrication, etc.
- Machines d'exécution : on distingue deux types, celles qui réalisent le transfert et l'autre types pour la transformation des pièces (peinture, soudure, etc.).
- Machines de commande/contrôle : pour les machines d'exécution.

- Machines de communication : qui réalisent le transfert, la collecte et le traitement de l'information.

La tâche décision est effectuée par l'opérateur humain pour assurer la surveillance et la maintenance et les robots sont principalement considérés en tant que machines de transfert et moins fréquemment comme machine de procédé [3].

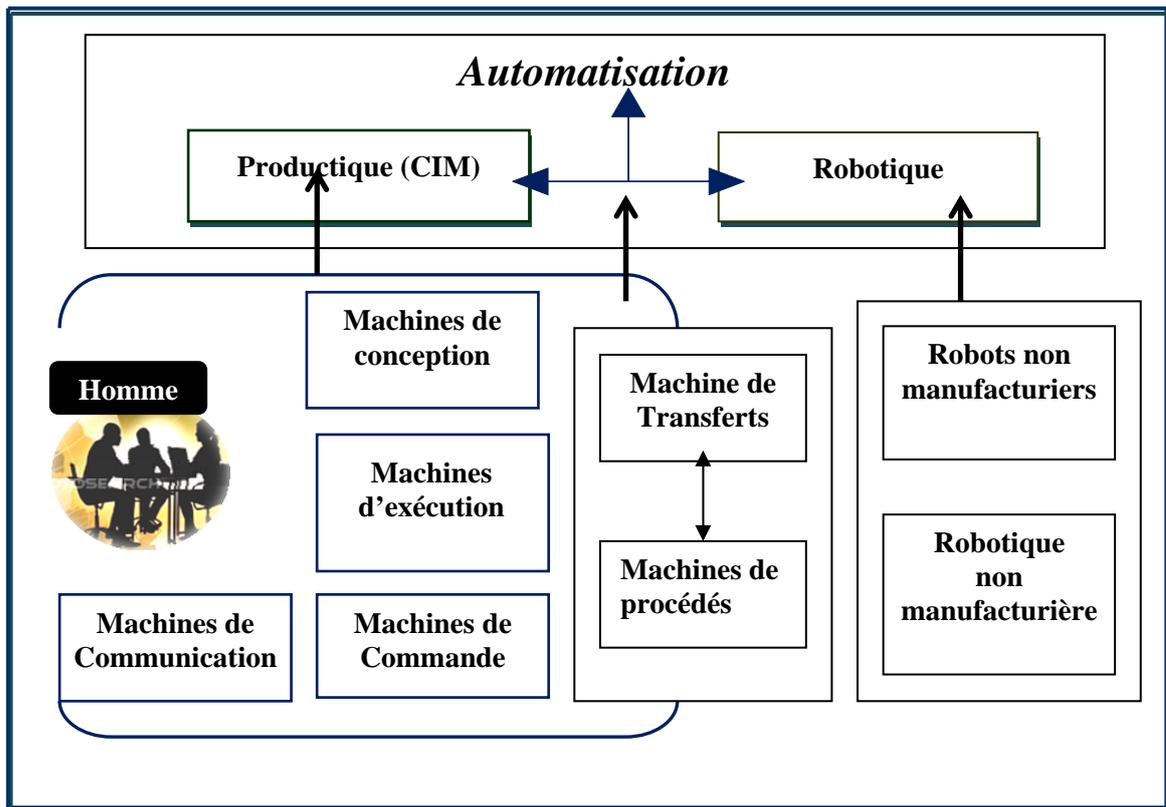


Fig. I.4 Automatisation, productique et robotique dans un système de production.

## I. 3. Partie 2 : Généralités sur les robots industriels

### I. 3. 1. Définition d'un robot :(définition de l'ISO)

C'est une machine formée par un mécanisme incluant plusieurs degrés de libertés, ayant souvent l'apparence d'un ou plusieurs bras se terminant par un poignet capable de tenir des outils, des pièces ou un dispositif d'inspection [5].

### I. 3. 2. Brève historique sur la robotique

L'historique de la robotique est résumé par le tableau I.1 suivant :

<b>Histoire de la Robotique</b>	<b>Période ou année</b>	<b>invention</b>
	<b>Seconde Guerre Mondiale</b>	<ul style="list-style-type: none"> <li>développement des servomécanismes.</li> </ul>
	<b>1950</b>	<ul style="list-style-type: none"> <li>Développement de manipulateurs « maître-esclave » contrôlés à distance pour manipuler des substances radioactives.</li> </ul>
	<b>1954</b>	<ul style="list-style-type: none"> <li>Premier robot programmable.</li> </ul>
	<b>1961</b>	<ul style="list-style-type: none"> <li>apparition d'un robot sur une chaîne de montage de General Motors, premier robot avec contrôle en effort.</li> </ul>
	<b>1968 à 1969</b>	<ul style="list-style-type: none"> <li>Début de recherches intensives sur les robots contrôlés par ordinateur (Boston arm, Stanford arm)).</li> </ul>
	<b>1971</b>	<ul style="list-style-type: none"> <li>Création de la JIRA au Japon.</li> </ul>
	<b>1974</b>	<ul style="list-style-type: none"> <li>La firme Cincinnati Milacron introduit le robot industriel contrôlé par ordinateur dénommé «The Tomorrow Tool» ou T3.</li> </ul>
	<b>1975</b>	<ul style="list-style-type: none"> <li>Création de la RIA aux USA.</li> </ul>
	<b>1990</b>	<ul style="list-style-type: none"> <li>Prévisions réalisées, 70% du marché est contrôlé par le Japon.</li> </ul>
	<b>1998</b>	<ul style="list-style-type: none"> <li>La robotique industrielle représente un chiffre d'affaire de 4.2 milliards de \$. Il y a 700.000 robots industriels dans le monde.</li> </ul>

Tableau I.1 Les inventions en robotique [4] et [5]

### I. 3. 3. Domaine d'application des robots

La robotique peut permettre de promouvoir la prévention des accidents de travail et des maladies professionnelles, durant leur phase automatique, les robots fonctionnent hors présence du personnel. Cette technique conduit donc à éloigner l'homme des machines et des

produits traités pour des raisons de sécurité dans les différents systèmes. Plusieurs facteurs peuvent provoquer des risques dans le système tel que :

- Les vibrations et le bruit
- La température élevée
- La poussière et le gaz
- La présence des produits chimique qui peuvent influencer d'une façon négative sur le corps humain, etc.

Une robotisation adéquate empêchera que ces facteurs puissent avoir une influence sur le personnel dans la zone de travail. Les robots industriels, signalent un grand nombre d'applications en insistant sur l'intérêt de confier aux robots les rôles ingrats, routiniers, monotones et dangereux. Dans ce qui suit nous en citons les plus intéressantes.

#### **I. 3. 3. a. Robotique industrielle** (Automobile et chaîne de production)

- **Le moulage sous pression** : le robot peut être utilisé pour les opérations de déchargement, de refroidissement du moulage et son rangement d'une façon automatique.
- **Le soudage à l'arc et par point** : le robot peut effectuer un soudage continu ou par point après positionnement (voir figure I.5).
- **Le découpage au chalumeau** : découpage des tôles d'acier.
- **Les applications de peinture par jet** : le robot remplace l'opérateur humain après apprentissage (voir figure I.5).
- **Applications industrielles diverses** : qui manipulent des matériaux de différentes natures tels que le verre, le métal, etc.

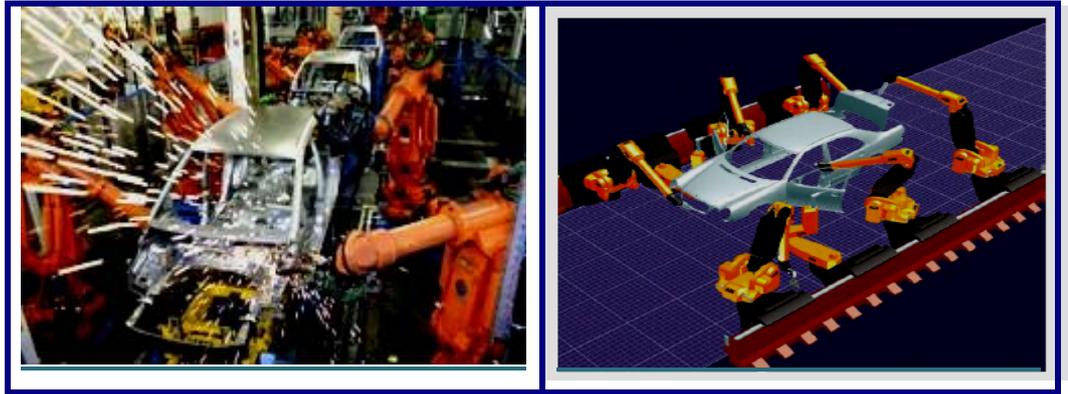


Fig. I.5 Robots industriels de soudage et de peinture

### **I. 3. 3. b. Environnement hostile**

Les robots sont devenus des outils essentiels pour la manipulation à distance en milieu hostile afin d'explorer des environnements inconnus ou dangereux :

- Nucléaires : mesure de la radioactivité.
- sous-marins.
- exploration spatiale : le CANADARM et la NASA.

### **I. 3. 3. c. Sécurité militaire**

Les robots de surveillance, sécurité et défense sont plus robustes avec une meilleure précision. Ils sont conçus pour prendre de l'abus et pour protéger des vies en se mettant dans des situations difficiles ou dangereuses où les humains préfèrent ne pas se retrouver pour éviter les dégâts. Ils permettent à l'opérateur de surveiller, d'une façon sécuritaire une situation à distance et accéder à des endroits difficiles, tout en augmentant la compréhension de la situation ainsi que les options tactiques.

### **I. 3. 3. d. Le médicale et la chirurgie**

Cette technologie permet d'améliorer le pouvoir de perception au chirurgien, de prendre la décision convenable et d'agir grâce à l'utilisation des moyens tels que les systèmes de vision, les capteurs et les actionneurs spécifiques (voir figure I.6).

La sécurité des robots médicaux doit être prise en considération au moment de la conception car les tâches sont effectuées sur un être humain ce qui implique que les conditions de travail évoluent en fonction du patient, de sa position sur la table, de l'accessibilité des organes, etc.

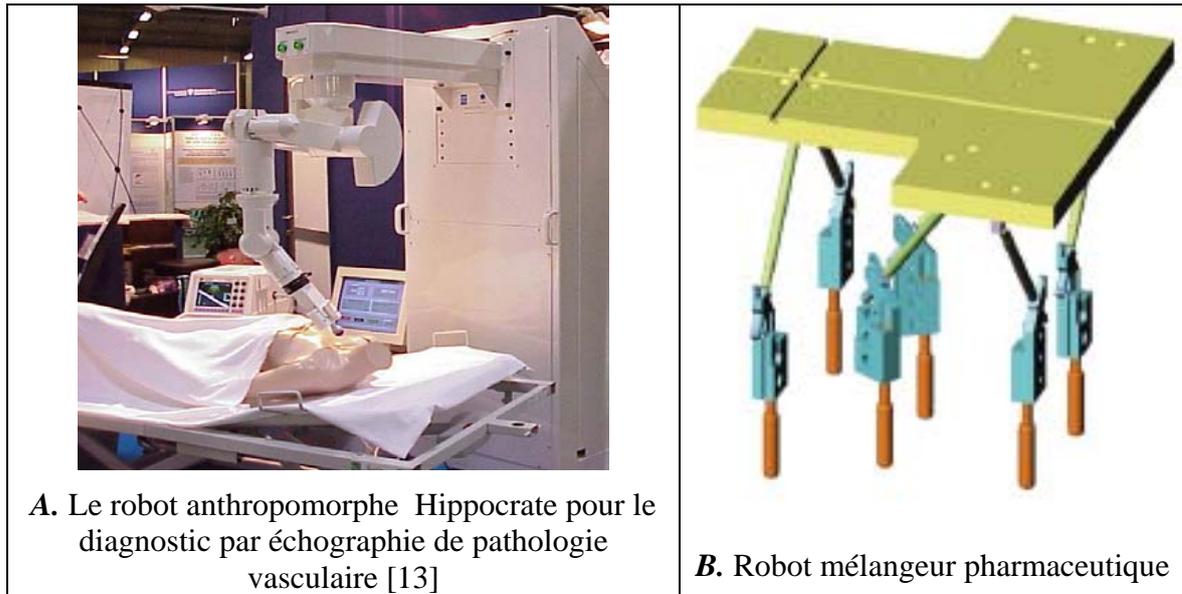


Fig. I.6 Robotique médicale

### I. 3. 3. e. Autres domaines

- Agriculture.
- Service clientèle et Loisirs, etc [7], [6].

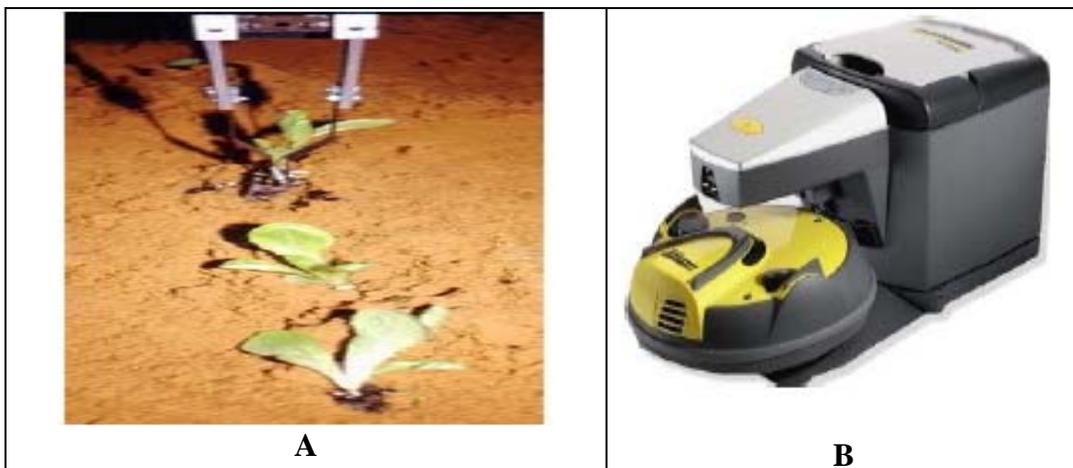


Fig. I.7 Robot pour planter les melons (A), Robot aspirateur (B)

### I. 3. 4. Schéma fonctionnel des robots de 3<sup>ème</sup> génération (adaptation à l'environnement)

Un robot industriel est constitué d'une structure mécanique animée par des actionneurs (moteurs électriques, pneumatiques ou hydrauliques, vérins linéaires ou rotatifs, etc.) à partir d'ordres élaborés par un calculateur, qui peut, par ailleurs, piloter des outils (de soudage,

d'encollage, d'usinage, etc.) ou gérer son environnement (positionneurs, convoyeurs, magasins d'outils ou de pièces, etc.). Cette constitution est illustrée par la figure I.8 [8].

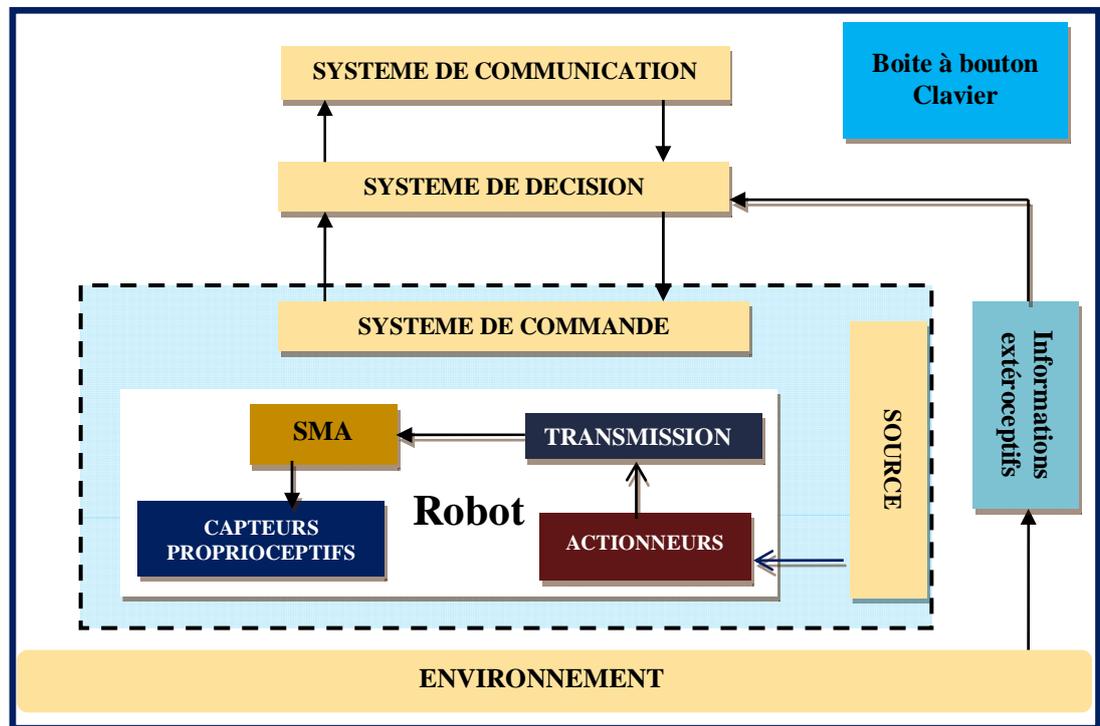


Fig. I.8 Structure générale d'un robot

#### I. 3. 4. a. Source d'énergie

On peut trouver des robots à actionneurs pneumatiques hydraulique ou électriques [3].

#### I. 3. 4. b. Actionneurs

Rôle : Le rôle d'un actionneur est de convertir une énergie électrique, pneumatique, etc, en une énergie bien souvent mécanique [9].

#### I. 3. 4. c. Transmission

Ensemble des pièces mécaniques assurant le mouvement des articulations du robot à partir des axes des moteurs.

#### I. 3. 4. d. Les capteurs

Partie fournissant des informations sur l'état du robot au moment de l'exécution d'une tâche, on distingue deux types : les **capteurs internes** pour le contrôle de la trajectoire et les **capteurs externes** destinés à la vérification si la trajectoire est correcte et la mesure des forces robot-environnement, etc.

### I. 3. 4. e. Le système de commande

Il comporte pour un robot archétype quatre niveaux :

#### 1) Niveau régulation

C'est le niveau le plus bas il permet de suivre les trajectoires connues à l'avance.

#### 2) Le niveau adaptation

Le robot sauvegarde l'environnement en évitant de tout obstacle. Ce niveau fournit en permanence des informations grâce à ces capteurs locaux. Il permet de modifier la trajectoire suivie primitivement.

#### 3) Niveau raisonnement

Les capteurs externes globaux fournissent la fonction perception et permettent d'analyser l'environnement et d'établir la stratégie d'exécution de la tâche.

#### 4) Niveau supervision

L'homme joue un rôle de superviseur pour fournir des informations pertinentes aux autres niveaux et la prise de décision [3].

### I. 3. 4. f. Constituants mécanique des robots

Deux sous-ensembles distincts : système mécanique articulé(SMA) et un ou plusieurs organes terminaux (OT) (voir figure I.9).

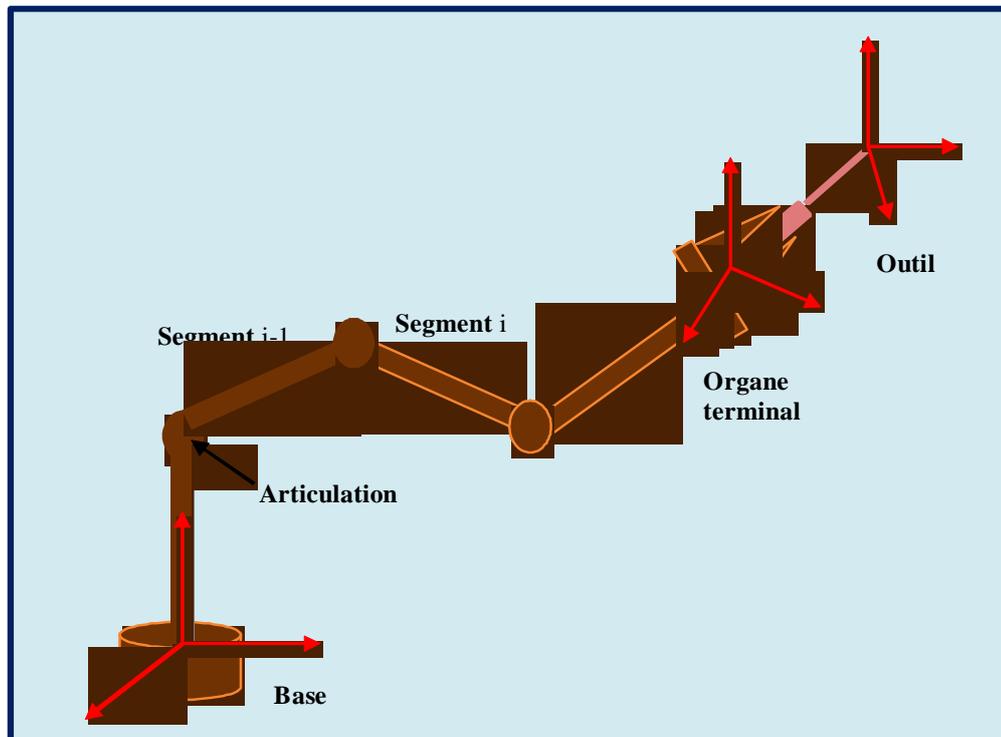


Fig. I.9 Composants mécaniques d'un robot

### a. Organe terminal

La position et l'orientation d'un manipulateur est décrite par le système de coordonnées de l'outil terminal attaché à l'organe terminal. Ce système est décrit par rapport au système de coordonnées de la base du manipulateur.

**Rôle de l'OT :** manipuler et/ou transformer des objets. On distingue deux types :

- Monofonctionnel
- Multifonctionnel

### b. Le système mécanique articulé (SMA)

Une chaîne cinématique de segments généralement rigides assemblés par des liaisons appelées *articulation* pour obtenir un mouvement relatif de segments voisins. Les positions relatives des segments voisins sont mesurées par des capteurs de position attachés à chaque articulation, le schéma général du SMA est illustré par la figure I.10.

**Rôle du SMA :** amener l'organe terminal dans une situation (position et orientation) donnée selon des caractéristiques de vitesse et d'accélération données [10].

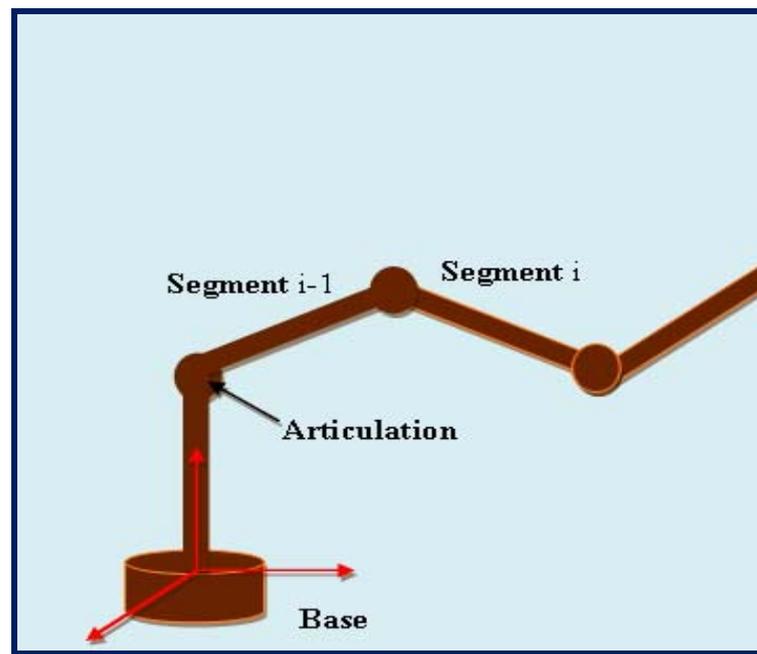


Fig. I.10 SMA d'un robot

### I. 3. 4. g. Les articulations

Une articulation lie deux corps successifs en limitant le nombre de degré de liberté (voir § I.3.5) de l'un par rapport à l'autre. Une articulation peut être rotoïde ou prismatique. Les figures I.11, I.12 et I.13 représentent différentes articulations.

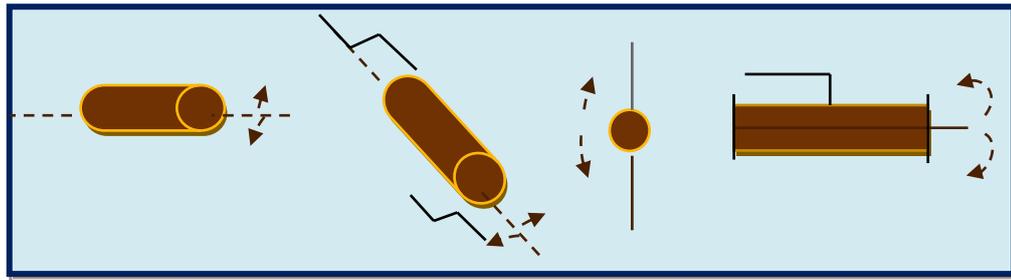


Fig. I.11 Articulation rotoïde

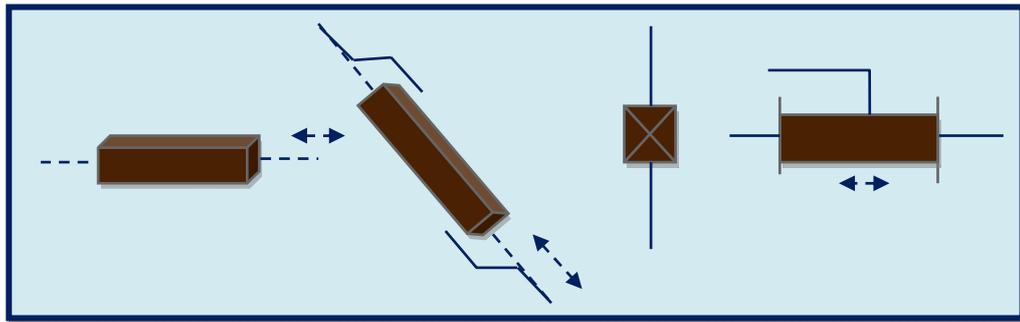
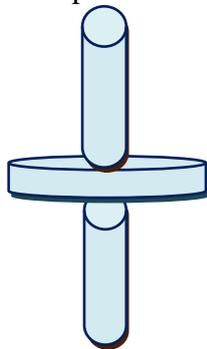


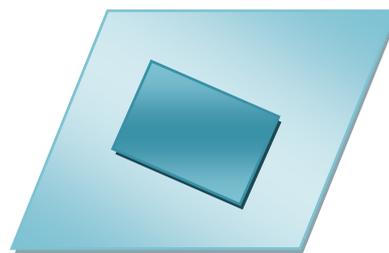
Fig. I.12 Articulation prismatique

### I. 3. 4. h. Autres types d'articulations

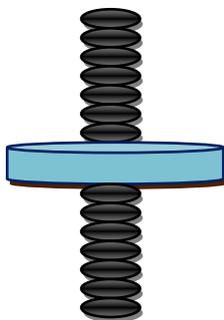
La figure I.13 représente différentes articulations.



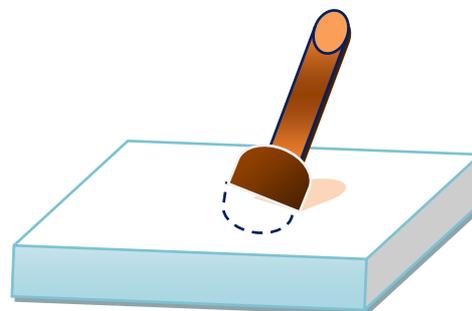
(a) Articulation cylindrique



(b) Articulation



(c) Articulation Visse.



(d) Articulation sphérique

Fig. I.13. (a), (b), (c), (d) Articulations diverses

### I. 3. 5. Notion de degré de liberté

#### I. 3. 5. a. Décomposition du déplacement d'un solide dans un repère

La position d'un solide (S) en déplacement quelconque dans l'espace affine est déterminée par un repère  $R = (o, \vec{e}_x, \vec{e}_y, \vec{e}_z)$ . Son déplacement peut se décomposer, dans un ordre quelconque, en six déplacements élémentaires qui sont illustrés par la figure I.14 :

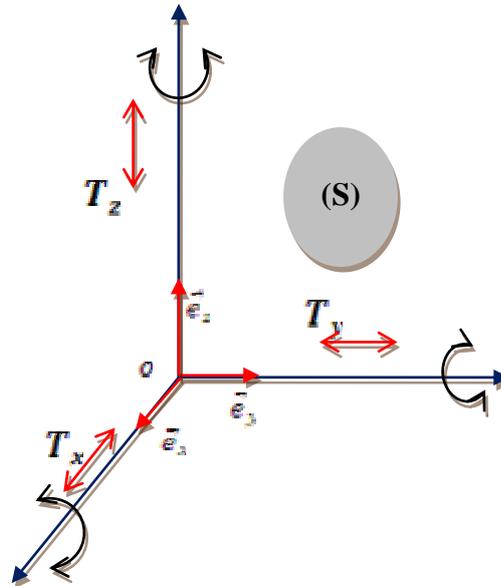


Fig. I.14 Ddl d'un solide

Rotation  $R_x$  autour de  $(o, \vec{e}_x)$ , rotation  $R_y$  autour de  $(o, \vec{e}_y)$ , rotation  $R_z$  autour de  $(o, \vec{e}_z)$ .

Translation  $T_x$  suivant  $(o, \vec{e}_x)$ , translation  $T_y$  suivant  $(o, \vec{e}_y)$ , translation  $T_z$  suivant  $(o, \vec{e}_z)$ .

#### I. 3. 5. b. Degré de liberté d'un solide (ddl)

Par définition, les ddl d'un solide sont ces déplacements élémentaires :

- Dans l'espace : six ddl, trois rotations et trois translations
- Dans le plan : trois ddl, une rotation et deux translations.

#### I. 3. 5. c. Degré de liberté d'une liaison – degré de liaison

Une liaison élémentaire entre deux solides ( $S_1$ ) et ( $S_2$ ) est créée par le contact d'une surface associée au solide ( $S_1$ ) sur une surface associée à ( $S_2$ ). Pour caractériser la nature de leur liaison, il faut étudier les mouvements relatifs de ( $S_1$ ) par rapport à ( $S_2$ ).

### Définition

- Le degré de liberté (ddl) d'une liaison est le nombre de déplacements élémentaires indépendants autorisés par cette liaison.
- Par opposition, le degré de liaison (ddl<sub>n</sub>) est le nombre de déplacements élémentaires interdits.

Le degré de liberté est donc une variable qui peut prendre deux états auxquels on peut associer :

- Le chiffre 0 lorsque le ddl est impossible.
- Le chiffre 1 lorsque le ddl est possible.

A un degré de liberté supprimé correspond un degré de liaison [11].

$$\text{Degré de liberté (ddl)} + \text{degré de liaison (ddl}_n\text{)} = 6$$

### I. 3. 6. Mécanismes

On appelle mécanisme un ensemble de solides reliés deux à deux par des liaisons. On distingue deux types de mécanismes :

- Les mécanismes en chaîne simple ouverte (ou en série) montrés dans la figure I.15. Lorsque l'on parcourt le mécanisme, on ne repasse jamais deux fois sur la même liaison, ou sur le même solide. Ce type de système est le plus répandu.
- Les mécanismes en chaîne complexe, i.e., tout ce qui n'est pas en série (au moins un solide avec plus de 2 liaisons). De tels systèmes se subdivisent en 2 groupes : les chaînes structurées en arbre, et les chaînes fermées (dont l'avantage est d'être à priori plus rigide, plus précis et capable de manipuler de lourdes charges)(voir figure I.16) [12].

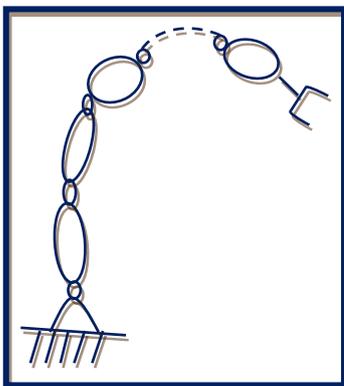


Fig. I.15 Chaîne simple ouverte

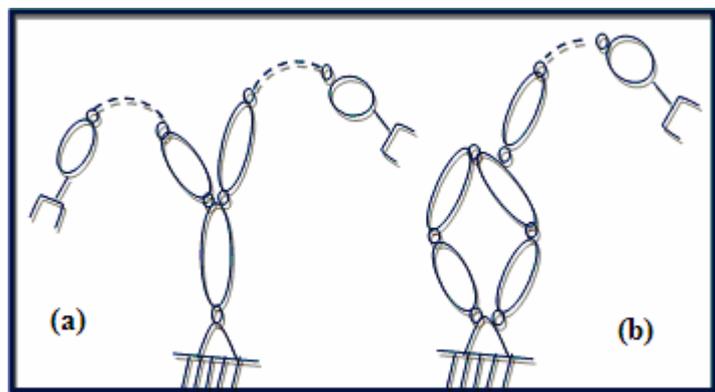


Fig. I.16 (a), (b). Chaînes complexes

### I. 3. 7. Espace articulaire et espace opérationnel

#### I. 3. 7. a. Espace articulaire d'un robot (joint space)

C'est l'espace des variables articulaires décrit par l'état des segments du robot, autrement dit c'est l'état du robot exprimé par la situation de ces différents corps.

#### I. 3. 7. b. Espace opérationnel (operational space)

C'est l'espace où la situation de l'organe terminal est déterminée tels que l'espace de coordonnées :

- Cartésiennes
- Cylindrique
- Sphériques [10]

### I. 3. 8. Performances des robots industriels

#### I. 3. 8. a. Volume de travail ou volume atteignable

C'est l'ensemble de points que peut atteindre un élément particulier du robot. Ce critère possède une forme et un volume, la figure I.17 indique le volume de travail pour le robot SCARA.

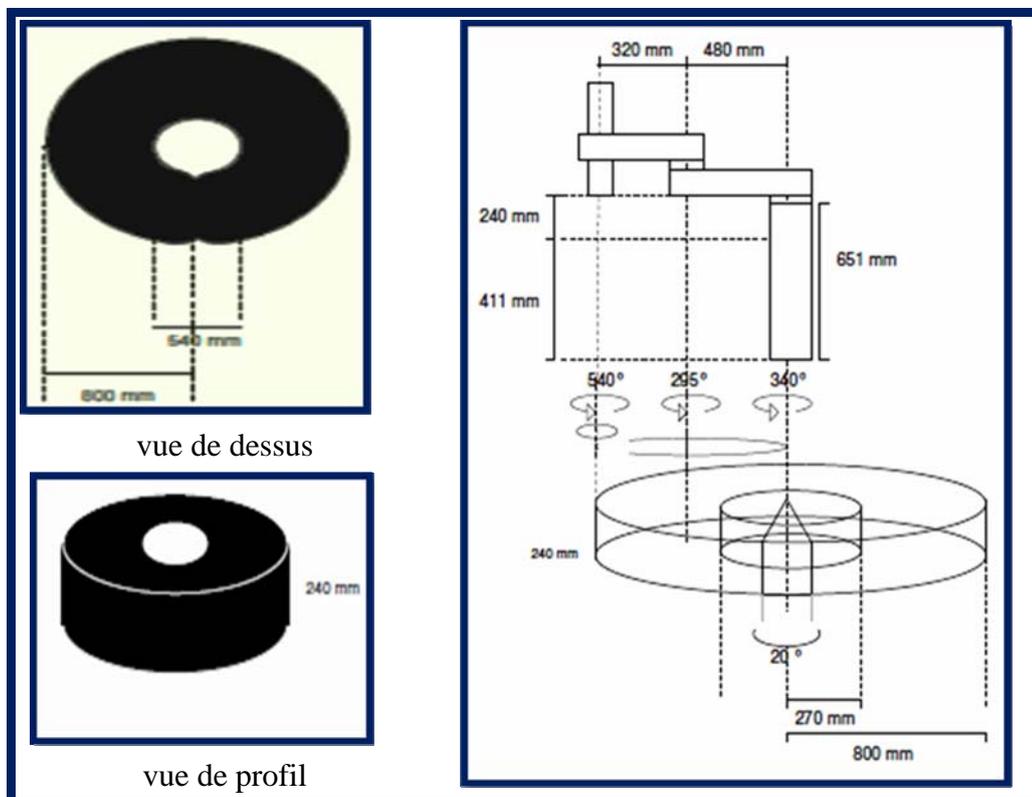


Fig. I.17 Volume de travail du robot SCARA [13]

### **I. 3. 8. b. Charge**

Les couples et forces subis par les différents éléments des chaînes cinématiques des robots résultent de l'effet combiné de différents efforts :

#### **1. Efforts statiques**

- forces de gravité qui s'exercent sur la charge manutentionnée et sur la structure,
- dispositifs d'équilibrage qui compensent, au moins partiellement,
- les forces de gravité,
- forces et couples exercés par un outil.

#### **2. Efforts dynamiques**

Les effets des vitesses et les accélérations des différents axes.

### **I. 3. 8. c. La vitesse**

La vitesse mis pour accomplir les tâches on faisant abstraction sur la nature des tâches à réaliser.

### **I. 3. 8. d. Précision**

Caractérisée par plusieurs critères normalisés tels que :

#### **1) L'exactitude de pose**

C'est un critère qui caractérise la précision du robot lorsque la trajectoire est apprise hors ligne, par exemple à partir d'un poste CAO (Conception Assistée par Ordinateur), elle compare les distances entre deux poses programmées et les barycentres des points restitués. Outre les inexactitudes de pose locale et de répétabilité, elle intègre les écarts entre le modèle géométrique du robot et sa réalité physique. Des procédures d'étalonnage sont parfois utilisées pour minimiser cet écart.

#### **2) L'exactitude et la répétabilité de trajectoire**

Elles définissent des performances similaires à celles de pose ; elles représentent respectivement la distance de la trajectoire apprise à la ligne joignant les barycentres des points restitués et le rayon du cylindre enveloppe de ces points. etc.

### **I. 3. 8. e. La fiabilité**

Définie par le taux de pannes

### **I. 3. 8. f. Sensibilité à l'environnement**

Les robots industriels doivent accepter les sollicitations de l'environnement dans lequel ils évoluent (température, degré d'humidité, un niveau d'étanchéité contre le polluant tel que la poussière, la projection d'eau, etc.).[3], [8], [13]

## **I. 4. Conclusion**

Nous avons présentés dans ce premier chapitre

- L'automatisation en tant que solution pour l'élimination des tâches répétitives
- La productique est forme d'évolution en intégrant l'ordinateur dans les processus de fabrication des biens manufacturiers.
- La robotique afin de décharger l'homme de travailler dans les milieux hostiles et dangereux.



**Chapitre02**  
*Chapitre02*

**Modélisation en Robotique**  
*Modélisation en Robotique*

## Chapitre II

### Modélisation en Robotique

#### II. 1. Introduction

La conception et la commande des systèmes mécanique articulés nécessitent le calcul d'un modèle mathématique pour le but de le représenter dans l'environnement où il évolue.

Cette modélisation consiste à exprimer d'une part :

- La situation de l'OT et la transformation entre l'espace articulaire et l'espace opérationnel, c'est le modèle géométrique.
- La vitesse de l'OT, on parle du modèle cinématique.
- Les couples et les forces exercés par les actionneurs de l'autre part, c'est le modèle dynamique.

#### II. 2. Modèle géométrique

##### II. 2. 1. Modèle géométrique directe (MGD)

Le modèle géométrique d'un mécanisme regroupe les contraintes géométriques qui doivent être respectées par les variables articulaires  $\theta_i$  afin d'établir la relation entre la configuration du mécanisme définie dans l'espace des coordonnées généralisées et la configuration du mécanisme décrite dans le repère cartésien [14].

$$\begin{aligned} f : N &\rightarrow M \\ \theta &\rightarrow x = f_{MGD}(\theta) \end{aligned} \tag{II.1}$$

##### II. 2. 1. a. Calcul du MGD

Pour calculer le MGD, on suppose que les segments du robot sont parfaitement rigides et les articulations mécaniquement parfaites, il existe plusieurs méthodes pour déterminer le MGD tel que :

- La matrice 3\*3
- La matrice 4\*4

- La méthode des Denavit Hartenberg, etc.

La méthode des Denavit Hartenberg modifié est la plus utilisée. On considère une structure ouverte simple (voir figure II.1)

A chaque corps  $i$  du bras est associé un repère  $R_i = (O_i, \vec{x}_i, \vec{y}_i, \vec{z}_i)$ , puis on détermine  $T_{i-1}^i$  matrices de passage du repères  $R_{i-1} = (O_{i-1}, \vec{x}_{i-1}, \vec{y}_{i-1}, \vec{z}_{i-1})$  aux repères  $R_{i-1}$ .

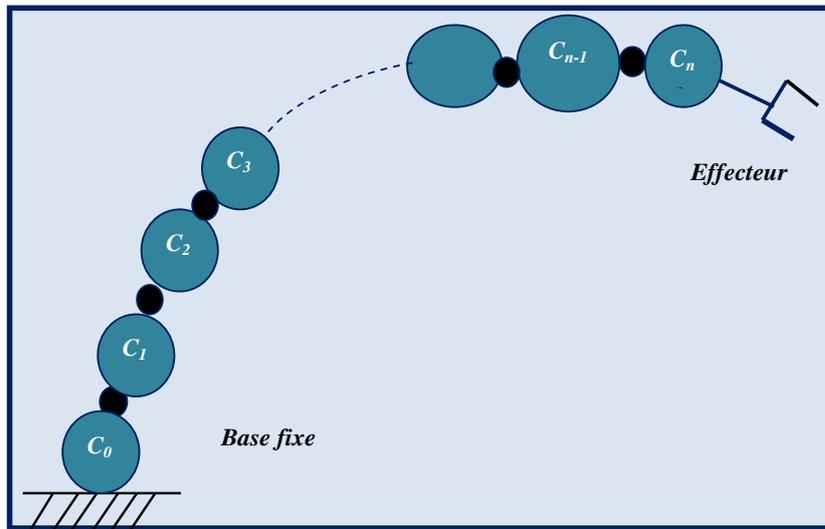


Fig. II.1 structure simple ouverte [16]

### ➤ Principes de la méthode de convention de Denavit-Hartenberg (DH)

Chaque segment peut être décrit à l'aide de quatre paramètres. Deux paramètres sont utilisés pour définir la forme du segment ( $a$  et  $\alpha$ ) et deux autres paramètres décrivent sa position ou son orientation par rapport au précédent dans la chaîne ( $d$  et  $\theta$ ).

Habituellement, pour une articulation rotoïde,  $\theta$  est le paramètre variable et les autres paramètres sont constants.

Les quatre paramètres sont définis en respectant les conventions suivantes:

- $\vec{z}_{i-1}$  désigne l'axe de la liaison entre les segments  $S_{i-1}$  et  $S_i$  son sens est arbitraire.
- $\vec{x}_i$  est la perpendiculaire commune aux axes  $\vec{z}_{i-1}$  et  $\vec{z}_i$ , son sens est tel qu'il coupe d'abord l'axe  $\vec{z}_{i-1}$  puis  $\vec{z}_i$ .
- $d_i$  est l'écart entre origines successives suivant l'axe d'articulation :

$$d_i = \overrightarrow{O_{i-1}O_i} \cdot \vec{z}_{i-1} \quad (\text{II.2}).$$

- $\theta_i$  est l'angle entre les axes  $\vec{x}_{i-1}$  et  $\vec{x}_i$  défini suivant  $\vec{z}_{i-1}$  et variable dans le cas d'une articulation rotoïde.

- $a_i$  est la distance entre origines successives suivant  $\vec{x}_i$  :

$$\vec{a}_i = \overrightarrow{O_{i-1}O_i} \quad (II.3)$$

- $\alpha_i$  est l'angle entre les axes  $\vec{z}_{i-1}$  et  $\vec{z}_i$  défini suivant  $\vec{x}_i$ . [17]

### ➤ Inconvénient de la méthode DH

Dédié aux chaînes simples, ne permet pas se rendre en compte des boucles et des arborescences dans les chaînes cinématiques.

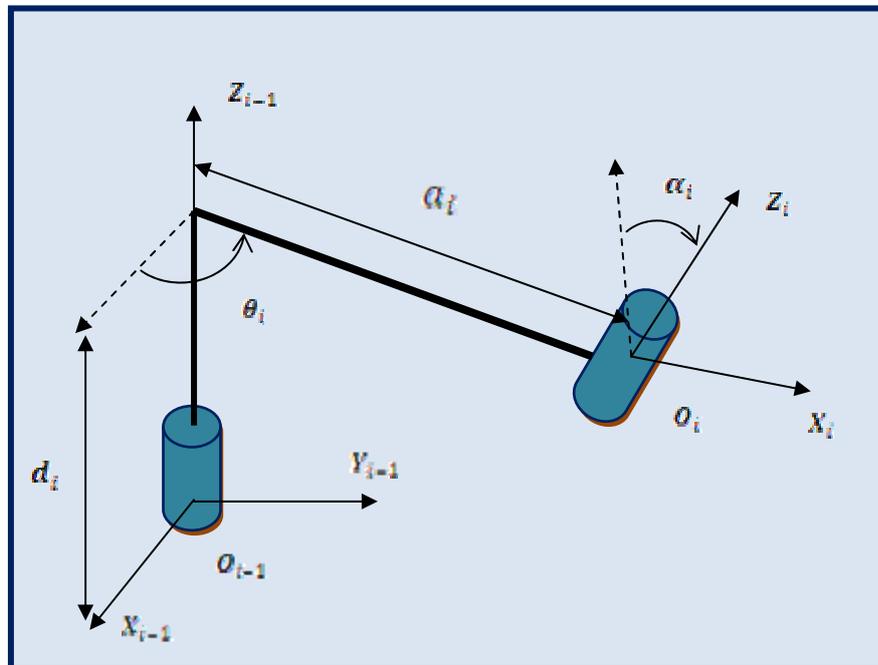


Fig. II.2 structure de solides successifs dans une chaîne

En utilisant cette convention, l'opérateur homogène élémentaire permet de passer du repère  $R_{i-1}$  au repère  $R_i$ , (voir figure II.2).

### II. 2. 1. b. Matrice de passage

La matrice permettant le passage du repère  $R_{i-1}$  au repère  $R_i$  s'écrit comme suit :

$$T_{i-1}^i = \begin{bmatrix} \cos \theta_i & -\sin \theta_i \cdot \cos \alpha_i & \sin \theta_i \cdot \sin \alpha_i & a_i \cdot \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cdot \cos \alpha_i & -\cos \theta_i \cdot \sin \alpha_i & a_i \cdot \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{II.4})$$

On peut obtenir la matrice de passage de l'effecteur à base fixe en multipliant les matrices élémentaires de translation et de rotation, d'où l'expression II.5 [17].

$$T_0^n = T_0^1 \cdot T_1^2 \cdots T_{n-1}^n \quad (\text{II.5})$$

## II. 2. 2. Modèle géométrique inverse (MGI)

### 1) Description du MGI

Le modèle géométrique inverse permet de trouver l'ensemble des configurations articulaires possibles pour une position et une orientation données de l'effecteur ou de la plateforme.

Pour les manipulateurs sériels, la détermination du modèle géométrique inverse (MGI) est un problème complexe. On doit inverser un système d'équations non linéaires ce qui n'est pas trivial.

Toutefois et selon la structure du manipulateur étudié, il existe des méthodes permettant de résoudre le MGI sous une forme explicite. Par exemple pour la plupart des manipulateurs utilisés dans l'industrie, la méthode de Paul [Paul 81]. permet de donner les solutions du MGI sous forme explicite.

Dans le cas général, la résolution du MGI pour un manipulateur sériel consiste à résoudre un polynôme, appelé le polynôme du MGI. Les coefficients de ce polynôme dépendent des paramètres de DH du manipulateur étudié ainsi que de la position de l'effecteur dans l'espace cartésien.

Pour les manipulateurs parallèles, la résolution du MGI ne pose généralement pas de problème. Pour calculer le MGI, nous écrivons un système d'équations non linéaires dont chaque équation est associée à une jambe du manipulateur. Chaque jambe est caractérisée par une origine  $O_i$  et une extrémité  $C_i$ . La configuration  $\mathbf{X}$  de la plateforme permet de définir la position des points extrêmes de chaque jambe. Nous pouvons ainsi écrire le MGI de chaque jambe :  $O_i C_i = f^{-1}(X)$ . Ce modèle est parfois difficile à résoudre, notamment pour les manipulateurs spatiaux et lorsque la structure de leurs jambes est complexe [15].

$$\begin{aligned} f^{-1} : M &\rightarrow N \\ X &\rightarrow \theta = f_{MGD}^{-1}(X) \end{aligned} \quad (\text{II.6})$$

## 2) Problèmes du MGI

### a. Problème de géométrie

Le choix d'un point  $C$  (l'extrémité de l'OT) situant en dehors du volume atteignable du robot, dans ce cas le système d'équation n'a pas de solution.

### b. Problème mécanique

l'hypothèse sur les rotations, on supposant qu'elles soient de  $360^0$  à l'opposé de la réalité à cause des contraintes liées à la structure mécanique. Ceci implique que le volume atteignable sera réduit.

### c. Problème mathématique

On suppose que le point  $C(x_c, y_c, z_c)$  est choisi dans le volume atteignable avec  $x_c, y_c, z_c$  des données connues, dans ce cas on possède deux situations éventuelles :

- ✓ Le nombre d'inconnues > nombre d'équations qu'on a.
- ✓ La simplification du problème en négligeant une des inconnues (on met sa valeur à 0), alors le problème peut avoir :

Le modèle mathématique admet

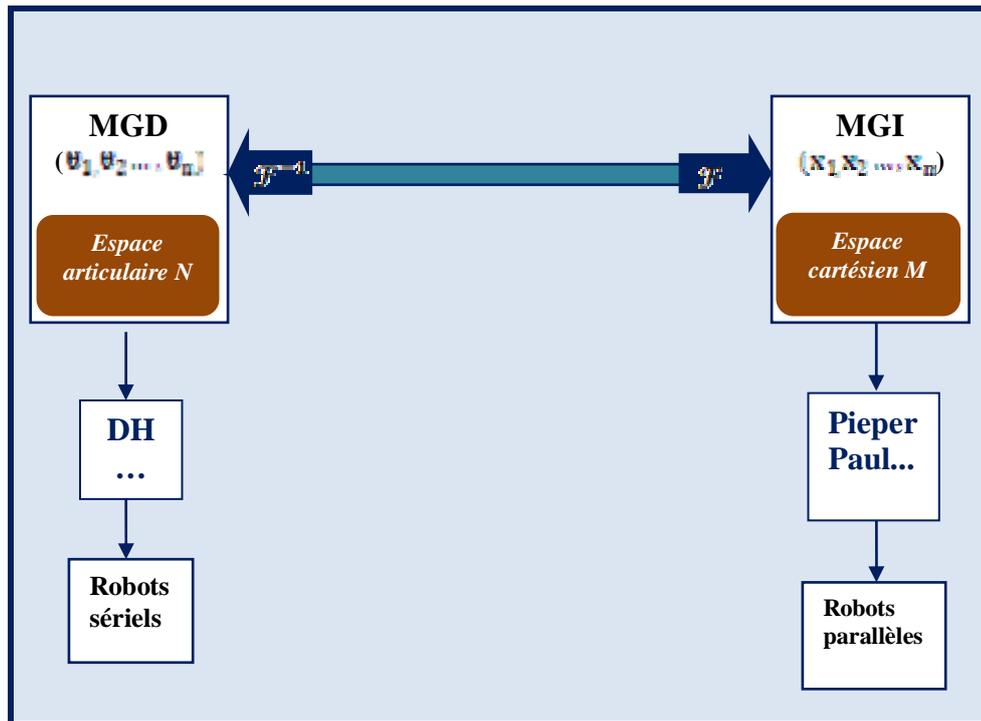
- Un nombre fini de solutions
- Un nombre limité de solutions
- Une seule solution [3].

## 3) Passage d'un MGD vers un MGI

Pour résumer, le schéma de la figure II.3 est proposé :

- MGD est la situation de l'OT du robot (sériel) en fonction de sa configuration.

C'est un ensemble de relation en utilisant l'une des méthodes citées précédemment. L'inversion des relations permet de déduire le MGI (système d'équations non linéaire) dont la résolution est la détermination des coordonnées articulaires.

Fig. II.3 Passage entre MG ( $D \Leftrightarrow I$ )

- Le MGI est beaucoup plus adaptés pour la modélisation des robots parallèles.

## II. 3. Le modèle cinématique

### II. 3. 1. Le modèle cinématique directe

Le modèle cinématique complète le modèle géométrique, en écrivant les relations entre les vitesses des variables articulaires et les vitesses de l'organe terminal. La propriété intéressante du modèle cinématique est sa linéarité par rapport aux vitesses. Ainsi il est plus simple à manipuler que le modèle géométrique et permet d'utiliser des propriétés qui découlent de la résolution des systèmes linéaires.

Pour un robot sériel le modèle cinématique est simple à trouver. La matrice jacobienne est obtenue en écrivant les torseurs cinématiques correspondants aux liaisons entre les différentes pièces du robot.

Soit  $\vec{e}$  le vecteur unitaire de l'axe de la liaison entre les solides  $S_{i-1}$  et  $S_i$  (voir figure II.4).

$$\overline{\sigma}_i = 1 - \sigma_i \quad (\text{II.7})$$

La rotation du vecteur vitesse lors du mouvement de  $S_i / S_{i-1}$  :

$$\vec{\omega}_{i/i-1} = \dot{\theta}_i \cdot \vec{u}_i \quad (\text{II.8})$$

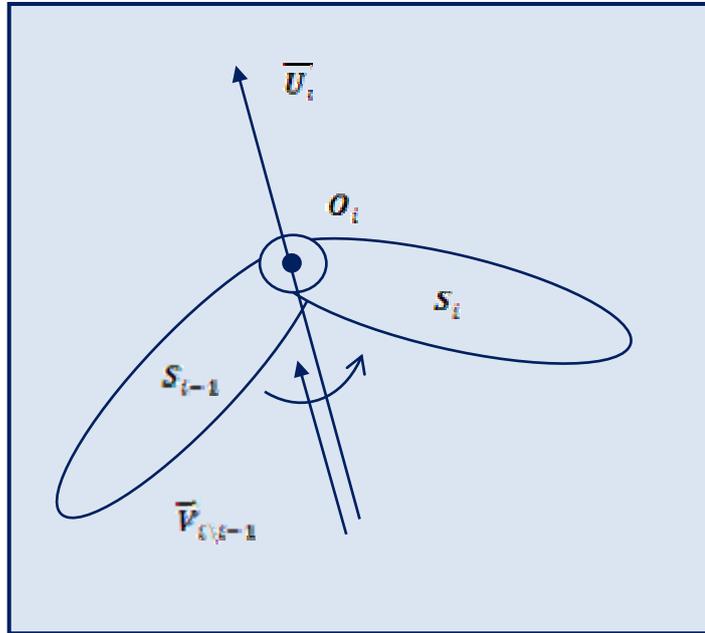


Fig. II.4 Vecteurs vitesse pour une liaison de deux solides

Soit une translation du vecteur vitesse :

$$\vec{v}_{0_{i/i-1}} = \dot{\theta}_i \cdot \vec{u}_i \quad (\text{II.9})$$

Le torseur cinématique du mouvement de  $S_i \setminus S_{i-1}$  exprimé en un point P quelconque, s'écrit :

$$C(P \in S_i / S_{i-1}) = \begin{cases} \vec{\omega}_{i/i-1} = \overline{\sigma}_i \cdot \dot{\theta}_i \cdot \vec{u}_i \\ \vec{v}_{P \in S_i / S_{i-1}} = v_{0_{i/i-1}} + \vec{\omega}_{i/i-1} \wedge \overline{O_i P} = \sigma_i \cdot \dot{\theta}_i \cdot \vec{u}_i + \overline{\sigma}_i \cdot \dot{\theta}_i \cdot \vec{u}_i \end{cases} \quad (\text{II.10})$$

$$\vec{u}_{ip} = \vec{u}_i \wedge \overline{O_i P} \quad (\text{II.11})$$

Le modèle cinématique des robots sériels s'écrit sous la forme suivante :

$$MCD : X = J \cdot \theta \quad (\text{II.12})$$

J : le Jacobien du manipulateur [18].

**Exemple :**

Robot sériel plan (2 Rotations R et une translation P de la figure II.5).

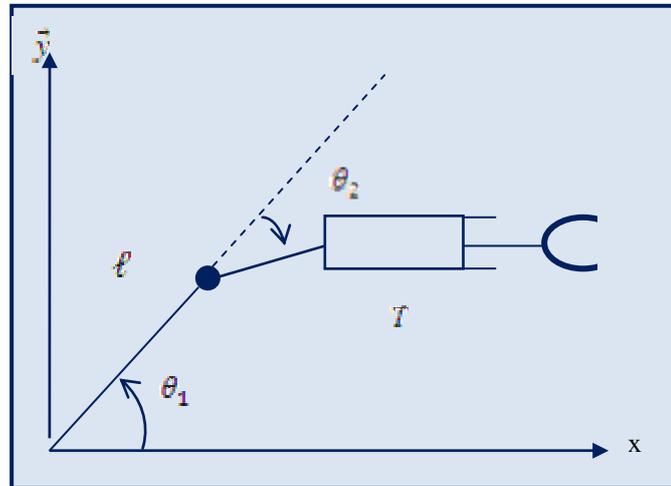


Fig. II.5 Robot plan 2R1P

Le modèle cinématique direct est donné par l'expression (II.13) suivante :

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} = \begin{bmatrix} -l \cdot \sin \theta_1 - T \sin(\theta_1 + \theta_2) & -T \cdot \sin(\theta_1 + \theta_2) & \cos(\theta_1 + \theta_2) \\ l \cdot \cos \theta_1 + T \cos(\theta_1 + \theta_2) & T \cdot \cos(\theta_1 + \theta_2) & \sin(\theta_1 + \theta_2) \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{T} \end{bmatrix} \quad (\text{II.13})$$

### II. 3. 2. Modèle cinématique inverse

Afin de suivre le contour à une vitesse constante, ou toute vitesse prescrite, on doit connaître la relation qui existe entre la vitesse de l'outil et les vitesses des différentes articulations (MCD).

La détermination des vitesses articulaires à partir de la vitesse de l'organe terminal est conceptuellement simple étant donné que la relation de vitesse est linéaire. Ainsi les vitesses articulaires sont trouvées via le *Jacobien Inverse*  $J^{-1}$ , comme suit [19] :

$$MCI : \dot{\theta} = j^{-1} \cdot \dot{X} \quad (\text{II.14})$$

## II. 4. Modele dynamique

### II. 4. 1. Modele dynamique directe

Le modèle dynamique est la relation entre les efforts appliqués aux actionneurs et les positions, les vitesses et les accélérations articulaires. Les expressions des modèles dynamiques sont utilisées pour des applications telles que la simulation, le dimensionnement des actionneurs, l'identification des paramètres inertiels ou la commande.

Defferentes méthodologies couramment rencontrées :

- Anaitique(formalisme d'*Alembert*,formalise de *lagrange*).
- Réccursive(formalisme d'*Euler*).

Le modele dynamique par application de formalisme de lagrange est donnés par l'equation(II.15) suivante :

$$\text{MDD} : f(\theta, \dot{\theta}, \ddot{\theta}, F_d) = 0 \quad (\text{II.15})$$

Avec :

- $\theta$  le vecteur de coordonnées articulaires
- $\dot{\theta}$  vecteur des vitesses articulaires
- $\ddot{\theta}$  vecteur des accélérations articulaires
- $F_d$  le vecteur des efforts appliqués sur le système [21].

### II. 4. 2. Modèle dynamique inverse

L'établissement du modele dynamique inverse d'un robot manipulateur basé sur le formalisme de *lagrange* est particulierement interessant,ce modèle est donnés par la relation suivante :

Où :

$$\frac{d}{d} \left( \frac{\partial T_{S_i}}{\partial \theta_i} \right) - \left( \frac{\partial T_{S_i}}{\partial \theta_i} \right) = G_i + T_i \quad (\text{II.16})$$

$\theta_i$  : variable articulaire

$G_i$  : Force de gravité relative à  $\theta_i$

$T_i$  : Force articulaire  $i$

$T_{S_i}$  : Énergie cinétique du solide  $S_i$

Le modèle dynamique pour l'intégralité du système ( $n$  solides) est exprimé par la relation suivante [20].

$$\text{MDI} : \sum_{i=1}^n \frac{d}{dt} \left( \frac{\partial T_{S_i}}{\partial \dot{\theta}_i} \right) - \left( \frac{\partial T_{S_i}}{\partial \theta_i} \right) = G_i + T_i \quad (\text{II.17})$$

## II. 5. Modèle statique

### II. 5. 1. Modèle statique indirecte (MSI)

Le modèle statique inverse nous permet de calculer les efforts articulaires en fonction des efforts opérationnels. Le MSI est représenté par une relation d'équilibre pour chaque solide de la chaîne cinématique, et exprimé par la relation suivante:

$$\text{MSI} : \ddot{\theta} = J_i^T . F \quad (\text{II.18})$$

Avec :

$J_i^T$ , est une matrice ( $n \times m$ ).

$n$ , la dimension de l'espace articulaire.

$m$ , la dimension de l'espace opérationnel.

### II. 5. 2. Le modèle statique directe (MSD)

On peut déduire le MSD, on inversant la relation (II.18) précédente et on obtient l'équation (II.19) suivante :

$$\text{MSD} : F = J_i^{T-1} . \ddot{\theta} \quad (\text{II.19})$$

Où  $J_i^{T-1}$  matrice ( $n \times m$ ) cas d'un robot sériel et non redondant i.e.  $n = m$  .[22]

## **II. 6. Conclusion**

Ce chapitre nous a permis d'introduire les notations utilisées fréquemment en robotique. Nous avons abordé la définition des modèles géométriques, cinématiques, dynamiques et statiques, directs et indirects. Nous disposons donc d'un ensemble de modèles mathématiques qui peuvent être mise en œuvre après avoir procédé à une identification des paramètres géométriques inertiels.



# Chapitre 03

## Planification de la Trajectoire d'un Bras Manipulateur

**Chapitre III****Planification de la Trajectoire d'un Bras Manipulateur****III. 1. Introduction**

L'effecteur de l'organe terminal est requis pour se déplacer d'une manière particulière afin d'accomplir les tâches qui lui sont affectées. Dans le chapitre précédent on a décrit les modèles mathématiques (géométrique, cinématique,...etc.) régissant le comportement des manipulateurs.

Les manipulateurs doivent suivre des chemins pré-planifiés pour l'exécution des tâches spécifiés. Le plus grand problème de planification de trajectoire est la commande de mouvement des manipulateurs.

L'objectif de la planification de trajectoire est de décrire le mouvement comme étant une séquence temporelle de la situation de l'articulation, segment et l'effecteur et ces dérivations générées par l'interpolation ou l'approximation du chemin souhaité par une fonction polynomiale. Ce chapitre est dédié pour décrire les terminologies impliquées dans la planification de trajectoire, les techniques associées pour cela ainsi que la description du problème dans l'espace articulé et l'espace cartésien.

L'un des objectifs d'un algorithme de planification de trajectoire est d'impliquer une fonction continue assurant la continuité de ces dérivées.

**III. 2. Définitions**

Les termes impliqués dans la planification de trajectoire sont :

**III. 2. 1. Chemin (path)**

Un chemin représente le lieu de points à traverser par le manipulateur pour la réalisation de la tâche spécifiée.

### **III. 2. 2. Trajectoire**

Séquence temporelle de position, de vitesse et d'accélération pour chacune des articulations ou de l'effecteur du manipulateur. La trajectoire est une représentation spatiale et temporelle du mouvement.

### **III. 2. 3. Les via points (Knot points)**

Appelés aussi points d'interpolation. Ensemble de points intermédiaires sur la trajectoire qui se situent entre la position initiale et la position finale et sur lesquels il passe le manipulateur.

### **III. 2. 4. Planification de la trajectoire dans l'espace articulaire**

Dans l'espace articulaire, chaque point du chemin est spécifiés en termes de position et d'orientation de l'effecteur souhaité relativement à la base du manipulateur. Chaque point est converti en un ensemble de positions articulaires souhaitées on appliquant la cinématique inverse. La fonction continue est donc trouvée pour chaque articulation on passant sur ces points.

### **III. 2. 5. Planification de la trajectoire dans l'espace cartésien**

Dans ce cas, le chemin est spécifié explicitement dans l'espace cartésien. Les contraintes du chemin (vitesse, accélération, etc.) sont spécifiées par des coordonnées cartésiennes. Les moteurs au niveau des articulations sont asservis en coordonnées articulaires par la trajectoire spécifiée.

### **III. 2. 6. Génération de trajectoire**

C'est le fait de calculer la trajectoire en tant qu'une séquence temporelle des valeurs en temps réel, en utilisant un algorithme de planification de trajectoire basé sur des contraintes spatiales et temporelles. [23]

## **III. 3. Les étapes de planification de trajectoire**

Trois étapes pour résoudre un problème de planification de trajectoire :

### **III. 3. 1. Description de la tâche**

La détermination du type de mouvement requis est la première étape de planification de mouvement. Cette spécification représente l'entrée de l'algorithme de planification de trajectoire. Trois types de tâches sont distingués :

**a) Mouvement point à point (cas des applications de types pick and place)**

La tâche est spécifiée à l'aide de la situation initiale et la situation finale. Dans ce cas la continuité est la seule contrainte sur le mouvement. Pas de spécifications sur les situations intermédiaires. Le planificateur est libre pour formuler n'importe quel chemin convenable.

**b) Mouvement à chemin continu entre la position initiale et la position finale**

Un chemin spécifique est requis pour être tracé par l'organe terminal dans l'espace cartésien. Le soudage à l'arc est l'une des applications de ce type. L'utilisateur spécifie le type et les paramètres.

**c) Description de la tâche**

Chemin à plus de deux points spécifiés. Ce type est utilisé pour une meilleure surveillance lors de l'exécution de la trajectoire. L'opération *pick and place* en présence d'obstacles est un exemple typique de cette description.

**III. 3. 2.Sélection et utilisation d'une technique de planification de trajectoire**

Il existe deux catégories de techniques :

- Techniques utilisées dans l'espace articulaire.
- Techniques utilisées dans l'espace cartésien.

Dans le cas d'un mouvement point à point (sans ou avec points intermédiaires) la technique articulaire est utilisée. La planification du mouvement se fait au niveau de l'articulation. Le système articulaire génère une fonction qui dépend du temps et ces deux premières dérivées pour chacune des variables articulaires afin de décrire le mouvement du manipulateur.

La technique cartésienne est utilisée pour les applications à chemin continu. Cette technique permet de fournir un historique temporel pour la position, la vitesse et l'accélération de l'effecteur par rapport à la base. Les variables articulaires et les dérivées sont calculés à l'aide du modèle cinématique inverse.

### III. 3. 3 Calcul de la trajectoire

Le calcul de la séquence du temps des valeurs atteintes par l'effecteur générées à partir d'une technique de planification de trajectoire. Ces valeurs sont calculées aux mises à jour de chemin particulier ou bien par échantillonnage par fréquence.

### III. 4. Obtention du modèle cinématique directe pour un mécanisme 2R

On considère le manipulateur 2R suivant (voir figure III.1) :

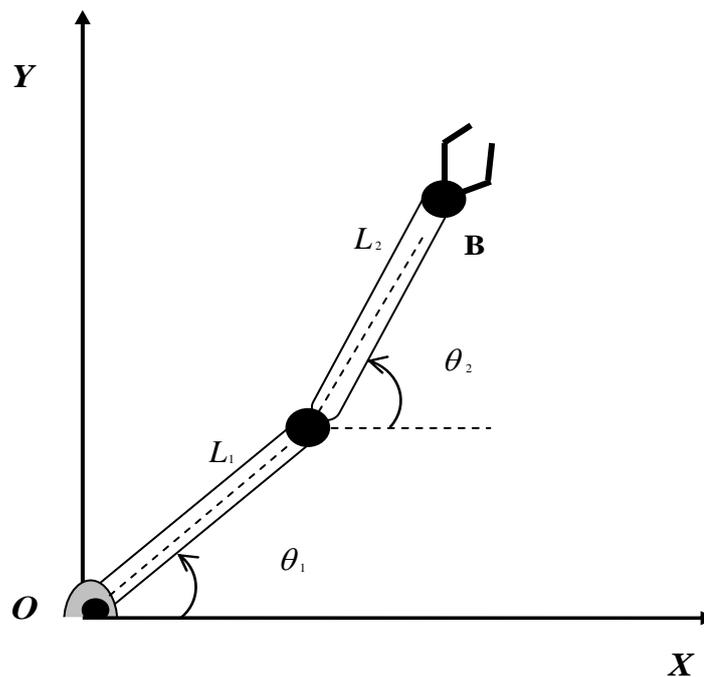


Fig. III.1 Manipulateur 2R simple

Ce manipulateur comporte les éléments suivants :

- Une base O.
- Deux segments de longueurs  $L_1$  et  $L_2$  .
- Deux articulations (joints) réalisant des rotations autour de ses axes.
- Le point B est l'effecteur de ce manipulateur.

Le déplacement du point B est schématisé par le mécanisme de la figure III.2 :

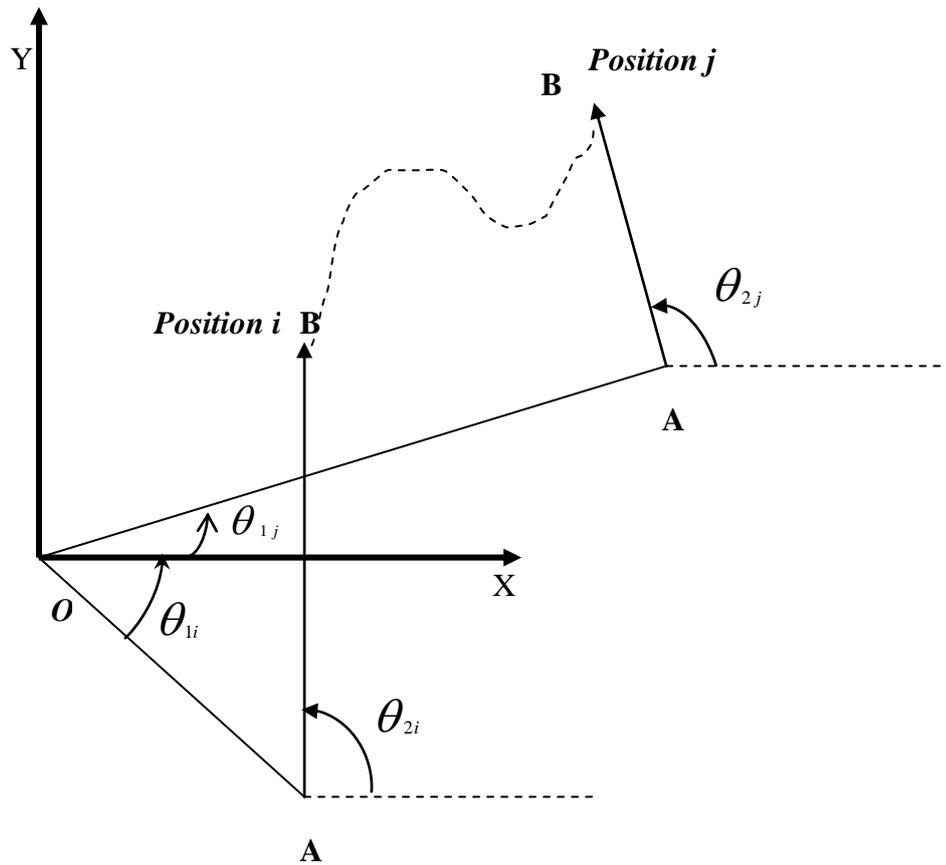


Fig. III.2 Déplacement du point B de l'effecteur

Les valeurs initiales du mouvement sont :  $\theta_{1i}, \theta_{2i}, \theta_{1j}, \theta_{2j}$

Les caractéristiques du déplacement du point B sont données par le tableau III.1 :

<i>Position</i>	<i>Vitesse</i>	<i>Accélération</i>
$x_B(0)$	$\dot{x}_B(0)$	$\ddot{x}_B(0)$
$y_B(0)$	$\dot{y}_B(0)$	$\ddot{y}_B(0)$
$x_B(tf)$	$\dot{x}_B(tf)$	$\ddot{x}_B(tf)$
$x_B(tf)$	$\dot{y}_B = (tf)$	$\ddot{y}_B = (tf)$

Tableau III.1 Position, vitesse et accélération du mouvement du point B

Le modèle mathématique est obtenu en appliquant le modèle cinématique directe pour trouver les coordonnées cartésiennes et le modèle cinématique inverse pour la détermination des coordonnées articulaires, ce que montrent les équations suivantes :

- Equations de position :

$$\begin{cases} x_B = L_1 \cos \theta_1 + L_2 \cos \theta_2 \\ y_B = L_1 \sin \theta_1 + L_2 \sin \theta_2 \end{cases} \quad (\text{III.1})$$

Ces équations sont appelées également équations algébriques linéaires (LAE)

- Equations de vitesse :

$$\begin{cases} \dot{x}_B = -l_1 \dot{\theta}_1 \sin \theta_1 - l_2 \dot{\theta}_2 \sin \theta_2 \\ \dot{y}_B = l_1 \dot{\theta}_1 \cos \theta_1 + l_2 \dot{\theta}_2 \cos \theta_2 \end{cases} \quad (\text{III.2})$$

$$\begin{bmatrix} \dot{x}_B \\ \dot{y}_B \end{bmatrix} = \begin{bmatrix} -l_1 \sin \theta_1 & -l_2 \sin \theta_2 \\ l_1 \cos \theta_1 & l_2 \cos \theta_2 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} \quad (\text{III.3})$$

Les équations de vitesse ont la forme générale suivante :

$$[\dot{X}] = [C][\dot{\theta}] \quad (\text{III.4})$$

On multiplie par  $[C]^{-1}$

$$[C]^{-1}[\dot{X}] = [C]^{-1}[C][\dot{\theta}] \quad (\text{III.5})$$

On obtient l'équation III.6 suivante:

$$[\dot{\theta}] = [C]^{-1}[\dot{X}] \quad (\text{III.6})$$

L'équation III.6 est une équation algébrique non linéaire, elle permet de trouver les valeurs des variables articulaires en connaissance des valeurs des variables cartésiennes.

### III. 5. Techniques de planification de trajectoire

#### III. 5. 1. Les techniques articulaires pour la planification de trajectoire

La planification de la trajectoire d'un manipulateur nécessite la détermination des valeurs des variables articulaires à partir de la position de l'organe terminal spécifiée par l'utilisateur :

**Etape 1 :** L'obtention de l'ensemble des valeurs correspondantes aux variables articulaires pour chaque point du chemin spécifié on appliquant l'algorithme cinématique inverse.

**Etape 2 :** Trouver une fonction continue pour chaque articulation.

Les caractéristiques exigées par un algorithme de planification de trajectoire dans l'espace articulaire sont :

- Une contrainte temporelle importante est que le temps de déplacement entre deux chemins quelconques est le même pour chaque articulation pour que toutes les articulations atteignent simultanément les positions de leurs chemins correspondants.
- Pour garantir un mouvement doux, la continuité des positions, vitesses et accélérations est nécessaires aux niveaux des articulations.
- La fonction d'interpolation ne doit pas avoir des calculs intensifs.
- Réduction des trajectoires non continues et les effets indésirables.

Pour déplacer l'effecteur d'une position initiale vers une position finale dans une certaine durée de temps, il existe des fonctions innombrables satisfaisant les critères précités comme le montre la figure III.3, le Choix du polynôme approprié est une tâche cruciale pour la planification de trajectoire dans l'espace articulaire. Pour choisir un polynôme, il existe deux approches :

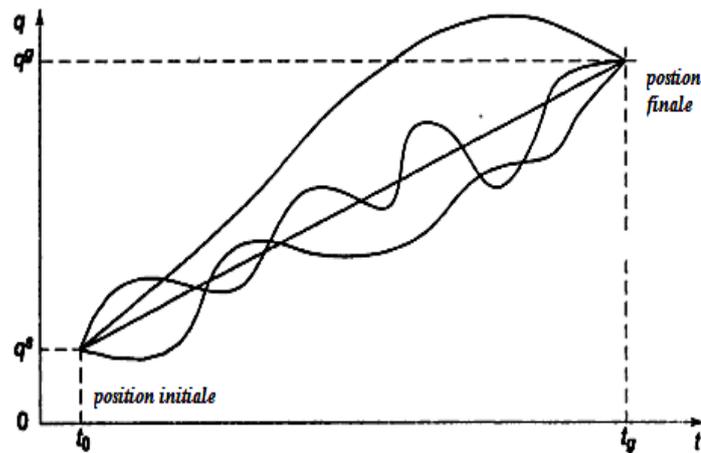


Fig. III.3 Fonction continue pour le mouvement vers une position finale

- Choix d'un polynôme ayant le degré le plus élevé avec au moins il existe autant de coefficients que les contraintes spécifiées.
- Diviser la trajectoire en segments et puis utiliser le polynôme de moindre degré pour l'interpolation de chaque segment avec une transition continue d'un segment à un autre.

### III. 5. 1. a. Planification de trajectoire par la technique polynomiale

Elle consiste en un mouvement d'une position initiale vers une position finale en passant par des points intermédiaires pendant une durée de temps spécifiée sous certaines contraintes. On suppose qu'on dispose d'une seule liaison rotoïde  $\theta(t)$ , dans ce cas une variable locale du temps  $\tau$  est introduite et les contraintes du mouvement sont :

#### 1. Utilisation du polynôme de degré trois

La trajectoire basée sur un polynôme de degré trois est caractérisé par des profils continus pour la position et la vitesse alors que l'accélération est en général discontinue.

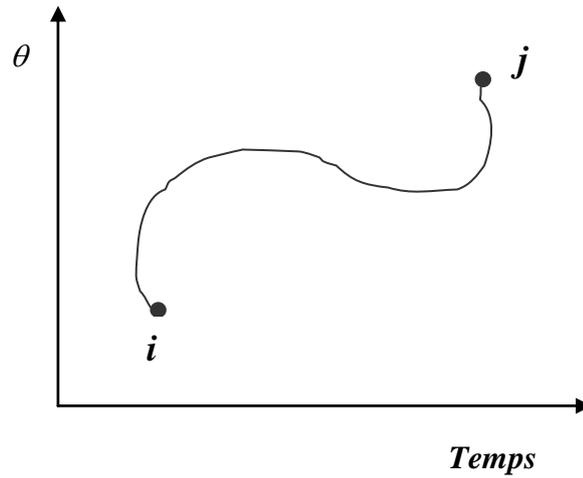
##### 1.a. Cas de deux points spécifiés

i : Position initiale

j : Position finale

Position i :  $\theta_i, \dot{\theta}_i, \ddot{\theta}_i$

Position j :  $\theta_j, \dot{\theta}_j, \ddot{\theta}_j$



$T_{ij}$  : Le temps de déplacement de  $i$  vers  $j$

$\tau \rightarrow [0 : T_{ij}]$

$\theta(\tau)$  : Polynôme dont le degré dépend du nombre de contraintes.

Si,  $\theta_i, \dot{\theta}_i, \theta_j, \dot{\theta}_j$  sont spécifiées, alors :

$$\theta(t) = c_0 + c_1 \tau + c_2 \tau^2 + c_3 \tau^3 \quad (\text{III.7})$$

$$\text{Au point } i : \theta(0) = \theta_i, \dot{\theta}(0) = \dot{\theta}_i \quad (\text{III.8})$$

$$\text{Au point } j : \theta(T_{ij}) = \theta_j, \dot{\theta}(T_{ij}) = \dot{\theta}_j \quad (\text{III.9})$$

Au point  $i$ , ( $\tau = 0$ )

Le coefficient  $c_0$  est donné par :

$$\theta_i = c_0 \quad (\text{III.10})$$

La vitesse est obtenue en dérivant le polynôme décrivant la position  $\theta(\tau)$

$$\dot{\theta}(\tau) = c_1 + 2c_2 \tau + 3c_3 \tau^2 \quad (\text{III.11})$$

Le coefficient  $c_1$  est donné par :

$$\dot{\theta}_i = c_1 \quad (\text{III.12})$$

La position au point  $j$  est :

$$\theta_j = \theta(T_{ij}) = c_0 + c_1 T_{ij} + c_2 T_{ij}^2 + c_3 T_{ij}^3 \quad (\text{III.13})$$

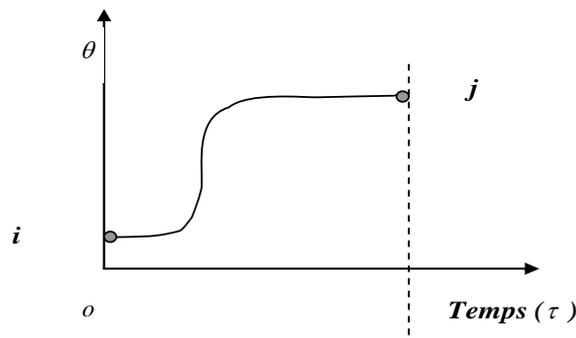
Et la vitesse au même point est :

$$\dot{\theta}_j = \dot{\theta}(T_{ij}) = c_1 + 2c_2 T_{ij} + 3c_3 T_{ij}^2 \quad (\text{III.14})$$

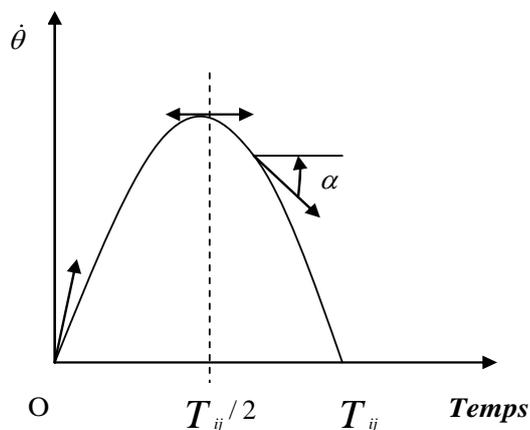
#### • Ajustement de la courbe cubique

Pour se déplacer d'une position initiale vers une position finale en passant par  $n$  points intermédiaires, l'utilisateur doit spécifier  $n + 2$  positions et vitesses et il doit également fournir des données constantes cinématiquement importantes.

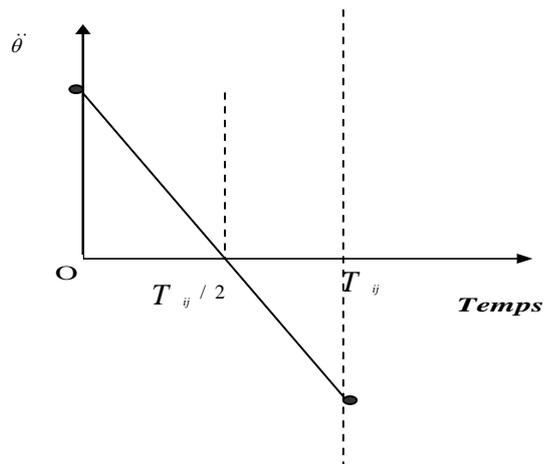
##### ➤ La courbe de la position :



##### ➤ La courbe de la vitesse :

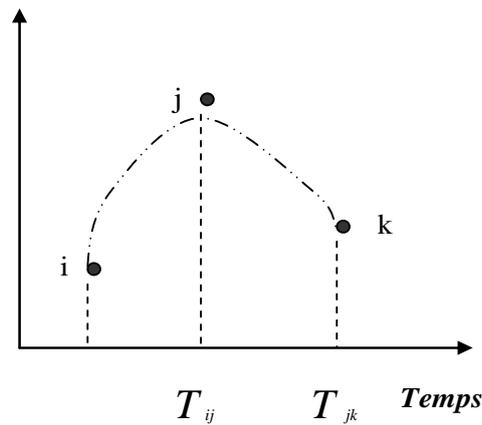


➤ La courbe de l'accélération :



**1.b. Cas de trois points spécifiés (i, j et k)**

Les données qu'on doit connaître sont la position et la vitesse au niveau des points i et k, et au point j la position est suffisante car la vitesse doit être choisie par le système. On considère deux segments cubiques comme l'indique la courbe suivante :



Le segment  $i \rightarrow j$  :

Le polynôme décrivant la position est :

$$\theta(\tau) = c_0 + c_1 \tau + c_2 \tau^2 + c_3 \tau^3$$

$$\tau : 0 \rightarrow T_{ij}$$

$$\theta(0) = \theta_i$$

(III.15)

$$\dot{\theta}(0) = \dot{\theta}_i \quad (\text{III.16})$$

$$\theta(T_{ij}) = \theta_j \quad (\text{III.17})$$

Le segment  $j \rightarrow k$  :

$$\theta(\tau) = b_0 + b_1\tau + b_2\tau^2 + b_3\tau^3$$

$$\tau : 0 \rightarrow T_{jk}$$

$$\theta(0) = \theta_j \quad (\text{III.18})$$

$$\theta(T_{jk}) = \theta_k \quad (\text{III.19})$$

$$\dot{\theta}(T_{jk}) = \dot{\theta}_k \quad (\text{III.20})$$

Au point j, on désire une continuité de vitesse et accélération comme suit :

$$\dot{\theta}_{ij} \Big|_{T_{ij}} = \dot{\theta}_{jk} \Big|_0$$

$$\ddot{\theta}_{ij} \Big|_{T_{ij}} = \ddot{\theta}_{jk} \Big|_0$$

Segment  $ij$

$$\theta(\tau) = c_0 + c_1\tau + c_2\tau^2 + c_3\tau^3$$

$$\dot{\theta}(\tau) = c_1 + 2c_2\tau + 3c_3\tau^2$$

$$\ddot{\theta}(\tau) = 2c_2 + 6c_3\tau$$

Segment  $jk$

$$\theta(\tau) = b_0 + b_1\tau + b_2\tau^2 + b_3\tau^3$$

$$\dot{\theta}(\tau) = b_1 + 2b_2\tau + 3b_3\tau^2$$

$$\ddot{\theta}(\tau) = 2b_2 + 6b_3\tau$$

La continuité de la vitesse au point  $j$

$$c_1 + 2c_2T_{ij} + 3c_3T_{ij}^2 = b_1 \quad (\text{III.21})$$

La continuité de l'accélération :

$$2c_2 + 6c_3T_{ij} = 2b_2 \quad (\text{III.22})$$

Donc on a 8 équations avec 8 coefficients :  $c_0, c_1, c_2, c_3, b_0, b_1, b_2, b_3$

### 1.c. Cas général

En général, pour aller d'une position initiale vers une position finale via  $n$  points intermédiaires on a :

- L'ajustement de  $n + 1$  segments cubiques.
- Chaque segment a 4 coefficients.
- $4(n + 1)$  coefficients à déterminer.
- $4(n + 1)$  équations pour à résoudre.

Conditions initiales (position/vitesse) : 2 équations

Conditions finales : (position/vitesse) : 2 équations

Au niveau de chaque point intermédiaire : (condition de la position) : 2 équations

Continuité de vitesse : 1 équation

Continuité d'accélération : 1 équation

Donc chaque point intermédiaire nécessite 4 équations, et pour  $n$  points intermédiaires on a  $4 * n$  équations.

Le nombre total d'équations est  $4n + 2 + 2 = 4(n + 1)$ .

Tous les coefficients peuvent être déterminés.

♦ *Pourquoi le polynôme de degré trois est utilisé?*

♦ **Motivation** :

- Le polynôme de troisième degré est le polynôme de degré minimal assurant la continuité de vitesse et l'accélération.
- Facile à manipuler.

## 2. Utilisation d'un polynôme de degré cinq

Bien que la trajectoire basée sur un polynôme de degré trois est suffisamment continue, la discontinuité de l'accélération peut générer des effets indésirable dans certaines applications sur les chaînes cinématiques et les charges inertielles. Afin d'obtenir une trajectoire avec une accélération continue en dehors des conditions sur la position et la vitesse, il est également nécessaire d'assigner des valeurs d'accélération initiales et finales convenables. Par conséquent, étant donné qu'il y a six conditions aux limites (position, vitesse et accélération), un polynôme de cinquième degré est adopté. [25]

Toujours le manipulateur réalise un mouvement d'un point  $i$  vers un point  $j$ , les six conditions du mouvement sont les suivantes :

$$\begin{cases} \theta_i, \dot{\theta}_i, \ddot{\theta}_i \\ \theta_j, \dot{\theta}_j, \ddot{\theta}_j \end{cases}$$

Le polynôme décrivant la position est donné par l'expression suivante :

$$\theta(\tau) = c_0 + c_1 \tau + c_2 \tau^2 + c_3 \tau^3 + c_4 \tau^4 + c_5 \tau^5$$

**Exemple** : *l'application pick and place*

Les techniques de planification de trajectoire ont été développés pour les opérations pick and place dans lesquelles un manipulateur est nécessaire pour déplacer un objet d'une position à une autre. Le chemin intermédiaire n'est pas défini. Dans ces techniques deux

points de passages sont fournies : configuration pick et configuration place, sont spécifiées dans l'espace de la tâche et ont des conditions de vitesse nulles. Les coordonnées des points de passage sont généralement converties en coordonnées articulaires afin de permettre la planification de trajectoire dans l'espace articulaire pour réduire les complexités de l'algorithme de calcul. [26]

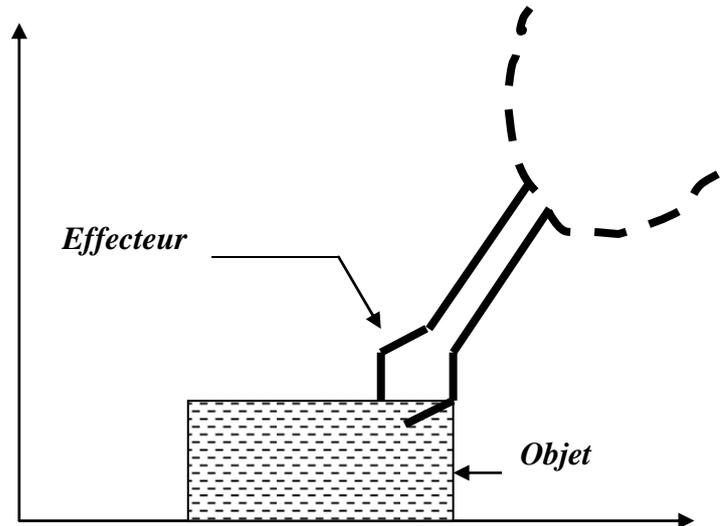


Fig. III.4 Application pick and place

Les différentes étapes :

- Position initiale  $i$
- Position de décollage  $l$
- Déposer  $s$
- Position finale  $f$

### 3. Application de la stratégie 4-3-4 trajectory

$i - l$  : Polynôme de quatrième degré

$l - s$  : Polynôme de troisième degré (polynôme cubique)

$s - f$  : Polynôme de quatrième degré

Prise en compte toujours d'une variable locale de temps, et les temps pour les différents segments sont notés par :

$$i - l : T_{il}$$

$$l - s : T_{ls}$$

$$s - f : T_{sf}$$

Les positions des segments sont

Le segment  $i - l$  :

$$\theta_{il} = a_0 + a_1\tau + a_2\tau^2 + a_3\tau^3 + a_4\tau^4, 5 \text{ coefficients à déterminer}$$

Le segment  $l - s$  :

$$\theta_{ls} = b_0 + b_1\tau + b_2\tau^2 + b_3\tau^3 + b_4\tau^4, 4 \text{ coefficients à déterminer}$$

Le segment  $s - f$  :

$$\theta_{sf} = c_0 + c_1\tau + c_2\tau^2 + c_3\tau^3 + c_4\tau^4, 5 \text{ coefficients à déterminer}$$

Le nombre d'équations est :  $5 + 4 + 5 = 14$  équations

Les équations au point  $i$  :

$$\text{L'équation 1 : } \theta_{il}(0) = \theta_i \quad (\text{III.23})$$

$$\text{L'équation 2 : } \dot{\theta}_{il}(0) = \dot{\theta}_i \quad (\text{III.24})$$

$$\text{L'équation 3 : } \ddot{\theta}_{il}(0) = \ddot{\theta}_i \quad (\text{III.25})$$

Les équations au point  $f$  :

$$\text{L'équation 4 : } \theta_{sf}(T_{sf}) = \theta_f \quad (\text{III.26})$$

$$\text{L'équation 5 : } \dot{\theta}_{sf}(T_{sf}) = \dot{\theta}_f \quad (\text{III.27})$$

$$\text{L'équation 6 : } \ddot{\theta}_{sf}(T_{sf}) = \ddot{\theta}_f \quad (\text{III.28})$$

Les équations au point  $l$  :

$$\text{L'équation 7 : } \theta_{il}(T_{il}) = \theta_l \quad (\text{III.29})$$

$$\text{L'équation 8 : } \theta_{ls}(0) = \theta_l \quad (\text{III.30})$$

$$\text{L'équation 9 : La continuité de vitesse : } \dot{\theta}_{il}(T_{il}) = \dot{\theta}_{ls}(0) \quad (\text{III.31})$$

$$\text{L'équation 10 : La continuité d'accélération : } \ddot{\theta}_{il}(T_{il}) = \ddot{\theta}_{ls}(0) \quad (\text{III.32})$$

Les équations au point  $s$  :

$$\text{L'équation 11 : } \theta_{ls}(T_{ls}) = \theta_s \quad (\text{III.33})$$

$$\text{L'équation 12 : } \theta_{sf}(0) = \theta_s \quad (\text{III.34})$$

$$\text{L'équation 13 : La continuité de vitesse : } \dot{\theta}_{ls}(T_{ls}) = \dot{\theta}_{sf}(0) \quad (\text{III.35})$$

$$\text{L'équation 14 : La continuité d'accélération : } \ddot{\theta}_{ls}(T_{ls}) = \ddot{\theta}_{sf}(0) \quad (\text{III.36})$$

14 coefficients à déterminer pour une trajectoire 4-3-4

### Résolution du problème appliqué au mouvement du robot :

La figure III.5 représente un robot manipulateur à deux segments connectés à l'aide de deux joints (base et coude). Il existe un moteur au niveau de chaque liaison et les angles articulaires sont  $\theta_1$  et  $\theta_2$ . Les coordonnées de l'effecteur du manipulateur sont données par l'équation III.37 :

$$\begin{cases} x_1 = l_1 \cos \theta_1 + l_2 \cos(\theta_1 + \theta_2) \\ x_2 = l_1 \sin \theta_1 + l_2 \sin(\theta_1 + \theta_2) \end{cases} \quad (\text{III.37})$$

Avec  $l_1$  et  $l_2$  : les longueurs des segments

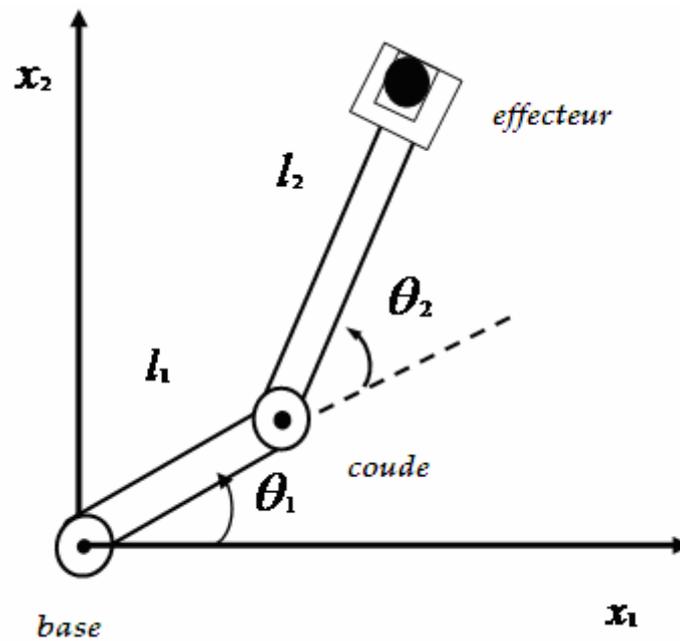


Fig. III.5 Manipulateur à deux segments

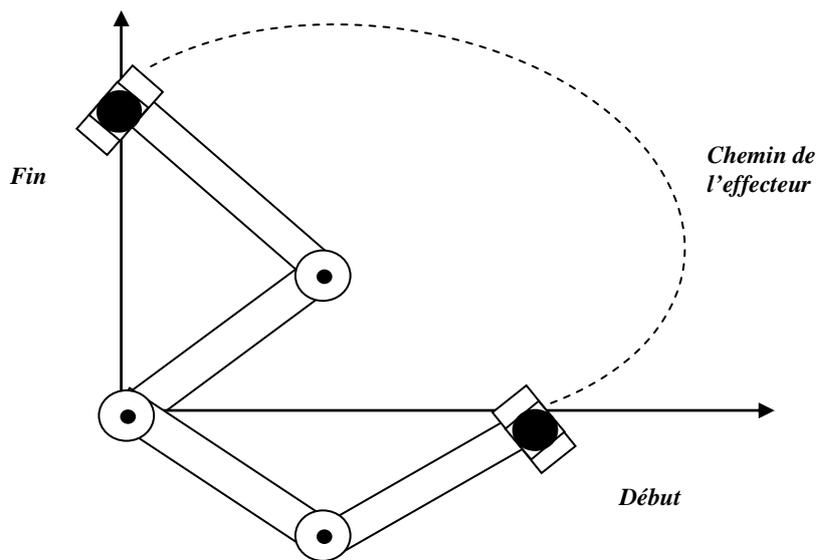


Fig. III.6 Mouvement d'un robot manipulateur

Le problème est la commande des angles articulaires par les moteurs afin de déplacer l'effecteur d'une position à une autre.

Le manipulateur part d'un état de repos d'une position connue et se déplace vers la position désirée. Il commence et termine son mouvement avec une vitesse et une accélération nulles. Le polynôme suivant est utilisé pour contrôler le mouvement en générant les commandes à envoyer aux contrôleurs des moteurs au niveau des articulations :

$$\theta_1(t) = \theta_1(0) + a_1 t^5 + a_2 t^4 + a_3 t^3 + a_4 t^2 + a_5 t \quad (\text{III.38})$$

$$\theta_2(t) = \theta_2(0) + b_1 t^5 + b_2 t^4 + b_3 t^3 + b_4 t^2 + b_5 t \quad (\text{III.39})$$

Ou  $\theta_1(0)$  et  $\theta_2(0)$  : les angles initiales à  $t=0$  et les vecteurs coefficients  $a = [a_1 \ a_2 \ a_3 \ a_4 \ a_5]$  et  $b = [b_1 \ b_2 \ b_3 \ b_4 \ b_5]$  sont à déterminer pour fournir le mouvement adéquat.

Pour des coordonnées initiales connus  $(x_1, x_2)$  à  $t=0$  et des coordonnées finales connues à  $t=t_f$ , les valeurs requises pour les angles  $\theta_1(0), \theta_2(0), \theta_1(t_f)$  et  $\theta_2(t_f)$  peuvent être trouvées en utilisant les lois de la trigonométrie.

Pour des valeurs de  $\theta_1(0), \theta_1(t_f)$  et  $t_f$  données, des équations matricielles sont mises en place et résolues pour le vecteur de coefficients  $a$  et de même pour  $\theta_2(0), \theta_2(t_f)$  et  $t_f$  la mise en place et la résolution des équations matricielles pour le vecteur d'équations  $b$ . Ces résultats sont à être utilisés pour tracer le chemin de l'effecteur.

Les contraintes restantes du mouvement du bras sont des contraintes de vitesse et d'accélération des segments qui sont nulles à la position initiale et finale voulues. Ce qui implique que la vitesse et l'accélération angulaires des deux angles sont également nulles à  $t=0$  et  $t=t_f$ .

La vitesse angulaire du 1<sup>er</sup> segment à un temps  $t$  est la dérivée de l'angle  $\theta_1(t)$  par rapport au temps

$$\dot{\theta}_1(t) = 5a_1 t^4 + 4a_2 t^3 + 3a_3 t^2 + 2a_4 t + a_5 \quad (\text{III.40})$$

La vitesse à  $t=0$  est

$$\dot{\theta}_1(0) = a_5 = 0 \quad (\text{III.41})$$

Et donc le coefficient  $a_5 = 0$ .

L'accélération angulaire est la deuxième dérivée de l'angle  $\theta_1(t)$  par rapport au temps.

$$\ddot{\theta}_1(t) = 20a_1t^3 + 12a_2t^2 + 6a_3t + 2a_4 \quad (\text{III.42})$$

L'accélération à  $t=0$  est :

$$\ddot{\theta}_1(0) = 2a_4 = 0 \quad (\text{III.43})$$

$$\text{Et donc, le coefficient } a_4 = 0 \quad (\text{III.44})$$

Maintenant, on peut écrire les trois contraintes sur  $\theta_1(t)$  et ses dérivées sous forme matricielle à  $t=t_f$  comme suit :

$$\begin{bmatrix} t_f^5 & t_f^4 & t_f^3 \\ 5t_f^4 & 4t_f^3 & 3t_f^2 \\ 20t_f^3 & 12t_f^2 & 6t_f \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} \theta_1(t_f) - \theta_1(0) \\ 0 \\ 0 \end{bmatrix} \quad (\text{III.45})$$

De même, pour  $\theta_2(t)$

$$\text{On a } \dot{\theta}_2(t) = 5b_1t^4 + 4a_2t^3 + 3b_3t^2 + 2b_4t + b_5 \quad (\text{III.46})$$

La vitesse à  $t=0$  est

$$\dot{\theta}_2(0) = b_5 = 0 \quad (\text{III.47})$$

Et donc le coefficient  $b_5 = 0$  .

L'accélération angulaire est la deuxième dérivée de l'angle  $\theta_2(t)$  par rapport au temps.

$$\ddot{\theta}_2(t) = 20b_1t^3 + 12b_2t^2 + 6b_3t + 2b_4 \quad (\text{III.48})$$

L'accélération à  $t=0$  est

$$\ddot{\theta}_2(0) = 2b_4 = 0 \quad (\text{III.49})$$

Et donc, le coefficient  $b_4 = 0$

La forme matricielle des trois contraintes sur  $\theta_2(t)$  et ses dérivées à  $t = t_f$  est la suivante :

$$\begin{bmatrix} t_f^5 & t_f^4 & t_f^3 \\ 5t_f^4 & 4t_f^3 & 3t_f^2 \\ 20t_f^3 & 12t_f^2 & 6t_f \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} = \begin{bmatrix} \theta_2(t_f) - \theta_2(0) \\ 0 \\ 0 \end{bmatrix} \quad (\text{III.50})$$

A noter qu'on a trois équations avec trois inconnus pour chaque angle ainsi que deux dérivées. C'est la raison du choix d'un polynôme de degré cinq pour contrôler le mouvement. Trois termes de faible degré ( $t^2, t^1, t^0$ ) possédant des coefficients nuls afin de satisfaire les contraintes à  $t = 0$ . Ce qui rend les termes de fort degrés ( $t^5, t^4, t^3$ ) avec trois coefficients non nuls. Ajouter des termes supplémentaires de degré élevé exigerait des équations de contraintes supplémentaires pour calculer les valeurs correspondantes des coefficients. [27]

Supposant qu'on dispose les valeurs suivantes :

$$t_f = 2s$$

$$\theta_1 = -19^\circ$$

$$\theta_2 = 44^\circ$$

$$\theta_1(t_f) = 43^\circ$$

$$\theta_2(t_f) = 151^\circ$$

$$L_1 = 4 \text{ cm}$$

$$L_2 = 3 \text{ cm}$$

Ce qui correspond à un emplacement de départ (6.5, 0) et une destination de (0,2),

Les coefficients de mouvement pour  $\theta_1$  sont :

$$\mathbf{a} = \begin{pmatrix} 0.2029 \\ -1.0145 \\ 1.3526 \end{pmatrix}$$

Et les coefficients de mouvement pour  $\theta_2$  sont :

$$\mathbf{b} = \begin{pmatrix} 0.3502 \\ -1.7508 \\ 2.3344 \end{pmatrix}$$

Le chemin tracé par l'effecteur est montré sur la figure III.7 suivante :

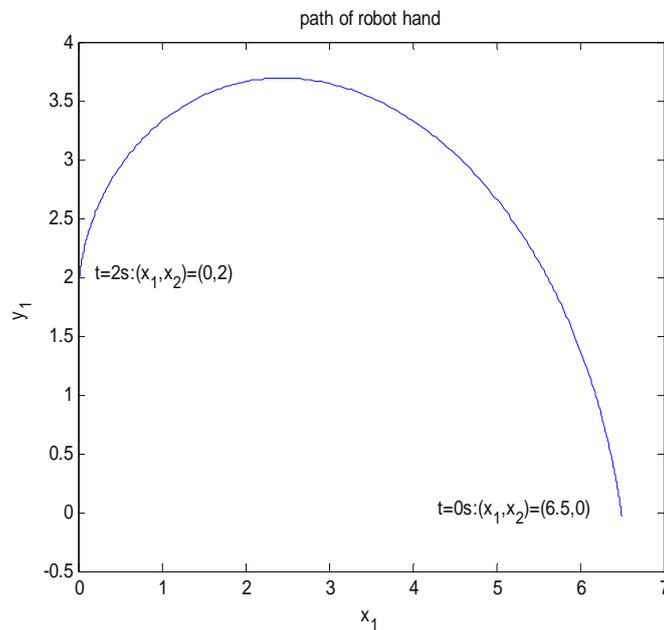


Fig. III.7 Chemin d'un bras manipulateur

### III. 5. 1. b. Segments linéaire combinés avec des paraboles

Dans ce cas la planification de trajectoire consiste à exécuter les articulations avec une vitesse constante entre la position initiale et la position finale, c'est l'équivalent d'un polynôme de premier degré où la vitesse est constante et l'accélération est nulle. Cela signifie que l'accélération doit être infinie au début et à la fin du segment du mouvement afin de créer une vitesse instantanée aux frontières. Pour éviter cette situation, le segment linéaire peut être

combiné avec des sections paraboliques au début et en fin du segment du mouvement, on créant une position et une vitesse continues. Ce qui est montré dans la figure III.8.

On suppose que les segments paraboliques sont combinés symétriquement avec le segment linéaire (la section entre  $t_b$  et  $t_f - t_b$ ) où :

$\theta_i$  : Position initiale

$\theta_f$  : Position finale

$t_i = 0$

$t_f$  : Temps final

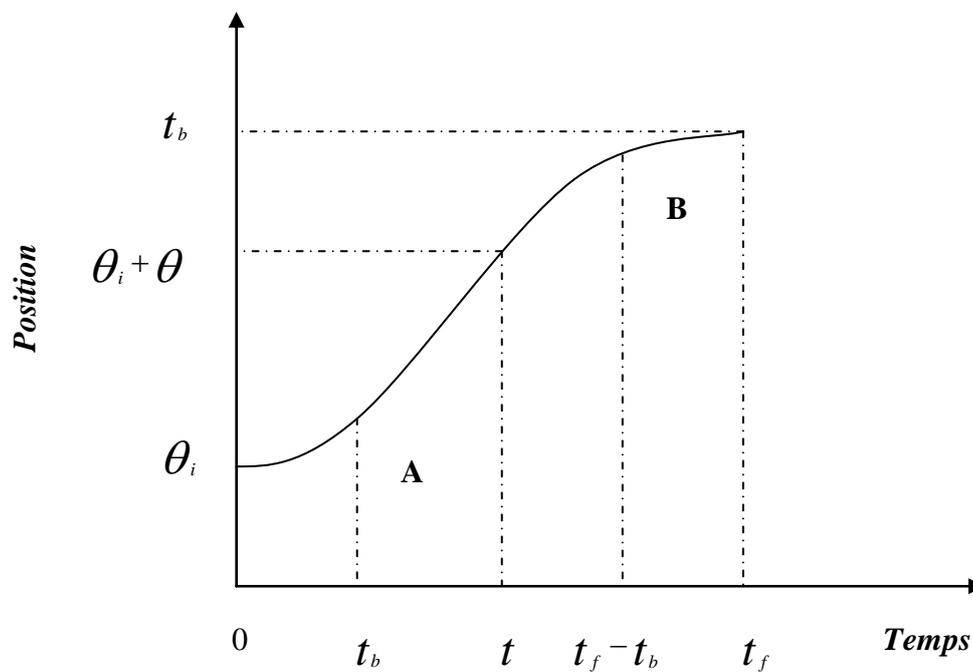


Fig. III.8 Schéma pour segments linéaire combinés avec des sections paraboliques

On peut écrire :

$$\text{La position : } \theta(t) = c_0 + c_1 t + \frac{1}{2} c_2 t^2 \quad (\text{III.51})$$

$$\text{La vitesse : } \dot{\theta}(t) = c_1 + c_2 t \quad (\text{III.52})$$

$$\text{L'accélération : } \ddot{\theta}(t) = c_2 \quad (\text{III.53})$$

Donc, l'accélération est évidemment constante pour la section parabolique et par conséquent une vitesse continue aux points communs (A et B).

On substitue les conditions aux limites dans l'équation du segment parabolique comme suit :

$$\theta(t=0) = \theta_i = c_0 \quad (\text{III.54})$$

$$\dot{\theta}(t=0) = 0 = c_1 \quad (\text{III.55})$$

$$\ddot{\theta}(t) = c_2 \quad (\text{III.56})$$

$$\Rightarrow \begin{cases} c_0 = \theta_i \\ c_1 = 0 \\ c_2 = \ddot{\theta} \end{cases}$$

Le segment parabolique devient :

$$\theta(t) = \theta_i + \frac{1}{2} c_2 t^2 \quad (\text{III.57})$$

$$\dot{\theta}(t) = c_2 t \quad (\text{III.58})$$

$$\ddot{\theta}(t) = c_2 \quad (\text{III.59})$$

Pour le segment linéaire, la vitesse sera constante et peut être choisie en se basant sur les capacités physiques de l'actionneur, la vitesse initiale est nulle. Une vitesse articulaire constante connue  $\omega$  dans la portion linéaire et une vitesse finale nulle dans l'équation III.58. On peut trouver les positions et les vitesses articulaires des points A, B et le point final comme suit :

$$\theta_A = \theta_i + \frac{1}{2} c_2 t_b^2 \quad (\text{III.60})$$

$$\dot{\theta}_A = c_2 t_b = \omega \quad (\text{III.61})$$

$$\theta_B = \theta_A + \omega((t_f - t_b) - t_b) = \omega(t_f - 2t_b) \quad (\text{III.62})$$

$$\dot{\theta}_B = \dot{\theta}_A = \omega \quad (\text{III.63})$$

$$\theta_f = \theta_B + (\theta_A - \theta_i) \quad (\text{III.64})$$

$$\dot{\theta} = 0 \quad (\text{III.65})$$

### III. 5. 1. c. Segments linéaires combinés avec des courbes paraboliques et des points intermédiaires

On suppose qu'il y a plus d'un segment de mouvement à la fin du 1<sup>er</sup> segment, le robot continuera à se déplacer vers le prochain point. L'objectif est de combiner les segments des mouvements l'un avec l'autre afin de prévenir le mouvement arrêt/marche. Dans ce cas la, position du robot est connue à  $t_0$  et l'équation cinématique inverse du robot est utilisée. On peut donc calculer les angles des articulations au niveau des points intermédiaires à la fin du mouvement.

Utilisation des conditions aux limites de chaque point pour calculer les coefficients de la courbe parabolique. Par exemple, au début du mouvement la position et la vitesse doivent être continues. Ces données seront considérées comme des conditions limites pour le point intermédiaire, et par conséquent un nouveau segment peut être calculé. Ce processus se poursuit jusqu'à ce que tous les segments soient calculés et la destination est atteinte. Dans cette approche, il faut s'assurer que des accélérations maximales admissibles ne doivent pas être dépassées.

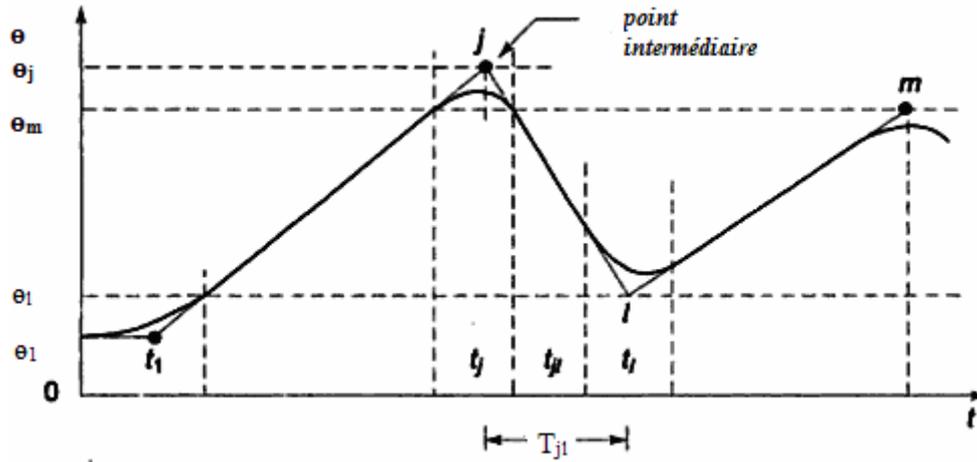


Fig. III.9 Segments linéaires combinés avec des courbes paraboliques et des points intermédiaires (adaptée)

$\theta_j$  : Valeur de la variable articulaire au point  $j$

$T_{il}$  : Temps de déplacement du point  $i$  au point  $l$

$t_{jl}$  : Temps de déplacement entre les points  $j$  et  $l$

$t_j$  : Durée de combinaison autour du point  $j$

### III. 5. 1. d. Trajectoires d'ordre supérieur

Lorsque en plus des points de destination initiaux et finaux, d'autres points sont spécifiés (marche/arrêt), on peut étudier les conditions aux limites des positions, vitesses et accélérations des deux segments au niveau de chaque point pour planifier une trajectoire continue.

L'application des conditions aux limites (initiales et finales), permet d'utiliser un polynôme d'ordre supérieur ayant la forme suivante :

$$\theta(t) = c_0 + c_1 t + c_2 t^2 + \dots + c_{n-1} t^{n-1} + c_n t^n \quad (\text{III.66})$$

La trajectoire passe par tous points spécifiés. Ce processus nécessite des calculs détaillés pour chaque articulation et pour chaque polynôme d'ordre supérieur.

### III. 5. 2. Algorithme de la planification de trajectoire

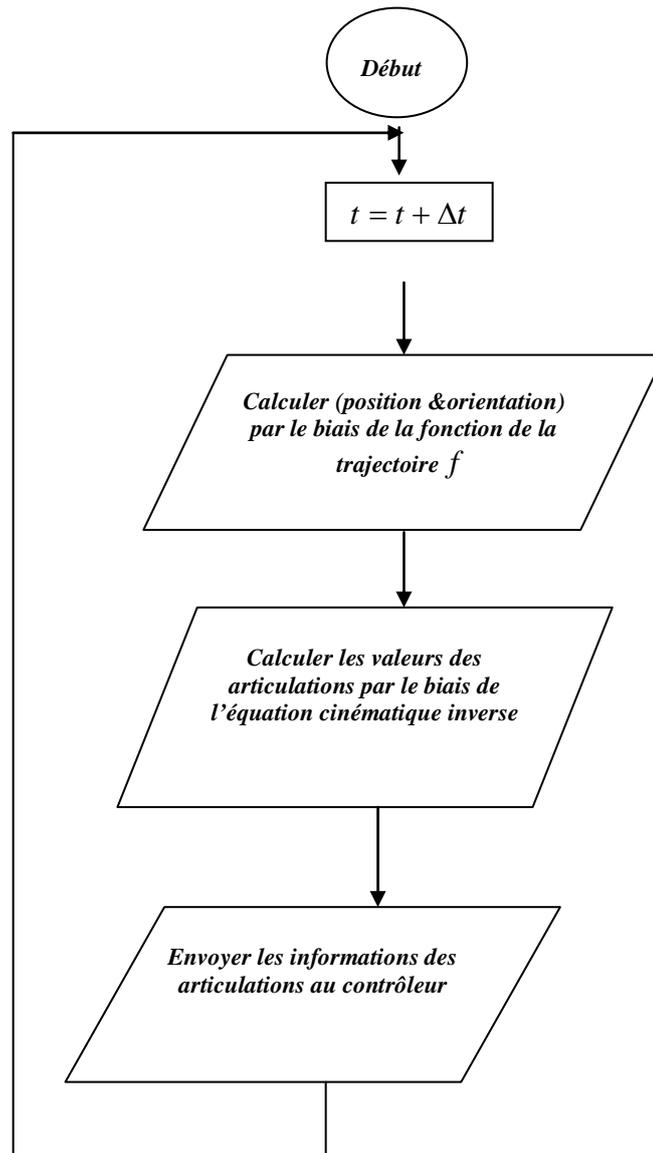


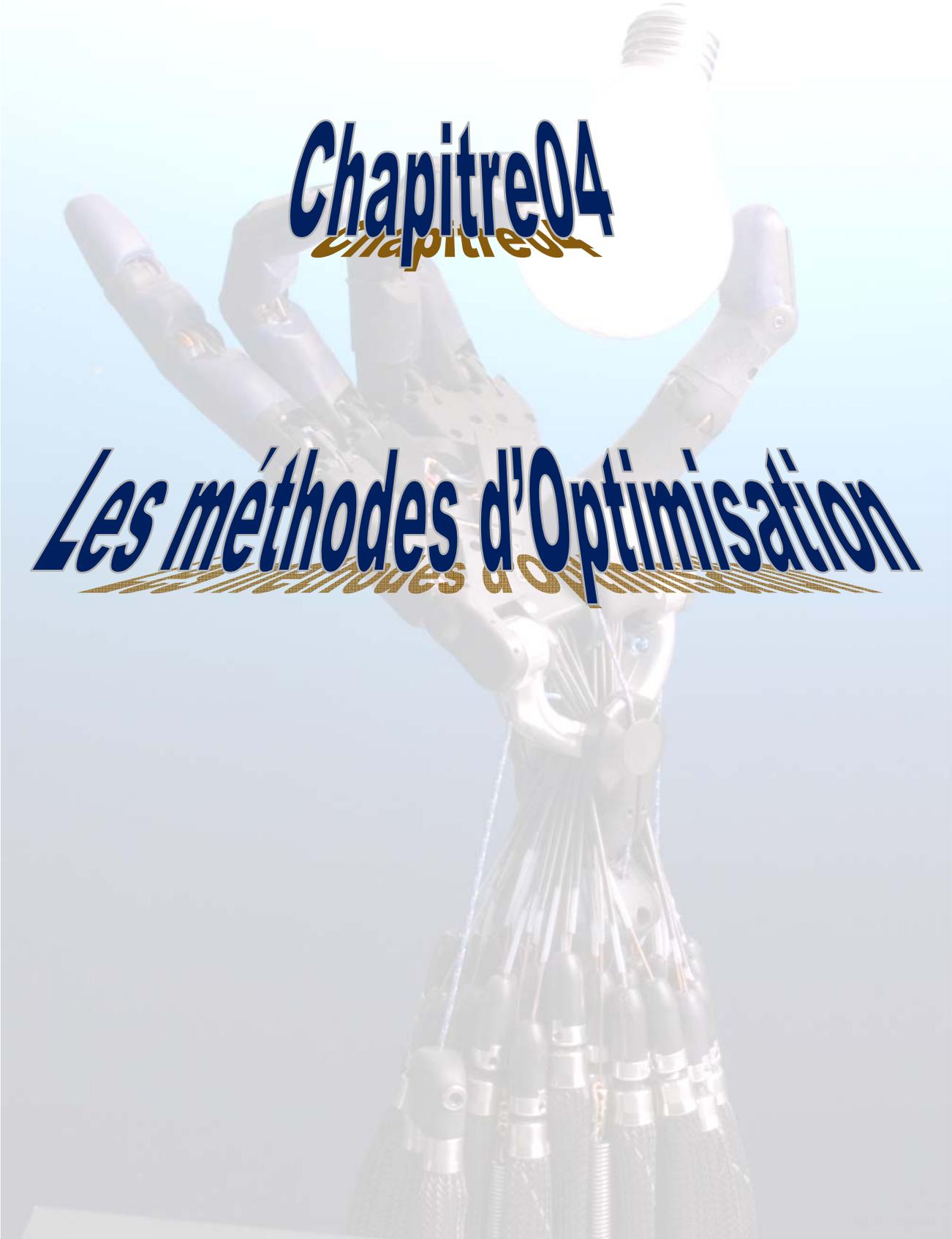
Fig. III.10 Technique cartésienne de planification de trajectoire

### III. 6. Conclusion

Un chemin est une séquence de configurations dans un ordre quelconque on faisant abstraction sur le facteur temps. La trajectoire est une partie de chemin dont le temps est un facteur nécessaire.

La description d'une trajectoire dans l'espace articulaire est une description du mouvement que le robot réalise pendant son déplacement à travers ces valeurs articulaires.

Par contre dans l'espace cartésien le mouvement entre deux points est connu est contrôlé tout le temps. Dans ce chapitre plusieurs techniques ont été proposés pour les deux approches (articulaire/cartésienne), technique polynomiale de différents degrés, techniques utilisant des segments linéaires avec des courbes paraboliques, etc.



**Chapitre04**

***Les méthodes d'Optimisation***

## Chapitre IV

### Les méthodes d'Optimisation

#### IV. 1. Introduction

L'optimisation a connu un essor important ces dernières années avec l'avènement de l'ordinateur. Elle est souvent l'ultime étape de l'analyse numérique où, après avoir étudié un phénomène, l'avoir mis en équation, avoir étudié ces équations et avoir montré que l'on pouvait calculer les solutions avec un ordinateur, on commence à optimiser le système en changeant certains paramètres pour changer la solution dans un sens désiré.

L'optimisation combinatoire occupe une place très importante en recherche opérationnelle, en mathématiques discrètes et en informatique. Son importance se justifie d'une part par la grande difficulté des problèmes d'optimisation et d'autre part par de nombreuses applications pratiques pouvant être formulées sous la forme d'un problème d'optimisation combinatoire.

Généralement, les problèmes d'optimisation combinatoires sont difficiles à résoudre. Etant donnée l'importance de ces problèmes, de nombreuses méthodes de résolution ont été développées en recherche opérationnelle (RO) et en intelligence artificielle (IA). Ces méthodes peuvent être classées essentiellement en deux grandes catégories: les méthodes exactes (complètes) qui garantissent la complétude de la résolution et les méthodes approchées (incomplètes) qui perdent la complétude pour gagner en efficacité. Dans ce chapitre on va vous présenter pour chacune des catégories une méthode avec quelques détails précis, se rapportant à une application très importante et très connue par sa complexité, la TSP.

.

#### IV. 2. Le problème de voyageur de commerce TSP

Un voyageur de commerce désire visiter un certain nombre de villes, débutant et finissant son parcours dans la même ville en visitant chacune des autres villes une et une seule fois. Il désire sélectionner la tournée qui minimise la distance totale parcourue.

Ce problème est connu sous le nom du problème du voyageur de commerce (en anglais *travelling salesman problem* : TSP) et est NP- Complet. [31]

Un des problèmes le plus étudié en recherche opérationnelle est le problème du voyageur de commerce (TSP) appelé aussi «problème du commis voyageur». C'est un problème d'optimisation combinatoire de type NP- difficile très complexe à résoudre, et dont nous disposons de différentes variantes.

Pour résumer, un voyageur de commerce doit visiter  $n$  villes données en passant par chaque ville exactement une fois. Il commence par une ville quelconque et termine en retournant à la ville de départ. Les distances entre les villes étant connues, la question que l'on pose est de savoir quel chemin faut-il choisir pour minimiser la distance totale parcourue. [28]



Fig. IV.1 voyageur de commerce

### Formellement:

Soit un graphe complet non orienté et pondéré :

**L'entrée:**

$$G = (V, E) \text{ Avec } |V| \leq 3$$

Les poids sur les arcs  $e = \{v_i, v_j\}$  sont écrits comme suit :

$$w(e) = W_{i,j} = W_{j,i} , \text{ et tous les poids sont positifs}$$

**La sortie :**

On considère tous les cycles  $C'$  où les sommets sont visités exactement une et une seule fois

$$C' = v_{\pi(0)}, v_{\pi(1)}, \dots, v_{\pi(n-2)}, v_{\pi(n-1)}$$

$$i \neq j : v_{\pi(i)} \neq v_{\pi(j)}$$

Le poids des cycles  $w(C')$  est la somme des poids des arcs parcourus pour visiter le cycle

$$w(C') = \sum_{i=0}^{n-1} w_{\pi(i)}, w_{\pi((i+1)\%n)}$$

La sortie est un cycle  $C$  avec son poids  $w(C)$  où  $C$  est le cycle du poids minimal OPT (optimal). à noter que ce cycle n'est pas nécessairement unique.

$$\forall c' : OPT = w(c) \leq w(c')$$

On peut également écrire la sortie comme un tour  $T$ , une séquence de nœuds :

$$e_i = \{v_{\pi(i)}, v_{\pi((i+1)\%n)}\}$$

$$T = e_0, e_1, \dots, e_{n-2}, e_{n-1}$$

$$w(T) = \sum_{i=0}^{n-1} w(e_i)$$

**La complexité :** le problème de TSP est NP-difficile. [29]

### IV. 3. Les méthodes d'optimisation

Les algorithmes de résolution du TSP peuvent être répartis en deux classes :

#### IV. 3. 1. Les méthodes exactes

Les méthodes exactes cherchent à trouver de manière certaine la solution optimale en examinant de manière explicite ou implicite la totalité de l'espace de recherche. Elles ont l'avantage de garantir la solution optimale néanmoins le temps de calcul nécessaire pour atteindre cette solution peut devenir très excessif en fonction de la taille du problème (explosion combinatoire) et le nombre d'objectifs à optimiser. Ce qui limite l'utilisation de ce type de méthode aux problèmes bi-objectifs de petites tailles. Ces méthodes génériques sont : Branch & Bound, Branch and cut et La programmation dynamique. La méthode de branch and bound sera présentée en détail par la suite. [30]

La complexité en temps des algorithmes exacts proposés croît exponentiellement avec  $n$  (la taille du problème ou le nombre de villes). [31]

### IV. 3. 1. a. Principes de la méthode de branch and bound

Connue aussi sous le nom de méthode de la recherche arborescente ou de la procédure de séparation et d'évaluation, cette méthode consiste en la construction d'un arbre de recherche qui sera exploré de manière à éviter les branches inutiles qui sont des branches contenant des solutions non intéressantes ou carrément non réalisables. L'exploration se fait avec des évaluations des branches et des comparaisons avec une valeur seuil du critère à optimiser. [33]

Branch and bound est une méthode générique de résolution exacte de problèmes d'optimisation, et plus particulièrement d'optimisation combinatoire. Elle utilise l'énumération implicite basée sur la notion de bornes afin d'éviter l'énumération de larges classes de mauvaises solutions. Elle utilise aussi la stratégie diviser pour régner, en se basant sur deux concepts : le branchement (séparation) qui consiste à partitionner ou diviser l'espace des solutions en sous problèmes pour les optimiser chacun individuellement et l'évaluation qui consiste à déterminer l'optimum de l'ensemble des solutions réalisables associé au nœud en question ou, au contraire, de prouver mathématiquement que cet ensemble ne contient pas de solution optimale, la méthode la plus générale consiste à borner le coût des solutions contenues dans l'ensemble. [30]

### IV. 3. 1. b. Application de la méthode branch and bound au problème de TSP

Le problème du voyageur de commerce (TSP) est composé de cinq villes illustré sur la figure IV.2 dont les distances en km sont données par la matrice ci-dessous.

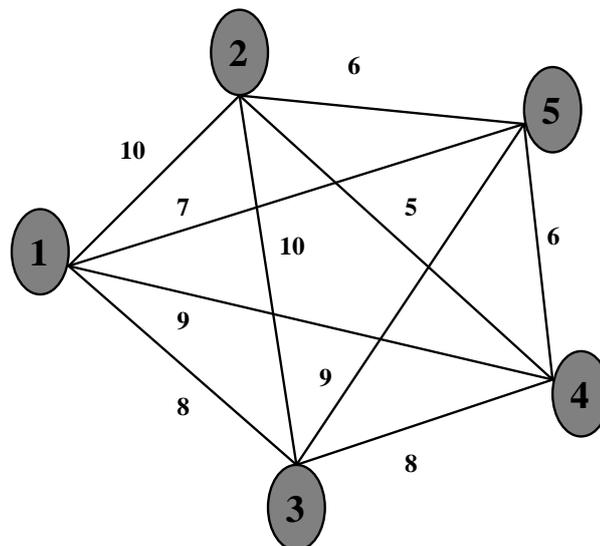


Fig. IV.2 Problème du TSP pour cinq villes.

La matrice initiale est :

	1	2	3	4	5
1	---	10	8	9	7
2	10	---	10	5	6
3	8	10	---	8	9
4	9	5	8	---	6
5	7	6	9	6	---

Fig. IV.3 Matrice initiale

## 1. Principe de la méthode

Commençons tout d'abord, par comprendre la signification de la matrice, autrement dit, que pouvons-nous tirer de cette matrice ?

Si à l'instant initial le marchand se trouve dans la ville 1 (ligne 1), en réalisant son circuit il doit alors quitter la ville 1 en allant vers l'une des quatre autres villes (2, 3, 4, 5). Ainsi le marchand doit parcourir une distance minimale de 7 km afin de quitter la ville 1 dans laquelle il se trouvait. De la même manière, s'il se trouve dans la ville 2, il devrait parcourir une distance minimale de 5 km afin de quitter la ville 2. A partir de la ville numéro 3, il doit parcourir une distance minimale de 8 km afin de la quitter, et ainsi de suite pour le reste des villes. Par conséquent, afin de quitter la ville 4 une distance minimale de 5 km doit être parcourue, et enfin pour quitter la ville 5 une distance minimale de 6 km doit être parcourue.

	1	2	3	4	5	min
1	---	10	8	9	7	7
2	10	---	10	5	6	5
3	8	10	---	8	9	8
4	9	5	8	---	6	5
5	7	6	9	6	---	6

Fig. IV.4 Minimum de lignes RM

La somme des distances minimales doit nécessairement représenter une distance minimale totale que doit parcourir le marchand ambulant. Cette distance minimale totale est obtenue par la somme des minimums de lignes (RM, pour Row Minima):

$$RM = (7+5+8+5+6) = 31 \text{ km}$$

Cette dernière valeur (31 km) représente une valeur minimale que le marchand doit parcourir afin de visiter toutes les villes une et une seule fois. Par conséquent 31 km est une limite (borne) inférieure (LB, pour lower bound) par rapport à la distance réelle que cette personne doit parcourir, et la solution optimale (parcours optimal) devra être supérieure à (ou supérieur ou égale) à la borne inférieure, c.à.d. 31 km. Puisque le problème du TSP en question possède une matrice symétrique, la recherche de minimums via les colonnes n'est pas nécessaire, car on retrouve les mêmes valeurs que celles des lignes.

D'habitude, les matrices des problèmes du voyageur de commerce sont symétriques, parce qu'elles représentent la distance, et la distance satisfait en général la symétrie et l'inégalité triangulaire.

La matrice du problème du voyageur de commerce TSP est en général:

- a. carrée
- b. symétrique
- c. satisfait l'inégalité triangulaire  $d_{ij} + d_{jk} \geq d_{ik}$

En raison de cette inégalité nous pouvons dire que cette personne doit visiter toutes les villes une et une seule fois. S'il se trouve qu'on est dans une situation où le marchand doit revenir à la ville déjà visitée auparavant, cela veut dire que l'inégalité triangulaire est violée, ce qui n'est pas acceptable. Ainsi à chaque fois que la matrice satisfait l'inégalité triangulaire, on peut toujours confirmer que cette personne visitera chacune des villes une et une seule fois. Il peut y avoir des situations où la matrice initiale n'est pas symétrique, dans ce cas la somme des minimums de ligne (RM) est différente de la somme des minimum de colonnes, par conséquent on utilise constamment les minimums de ligne ou de colonnes pour trouver la borne inférieure LB.

## 2. Première étape

Créons tout d'abord quatre branchements soit (n-1) branches, avec  $n = 5$ .  $x_{11} = 1$  n'est pas pris en considération car  $x_{11}$  est un circuit secondaire (subtour) de longueur 1, Par conséquent  $x_{ji}$  ne figurera pas dans la solution.

Quand on fixe la valeur de  $x_{12}$  à 1, cela signifie que nous supposons que le marchand va se déplacer de la ville 1 à la ville 2, par conséquent nous éliminons temporairement la

première ligne et la deuxième colonne de la matrice principale, et cherchons par la suite la distance minimale additionnelle que doit faire le voyageur afin d'accomplir son circuit (tour).

Ayant atteint la ville 2, le voyageur doit quitter cette ville (ville 2) pour visiter le reste des villes. A partir de la matrice restante nous tirons les distances minimales de que devrait donc parcourir et qui sont de : 5 km, 8 km, 6 km et 6 km voir figure ci-dessous.

	1	3	4	5	min
2	10	10	5	6	5
3	8	---	8	9	8
4	9	8	---	6	6
5	7	9	6	---	6

Fig. IV.5 Matrice réduite

En posant  $x_{12} = 1$ , la somme des minimums de ligne (RM), est obtenue en éliminant la première ligne et la deuxième colonne de la matrice principale. Une matrice réduite (4x4) est enfin obtenue, à partir de cette dernière nous calculons les minimums de ligne comme suit:

$$5 + 8 + 6 + 6 + (10 \equiv 1 \rightarrow 2) = 35 \text{ km.}$$

La valeur 35 km représente la borne inférieure du circuit (tour). Cette valeur est obtenue en réalisant le calcul suivant:

En partant de la ville 2 dans laquelle se trouvait, le marchand doit parcourir au minimum une distance de 5 km. De même il doit parcourir au minimum une distance de 8 km pour quitter la ville 3. Pour quitter les villes 4 ou 5 il doit parcourir une distance minimale de 6 km. La somme de toutes ces distances minimales est égale à 35 km est appelée borne inférieure LB. La valeur de  $x_{12} = 1$ , indique que le marchand entame immédiatement son circuit en allant de la ville 1 à la ville 2 et que la distance minimale que doit parcourir ce voyageur est supérieure ou égale à 35 km.

De même, si le marchand commence son circuit à partir de la ville 1 en allant vers la ville 3, c.à.d.  $x_{13} = 1$ , nous éliminons la première ligne et la troisième colonne de la matrice principale, et on calcul par la suite les minimums de lignes RM comme précédemment.

La matrice des minimums de lignes correspondante à ce cas est donnée ci-dessous :

	1	2	4	5	min
2	10	---	5	6	5
3	8	10	8	9	8
4	9	5	---	6	5
5	7	6	6	---	6

Fig. IV.6 Matrice restante cas  $x_{13} = 1$ 

Le calcul des minimums de lignes est obtenu comme suit :

$$RM = 5 + 8 + 5 + 6 + (8 \equiv 1 \rightarrow 3) = 32 \text{ km.}$$

Ainsi si le marchand va de ville 1 à 3, nous trouvons que la distance minimale totale que doit parcourir est de 32 km. Ainsi pour chaque couple de distance  $ij$  donné, on élimine la  $i^{\text{ième}}$  ligne et la  $j^{\text{ième}}$  colonne de la matrice principale, et on calcul après les minimums de lignes à partir de la matrice réduite tout en lui ajoutant la valeur de la distance fixe correspondant à l'arc (1-3) afin d'obtenir toute la distance qui doit être parcourue.

Maintenant on fixe la première ligne et la quatrième colonne ( $x_{14} = 1$ ), 9km correspond à l'intersection de la ligne 1 et la colonne quatre,

	1	2	3	5	min
2	10	---	10	6	6
3	8	10	---	9	8
4	9	5	8	6	5
5	7	6	9	---	6

Fig. IV.7 Matrice réduite cas  $x_{14} = 1$ 

Le minimum de lignes est donné ci-dessous:

$$RM = 6 + 8 + 5 + 6 + (9 \equiv 1 \rightarrow 4) = 34 \text{ km.}$$

Lorsqu'on fixe la première ligne et la cinquième colonne c.à.d.  $x_{15} = 1$ , on trouve la matrice réduite suivante :

	1	2	3	4	min
2	10	---	10	5	5
3	8	10	---	8	8
4	9	5	8	---	5
5	7	6	9	6	6

Fig. IV.8 Matrice réduite cas  $x_{15} = 1$ 

On trouve que la borne inférieure est égale à :

$$RM = 5 + 8 + 5 + 6 + (7 \equiv 1 \rightarrow 5) = 31 \text{ km.}$$

Résumons les étapes précédentes comme suit :

- si le marchand se déplace de la ville 1 à la ville 2, il aura besoin de 35 km ou plus pour parcourir toute la distance.
- si le marchand se déplace de la ville 1 à la ville 3, il aura besoin de 32 km ou plus pour parcourir toute la distance.
- si le marchand se déplace de la ville 1 à la ville 4, il aura besoin de 34 km ou plus pour parcourir toute la distance.
- si le marchand se déplace de la ville 1 à la ville 5, il aura besoin de 32 km ou plus pour parcourir toute la distance.

Puisque nous voulons réduire au minimum toute la distance du circuit à parcourir, nous continuons le branchement à partir de la valeur 31 parce qu'elle représente la valeur minimale ainsi trois branches supplémentaires doivent être créées.

### 3. Deuxième étape

L'arc  $x_{22}$  sera éliminé du circuit parce qu'il représente un circuit secondaire. On n'aura pas aussi le pas (2-5) car le pas (1-5) est déjà dans le circuit, par conséquent le marchand ne voyagera pas via le chemin de l'arc (2-5).

Considérons maintenant les pas (1-5) et (2-1), que doit prendre immédiatement le marchand, par conséquent on élimine instantanément les lignes 1 et 2, et les colonnes 1 et 5

Ceci veut dire qu'à partir de l'arbre 2-1-5, je ne dois pas avoir de pas 5-2 sinon on tombe sur un circuit secondaire, par conséquent l'intersection de la ligne 5 et la colonne 2 doit être remplacée temporairement par (---) et on calcul par la suite les minimums de lignes.

	2	3	4	min
3	10	---	8	8
4	5	8	---	5
5	---	9	6	6

Fig. IV.9 Matrice réduite cas (1-5) et (2-1)

Le calcul des minimums de lignes est donné ci-dessous :

$$RM = 8 + 5 + 6 + (10 \equiv 2 \rightarrow 1) + (7 \equiv 1 \rightarrow 5) = 36 \text{ km.}$$

Considérons le pas initial 1-5 et 2-3. Puisque le pas 2-3 est présent dans le circuit, l'intersection 3-2 sera substituée par (---) pour éviter de se retrouver avec un circuit secondaire.

	1	2	4	min
3	8	---	8	8
4	9	5	---	5
5	7	6	6	6

Fig. IV.10 Matrice réduite cas (1-5) et (2-3)

Le calcul des minimums de lignes est donné ci-dessous :

$$RM = 8 + 5 + 6 + (10 \equiv 2 \rightarrow 3) + (7 \equiv 1 \rightarrow 5) = 36 \text{ km}$$

Considérons le chemin initial composé des pas 1-5 et 2-4. Pour éviter de se retrouver avec un circuit secondaire, l'intersection 4-2 sera substituée par (---).

	1	2	3	min
3	8	10	---	8
4	9	---	8	8
5	7	6	9	6

Fig. IV.11 Matrice réduite cas (1-5) et (2-4)

Le calcul des minimums de lignes est donné ci-dessous :

$$RM = 8 + 8 + 6 + (7 \equiv 1 \rightarrow 5) + (5 \equiv 2 \rightarrow 4) = 34 \text{ km}$$

Maintenant, choisissons parmi tous les nœuds, celui dont la valeur est minimale, soit le nœud dont la valeur est (32) et on continue le branchement à partir de celui-ci.

Considérons maintenant les pas 1-3 et 2-1. La matrice réduite est obtenue ci-dessous. Puisque le chemin est composé des pas 2-1-3, le pas 3-2 constitue un circuit secondaire, donc à exclure.

	2	4	5	min
3	---	8	9	8
4	5	---	6	5
5	6	6	---	6

Fig. IV.12 Matrice réduite cas (1-3) et (2-1)

Le calcul des minimums de lignes est donné ci-dessous :

$$RM = 8 + 5 + 6 + (8 \equiv 1 \rightarrow 3) + (10 \equiv 2 \rightarrow 1) = 37 \text{ km.}$$

Considérons maintenant les pas 1-3 et 2-4. Donc le pas 4-2 est un circuit secondaire qu'il faut éliminer. La matrice réduite est montrée ci-dessous.

	1	2	5	min
3	8	10	9	8
4	9	---	6	6
5	7	6	---	6

Fig. IV.13 Matrice réduite cas (1-3) et (2-4)

Le calcul des minimums de lignes est donné ci-dessous :

$$RM = 8 + 6 + 6 + (8 \equiv 1 \rightarrow 3) + (5 \equiv 2 \rightarrow 4) = 33 \text{ km.}$$

Considérons les pas 1-3 et 2-5. Le pas 5-2 est un circuit secondaire. Par analogie aux précédents cas la matrice réduite est donnée comme suit :

	1	2	4	min
3	8	10	8	8
4	9	5	---	5
5	7	---	6	6

Fig. IV.14 Matrice réduite cas (1-3) et (2-5)

Le calcul des minimums de lignes est donné ci-dessous :

$$RM = 8 + 5 + 6 + (8 \equiv 1 \rightarrow 3) + (6 \equiv 5 \rightarrow 2) = 33 \text{ km.}$$

#### 4. Troisième étape

Considérons le circuit composé des pas 1-3, 2-4 et 3-2. La matrice réduite est donc la suivante :

	1	5	min
4	9	6	6
5	7	...	7

Fig. IV.15 Matrice réduite cas (1-3), (2-4) et (3-2)

Le calcul des minimums de lignes est donné ci-dessous :

$$RM = 7 + 6 + (8 \equiv 1 \rightarrow 3) + (5 \equiv 2 \rightarrow 4) + (10 \equiv 3 \rightarrow 2) = 36 \text{ km}$$

Considérons le circuit composé des pas 1-3, 5-2 et 3-5. La matrice réduite est donc la suivante :

	1	2	min
4	9	...	9
5	7	6	6

Fig. IV.16 Matrice réduite cas (1-3), (5-2) et (3-5)

Le calcul des minimums de lignes est donné ci-dessous :

$$RM = 9 + 6 + (8 \equiv 1 \rightarrow 3) + (5 \equiv 2 \rightarrow 4) + (9 \equiv 3 \rightarrow 5) = 37 \text{ km.}$$

Considérons le circuit composé des pas 1-3, 2-5 et 3-2. La matrice réduite est donc la suivante :

	1	4	min
4	9	...	9
5	7	6	6

Fig. IV.17 Matrice réduite cas (1-3), (2-5) et (3-2)

Le calcul des minimums de lignes est donné ci-dessous :

$$RM = 9 + 6 + (8 \equiv 1 \rightarrow 3) + (6 \equiv 2 \rightarrow 5) + (10 \equiv 3 \rightarrow 2) = 39 \text{ km}$$

Considérons le circuit composé des pas 1-3, 2-5 et 3-4. La matrice réduite est donc la suivante :

	1	2	min
4	9	5	5
5	7	...	7

Fig. IV.18 Matrice réduite cas (1-3), (2-5) et (3-4)

Le calcul des minimums de lignes est donné ci-dessous :

$$RM = 5 + 7 + (8 \equiv 1 \rightarrow 3) + (6 \equiv 2 \rightarrow 5) + (8 \equiv 3 \rightarrow 4) = 34 \text{ km}$$

### 5. Circuit Optimal

1. le pas 1-3-2-4-5-1

$$Z = 8 + 10 + 5 + 6 + 7 = 36 \text{ km}$$

2. le pas 1-3-5-2-4-1

$$Z = 8 + 9 + 6 + 5 + 9 = 37 \text{ km}$$

3. le pas 1-3-2-5-4-1

$$Z = 8 + 10 + 6 + 6 + 9 = 39$$

4. le pas 1-3-4-2-5-1

$$Z = 8 + 8 + 5 + 6 + 7 = 34, \text{ représente la meilleure solution.}$$

### 6. Construction de l'arbre branch and bound

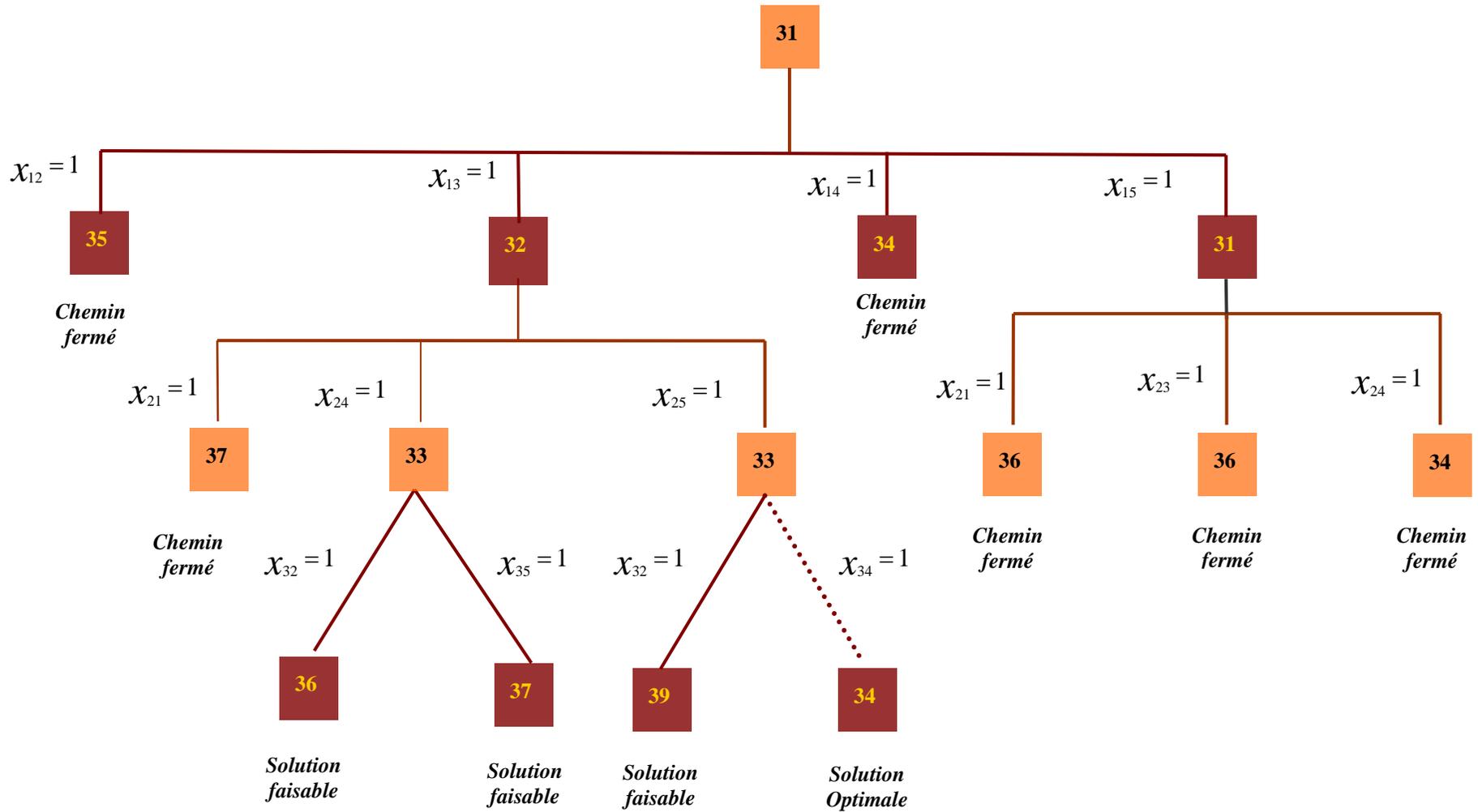


Fig .IV.19 Arbre Branch and Bound (TSP à 5 villes)

### IV. 3. 2. Les méthodes approchées

Se sont des méthodes qui sont souvent inspirées des mécanismes d'optimisation rencontrés dans la nature. Elles sont utilisées pour les problèmes où on ne connaît pas d'algorithmes de résolution en temps polynomial et pour lesquels on espère trouver une solution approchée de l'optimum global. Elles cherchent à produire une solution de meilleure qualité possible dictée par des heuristiques avec un temps de calcul raisonnable en examinant seulement une partie de l'espace de recherche.

Plusieurs méthodes d'approximation (heuristiques) ont été proposées qui approchent en temps raisonnable la solution optimale.

Dans ce cas l'optimalité de la solution n'est pas garanti ni l'écart avec la valeur optimale. Parmi ces heuristiques, on trouve les méta-heuristiques qui fournissent des schémas de résolution généraux permettant de les appliquer potentiellement à tous les problèmes.

Plusieurs classifications des méta-heuristiques ont été proposées, la plupart distinguant globalement deux catégories : celles se basant sur *une solution unique* et celles se basant sur *une population de solutions*. [30]

Une autres classification aussi proposée est que ces heuristiques peuvent être réparties en trois classes : *les heuristiques de construction*, *les heuristiques d'amélioration* et *les algorithmes composés* (qui combinent les deux premières). Les heuristiques de construction élaborent graduellement le tour en ajoutant une ville (nœud) à chaque étape, et cela lorsqu'on les applique au problème de voyageur de commerce, elles s'arrêtent dès que la solution est trouvée et n'essayent pas de l'améliorer. Dans cette catégorie il y a l'heuristique du plus *proche voisin* (en anglais *nearest neighbor*), l'algorithme glouton, etc. Les heuristiques d'amélioration consistent, une fois qu'un tour est générée par une heuristique de construction, à l'améliorer pour obtenir un tour de meilleure qualité. Dans cette catégorie, il y a également l'algorithme de la recherche tabou, le recuit simulé et les algorithmes génétiques. L'heuristique de colonie de fourmis combine les caractéristiques des deux classes. [31]

#### IV. 3. 2. a. Métaheuristiques à base de solution unique

Travaillent sur un seul point de l'espace de recherche à un instant donné en commençant avec une solution initiale puis de l'améliorer itérativement en choisissant une nouvelle solution dans son *voisinage*.

Voisinage de taille  $\varepsilon$  :  $N(X) = \{X' \in S / \|X' - X\| < \varepsilon\}$ , minimum local :  $\forall X' \in N(X)$  on a  $F(X') \leq F(X)$ .

### 1) Algorithmes de recherche locale pour une optimisation locale

Tout algorithme basé sur la notion de voisinage, est piégé par le premier optimum local. Le principe est simple, à partir d'une configuration initiale  $X_0$  on applique successivement des transformations à la solution courante. La recherche consiste à passer d'une solution à une solution voisine. De manière générale, les opérateurs de recherche locale s'arrêtent quand une solution localement optimale est trouvée, c'est à dire quand il n'existe pas de meilleure solution dans le voisinage. [30]

On distingue dans cette catégorie les algorithmes de descente. La méthode de *Descente simple* à titre d'exemple permet à partir d'une solution existante, à chercher *aléatoirement* une solution dans le voisinage et accepter cette solution si elle améliore la solution courante.

### 2) Algorithmes de recherche locale pour une optimisation globale

Tout algorithme local par son mécanisme de fonctionnement et global par sa recherche d'optimum global utilisant des mécanismes pour s'échapper de ces minima locaux.

#### a. Recuit simulé

Le recuit simulé est un algorithme stochastique de minimisation. Étant donné un espace  $E$  et une fonction  $V$  sur  $E$ , on cherche à localiser le ou les points où  $V$  atteint son minimum, et à estimer la valeur de  $V$  en ces points.

Cette approche présente cependant plusieurs inconvénients

- On peut par exemple remarquer qu'à l'approche du minimum, le processus ralentit de plus en plus.
- Mais un des aspects le plus limitant est que le processus gouverné par une équation différentielle est piégé par les minima locaux. [32]

#### b. Recherche tabou

La recherche Tabou est une méta-heuristique originalement développée par Glover. Elle est basée sur des idées simples, mais reste néanmoins efficace. Cette méthode combine une procédure de recherche locale avec un certain nombre de règles et de mécanismes lui permettant de surmonter l'obstacle des extremums locaux, tout en évitant les problèmes de

cycles. Elle a été appliquée avec succès pour résoudre de nombreux problèmes difficiles d'optimisation combinatoire. [33]

### **IV. 3. 2. b. Méta heuristiques à base de population de solutions**

Elles travaillent sur un ensemble de points de l'espace de recherche en commençant avec une population de solution initiale puis de l'améliorer au fur et à mesure des itérations. L'intérêt de ces méthodes est d'explorer un très vaste espace de recherche et d'utiliser la population comme facteur «de diversité» de plus elles sont très adaptées et très largement utilisées pour l'optimisation multi objectifs. [30]

L'évolution dans ce type de méthodes se fait à partir de transformation et de coopérations entre individus qui représentent individuellement une solution de l'espace total du problème, on distingue essentiellement les algorithmes génétiques, et les colonies de fourmis.

#### **i. Les algorithmes génétiques**

Les algorithmes génétiques ont été développés par Holland en 1975 [Holland 1975]. Ils sont basés sur la théorie de l'évolution naturelle des espèces énoncée par Darwin, ils présentent des qualités intéressantes pour la résolution des problèmes d'ordonnancement. Les individus ou chromosomes d'un algorithme génétique sont des codages des solutions possibles du problème. Comme dans la nature, ces individus forment une population qui va évoluer dans le temps selon des lois de sélection qui vont favoriser les mieux adaptés à se croiser entre eux en produisant des populations meilleures. L'évolution des individus d'une population à une autre se fait à l'aide de la reproduction. Les individus parents vont se reproduire pour donner des individus enfants qui seront, après avoir subis des opérations génétiques de croisement et de mutation, plus performants.

Comme dans la nature, ces reproductions se font avec une part de hasard. En effet, les parents candidats à la reproduction sont choisis d'une manière probabiliste proportionnelle à leurs aptitudes et l'étape de reproduction est choisie d'une façon totalement aléatoire.

Finalement, passant d'une génération à une autre, les individus forment une progéniture plus performante qui s'approche au mieux de la solution optimale.

Le schéma de la figure IV.20 représente le principe des algorithmes génétiques.

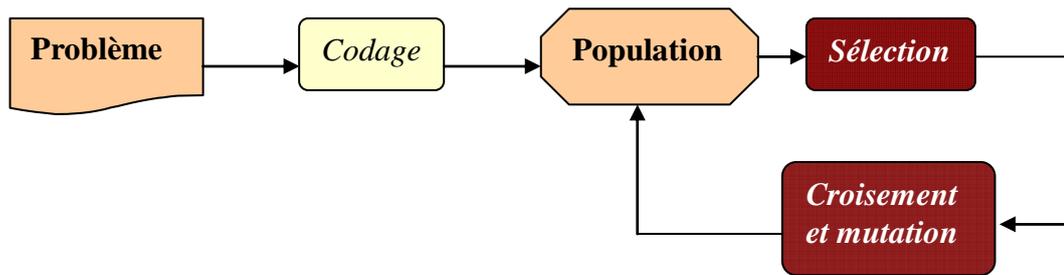


Fig. IV.20 Principes des algorithmes génétiques

## ii. Colonie de fourmis

Les algorithmes à base de colonies de fourmis ont été introduits par Dorigo. Une des applications principales de la méthode originale était le problème du voyageur de commerce et depuis elle a considérablement évolué. Cette nouvelle méta-heuristique imite le comportement de *fourmis* cherchant de la nourriture. A chaque fois qu'une fourmi se déplace, elle laisse sur la trace de son passage une odeur (la phéromone). Avec plusieurs de ses congénères, elle explore une région en quête de nourriture. Face à un obstacle, le groupe des fourmis explore les deux côtés de l'obstacle et se retrouvent, puis elles reviennent au nid avec de la nourriture. Les autres fourmis qui veulent obtenir de la nourriture elles aussi vont emprunter le même chemin. Si celui-ci se sépare face à l'obstacle, les fourmis vont alors emprunter préférentiellement le chemin sur lequel la phéromone sera la plus forte. Mais la phéromone étant une odeur elle s'évapore. Si peu de fourmis empruntent une trace, il est possible que ce chemin ne soit plus valable au bout d'un moment, il en est de même si des fourmis exploratrices empruntent un chemin plus long. Par contre, si le chemin est fortement emprunté, chaque nouvelle fourmi qui passe redépose un peu de phéromone et renforce ainsi la trace, donnant alors à ce chemin une plus grande probabilité d'être emprunté. [30]

### IV. 3. 2. c. Métaheuristiques avancées

Les algorithmes à base de population permettent d'explorer un espace de solutions très vaste.

Afin d'améliorer les performances d'une recherche globale, de nombreux auteurs proposent d'utiliser une recherche globale pour bien explorer l'espace de recherche conjointement avec une recherche locale pour mieux exploiter les zones prometteuses. Par exemple en ajoutant des techniques de recherche locale aux algorithmes génétiques, on obtient les algorithmes mémétiques.

### i. Algorithme mémétique

Moscato introduit en 87 pour la première fois les algorithmes mémétiques. L'idée principale de cette technique est de rendre plus agressif un algorithme génétique par l'ajout d'une recherche locale en plus de la mutation. On rencontre aussi le nom *genetic local search*. Une des observations générales provenant de l'implémentation d'un algorithme génétique basique est souvent la faible vitesse de convergence de l'algorithme. L'idée de Moscato est donc d'ajouter une recherche locale qui peut être une méthode de descente ou une recherche locale plus évoluée (recuit simulé ou recherche tabou par exemple). Cette recherche locale sera appliquée à tout nouvel individu obtenu au cours de la recherche. [30]

Le schéma de la figure IV.21 représente une classification des méthodes approchées

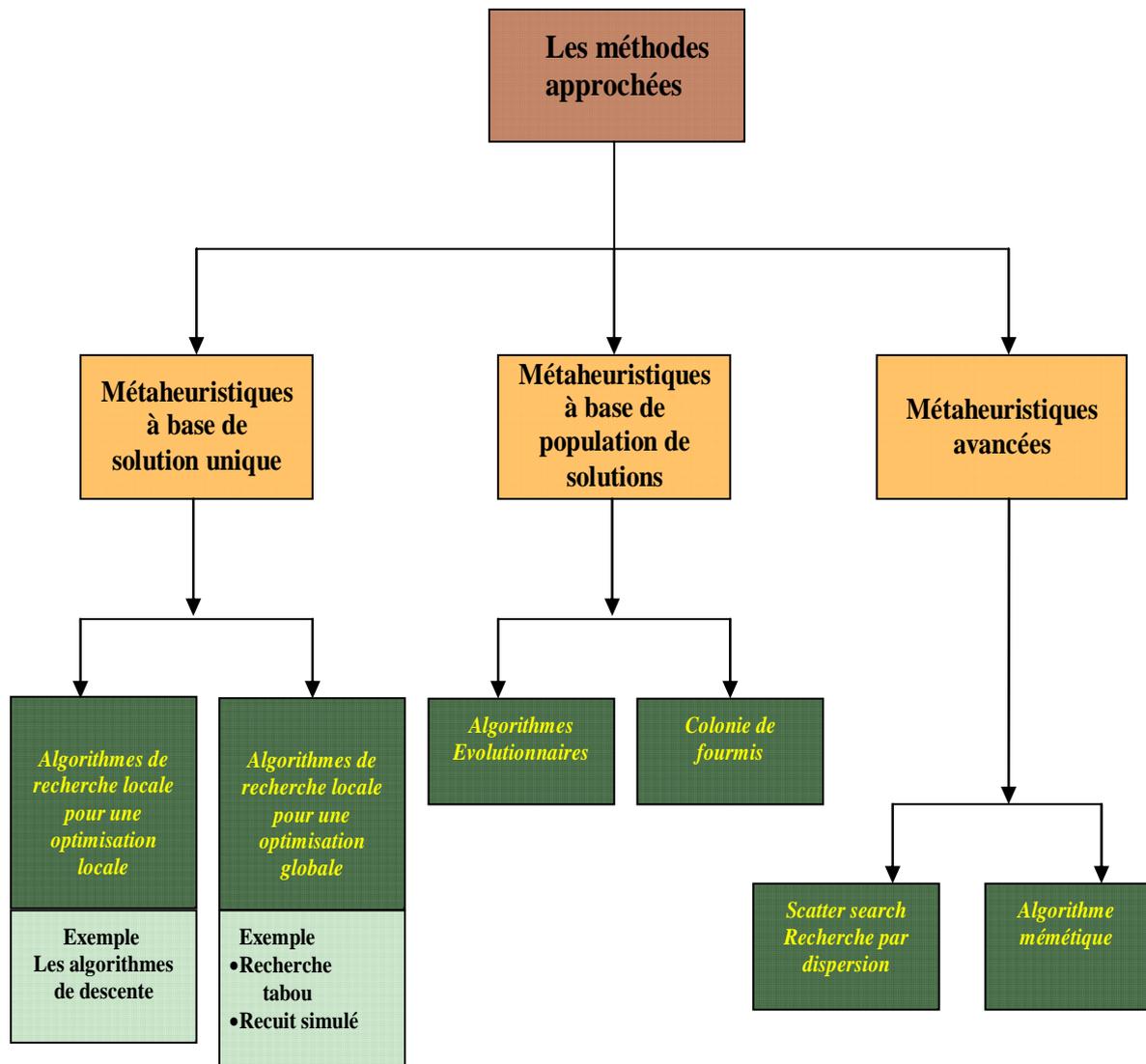


Fig. IV.21 Classification des méthodes approchées

### IV. 3. 2. d. Application des heuristiques au problème de TSP

Deux heuristiques sont combinés: le nearest neighbor et le subtour reversal. L'implémentation du premier est simple mais le deuxième nécessite plus de calculs. Le subtour reversal généralement donne de meilleurs résultats. Dans cette combinaison les résultats du nearest neighbor sont utilisés comme donnés d'entrée pour le subtour reversal

#### a. L'heuristique Nearest Neighbor

Une façon de calculer une approximation du voyageur du commerce est d'appliquer l'heuristique du « plus proche voisin » qui fonctionne comme suit :

1. on sélectionne une ville au hasard
2. trouver une ville plus proche et non visitée et y aller
3. répéter l'étape 2 s'il en existe des villes non encore visitées
4. retour à la première ville [34]

Le fonctionnement de l'algorithme est illustré par l'exemple suivant :

On dispose de 7 villes (a, b, c, d, e, f, g) ayant les coordonnées décrites à l'aide du tableau suivant :

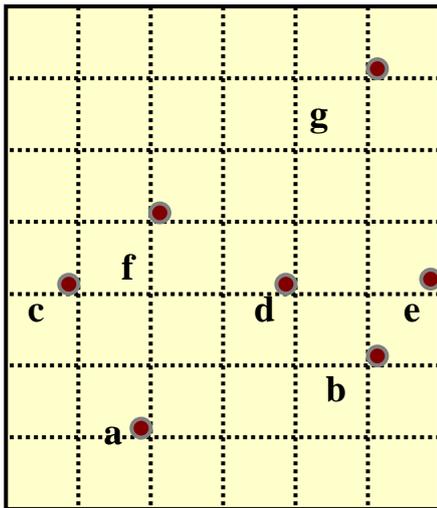
<i>ville</i>	<i>x</i>	<i>y</i>
<b>a</b>	2	1
<b>b</b>	5	2
<b>c</b>	1	3
<b>d</b>	4	3
<b>e</b>	6	3
<b>f</b>	2	4
<b>g</b>	5	5

Tableau IV.1 Coordonnées des villes

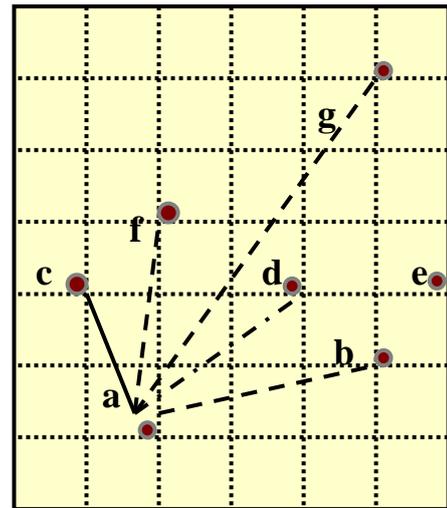
Initialement tous les sommets sont blancs, on commence par le choix d'une ville de départ (a) dans l'exemple, puis on colorise ce sommet en noir. À chaque fois on colorise un sommet et on l'ajoute au tour. On continue ce processus jusqu'à la colorisation de tous les sommets existants. Au moment de chaque passage d'un sommet à un autre on calcule le poids de cet arc on utilisant la formule suivant :

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

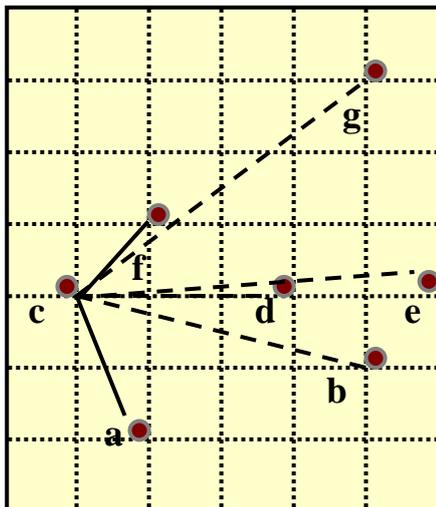
Les schémas de la figure IV.22 montrent l'application de la méthode nearest neighbor



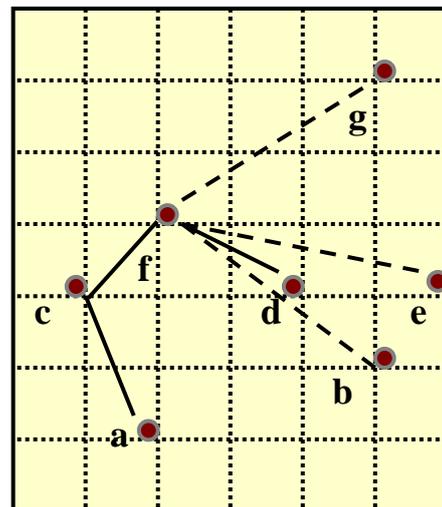
(1) a est la ville de départ



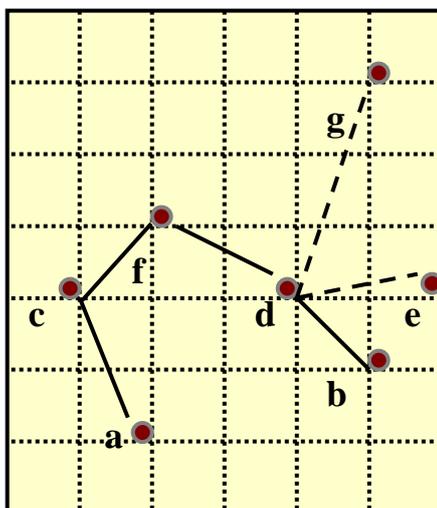
(2) Après le choix de la ville c



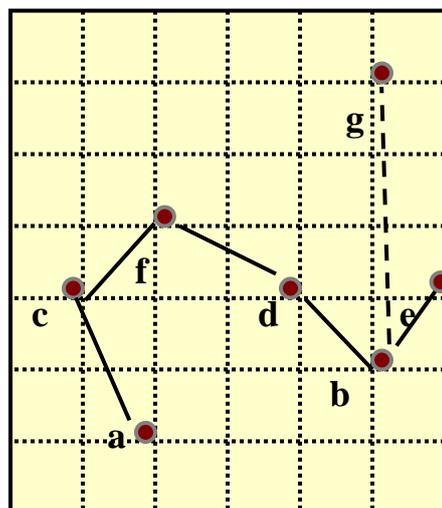
(3) Après le choix de la ville f



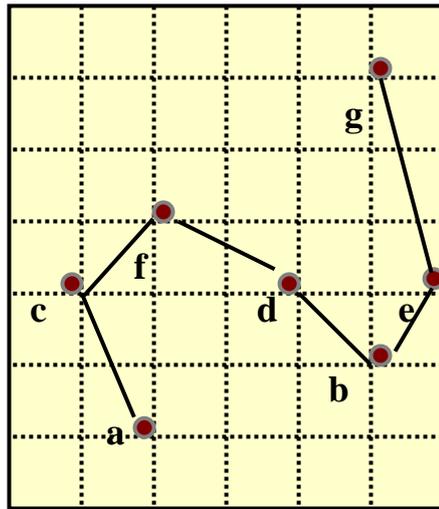
(4) Après le choix de la ville d



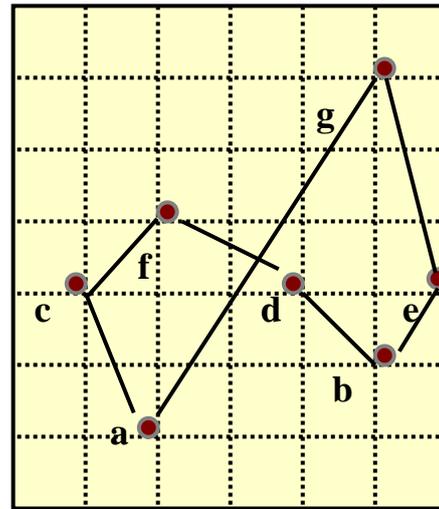
(5) Après le choix de la ville b



(6) Après le choix de la ville e



(7) Après le choix de la ville g



(8) le tour après le choix de la ville a encore

Fig. IV.22 Les figures (1 à 8) Application de l'algorithme du plus proche voisin

La longueur du tour obtenue par l'application de nearest neighbor est 15.95. Cette algorithme est basé sur le principe de choix « semble le meilleure pour l'instant » [35]

**b. L'heuristique de subtour reversal**

Soit la matrice suivante :

$$\|d_{ij}\| = \begin{pmatrix} \infty & 120 & 220 & 150 & 210 \\ 120 & \infty & 100 & 110 & 130 \\ 220 & 80 & \infty & 160 & 185 \\ 150 & \infty & 160 & \infty & 190 \\ 210 & 130 & 185 & \infty & \infty \end{pmatrix}$$

Dans une situation de  $n$  villes, cet algorithme commence par un tour faisable et on essaye de l'améliorer on inversant les sous tours 2-villes, suivis par les sous tours 3-villes, on continue de cette façon jusqu'aux sous tours de poids  $n-1$ .

Pour appliquer le reversal subtour à cet exemple, on considère le sous tour faisable suivant : 1-4-3-5-2-1 de longueur 745 et les étapes sont résumés dans le tableau IV.2 comme suit :

<i>type</i>	<i>Reversal</i>	<i>Tour</i>	<i>longueur</i>
<i>Début</i>	---	(1-4-3-5-2-1)	<u>745</u>
<i>Reversal 2 à la fois</i>	4-3	1-3-4-5-2-1	<u>820</u>
	3-5	1-4-5-3-2-1	<u>725</u>
	5-2	1-4-3-2-5-1	<u>730</u>
<i>Reversal 3 à la fois</i>	4-5-3	1-3-5-4-2-1	$\infty$
	5-3-2	1-4-2-3-5-1	$\infty$
<i>Reversal 4 à la fois</i>	4-5-3-2	1-2-3-5-4-1	$\infty$

Tableau IV.2 Le Reversal Subtour

Les reversaux deux à la fois du tour initial 1-4-3-5-2-1 sont : 4-3,3-5 et 5-2, ce qui conduit aux tours avec leurs longueurs associés : 820, 725 et 730. Puisque 1-4-5-3-2-1 donne le tour de plus petites longueur (=725), Ce tour est utilisé comme tour de départ pour construire les reversaux trois à la fois. Comme indique le tableau ces reversaux ne produisent aucun bon résultat. Le même résultat est appliqué pour les réversaux quatre à la fois, Donc 1-4-5-3-2-1 (avec une longueur de 725) qui fournis la bonne solution à l'heuristique.

A noter que les reversaux trois à la fois ne produisent pas le bon tour, et pour cette raison on a continué à utiliser le bon tour des reversaux deux à la fois avec les reversaux quatre à la fois. A noter également que les reversaux n'introduisent pas la ville de départ du tour (=1 dans l'exemple) par ce que le processus ne donne pas un tour. A titre d'exemple, 1-4 conduit à 4-1-3-5-2-1qui n'est pas un tour.

La solution déterminée par l'heuristique reversal est une fonction des tours initiaux utilisés pour commencer l'algorithme. Par exemple si on commence par 2-3-4-1-5-2 avec une longueur de 750 miles, l'heuristique produit le tour 2-1-4-3-5-2 avec une longueur de 745 miles, inférieur à la solution qu'on a dans le tableau ci-dessus. Pour cette raison il est préférable d'utiliser d'abord l'heuristique du plus proche voisin pour déterminer tous les tours qui se résultent à partir de l'utilisation de chaque nœud comme ville de départ et sélectionner le meilleur comme tour de départ pour l'algorithme reversal.

Le tableau suivant donne une application des l'heuristicques combinées pour le même exemple :

<i>heuristique</i>	<i>La ville de départ</i>	<i>Tour</i>	<i>longueur</i>
<b><i>Plus proche voisin</i></b>	1	1-2-3-4-5-1	780
	2	2-3-4-1-5-2	750
	3	(3-2-4-1-5-3)	735
	4	4-1-2-3-5-4	$\infty$
	5	5-2-3-4-1-5	750
<b><i>Les reversaux</i></b>	2-4	3-4-2-1-5-3	$\infty$
	4-1	(3-2-1-4-5-3)	725
	1-5	3-2-4-5-1-3	810
	2-1-4	3-4-1-2-5-3	745
	1-4-5	3-2-5-4-1-3	$\infty$
	2-1-4-5	3-5-4-1-2-3	$\infty$

Tableau IV.3 Combinaison Nearest neighbor &amp; reversal sub tours [36]

#### IV. 4. Conclusion

Les problèmes d'optimisation font apparaitre dans de nombreux domaines d'application tels que :

- Domaines Civil : services publics, hôpitaux, transport public, informatique
- Domaines Industriels: automobile, aviation, énergie, télécom, production
- Financier: gestion de portefeuille
- Militaire : gestion des ressources, logistique

L'optimisation est l'analyse et la résolution analytique ou numérique des problèmes qui consistent à déterminer le meilleur élément d'un ensemble, au sens d'un critère quantitatif donné.

Les problèmes d'optimisation combinatoires sont des problèmes dont le nombre de solutions est fini mais très grand car l'énumération de toutes les solutions possibles n'est pas une tâche pratique.

L'application d'un algorithme heuristique permet d'identifier au moins une solution réalisable à un problème d'optimisation mais sans garantir que cette solution soit optimale et le temps de calculs est le plus souvent important.



# Chapitre05

## *Optimisation du Processus de Soudage*

## Chapitre V

### Optimisation du Processus de Soudage

#### V. 1. Introduction

De nombreuses industries réalisent des assemblages d'éléments métalliques par soudage telles que : la Chaudronnerie, Menuiserie métallique, constructions navale, ferroviaire et automobile, etc.

Il existe aujourd'hui de nombreux procédés de soudage répondant à la très grande variété d'assemblage à réaliser et aux caractéristiques des métaux utilisés.

Afin d'augmenter la production et la qualité de soudage, des robots sont introduits. Ces robots offrent une flexibilité qui permet de disposer en un court laps de temps d'une solution d'automatisation complète. Les modifications à apporter dans le processus de production sont souvent rapidement possibles sans trop d'interventions.

L'utilisation des technologies de soudage robotisé est aujourd'hui devenue très courante chez les manufacturiers, principalement en raison des nombreux avantages qu'elle propose : meilleure précision et exactitude, les dépenses pour arriver au produit fini sont diminuées en matière de coût de main d'œuvre, de stockage, d'expédition. En plus ces robots peuvent fonctionner dans des conditions dures telles que les milieux à température élevée et les milieux hostiles.

L'objectif de ce chapitre est d'optimiser le processus de soudage en se basant sur des méthodes d'optimisation dans deux variantes, la première est une optimisation heuristique de construction on appliquant la méthode « Nearest Neighbor » et la deuxième est une méthode heuristique d'amélioration on appliquant la méthode de « Subtour Reversal Heuristic ».

Notre travail est effectué au niveau de TIRSAM. Cette entreprise est spécialisée dans la fabrication des semi remorques.

Nous commençons ce chapitre par une présentation générale du processus de soudage, puis une présentation de l'entreprise et en fin par une application des méthodes d'optimisation en se basant sur des données issues de cette entreprise.

## V. 2. Généralités sur le soudage

### V. 2. 1. Définition

Le soudage est une opération consistant à assembler deux éléments métalliques, par fusion et re-solidification. En d'autres termes, il s'agit d'assurer la continuité mécanique entre les parties à assembler avec ou sans l'aide d'un produit d'apport permettant un meilleur lien entre les pièces. Dans le cas de l'utilisation d'un métal d'apport, ce composé doit avoir des propriétés au moins égales et une température de fusion du même ordre de grandeur que celle des matériaux à assembler. Différents types d'assemblage peuvent être rencontrés en fonction des contraintes appliquées à la structure réalisée, de la nature et des dimensions de l'élément à assembler (voir figure V.1).

Durant la phase de soudage, le métal d'apport ainsi que le métal de base (c'est à dire les éléments à assembler) sont portés à la température de fusion permettant ainsi un mélange des éléments (voir figure V.2). L'utilisation du métal d'apport répond à deux besoins : tout d'abord, il est nécessaire d'apporter de la matière afin de combler une préparation de joint (chanfrein) assurant une section efficace entre les deux parties à assembler et d'autre part, il est aussi nécessaire de changer la composition métallographique de la zone fondue afin de lui donner des caractéristiques mécaniques voulues.

La zone fondue ou cordon de soudage est constituée du mélange des deux métaux de base et dans la plupart des cas d'un métal d'apport. De part et d'autre de cette zone fondue, les éléments à assembler, appelés métaux de base ont subi, des transformations de leur structure cristalline dues à l'élévation de température et cette région proche du cordon de soudage est appelée zone thermiquement affectée. Enfin, la surface de séparation entre la zone fondue et la zone thermiquement affectée porte le nom de zone de liaison. En fonction de l'épaisseur des éléments à assembler et de la densité d'énergie fournie par la source, il est possible de faire varier la profondeur de la zone fondue, on parle alors de niveau de pénétration.

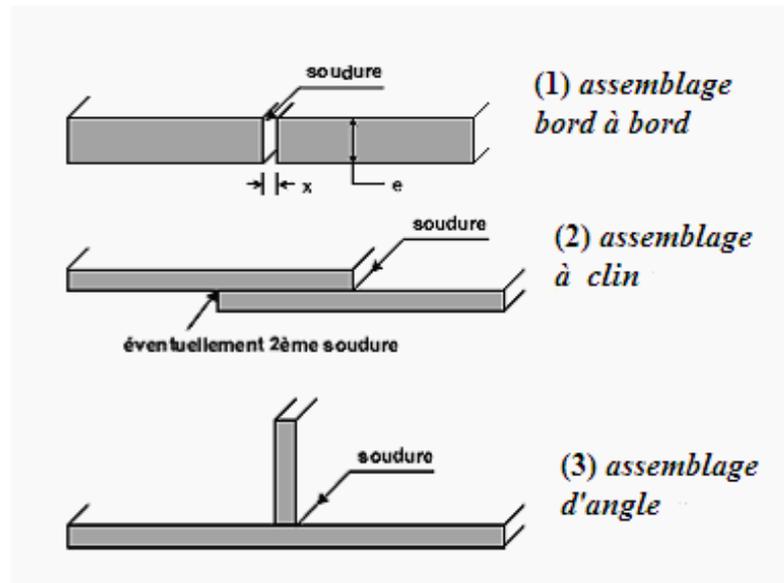


Fig. V.1 Les types de soudage

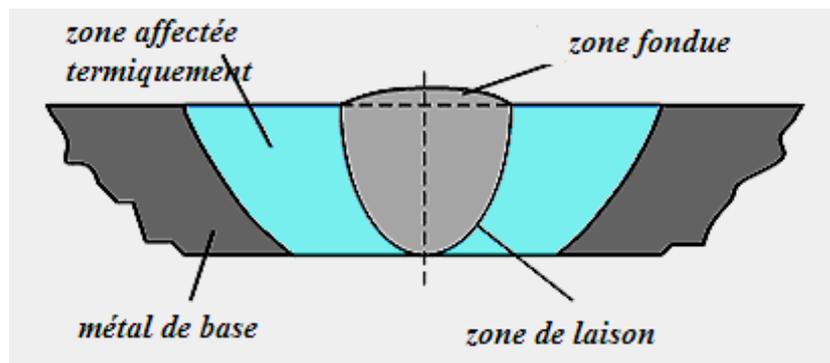


Fig. V.2 Coupe transversale d'un assemblage [37]

## V. 2. 2. Types de soudage

### V. 2. 2. a. Procédé de soudage par résistance électrique

Le soudage par résistance est un procédé simple et rapide, qui produit la chaleur grâce à la résistance électrique à l'interface en appliquant un effort de serrage, et une forte intensité électrique. C'est un procédé de soudage sans atmosphère protectrice.

Les raisons pour lesquelles le soudage par point est largement répandu dans l'industrie automobile sont le faible coût de fabrication par rapport à la qualité de l'assemblage obtenu, et l'automatisation de procédé. Environ (3000 à 4000) points soudés par ce procédé se trouvent dans une structure caisse en blanc. [38]

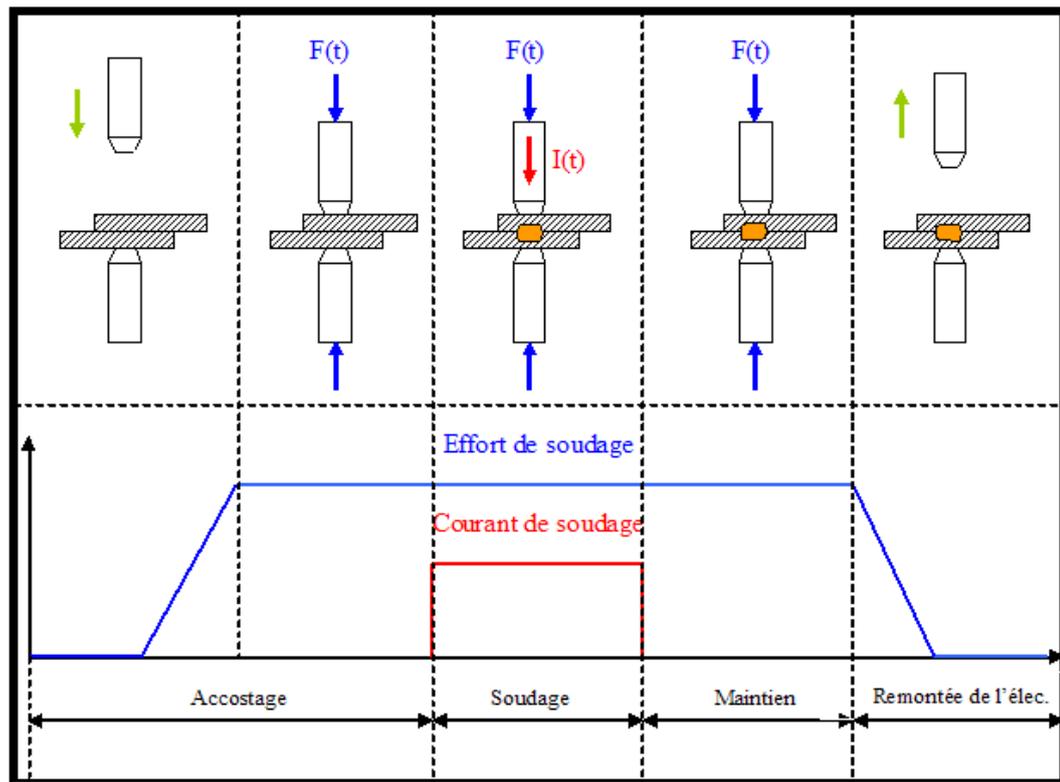


Fig. V.3 Les phases du procédé de soudage par point

### V. 2. 2. b. Le soudage à l'arc électrique

La soudure à l'arc est actuellement très utilisée dans le domaine de la métallurgie [Caz\_1, Cep\_1, Fau\_1, Fre\_1, Ili\_1, Kie\_1]. Un arc électrique, établi entre une électrode et les pièces métalliques à souder, génère un plasma de très haute température pour fusionner et unifier les pièces à souder.

On distingue plusieurs variantes de la soudure à l'arc pouvant être classées en deux catégories. La première catégorie regroupe les soudures avec des électrodes enrobées connues aussi sous le nom de SMAW (Shielded Metal Arc Welding) ou MMA (Manual Metal Arc). Les électrodes sont constituées d'un côté par les matériaux à souder et d'un autre côté par une baguette de métal d'apport. La soudure est directement effectuée dans l'air. La baguette est alors enrobée d'un laitier qui, lors de sa fusion, forme une couche protectrice pour éviter l'oxydation du métal (voir figure V.4(a)).

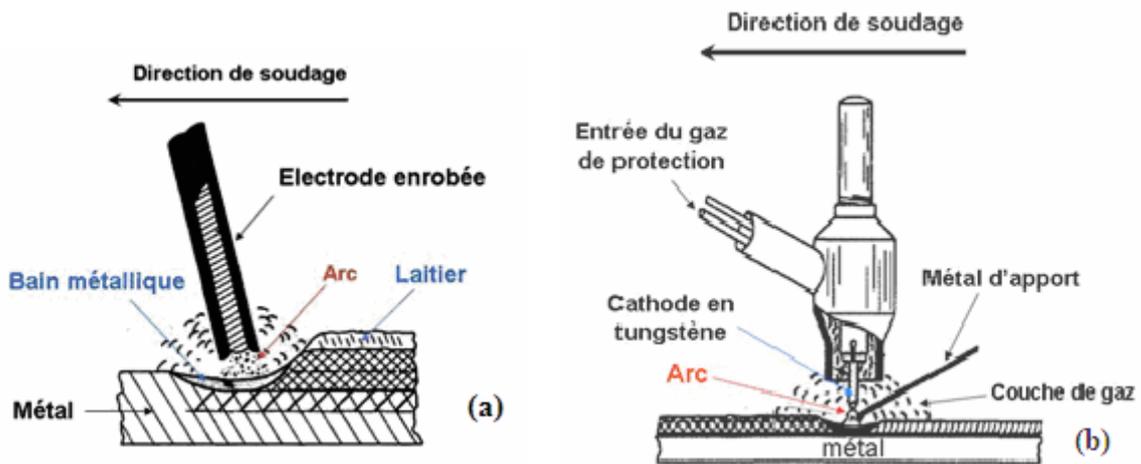


Fig. V.4 (a) Soudure MMA. (b) Soudure TIG.

La deuxième catégorie comprend les soudures avec des électrodes non fusibles comme le procédé TIG (Tungsten Inert Gas), illustré sur la figure V.4 (b), et le PAW (Plasma Arc Welding). L'arc est généré entre les pièces à souder et une cathode constituée d'un matériau infusible aux conditions opératoires. On utilise généralement une cathode en tungstène pur ou thorium selon les caractéristiques du métal à souder. Ce choix de l'électrode négative permet essentiellement d'améliorer le caractère thermo émissif de la pointe, ce qui facilite l'amorçage de l'arc et assure une meilleure stabilité comparé au cas d'une cathode en tungstène pur. Le pourcentage en thorium est typiquement aux alentours de 2%. Un gaz inerte est injecté en permanence pour servir de protection contre l'oxydation. La dernière catégorie regroupe les soudures à électrodes fusibles qui sont surtout utilisées en mode semi automatique. Parmi elles figure la soudure MIG MAG (Metal Inert Gas Metal Active Gas) qui fonctionne suivant le même principe que la soudure TIG sauf au niveau de la cathode. En effet, au lieu d'une cathode infusible, un fil massif enroulé dans un dévidoir est amené automatiquement au niveau de l'arc pour constituer le métal d'apport [39].

### V. 2. 2. c. Soudage MIG/MAG

- Un fil métallique passe dans la torche et se trouve fondu dans l'arc.
- Le fil constitue à la fois un conducteur de courant et un métal d'apport.
- L'énergie électrique est fournie par une source d'alimentation de soudage.
- L'arc et le bain de fusion sont protégés par un gaz de protection inerte ou actif.

Le procédé de soudage MIG/MAG présente les avantages suivants :

- De hautes performances.
- Un apport d'énergie vers la pièce relativement faible.

- Une automatisation simple.
- La productivité en MIG/MAG face au MMA
- Pas de remplacement continu d'électrodes.
- Pas de laitier à enlever, sauf sur certains fils fourrés.
- Le taux de fusion est plus élevé qu'en MMA.

### V. 3. Application des méthodes d'optimisation sur un système industriel de fabrication

#### V.3.1. Présentation de l'entreprise et situation géographique

Tirsam siège à Batna (la zone industrielle Kechida) a commencé son activité dans le secteur de la fabrication de semi remorque en 2007, elle comptait 70 employés à peine trois ans passés.

Tirsam, fabricant et distributeur de la remorque et de la semi remorque devient leader national, avec une production annuelle de 3000 semi remorques (tous types confondus), soit 250 unités par mois.

Aujourd'hui l'entreprise a beaucoup grandi et emploie plus de 800 employés, travaillant dans plusieurs unités qui façonnent l'effort au service de l'efficacité et de la qualité.



Fig. V.5 Localisation géographique de TIRSAM

### V.3.2. Types de produits et services offerts

- Fabrication De Semi remorques
- Plateaux
- Benne Travaux public
- Benne Céréalière
- Citerne Carburant
- Porte Engins
- Châssis Porte Conteneurs

### V.3.3. Processus de soudage au niveau de Tirsam

La fabrication de la semi remorque au niveau de l'entreprise est basée sur le processus de soudage

Pour clarifier, on prend une partie très importante de la semi remorque à savoir le longeron. Le Longeron est la pièce maitresse dans la construction d'une semi remorque. Il est constitué de pièces en fer plat en acier de type E36 ayant les dimensions suivantes :

- longueur = 12 500 mm,
- largeur = 140 mm,
- épaisseur = 15 mm.

Les fers plats servent de support latéral de quatre tôles d'acier de type E24 dont trois ayant comme épaisseur 6mm et l'autre se trouvant dans l'extrême partie du Longeron ayant comme épaisseur 8 mm. (Voir figure V.6)

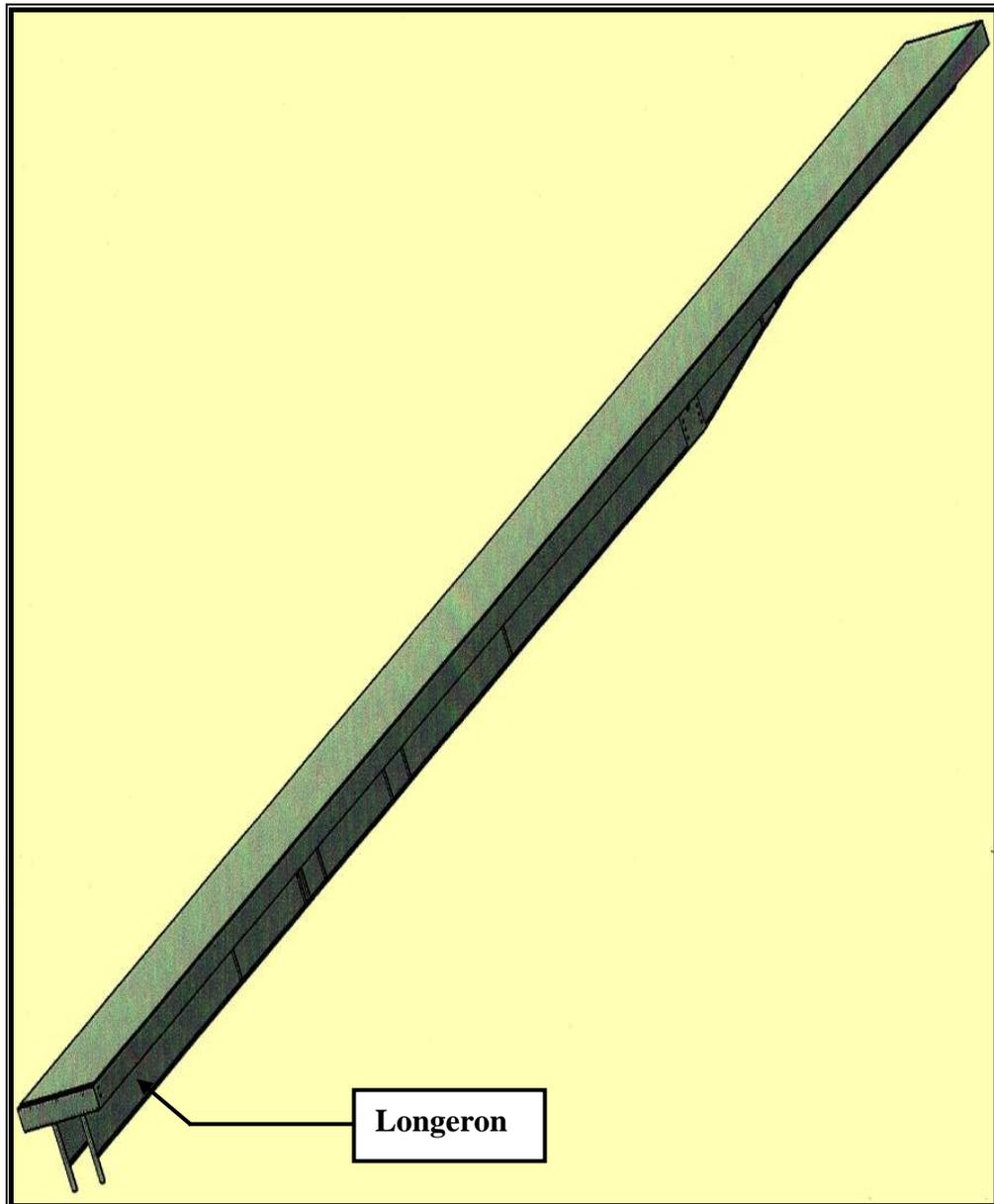


Fig. V.6 Positionnement du Longeron dans la semi remorque

Les tôles d'acier sont placées sur des calles bien réparties sur la surface des tôles pour assurer d'une part une bonne base de portance et de les ajuster au milieu de la hauteur du fer plat d'autre part. Après avoir placé les quatre tôles entre les deux fers plats un processus d'assemblage est effectué par le biais de vérins hydrauliques installés de part et d'autre des fer-plats.

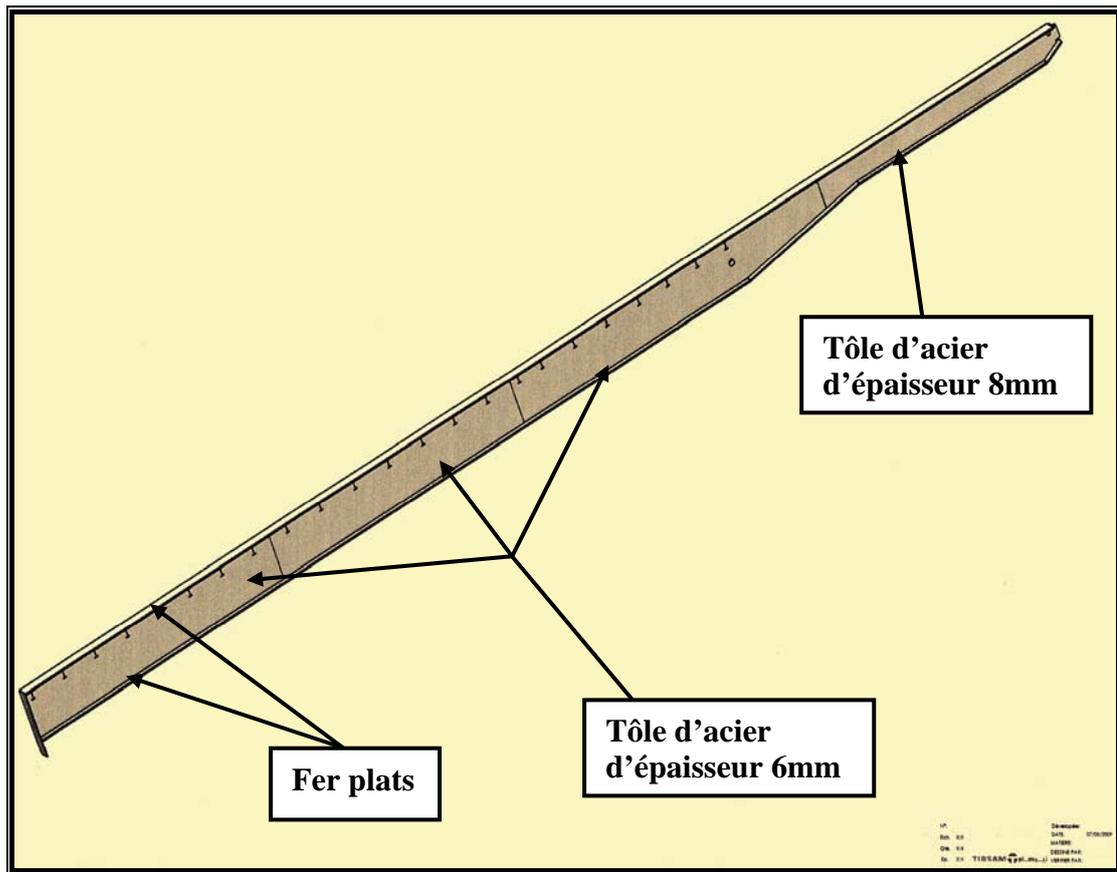


Fig. V.7 Schéma du Longeron en 3D

Après serrage par le biais de vérins un processus de soudage manuel (un operateur qui effectue cette opération) permettant la fixation de l'ensemble constitué des tôles avec les fer plats, en réalisant un ensemble de points de soudures répartis selon le schéma de la figure ci dessous (voir figure V.8). Noter que les lignes d'intersection entre les tôles ont préalablement subit au chanfrinage.

#### V.3.4. Dispersion des points de soudage sur le longeron

Le nombre total de points de soudage considéré est 34 points dispersés sur le longeron comme la montre la figure V.8.

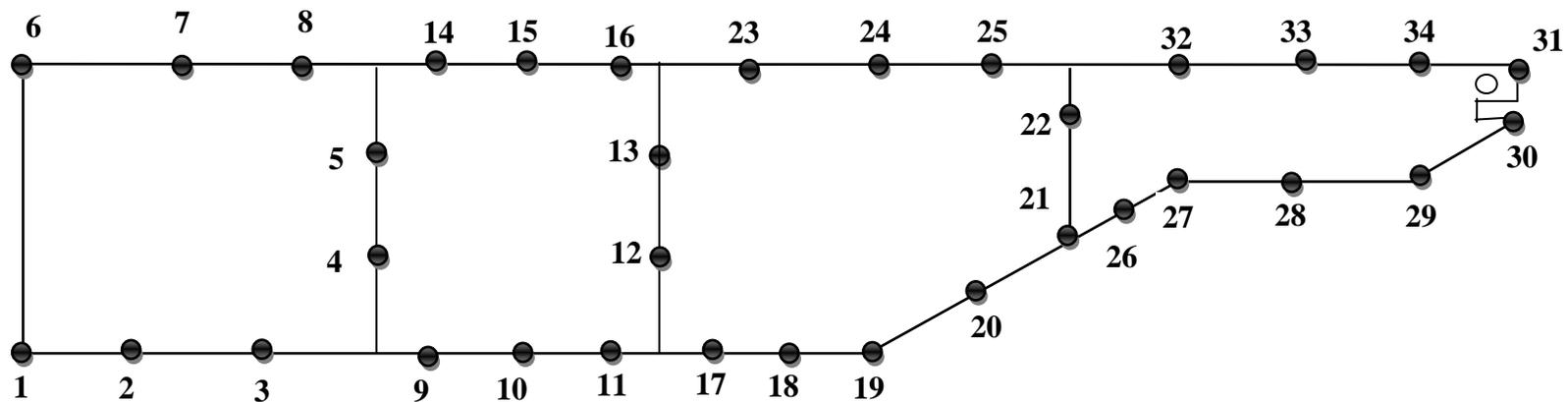


Fig. V.8 les points des soudages essentiels sur le longeron

### V.3.5. Les données nécessaires

- Les coordonnées des points sont données dans le tableau suivant :

	X	Y		X	Y		X	Y		X	Y
<b>1</b>	<b>0</b>	<b>0</b>	<b>9</b>	<b>3500</b>	<b>0</b>	<b>17</b>	<b>7045</b>	<b>0</b>	<b>26</b>	<b>9668</b>	<b>210</b>
<b>2</b>	<b>1000</b>	<b>0</b>	<b>10</b>	<b>4200</b>	<b>0</b>	<b>18</b>	<b>7645</b>	<b>0</b>	<b>27</b>	<b>9814</b>	<b>235</b>
<b>3</b>	<b>2000</b>	<b>0</b>	<b>11</b>	<b>5000</b>	<b>0</b>	<b>19</b>	<b>8440</b>	<b>0</b>	<b>28</b>	<b>10814</b>	<b>235</b>
<b>4</b>	<b>2955</b>	<b>200</b>	<b>12</b>	<b>5845</b>	<b>240</b>	<b>20</b>	<b>8960</b>	<b>90</b>	<b>29</b>	<b>12162</b>	<b>235</b>
<b>5</b>	<b>2955</b>	<b>400</b>	<b>13</b>	<b>5845</b>	<b>390</b>	<b>21</b>	<b>9545</b>	<b>189</b>	<b>30</b>	<b>12376</b>	<b>260</b>
<b>6</b>	<b>0</b>	<b>494</b>	<b>14</b>	<b>3300</b>	<b>494</b>	<b>22</b>	<b>9545</b>	<b>360</b>	<b>31</b>	<b>10445</b>	<b>494</b>
<b>7</b>	<b>1200</b>	<b>494</b>	<b>15</b>	<b>4500</b>	<b>494</b>	<b>23</b>	<b>7045</b>	<b>494</b>	<b>32</b>	<b>11045</b>	<b>494</b>
<b>8</b>	<b>2500</b>	<b>494</b>	<b>16</b>	<b>5500</b>	<b>494</b>	<b>24</b>	<b>7945</b>	<b>494</b>	<b>33</b>	<b>11945</b>	<b>494</b>
						<b>25</b>	<b>9545</b>	<b>494</b>	<b>34</b>	<b>12500</b>	<b>494</b>

Tableau V.1 Coordonnées de points de soudage

- La matrice des distances est calculée à l'aide de la formule de distance euclidienne comme suit tout en utilisant les coordonnées des points de soudage précédentes :

$$\sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

- La matrice résultante est une matrice (34x34) de nature symétrique.

## V. 4. Détermination de problème et l'environnement de travail

L'objectif principal est trouver un plus cours chemin possible au moment de la réalisation des points de soudage par l'opérateur humain afin de fixer les tôles du longeron en premier temps et puis de proposer un robot de soudage réalisant cette tâche pour la rendre automatique on éloignant l'homme, pour avoir un gain du temps qui va influencer d'une manière positive sur la productivité de l'entreprise. Pour cet effet notre problème est considéré comme un problème de voyageur de commerce, il s'agit aussi d'un problème de minimisation de la distance parcouru lors de l'accomplissement de la tâche précitée. Le problème TSP est extrêmement important dans le domaine de la recherche opérationnelle, il appartient à une classe de problèmes connus sous le nom de problèmes d'optimisation combinatoire qui est très facile à décrire et très difficiles à résoudre. le TSP est également classés parmi les

problèmes dits NP hard class. Par conséquent, la probabilité d'existence d'un algorithme efficace pour ce type de problème est presque nulle.

Les algorithmes de branch and bound donnent une solution exacte pour ce type de problème, mais ils sont également considérés comme étant des problèmes à caractère exponentiel les plus pessimistes car ils ne garantissent pas un temps d'exécution polynomial, on peut tomber dans une situation où la consommation du temps CPU est énorme après terminaison et parfois ne se termine plus. Car pour un problème disposant de  $n$  villes, on a  $(n-1)!$  solutions possibles, lorsque on prend à titre d'exemple un problème à 20 villes le nombre de solutions possibles est  $19!$  (en chiffres est égale à 121645100408832000), donc on ne peut pas résoudre un tel problème avec un temps CPU raisonnable à cause de la nature exponentiel du TSP qui fait partie de la théorie de la complexité, ce dernier est un autre domaine de recherche.

Le temps d'exécution d'un programme informatique est indicateur de performance mais pas le seul on peut aussi tomber dans un autre problème qui est l'espace de stockage mais c'est négligeable pour notre problème. Toutes ces contraintes nous mène à résoudre notre problème en utilisant des algorithmes heuristiques ayant les caractéristiques suivantes :

- Bonnes solutions
- Solutions approchées
- Solutions relativement rapides car ils s'exécutent dans un temps de calcul polynomial

On a résolu notre problème on procédant comme suit :

Utilisation de l'algorithme nearest neighbor comme une heuristique de construction dans la première phase, et l'algorithme subtour reversal heuristic comme un algorithme d'amélioration dans une deuxième phase comme l'indique l'organigramme suivant :

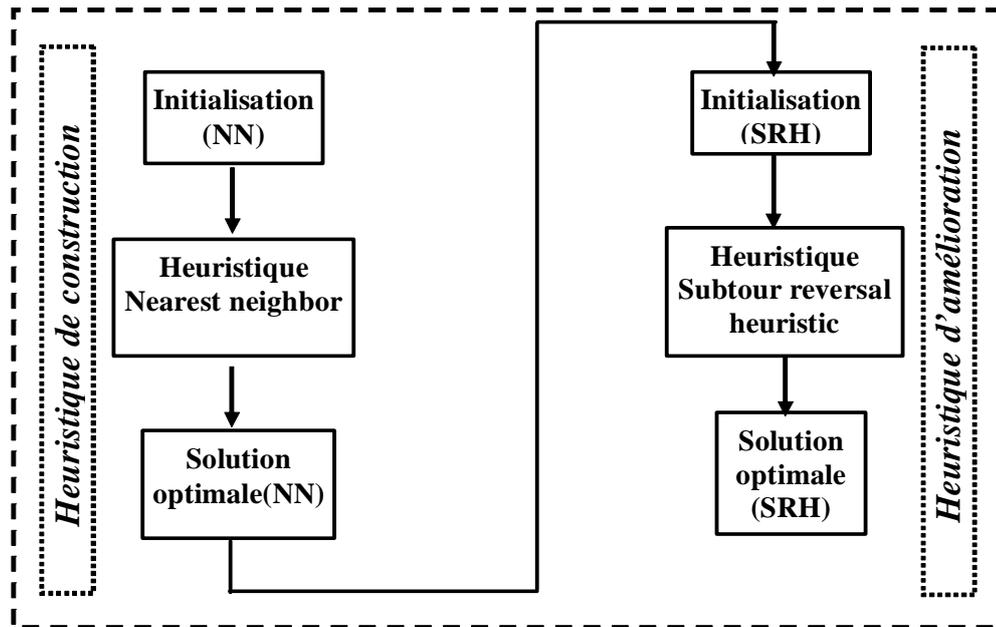


Fig. V.9 Schéma des heuristiques appliquées pour l'optimisation

#### ✓ L'environnement de travail

La simulation est réalisée sous le logiciel MATLAB, MATLAB est un langage de calcul scientifique de haut niveau et un environnement interactif pour le développement d'algorithmes, la visualisation et l'analyse de données, ou encore le calcul numérique.

### V. 5. Application d'une heuristique de construction

Après la récolte des données nécessaires, notre objectif est de minimiser la distance traversée pour fixer les tôles on réalisant des points.de soudage.

#### ◆ La méthode nearest neighbor

La matrice globale dispose de 34 points dispersées sur la totalité du longeron, après l'application de la méthode nearest neighbor le résultat de la simulation sous matlab obtenu est :

La distance optimale est :  $d = 29332$  mm

Le chemin optimal est :

**29 30 34 33 32 28 31 27 26 21 22 20 19 24 18 17 23 13 12 16 11  
15 10 9 14 5 4 8 3 7 2 1 6 25 29**

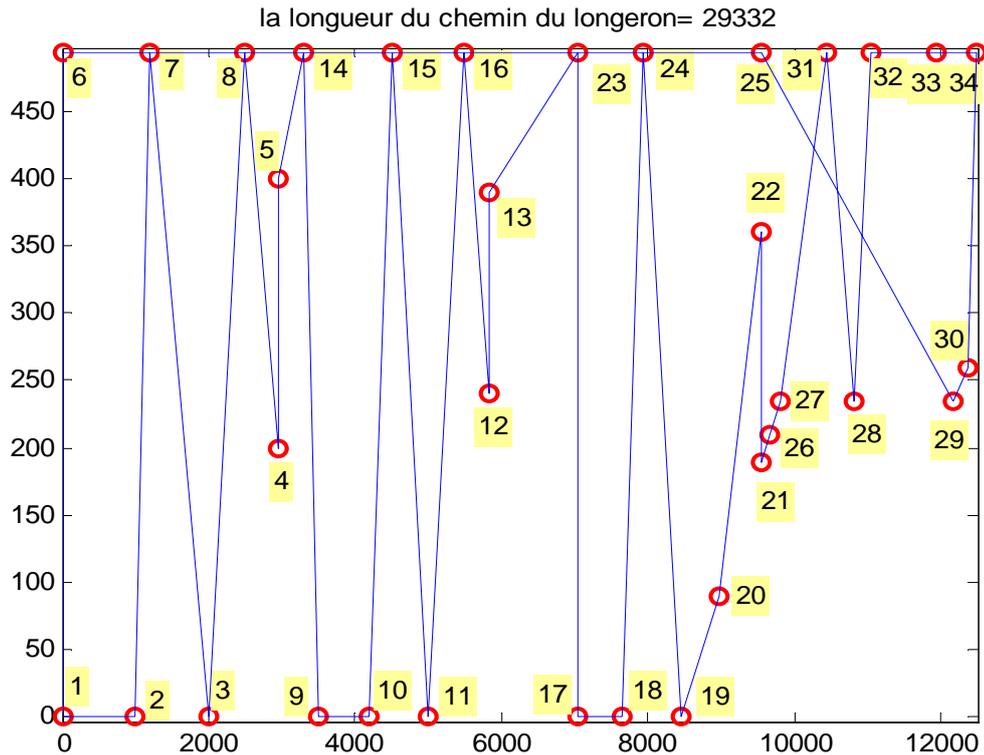


Fig. V.10 Nearest Neighbor appliquée au longeron

## V. 6. Application d'une heuristique d'amélioration

### ◆ La méthode de Subtour Reversal Heuristic

Pour améliorer les résultats obtenus par l'application de la méthode nearest neighbor, on applique une autre méthode heuristique appelée « **subtour reversal heuristic** », dans cette section on présente plusieurs itérations permettant l'amélioration des résultats de la première méthode appliquée.

#### • Inversion deux à la fois

- ✓ Initialement, la distance  $d : d = 29332mm$ .
- ✓ L'inversion de la séquence (13 12) provoque une amélioration par la réduction de la distance parcourue ( $d = 29286mm$ )
- ✓ Le chemin après inversion est le suivant :

**29 30 34 33 32 28 31 27 26 21 22 20 19 24 18 17 23 12 13 16 11 15**  
**10 9 14 5 4 8 3 7 2 1 6 25 29**

- ✓ La figure V.11 représente une simulation Matlab pour l'inversion deux à la fois de la méthode subtour reversal heuristic.

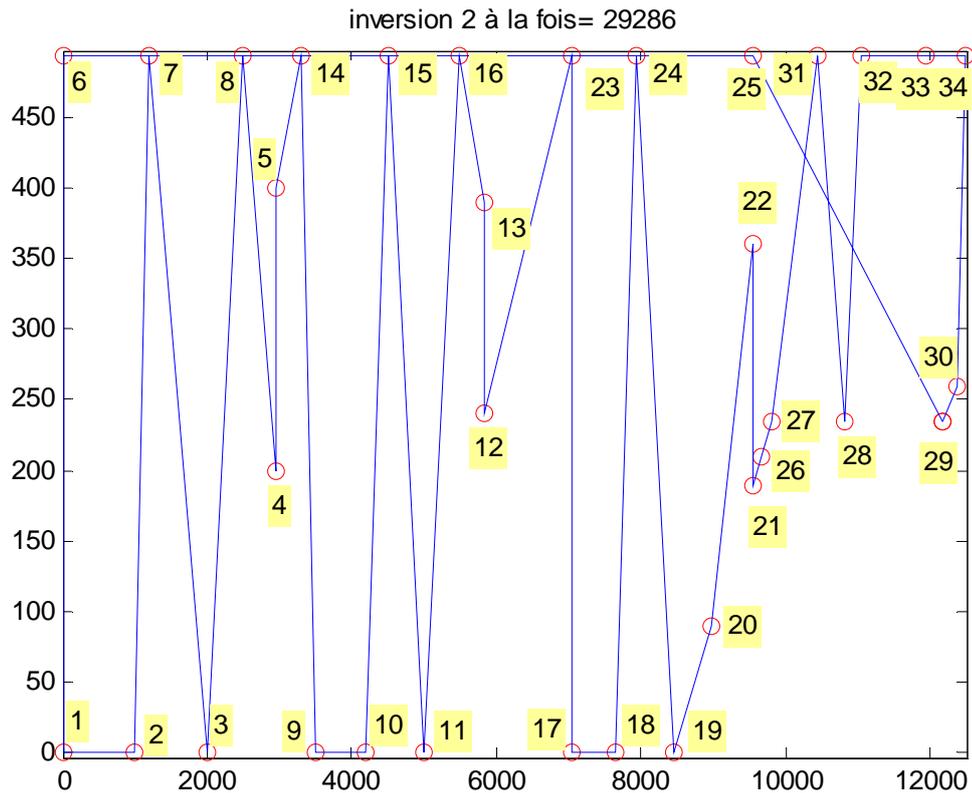


Fig. V.11 Subtour reversal heuristic (inversion deux à la fois)

- **Inversion trois à la fois**

- ✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 29286mm$
- ✓ La séquence à inverser est **(2 1 6)**
- ✓ La distance après inversion est :  $d = 28968mm$
- ✓ Le taux d'amélioration est  $\Delta d = 318mm$
- ✓ Le chemin obtenu est le suivant :

**29 30 34 33 32 28 31 27 26 21 22 20 19 24 18 17 23 12 13 16**  
**11 15 10 9 14 5 4 8 3 7 6 1 2 25 29**

- ✓ La simulation par matlab est représentée par la figure V.12.

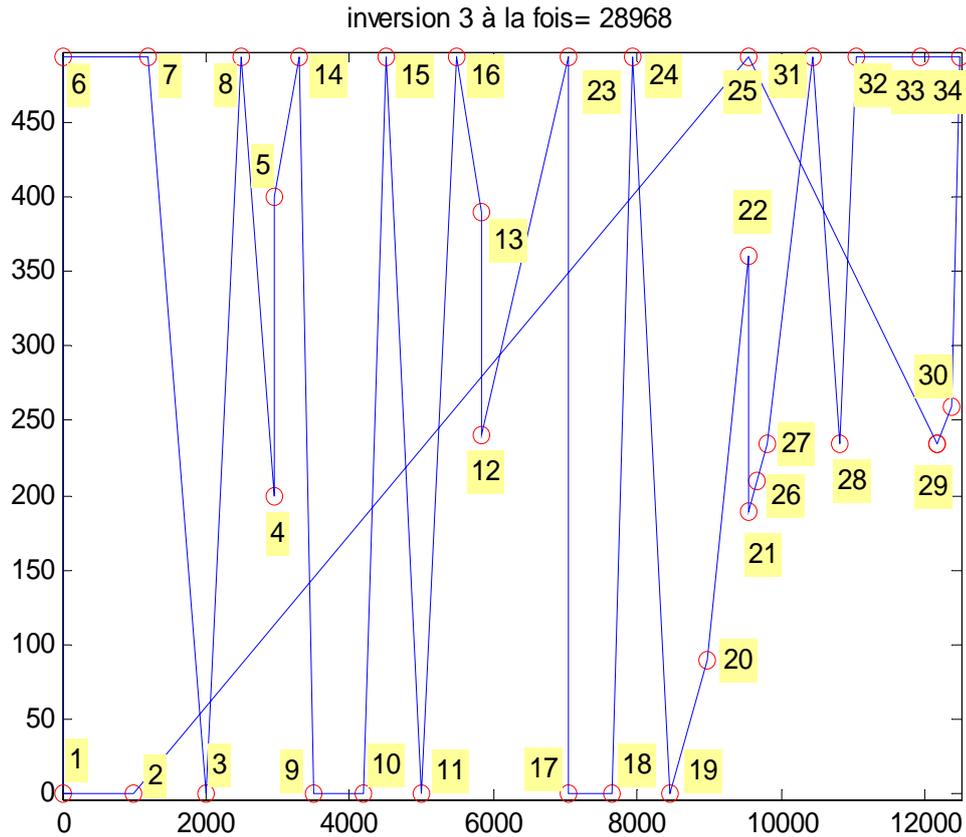


Fig. V.12 Subtour reversal heuristic (inversion trois à la fois)

• **Inversion quatre à la fois**

- ✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 28968mm$
- ✓ La séquence à inverser est (7 6 1 2)
- ✓ La distance après inversion est :  $d = 28814mm$
- ✓ Le taux d'amélioration est  $\Delta d = 154mm$
- ✓ Le chemin obtenu est le suivant :

**29 30 34 33 32 28 31 27 26 21 22 20 19 24 18 17 23 12 13 16  
11 15 10 9 14 5 4 8 3 2 1 6 7 25 29**

- ✓ La simulation par matlab est comme indique la figure V.13

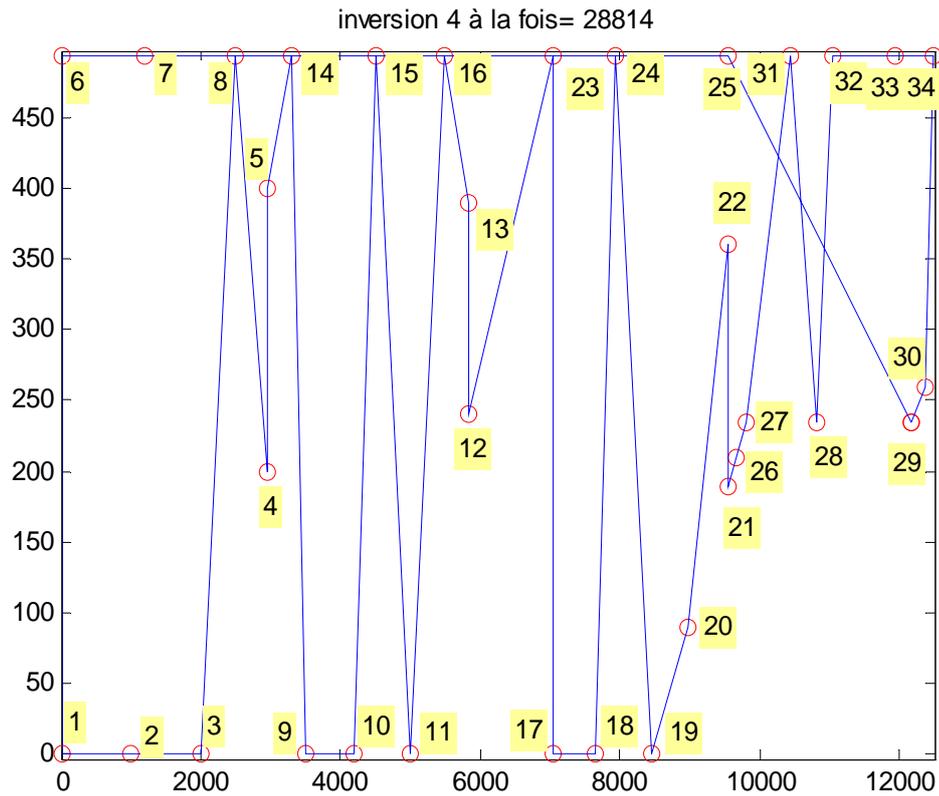


Fig. V.13 Subtour reversal heuristic (inversion quatre à la fois)

• **Inversion cinq à la fois**

- ✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 28814mm$
- ✓ La séquence à inverser est (3 2 1 6 7)
- ✓ La distance après inversion est :  $d = 28628mm$
- ✓ Le taux d'amélioration est  $\Delta d = 186mm$
- ✓ Le chemin obtenu est le suivant :

**29 30 34 33 32 28 31 27 26 21 22 20 19 24 18 17 23 12 13 16  
11 15 10 9 14 5 4 8 7 6 1 2 3 25 29**

- ✓ La simulation par matlab est illustrée dans la figure V.14

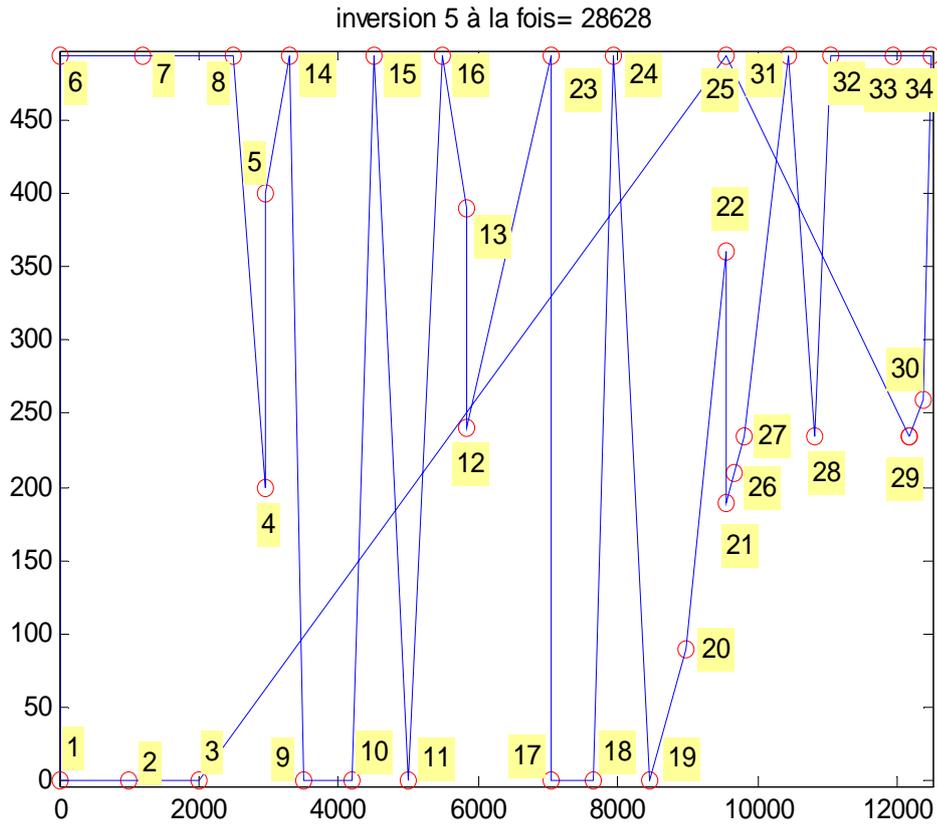


Fig. V.14 Subtour reversal heuristic (inversion cinq à la fois)

- **Inversion six à la fois**

✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 28628mm$

✓ La séquence à inverser est **(8 7 6 1 2 3)**

✓ La distance après inversion est :  $d = 28546mm$

✓ Le taux d'amélioration est  $\Delta d = 82mm$

✓ Le chemin obtenu est le suivant :

**29 30 34 33 32 28 31 27 26 21 22 20 19 24 18 17 23 12 13 16**

**11 15 10 9 14 5 4 3 2 1 6 7 8 25 29**

✓ Cette inversion est représentée en matlab par la figure V.15.

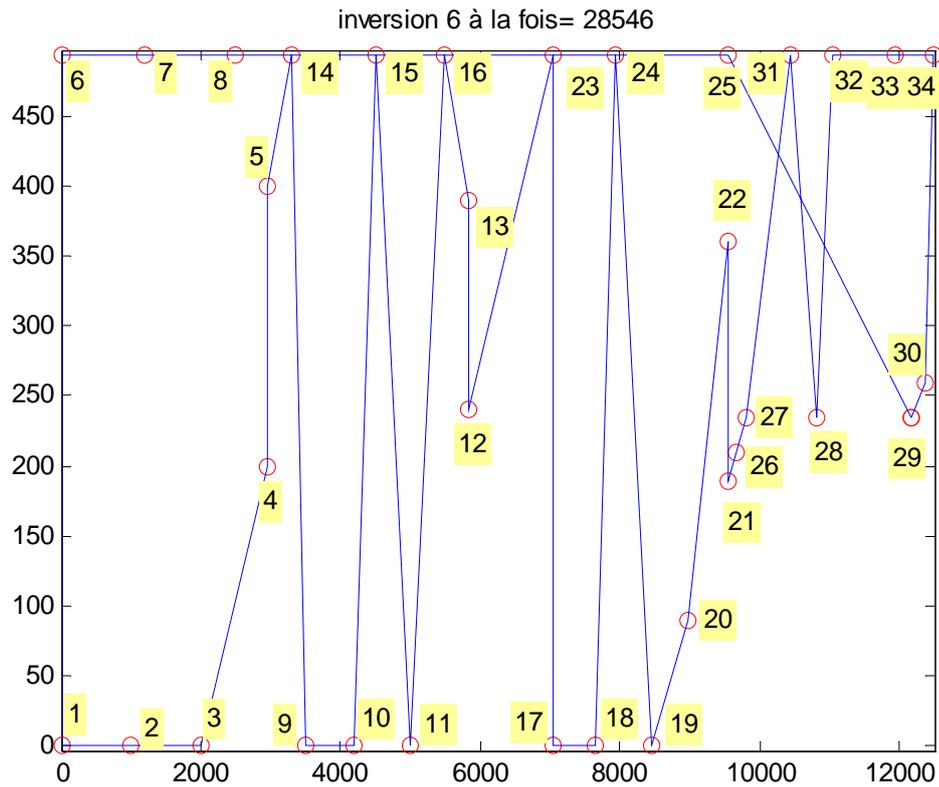


Fig. V.15 Subtour reversal heuristic (inversion six à la fois)

- **Inversion sept à la fois**

- ✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 28546\text{mm}$

- ✓ La séquence à inverser est (4 3 2 1 6 7 8)

- ✓ La distance après inversion est :  $d = 28361\text{mm}$

- ✓ Le taux d'amélioration est  $\Delta d = 185\text{mm}$

- ✓ Le chemin obtenu est le suivant :

**29 30 34 33 32 28 31 27 26 21 22 20 19 24 18 17 23 12 13 16**

**11 15 10 9 14 5 8 7 6 1 2 3 4 25 29**

- ✓ La simulation de l'inversion via matlab est comme indique la figure V.16.

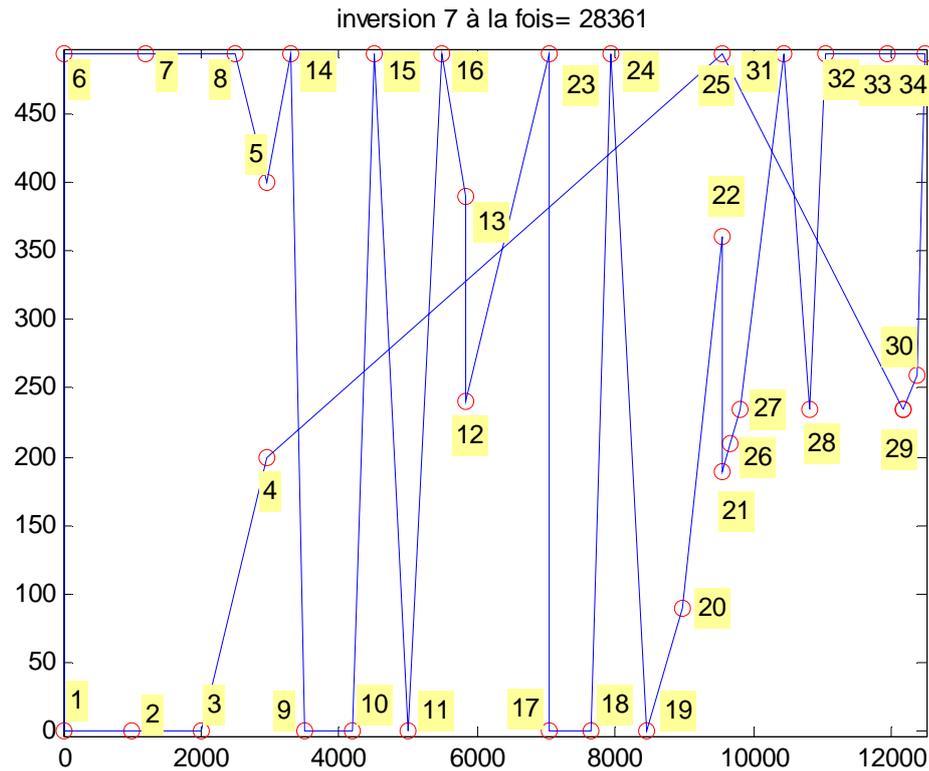


Fig. V.16 Subtour Reversal Heuristic (inversion sept à la fois)

- **Inversion neuf à la fois**

✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 28361mm$

✓ La séquence à inverser est (14 5 8 7 6 1 2 3 4)

✓ La distance après inversion est :  $d = 28058mm$

✓ Le taux d'amélioration est  $\Delta d = 303mm$

✓ Le chemin obtenu est le suivant :

**29 30 34 33 32 28 31 27 26 21 22 20 19 24 18 17 23 12 13 16**

**11 15 10 9 4 3 2 1 6 7 8 5 14 25 29**

✓ La simulation en matlab est illustrée dans la figure V.17

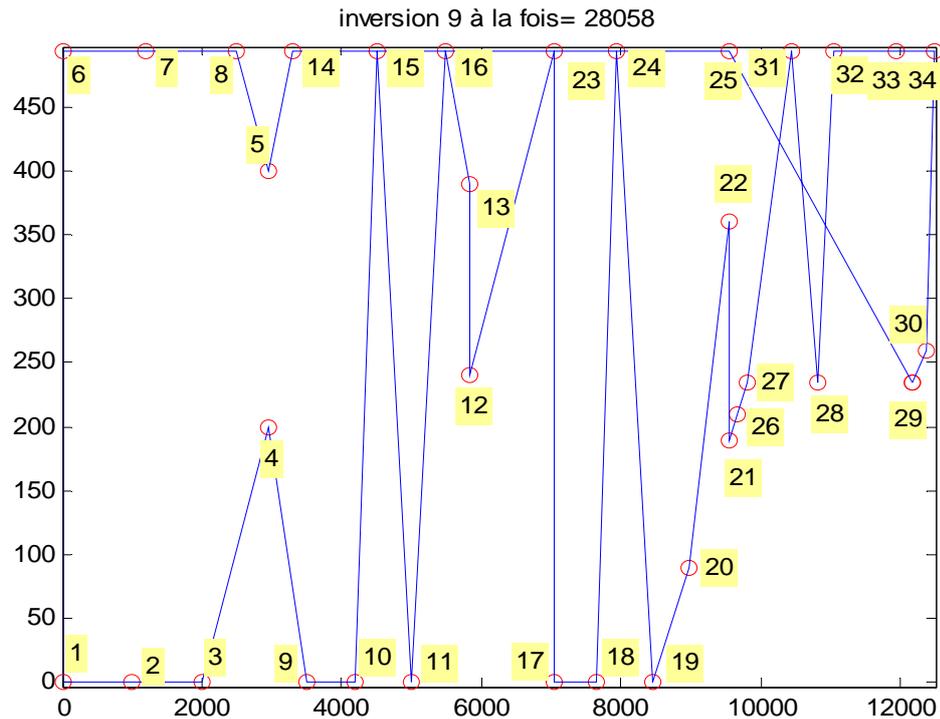


Fig. V.17 Subtour reversal heuristic (inversion neuf à la fois)

- **Inversion onze à la fois**

✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 28058mm$

✓ La séquence à inverser est **(10 9 4 3 2 1 6 7 8 5 14)**

✓ La distance après inversion est :  $d = 27803mm$

✓ Le taux d'amélioration est  $\Delta d = 255mm$

✓ Le chemin obtenu est le suivant :

**29 30 34 33 32 28 31 27 26 21 22 20 19 24 18 17 23 12 13 16**

**11 15 14 5 8 7 6 1 2 3 4 9 10 25 29**

✓ Cette inversion est représentée en matlab comme indique la figure suivante :

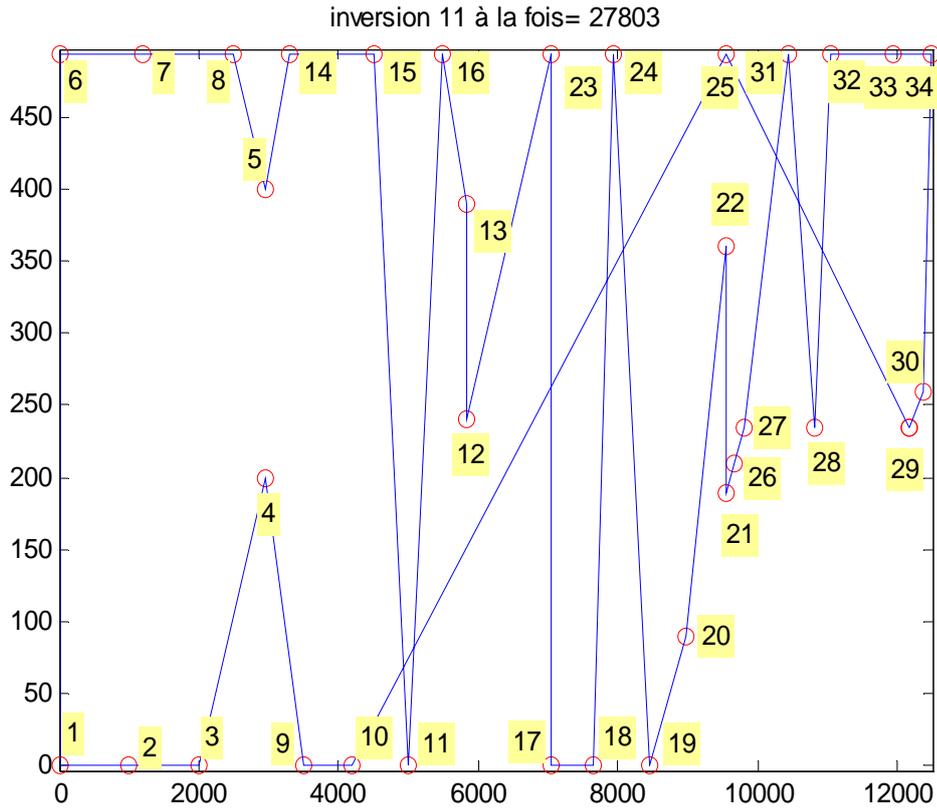


Fig. V.18 Subtour reversal heuristic (inversion onze à la fois)

• **Inversion douze à la fois**

- ✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 27803mm$
- ✓ La séquence à inverser est **(15 14 5 8 7 6 1 2 3 4 9 10)**
- ✓ La distance après inversion est :  $d = 27579mm$
- ✓ Le taux d'amélioration est  $\Delta d = 224mm$
- ✓ Le chemin obtenu est le suivant :

**29 30 34 33 32 28 31 27 26 21 22 20 19 24 18 17 23 12 13 16**  
**11 10 9 4 3 2 1 6 7 8 5 14 15 25 29**

- ✓ La figure V.19 est représentation en matlab de l'inversion douze à la fois.

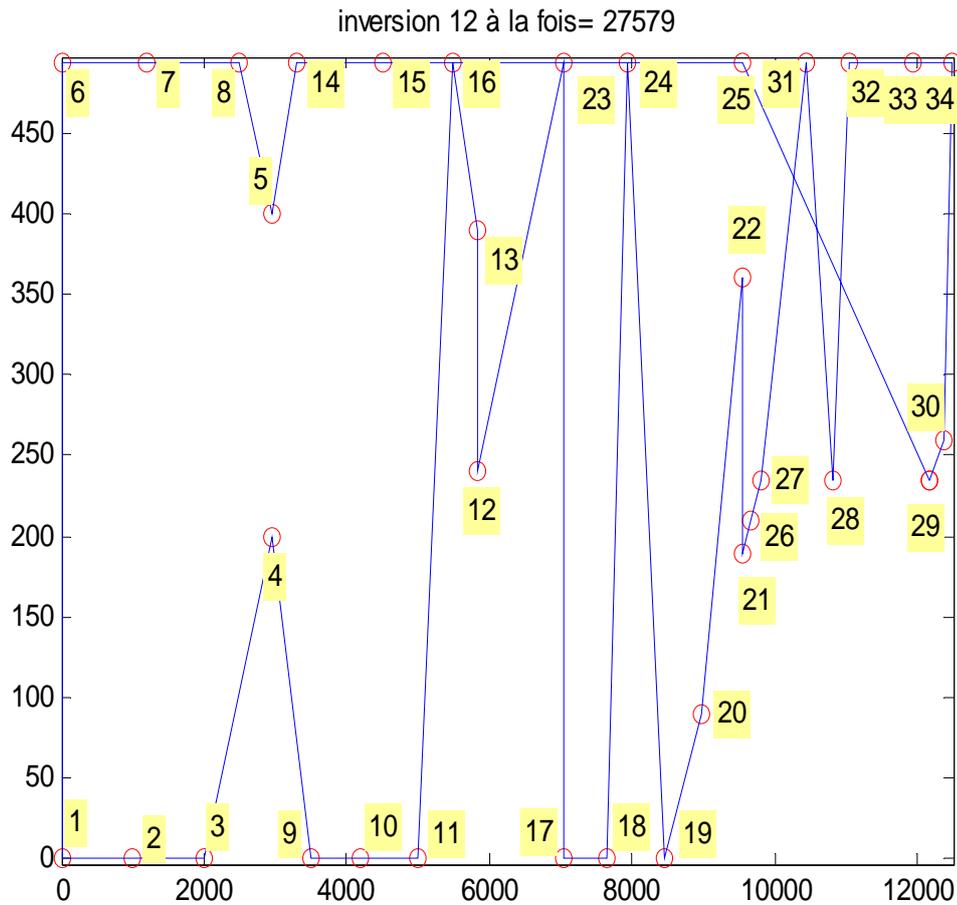


Fig. V.19 Subtour Reversal Heuristic (inversion douze à la fois)

- **Inversion treize à la fois**

✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 27579mm$

✓ La séquence à inverser est (11 10 9 4 3 2 1 6 7 8 5 14 15)

✓ La distance après inversion est :  $d = 27403mm$

✓ Le taux d'amélioration est  $\Delta d = 176mm$

✓ Le chemin obtenu est le suivant :

**29 30 34 33 32 28 31 27 26 21 22 20 19 24 18 17 23 12 13 16**

**15 14 5 8 7 6 1 2 3 4 9 10 11 25 29**

✓ la simulation par matlab pour cette inversion est donnée par la figure V.20.

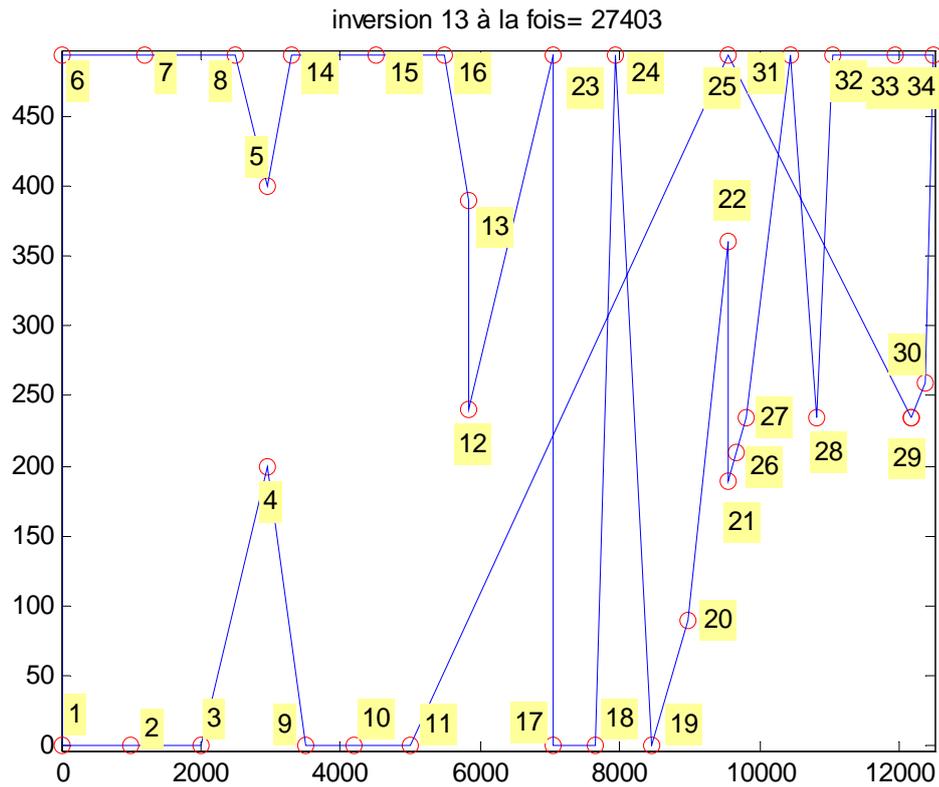


Fig. V.20 Subtour Reversal Heuristic (inversion treize à la fois)

• **Inversion quinze à la fois**

✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 27403mm$

✓ La séquence à inverser est (13 16 15 14 5 8 7 6 1 2 3 4 9 10 11)

✓ La distance après inversion est :  $d = 27261mm$

✓ Le taux d'amélioration est  $\Delta d = 142mm$

✓ Le chemin obtenu est le suivant :

**29 30 34 33 32 28 31 27 26 21 22 20 19 24 18 17 23 12 11 10 9**

**4 3 2 1 6 7 8 5 14 15 16 13 25 29**

✓ La figure V.21 est une représentation du chemin optimale via matlab de cette inversion

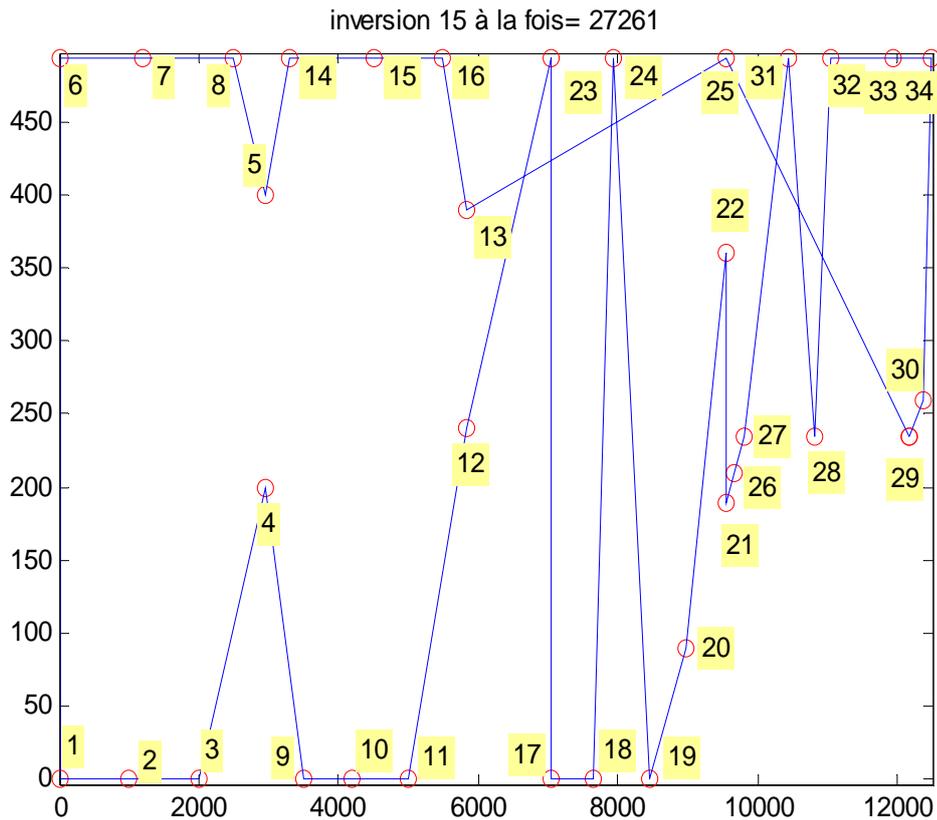


Fig. V.21 Subtour reversal heuristic (inversion quinze à la fois)

- **Inversion seize à la fois**

- ✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 27261mm$
- ✓ La séquence à inverser est (12 11 10 9 4 3 2 1 6 7 8 5 14 15 16 13)
- ✓ La distance après inversion est :  $d = 27246mm$
- ✓ Le taux d'amélioration est  $\Delta d = 15mm$
- ✓ Le chemin obtenu est le suivant :

**29 30 34 33 32 28 31 27 26 21 22 20 19 24 18 17 23 13 16 15**  
**14 5 8 7 6 1 2 3 4 9 10 11 12 25 29**

- ✓ Sous matlab l'inversion est donnée par la figure V.22 suivante :

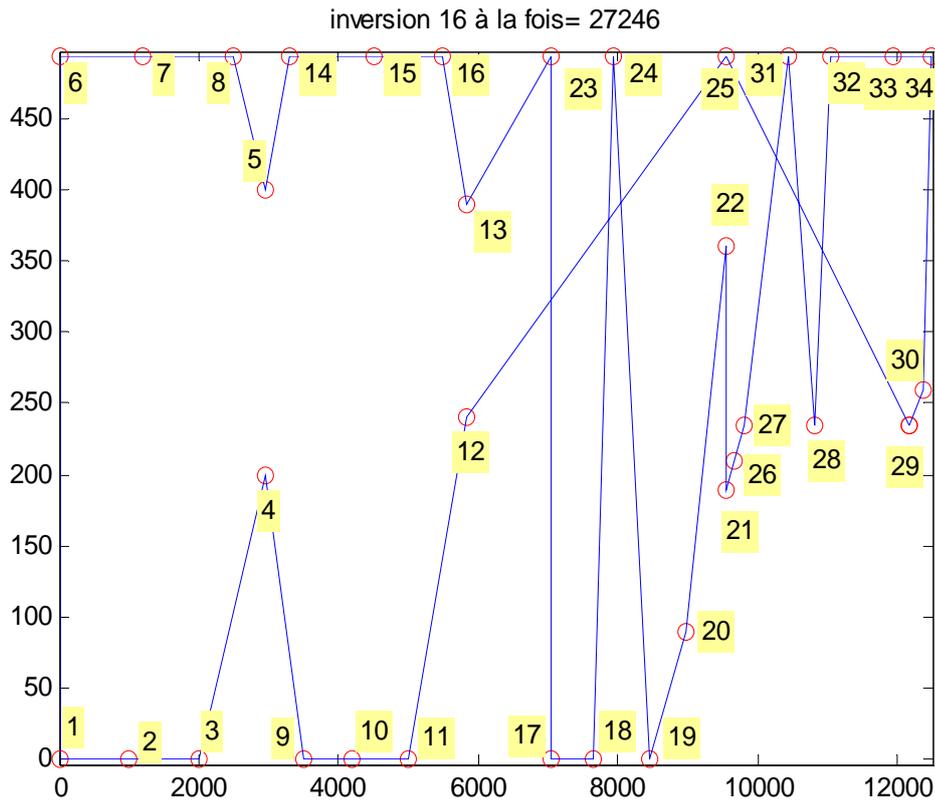


Fig. V.22 Subtour Reversal Heuristic (inversion seize à la fois)

- **Inversion dix sept à la fois**

- ✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 27246mm$
- ✓ La séquence à inverser est (23 13 16 15 14 5 8 7 6 1 2 3 4 9 10 11 12)
- ✓ La distance après inversion est :  $d = 26767mm$
- ✓ Le taux d'amélioration est  $\Delta d = 479mm$
- ✓ Le chemin obtenu est le suivant :

**29 30 34 33 32 28 31 27 26 21 22 20 19 24 18 17 23 13 16 15**

**14 5 8 7 6 1 2 3 4 9 10 11 12 25 29**

- ✓ On peut schématiser cette inversion par la figure V.23

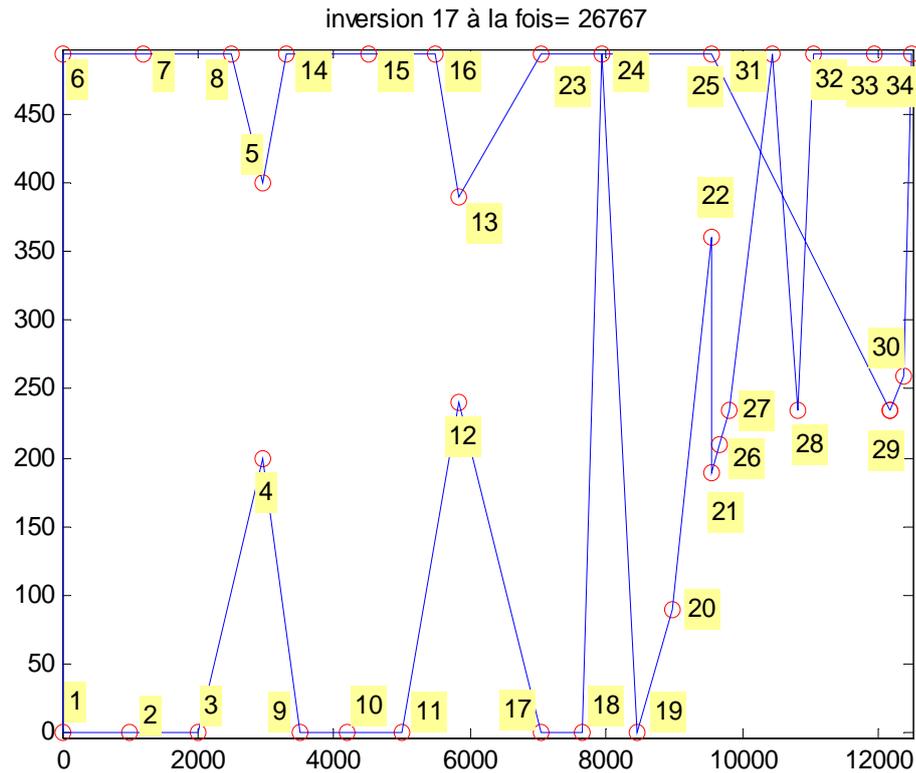


Fig. V.23 Subtour Reversal Heuristic (inversion dix sept à la fois)

- **Inversion dix neuf à la fois**

- ✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 26767mm$
- ✓ La séquence à inverser est (18 17 12 11 10 9 4 3 2 1 6 7 8 5 14 15 16 13 23)
- ✓ La distance après inversion est :  $d = 26553mm$
- ✓ Le taux d'amélioration est  $\Delta d = 214mm$
- ✓ Le chemin obtenu est le suivant :  
**29 30 34 33 32 28 31 27 26 21 22 20 19 24 23 13 16 15 14 5 8**  
**7 6 1 2 3 4 9 10 11 12 17 18 25 29**
- ✓ Le chemin de l'inversion 19 à la fois est indiqué par la figure V.24

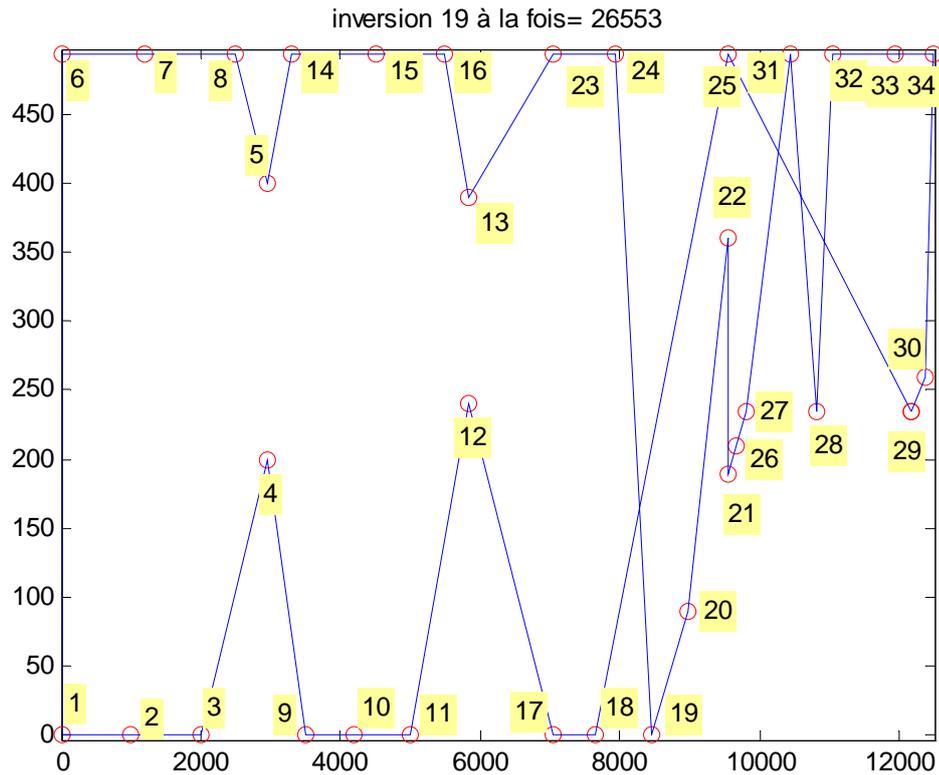


Fig. V.24 Subtour Reversal Heuristic (inversion dix neuf à la fois)

- **Inversion vingt à la fois**

✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 26553mm$

✓ La séquence à inverser est (24 23 13 16 15 14 5 8 7 6 1 2 3 4 9 10 11 12 17 18)

✓ La distance après inversion est :  $d = 26286mm$

✓ Le taux d'amélioration est  $\Delta d = 267mm$

✓ Le chemin obtenu est le suivant :

**29 30 34 33 32 28 31 27 26 21 22 20 19 18 17 12 11 10 9 4 3 2**

**1 6 7 8 5 14 15 16 13 23 24 25 29**

✓ La simulation par matlab est la suivante :

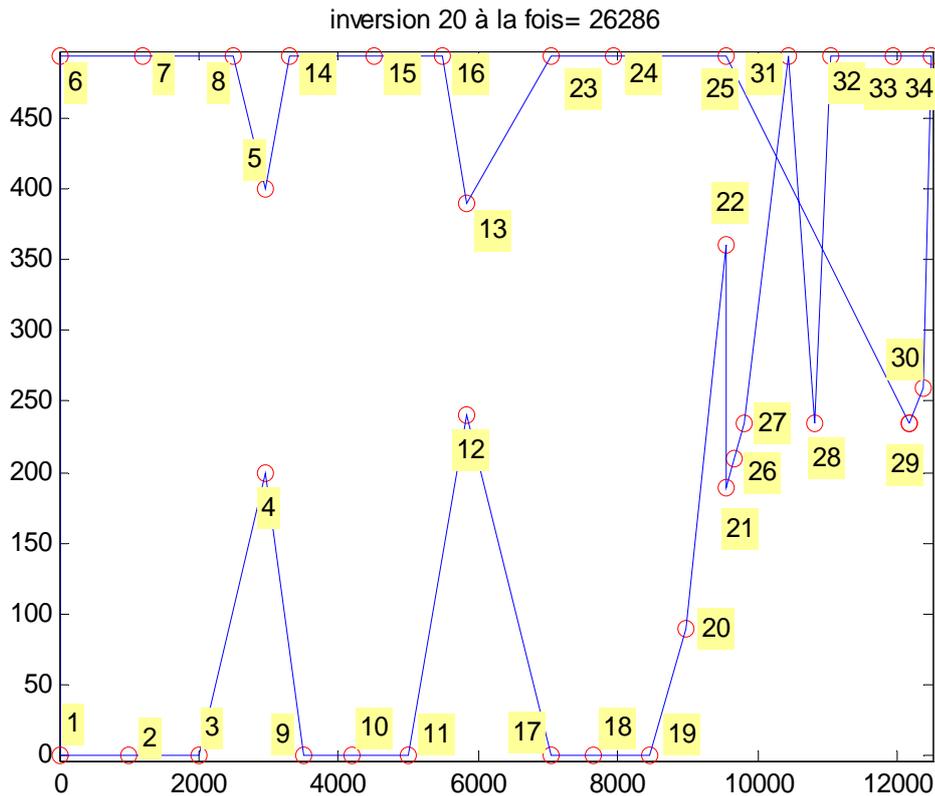


Fig. V.25 Subtour Reversal Heuristic (inversion vingt à la fois)

- **Inversion vingt sept à la fois**

- ✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 26286\text{mm}$
- ✓ La séquence à inverser est (27 26 21 22 20 19 18 17 12 11 10 9 4 3 2 1 6 7  
8 5 14 15 16 13 23 24 25)
- ✓ La distance après inversion est :  $d = 26223\text{mm}$
- ✓ Le taux d'amélioration est  $\Delta d = 63\text{mm}$
- ✓ Le chemin obtenu est le suivant :  
**29 30 34 33 32 28 31 25 24 23 13 16 15 14 5 8 7 6 1 2 3 4 9**  
**10 11 12 17 18 19 20 22 21 26 27 29**
- ✓ La figure V.26 indique le parcours optimal dans cette inversion

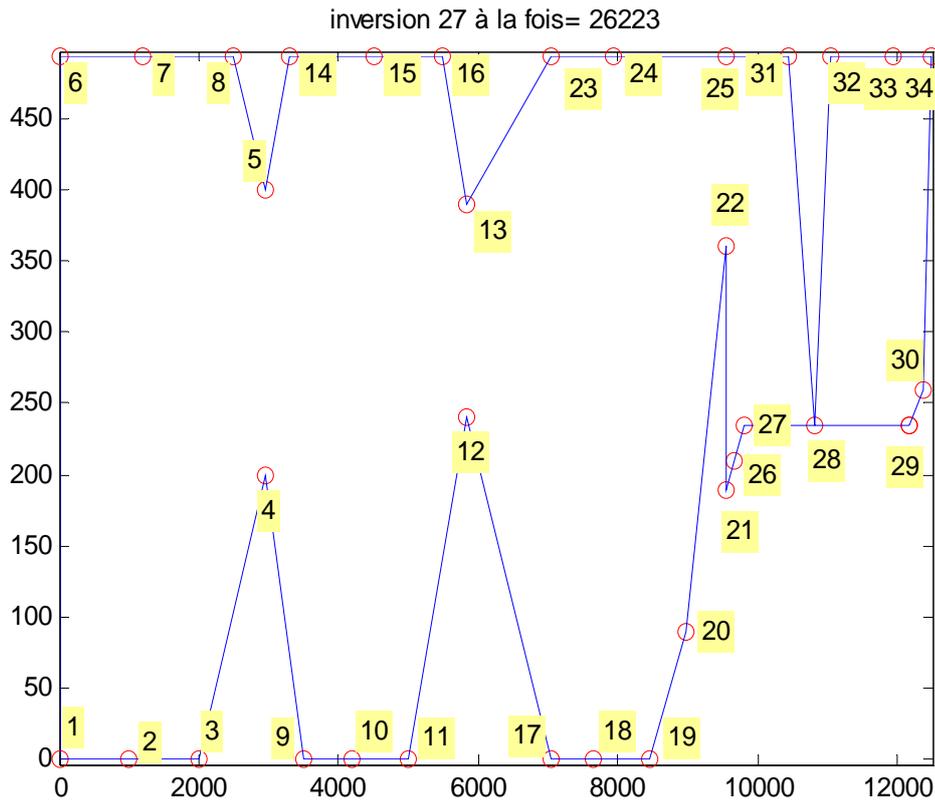


Fig. V.26 Subtour Reversal Heuristic (inversion vingt sept à la fois)

- **Inversion vingt huit à la fois**

- ✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 26223mm$

- ✓ La séquence à inverser est (31 25 24 23 13 16 15 14 5 8 7 6 1 2 3 4 9 10 11 12 17 18 19 20 22 21 26 27)

- ✓ La distance après inversion est :  $d = 26161mm$

- ✓ Le taux d'amélioration est  $\Delta d = 62mm$

- ✓ Le chemin obtenu est le suivant :

**29 30 34 33 32 28 27 26 21 22 20 19 18 17 12 11 10 9 4 3 2 1**

**6 7 8 5 14 15 16 13 23 24 25 31 29**

- ✓ La figure suivante est une simulation par matlab de l'inversion

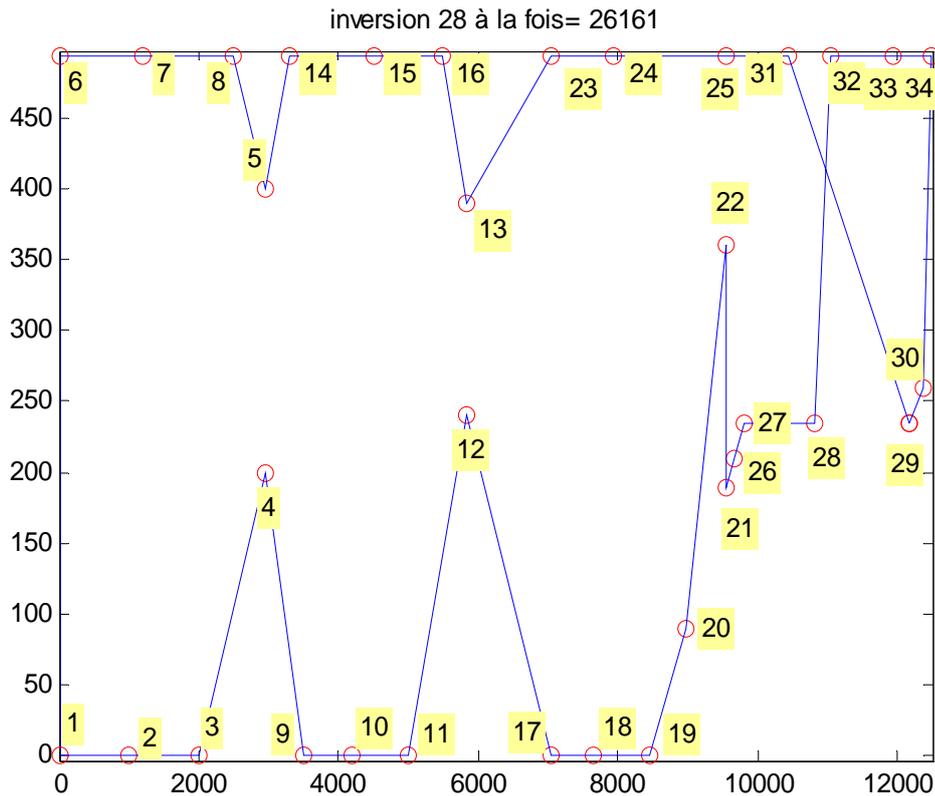


Fig. V.27 Subtour Reversal Heuristic (inversion vingt huit à la fois)

- **Inversion vingt neuf à la fois**

✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 26161mm$

✓ La séquence à inverser est (28 27 26 21 22 20 19 18 17 12 11 10 9 4 3 2 1

6 7 8 5 14 15 16 13 23 24 25 31)

✓ La distance après inversion est :  $d = 26026mm$

✓ Le taux d'amélioration est  $\Delta d = 135mm$

✓ Le chemin obtenu est le suivant :

**29 30 34 33 32 31 25 24 23 13 16 15 14 5 8 7 6 1 2 3 4 9 10**

**11 12 17 18 19 20 22 21 26 27 28 29**

✓ La figure V.28 est le chemin optimal de l'inversion 29 à la fois.

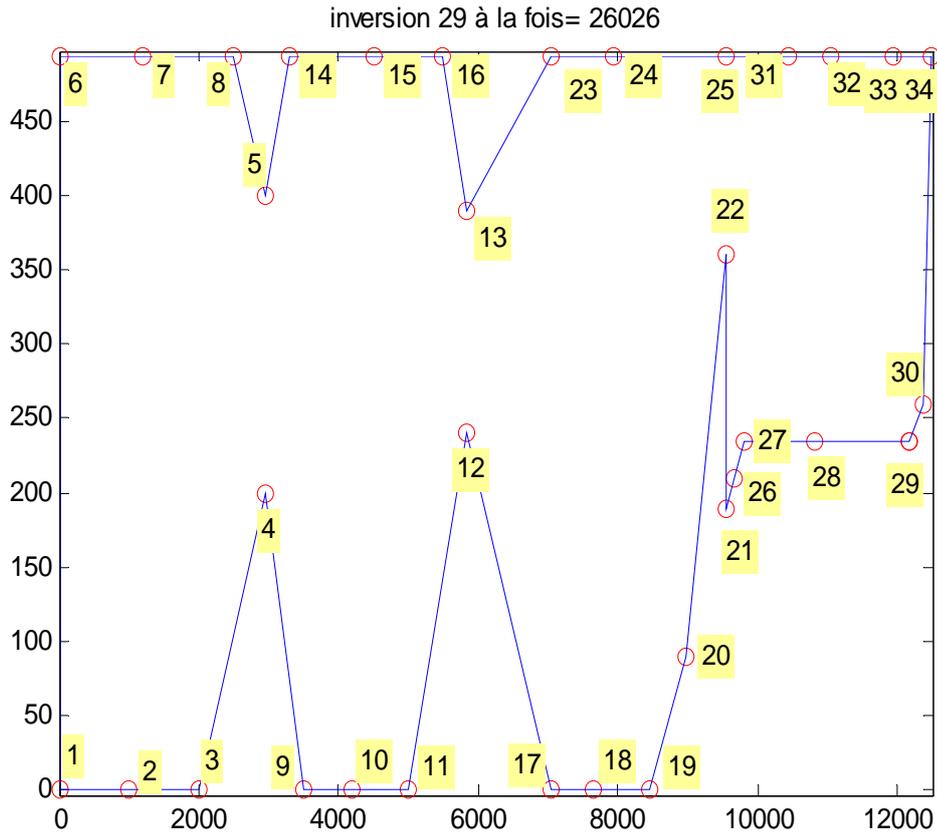


Fig. V.28 Subtour Reversal Heuristic (inversion vingt neuf à la fois)

• **Inversion trente trois à la fois**

- ✓ La distance initiale  $d$  est celle de l'itération précédente :  $d = 26026mm$
- ✓ La séquence à inverser est **(30 34 33 32 31 25 24 23 13 16 15 14 5 8 7 6 1 2 3 4 9 10 11 12 17 18 19 20 22 21 26 27 28 )**

- ✓ La distance après inversion est :  $d = 26026mm$
- ✓ Le taux d'amélioration est  $\Delta d = 0mm$  , pas d'amélioration dans cette étape
- ✓ Le chemin obtenu est le suivant :

**29 28 27 26 21 22 20 19 18 17 12 11 10 9 4 3 2 1 6 7 8 5 14**  
**15 16 13 23 24 25 31 32 33 34 30 29**

✓ Le chemin optimal est montré dans la figure V.29

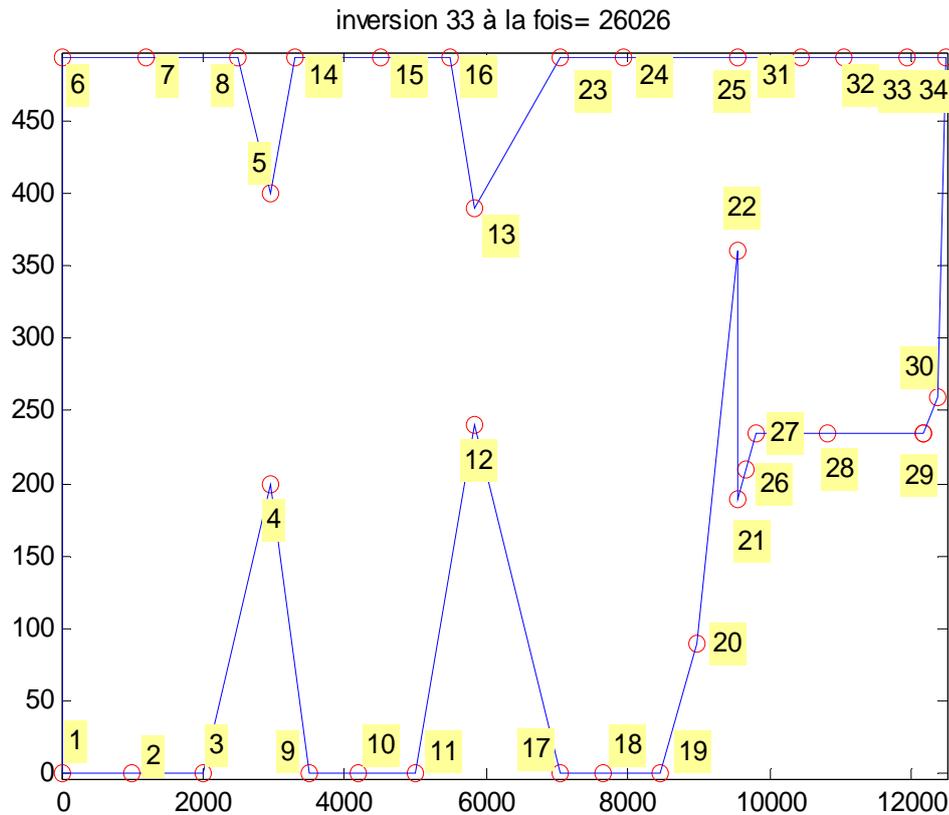


Fig. V.29 Subtour Reversal Heuristic (inversion trente trois à la fois)

On peut maintenant faire une consolidation des résultats obtenus par l'application de subtour reversal heuristic afin de tracer la courbe qui détermine l'évolution de notre problème de minimisation.

Le taux d'amélioration est : la différence entre la valeur d'une inversion de la méthode subtour reversal heuristic par rapport à la valeur obtenue par l'application de la méthode nearest neighbor comme suit :

$$\text{Taux d'amélioration} = \text{distance après inversion} - 29332$$

Ce qu'indique le tableau V.2 suivant :

<b>N° itération</b>	<b>N° inversion</b>	<b>Distance (mm)</b>	<b>Taux d'amélioration (mm)</b>
<b>1</b>	<b>2</b>	<b>29286</b>	<b>46</b>
<b>2</b>	<b>3</b>	<b>28268</b>	<b>1064</b>
<b>3</b>	<b>4</b>	<b>28814</b>	<b>518</b>
<b>4</b>	<b>5</b>	<b>28628</b>	<b>704</b>
<b>5</b>	<b>6</b>	<b>28546</b>	<b>786</b>
<b>6</b>	<b>7</b>	<b>28361</b>	<b>971</b>
<b>7</b>	<b>9</b>	<b>28058</b>	<b>1274</b>
<b>8</b>	<b>11</b>	<b>27803</b>	<b>1529</b>
<b>9</b>	<b>12</b>	<b>27579</b>	<b>1753</b>
<b>10</b>	<b>13</b>	<b>27403</b>	<b>1929</b>
<b>11</b>	<b>15</b>	<b>27261</b>	<b>2071</b>
<b>12</b>	<b>16</b>	<b>27246</b>	<b>2086</b>
<b>13</b>	<b>17</b>	<b>26767</b>	<b>2565</b>
<b>14</b>	<b>19</b>	<b>26553</b>	<b>2779</b>
<b>15</b>	<b>20</b>	<b>26286</b>	<b>3046</b>
<b>16</b>	<b>27</b>	<b>26223</b>	<b>3109</b>
<b>17</b>	<b>28</b>	<b>26161</b>	<b>3171</b>
<b>18</b>	<b>29</b>	<b>26026</b>	<b>3306</b>
<b>19</b>	<b>33</b>	<b>26026</b>	<b>3306</b>

Tableau V.2 variation des améliorations subtour reversal heuristic

La courbe de la figure suivante représente l'évolution des résultats de la méthode Subtour Reversal Heuristique par rapport à Nearest Neighbor.

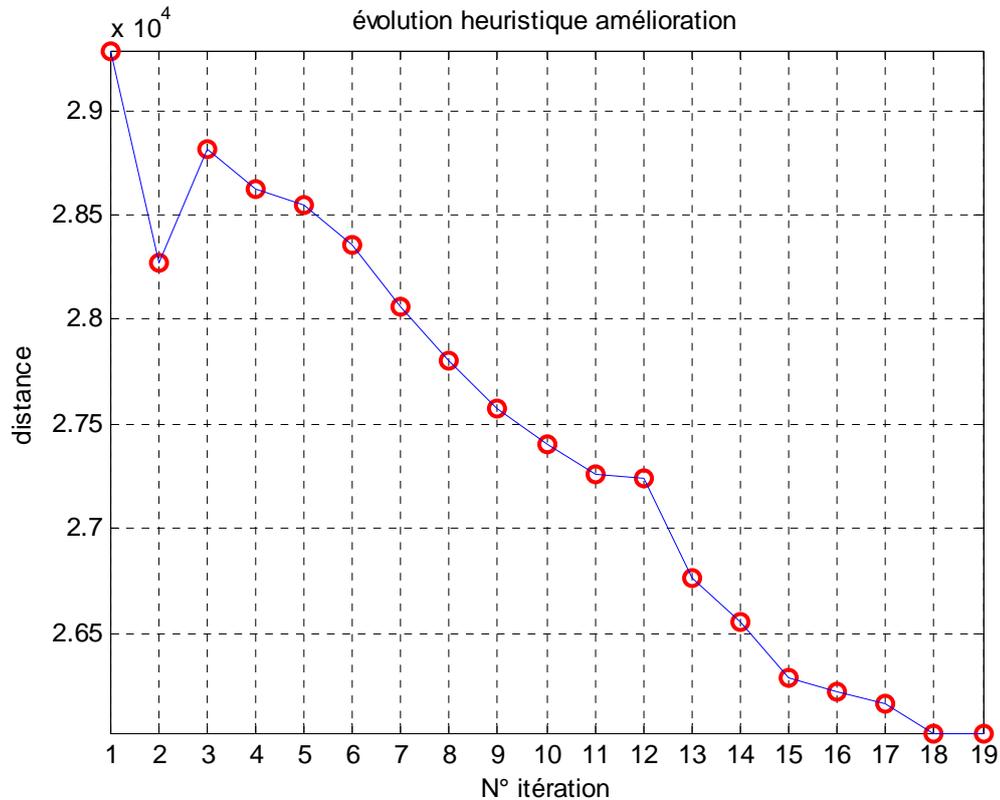


Fig. V.30 Amélioration par la méthode Subtour Reversal Heuristic.

## V. 7. Proposition d'un robot de soudage

Afin de d'automatiser la tâche de soudage manuelle effectuée sur le longeron de la semi-remorque fabriquée par TIRSAM, Le robot proposé est KUKA KR210 VKRC2 CONTROL ayant les caractéristiques suivantes :

- \* Charge maximale du robot: 150/180/210Kg.
- \* Etendue maximale: 2700/2900/3100 mm.DONC ~ 3m et 10cm
- \* Répétabilité: 0.15 0.2 mm.
- \* Contrôleur: VKR C2 logiciel 3.3.4
- \* Vitesse de mouvement du robot:
  - ◆ Axe 1: 105°/seg
  - ◆ Axe 2: 110°/seg
  - ◆ Axe 3: 95°/seg
  - ◆ Axe 4: 170°/seg
  - ◆ Axe 5: 170°/seg

- ◆ Axe 6: 238°/seg

Les différentes applications du robot:

- Usinage
- Polissage, meulage et d'ébavurage
- Revêtement et collage
- Manipulation
- Mesure et essai
- Assemblage, Palettisation, Coupure, etc.

### V.7.1 Motivation du choix de ce robot

Ce type de robots effectue un soudage à l'arc et le soudage à l'arc est le procédé le plus utilisé en construction métallique.

En plus, Ces robot possèdent une étendue maximale convenable (3100mm). Dans ce cas le robot effectue quatre déplacements pour mettre les points de soudage sur les extrémités des tôles du longeron car un déplacement unique n'est pratique car le robot effectuant un seul déplacement nécessite plus de 12500mm d'étendue (longueur du longeron).

### V.7.2 Résultats obtenus lors de l'utilisation du robot proposé

Dans ce cas, on divise notre matrice globale des distances en quatre sous matrices également symétriques, la simulation en utilisant le logiciel matlab donne les résultats suivants :

#### V.7.2.a Nearest neighbor

Les résultats lors de l'application de NN aux différentes parties du longeron sont présentés dans le tableau V.3

N° Partie	Distance NN (mm)	Le Chemin NN Optimal
<b>1</b>	<b>7105</b>	<b>8 5 4 3 7 2 1 6 8</b>
<b>2</b>	<b>11969</b>	<b>3 7 2 1 6 8 5 4 3</b>
<b>3</b>	<b>6175</b>	<b>3 4 5 6 9 8 2 1 7 3</b>
<b>4</b>	<b>4342</b>	<b>4 5 8 9 1 2 6 3 7 4</b>

Tableau V.3 Résultats NN pour les quatre parties du longeron

### V.7.2.b Subtour Reversal Heuristic

L'amélioration des résultats par application de subtour reversal heuristic est donnée dans les tableaux suivants:

<b>Partie</b>	<b>3 à la fois (mm)</b>	<b>4 à la fois (mm)</b>	<b>7 à la fois (mm)</b>	<b>8 à la fois (mm)</b>
<b>1</b>	<b>6852</b>	<b>6633</b>	<b>6633</b>	/
<b>2</b>	<b>11782</b>	<b>11490</b>	<b>11490</b>	/
<b>3</b>	<b>5814</b>	<b>Pas d'amélioration</b>	<b>Pas d'amélioration</b>	<b>5814</b>
<b>4</b>	<b>Pas d'amélioration</b>	<b>Pas d'amélioration</b>	<b>Pas d'amélioration</b>	<b>4342</b>

Tableau V.4 Résultats RSH pour les quatre parties du longeron

<b>N° Partie</b>	<b>Distance SRH (mm)</b>	<b>Le Chemin SRH Optimal</b>
<b>1</b>	<b>6633</b>	<b>8 5 4 3 2 1 6 7 8 (4 à la fois) 8 7 6 1 2 3 4 5 8 (7 à la fois)</b>
<b>2</b>	<b>11490</b>	<b>3 2 1 6 7 8 5 4 3 (4 à la fois) 3 4 5 8 7 6 1 2 3 (7 à la fois)</b>
<b>3</b>	<b>5814</b>	<b>3 4 5 6 9 8 7 1 2 3 (3 à la fois) 3 2 1 7 8 9 6 5 4 3 (8 à la fois)</b>
<b>4</b>	<b>4342</b>	<b>4 7 3 6 2 1 9 8 5 4 (8 à la fois)</b>

Tableau V.5 Chemins RSH des quatre parties du longeron

## **V. 8. Conclusion**

Dans ce chapitre nous avons exposé les résultats de simulation effectuée sous l'environnement MATLAB, afin de tester les performances des approches appliquées sur un système de production pour déterminer un tour caractérisé par une distance minimale pour placer les point de soudage afin de fixer les tôles d'acier les unes avec les autres .

Les différentes simulations effectuées montrent que l'approche heuristique adoptée permet l'obtention des résultats performants et plus particulièrement l'algorithme de Subtour Reversal Heuristic qui permet l'amélioration de résultat de l'algorithme de Nearest Neighbor à chaque itération de son application. Ce qui nous permet de déduire l'efficacité de ces algorithmes.

## **Conclusion générale**

Le travail présenté concerne deux axes de recherche très intéressants : le premier est lié aux actualités de la robotique, et le deuxième est relatif à l'application des algorithmes d'optimisation à un problème appartenant à ce domaine.

Les robots industriels sont introduits dans des champs divers d'application tels que la manipulation des pièces, et les outils, l'assemblage, la peinture, le soudage, etc. Ce qui rend les tâches et les opérations dangereuses (dans les milieux: hostiles, à haute température, forte radiations, etc.) possibles.

Plusieurs concepts liés à l'identification et la modélisation mathématique de ces entités sont présentés au sein des deux premiers chapitres.

L'aspect comportemental des robots lié au mouvement est aussi présenté dans ce travail, différentes stratégies de mouvement sont exposées dans le troisième chapitre.

On a aussi appliqué deux méthodes heuristiques d'optimisation présentées dans le quatrième chapitre grâce aux avantages (rapidité et moins de complexité) qu'elles présentent et aux lacunes (en matière de complexité et d'espace mémoire nécessaires) des méthodes déterministes. Nous avons appliqué dans un premier temps une heuristique constructive pour avoir une solution optimale au problème présenté à savoir nearest neighbor. On a également appliqué un algorithme heuristique mélioratif au résultat de la première méthode dans la deuxième étape. Cette dernière a un apport très important et non négligeable sur la performance des résultats.

La contribution essentielle de ce mémoire est de déterminer un tour avec une distance minimale sur le longeron pour réaliser tous les points de soudage et ensuite la proposition d'un robot afin d'automatiser cette tâche au niveau de la société TIRSAM pour objectif d'améliorer la production des semi-remorques et puis l'augmentation de la quantité produite.

Comme perspectives pour un travail de recherche futur, on suggère que :

- ✓ Si ce travail est appliqué aux objets plats (à deux dimensions 2D), on propose la manipulation des objets (pièces) ayant trois dimensions (3D).
- ✓ On propose aussi l'utilisation des approches évolutionnaires et les colonies de fourmis pour résoudre ce problème.

## REFERENCES BIBLIOGRAPHIQUES

1. C. JOSSIN. *Buts de l'automatisation*, [http://lycees.ac-rouen.fr/modeste-leroy/spip/IMG/pdf/Buts\\_de\\_l\\_automatisme.pdf](http://lycees.ac-rouen.fr/modeste-leroy/spip/IMG/pdf/Buts_de_l_automatisme.pdf), consulté le 02/01/2013.
2. M CAUSSADE. *Cours d'automatisme*, <http://mei-csi.wifeo.com/documents/automatisme-prof-n1.pdf>, consulté le 05/01/2013
3. PHILIPPE COIFFET. *La Robotique, principes et applications*. 3<sup>ème</sup> Edition. Hermès, Paris, 1992, 353p.
4. IUP TASV, *Robotique*, 2004-2005, [http://icube-avr.unistra.fr/fr/img\\_auth.php/a/a4/Cours\\_rob\\_intro.pdf](http://icube-avr.unistra.fr/fr/img_auth.php/a/a4/Cours_rob_intro.pdf), consulté le 10/02/2013.
5. PIERRE DUYSINX, *Eléments de robotique : aspects mécaniques*. Département ROMETHE, (Productique, Mécanique et Thermodynamique)
6. D. DANEY INRIA, *Cours de robotique fondamentale*, Projet Coprin INRIA Sophia Antipolis
7. PHILIPPE POIGNET, GERARD POISSON, *La conception de robots pour l'assistance aux gestes médicaux et chirurgicaux*, Laboratoire d'Informatique, de Robotique et de Microélectronique de Montpellier.
8. MICHEL FAYOLLE, *Aspects mécaniques en robotique*, ENSTA
9. L.P. AULNOYE. *Mécanique appliquée : énergétique - Actionneurs*
10. T. AL ANI. *Introduction à la robotique*. Département Informatique - ESIEE, Paris.
11. CATHERINE OPTTEL, PHILIPPE GATIGNOL, *Université de Maine-UFR sciences et techniques, le mans*
12. JEAN-LOUIS BOIMOND, *robotique*, Université Angers. [http://lars.mec.ua.pt/public/LAR%20Projects/Humanoid/2011\\_RicardoGodinho/Pesquisa%20Disserta%C3%A7%C3%A3o/Cours\\_robotique.pdf](http://lars.mec.ua.pt/public/LAR%20Projects/Humanoid/2011_RicardoGodinho/Pesquisa%20Disserta%C3%A7%C3%A3o/Cours_robotique.pdf), consulté le 21/12/2012.
13. JEAN MICHEL OLIVE, *Architecture mécanique des robots*, Marseille, 2000.
14. CYRIL QUENNOUELLE, *Modélisation Géométrico-Statique Des Mécanismes Parallèles Complaints*, université Laval Québec : thèse de doctorat, 2009.
15. MAZEN ZEIN, *Analyse Cinématique Des Manipulateurs Sériels 3R Orthogonaux Et Des Manipulateurs Parallèles Plans*, Université de Nantes : thèse de doctorat, 2007.

16. PLÉDEL PATRICK. *Génération de mouvements optimaux pour un robot manipulateur*, université de Nantes : thèse de doctorat, 1996.
17. SAVESCU (STAIKU) ADRIANA VIOLETA, *Analyse Et Modélisation Du Mouvement De Préhension*, université Claude Bernard - Lyon 1 : thèse de doctorat, 2006.
18. SAMIR LAHOUAR, *Contribution A La Modélisation Et A La Planification De Trajectoires De Robots Parallèles Evitant Les Singularités Et Les Collisions*, Université de Poitiers - UP (FRANCE) ,2008.
19. ALIA KAMEL, *Planificateur De Trajectoire Neuronal*, Université El-Hadj Lakhdar- Batna, Thèse De Magistère, 2011.
20. DIANE BOROJENI, *modélisation cinématique et dynamique des systèmes poly-articulé a chaine ouverte ou fermée.cas des robots parallèles*, université paris XII, VAL DE MARNE : thèse de doctorat, 2006.
21. GENTIANE VENTURE, *identification des paramètres dynamiques d'une voiture*, Université de Nantes : thèse de doctorat, 2003.
22. FLORIAN GOSSELIN, Jean-Paul LALLEMAND, XVème Congrès Français de Mécanique, Nancy, 3 – 7 Septembre 2001:*Contribution A L'étude De La Dualité Entre Les Robots Série Et Les Robots Parallèles*.
23. R K MITTAL, I J NAGRATH, *robotics and control*, 6<sup>eme</sup> edition, 2007, p247,248,249,
24. SAEED B. NIKU, *introduction to robotics, Analysis, Control, Applications*, 2<sup>e</sup> edition, 2010
25. LIGUI BIAGIATTO, CLAUDIO MELCHIORRI, *Trajectory Planning For Automatic Machines and Robots*, 2008.
26. SONJA MACFARLANE, *On-Line Smooth Trajectory Planning for Manipulators*, université COLOMBIE-BRITANNIQUE, 2001.
27. [http://teacher.buet.ac.bd/bmsirajeel/ChE310/Engineering\\_Matlab\\_Programming.pdf](http://teacher.buet.ac.bd/bmsirajeel/ChE310/Engineering_Matlab_Programming.pdf)
28. <http://ebookbrowse.com/rapport m1 2010 voyageur commerce.pdf>
29. LENORE COWEN, DERRICK RICE ,*The Travelling Salesman Problem*, Tufts University, Spring 2009.
30. MAHDI SAMIR, *Optimisation Multiobjectif Par Un Nouveau Schéma De Coopération Méta/Exacte*.

31. TADUNFOCK TETI BERNARD, FOTSO Laure Pauline, "Heuristiques du problème du voyageur de commerce", Université de Yaoundé I, 2006.
32. PIERRE-ANDRE ZITT, « Applications d'inégalités fonctionnelles à la mécanique statistique et au recuit simulé », UNIVERSITÉ PARIS X – NANTERRE, thèse de doctorat, 2006.
33. RYAN KAMMARTI, «Approches évolutionnistes pour la résolution du 1-PDPTW statique et dynamique », Université des sciences et technologies de Lille, thèse de doctorat, 2006.
34. CHRISTIAN NILSSON, « Heuristics for the Travelling Salesman Problem », Université Linköping
35. KYLE LOUDON, « Mastering Algorithms with C », les états unis américaines, Aout 1999, 560p.
36. HAMDY A.TAHA , «operations research: An introduction»,8<sup>e</sup> edition, université d'Arkansas-fayetteville ,2007.
37. JEAN-PIERRE PLANCKAERT, «Modélisation du soudage MIG/MAG en mode short-arc», Université Henri Poincaré, Nancy 1, thèse de doctorat,2008.
38. CHAINARONG SRIKUNWONG, Modélisation Du Procédé De Soudage Par Points », these de doctorat ,2005.
39. RAVO RAMANANTSOA," Caractérisation Expérimentale D'un Plasma D'arc Et De Son Interaction Avec Les Electrodes ", these de doctorat;2011.
40. D.B MARGHITU, Mechanisms and Robots Anatsysis With Matlab, Britich library Cataloguing in publication Data, Springer-verlag London Limited, 2009.
41. [www.tirsam.com](http://www.tirsam.com)