
Une approche génétique pour la résolution du problème VRPTW dynamique

THÈSE

présentée et soutenue publiquement le 03 mai 2005

en vue de l'obtention du

Doctorat de l'Université d'Artois

(Spécialité Informatique)

par

HOUSROUM HAIYAN

Composition du jury

| | | |
|----------------------|----------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Président :</i> | Anne-Marie JOLLY-DESODT | Professeur à l'ENSAIT |
| <i>Rapporteurs :</i> | Frédéric SEMET Christian PRINS | Professeur à l'Université de Valenciennes Professeur à l'Université de Technologie de Troyes |
| <i>Examineurs :</i> | Gilles GONCALVES Daniel JOLLY Slim HAMMADI Rémy DUPAS | Professeur à l'Université d'Artois (Directeur de thèse) Professeur à l'Université d'Artois (Directeur du laboratoire) Professeur à l'Ecole Centrale de Lille Maître de conférence H.D.R. à l'Université d'Artois (Co-directeur de thèse) |

Mis en page avec la classe thloria.

Remerciements

Cette thèse n'aurait vu le jour sans la confiance, la patience et la générosité de mon directeur de thèse, monsieur **Gilles Goncalves**, que je veux vivement remercier. Je voudrais aussi le remercier pour le temps et la patience qu'il m'a accordé tout au long de ces années. Les conseils et les aides qu'il m'a divulgués tout au long de la rédaction, ont permis la production de cette thèse.

Je remercie particulièrement monsieur **Daniel Jolly** directeur du LGI2A, de m'avoir accueilli dans son laboratoire de recherche.

Je remercie monsieur le professeur **Frédéric Semet** de l'Université de Valenciennes et monsieur le professeur **Christian Prins** de l'Université de Technologie de Troyes, de me faire l'honneur d'être rapporteurs de ce mémoire.

Je tiens également à remercier madame le professeur **Anne-Marie Jolly-Desodt** de l'ENSAIT, qui a accepté de présider le jury de cette thèse. ‘

Mes remerciements vont également à monsieur **Slim Hammadi** de l'Ecole Centrale de Lille qui a bien voulu examiné ce travail.

Mes plus sincères remerciements vont également à monsieur **Rémy DUPAS**, qui en agissant au titre de co-directeur a fortement enrichi ma formation. Ses conseils et ses commentaires auront été fort utiles.

Je tiens à remercier tout particulièrement monsieur **Tienté HSU** pour son intuition et ses nombreux conseils. Je souhaite également remercier tous les membres de LGI2A.

Merci enfin à toute ma famille, en particulier **Abir**, de m'avoir supporté et aidé pendant ces dernières années.

À Abir ...,

Table des matières

| | |
|---------------------------------------------------------------------------|----------|
| Table d'abréviations | 1 |
| Introduction générale | 3 |
| Chapitre 1 Le problème d'élaboration de tournées de véhicules CVRP | 7 |
| 1.1 Introduction | 7 |
| 1.2 Formulation | 8 |
| 1.2.1 Détail des équations | 10 |
| 1.3 Complexité | 10 |
| 1.4 Variantes du problème VRP | 10 |
| 1.5 Les méthodes de résolution du VRP | 12 |
| 1.5.1 Les méthodes exactes | 13 |
| 1.5.1.1 Les méthodes de recherche arborescente (Branch & Bound) . . . | 13 |
| 1.5.1.2 Programmation Linéaire en nombres entiers (Branch & Cut) . . | 14 |
| 1.5.1.3 Programmation Dynamique | 14 |
| 1.5.2 Les heuristiques | 14 |
| 1.5.2.1 Algorithme de gain (Clarke & Wright) | 15 |
| 1.5.2.2 Méthode d'insertion séquentielle | 15 |
| 1.5.2.3 Algorithme groupe en premier, route en second | 15 |
| 1.5.2.4 Algorithme route en premier, groupe en second | 16 |
| 1.5.2.5 Algorithme de recherche locale | 16 |
| 1.5.3 Les métaheuristiques | 17 |
| 1.5.3.1 Les algorithmes génétiques | 17 |
| Codage | 19 |
| Opérateurs de reproduction | 19 |
| 1.5.3.2 Recuit simulé | 20 |
| 1.5.3.3 La méthode de recherche tabou | 22 |
| 1.5.3.4 La recherche par dispersion | 23 |
| 1.5.3.5 La colonie de fourmis | 24 |

| | | |
|---------|-----------------------------------------|----|
| 1.6 | Le problème VRPTW | 26 |
| 1.6.1 | La formulation du VRPTW | 27 |
| 1.6.2 | Revue de la littérature VRPTW | 28 |
| 1.6.2.1 | Méthodes exactes | 28 |
| 1.6.2.2 | Méthodes approchées | 29 |
| 1.7 | Conclusion | 30 |

Chapitre 2 Le problème d’élaboration dynamique de tournées de véhicules

| | | |
|-------------|-----------------------------------------------------------------------|-----------|
| DVRP | | 33 |
| 2.1 | Introduction | 33 |
| 2.2 | Le problème d’élaboration de tournées de véhicule dynamique | 35 |
| 2.3 | Classification du VRP | 35 |
| 2.3.1 | VRP statique | 37 |
| 2.3.2 | VRP dynamique | 37 |
| 2.4 | Problème statique vs. problème dynamique | 38 |
| 2.5 | Formulation | 39 |
| 2.6 | Le degré de dynamisme | 39 |
| 2.6.1 | Dynamisme sans fenêtres de temps (dod) | 39 |
| 2.6.2 | Degré de dynamisme effectif - edod | 40 |
| 2.6.3 | Dynamisme avec fenêtres de temps - edod-tw | 41 |
| 2.7 | Classification du DVRP | 41 |
| 2.7.1 | Problèmes faiblement dynamiques | 41 |
| 2.7.2 | Problèmes modérément dynamiques | 43 |
| 2.7.3 | Problèmes fortement dynamiques | 43 |
| 2.8 | Le DVRP dans la littérature | 43 |
| 2.8.1 | Méthodes séquentielles | 43 |
| 2.8.1.1 | Stratégies simples | 44 |
| 2.8.1.2 | La méthode d’insertion classique | 47 |
| 2.8.1.3 | Les métaheuristiques | 50 |
| 2.8.2 | Méthodes parallèles | 50 |
| 2.8.2.1 | La recherche tabou parallèle | 50 |
| 2.9 | Conclusion | 52 |

Chapitre 3 Un algorithme génétique “en ligne” comme méthode de résolution du

| | | |
|---------------|------------------------|-----------|
| DVRPTW | | 55 |
| 3.1 | Introduction | 55 |
| 3.2 | Motivation | 56 |

| | | |
|-----------------------------------------------------------------------|----------------------------------------------------------------------------------|-----------|
| 3.3 | Algorithme Génétique | 56 |
| 3.3.1 | Cycle de vie | 57 |
| 3.3.2 | Le codage de la solution | 59 |
| 3.3.3 | Fonction d'évaluation | 60 |
| 3.3.3.1 | La phase de décodage d'un chromosome | 62 |
| 3.3.4 | Population initiale | 65 |
| 3.3.5 | Sélection | 65 |
| 3.3.6 | Reproduction | 65 |
| 3.3.6.1 | Le croisement (Crossover) | 66 |
| | IX (1-point) | 66 |
| | PMX (Partially Mapped Crossover) | 67 |
| | MX1T/D | 68 |
| 3.3.6.2 | La mutation | 69 |
| 3.3.6.3 | L'élitisme | 71 |
| 3.4 | Environnement de Simulation | 72 |
| 3.4.1 | Le composant Algorithme Génétique (AG) | 72 |
| 3.4.2 | Le composant Gestion d'Evénements (GE) | 72 |
| 3.4.3 | Le fonctionnement du simulateur | 73 |
| 3.4.3.1 | L'évènement "fin de service" | 75 |
| 3.4.3.2 | L'évènement "arrivée d'un nouveau client" | 76 |
| 3.5 | Conclusion | 78 |
| Chapitre 4 Le réglage des paramètres de l'algorithme génétique | | 79 |
| 4.1 | Introduction | 79 |
| 4.2 | Revue sur la littérature | 81 |
| 4.2.1 | Réglage de paramètres | 81 |
| 4.2.2 | Contrôle de paramètres | 82 |
| 4.3 | Les plans d'expériences | 84 |
| 4.3.1 | Introduction | 84 |
| 4.3.2 | Terminologie | 85 |
| 4.3.3 | Les types des plans | 85 |
| 4.3.4 | Démarche générale pour utiliser des plans d'expériences | 87 |
| 4.4 | Application des plans d'expériences pour le réglage des paramètres de l'AG . . . | 88 |
| 4.4.1 | Etape 1 - Le plan de criblage | 92 |
| 4.4.2 | Etape 2 - Plans d'expériences complets vs. fractionnaires | 97 |
| 4.4.2.1 | Le plan factoriel complet | 97 |
| 4.4.2.2 | Le plan fractionnaire | 101 |

| | | |
|----------------------------------------------|------------------------------------------------------------------------------------|------------|
| 4.4.2.3 | Plan complet vs. plan fractionnaire | 103 |
| 4.4.3 | Etape 3 - Plans d'expériences fractionnaires pour les problèmes dynamiques | 104 |
| 4.5 | Conclusion | 106 |
| Chapitre 5 | Plate-forme et résultats | 109 |
| 5.1 | Introduction | 109 |
| 5.2 | La plate-forme développée | 110 |
| 5.3 | Le problème DVRPTW | 113 |
| 5.3.1 | Le benchmark utilisé | 113 |
| 5.3.2 | Les résultats | 116 |
| 5.4 | Le problème DVRP sans fenêtres de temps | 119 |
| 5.4.1 | Le benchmark utilisé | 119 |
| 5.4.2 | Les résultats | 119 |
| 5.5 | Le problème DTRP | 120 |
| 5.5.1 | Le benchmark utilisé | 120 |
| 5.5.2 | Les résultats | 122 |
| 5.6 | Le problème DTSPTW | 124 |
| 5.6.1 | Le benchmark utilisé | 124 |
| 5.6.2 | Les résultats | 124 |
| 5.7 | Le problème DPDPTW | 124 |
| 5.7.1 | Le benchmark utilisé | 125 |
| 5.7.2 | Les résultats | 126 |
| 5.8 | Conclusion | 127 |
| Conclusions générales et perspectives | | 129 |
| Annexes | | 131 |
| Annexe A | Les plans d'expériences | 133 |
| A.1 | Plan de criblage | 133 |
| A.2 | Plans d'expériences complets vs. fractionnaires | 139 |
| A.3 | Le contraste vs. l'effet | 151 |
| A.4 | Plans d'expériences fractionnaires de problèmes dynamiques | 153 |
| A.5 | Plans d'expériences fractionnaires de problèmes DPDPTW | 159 |
| Annexe B | Exemple : Le problème C101 dynamique | 161 |
| Annexe C | Courbes d'évolutions du fitness (Distance) de 6 problèmes DVRPTW Dynamiques | 167 |

| | |
|----------------------|------------|
| Bibliographie | 175 |
| Index | 189 |

Table des figures

| | | |
|------|----------------------------------------------------------------------------------------------------------------------|----|
| 1.1 | Le problème VRP | 8 |
| 1.2 | Les méthodes de résolution du VRP | 13 |
| 1.3 | Fusion dans (Clarke & Wright) | 15 |
| 1.4 | Groupe en premier, route en second | 16 |
| 2.1 | Les problèmes dynamiques | 34 |
| 2.2 | Le problème DVRP | 36 |
| 2.3 | Classification du VRP | 36 |
| 2.4 | Le cas d'échec | 37 |
| 2.5 | Deux scénarios ayant des "dod" équivalents | 40 |
| 2.6 | Classification du DVRP | 42 |
| 2.7 | Les méthodes de résolution du DVRP | 43 |
| 2.8 | mod TSP | 45 |
| 2.9 | L'heuristique GEN | 45 |
| 2.10 | Heuristique d'Insertion | 48 |
| 2.11 | Simulation du système dynamique | 48 |
| 3.1 | Méthodes de résolution | 57 |
| 3.2 | L'algorithme génétique | 58 |
| 3.3 | Représentation directe | 59 |
| 3.4 | Représentation indirecte | 60 |
| 3.5 | Le chromosome représentant les clients qui restent à servir à l'instant t | 60 |
| 3.6 | L'ensemble des tournées représentant les clients desservis à l'instant t | 61 |
| 3.7 | Le décodage d'un chromosome | 62 |
| 3.8 | Le chromosome du problème DPDPTW | 64 |
| 3.9 | Principe du croisement | 66 |
| 3.10 | Principe de la mutation | 70 |
| 3.11 | L'opérateur de mutation 1-Opt | 70 |
| 3.12 | L'opérateur de mutation Or-Opt | 71 |
| 3.13 | L'évolution générale de la population | 71 |
| 3.14 | Simulateur pour le DVRPTW | 72 |
| 3.15 | Durée de la simulation | 73 |
| 3.16 | La simulation de la journée de service | 73 |
| 3.17 | Diagramme UML montrant la séquence d'interactions entre les composants AG et GE, à la fin d'une génération | 74 |
| 3.18 | Procédure d'évènement fin de service | 76 |
| 3.19 | Procédure d'évènement nouveau client | 77 |

| | | |
|-----|--------------------------------------------------------------------------------|-----|
| 4.1 | Les méthodes utilisées pour la configuration de paramètres d'AG | 81 |
| 4.2 | Système vu comme une boîte noire | 85 |
| 4.3 | Effet du facteur X1 sur la réponse Y | 86 |
| 4.4 | Interaction entre facteurs | 86 |
| 4.5 | Les réponses Y_1 et Y_2 | 89 |
| 4.6 | Le système à analyser | 90 |
| 4.7 | Effets des facteurs pour la première réponse Y1 (problème C109) | 94 |
| 4.8 | Effets des facteurs pour la deuxième réponse Y2 (problème C109) | 95 |
| 4.9 | Courbe d'évolution du Y_1 (Distance) du problème R109 Dynamique | 105 |
| 5.1 | Plate-forme | 111 |
| 5.2 | L'onglet "Données" | 112 |
| 5.3 | L'onglet "Simulation" | 113 |
| 5.4 | Les jeux de tests de Solomon pour le VRPTW | 115 |
| 5.5 | Les résultats du problème DTRP | 123 |
| 5.6 | Les résultats du problème DTSPWTW | 125 |
| B.1 | La solution du problème DVRPTW C101 Dyn (Au début de la simulation) | 165 |
| B.2 | La solution du problème DVRPTW C101 Dyn (Au moment donné t) | 165 |
| B.3 | La solution du problème DVRPTW C101 Dyn (A la fin de la simulation) | 166 |
| C.1 | Courbe d'évolution du fitness (Distance) du problème C108 Dynamique | 168 |
| C.2 | Courbe d'évolution du fitness (Distance) du problème R109 Dynamique | 169 |
| C.3 | Courbe d'évolution du fitness (Distance) du problème RC106 Dynamique | 170 |
| C.4 | Courbe d'évolution du fitness (Distance) du problème C207 Dynamique | 171 |
| C.5 | Courbe d'évolution du fitness (Distance) du problème R205 Dynamique | 172 |
| C.6 | Courbe d'évolution du fitness (Distance) du problème RC201 Dynamique | 173 |

Liste des tableaux

| | | |
|------|------------------------------------------------------------------------------------------------------|-----|
| 1 | Table d'abréviations | 2 |
| 1.1 | Table des variantes du problème VRP | 11 |
| 1.2 | Les fonctions objectifs choisies pour la résolution du VRPTW | 31 |
| 2.1 | Le problème DVRP et ses extensions dans la littérature | 53 |
| 4.1 | Un plan d'expériences complet | 87 |
| 4.2 | Les niveaux des facteurs | 90 |
| 4.3 | Le schéma général de la démarche | 91 |
| 4.4 | Plan de criblage du problème C109 statique | 93 |
| 4.5 | Plans de criblage des problèmes statiques : classement des niveaux par effets décroissants | 96 |
| 4.6 | Les niveaux des facteurs après l'élimination d'un niveau | 97 |
| 4.7 | Plan complet du problème C109 statique | 99 |
| 4.8 | Les meilleures combinaisons pour le problème C109 | 101 |
| 4.9 | Plan fractionnaire du problème C109 statique | 102 |
| 4.10 | Les meilleures combinaisons pour le problème C109 en utilisant un plan fractionnaire | 102 |
| 4.11 | Tableau de comparaison entre plans fractionnaires et plans complets | 103 |
| 4.12 | Comparaison entre les deux plans | 103 |
| 4.13 | Les meilleurs réglages de chaque groupe des benchmarks dynamiques | 106 |
| 5.1 | Comparaison des résultats obtenus par plusieurs méthodes | 117 |
| 5.2 | AGD vs. l'heuristique d'insertion | 119 |
| 5.3 | AGD vs. ACS (DVRP) | 121 |
| 5.4 | Les 2 niveaux des paramètres de l'AG à régler pour le problème DPDPTW | 126 |
| 5.5 | Les résultats des plans fractionnaires pour le problème DPDPTW | 126 |
| 5.6 | Comparaison entre les résultats | 127 |
| A.1 | Plan de criblage du problème C109 statique | 133 |
| A.2 | Plan de criblage du problème C204 statique | 134 |
| A.3 | Plan de criblage du problème R101 statique | 135 |
| A.4 | Plan de criblage du problème R201 statique | 136 |
| A.5 | Plan de criblage du problème RC104 statique | 137 |
| A.6 | Plan de criblage du problème RC201 statique | 138 |
| A.7 | Plan complet du problème C109 statique | 140 |
| A.8 | Plan fractionnaire du problème C109 statique | 140 |
| A.9 | Plan complet du problème C204 statique | 142 |

| | | |
|------|---------------------------------------------------------------------------|-----|
| A.10 | Plan fractionnaire du problème C204 statique | 142 |
| A.11 | Plan complet du problème R101 statique | 144 |
| A.12 | Plan fractionnaire du problème R101 statique | 144 |
| A.13 | Plan complet du problème R201 statique | 146 |
| A.14 | Plan fractionnaire du problème R201 statique | 146 |
| A.15 | Plan complet du problème RC104 statique | 148 |
| A.16 | Plan fractionnaire du problème RC104 statique | 148 |
| A.17 | Plan complet du problème RC201 statique | 150 |
| A.18 | Plan fractionnaire du problème RC201 statique | 150 |
| A.19 | Plan fractionnaire | 151 |
| A.20 | Plan fractionnaire ordonné | 151 |
| A.21 | Plan complet pour 4 facteurs à 2 niveaux | 152 |
| A.22 | Plan fractionnaire du problème C104 dynamique | 153 |
| A.23 | Plan fractionnaire du problème C108 dynamique | 153 |
| A.24 | Plan fractionnaire du problème C204 dynamique | 154 |
| A.25 | Plan fractionnaire du problème C207 dynamique | 154 |
| A.26 | Plan fractionnaire du problème R104 dynamique | 155 |
| A.27 | Plan fractionnaire du problème R109 dynamique | 155 |
| A.28 | Plan fractionnaire du problème R208 dynamique | 156 |
| A.29 | Plan fractionnaire du problème R205 dynamique | 156 |
| A.30 | Plan fractionnaire du problème RC104 dynamique | 157 |
| A.31 | Plan fractionnaire du problème RC106 dynamique | 157 |
| A.32 | Plan fractionnaire du problème RC204 dynamique | 158 |
| A.33 | Plan fractionnaire du problème RC201 dynamique | 158 |
| A.34 | Plan fractionnaire du problème Rnd8-10-100-005 Dynamique | 159 |
| A.35 | Plan fractionnaire du problème Rnd8-10-100-015 Dynamique | 159 |
| A.36 | Plan fractionnaire du problème Rnd8-10-100-025 Dynamique | 160 |
| B.1 | Fichier de données du problème dynamique C101Dyn (Format texte) | 161 |

Table des algorithmes

| | | |
|-----|-----------------------------------------------------------------------|----|
| 1.1 | Pseudo-Code d'un algorithme génétique de base | 18 |
| 1.2 | Pseudo-Code de la règle de Metropolis | 21 |
| 1.3 | Pseudo-Code de l'algorithme du recuit simulé | 21 |
| 1.4 | Pseudo-Code de la méthode de recherche tabou | 23 |
| 1.5 | Pseudo-Code de la méthode de colonie de fourmis | 25 |
| 2.1 | Pseudo-Code de l'algorithme de la recherche tabou parallèle | 51 |
| 3.1 | Algorithme d'insertion (décodage d'un chromosome) | 63 |
| 3.2 | Algorithme de traitement de l'évènement "fin de service" | 75 |
| 3.3 | Algorithme de traitement de l'évènement "nouveau client" | 77 |

Table d'abréviations

| Abréviation | Description |
|---------------|-------------------------------------------------------------------|
| RO | Recherche Opérationnelle |
| AE | Algorithmes Evolutionnistes |
| AG | Algorithme Génétique |
| RT | Recherche Tabou |
| PE | Plans d'Expériences |
| PC | Plan d'expériences Complet |
| PF | Plan d'expériences Fractionnaire |
| GIS | Graphical Information Systems |
| GPS | Global Positioning System |
| DTP | Dynamics Transports Problems |
| VRP | Vehicle Routing Problem |
| VRPTW | Vehicle Routing Problem with Time Windows |
| VRPB | Vehicle Routing Problem with Backhauls |
| DVRP | Dynamic Vehicle Routing Problem |
| DVRPTW | Dynamic Vehicle Routing Problem with Time Windows |
| SVRP | Stochastic Vehicle Routing Problem |
| PVRP | Periodic Vehicle Routing Problem |
| MDVRP | Multi-Depot Vehicle Routing Problem |
| TRP | Traveling Repairman Problem |
| DTRP | Dynamic Traveling Repairman Problem |
| TSP | Traveling Salesman Problem |
| DTSP | Dynamic Traveling Salesman Problem |
| ADTSPTW | A-priori Dynamic Traveling Salesman Problem with Time Windows |
| VRPPD | Vehicle Routing Problem with Pick-up and Delivery |
| DPDPTW | Dynamic Pick-up and Delivery Problem with Time Windows |
| OVRP | Open Vehicle Routing Problem |
| VRPHF | Vehicle Routing Problem with Heterogeneous Fleet |
| VRPSD | Vehicle Routing Problem with Split Delivery |
| <i>m</i> -VRP | Vehicle Routing Problem with Limited Number of Vehicles |
| dod | Degré de dynamisme (Degree Of Dynamism) |
| edod | Degré de dynamisme effectif (Effective Degree Of Dynamism) |
| edod-tw | Degré de dynamisme effectif pour les problèmes à fenêtre de temps |
| IP | Idle Point |
| FCFS | First Come First Serve |
| A suivre ⇒ | |

Table d'abréviations

| Abréviation | Description |
|-------------|-----------------------------------------------|
| NN | Nearest Neighbor |
| PART | The partitioning policy |
| SQM | Stochastic Queue Median |
| GEN | GENeration strategy |
| HI-REQ | Highest expected number of immediate REQuests |
| DF | Drive-First |
| WF | Wait-First |
| DW | Dynamic Waiting |
| ADW | Advanced Dynamic Waiting |
| PFIH | Push Forward Insertion Heuristic |
| PMX | Partially Mapped Crossover |
| MX1T/D | Merge Heuristic (Time, Distance) |
| GE | Gestion d'Événements |

TAB. 1: Table d'abréviations

Introduction générale

Ces dernières années les systèmes de transport utilisés pour le ramassage et la distribution de biens ou de services ont fait l'objet de nombreuses études dans la communauté scientifique **RO : Recherche Opérationnelle**. Cet intérêt est partiellement dû au nombre élevé de fusions de compagnies qui laissent les planificateurs de flotte de véhicules avec des problèmes encore plus grands et plus complexes, à résoudre. Un fait qui complique la planification est la prise en compte du temps qui est devenu une ressource extrêmement critique pour les entreprises. De nos jours, la plupart des systèmes de transport doivent pouvoir fonctionner en respectant des contraintes temporelles strictes et ceci en s'adaptant aux aléas du problème. En effet, les clients ou partenaires d'une entreprise exigent de celle-ci une qualité de service garantie (délais à respecter). De plus, l'environnement dans lequel une entreprise doit évoluer est bien souvent incertain et donc sa réactivité est également un atout important. Ceci a conduit à définir des modèles de pilotage des systèmes de transport dits "**dynamiques**" dans lesquels une partie des données est considérée comme dépendante du temps.

Le domaine dans lequel s'inscrit nos travaux, concerne l'élaboration des tournées de véhicules (*VRP : Vehicle Routing Problem*). Le problème classique de l'élaboration des tournées de véhicules (**VRP**) consiste à construire des routes avec un coût minimum pour que les véhicules puissent visiter exactement une fois chaque client géographiquement distribué. Le VRP est un sous problème important dans le domaine des systèmes de distribution et beaucoup d'efforts ont été consacrés en recherche sur divers aspects du VRP. Cependant, une grande partie de ces travaux ont surtout porté sur le contexte *statique* qui suppose que les données du problème sont d'une part parfaitement et exhaustivement connues et d'autre part ne sont pas sujettes à des changements pendant l'horizon de planification. Le développement rapide observé conjointement dans les télécommunications et les équipements informatiques a suscité un intérêt croissant de la communauté scientifique et des utilisateurs potentiels de ces méthodes pour l'étude de problèmes *dynamiques* d'élaboration de tournées de véhicules bien plus réalistes que le problème classique d'élaboration de tournées de véhicules. La gestion des tournées en temps réel se fait alors "au fil de l'eau", c'est-à-dire sur la base d'informations qui deviennent accessibles au fur et à mesure du temps.

Le travail présenté dans cette thèse traite de la résolution du problème de l'élaboration dynamique de tournées de véhicules avec fenêtres de temps (**DVRPTW : Dynamic Vehicle Routing Problem with Time Windows**) et de quelques unes de ses extensions.

L'objectif de ces travaux au démarrage de cette thèse était double. D'une part, il s'agissait de montrer qu'une approche de type évolutionniste était utilisable dans un cadre dynamique. D'autre part, de vérifier que les performances que l'on pouvait attendre de ce type d'approche étaient comparables voire supérieures à celles des meilleures techniques utilisées à ce jour.

Pour atteindre ces objectifs, nous avons utilisé la technique des **Algorithmes Génétiques**

(**AG**). A notre connaissance, aucune approche dynamique de ce type n'était utilisée pour résoudre ce problème bien que de nombreux travaux aient été effectués sur des approches de type AG mais dans un cadre statique.

Nous avons donc défini un modèle de planification continu dirigé par les événements. Le modèle est qualifié de continu car l'optimisation est réalisée de façon incrémentale à chaque événement significatif survenant tout au long de la journée d'ouverture. L'algorithme génétique défini, est basé sur des chromosomes de taille variable dans le temps permettant de prendre en compte l'arrivée de nouveaux clients pendant l'exécution effective des tournées de véhicules. Un codage *indirect* de type "liste à servir" permettant d'engendrer tout l'espace de recherche a été choisi pour faciliter l'insertion dynamique de nouveaux clients. Le codage indirect permet d'obtenir des chromosomes représentant des solutions pertinentes après chaque application d'opérateur génétique. Le décodage utilisé pour construire les tournées de véhicules est basé sur une *heuristique d'insertion constructive*, autorisant ainsi un décodage rapide qui peut s'avérer utile dans un cadre dynamique avec des contraintes de temps de réponse serrées liées à une exploitation "en ligne" temps réel.

L'efficacité des approches **AG** et plus généralement des métaheuristiques en général pose la question délicate du réglage de certains paramètres par rapport au problème à traiter. Pour ne pas pénaliser également les temps de réponse, nous avons utilisé un réglage a priori de ces paramètres plutôt qu'une approche adaptative. Pour ce faire, nous avons d'abord classé les instances de problèmes de [Solomon 1987] modifiées par [Gendreau et al. 1999a] en fonction du degré de dynamisme défini par [Lund et al. 1996]. Puis, nous avons utilisé un **Plan d'Expériences (PE)** pour déterminer un bon compromis entre plusieurs choix de réglage prédéterminés pour chaque représentant de classe.

Les réglages ainsi obtenus, ont été ensuite utilisés sur chaque instance de chaque classe de problème. Les résultats obtenus sont encourageants et montrent des performances comparables à celles des meilleures méthodes de ce jour.

Le présent document est structuré de la façon suivante :

- Le premier chapitre présente la problématique du **VRP**. Dans ce chapitre, nous rappelons les différents éléments qui composent ce problème, ainsi que les contraintes à satisfaire et l'objectif à optimiser. La formulation mathématique de ce problème est ensuite rappelée. Nous rappelons la plupart des variantes de ce problème trouvées dans la littérature, puis nous présentons un état de l'art sur les méthodes qui sont utilisées pour la résolution de ce problème. Enfin, une étude approfondie sur le problème particulier du VRPTW est présentée.
- Le deuxième chapitre est consacré à la version dynamique du problème VRP. Dans ce chapitre, nous commençons par une présentation des aspects dynamiques de ce problème, puis une comparaison entre les deux versions (statique et dynamique) est présentée. Ensuite, la notion de *degré de dynamisme* d'un problème dynamique est rappelée et une classification des problèmes dynamiques **DVRP** (Dynamic Vehicle Routing Problem) est proposée à partir de cette notion. Nous terminons ce chapitre par une revue sur la littérature concernant les méthodes de résolution du problème VRP dynamique et de quelques unes de ses extensions.
- Le troisième chapitre se focalise sur l'approche utilisée pour la résolution du VRPTW dynamique. Il est divisé principalement en deux parties. La première partie est consacrée à notre approche évolutionniste qui utilise l'algorithme génétique comme méthode de résolu-

tion métaheuristique ; des extensions de cet algorithme ont été proposées pour qu'il puisse traiter les deux types d'évènements issus de ce problème dynamique :

1. L'évènement "**arrivée d'un nouveau client**".
2. L'évènement "**fin de service d'un client**".

Dans la deuxième partie nous présentons l'architecture logicielle du prototype réalisé pour gérer les tournées de véhicules et simuler les évènements à traiter pendant la journée de service.

- Le quatrième chapitre présente une étude basée sur les *plans d'expériences* pour résoudre le problème sensible du réglage des paramètres de l'algorithme génétique. Ce chapitre commence par une introduction sur les méthodes utilisées pour régler ces paramètres. Puis, nous présentons la technique des plans d'expériences comme méthode de réglage pour ces paramètres. Nous avons essayé de régler 6 paramètres de l'algorithme génétique, ces paramètres sont : la taille de la population, le type et la probabilité de croisement, le type et la probabilité de mutation et le taux d'élitisme. A ce stade nous avons utilisé le benchmark de Solomon conçu spécialement pour le problème VRPTW. De plus nous avons aussi utilisé une version dynamique de ce benchmark, proposée par [Gendreau et al. 1999a] pour la version dynamique du VRPTW. La démarche que nous avons suivie, se compose de trois étapes principales :
 - un **plan de criblage** pour diminuer le nombre de niveaux de chaque paramètre à régler.
 - une étude comparative **plan complet vs. plan fractionnaire** pour mesurer la dégradation engendrée par le choix d'un plan fractionnaire.
 - un **plan fractionnaire** pour chaque représentant de chaque classe du benchmark dynamique.
- Pour évaluer notre approche évolutionniste, nous avons développé une **plate-forme en JAVA** adaptée à l'élaboration dynamique de tournées de véhicules. Elle permet de représenter graphiquement les résultats d'une simulation de recherche de tournées de véhicules sur une journée complète de service en tenant compte de tous les évènements qui se produisent au cours du temps. Le dernier chapitre détaille cette plate-forme et présente les résultats obtenus en utilisant le benchmark dynamique VRPTW. Nous avons comparé ces résultats avec ceux obtenus par d'autres approches et avec une approche heuristique simple que nous avons développée. Nous comparons également notre approche sur différentes variantes de problèmes dynamiques avec d'autres approches de résolution de la littérature.

Une conclusion de ces travaux est donnée, puis des perspectives terminent ce mémoire de thèse.

Chapitre 1

Le problème d'élaboration de tournées de véhicules CVRP¹

Sommaire

| | | |
|------------|------------------------------------------|-----------|
| 1.1 | Introduction | 7 |
| 1.2 | Formulation | 8 |
| 1.2.1 | Détail des équations | 10 |
| 1.3 | Complexité | 10 |
| 1.4 | Variantes du problème VRP | 10 |
| 1.5 | Les méthodes de résolution du VRP | 12 |
| 1.5.1 | Les méthodes exactes | 13 |
| 1.5.2 | Les heuristiques | 14 |
| 1.5.3 | Les métaheuristiques | 17 |
| 1.6 | Le problème VRPTW | 26 |
| 1.6.1 | La formulation du VRPTW | 27 |
| 1.6.2 | Revue de la littérature VRPTW | 28 |
| 1.7 | Conclusion | 30 |

1.1 Introduction

Le problème de construction de tournées de véhicules VRP (Vehicle Routing Problem) est un problème d'optimisation formulé en 1959 par [Dantzig et al. 1959] ; il modélise bien un problème de transport très répandu qui est celui de la livraison (ou de la collecte) de produits auprès d'un ensemble de clients répartis géographiquement en minimisant la distance totale parcourue. Les contraintes que la solution doit respecter sont les suivantes :

- Tous les clients doivent être desservis.
- Un client ne peut être servi que par un et un seul véhicule.
- En quittant un client, un véhicule ne peut se diriger que vers un seul autre client.
- Un véhicule ne peut servir un client que s'il a assez de capacité pour servir la demande de ce client.

¹Capacitated Vehicle Routing Problem

Ainsi, l'objectif est de trouver des itinéraires (routes) de visite des clients (voir la figure [1.1]) qui minimise la distance totale parcourue par les véhicules. En sachant que dans ce problème la distance entre deux clients soit la distance euclidienne entre les positions de ces deux clients.

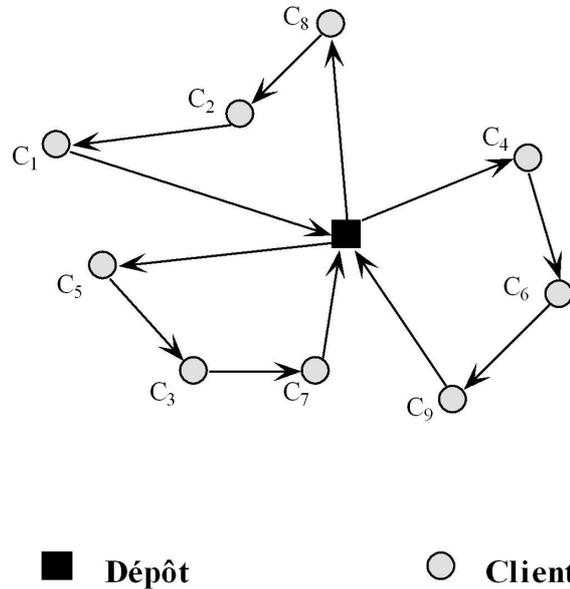


FIG. 1.1 – Le problème VRP

Ce chapitre est divisé en deux parties principales. La première partie est consacrée au problème du VRP ; dans cette partie, nous présentons la formulation mathématique de ce problème, nous définissons la plupart des variantes du VRP rencontrées dans la littérature et nous présentons les méthodes utilisées pour la résolution de ce problème. La deuxième partie est consacrée au problème du VRPTW.

1.2 Formulation

Rappelons que le problème VRP consiste à affecter chaque client à une tournée (i.e. route) effectuée par un seul véhicule et à trouver un ordre de visites des clients pour chaque véhicule de façon à satisfaire les contraintes de capacité des véhicules, et les quantités de produit demandé par chaque client, dans le cas d'un problème de livraison. Il convient de noter que la formulation proposée correspond également au VRP dans le cas d'un problème de collecte ou de ramassage. L'objectif dans ce problème est de trouver l'ensemble des *Tournées* qui minimisent la distance totale parcourue pour un nombre minimal de véhicules partant d'un dépôt et y retournant. Ce problème est une extension du problème classique du voyageur de commerce (TSP) [Flood 1956].

Nous allons formuler le VRP selon un modèle de recherche opérationnelle de [Solomon et al. 1988] dans la forme utilisée par [LeBouthillier 2000]. En fait, ces travaux ont formulé le problème VRPTW (VRP with Time Windows), dans cette section nous allons présenter la formulation du problème VRP classique sans contraintes temporelles et dans la section (1.6) nous allons continuer à formuler le problème VRPTW en ajoutant ces contraintes.

Un graphe $G = (N, A)$ représente notre problème où :

- N représente les positions des clients et du dépôt,
- A représente les arcs entre deux clients $i, j \in N$.

Plus spécifiquement, nous avons un ensemble $C = \{1, \dots, n_c\}$ de clients qui doivent obtenir une livraison de marchandise provenant du dépôt. L'ensemble des positions de ces clients ou nœuds est défini comme l'ensemble $N = C \cup \{0, n_c + 1\}$ où 0 et $n_c + 1$ représentent le dépôt (aller et retour). Une demande positive de produit d_i est associée à chaque client i appartenant à C .

Une flotte de véhicules $V = \{1, \dots, n_v\}$ est disponible au dépôt et chaque véhicule possède la même capacité (flotte homogène) Q telle que $Q \geq \max d_i, \forall i \in N$. Pour tous clients i et $j, \forall i, j \in N$, nous connaissons le coût $c_{i,j}$ de transport direct entre i et j (proportionnel à la distance à parcourir). Pour trouver l'ordre de visite des clients, nous définissons les variables de décisions comme suit :

$$x_{i,j}^v = \begin{cases} 1 & \text{si le véhicule } v \in V \text{ visite le client } j \text{ après le client } i, \\ 0 & \text{sinon.} \end{cases}$$

En définissant y_i comme étant la charge résiduelle du véhicule après avoir desservi le client $i \in C$. Il nous est possible d'écrire formellement le modèle de VRP.

$$\text{Min} \sum_{v \in V} \sum_{i \in N} \sum_{j \in N} c_{i,j} x_{i,j}^v \quad (1.1)$$

Avec les contraintes :

$$\sum_{v \in V} \sum_{j \in N} x_{i,j}^v = 1, \forall i \in C \quad (1.2)$$

$$\sum_{j \in N} x_{i,j}^v - \sum_{j \in N} x_{j,i}^v = 0, \forall i \in C, v \in V \quad (1.3)$$

$$\sum_{j \in N} x_{0,j}^v = 1, \forall v \in V \quad (1.4)$$

$$\sum_{j \in N} x_{j,n+1}^v = 1, \forall v \in V \quad (1.5)$$

$$x_{i,j}^v = 1 \Rightarrow y_i - d_j = y_j, \forall i, j \in N, v \in V \quad (1.6)$$

$$y_0 = Q, 0 \leq y_i, \forall i \in C \quad (1.7)$$

$$x_{i,j}^v \in \{0, 1\}, \forall i, j \in N, v \in V \quad (1.8)$$

La fonction de coût euclidien de la solution $X = (x_{i,j}^v), \forall i, j \in N, \forall v \in V$ est définie par :

$$\text{coût}(X) = \sum_{v \in V} \sum_{i \in N} \sum_{j \in N} c_{i,j} x_{i,j}^v \quad (1.9)$$

Le nombre de véhicules utilisés par la solution X , est défini par :

$$\text{Nb véhicules}(X) = \sum_{v \in V} \sum_{j \in C} x_{0,j}^v \quad (1.10)$$

1.2.1 Détail des équations

La fonction objectif (équation 1.1) représente le nombre de véhicules utilisés pour les n_v routes et la somme des coûts de parcours.

La formulation du problème nécessite de satisfaire certaines contraintes :

- L'équation 1.2 assure qu'on part une et une seule fois de chaque client, avec un seul véhicule.
- L'équation 1.3 assure que le véhicule qui arrive chez un client est le même que celui qui part de ce client.
- L'équation 1.4 assure que chaque véhicule ne sort qu'une seule fois du dépôt.
- L'équation 1.5 assure le retour unique au dépôt pour chaque véhicule (ou tournée).
- Les équations 1.6 et 1.8 définissent les contraintes de capacité et d'intégrité.
- Les équations 1.9 et 1.10 sont des fonctions de mesure qui permettent respectivement de quantifier la solution selon la distance totale parcourue, ainsi que le nombre de véhicules utilisés.

1.3 Complexité

Le problème d'élaboration de tournées de véhicules est un problème NP-difficile, c'est-à-dire qu'il n'existe pas à ce jour un algorithme déterministe pouvant résoudre ce problème en temps polynomial [Zhu 1999]. Pour des problèmes comportant un grand nombre de clients (> 100 clients), des méthodes approchées sont nécessaires pour les résoudre.

1.4 Variantes du problème VRP

Le problème d'élaboration de tournées de véhicules présenté ci-dessus, souvent appelé CVRP (pour Capacitated Vehicle Routing Problem), est la définition de base de ce problème de routage générique. De nombreuses extensions existent. Nous pouvons étendre ses caractéristiques et construire la table des variantes de ce problème (voir la table 1.1).

Par conséquent, diverses variantes du VRP apparaissent dans la littérature grâce à leurs applications importantes dans le domaine du transport et de la distribution. Nous allons présenter les principales variantes du VRP :

1. **VRPTW** (Vehicle Routing Problem with Time Windows) : le problème d'élaboration de tournées de véhicules avec fenêtres de temps est le problème le plus étudié. C'est un problème de VRP classique auquel on a ajouté une contrainte supplémentaire : chaque client doit fournir une limite de temps supérieure et inférieure entre lesquelles il désire être servi [Cordeau et al. 2002]. Nous allons présenter ce problème en détail dans les sections suivantes.
2. **VRPB** (Vehicle Routing Problem with Backhauls) est une prolongation du VRP classique qui inclut un ensemble de clients à qui des produits doivent être livrés et un ensemble de fournisseurs dont les marchandises doivent être amenées au dépôt. En outre, sur chaque tournée, toutes les livraisons doivent être effectuées avant que les marchandises puissent être ramassées pour éviter de réarranger les charges sur le véhicule [Jacobs-Blecha et al. 1993].

| Caractéristiques | Options possibles |
|---------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Nombre de véhicules disponibles | - un - plusieurs |
| Type de véhicule | - homogène - hétérogène |
| Capacité de véhicule | - finie - infinie |
| Dépôts | - un - plusieurs |
| Demandes des clients | - statiques (connues en avance) - dynamiques (apparaissent au cours de temps) - stochastiques (les demandes suivent des lois aléatoires) - fenêtre de temps. |
| Service proposé | - ramassage ou livraison - ramassage et livraison - ramassage avant livraison - ... |
| Période considérée | - jour - semaine - périodique - ... |

TAB. 1.1 – Table des variantes du problème VRP

3. **DVRP** (Dynamic Vehicle Routing Problem) : dans le VRP classique, toutes les demandes pour tous les clients sont connues a priori. Cependant, de nombreux problèmes de notre vie pratique incluent au moins un (ou plusieurs) caractère dynamique comme par exemple l'apparition d'un nouveau client en cours de journée. Dans ce cas, le décideur doit changer la planification des véhicules en réponse aux nouvelles demandes [Kilby et al. 1998a] qui arrivent au cours du temps.
4. **SVRP** (Stochastic Vehicle Routing Problem) : le problème VRP est dit stochastique lorsque au moins un élément du problème est aléatoire [Gendreau et al. 1999b]. Autrement dit, un élément du problème ne peut être connu avec certitude. Ce peut être les demandes (quantité à livrer ou à ramasser) des clients, le temps ou le coût du transport, ou bien l'ensemble des clients à visiter. Le problème avec demandes stochastiques est le plus étudié. Il est alors supposé que la demande suit une loi de distribution connue (généralement une loi normale).
5. **PVRP** (Periodic Vehicle Routing Problem) : le problème de tournées de véhicules multi-périodique (PTVMP) consiste à livrer pour un ensemble de clients, la quantité demandée d'un ou plusieurs produits sur un horizon de temps donné [Cordeau et al. 2001]. Dans ce problème, chaque client détient un stock de produits dont il consomme une certaine quantité par jour. La tâche principale est de planifier les jours de livraison pour chaque client sur un horizon de temps prédéterminé et ensuite d'organiser les tournées de la flotte de véhicules afin d'effectuer les livraisons nécessaires. Cette planification doit respecter les contraintes sur les niveaux de stocks à maintenir chez les clients et en même temps elle doit minimiser le coût total de livraison sur l'ensemble de l'horizon considéré. Le problème

ainsi présenté est une extension du problème de tournées de véhicules classique par l'ajout de la notion de période et par la nécessité d'intégrer des éléments de la gestion des stocks dans la planification des transports.

6. **MDVRP** (Multi-Depot Vehicle Routing Problem) : dans ce type de problème il y a plusieurs dépôts disponibles qui sont distribués géographiquement. Chaque véhicule part et revient vers son dépôt initial [Fischetti et al. 1999].
7. **TSP** (Traveling Salesman Problem) : le problème du voyageur de commerce est un cas particulier du problème VRP sans contrainte de capacité; il consiste en la détermination d'un parcours de coût minimal (distance, temps, etc.) pour un seul véhicule partant d'une localité, visitant n autres localités et revenant à son point départ [Rego et al. 1994].
8. **TRP** (Traveling Repairman Problem) : le problème de la tournée du dépanneur (TRP) est une variante du problème d'élaboration de tournées de véhicule VRP. L'objectif est de minimiser la latence, autrement dit, la somme pondérée des temps de service des clients, celle-ci est la somme des temps d'attente des clients avant que ceux-ci ne soient atteints [Krumke et al. 2003].
9. **VRPPD** (Vehicle Routing Problem with Pick-up and Delivery) : le problème de ramassage et de livraison (VRPPD) a les mêmes caractéristiques qu'un problème VRP dans lequel chaque client doit fournir deux positions géographiques différentes : la première indique le lieu de ramassage du produit et la seconde indique le lieu de livraison du produit. Ce type de problème introduit une contrainte de précédence : pour chaque tournée, l'opération de ramassage chez un client doit précéder l'opération de livraison [Nanry et al. 2000].
10. **OVRP** (Open Vehicle Routing Problem) : la caractéristique importante de ce problème, qui le distingue du VRP de base, est que les véhicules ne sont pas obligés de retourner au dépôt ou lorsqu'ils sont obligés d'y retourner, ils revisitent les clients qui leurs sont affectés dans l'ordre inverse, par conséquent, les itinéraires des véhicules ne sont pas des chemins fermés mais ouverts [Eglese et al. 2005].
11. **VRPHF** (Vehicle Routing Problem with Heterogeneous Fleet) : ce type de problème cherche à satisfaire les objectifs du VRP mais en utilisant une flotte de véhicules de type différent. En fait, nous pouvons distinguer ces véhicules par leurs capacités, leurs vitesses ou leurs coûts de déplacement, etc. [Prins 2002] et [Taillard 1999].
12. **VRPSD** (Vehicle Routing Problem with Split Delivery) : chaque client peut être visité plus d'une fois si cela est nécessaire. Autrement dit, la demande de client peut être divisée sur plusieurs tournées. Contrairement à ce qui est habituellement supposé dans le problème classique (VRP) la demande de chaque client peut être plus grande que la capacité des véhicules [Archetti et al. 2002].
13. **m -VRP** (Vehicle Routing Problem with Limited Number of Vehicles) : dans ce problème, nous cherchons à résoudre le problème VRP en utilisant seulement un nombre limité de véhicules [Lau et al. 2003]. En sachant que ce nombre est supposé illimité (ou très grand) dans le VRP de base et que dans le cas présent, on cherche à le minimiser.

1.5 Les méthodes de résolution du VRP

Depuis l'introduction du problème VRP formulé par [Dantzig et al. 1959], de nombreuses méthodes ont été proposées pour la résolution de ce type de problème. On trouvera une synthèse

de ces méthodes dans les articles de [Laporte et al. 2000]. Ces méthodes sont principalement divisées en deux groupes, les méthodes exactes et les méthodes approchées. Ces dernières sont aussi divisées en deux sous groupes (voir la figure 1.2) :

- Les méthodes *exactes*
- Les méthodes *approchées*
 - Les *heuristiques*
 - Les *métaheuristiques*

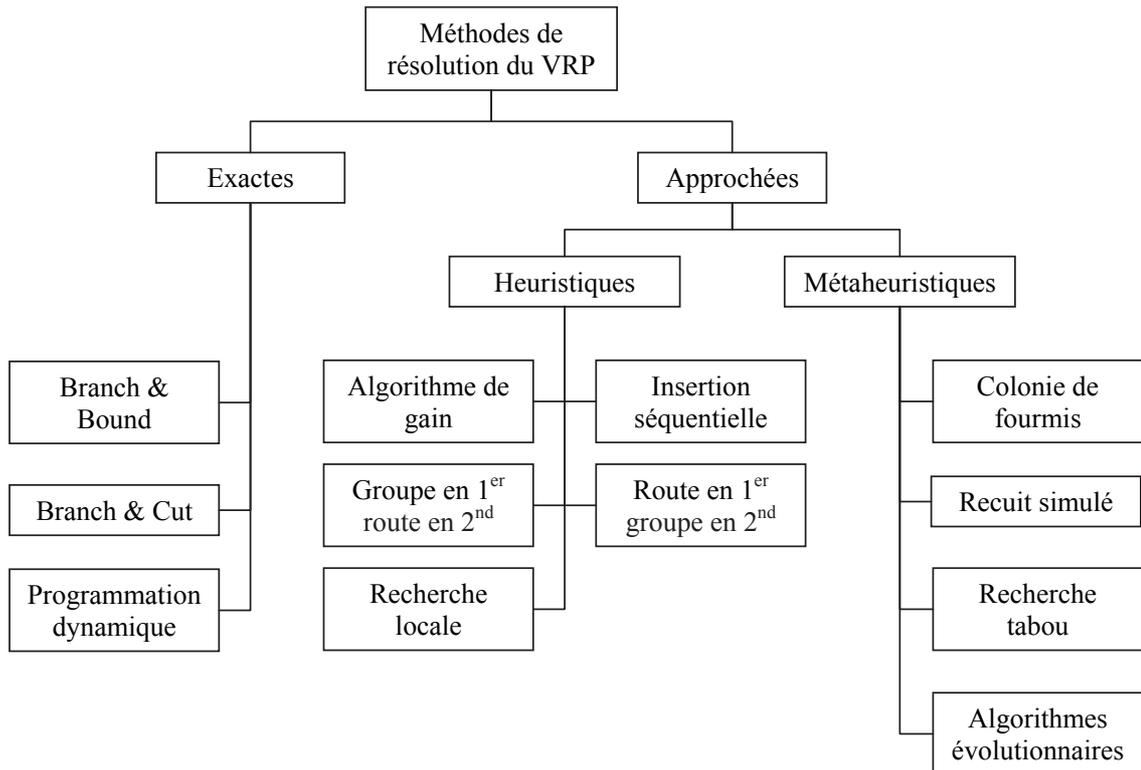


FIG. 1.2 – Les méthodes de résolution du VRP

1.5.1 Les méthodes exactes

Nous avons mentionné précédemment que le problème du VRP était NP-difficile. Ainsi, les méthodes exactes qui le résolvent avec une durée de temps de calcul raisonnable sont difficiles à développer. Néanmoins plusieurs méthodes ont été développées ; elles permettent de résoudre efficacement des problèmes avec 50 clients, et une méthode a résolu un problème avec 100 clients [Ralphs 2003]. Les méthodes exactes pour le VRP peuvent être classifiées dans trois catégories :

1.5.1.1 Les méthodes de recherche arborescente (Branch & Bound)

C'est une méthode qui utilise un arbre de recherche représentant l'espace des solutions et élague des parties de cet arbre [Aronson 1995] pour accélérer la recherche. Elle peut être employée pour trouver les solutions exactes au VRP, mais en général la résolution nécessite un temps exponentiel en fonction de la taille du problème. Elle construit incrémentalement les routes arc par arc. Le problème est transformé sous la forme d'un TSP en créant des dépôts artificiels (égal

au nombre de véhicules disponibles) et en les ajoutant au problème initial. Un branchement dans l'arbre est créé en incluant ou en excluant un arc d'une route. La difficulté de ces méthodes est l'obtention d'une borne de bonne qualité pour améliorer la phase d'élagage. Des méthodes comme la relaxation Lagrangienne [Miller 1995] donne de meilleures bornes que les techniques de relaxation classiques (contrainte de capacité, contrainte de connectivité) [Laporte et al. 1986].

1.5.1.2 Programmation Linéaire en nombres entiers (Branch & Cut)

Lorsque le nombre de contraintes est trop élevé, les méthodes précédentes ne sont plus applicables. Les méthodes de type *Branch & Cut* consiste à rien traduire progressivement les contraintes du problème (i.e. Phase de coupes dans l'espace de solutions) dont on a relâché la contrainte d'intégrité sur les variables entières. Un solveur PL (Programmation Linéaire) est utilisé pour tenter de trouver une solution optimale entière qui respecte les contraintes du problème PLNE (Programmation Linéaire en Nombres Entiers). Dans le cas contraire, une phase de décomposition (i.e. Branch) du problème en 2 sous problèmes est nécessaire et la phase de coupes est relancée sur ces sous problèmes [Toth et al. 2002].

1.5.1.3 Programmation Dynamique

Basée sur le principe de Bellman qui dit : "que toute solution du problème initial de taille N contient la solution optimale du sous-problème de taille $N - 1$ ". Dans la pratique, on commence à résoudre une famille de problèmes de taille 1, puis on passe à l'étape suivante pour résoudre une famille de problèmes de taille 2. Après un certain nombre d'étapes, on retrouve le problème initial de taille N . À chaque étape, des états intermédiaires doivent être considérés et correspondent à une famille de problèmes à résoudre. Pour que l'approche soit faisable, le nombre d'états intermédiaires et le nombre d'étapes doivent être le plus petit possible. Il est parfaitement connu que seul un très petit nombre d'instances de problèmes d'Optimisation Combinatoire peuvent être résolus par la programmation dynamique (10 à 25 clients dans notre cas) [Rego et al. 1994].

1.5.2 Les heuristiques

La plupart des travaux réalisés sur le VRP sont liés à des méthodes approchées, souvent basées sur des *heuristiques*. Elles se contentent d'obtenir en temps raisonnable des solutions aussi bonnes que possible mais elles ne garantissent pas l'optimalité. Les récents développements de la théorie de l'optimisation combinatoire ont grandement enrichi les heuristiques, ce qui s'est concrètement traduit par un grand nombre d'applications réussies. Le mot "heuristique" vient du grec *heureîn* (découvrir). Il qualifie tout ce qui sert à la découverte, à l'invention et à la recherche. Pour l'algorithmique, les heuristiques sont des méthodes qui cherchent à approcher une solution optimale ; nous les appelons **méthodes approchées**.

En général, les heuristiques qui ont été développées pour le VRP sont fondées sur des heuristiques initialement développées pour le TSP.

Nous allons présenter les diverses catégories d'heuristiques de résolutions telles que

- La méthode des gains,
- La méthode d'insertion,
- La méthode de groupe en premier et route en second,
- La méthode de route en premier et groupe en second,
- La méthode de la recherche locale.

1.5.2.1 Algorithme de gain (Clarke & Wright)

L'algorithme des gains de Clarke et Wright [Clarke et al. 1964] est sans aucun doute l'une des méthodes les plus connues pour le VRP. Dans cet algorithme, le nombre de véhicules est une variable de décision. Il nécessite le calcul du gain possible entre (i, j) lié à la fusion de routes comme suit :

Au départ on crée n_c routes $(0, i, 0)$ (un client i par tournée, 0 représentant le dépôt) pour $i = 1, \dots, n_c$. Puis on regarde le gain réalisé par des fusions possibles de routes. Pour calculer le gain lié à une fusion de routes on évalue $S_{i,j} = c_{i,0} + c_{0,j} - c_{i,j}$ pour $i, j = 1, \dots, n_c$ et $i \neq j$, $c_{i,j}$ étant le coût de l'arc (i, j) .

Trier ensuite les gains possibles par ordre décroissant et choisir la fusion réalisant le gain le plus grand. Pour les étapes suivantes, on cherchera à fusionner des routes deux à deux soit au début ou en fin de tournée. Par exemple si on considère une route $(0, i, \dots, j, 0)$ on va chercher à déterminer le meilleur gain $S_{k,i}$ ou $S_{j,l}$ parmi toutes les fusions possibles qui peuvent être réalisées en formant une route réalisable (respectant les contraintes du problème) par la fusion de la route courante et d'une autre route se terminant par $(k, 0)$ ou commençant par $(0, l)$. L'algorithme s'arrête dès qu'il n'est plus possible de trouver une fusion améliorante (voir la figure 1.3).

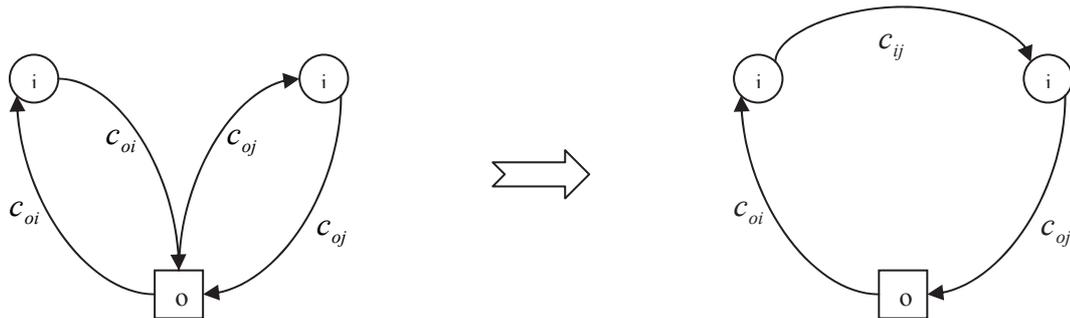


FIG. 1.3 – Fusion dans (Clarke & Wright)

1.5.2.2 Méthode d'insertion séquentielle

Les algorithmes d'insertion sont constitués de deux phases, la première sélectionne le prochain noeud à insérer et la deuxième réalise l'opération d'insertion.

L'heuristique d'insertion du plus proche voisin choisi séquentiellement et de façon répétitive l'arc le plus court qui n'a pas encore été sélectionné pour étendre le cycle (i.e. route) courant. Plusieurs variantes incluent des critères de sélection plus subtils tel que le plus grand angle formé entre les arcs. Pour sa part, l'insertion de Mole et Jameson [Mole et al. 1976] construit une seule route à la fois en insérant un arc à la fois, tandis que la construction parallèle de [Christofides et al. 1979] construit alternativement en séquentiel et en parallèle les différentes routes.

1.5.2.3 Algorithme groupe en premier, route en second

Probablement les plus connus, les algorithmes de type "groupe en premier, route en second" se basent sur une approche géométrique permettant de former des groupes de clients pour ensuite

trouver des routes à l'intérieur de ces groupes [Gillett et al. 1974] (voir la figure 1.4).

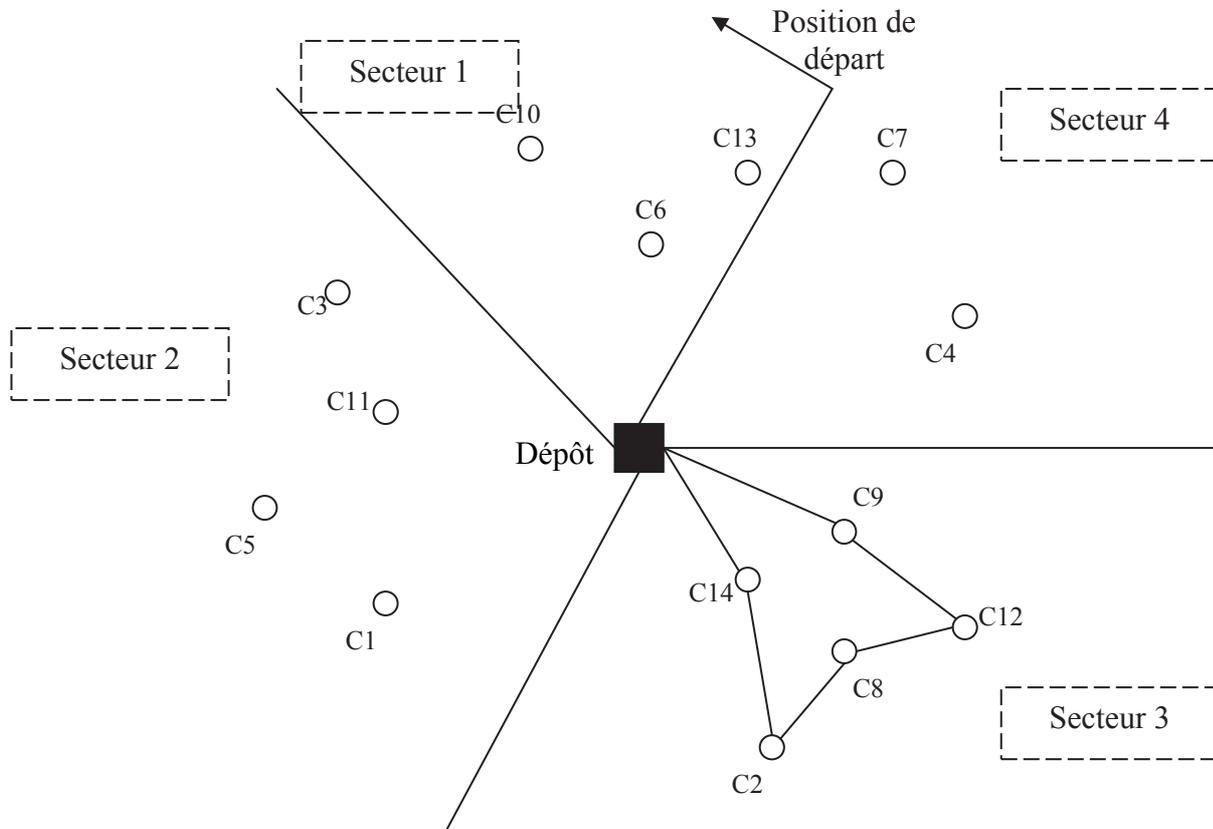


FIG. 1.4 – Groupe en premier, route en second

1.5.2.4 Algorithme route en premier, groupe en second

[Beasley 1983] a utilisé ce genre d'algorithme qui effectue d'abord la construction de cycles comportant de très grandes routes généralement non réalisables. Ces grands parcours sont, par la suite, divisés en parcours plus petits réalisables, formant ainsi des solutions au VRP.

1.5.2.5 Algorithme de recherche locale

L'application de la recherche locale permet de trouver de meilleures solutions dans un voisinage immédiat si le minimum local n'a pas été atteint [Breedam 1996]. Elle consiste à faire des permutations entre arcs à l'intérieur de la même tournée ou entre tournées. Les recherches locales les plus connues sont de la famille des $\lambda - Opt$ ($2 - Opt$, $3 - Opt$, $n - Opt$).

Dans la méthode heuristique d'amélioration de tournée $\lambda - Opt$ ($\lambda = 2, 3, 4, \dots, n$) proposée par [Lin 1965], il s'agit d'enlever λ arcs pour remettre les chaînes associées dans la meilleure combinaison possible. Une variation intéressante est le $\alpha - \lambda - Opt$, où seulement les α plus courts gains sont explorés.

1.5.3 Les métaheuristiques

Les métaheuristiques sont un ensemble de méthodes qui permettent de résoudre des problèmes d'optimisation auxquels les ingénieurs et les décideurs sont régulièrement confrontés.

Une métaheuristique est constituée d'un ensemble de concepts fondamentaux indépendant du problème que l'on cherche à optimiser comme par exemple, la méthode *tabou* et les mécanismes d'intensification et de diversification. Ainsi les métaheuristiques sont adaptables et applicables à une large classe de problèmes.

Ces méthodes, telles que les **algorithmes génétiques**, la méthode de **recherche tabou**, ou le **recuit simulé** apparues au cours des années 1980, sont inspirées d'analogies avec la biologie (algorithme génétique), ou la physique (recuit simulé). Elles sont souvent performantes dans le traitement des problèmes d'"optimisation difficile", c'est-à-dire les problèmes combinatoires - discrets ou continus - pour lesquels nous ne connaissons pas d'algorithme capable de trouver la solution exacte - ou un optimum global - en un temps "raisonnable" [Siarry et al. 2003].

Deux principes généraux sont largement utilisés dans les Métaheuristiques : **voisinage** et **mémoire (population)**

- **Voisinage** : à chaque solution s du problème, nous associons un sous-ensemble $V(s)$ de solutions voisines de s . Ce sous-ensemble peut être statique, comme dans le cas du recuit simulé, ou dynamique ou dépendre du temps t (plus précisément de l'itération à laquelle nous nous trouvons). Notons que la dépendance du voisinage par rapport au temps est étroitement couplée à d'autres principes de base contenus dans la métaheuristique, comme par exemple la liste de tabou dans la méthode de recherche tabou.
- **Mémoire (population)** : quelques méthodes métaheuristiques sont basées sur une population de solutions qui forme une mémoire traduisant l'évolution du processus au cours du temps. Dans ce cas, la méthode traite cette population globalement, et à chaque itération, nous obtenons une nouvelle population basée sur la précédente. C'est le cas notamment des algorithmes génétiques. D'autres méthodes utilisent la notion de mémoire adaptative, elle est basée sur l'enregistrement de statistiques sur les solutions visitées, sur des caractéristiques de ces dernières ou encore sur les modifications apportées aux solutions dans le contexte d'une recherche locale. La recherche avec tabous de même que les colonies de fourmis artificielles sont des métaheuristiques qui sont principalement basées sur des informations statistiques [Taillard 1998].

Maintenant, nous allons présenter les méthodes de résolution les plus connues qui sont utilisées pour la résolution de problème VRP :

1. Algorithmes génétiques
2. Recuit simulé
3. Recherche Tabou
4. Recherche par dispersion
5. Colonie de fourmis

1.5.3.1 Les algorithmes génétiques

Les principes fondamentaux de ces algorithmes ont été exposés par [Holland 1975]. Ces algorithmes s'inspirent du fonctionnement de l'évolution naturelle, notamment la sélection de Darwin, et de la reproduction selon les règles de Mendel. La sélection naturelle, que Darwin appelle

“élément propulseur” de l'évolution, favorise les individus d'une population qui sont le mieux adaptés à un *environnement*. La sélection est suivie de la procréation. Elle est réalisée à l'aide de croisements et de mutations au niveau du patrimoine génétique des individus (ou “génotypes”) qui est constitué d'un ensemble de gènes. Ainsi deux individus “parents”, qui se croisent, transmettent une partie de leur patrimoine génétique à leurs descendants. Le génotype de l'enfant fait que celui-ci est plus ou moins bien adapté à l'environnement. S'il est bien adapté, il a une plus grande chance de se reproduire dans la génération future. Au fur et à mesure des générations, on sélectionne les individus les mieux adaptés, et l'augmentation du nombre des individus bien adaptés fait évoluer la population entière vers un optimum (local ou global).

Dans les algorithmes génétiques, nous essayons de simuler le processus d'évolution d'une population. Nous partons d'une population de N solutions du problème représentées par des individus (chromosomes). Cette population choisie aléatoirement est appelée population parent. Le degré d'*adaptation* d'un individu à l'environnement est exprimé par la valeur de la fonction coût $f(x)$ (ou *fitness*), où x est la solution que l'individu représente. Nous disons qu'un individu est d'autant mieux adapté à son environnement, que le coût de la solution qu'il représente est plus faible (pour un problème de minimisation). Au sein de cette population, intervient alors la sélection au hasard d'un ou deux parents qui produisent une nouvelle solution, à l'aide des *opérateurs génétiques*, tels que le *croisement* et la *mutation*. La nouvelle population, obtenue par le choix de N individus parmi les parents et les enfants de l'étape courante, est appelée génération suivante. En itérant ce processus, nous produisons une population plus riche en individus mieux adaptés. L'algorithme (1.1) montre le Pseudo-Code de l'algorithme génétique de base.

Algorithme 1.1 Pseudo-Code d'un algorithme génétique de base

POP, POP' : deux tableaux de taille N
Initialiser la population POP
Evaluer la population POP
POP' est VIDE
Rechercher $x \mid f(x) = \min_i[f(x_i)], 1 \leq i \leq N$
 $f_{min} = f(x)$
 $x_{min} = x$
REPETER
 Evaluer la population POP
 REPETER //phase de reproduction génétique
 sélection
 croisement
 mutation
 JUSQU'A POP' remplie par les nouveaux individus
 Sélectionner la nouvelle population à partir de (POP, POP')
 Rechercher $x \mid f(x) = \min_i[f(x_i)], 1 \leq i \leq N$
 SI $f(x) < f_{min}$
 $f_{min} = f(x)$
 $x_{min} = x$
 FIN DE SI
JUSQU'A conditions d'arrêt satisfaites
Résultat (x_{min}, f_{min})

Où, POP représente la population courante et POP' représente la population des individus générés par l'étape de reproduction.

Cet algorithme comporte trois phases distinctes :

1. La sélection de la population d'individus la mieux adaptée pour contribuer à la reproduction de la génération suivante (version artificielle de la sélection naturelle) ; elle peut être mise en œuvre sous plusieurs formes algorithmiques.
2. La phase de reproduction, qui exploite essentiellement les opérateurs de croisement et de mutation sur les individus sélectionnés précédemment
3. La stratégie de remplacement des populations parent et enfant par la génération suivante. Elle pourra être mise en œuvre sous plusieurs formes.

Codage Chaque individu de la population est une solution, cet individu est représenté par un *chromosome*. Ce chromosome est constitué de *gènes* qui peuvent prendre des valeurs appelées *allèles*. Il existe plusieurs manières de coder une solution au problème VRP (voir chapitre 3).

Opérateurs de reproduction La phase de reproduction exploite principalement deux opérateurs : le **croisement** et la **mutation**. Elle comporte aussi l'opération de sélection et l'opération de production de la génération suivante, à partir des populations parent et enfant.

– *Sélection*

La sélection consiste à choisir les individus qui vont participer à la reproduction de la population future. La fonction de sélection choisit, de façon déterministe ou selon une méthode probabiliste (roulette, tournoi, etc.), un individu pour qu'il participe à l'étape de reproduction pour former la nouvelle population.

– *Croisement*

Le principal opérateur agissant sur la population des parents est le croisement, qui est appliqué avec une certaine probabilité appelée taux ou probabilité de croisement P_c (typiquement proche de l'unité). Le croisement consiste à choisir deux individus tirés au hasard dans la population courante, et à définir aléatoirement un ou plusieurs points de croisement. Les nouveaux individus sont alors créés en échangeant les différentes parties de chaque chaîne. Cet opérateur permet de bien **exploiter** le domaine de variation des individus, et de diriger la recherche vers des régions intéressantes de l'espace d'étude en utilisant la connaissance déjà présente dans la population courante.

– *Mutation*

L'opération de mutation protège les algorithmes génétiques des pertes prématurées d'informations pertinentes. Elle permet d'introduire une certaine information dans la population, qui a pu être perdue lors de l'opération de croisement. Ainsi elle participe au maintien de la diversité, utile à une bonne **exploration** du domaine de recherche. L'opérateur de mutation s'applique avec une certaine probabilité, appelée taux ou probabilité de mutation P_m , typiquement faible.

– *Élitisme*

Cette méthode consiste à copier un ou plusieurs des meilleurs chromosomes dans la nouvelle génération avec un pourcentage prédéfini appelé "*taux d'élitisme*". Ensuite, on génère le reste de la population selon l'algorithme de reproduction usuel. Cette méthode améliore considérablement les algorithmes génétiques, car elle évite de perdre les meilleures solutions trouvées jusqu'ici [Davis 1991].

L'efficacité de l'algorithme génétique dépend fortement du choix de ces paramètres de contrôle (les paramètres ci-dessus), dont le réglage reste très empirique ; elle dépend aussi fortement de la représentation choisie pour le codage des solutions et des opérateurs de reproduction utilisés. Nous proposons dans cette thèse une approche basée sur les plans d'expériences (voir chapitre 4) pour le réglage des paramètres de l'algorithme génétique. Les principaux paramètres sont les suivants :

1. la taille de la population N ,
2. le type et taux de croisement,
3. le type et le taux de la mutation,
4. le type de la sélection,
5. le taux d'élitisme.

Il existe beaucoup de travaux relatifs au VRP et à ses extensions ; parmi eux nous pouvons citer les travaux de [Rego et al. 1994] qui présentent plusieurs méthodes de résolution (heuristiques et métaheuristiques) du problème VRP et TSP de [Pereira et al. 2002] qui proposent un algorithme génétique avec une nouvelle représentation chromosomique du VRP et de [Prins 2001] et [Baker et al. 2003] qui utilisent l'algorithme génétique comme méthode de résolution avec aussi une nouvelle représentation des solutions, etc.

1.5.3.2 Recuit simulé

Cette méthode trouve ses origines dans la *thermodynamique*. Elle est issue d'une analogie entre le phénomène physique de refroidissement lent d'un corps en fusion, qui le conduit à un état solide de basse énergie. Il faut abaisser lentement la température, en marquant des paliers suffisamment longs pour que le corps atteigne l'*équilibre thermodynamique* à chaque palier de température. Pour les matériaux, cette énergie basse se manifeste par l'obtention d'une structure régulière comme dans les cristaux ou l'acier. L'analogie exploitée par le recuit simulé consiste à considérer une fonction f à minimiser comme fonction d'énergie. Une solution x peut être considérée comme un état donné de la matière dont $f(x)$ est l'énergie. Le recuit simulé exploite généralement le critère défini par l'algorithme de Metropolis (voir le Pseudo-Code dans l'algorithme 1.2) pour l'acceptation d'une solution obtenue par perturbation de la solution courante. Pour une "température" T donnée, à partir d'une solution courante x , on considère une transformation élémentaire qui changerait x en $s(x)$. Si cette perturbation induit une diminution de la valeur de la fonction objectif f , $\Delta f = f(s(x)) - f(x) < 0$, elle est acceptée, ceci dans le cas d'une minimisation. Dans le cas contraire, si $\Delta f = f(s(x)) - f(x) \geq 0$, la perturbation est acceptée avec une certaine probabilité $p = \exp\frac{-\Delta f}{T}$ qui décroît avec le temps [Thangiah et al. 1994].

Le paramètre de contrôle T de cet algorithme est la "Température" du système, qui influe sur la probabilité d'acceptation d'une solution moins bonne. A une température élevée, la probabilité d'acceptation d'un mouvement quelconque tend vers 1 : presque tous les changements sont acceptés. L'algorithme équivaut alors à une marche aléatoire dans l'espace des solutions. Cette température est diminuée lentement au fur et à mesure du déroulement de l'algorithme pour simuler le processus de refroidissement des matériaux, et sa diminution est suffisamment lente pour que l'équilibre thermodynamique soit maintenu (voir le Pseudo-Code de l'algorithme du recuit simulé 1.3) :

La génération des voisins $S(x)$ se fait par la technique de la recherche locale déjà présentée dans ce chapitre.

Algorithme 1.2 Pseudo-Code de la règle de Metropolis

SI $f(s(x)) \leq f(x)$
 $f(x) = f(s(x))$
 $x = s(x)$
SINON
 $p = \exp \frac{-\Delta f}{T}$
 $r =$ solution aléatoire entre $[0, 1]$
 SI $r \leq p$
 $f(x) = f(s(x))$
 $x = s(x)$
 FIN SI
FIN SI

Algorithme 1.3 Pseudo-Code de l'algorithme du recuit simulé

$x =$ solution aléatoire
 $f_{min} = f(x)$
 $x_{min} = x$
Initialiser température T (assez élevée)
REPETER
 REPETER
 générer un voisin $s(x) \in$ voisinage $S(x)$
 appliquer la règle de Metropolis
 SI $f(x) < f_{min}$
 $f_{min} = f(x)$
 $x_{min} = x$
 FIN DE SI
 JUSQU'A équilibre thermodynamique atteint
 décroître température T
JUSQU'A conditions d'arrêt satisfaites

L'efficacité du recuit simulé dépend fortement du choix de ses paramètres de contrôle, dont le réglage reste lui aussi très empirique. Les principaux paramètres de contrôle sont les suivants :

- la valeur initiale de la température,
- la fonction de décroissance de la température,
- le critère de changement de palier de température,
- les critères d'arrêt.

Le rôle de la température T au cours du processus de recuit simulé est très important. Une forte décroissance de température risque de piéger l'algorithme dans un minimum local, alors qu'une faible décroissance au début du processus entraîne une convergence très lente de l'algorithme. Un compromis pour adapter la décroissance de la température à l'évolution du processus consiste à utiliser une variation logarithmique. La loi logarithmique de décroissance de la température, qui assure la convergence théorique du recuit simulé, est la suivante :

$$T_k = \frac{\mu}{\text{Log}(1+k)}$$

Où k est le nombre de paliers de température effectués, et μ est une constante positive. En pratique, on adopte souvent une décroissance géométrique $T_{k+1} = \alpha.T_k$, avec ($0 < \alpha < 1$), car la loi précédente induit un temps de calcul prohibitif.

Nous trouverons dans [Rego et al. 1994] une utilisation de cette méthode pour résoudre le problème VRP.

1.5.3.3 La méthode de recherche tabou

La recherche tabou **RT** est une métaheuristique originalement développée par [Glover 1986]. Elle est basée sur des idées simples, mais elle est néanmoins très efficace. Cette méthode combine une procédure de recherche locale avec un certain nombre de règles et de mécanismes permettant à celle-ci de surmonter l'obstacle des optima locaux, tout en évitant les problèmes de cycle. Elle a été appliquée avec succès pour résoudre de nombreux problèmes difficiles d'optimisation combinatoire dont le problème de routage de véhicules (VRP).

Dans une première phase, la méthode de *recherche tabou* part d'une solution quelconque x appartenant à l'ensemble de solutions X , puis on se déplace vers une solution $s(x)$ située dans le voisinage $S(x)$ de x . L'algorithme va explorer itérativement un sous-ensemble de l'espace de solutions X . Afin de choisir le meilleur voisin $s(x)$ dans le voisinage $S(x)$, l'algorithme évalue la fonction objectif f en chaque point $s(x)$, et retient le voisin qui améliore la valeur de la fonction objectif f ou si ce n'est pas possible celui qui la dégrade le moins.

L'originalité de la méthode de recherche tabou par rapport aux méthodes de recherches locales réside dans le fait que l'on retient le meilleur voisin, même si celui-ci est plus mauvais que la solution d'où l'on vient. Ce caractère autorisant les dégradations de la fonction objectif évite à l'algorithme d'être piégé dans un minimum local. Cependant il induit un risque de création de cycles. En effet, lorsque l'algorithme quitte un minimum local par acceptation de la dégradation de la fonction objectif, il peut revenir sur la même trace déjà suivie, à l'itération suivante. Pour régler ce problème, l'algorithme a besoin d'une mémoire pour conserver la trace des dernières meilleures solutions déjà visitées. Ces solutions sont déclarées *tabous*, d'où le nom de la méthode. Elles sont stockées dans une liste de longueur L donnée, appelée *liste tabou*. Une nouvelle solution n'est acceptée que si elle n'appartient pas à cette liste tabou. Ce critère d'acceptation d'une nouvelle solution évite la formation de cycles de longueur inférieure ou égale à la longueur de la liste tabou. Il dirige l'exploration de la méthode vers des régions du domaine de solutions non

encore visitées. Le Pseudo-Code de l'algorithme tabou classique est présenté dans l'algorithme 1.4.

Algorithme 1.4 Pseudo-Code de la méthode de recherche tabou

```

 $x$  = solution aléatoire
 $f_{min} = f(x)$ 
 $x_{min} = x$ 
TABOU est VIDE // liste de solutions  $s(x)$ , de longueur  $L$ 
REPETER
  Générer 1 solution TEL QUE  $s_i(x) \in$  voisinage  $S(x)$  et  $\{x, s_i(x)\} \notin$  TABOU
   $f(s(x)) = \min_i[f(s_i(x))]$ ,  $1 \leq i \leq N$ 
  Ajouter ( $\{x, s_i(x)\}$ , TABOU)
   $x = s(x)$ 
  SI  $f(x) < f_{min}$ 
     $f_{min} = f(x)$ 
     $x_{min} = x$ 
  FIN DE SI
JUSQU'A conditions d'arrêt satisfaites

```

On élimine à chaque itération la solution tabou la plus ancienne, en la remplaçant par la nouvelle solution retenue. Mais le codage d'une telle liste est encombrant, car il faudrait garder en mémoire tous les éléments qui définissent une solution. Pour pallier cette contrainte, il est possible de remplacer la liste tabou de solutions interdites par une liste de “*transformations interdites*”, en interdisant la transformation inverse d'une transformation faite récemment.

Cette méthode métaheuristique est largement utilisée pour le problème VRP [Rego et al. 1994], [Gulay et al. 1999] et [Rego 2001]. Nous trouvons aussi une version parallèle qui traite ce problème [Caricato et al. 2003].

1.5.3.4 La recherche par dispersion

La recherche par dispersion, ou *scatter search*, n'est pas fondamentalement nouvelle (voir ci-dessous). Elle a été proposée dans un premier temps par [Glover 1977] à la fin des années 70, dans le but de traiter les problèmes linéaires en nombres entiers. Elle n'a cependant pas suscité d'intérêt particulier à l'époque en dépit de concepts pourtant novateurs. Fondamentalement, son principe de fonctionnement présente de nombreuses similitudes avec le schéma des algorithmes génétiques. La métaheuristique fait évoluer une population de solutions à l'aide de quatre opérateurs :

- Un opérateur de sélection
- Un opérateur de combinaison
- Un opérateur de projection
- Un opérateur d'élimination

L'opérateur de sélection détermine quelles solutions vont servir de support à la combinaison. La sélection ne se limite pas à deux solutions, comme c'est le cas pour les algorithmes génétiques. Son arité est variable et souvent supérieure à 2, ce qui lui permet de travailler sur une population répartie en amas distincts (clusters) de solutions. L'opérateur de combinaison vise à produire une nouvelle solution par combinaison des solutions fournies par l'opérateur de sélection. Lorsque la solution viole certaines contraintes, il est nécessaire d'appeler l'opérateur de projection, de

manière à se ramener dans le domaine de solutions réalisables. La mise à jour de la population est assurée par l'opérateur d'élimination, dont l'objectif consiste couramment à favoriser les solutions pertinentes.

Cette métaheuristique a déjà été utilisée pour la résolution d'une extension du problème VRP, qui est le problème **VRPHF** (Vehicle Routing Problem with Heterogeneous Fleet) [Luiz et al. 1998] et le problème **VRPTW** (Vehicle Routing Problem with Time Windows) [Chiang et al. 2004].

1.5.3.5 La colonie de fourmis

Les algorithmes de fourmis sont issus des travaux de Colorni, Dorigo et Maniezzo [Colorni et al. 1992] au début des années 90. Ils reposent sur une analogie avec le comportement collectif des fourmis à s'organiser pour la recherche de nourriture.

Schématiquement, les fourmis explorent leur environnement immédiat en laissant derrière elles des traces chimiques, la *phéromone*. Elles se servent de la phéromone pour se guider et tendent naturellement à suivre les traces existantes. En l'absence de phéromone, leur exploration devient complètement aléatoire. Elles possèdent donc une vision très limitée de leur environnement. Une fois la nourriture trouvée, elles se servent des traces qu'elles viennent de déposer pour retrouver le chemin du retour vers le nid. Durant le trajet, elles laissent de nouveau sur leur passage de la phéromone, en quantité proportionnelle à l'intérêt de la source de nourriture (pondération). Le chemin est donc fortement imprégné de phéromones et constitue une piste de choix pour la colonie de fourmis. Plus la source de nourriture a été jugée intéressante, plus les fourmis auront tendance à suivre ce chemin. Ainsi, peu à peu, les traces vers les sources de nourriture seront de plus en plus marquées. Il est intéressant de noter que les fourmis ont la remarquable propriété de suivre naturellement le chemin le plus court vers la source de nourriture. En effet, [Goss et al. 1989] a observé qu'en présence de deux routes possibles vers la nourriture, les fourmis adoptent rapidement la plus courte. Ceci est dû au fait que le retour à l'embranchement des fourmis ayant pris la bonne option s'opère avant celles ayant effectué le mauvais choix. De fait, dans l'intervalle de temps, la bonne route se trouve favorisée par une plus grande concentration de phéromones. Le biais est ensuite renforcé par le phénomène d'évaporation de la phéromone, qui tend à effacer progressivement les traces les moins fréquentées. Colorni, Dorigo et Maniezzo [Colorni et al. 1992] ont adapté ce principe au domaine de l'Optimisation Combinatoire. Pour ce faire, ils ont associé le voisinage du nid à l'espace des solutions. Chaque solution s'apparente à une source de nourriture dont la qualité est fournie par la fonction d'évaluation. Chaque fourmi est assimilée à un processus répétitif de construction de solutions. La construction est guidée par un ensemble de données globales "*la phéromone*". Cet ensemble est typiquement une mémoire sur les attributs de solution, régulièrement mise à jour par les processus de construction (en fonction de la qualité de la solution) et par un mécanisme simulant l'évaporation de la phéromone. Le problème majeur de ce modèle réside dans sa grande sensibilité vis-à-vis du principe de la phéromone. En fait, la qualité des résultats fournis par la métaheuristique dépend très fortement de l'importance accordée à la phéromone. Lorsque la pondération est trop élevée, le mécanisme de construction tend à favoriser la génération de solutions trouvées dans les itérations précédentes. A l'inverse, lorsqu'il est trop faible, il tend à produire des solutions aléatoires.

Le pseudo-code de l'algorithme de la colonie de fourmis est présenté dans l'algorithme 1.5 :

Nous pouvons améliorer la performance en tenant en compte des modifications suivantes qui touchent pratiquement tous les aspects de la métaheuristique :

Algorithme 1.5 Pseudo-Code de la méthode de colonie de fourmis

-
1. Initialiser les traces,
 2. Tant qu'un critère d'arrêt n'est pas satisfait, répéter en parallèle pour chacune des p fourmis :
 - (a) Construire une nouvelle solution à l'aide des informations contenues dans les traces et une fonction d'évaluation partielle.
 - (b) Evaluer la qualité de la solution.
 - (c) Mettre à jour les traces.
-

1. le processus de génération de solutions s'est vu intégrer une phase d'amélioration itérative afin de fournir des solutions de meilleure qualité.
2. l'actualisation des traces a évolué pour permettre l'application explicite de stratégies de fouille ou de prospection. Seules les meilleures solutions donnent lieu à une mise à jour des traces.
3. la coordination des différentes fourmis a été sensiblement améliorée par l'apparition d'une fourmi particulière, la reine. Cette dernière prend en charge la gestion de la mémoire collective de la colonie (les traces) et prend les décisions stratégiques en fonction de l'évolution globale de la recherche. Elle constitue de fait une sorte d'agent "intelligent".

L'application de ces différents aménagements du modèle initial a permis d'augmenter sensiblement les performances de la métaheuristique. Ceci est particulièrement vrai pour le problème VRP [Gambardella et al. 2003].

La complexité du VRP exige des méthodes métaheuristicques pour la plupart des exemples réels. Dans les sections précédentes, nous avons largement examiné l'évolution des méthodologies métaheuristicques pour le problème VRP. La qualité des solutions obtenues avec ces différentes techniques est bien souvent meilleure que les heuristiques traditionnelles. En même temps, les métaheuristicques exigent plus de temps de calcul et sont plus complexes à mettre en application et à calibrer [Bräysy et al. 2001].

Le rapport de recherche du GERAD [Cordeau et al. 2004] compare les 9 meilleures méthodes actuelles pour le VRP. Ces méthodes sont :

1. **GTS** (Granular Tabu Search) [Toth et al. 2003] : Cette méthode utilise une recherche tabou et élimine les tronçons de taille supérieure à un seuil de granularité dans le graphe représentant la région de service.
2. L'heuristique **VRTR** (Variable-length neighbourhood list Record-to-Record) [Li et al. 2004] basée sur l'heuristique des gains de [Clarke et al. 1964] et l'opérateur de recherche locale 2-Opt.
3. **UTSA** (Unified Tabu Search Algorithm) [Cordeau et al. 1997] : Cette méthode utilise principalement la recherche tabou, elle utilise seulement une solution initiale et à partir de cette solution elle construit la liste tabou en utilisant l'opérateur (client réinsertion).
4. **VLNS** (Very Large Neighbourhood Search) [Ergun et al. 2003] : Cette méthode est basée sur l'opérateur de recherche locale 2-Opt, appliqué sur plusieurs routes.

5. **L'algorithme génétique hybride** de [Prins 2004] : Cette méthode hybride utilise un algorithme génétique combiné avec une recherche locale. Le chromosome est représenté par une séquence de clients et un algorithme de plus court chemin est utilisé pour découper de manière optimale cette séquence en tournées. L'opérateur de mutation consiste en une recherche locale (client réinsertion, tronçons réinsertion, 2-Opt, 2-Opt sur plusieurs routes, etc) appliquée sur l'ensemble des tournées.
6. L'algorithme "**Bone Route**" proposé par [Tarantilis et al. 2002] est basé sur le couplage d'une procédure constructive (la méthode des gains modifiée) pour obtenir une solution initiale, suivie d'une recherche tabou qui utilise l'opérateur 2-Opt pour obtenir les solutions voisines.
7. **AGES** (Active Guided Evolution Strategies) [Mester et al. 2005] combine des stratégies d'évolution avec l'opérateur de recherche locale 2-Opt.
8. **L'algorithme génétique hybride** [Berger et al. 2004] qui est basé sur l'utilisation d'un algorithme génétique couplé avec une recherche locale (2-Opt, etc).
9. L'heuristique **D-Ants** [Reimann et al. 2004], cette méthode utilise l'heuristique des gains modifiée avec la méthode des colonies de fourmis.

La comparaison entre ces méthodes montre que l'algorithme génétique de [Prins 2004], AGES [Mester et al. 2005] et Bone Route [Tarantilis et al. 2002] sont les meilleures pour toutes les instances des benchmarks traités (les 14 instances de Christofides et les 20 instances à grande taille de Golden).

1.6 Le problème VRPTW

Le Problème de Tournées de Véhicules avec Fenêtre de Temps (Vehicle Routing Problem with Time Windows - VRPTW) [Solomon 1987], [Duhamel 2001], constitue une généralisation du VRP dans la mesure où nous introduisons en plus une contrainte temporelle sur le service demandé. Chaque client dispose d'une fenêtre de temps à l'intérieur de laquelle il désire être servi. Le dépôt central possède également une fenêtre de temps que nous désignons couramment comme horizon de service ou temps d'ouverture de la journée. Son rôle est de fixer une plage horaire durant laquelle les véhicules peuvent effectuer leur tournée. Ces contraintes temporelles vont rendre nécessaire l'utilisation de plusieurs véhicules pour satisfaire l'ensemble des clients sur l'horizon de service. On peut vouloir borner le nombre de véhicules à utiliser et dans ce cas des clients risquent de ne pas être servis.

Bien que simple en apparence, ce problème est notablement très difficile à résoudre. En fait, il a été montré que le problème VRP classique était NP-difficile, et que ce résultat pouvait être étendu au VRPTW. Ainsi, s'il est tout à fait possible de déterminer une solution optimale pour des instances de petite taille, cela devient rapidement irréalisable pour des instances de moyenne ou de grande taille. Or la plupart des problèmes de la vie réelle se situent justement dans cette seconde catégorie. En dépit du caractère fortement restrictif de sa définition, le problème VRPTW conserve un pouvoir descriptif important. Il permet de modéliser un panel étendu d'applications réelles [Duhamel 2001] :

- **service postal** : il s'agit d'établir des tournées pour les facteurs de manière à distribuer le courrier le plus efficacement possible, mais on peut aussi réaliser des tournées pour effectuer le ramassage de courrier dans les boîtes aux lettres publiques.

- **coopératives agricoles** : il s’agit de mettre en place des tournées de véhicules pour collecter la production de différents exploitants agricoles. Ceci peut concerner le lait, les produits céréaliers, les produits maraîchers ou encore le ramassage d’animaux par exemple.
- **service bancaire** : il s’agit de déterminer des routes pour les convoyeurs de fond de manière à réaliser la collecte des recettes des commerçants et des grandes surfaces.
- **ramassage scolaire** : il s’agit d’établir les itinéraires de bus pour amener les écoliers des points de collecte vers l’école le matin et inversement le soir.
- etc.

Ces quelques exemples permettent d’illustrer la profonde implication de ce problème dans la vie courante.

1.6.1 La formulation du VRPTW

Nous formulons le problème VRPTW en utilisant la formulation de VRP classique déjà exprimée et en y ajoutant les variables, les constantes et les équations suivantes [LeBouthillier 2000] :

Variables à déterminer :

1. a_i = instant d’arrivée chez le client $i \in C$.
2. b_i = instant de début de service chez le client $i \in C$.
3. b_0^v = instant auquel le véhicule v quitte le dépôt.
4. $b_{n_c+1}^v$ = instant auquel le véhicule v retourne au dépôt.
5. w_i = temps d’attente chez le client $i \in C$.

Constantes connues :

1. e_i = borne inférieure de la fenêtre de temps du client $i \in C$.
2. l_i = borne supérieure de la fenêtre de temps du client $i \in C$.
3. $c_{i,j}$ = coût du déplacement de i à j : $i, j \in C$.
4. $t_{i,j}$ = le temps de parcours entre les deux clients i et j , $i, j \in C$.
5. s_i = temps de service chez le client $i \in C$.

L’attente est permise lorsqu’un véhicule arrive trop tôt chez le client j après que le service soit fini chez le client $i \in C$, autrement dit, avant e_j . Le temps auquel le service débute chez le client $j \in C$ se définit comme étant $b_j = \max\{e_j, a_j\}$ où $a_j = \{b_i + s_i + t_{i,j}\}$ et le temps d’attente chez le client j comme étant $w_j = b_j - a_j$. Il nous est possible d’écrire les contraintes supplémentaires de la formulation de VRP pour formuler le problème VRPTW comme suivantes :

$$x_{i,j}^v = 1 \Rightarrow b_i + s_i + t_{i,j} \leq b_j, \forall i, j \in C, v \in V \quad (1.11)$$

$$x_{0,j}^v = 1 \Rightarrow b_0^v + t_{0,j} \leq b_j, \forall j \in C, v \in V \quad (1.12)$$

$$x_{i,n+1}^v = 1 \Rightarrow b_i + s_i + t_{i,n+1} \leq b_{n+1}^v, \forall i \in C, v \in V \quad (1.13)$$

$$e_i \leq b_i \leq l_i, \forall i \in C \quad (1.14)$$

$$e_0 \leq b_0^v \leq l_0, \forall v \in V \quad (1.15)$$

$$e_{n+1} \leq b_{n+1}^v \leq l_{n+1}, \forall v \in V \quad (1.16)$$

Le temps d'utilisation réel des véhicules dans la solution (X) est :

$$\text{temps horaire}(X) = \sum_{v \in V} (b_{n_c+1}^v - b_0^v) \quad (1.17)$$

Les contraintes (1.11 - 1.16) définissent les contraintes temporelles. La mesure (1.17) permet d'identifier le temps total d'utilisation des véhicules.

Dans un VRP, la définition de distance entre deux clients qui nous vient naturellement à l'esprit est la distance que requiert le parcours direct entre ces deux clients. Cette distance est la même dans les deux sens, autrement dit, la distance entre le client i et j est la même entre j et i , donc la matrice des distances entre les clients est symétrique. Dans les algorithmes utilisés pour le VRPTW, cette même métrique de distance est généralement utilisée.

1.6.2 Revue de la littérature VRPTW

Dans la littérature, nous constatons que le problème VRPTW est le plus étudié parmi les extensions du VRP. Comme pour le problème VRP, nous avons deux catégories de méthodes de résolution qui traitent le VRPTW :

1.6.2.1 Méthodes exactes

Les approches basées sur des *méthodes exactes* ont tardé à apparaître. Une des raisons principales tient à la difficulté de trouver un modèle suffisamment performant pour exprimer le VRPTW. En particulier les contraintes de capacité et de fenêtre temps reposent sur un mécanisme récursif : l'application des contraintes sur un client dépend du résultat sur le client précédent. Nous trouverons dans l'articles de [Cordeau et al. 2002] une excellente revue de littérature sur le sujet. Nous pouvons classer les différentes approches en quatre grandes catégories :

- **Programmation dynamique** : [Kolen et al. 1987] reprend le principe de la méthode de relaxation mise au point par [Christofides et al. 1981] pour le problème VRP. Les décisions de branchement portent sur l'affectation des clients aux tournées. Cette approche leur a permis à l'époque de résoudre optimalement des problèmes comptant jusqu'à 15 clients. Notons que c'est l'un des travaux les plus anciens sur le VRPTW.
- **Génération de colonnes** : [Desrochers et al. 1992] propose l'application de la technique de génération de colonnes sur la relaxation du VRPTW exprimé comme problème de set covering. Cette approche, très performante, leur autorise la résolution d'instances comprenant jusqu'à 100 clients.
- **Décomposition Lagrangienne** : cette approche repose sur un schéma de partage des variables. Ceci permet d'obtenir des bornes inférieures pour la solution, que l'on peut ensuite intégrer dans un mécanisme de Branch and Bound. [Jörnsten et al. 1986], [Madsen 1990], et [Fisher et al. 1997] proposent chacun un moyen différent de réaliser la décomposition Lagrangienne. Des instances de 100 clients ont ainsi pu être résolues.

- **Relaxation Lagrangienne** : elle a été utilisée par Kohl et Madsen [Kohl et al. 1997] ainsi que par Fisher [Fisher et al. 1997]. Cette approche permet de résoudre quelques problèmes à 100 clients.

Actuellement, aucune approche basée sur le Branch and Cut ne semble avoir été tentée pour le VRPTW. La raison principale réside certainement dans la très grande complexité à pouvoir exhiber des facettes du polyèdre des contraintes.

Notons que les trois premières catégories font appel à un sous-problème de “Plus Court Chemin avec Capacité et Fenêtre de Temps”. Il est utilisé soit pour la génération des colonnes, soit pour mettre à jour la relaxation Lagrangienne. Bien qu’il soit reconnu NP-complet, ce sous-problème peut être aisément résolu par des algorithmes pseudo-polynômiaux à base de programmation dynamique.

1.6.2.2 Méthodes approchées

Du fait de sa complexité intrinsèque, le VRPTW constitue un champ d’investigation particulièrement intéressant et fécond pour les méthodes approchées. [Desrosiers et al. 1995], ainsi que [Fisher 1995] présentent des états de l’art précis sur le sujet. Nous nous contenterons de relever les contributions les plus significatives dans chaque catégorie :

1. Heuristiques de **construction de tournées** : reprenant les travaux précurseurs de Clarke et Wright [Clarke et al. 1964] sur le problème VRP, Solomon [Solomon 1987] décrit quelques méthodes de construction de tournées reposant sur l’insertion séquentielle des clients. [Potvin et al. 1993], puis [Russell 1995] proposent des méthodes de construction de tournées en parallèle. Notons qu’une analyse dans le pire des cas a été développée par Solomon [Solomon 1986] pour certaines heuristiques simples.
2. Heuristiques **d’amélioration de tournées** : ces méthodes partent d’une solution initiale obtenue par une heuristique du type précédent et l’améliorent progressivement en procédant à des échanges d’arcs. Les différents opérateurs proposés sont presque tous basés sur le principe du voisinage $\lambda - Opt$ [Lin 1965] qui consiste à échanger deux suites de clients de taille λ , tout en intégrant les contraintes de fenêtre de temps et de capacité.
3. Heuristiques **composites** : à la différence des heuristiques d’amélioration de tournées, qui nécessitent la connaissance d’une solution initiale, les heuristiques composites combinent une phase de construction et une phase d’amélioration des tournées. Elles sont d’inspiration relativement récente bien qu’on puisse remonter aux travaux de [Gillett et al. 1974]. Le lecteur pourra se référer aux travaux de [Derigs et al. 1993], [Kontoravdis et al. 1992], ou encore [Russell 1995] pour plus de détails.
4. Heuristiques **basées sur un principe exact** : elles forment une classe à part dans la mesure où elles sont déduites de méthodes exactes. Elles reprennent en fait un schéma d’optimisation exact, avec quelques aménagements, en particulier au niveau des critères décisionnels. Ainsi [Koskosidis et al. 1992] utilise des critères heuristiques pour réaliser l’assignation des clients aux tournées. Après décomposition, ces dernières sont ensuite obtenues via une méthode exacte de routage.
5. **Métaheuristiques** : elles constituent la branche la plus récente dans le domaine des méthodes approchées et sont réputées pour leur excellent niveau de performances. [Potvin et al. 1996b] comme [Rochat et al. 1995] ont proposé une résolution par recherche tabou. [Russell 1995] a eu recours au recuit simulé tandis que [Kontoravdis et al. 1995]

a appliqué GRASP² (Greedy Randomized Adaptive Search Procedure). De nombreux travaux ont aussi porté sur l'utilisation d'algorithmes génétiques pour résoudre ce problème tels que ceux de [Blanton et al. 1993], [Potvin et al. 1996a] ainsi que ceux de [Thangiah 1995].

Notons également le travail de [Bramel et al. 1993]. En se basant sur une analyse probabiliste du problème VRPTW, ils ont mis au point des heuristiques convergeant asymptotiquement vers la solution optimale. L'intérêt théorique de cette approche est indéniable puisque l'on dispose d'une garantie sur la qualité du résultat. Malheureusement, à l'instar du théorème de [Hajek 1988] pour le recuit simulé, il y a peu de chances pour que ce résultat puisse déboucher sur un algorithme combinant les avantages des méthodes exactes et approchées.

De ces différentes approches, il ressort que les métaheuristiques sont celles qui tirent le mieux leur épingle du jeu. Elles offrent une qualité de résultats bien supérieure aux autres approches, au prix il est vrai de temps de calculs parfois conséquents.

Enfin, nous remarquons que dans la littérature, les travaux traitant le problème VRPTW n'ont pas utilisé la même fonction objectif ; dans la table (1.2) nous présentons quelques références sur le problème VRPTW avec leur fonction objectif³.

1.7 Conclusion

Dans ce chapitre, nous avons présenté le problème d'élaboration de tournées de véhicules (le problème VRP) et son extension avec fenêtres de temps VRPTW. Nous avons introduit les méthodes de résolution utilisées pour ces deux problèmes.

Les méthodes métaheuristiques fournissent généralement des solutions de bonne qualité. C'est pourquoi nous les avons privilégiées dans la suite de nos travaux.

²le principe de GRASP consiste simplement à l'utilisation alternée de deux mécanismes, la construction d'une solution faisable et la recherche locale pour trouver l'optimum local

³En ordre (première, deuxième, troisième, . . . , etc. objectif)

| Référence | Fonction objectif |
|--------------------------|---------------------------------------------------------------------------------------------------------------------------------------|
| [Solomon 1987] | 1 - Minimiser le nombre de véhicules, 2 - Minimiser la distance totale parcourue, 3 - Minimiser le temps d'attente des clients. |
| [Desrochers et al. 1992] | 1 - Minimiser la distance totale parcourue. |
| [Potvin et al. 1993] | 1 - Minimiser le nombre de véhicules, 2 - Minimiser le temps de parcours. |
| [Thangiah et al. 1994] | 1 - Minimiser le nombre de véhicules, 2 - Minimiser la distance totale parcourue. |
| [Russell 1995] | 1 - Minimiser le nombre de véhicules, |
| [Rochat et al. 1995] | 1 - Minimiser le nombre de véhicules, 2 - Minimiser la distance totale parcourue. |
| [Potvin et al. 1996b] | 1 - Minimiser le nombre de véhicules, 2 - Minimiser la distance totale parcourue. |
| [Potvin et al. 1996a] | 1 - Minimiser le nombre de véhicules, 2 - Minimiser la distance totale parcourue. |
| [Chiang et al. 1996] | 1 - Minimiser le nombre de véhicules, 2 - Minimiser la distance totale parcourue. |
| [Badeau et al. 1997] | 1 - Minimiser la distance totale parcourue. |
| [Kohl et al. 1997] | 1 - Minimiser la distance totale parcourue. |
| [Taillard et al. 1997] | 1 - Minimiser la distance totale parcourue, 2 - Minimiser le retard total chez les clients. |
| [Kilby et al. 1998b] | 1 - Minimiser la distance totale parcourue. |
| [Hombberger et al. 1999] | 1 - Minimiser la distance totale parcourue, 2 - Minimiser le nombre de véhicules. |
| [Sushil et al. 1999] | 1 - Minimiser la distance totale parcourue, 2 - Minimiser le retard total chez les clients, 3 - Minimiser le retard au dépôt. |
| [Bräysy et al. 2000] | 1 - Minimiser la distance totale parcourue. |
| [Tan et al. 2001] | 1 - Minimiser la distance totale parcourue. |

TAB. 1.2 – Les fonctions objectifs choisies pour la résolution du VRPTW

Chapitre 2

Le problème d'élaboration dynamique de tournées de véhicules DVRP⁴

Sommaire

| | | |
|------------|--------------------------------------------------------------------|-----------|
| 2.1 | Introduction | 33 |
| 2.2 | Le problème d'élaboration de tournées de véhicule dynamique | 35 |
| 2.3 | Classification du VRP | 35 |
| 2.3.1 | VRP statique | 37 |
| 2.3.2 | VRP dynamique | 37 |
| 2.4 | Problème statique vs. problème dynamique | 38 |
| 2.5 | Formulation | 39 |
| 2.6 | Le degré de dynamisme | 39 |
| 2.6.1 | Dynamisme sans fenêtres de temps (dod) | 39 |
| 2.6.2 | Degré de dynamisme effectif - edod | 40 |
| 2.6.3 | Dynamisme avec fenêtres de temps - edod-tw | 41 |
| 2.7 | Classification du DVRP | 41 |
| 2.7.1 | Problèmes faiblement dynamiques | 41 |
| 2.7.2 | Problèmes modérément dynamiques | 43 |
| 2.7.3 | Problèmes fortement dynamiques | 43 |
| 2.8 | Le DVRP dans la littérature | 43 |
| 2.8.1 | Méthodes séquentielles | 43 |
| 2.8.2 | Méthodes parallèles | 50 |
| 2.9 | Conclusion | 52 |

2.1 Introduction

Ces dix dernières années, les problèmes de gestion dynamique de tournées sont devenus un domaine de recherche de plus en plus étudié. Aujourd'hui, le traitement des données en temps réel est possible grâce aux avancées des technologies de l'information et de la communication. En particulier, les systèmes de positionnement globaux (GPS), les systèmes d'information géographiques (GIS), les systèmes intelligents d'autoroutes et autres systèmes embarqués, ont augmenté l'intérêt des méthodes et des stratégies pour la résolution du problème d'élaboration de tournées

⁴Dynamic Vehicle Routing Problem

de véhicules dans un contexte “*en ligne*” où l'échange d'information peut se faire en temps réel avec un outil d'aide à la décision.

En effet, dans beaucoup de situations pratiques, les données du problème VRP ne sont pas complètement connues au moment où celui-ci doit être résolu. Ces données peuvent concerner par exemple les clients à servir dans la journée. Seul un sous ensemble de ces clients peut être connu au moment où la planification est établie. Ceci signifie qu'il peut y avoir en cours de journée de nouveaux clients qui demandent un service après que la planification ait été déjà effectuée et alors que les véhicules sont déjà en route pour servir les clients précédemment planifiés. Ces nouveaux clients doivent alors pouvoir être insérés dans la solution courante (i.e. planification existante). Ce type de problème est un problème dynamique, tandis que les problèmes évoqués dans le chapitre précédent sont qualifiés de problèmes statiques.

Le problème DVRP (Dynamic Vehicle Routing Problem) fait partie de la famille des problèmes de transports dynamiques (DTP). Cette famille comporte différents types de problèmes de transports dynamiques [Bianchi 2000] tels que par exemple : le problème dynamique du plus court chemin (Dynamic Shortest Path), la gestion dynamique de flotte de véhicules (Dynamic Fleet Management), la gestion dynamique de trafic aérien (Dynamic Air Traffic Control), etc. Ce groupe de problèmes (DTP) appartient à une famille plus grande de problèmes dynamiques. Cette famille, représentée en figure (2.1), contient tous les types des problèmes dynamiques de la vie courante comme par exemple les problèmes de cheminement dans les réseaux (Network Routing Problems).

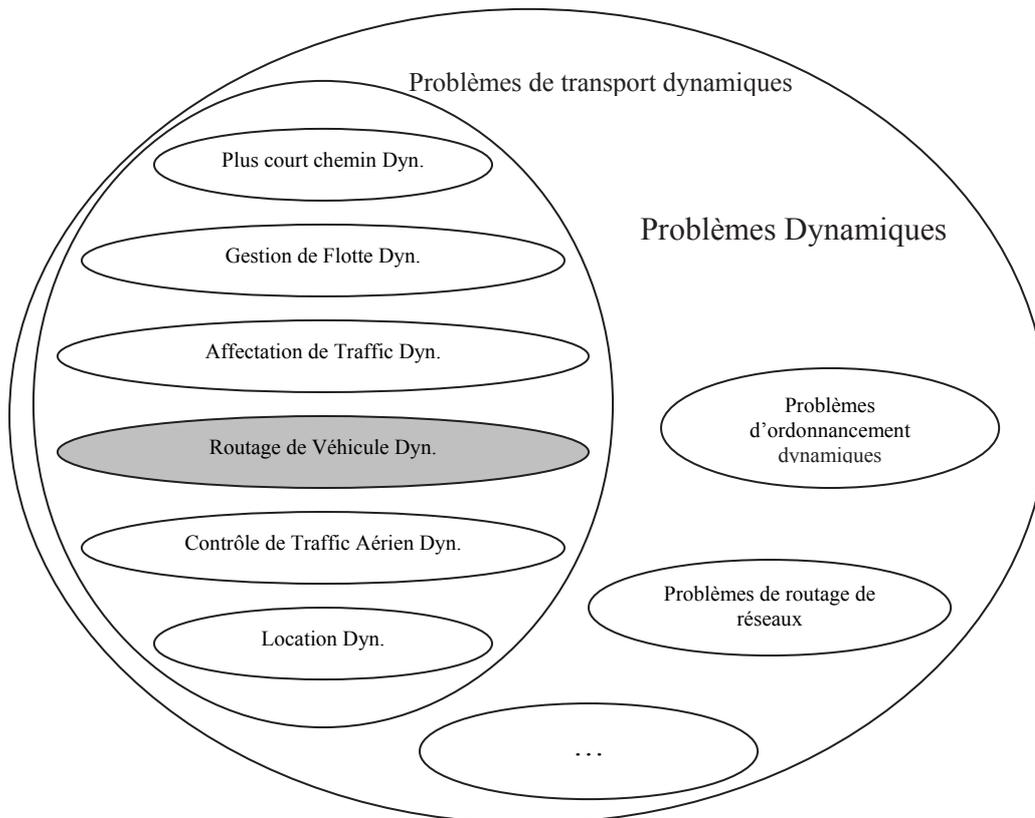


FIG. 2.1 – Les problèmes dynamiques

Ce chapitre est divisé de la manière suivante : nous présentons dans premier temps le pro-

blème du VRP dynamique, et les différences entre ce problème et sa version statique. Puis, nous rappelons la notion de degré de dynamisme d'un problème et la classification d'un problème par rapport à son degré de dynamisme. Finalement, nous finirons par une revue de la littérature sur ce problème.

2.2 Le problème d'élaboration de tournées de véhicule dynamique

Dans cette section, le problème dynamique d'élaboration de tournées de véhicules (DVRP) sera défini et un exemple simple de scénario DVRP sera présenté.

Dans le chapitre précédent, nous avons présenté le problème VRP et ses extensions. Nous disons que ce problème est statique puisque :

1. Toute l'information (données du problème) concernant la planification des routes est supposée connue par le planificateur avant que le processus de planification ne commence.
2. De plus, cette information ne change pas après que les routes aient été construites.

Cette information inclut tous les attributs des clients tels que la position géographique des clients, le temps de service à passer sur place et la demande (i.e. quantité à collecter ou à livrer) de chaque client. En outre, les informations concernant les temps de trajets nécessaires pour relier les clients à servir doivent être connues ou calculables par le planificateur.

Par contre dans le cas dynamique, nous pouvons définir le problème DVRP comme un problème VRP classique mais avec les différences suivantes [Larsen 2001] :

1. L'information concernant la planification des routes n'est pas connue entièrement par le planificateur quand le processus de planification commence. Autrement dit, certaines données du problème dépendent explicitement du temps (i.e. apparition d'un nouveau client).
2. L'information peut changer après que les routes initiales aient été construites.

Dans la figure (2.2), nous avons un exemple simple de problème dynamique où deux véhicules doivent servir un ensemble de clients. Les clients statiques sont représentés par des nœuds noirs, alors que les clients dynamiques (i.e. nouveaux clients) sont représentés par les nœuds blancs. Les arcs en trait fin représentent la route planifiée pour chaque véhicule. Les deux arcs en trait plein épais indiquent les trajets en cours des véhicules lorsque les demandes dynamiques sont reçues. Les nouveaux clients doivent être insérés dans les routes déjà construites en tenant compte des clients non encore visités (en trait pointillé).

2.3 Classification du VRP

Dans la littérature [Ghiani et al. 2003], nous pouvons distinguer quatre types de problèmes de VRP (voir la figure 2.3). Un problème est dit *statique* si toutes les données nécessaires à la planification ne dépendent pas du temps, sinon il est *dynamique*. Le caractère *déterministe* traduit le fait que toutes les données sont parfaitement connues.

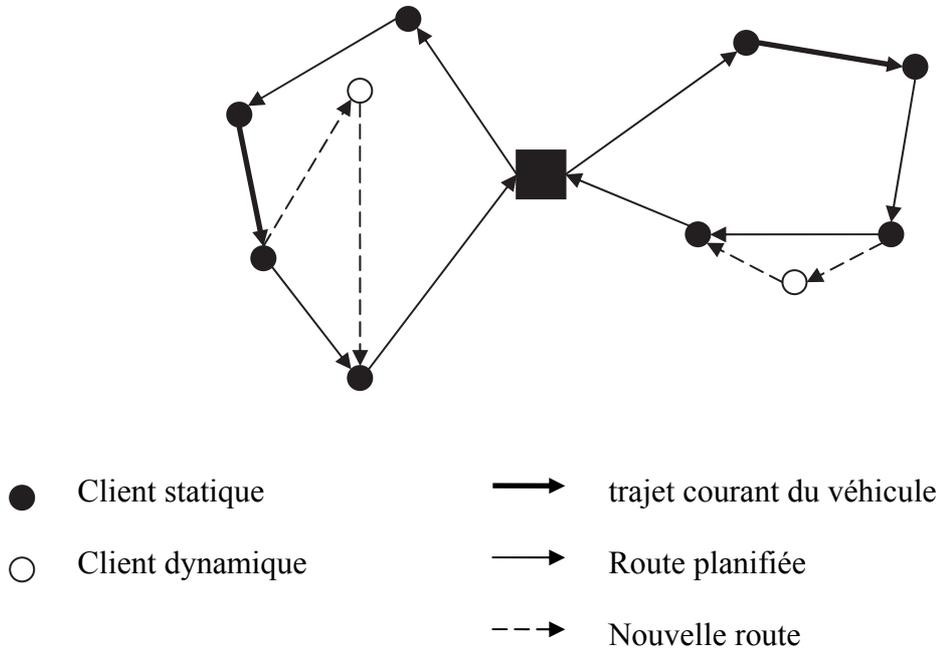


FIG. 2.2 – Le problème DVRP

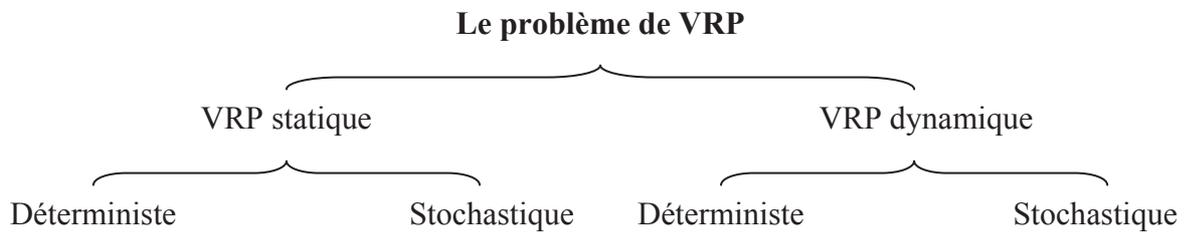


FIG. 2.3 – Classification du VRP

2.3.1 VRP statique

Ce type de problème peut être *déterministe* ou *stochastique*. Dans le VRP *statique déterministe* ou VRP classique, toutes les données sont connues à l'avance et le temps n'est pas pris en compte explicitement. Par contre dans le problème *statique stochastique*, l'incertitude sur les données peut être modélisée par des variables stochastiques. L'incertitude peut porter sur la quantité de la demande, le temps de service chez le client, ou le temps de trajet d'un tronçon routier. Si les données sont incertaines, il devient impossible de satisfaire les contraintes avec (100%) de réussite. Si l'incertitude affecte uniquement les contraintes mais que la fonction objectif est *déterministe*, il est possible que les contraintes soient satisfaites avec une probabilité donnée ou un taux de réussite donné. Dans une approche générale [Gendreau et al. 1999b], une solution est construite sur la base de ces données incertaines. Cette solution représente une permutation des clients à visiter par un seul véhicule (une seule tournée). On doit alors prendre une décision, après chaque client servi, qui consiste soit à retourner au dépôt soit à se déplacer au client suivant. Une mauvaise décision peut entraîner pendant l'exécution de la tournée, un défaut de capacité et conduire à un retour au dépôt prématuré. (voir la figure 2.4).

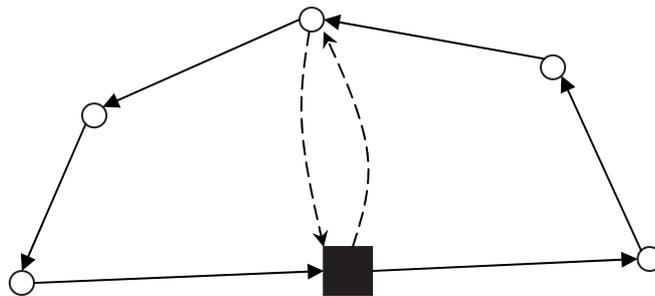


FIG. 2.4 – Le cas d'échec

2.3.2 VRP dynamique

Un problème dynamique peut également être *déterministe* ou *stochastique*. Dans les *problèmes dynamiques déterministes*, les données sont connues à l'avance avec certitude et seuls quelques éléments d'information dépendent du temps [Larsen 2001]. Dans des *problèmes dynamiques stochastiques* [Bent et al. 2003], également connus sous le nom du problème VRP en *temps réel*, les données incertaines sont représentées par des variables stochastiques. Par exemple, l'arrivée des demandes des clients peut suivre une distribution de type *loi de Poisson*.

Les événements qui mènent à une modification de la planification peuvent être :

1. L'arrivée d'une nouvelle demande.
2. La panne d'un véhicule.
3. L'apparition d'une congestion de trafic suite à des travaux ou à un accident.

Chaque événement doit être traité selon les politiques réglées par le planificateur de la flotte de véhicules. En général, quand une nouvelle demande est reçue, il doit décider s'il peut la satisfaire le jour même, ou s'il doit la retarder pour le jour suivant ou plus radicalement la rejeter définitivement. Si la demande est acceptée, elle est temporairement affectée à une position dans

la tournée existante d'un véhicule. Par conséquent, c'est au moment où un véhicule atteint sa destination courante que le planificateur lui assigne sa prochaine destination.

2.4 Problème statique vs. problème dynamique

Dans cette section les différences entre le problème VRP statique et dynamique seront discutées. Puis, nous donnons un bref résumé des principales caractéristiques permettant de différencier les deux types de problèmes [Larsen 2001].

1. **Le temps est essentiel** : Dans un problème statique, le temps peut ou ne peut pas être important. Mais pour un problème dynamique, le temps est toujours essentiel. Le planificateur doit à tout moment savoir la position de tous les véhicules, la liste des clients servis et la liste des clients non encore servis quand une nouvelle demande de service est reçue.
2. **L'information future peut être imprécise ou inconnue** : Dans un problème statique on suppose que toute l'information est connue et ceci avec la même qualité. Dans un problème dynamique réel, le futur n'est jamais connu avec certitude. Au mieux des informations probabilistes sur le futur peuvent être fournies.
3. **Les événements à court terme sont plus importants** : Dans un problème dynamique, il est préférable que le planificateur se concentre sur le traitement des événements à court terme.
4. **Les mécanismes de mise à jour de l'information sont essentiels** : Presque toutes les données d'entrées relatives à un problème VRP dynamique sont sujettes à des changements pendant la journée de travail. Il est donc essentiel que des mécanismes de mise à jour de l'information soient intégrés dans la solution. Ces mécanismes de mise à jour ne sont pas nécessaires dans les problèmes statiques.
5. **Réordonnement et prises de décisions sont justifiés** : Dans un problème dynamique, un nouvel événement peut impliquer que les décisions prises par le planificateur deviennent non optimales. Ceci le force à réordonner ou même à attribuer de nouveaux véhicules afin de répondre à la nouvelle situation.
6. **Un temps de calcul rapide est nécessaire** : Contrairement aux problèmes statiques, où le planificateur peut attendre quelques heures afin d'obtenir une solution de haute qualité qui dans certains cas peut être optimale, pour les problèmes dynamiques ceci n'est pas possible. Dans ce cas présent, le planificateur souhaite connaître la solution au problème posé le plus rapidement possible, de préférence dans les minutes ou les secondes qui suivent l'apparition d'un événement.
7. **Les mécanismes de retard des demandes sont essentiels** : Cela signifie qu'en raison des caractéristiques géographiques défavorables d'une demande, le planificateur peut être obligé de retarder indéfiniment ou de rejeter cette demande.
8. **La fonction objectif peut être différente** : Les objectifs traditionnels des problèmes statiques tels que la minimisation de la distance totale parcourue peuvent être sans signification dans les problèmes dynamiques. Par exemple, dans certains problèmes dynamiques (systèmes d'urgence), l'objectif prioritaire est de minimiser le temps de réponse du client (intervalle de temps entre la réception de la demande et le début de service).

9. **La file d'attente est importante** : Si le taux de demandes excède un certain seuil, le système deviendra vite encombré et le système aura besoin d'une file d'attente pour garder toutes les demandes reçues.

2.5 Formulation

Dans le chapitre précédent, nous avons présenté la formulation du problème VRP et du VRPTW proposée par [Solomon 1987], [Antes et al. 1995], [Solomon et al. 1988]. Nous allons prolonger cette formulation pour que nous puissions spécifier simplement le problème dynamique. En fait, nous avons à formuler seulement l'évènement "réception d'une nouvelle demande".

Variable de décision, à déterminer :

- tr_i = l'instant de réception de la demande du client i .

Pour étendre la formulation du VRPTW vers le DVRPTW, nous ajoutons les contraintes suivantes :

1. Contrainte 1 : $0 \leq tr_i < T, \forall i \in C$, T est l'instant de fermeture du dépôt.
2. Contrainte 2 : $tr_i = \alpha.e_i, \alpha \in [0, \dots, 1], \forall i \in C$.

Ces deux contraintes décrivent que l'instant de réception d'une demande est limité par le temps d'ouverture et de fermeture du dépôt, et que cet instant tr_i est toujours inférieur ou égal à la borne inférieure de la fenêtre de temps du client i .

2.6 Le degré de dynamisme

Contrairement à un problème statique d'élaboration de tournées de véhicules nous supposons que la performance dépend non seulement du nombre de clients et de la distribution spatiale de ces derniers, mais également du nombre d'évènements dynamiques et des instants où ces évènements ont lieu réellement. Par conséquent, une mesure pour décrire le dynamisme du système serait d'une grande utilité quand nous voulons examiner la performance d'un algorithme spécifique dans des conditions variables.

Le but de cette section est de présenter les mesures existantes données par [Larsen 2001], pour décrire le dynamisme d'un problème DVRP avec et sans fenêtres de temps.

2.6.1 Dynamisme sans fenêtres de temps (dod)⁵

Nous examinons les mesures décrivant le dynamisme d'un problème DVRP sans fenêtres de temps. Dans un tel problème, trois paramètres seulement sont appropriés :

1. Le nombre de clients statiques.
2. Le nombre de clients dynamiques (appelés aussi demandes immédiates).
3. Les instants d'arrivée des clients dynamiques.

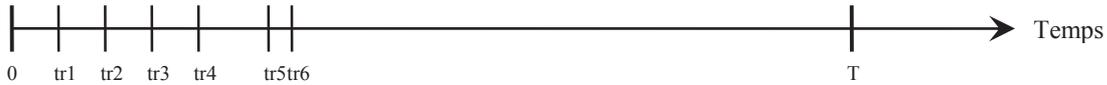
⁵Degree Of Dynamism

Dans ce cas le degré de dynamisme (*dod* : Degree Of Dynamism) est défini comme ci-dessous [Lund et al. 1996] :

$$dod = \frac{\text{le nombre de clients dynamiques}}{\text{le nombre total de clients}} \quad (2.1)$$

Cependant, cette mesure ne tient pas compte des temps d'arrivée des demandes immédiates. Ceci signifie que deux systèmes différents, un dans lequel les demandes immédiates sont reçues au début de l'horizon de planification et un autre dans lequel elles se produisent tout au long de la journée, seront perçus comme étant équivalents (voir la figure 2.5).

Scénario A



Scénario B

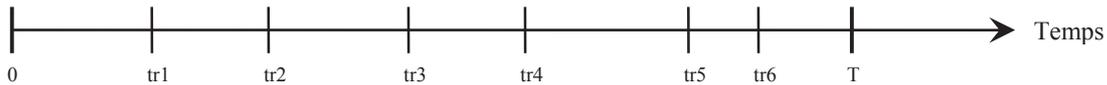


FIG. 2.5 – Deux scénarios ayant des “*dod*” équivalents

Naturellement, dans la pratique ces deux scénarios auront des comportements très différents. Cette figure (2.5) illustre deux scénarios de DVRP dans lesquels les instants de réception des demandes immédiates diffèrent considérablement. Nous remarquons que dans le scénario *A*, chacune des six demandes immédiates est reçue relativement tôt pendant l'horizon de planification. Par contre, dans le scénario *B*, les demandes sont uniformément distribuées sur tout l'horizon de planification.

Du point de vue des performances, il est clair qu'avoir un nombre élevé de demandes en file d'attente améliore la qualité de la solution produite en ce qui concerne l'objectif de minimiser la distance totale parcourue. Par conséquent, la distance parcourue prévue serait éventuellement plus courte dans le scénario *A* que dans le scénario *B*, étant donné que le planificateur connaît à l'instant tr_6 toute l'information sur les positions des clients.

2.6.2 Degré de dynamisme effectif - *edod*⁶

Maintenant on considère un scénario dans lequel l'horizon de planification commence au temps 0 et finit au temps T . Les demandes statiques sont reçues avant le commencement de l'horizon de planification. Le temps de réception de la demande numéro i est noté tr_i : $0 < tr_i \leq T$. Le nombre de demandes immédiates reçues pendant l'horizon de planification est noté n_{imm} et le nombre de demandes statiques est noté n_{adv} . Le nombre total des demandes, n_{tot} est égal à $n_{tot} = n_{adv} + n_{imm}$. Le degré de dynamisme effectif, noté *edod*, est défini par [Larsen 2001] :

$$edod = \frac{\sum_{i=1}^{n_{imm}} \left(\frac{tr_i}{T} \right)}{n_{tot}} \quad (2.2)$$

⁶Effective Degree Of Dynamism

Le degré de dynamisme effectif représente alors un pourcentage moyen du retard de chaque demande reçue par rapport au retard maximal admissible défini par T .

On peut facilement voir que : $0 \leq edod \leq 1$

Où quand $edod = 1$, le problème est complètement dynamique et lorsque $edod = 0$, le problème est purement statique.

2.6.3 Dynamisme avec fenêtres de temps - $edod-tw$ ⁷

Finalement, nous avons déjà vu que pour le problème VRPTW, chaque client doit fournir deux limites de temps :

- e_i : est la borne inférieure de la fenêtre de temps du client i .
- l_i : est la borne supérieure de la fenêtre de temps du client i .

Le degré de dynamisme effectif peut alors être étendu pour intégrer cette notion et montrer l'urgence de la demande. Pour chaque demande i , le terme $l_i - tr_i$ représente l'horizon de décision pour traiter la demande. Plus celui-ci sera petit plus la décision sera urgente à prendre. Une valeur moyenne du caractère d'urgence peut alors être définie par :

$$edod - tw = \frac{1}{n_{tot}} \sum_{i=1}^{n_{tot}} \left(\frac{T - (l_i - tr_i)}{T} \right) \quad (2.3)$$

Aussi quand $edod - tw = 1$, le problème est complètement dynamique et lorsque $edod - tw = 0$, le problème est purement statique.

2.7 Classification du DVRP

Nous allons utiliser la classification du DVRP présentée par [Larsen 2001]. Cette classification est faite en employant trois échelons de dynamisme. On distingue les problèmes faiblement, modérément, et fortement dynamiques (voir la figure 2.6).

Cette figure montre que pour chaque catégorie de problème, l'objectif principal à atteindre peut varier de la minimisation du coût de routage qui correspond à la distance totale parcourue, à la minimisation du temps de réponse.

2.7.1 Problèmes faiblement dynamiques

Les problèmes de VRP avec un degré faible de dynamisme incluent la distribution de mazout ou de gaz liquide chez les particuliers. Dans cet exemple, la plupart des clients (plus de 80 %) sont connus à l'heure de la planification des routes. D'autres exemples incluent des services d'installations chez les particuliers, tels que la télévision par câble et le téléphone. Le transport de personnes âgées ou handicapées peut être également classé comme un problème faiblement dynamique. En effet, l'objectif à atteindre est de minimiser un coût de routage tel que la distance totale parcourue par les véhicules.

L'approche traditionnelle pour résoudre de tels problèmes est basée sur l'adaptation des procédures statiques. Un problème statique d'élaboration de tournées de véhicules est résolu chaque fois qu'un évènement se produit et change l'environnement du problème.

⁷Effective Degree of Dynamism for problems with Time Windows

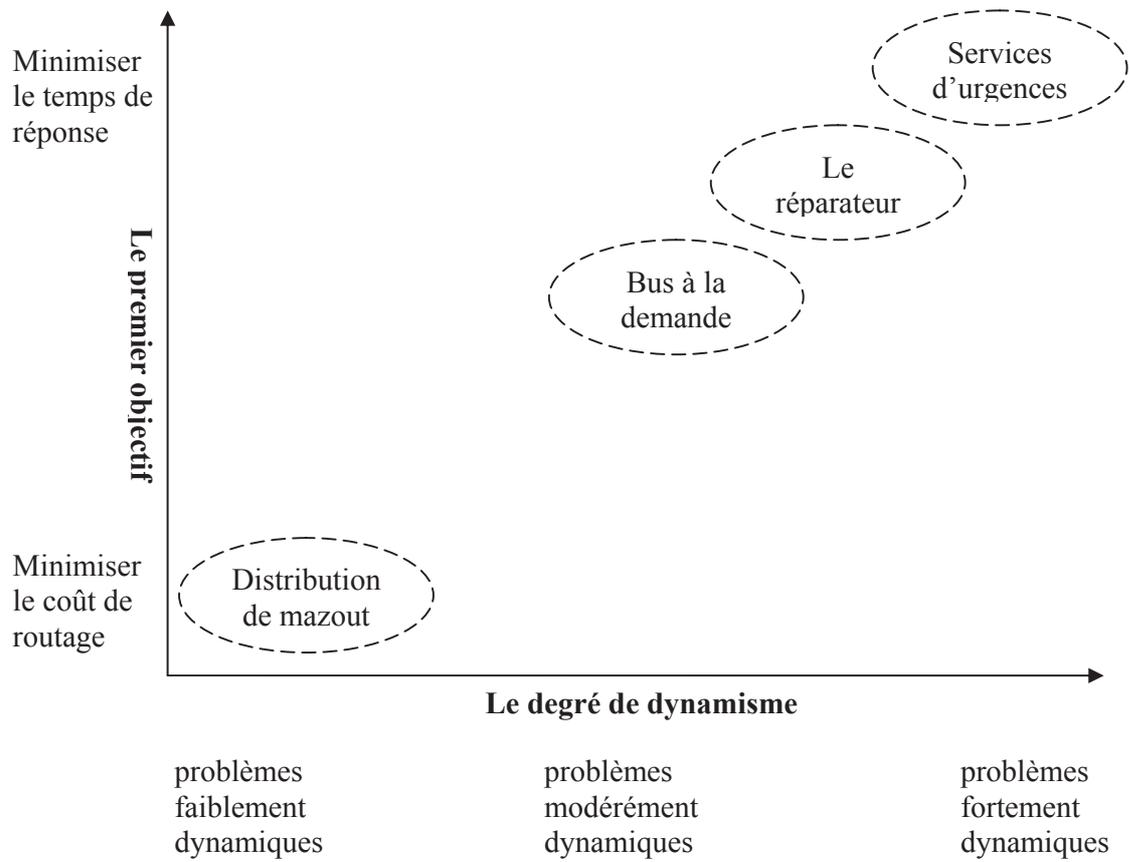


FIG. 2.6 – Classification du DVRP

2.7.2 Problèmes modérément dynamiques

En comparant ces problèmes avec ceux ayant un degré de dynamisme faible, le nombre de demandes immédiates forment une partie substantielle du nombre total de clients. Des exemples pratiques de ces problèmes incluent les services de courrier ou les services de dépannage chez les particuliers.

2.7.3 Problèmes fortement dynamiques

Les services de secours, tels que la police, les pompiers et les ambulances montrent un comportement fortement dynamique [Brotcorne et al. 2003]. Un autre exemple est celui du service des taxis dans lequel seul un nombre négligeable de clients ont commandé leur course à l'avance. En général, le but principal à satisfaire dans de tels problèmes est de minimiser le temps de réponse pour satisfaire la demande.

2.8 Le DVRP dans la littérature

Ces dernières années, nous remarquons que les problèmes de transports dynamiques sont devenus un domaine de recherche important. De nombreux travaux ont été consacrés à la résolution de ces problèmes. En général, les méthodes de résolution sont classées en deux catégories (voir la figure 2.7) :

1. Méthodes *séquentielles*,
2. Méthodes *parallèles* (recherche de solutions en parallèle).

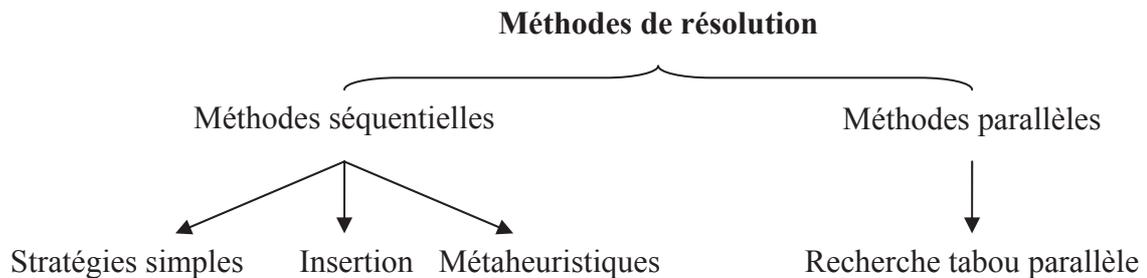


FIG. 2.7 – Les méthodes de résolution du DVRP

2.8.1 Méthodes séquentielles

Les algorithmes séquentiels utilisés comme méthodes de résolution, peuvent être divisés en trois catégories principales :

1. stratégies simples,
2. procédures d'insertion classique,
3. métaheuristiques.

2.8.1.1 Stratégies simples

Nous pouvons dire qu'une stratégie indique quelles sont les actions à prendre en fonction de l'état du système.

Dans leur rapport [Larsen 2001] et [Bianchi 2000] ont présenté plusieurs stratégies pour résoudre le problème dynamique de voyage de réparateur (DTRP). Nous allons donner une courte description de ces stratégies :

1. **FCFS** (First Come First Serve) : Les demandes sont servies dans l'ordre dans lequel elles ont été reçues par le planificateur.
2. **SQM** (Stochastic Queue Median) : C'est une modification de la stratégie FCFS où le véhicule part directement de la médiane de la région de service au prochain client. Après que le service ait été accompli, le réparateur revient si besoin est à la médiane et attend la prochaine demande.
3. **NN** (Nearest Neighbor) : Après avoir servi un client, le réparateur se dirige vers le client le plus proche non encore servi.
4. **PART** (PARTitioning policy) : La région de service est divisée en sous-régions dans lesquelles les demandes sont servies en utilisant la stratégie FCFS.
5. **TSP** (Traveling Salesman Problem strategy) :
 - (a) Au fur et à mesure que les demandes arrivent, on forme des ensembles de demandes consécutives (n demandes) que l'on dépose dans une file d'attente.
 - (b) Pour chaque ensemble, on applique alors un TSP.
6. **mod TSP** (modified TSP) :
 - (a) Nous découpons la région de service en k régions.
 - (b) Nous appliquons la stratégie TSP sur chaque région.
 - (c) Une file d'attente permet d'indiquer l'ordre dans lequel les régions vont être servies (Voir la figure 2.8).
7. **GEN** (GENeration strategy) : Cette stratégie a été appliquée au problème DTRP⁸ (Dynamic Traveling Repairman Problem) avec un seul véhicule.
 - (a) Initialement, on met le véhicule dans le point d'attente de la région de service.
 - (b) Dès qu'une demande arrive, le véhicule se déplace directement pour servir ce client (la première demande forme la première génération).
 - (c) Après avoir servi tous les clients d'une génération, on regarde si d'autres demandes ont été reçues (une nouvelle génération), et on applique un TSP sur celles-ci le cas échéant. Dans le cas contraire, on passe à l'étape 7a. (voir la figure 2.9)

Toutes ces stratégies ont été appliquées sur des petites instances de problème DTRP. Par ailleurs, ces stratégies peuvent être étudiées analytiquement. Si les demandes se produisent selon un processus de *Poisson* avec une cadence λ et que les positions des clients sont uniformément distribuées dans une région euclidienne, la stratégie SQM a montré une haute performance par

⁸Le problème dynamique de la tournée du dépanneur (TRP) est une variante du problème dynamique d'élaboration de tournées de véhicule DVRP. L'objectif est de minimiser la latence, autrement dit, la somme pondérée des temps de service des clients, celle-ci est la somme des temps d'attente des clients avant que ceux-ci ne soient atteints

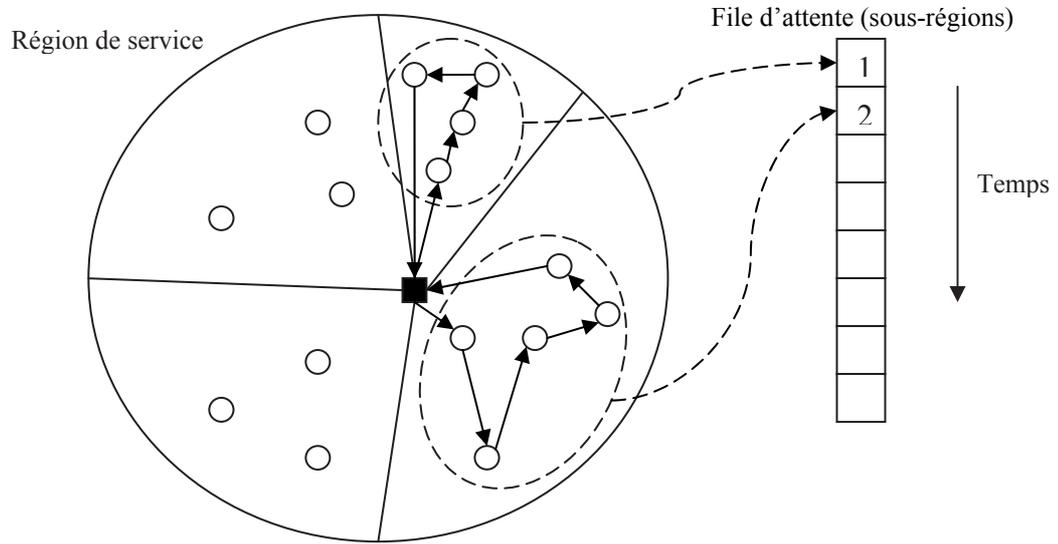


FIG. 2.8 – mod TSP

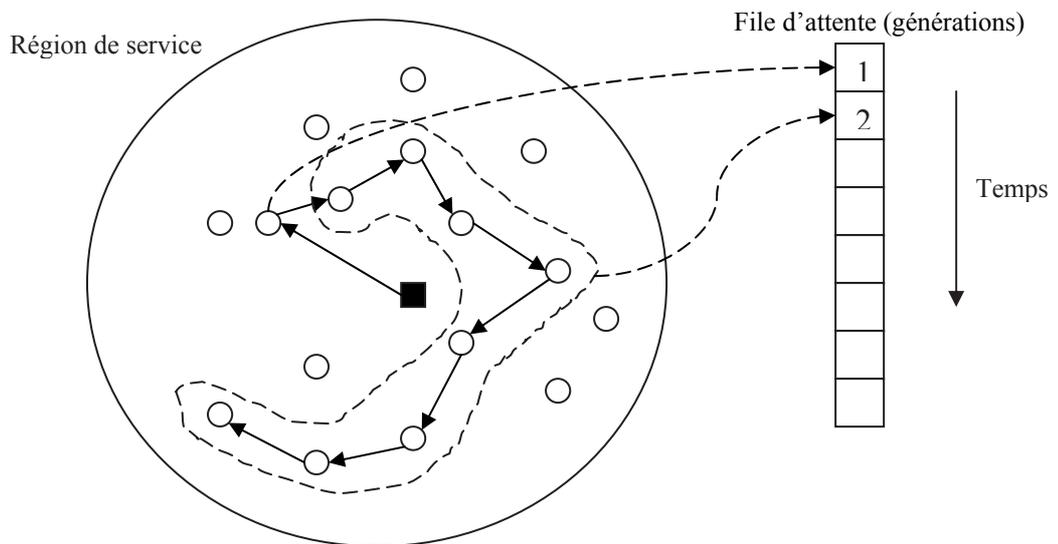


FIG. 2.9 – L'heuristique GEN

rapport aux autres stratégies. Malheureusement, celle-ci devient rapidement instable lorsque le taux d'arrivée des demandes augmente.

[Larsen et al. 2004] a proposé une heuristique dynamique très rapide pour la résolution du problème DTSPWT (Dynamic Traveling Salesman Problem with Time Windows). Il commence d'abord par la résolution du problème TSPTW pour les clients statiques (les clients connus en avance). La simulation commence à partir de l'instant t_0 , où le véhicule démarre le service en passant chez les clients selon l'ordre défini par l'étape. La journée de service est divisée en tranches de temps. Dans chaque tranche les nouveaux clients apparaissant sont collectés et ajoutés au groupe de clients restant à servir. Après chaque ajout l'heuristique de résolution du problème TSPTW est démarrée pour résoudre le problème contenant ce groupe de clients. Cette heuristique utilise l'opérateur 3 – *Opt* proposé par [Lin 1965] pour démarrer une recherche locale. La solution est initialisée avec la liste des clients à servir classés selon la borne inférieure de leur fenêtre de temps. Si le problème DTSPWT est grand, Larsen a proposé d'utiliser des opérateurs plus efficaces tel que l'opérateur de recherche locale *Or – Opt* proposé par [Or 1976].

Lorsque le véhicule finit le service chez le client courant, [Larsen et al. 2004] distingue deux situations différentes qui peuvent apparaître :

1. Soit la fenêtre de temps du prochain client à servir est déjà ouverte ou le sera avant l'arrivée du véhicule.
2. Soit le véhicule arrivera chez le client avant l'ouverture de sa fenêtre de temps, ou bien il n'existe plus de clients à servir.

Plusieurs stratégies ont été adoptées par [Larsen et al. 2004] pour répondre aux deux précédentes situations :

- **CURRENT** : Le véhicule attend à la position du dernier client servi jusqu'au moment où il peut se diriger vers le prochain client à servir, c'est-à-dire la fenêtre de temps du prochain client sera ouverte à l'arrivée du véhicule.

Larsen a proposé la notion de point d'attente par sous-région de service pour permettre au véhicule d'attendre la prochaine demande.

- **NEAREST** : Dans cette stratégie, le véhicule se dirige vers le plus proche point d'attente par rapport à sa position courante.
- **BUSIEST** : Le véhicule va au point d'attente ayant le plus grand λ_i , où λ_i est le taux d'arrivée des demandes dynamiques.
- **HI-REQ** (Highest expected number of immediate REQuests) : Le véhicule va au point d'attente qui a le plus haut score d'attraction (AS : Attractiveness Score). Le score d'attraction d'une sous région AS_j peut être interprété comme le nombre de demandes dynamiques prévues pour cette sous région pendant la tranche de temps courante.

$$AS_j = \Delta T_j \cdot \lambda_j, \quad j \in IP \text{ (Idle Point)}$$

où, ΔT_j est le temps d'inactivité du véhicule si le point d'attente de la région j a été choisi par ce véhicule, i.e. le temps séparant la fin de service du dernier client servi jusqu'au départ du véhicule du point j pour servir le prochain client.

Notons que ces deux dernières stratégies utilisent des informations historiques concernant le taux d'apparition des clients dynamiques dans les sous-régions, c'est pourquoi Larsen a renommé ce problème **ADTSPTW** (A-priori DTSPWT).

- **STATIC** : Cette stratégie est identique à la stratégie CURRENT, mais tous les clients dynamiques sont considérés comme des clients statiques. En fait, cette stratégie n'est pas une stratégie dynamique mais elle sert de comparaison référentielle pour les autres.
- **REOPT** : Celle-ci permet de servir tous les clients dès qu'ils apparaissent au cours du temps ; cela signifie que le véhicule ne se gare au point d'attente que dans le cas où il n'existe plus de client à servir. Pour les autres situations, par exemple, si le véhicule est arrivé au client avant l'ouverture de sa fenêtre de temps, cette situation est permise mais avec une pénalité ; dans ce cas la fenêtre de temps est dite flexible.

[Yang et al. 2004] a proposé plusieurs stratégies pour la résolution d'une variante spéciale de VRPTW dynamique, où les véhicules sont des camions. Chaque camion ne sert qu'un client à la fois, il doit transporter directement les produits depuis un lieu de ramassage vers un lieu de livraison. En plus, le planificateur a le droit d'accepter ou de refuser la demande. Nous allons donner une explication de chaque stratégie utilisée :

- **BENCH** : Dans cette stratégie, le planificateur calcule le coût marginal pour ajouter le nouveau client à la fin de la tournée de chaque camion, puis insère ce client dans la tournée de coût marginal minimum.
- **NS** : Cette stratégie est la même que la précédente, mais le planificateur vérifie toutes les positions d'insertion possibles à l'intérieur d'une tournée.
- **SE** : Les deux stratégies précédentes conservent toujours l'ordre relatif des demandes déjà placées. Dans celle-ci, le planificateur vérifie l'insertion de la nouvelle demande dans chaque position de la tournée avec la possibilité de changer l'ordre des anciennes demandes. Il calcule ainsi toutes les possibilités d'ordonnancement à l'intérieur d'une tournée.

Les trois stratégies prennent la décision d'accepter ou de refuser la nouvelle demande. Si la demande est acceptée, elle est insérée directement dans la position choisie par la stratégie. S'il n'est pas possible de la servir, elle sera refusée au prix d'un certain coût.

- **MYOPT** : Elle prend en compte l'ensemble des demandes et démarre une heuristique pour la résolution du problème en le considérant comme un problème statique.
- **OPTUN** : Il s'agit de la même stratégie que la précédente excepté que le planificateur prend en compte les demandes qui peuvent apparaître dans le futur. Pour cela une loi de probabilité sur les futures positions des demandes dynamiques est utilisée.

2.8.1.2 La méthode d'insertion classique

Au lieu d'utiliser des stratégies simples qui sont efficaces sous des hypothèses spécifiques (comme pour le problème du réparateur DTRP), une heuristique plus performante, ayant une bonne applicabilité dans plusieurs domaines de fonctionnement, peut être utilisée. Cette méthode est largement utilisée pour la résolution de problème VRP dynamique. De telles méthodes re-optimisent simplement les tournées de véhicules quand les nouvelles données deviennent disponibles. Elles cherchent à insérer le nouveau client dans la meilleure position des tournées courantes. Cette méthode est décrite dans [Roy et al. 1984]. Une telle heuristique est simple, elle peut être très rapide et donc être utilisable pour des problèmes de nature dynamique (voir la figure 2.10). Dans cette figure, le client (I) a demandé un service. Nous devons rechercher la meilleure position d'insertion possible dans la matrice des tournées. Cette matrice contient tous les clients à servir, et ne contient pas les clients déjà servis. Pour chaque tournée (par exemple $v1$), nous vérifions toutes les positions d'insertion possibles pour ce client (dans notre exemple,

deux positions). Puis nous pouvons simplement décider dans quelle position le client doit être inséré.

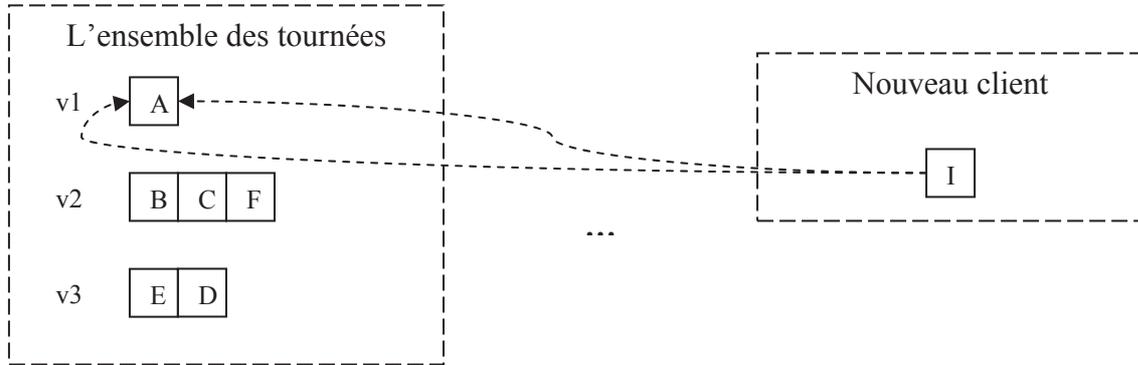


FIG. 2.10 – Heuristique d'Insertion

[Larsen 2001] a utilisé cette méthode pour résoudre le problème VRPTW dynamique. Il a réalisé un simulateur de ce problème (voir la figure 2.11), celui-ci se compose essentiellement de deux modules de base :

1. Un module de contrôle.
2. Un module de routage.

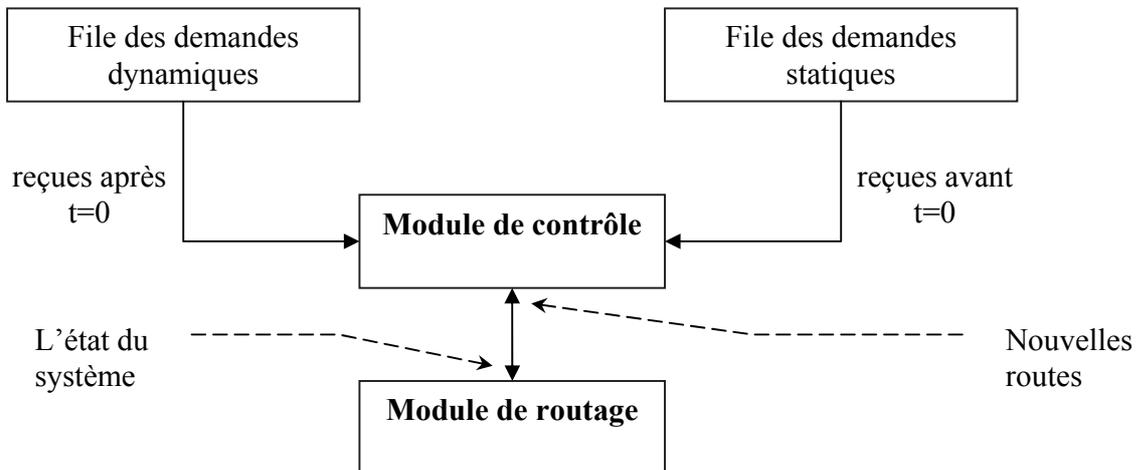


FIG. 2.11 – Simulation du système dynamique

Le module de contrôle gère les véhicules; il associe à chaque véhicule les clients à servir selon la solution fournie par le module de routage. Ce module traite les événements qui peuvent intervenir au cours du temps et change alors l'état du système. Les événements peuvent être :

1. L'arrivée d'une nouvelle demande.
2. La fin de service d'un client

Toutes les cinq minutes, ce module vérifie l'état du système; si cet état a changé, il envoie le nouvel état au module de routage. Celui-ci à son tour prend la décision de réarranger les tournées

existantes selon le nouvel état. Par exemple, il démarre la procédure d'insertion pour répondre à l'évènement d'une nouvelle demande, ou bien il associe au véhicule libre le prochain client à servir si l'évènement est une fin de service.

Une autre étude qui utilise cette méthode d'insertion concerne [Kilby et al. 1998a]. Dans ces travaux, le temps de simulation (le temps de la journée de service) est divisé en tranches (50 au total). Durant chaque tranche, le planificateur collecte les nouvelles demandes et les traite dans la prochaine tranche de temps. Autrement dit, le planificateur utilise la méthode d'insertion pour trouver les meilleures positions pour les nouvelles demandes reçues au cours de la tranche précédente. Dans le même temps, ce planificateur essaie d'améliorer la solution obtenue en appliquant un opérateur de recherche locale 2 – *Opt* proposé par [Lin 1965].

Récemment, [Mitrovic-Minic et al. 2004a] a introduit le concept de double horizon (court terme et moyen terme) pour la résolution du problème dynamique de ramassage et de livraison de colis avec fenêtres de temps DPDP. Lorsque une nouvelle demande arrive, un gros effort est fait pour optimiser la partie à court terme (short-term) de la solution courante qui correspond en général à la période de temps utilisée pour le ramassage d'un colis. Moins d'effort est nécessaire pour la partie éloignée de la solution correspondant en général à la période de temps contenant la livraison du colis. L'idée sous-jacente de cette approche est que les parties éloignées des tournées sont susceptibles d'être soumises à plusieurs changements dans le temps. Il n'est donc pas nécessaire que ces parties soient optimisées trop finement. Après que la solution soit construite, [Mitrovic-Minic et al. 2004b] a utilisé quatre stratégies d'attente pour servir le prochain client :

1. **DF** (Drive-First) : La stratégie d'attente (DF) exige qu'après la fin de service d'un client, le véhicule parte directement vers le prochain client. Cela peut obliger le véhicule à attendre une période de temps avant le commencement du service, si le temps d'arrivée de celui-ci est inférieur à la borne inférieure de la fenêtre de temps du client (e_i).
2. **WF** (Wait-First) : Dans celle-ci, on exige que le véhicule attende dans sa position courante et parte le plus tard possible chez le prochain client pour arriver juste à l'heure d'ouverture de la fenêtre. Pendant ce temps d'attente l'arrivée de nouvelles demandes peut modifier le choix du prochain client.

Pour expliquer les deux stratégies suivantes, il faut découper la tournée d'un véhicule en zones (partitions), chaque partition contenant un ensemble de positions consécutives regroupées.

3. **DW** (Dynamic Waiting) : La stratégie d'attente dynamique est une combinaison de DF et de WF. Elle construit une tournée de la façon suivante :
 - (a) Le véhicule utilise la stratégie DF dans chaque zone de service.
 - (b) Quand le véhicule a fini de servir la zone courante, le véhicule emploie la stratégie WF pour se déplacer vers une autre zone à servir.
4. **ADW** (Advanced Dynamic Waiting) : L'idée sous-jacente de cette stratégie est de répartir le temps d'attente total sur l'ensemble des clients d'une tournée. Il se fonde également sur la stratégie précédente DW. Cependant, le temps d'attente de la dernière position visitée d'une zone de service est déterminé différemment. Il peut être proportionnel au rapport de la durée d'attente totale de la zone considérée sur la somme des durées d'attente totale de l'ensemble de la tournée.

2.8.1.3 Les métaheuristiques

Peu de travaux existant dans la littérature utilisent les métaheuristiques comme méthodes pour la résolution des problèmes de transports dynamiques. Récemment, sont apparus quelques articles sur l'utilisation des métaheuristiques pour la résolution du problème VRP dynamique. Parmi eux, nous avons les travaux de [Montamenni et al. 2002], qui implémente une métaheuristique basée sur les colonies de fourmis pour résoudre le DVRP sans fenêtres de temps, comme suit :

1. La journée de service est divisée en tranches de temps. Pour chacune d'elles, un problème VRP statique, prenant en compte toutes les demandes déjà reçues, est créé puis résolu.
2. Les nouvelles demandes reçues pendant la tranche de temps courante sont traitées dans la tranche suivante. Dans une tranche de temps, les demandes reçues précédemment et les anciennes demandes non encore servies sont regroupées, puis on redémarre de nouveau la résolution d'un VRP statique.

Par ailleurs, [Mitrovic-Minic 2001] a utilisé la recherche tabou couplée avec la méthode d'insertion pour résoudre le problème dynamique de ramassage et livraison avec fenêtres de temps (DPDPTW). Ces travaux ont utilisé la notion de découpage de l'horizon en deux parties, le but étant d'optimiser plus finement la partie à court terme qui est la partie ramassage de colis, que la partie à long terme qui est également la partie livraison de colis.

Dans cette thèse nous utiliserons une métaheuristique basée sur les algorithmes génétiques pour résoudre le problème VRPTW dynamique [Housroum et al. 2003] et [Housroum et al. 2004]. En bref, notre méthode consiste à résoudre ce problème directement "en ligne". Autrement dit, nous traitons directement l'évènement dès qu'il se présente. Nous détaillerons cette méthode dans le chapitre suivant de cette thèse.

2.8.2 Méthodes parallèles

L'utilisation de métaheuristiques parallèles pour résoudre un problème en temps réel est possible voire nécessaire afin de réduire les temps de calcul. Le parallélisme peut être mis en œuvre de manière différente selon l'architecture du matériel utilisé pour le résoudre.

2.8.2.1 La recherche tabou parallèle

Cette méthode décrite dans [Gendreau et al. 1998] et [Gendreau et al. 1999a], permet de résoudre les deux problèmes dynamiques (DPDPTW) et (DVRPTW) respectivement. Dans les deux articles, l'algorithme d'optimisation principal que les auteurs veulent exploiter est la *recherche tabou adaptative parallèle*. Voici comment l'algorithme (2.1) interagit avec l'environnement dynamique.

En fait, cet algorithme conçu pour une utilisation en parallèle, a été testé sur une machine multi-processeurs où chaque processeur lance un algorithme de recherche tabou.

Par ailleurs, les auteurs ont comparé cette méthode avec des méthodes conçues spécialement pour résoudre ces problèmes dynamiques. Ces méthodes sont :

1. *La méthode d'insertion classique* : cette méthode a déjà été mentionnée ci-dessus.
2. *L'insertion avec recherche locale* (insertion+) : cette méthode est identique à la précédente, mais en plus elle applique une procédure d'amélioration de la solution courante après chaque insertion d'un nouveau client.

Algorithme 2.1 Pseudo-Code de l'algorithme de la recherche tabou parallèle

Tant que "aucun évènement"

Améliorer la solution courante en utilisant la recherche tabou ;

Si un évènement se produit :

Arrêter tous les processus de recherche (l'heuristique tabou) et ajouter les meilleures solutions à la mémoire adaptative qui représente un ensemble de bonnes solutions.

Si l'évènement est l'arrivée d'une nouvelle demande :

Mettre à jour la mémoire adaptative par l'insertion de la nouvelle demande dans chaque solution ;

Si l'insertion de la nouvelle demande est impossible :

Rejeter la demande ;

FinSi

Sinon (l'évènement est une fin de service d'un client)

Identifier la prochaine destination du conducteur, en utilisant la meilleure solution stockée dans la mémoire adaptative ;

Mettre à jour les autres solutions en conséquence ;

Relancer les processus de recherche de tabou avec de nouvelles solutions obtenues à partir de la mémoire adaptative.

FinSi.

FinSi.

FinTantQue.

3. *La construction* : cette méthode reconstruit la solution, en utilisant la méthode d'insertion, après chaque évènement correspondant à une nouvelle demande.
4. *La construction avec recherche locale* (construction+) : cette méthode est identique à la précédente ; elle applique aussi une procédure d'amélioration locale après chaque étape de construction.
5. *Descente adaptative* : Celle-ci est une version modifiée de la méthode tabou utilisée. Dans ce cas-ci, les processus tabous sont arrêtés dès qu'un optimum local est trouvé.

Les résultats ont montré que la méthode parallèle avec la recherche tabou est bien meilleure que les autres.

Enfin, nous trouvons aussi dans la littérature des approches [Ichoua et al. 2003] et [Park et al. 1997] qui traitent des problèmes VRPs dynamiques mais la dynamique ici considérée est la variation du temps de trajet entre deux clients par rapport à la période de la journée. D'autres approches considèrent que la vitesse des véhicules peut varier selon la région visitée (par exemple : ville, village, etc.) ou le type de liaison utilisée. Pour plus d'information, nous renvoyons aux travaux de [Park 2000] et [Schneider 2002].

Dans la table (2.1) nous présentons pour chaque problème dynamique : le nom du problème, les informations qui concernent la partie statique du problème, les informations concernant la partie dynamique, la méthode utilisée pour la résolution et finalement la référence bibliographique.

Dans cette thèse nous avons d'abord étudié le problème DVRPTW, puis nous avons étendu notre étude à d'autres problèmes dynamiques comme le problème DVRP, DTRP, DTSPTW et DPDPTW.

2.9 Conclusion

Nous avons vu dans ce chapitre que le problème du VRP dynamique est devenu un domaine de recherche important avec de nombreuses applications dans la vie pratique. Par ailleurs, ces travaux ont utilisé plusieurs méthodes de résolution. Parmi ces méthodes, nous avons les stratégies, la méthode d'insertion, ainsi que la recherche tabou, etc. Mais malgré sa nature évolutive, nous n'avons pas trouvé dans la littérature, une utilisation des algorithmes génétiques pour la résolution de ce type de problèmes.

Dans le chapitre suivant, nous présentons notre travail de thèse sur l'utilisation des algorithmes génétiques pour la résolution du problème DVRPTW (Dynamic Vehicle Routing Problem with Time Windows).

| Problème | Info. A-priori | Info. Dynamiques | Méthode | Référence |
|-----------|--------------------------------------------------|---------------------------------------------|------------------------------------|-------------------------------|
| DSVRP | m véhicules avec capacités | Quantités des demandes | Markov | [Laporte et al. 1989] |
| DTRP | 1 véhicule sans capacité | nouveaux clients | stratégies (SQM) | [Bertsimas et al. 1991] |
| DRIVE | - m véhicules hétérogènes, - n dépôts. | nouveaux clients | Branch & Price | [Savlesberg et al. 1991] |
| DTRP | m véhicules avec et sans capacités | nouveaux clients | stratégies | [Bertsimas et al. 1993b] |
| DTRP | m véhicules avec et sans capacités | nouveaux clients | stratégies | [Bertsimas et al. 1993a] |
| VSPTS | - m véhicules avec capacités, - n clients | le temps de trajet | Heuristiques | [Park et al. 1997] |
| Ambulance | m véhicules | nouveaux clients | Recherche Tabou | [Gendreau et al. 1997] |
| DVRP | - m véhicules avec capacités, - % de clients. | nouveaux clients | Insertion | [Kilby et al. 1998a] |
| DPDPTW | - m véhicules sans capacité, - % de clients. | nouveaux clients | Recherche Tabou Parallèle | [Gendreau et al. 1998] |
| DVRPTW | - m véhicules sans capacité, - % de clients. | nouveaux clients | Recherche Tabou Parallèle | [Gendreau et al. 1999a] |
| DTSP | 1 véhicule sans capacité | Supprimer et insérer de nouveaux clients | ACO | [Guntsch et al. 2000] |
| Ambulance | m véhicules | nouveaux clients | Recherche Tabou Parallèle | [Gendreau et al. 2001] |
| DVRPTW | - m véhicules avec capacités, - % de clients. | nouveaux clients | Insertion | [Larsen 2001] |
| ADTSPTW | - 1 véhicule sans capacité, - % de clients. | nouveaux clients | Stratégies | [Larsen et al. 2004] |
| DVRP | - m véhicules sans capacité, - % de clients. | nouveaux clients | Colonie de fourmis | [Montamenni et al. 2003] |
| DPDPTW | m véhicules avec capacités. | nouveaux clients | - Insertion, - Recherche Tabou. | [Mitrovic-Mimic et al. 2004b] |
| TDVRP | - m véhicules avec capacités, - n clients | le temps de trajet | Recherche Tabou Parallèle | [Ichoua et al. 2003] |

TAB. 2.1: Le problème DVRP et ses extensions dans la littérature

Chapitre 3

Un algorithme génétique “en ligne” comme méthode de résolution du DVRPTW

Sommaire

| | | |
|------------|----------------------------------------|-----------|
| 3.1 | Introduction | 55 |
| 3.2 | Motivation | 56 |
| 3.3 | Algorithme Génétique | 56 |
| 3.3.1 | Cycle de vie | 57 |
| 3.3.2 | Le codage de la solution | 59 |
| 3.3.3 | Fonction d'évaluation | 60 |
| 3.3.4 | Population initiale | 65 |
| 3.3.5 | Sélection | 65 |
| 3.3.6 | Reproduction | 65 |
| 3.4 | Environnement de Simulation | 72 |
| 3.4.1 | Le composant Algorithme Génétique (AG) | 72 |
| 3.4.2 | Le composant Gestion d'Evénements (GE) | 72 |
| 3.4.3 | Le fonctionnement du simulateur | 73 |
| 3.5 | Conclusion | 78 |

3.1 Introduction

Dans cette thèse, nous traitons le problème d'élaboration de tournées de véhicules dynamique avec fenêtres de temps (DVRPTW : Dynamic Vehicle Routing Problem with Time Windows) qui a été formalisé dans les chapitres précédents. On peut le résumer simplement [Gendreau et al. 1999a] par :

1. C'est un problème de type VRPTW.
2. Comportant un nombre limité de véhicules disponibles de capacité identique.
3. Deux évènements dynamiques sont à traiter :
 - (a) L'arrivée d'un nouveau client.
 - (b) La fin de service d'un client.

Dans ce chapitre, nous présentons l’implémentation de l’approche (algorithme génétique) que nous avons choisie pour résoudre le problème **DVRPTW**. Nous indiquons les adaptations que nous avons faites sur l’algorithme de base pour qu’il puisse traiter ce type de problème.

3.2 Motivation

Nous voulons profiter du caractère évolutif des algorithmes génétiques pour montrer leur bonne adaptation au traitement d’un problème d’optimisation difficile dans un contexte dynamique. A chaque fois qu’un changement se produit sur les hypothèses initiales d’un problème d’optimisation dynamique, la solution optimale du problème initial peut se trouver ou non affectée par cette perturbation. Si c’est le cas, une adaptation de l’ancienne solution est donc nécessaire.

L’approche classique utilisée pour traiter ce problème dynamique est de considérer que chaque changement engendre un nouveau problème d’optimisation qui doit être de nouveau résolu à partir de zéro. Cependant bien que cette approche soit simple, elle est souvent inapplicable dans un contexte à contraintes de temps serrées. En effet la résolution d’un problème de type NP-difficile prend généralement trop de temps pour trouver des solutions de bonne qualité. Bien souvent, la solution du nouveau problème ne diffèrera pas trop de la solution du problème précédent lorsque l’environnement est faiblement perturbé, ce qui est en général le cas dans notre application. Pourquoi dès lors ne pas réutiliser les anciennes solutions pour fournir de nouvelles solutions qui tiennent compte de la perturbation ? C’est cette possibilité que nous voulons exploiter dans le cadre de cette thèse.

L’idée serait d’avoir un algorithme d’optimisation capable d’adapter la ou les solutions en environnement dynamique, en réutilisant l’information obtenue dans le passé. Puisque les algorithmes évolutionnistes ont en commun avec le processus d’évolution naturelle leur caractère adaptatif, ils semblent être des candidats appropriés [Branke 2001] pour les environnements dynamiques.

La plus ancienne application connue des **algorithmes évolutionnistes (AE)** aux environnements dynamiques remonte à 1966 [Fogel et al. 1966]. Cependant, ce n’est qu’au milieu des années 80 que les chercheurs ont réellement commencé à les étudier et que le nombre de publications a augmenté. Au cours de ces dernières années, un certain nombre d’auteurs ont abordé ce problème de différentes manières.

3.3 Algorithme Génétique

C’est Holland qui a formulé le premier, le principe des Algorithmes Génétiques en 1975 [Holland 1975]. [Goldberg 1989] apporta également beaucoup à la création de ces algorithmes. Les algorithmes génétiques font partie des algorithmes dits “*évolutionnistes*”. Ces algorithmes sont classés dans les méthodes d’optimisation stochastiques et peuvent opérer dans des espaces de recherche extrêmement vastes. Le terme évolutionniste fait référence aux techniques basées sur les mécanismes de l’évolution naturelle. La figure (3.1) présente une hiérarchie des méthodes de résolution métaheuristique d’un problème. Cette figure permet de placer les algorithmes génétiques par rapport aux autres méthodes.

Les algorithmes génétiques ont été imaginés pour résoudre des problèmes complexes, où il n’existe pas de méthode simple et logique pour trouver la solution. Le qualificatif “*génétique*”

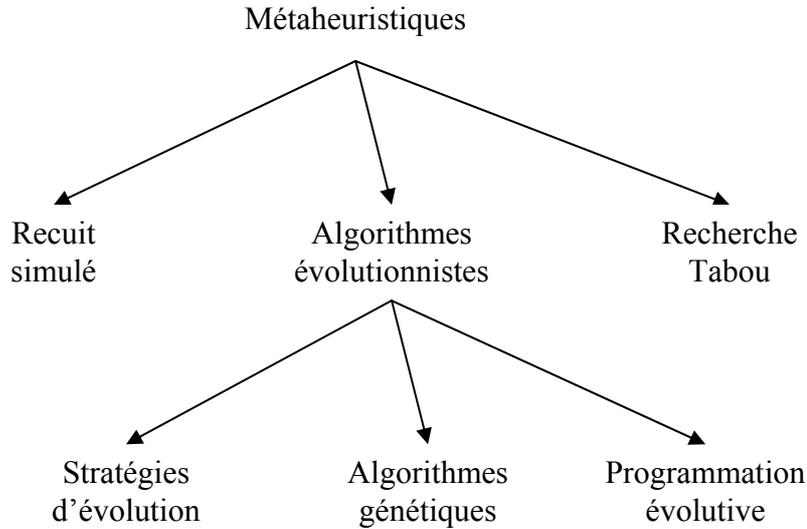


FIG. 3.1 – Méthodes de résolution

vient du fait que ce genre d'algorithme s'inspire de la théorie de l'évolution des espèces de **Charles Darwin**. Dans un premier temps, l'algorithme génétique crée au hasard des solutions possibles. Ensuite, il les fait "*évoluer*" jusqu'à obtenir des résultats les mieux adaptés au problème. Les solutions candidates possèdent chacune un "*patrimoine génétique*" représenté sous la forme d'une chaîne de "*gènes*" appelée "*chromosome*". Faisant office de "**sélection artificielle**", l'algorithme génétique favorise la "*survie*" des solutions les plus correctes c'est à dire les meilleures. Celles-ci peuvent alors se reproduire entre elles et transmettre leur "*patrimoine génétique*" à leurs descendants qui constituera la prochaine génération. Pour favoriser la diversité au moment de la reproduction, l'algorithme crée des mutations de façon aléatoire, ce qui provoque l'apparition de "*mutants*" issus des meilleurs parents. Ainsi les générations successives deviennent de mieux en mieux adaptées à la résolution du problème (voir la figure 3.2).

Un algorithme génétique est capable de fournir de bonnes solutions aux problèmes pour lesquels on ne connaît pas de méthode de résolution ou dont la solution exacte est trop complexe pour être calculée en un temps raisonnable. On utilise ce genre d'outil pour planifier des tournées de véhicules (**VRP**), constituer des équipes de travail, implanter de manière optimale des points de vente dans une région, gérer des portefeuilles financiers et même pour trouver la meilleure forme à donner à la turbine d'un réacteur nucléaire, etc.

3.3.1 Cycle de vie

Le cycle de vie de la figure (3.2) décrit les différents éléments et mécanismes qui participent à l'évolution de la population. Dans un premier temps, les individus de la population sont initialisés, puis une partie d'entre eux est sélectionnée pour être recombinaisonnée (croisement). Les nouveaux individus ainsi créés remplaceront une partie de la population initiale après mutation de certains d'entre eux. L'ensemble des opérateurs est décrit ci-après.

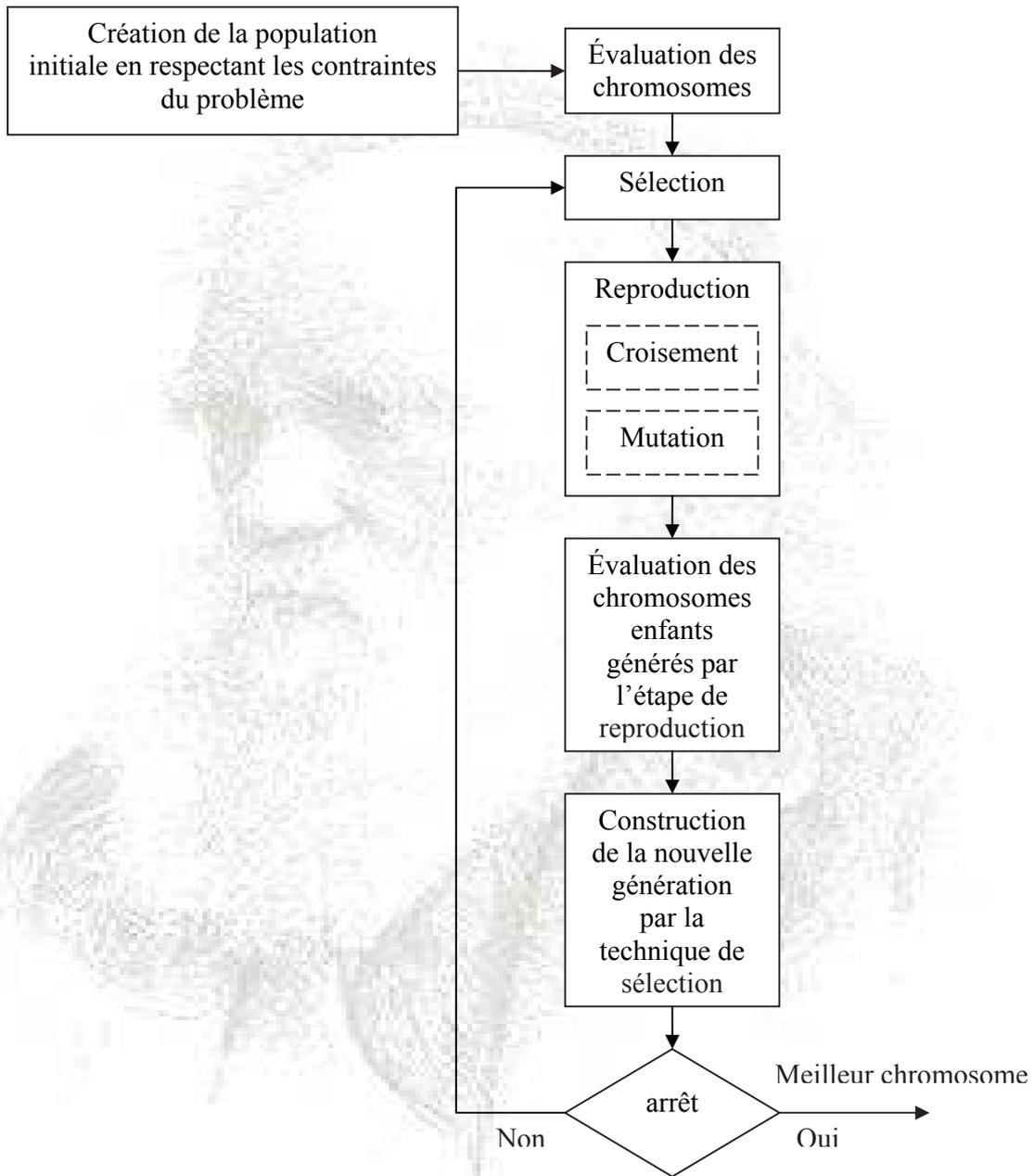


FIG. 3.2 – L'algorithme génétique

3.3.2 Le codage de la solution

Chaque individu de la population représente une solution au problème à optimiser. Un individu est représenté par un chromosome. Ce chromosome est constitué de gènes qui peuvent prendre des valeurs appelées “allèles”. La position du gène dans le chromosome se nomme “*locus*”. Il existe plusieurs manières de coder ces individus. Chacune de ces manières dépend du problème à traiter. Le contenu du gène peut être soit des nombres binaires, soit des nombres entiers ou réels, soit des caractères.

A chaque individu est associé une évaluation (appelée aussi *fitness*) qui mesure la qualité de la solution. L'évaluation représente la performance de l'individu vis-à-vis du problème. Le “*génotype*” représente l'ensemble des valeurs prises par les gènes du chromosome alors que le “*phénotype*” représente la solution réelle qui en découle. De la pertinence du codage va dépendre le choix des opérateurs de reproduction et l'efficacité globale de l'algorithme.

Il existe deux cas de figures pour coder un problème. Le premier utilise un codage “**direct**” dans lequel toute l'information de la solution est présente dans le chromosome. Le deuxième reporte la difficulté de représentation d'une solution du problème à l'extérieur du codage, en utilisant un codage “**indirect**”.

1. **Direct** : Dans un codage direct, le gène doit représenter une solution complète. Ce dernier doit contenir toutes les informations utiles à la solution [Tavares et al. 2003]. La représentation chromosomique du problème VRPTW qui utilise ce type de codage peut être tout simplement un ensemble de vecteurs de taille différente. Chaque vecteur de cet ensemble représente une tournée d'un véhicule et contient une suite des clients à visiter par ce véhicule. Ces vecteurs ont des tailles variées selon le nombre de clients à visiter pour chaque tournée (voir la figure 3.3).
2. **Indirect** : Dans ce type de codage, le génotype ne représente pas directement la solution. Il est nécessaire de passer par une étape complémentaire pour obtenir la solution [Sushil et al. 1999], [Prins 2004]. En utilisant ce codage pour représenter la solution du problème VRPTW, le chromosome vecteur va contenir une suite ordonnée de clients à servir. Dans cette représentation, nous ne pouvons pas distinguer les tournées de véhicules avant que le chromosome ne passe par une étape de décodage qui va construire les tournées en appliquant par exemple l'heuristique **PFIH** (Push Forward Insertion Heuristic) [Zhu 2000] (voir la figure 3.4).

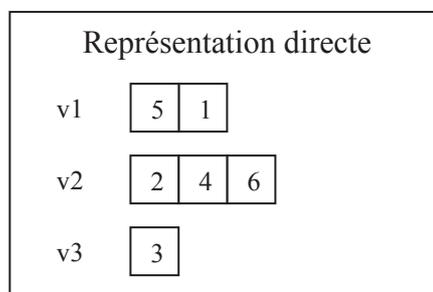


FIG. 3.3 – Représentation directe

Chaque type de codage a ses avantages et inconvénients :

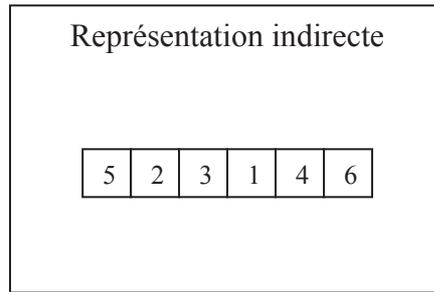


FIG. 3.4 – Représentation indirecte

- Dans le codage direct on n’a pas besoin d’une étape complémentaire pour construire les tournées. Cela signifie qu’avec ce codage nous gagnerons du temps de calcul.
- Il est plus simple d’appliquer les opérateurs génétiques sur un chromosome représenté par un codage indirect. Certaines contraintes du problème peuvent en effet être reportées dans la phase de décodage.

Dans notre problème d’élaboration dynamique de tournées de véhicules avec fenêtres de temps, nous avons préféré utiliser le codage indirect des solutions. D’une part ce codage est bien adapté pour traiter les événements dynamiques et d’autre part nous avons utilisé une heuristique rapide pour réaliser le décodage.

Ce codage indirect repose sur l’utilisation couplée de deux structures de données : d’une part un vecteur “chromosome” qui est dédié à la planification prévisionnelle ou future du routage et d’autre part “un ensemble de vecteurs” permettant de mémoriser la partie réalisée des tournées. Plus précisément, à chaque instant t de notre cadre dynamique, chaque chromosome va contenir dans un vecteur tous les clients restants à servir, c’est-à-dire tous ceux qui n’ont pas encore été servis, par la flotte de véhicules de la journée en cours (voir figure 3.5). De plus, les clients servis ou sur le point de l’être par les véhicules disponibles seront mémorisés à part dans une seconde structure de type matriciel (voir la figure 3.6). Chaque vecteur de cet ensemble représente les clients rangés dans l’ordre où ils ont été servis par un véhicule donné (i.e. tournée).

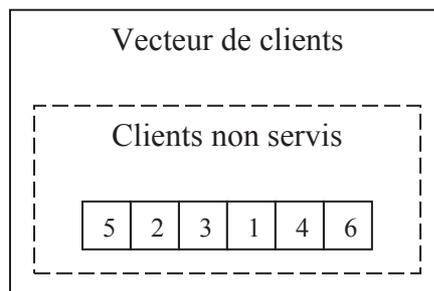


FIG. 3.5 – Le chromosome représentant les clients qui restent à servir à l’instant t

3.3.3 Fonction d’évaluation

La fonction d’évaluation permet de quantifier l’aptitude d’un chromosome à survivre. Cette quantité est appelée aussi *Fitness*. Cette fonction dépend directement du type de problème

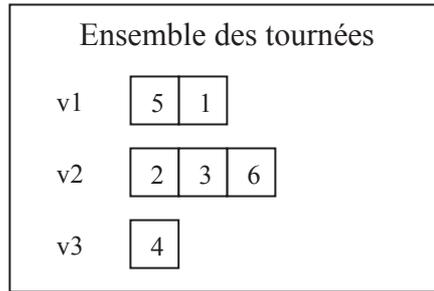


FIG. 3.6 – L'ensemble des tournées représentant les clients desservis à l'instant t

à résoudre. Dans le cas d'un problème d'optimisation, celle-ci est fortement liée à la fonction objectif que l'on cherche à minimiser ou à maximiser. Celle-ci peut également inclure un ou plusieurs critères que l'on cherche à optimiser simultanément.

Notre problème est de type **multi-objectifs**, puisque nous cherchons par exemple à minimiser la distance totale parcourue par les véhicules, à minimiser le retard total des clients, à minimiser le retard total des véhicules pour arriver au dépôt, etc. Les approches utilisées pour la résolution de problèmes multi-objectifs (*PMO*) peuvent être classées en trois catégories [Talbi 2001] :

1. **Approches basées sur la transformation du problème en un problème uni-objectif** : Parmi ces méthodes, nous avons les *méthodes d'agrégation* [Ishibuchi et al. 1998], *ϵ - contrainte* [Hertz et al. 1994] et les *méthodes de programmation par but* [Sandgren 1994].
2. **Approches non-Pareto** : Ces approches ne transforment pas le PMO en un problème uni-objectif. Elles utilisent des opérateurs de recherche qui traitent séparément les différents objectifs. Il existe plusieurs approches non-Pareto qui sont : *la sélection parallèle dans les algorithmes évolutionnistes* [Schaffer 1985], et *la sélection lexicographique* [Coello 1998].
3. **Approches Pareto** : Ces approches utilisent directement *la notion de dominance* dans la sélection des solutions générées. Contrairement aux autres approches déjà vues qui traitent séparément les différents objectifs, celles-ci doivent tenir compte de deux contraintes [Talbi 2001] :
 - (a) Converger vers la frontière Pareto optimale.
 - (b) Trouver des solutions diversifiées dans cette frontière Pareto.

Dans nos travaux, nous nous intéressons à la première catégorie. Dans cette catégorie, nous avons plusieurs méthodes qui visent à transformer le problème en un problème uni-objectif. Parmi ces méthodes nous avons la méthode d'agrégation qui consiste à affecter des *poids* (représentant des préférences) aux différents objectifs du problème. Ces poids sont déterminés en général par un expert du domaine.

$$F = w_1.D + w_2.T + w_3.R_1 + w_4.R_2 + w_5.O \quad (3.1)$$

Dans notre problème, nous avons pris la fonction objectif du problème VRPTW (l'équation 3.1) donnée par [Sushil et al. 1999] ; cette fonction est la somme pondérée de tous les objectifs de ce problème :

Où :

1. D : est la distance totale parcourue par les véhicules.
2. T : est le temps total de trajet de l’ensemble des véhicules.
3. R_1 : est le retard total chez les clients.
4. R_2 : est le retard total des véhicules au niveau du dépôt.
5. O : est la surcharge totale de capacité des véhicules.
6. w_i : sont des poids (préférences) associés à chaque critère.

Cette fonction a été réduite par [Gendreau et al. 1999a] de la manière suivante pour la résolution du problème dynamique (DVRPTW) :

$$F = w_1.D + w_2.R_1 \quad (3.2)$$

Pour pouvoir comparer notre approche avec celle de Gendreau (voir chapitre 5), nous avons donc utilisé cette fonction dans notre approche de résolution du DVRPTW. Nous constatons que la contrainte de capacité n’est pas prise en compte par Gendreau.

Pour évaluer la qualité d’une solution, c’est-à-dire calculer le fitness d’un chromosome, il faut au préalable décoder le chromosome pour obtenir la solution correspondante.

3.3.3.1 La phase de décodage d’un chromosome

Le décodage d’un chromosome pour obtenir la solution correspondante, utilise une heuristique d’insertion simple. Cette heuristique est décrite ci-dessous (algorithme 3.1) :

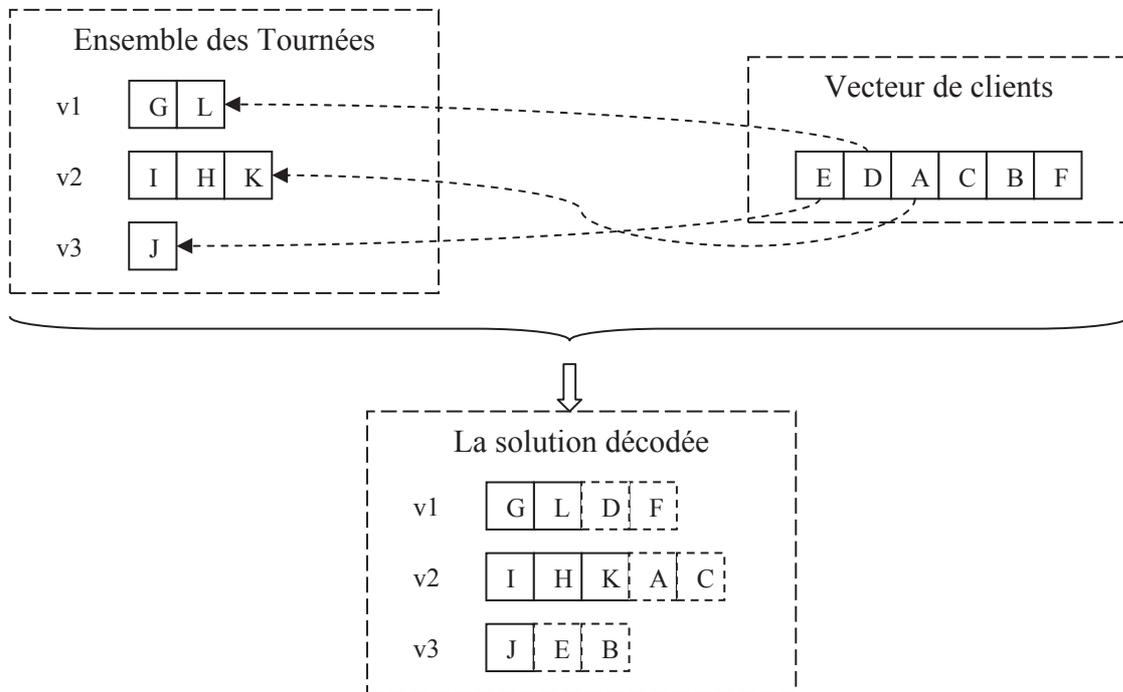


FIG. 3.7 – Le décodage d’un chromosome

L’insertion d’un client se fait toujours en fin d’une tournée existante (voir la figure 3.7). Le chromosome vecteur de clients représente en fait, une liste ordonnée de préférence décroissante

Algorithme 3.1 Algorithme d'insertion (décodage d'un chromosome)

Entrée :*Liste* : la liste des clients issue du chromosome à décoder*Tournees* : l'ensemble des tournées contenant les clients déjà servis*NbVehicules* : le nombre des véhicules**Sortie :***Solution* : L'ensemble des tournées prévisionnelles**Début :***Solution* = *Tournees***TantQue** *Liste* n'est pas vide **Répéter***v* = 0*v_{min}* = *v**c* = le premier client dans *Liste**f_{min}* = le sous fitness issu de l'insertion du client *c* à la fin de la tournée *v* dans *Solution*// Le sous fitness est la valeur de la fonction objectif de la partie de *Solution* faite// jusqu'à l'insertion du *c**v* ++**TantQue** *v* < *NbVehicules* **Répéter***f* = le sous fitness issu de l'insertion du client *c* à la fin de la tournée *v* dans *Solution***Si** (*f* < *f_{min}*)*f_{min}* = *f**v_{min}* = *v***FinSi***v* ++**FinTantQue**Insérer le client *c* à la fin de la tournée numéro *v_{min}* dans *Solution*Supprimer le client *c* de *Liste***FinTantQue****Fin.**

pour l’heuristique d’insertion. Un client i , placé avant un autre client j dans le chromosome et affecté au même véhicule sera toujours servi en premier dans la tournée correspondante. L’ordre indiqué dans le chromosome est donc important pour la qualité des solutions décodées ; en effet, nous avons utilisé une heuristique simple pour réduire le temps de décodage parce que nous traitons un problème dynamique où le temps joue un grand rôle. Pour compenser le fait que le décodage n’examine pas toutes les positions d’insertion possibles dans la solution décodée, nous avons utilisé une seconde heuristique pour l’insertion d’un nouveau client dans le chromosome (voir la section 3.4.3.2).

En fait, en utilisant ce codage indirect nous pouvons représenter aussi plusieurs classes de problèmes (i.e. le problème du voyageur de commerce TSP, le problème du réparateur TRP, le problème de ramassage et livraison des colis PDP, etc.) et les traiter avec le même outil. Comme nous l’avons déjà mentionné dans cette thèse, nous traitons principalement le problème DVRPTW, mais nous donnons dans le chapitre 5, les résultats de notre approche pour d’autres classes de problèmes. La complexité de cette procédure est en $O(n * m)$, où n est le nombre des clients à servir et m est le nombre des véhicules.

Nous avons pu ainsi traiter d’autres problèmes dynamiques en changeant principalement la fonction objectif, comme le problème DVRP sans fenêtres de temps, le problème dynamique du réparateur, le problème dynamique du voyageur de commerce avec fenêtres de temps DTSPWT et le problème dynamique du ramassage et de la livraison de colis avec fenêtres de temps DPDPTW.

Pour ce dernier problème, nous faisons une petite modification sur le décodage du chromosome (DPDPTW), puisque nous avons deux positions géographiques (la position de ramassage et celle de livraison) pour chaque client. Le chromosome contient deux occurrences identiques (la première est pour le ramassage d’un colis et la deuxième pour la livraison de ce même colis) de chaque client à servir (voir la figure (3.8)).

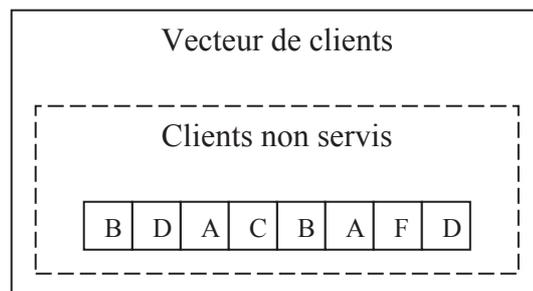


FIG. 3.8 – Le chromosome du problème DPDPTW

Pour décoder ce chromosome, nous utilisons la même heuristique de décodage sauf que nous vérifions d’abord si c’est la première ou la deuxième occurrence du client. Par exemple dans la figure (3.8), la première occurrence de B signifie que c’est la position de ramassage du client B , donc nous utilisons l’heuristique de décodage déjà expliquée. Dans ce cas, l’insertion peut se faire au choix sur l’un des véhicules de la flotte. Pour la deuxième occurrence de B , cela signifie que c’est la position de livraison du client B , dans ce cas là, nous devons obligatoirement associer celle-ci à la fin de la tournée précédemment associée au point de ramassage. Dans ce problème aussi on ne prend pas en compte la contrainte de capacité de véhicules.

3.3.4 Population initiale

La construction d'une population initiale représente l'étape de départ de l'algorithme génétique. En général, cette population est créée de façon aléatoire. Il est possible de placer les individus initiaux de façon uniforme dans l'espace de recherche. Une partie de ces individus peut être construite en utilisant des méthodes heuristiques simples pour obtenir des individus de bonne qualité. Par exemple l'heuristique de **PFIH** (Push Forward Insertion Heuristic) a été utilisée par [Zhu 2000] ou l'heuristique de *sweep* a été utilisée par [Baker et al. 2003]. Cette dernière méthode permet d'accroître la vitesse de convergence, mais empêche l'algorithme de se focaliser sur une autre partie de l'espace de recherche qui pourrait contenir l'optimum. Dans notre approche nous avons choisi de générer aléatoirement la population initiale.

3.3.5 Sélection

A chaque génération, l'opérateur de sélection sert à former une population intermédiaire à la population de la génération courante i . Ces individus seront ensuite croisés et mutés pour former la population de la génération suivante $i + 1$. Les chromosomes sélectionnés représentent les individus les mieux adaptés. Il existe de nombreuses techniques de sélection, nous allons présenter les plus connues :

1. La plus simple, connue sous le nom de "*ranking*" consiste à classer les n chromosomes de la population par ordre croissant de leur évaluation respective (ou décroissant selon l'objectif). Les m premiers individus sont ensuite sélectionnés. Ainsi, seuls les meilleurs individus sont conservés.
2. La sélection par roulette ou "*roulette wheel*" : elle consiste à associer à chaque chromosome un segment dont la longueur est proportionnelle à son fitness. Ces segments sont ensuite concaténés sur un axe gradué que l'on normalise entre 0 et 1. On tire alors un nombre aléatoire de distribution uniforme entre 0 et 1, puis on identifie le segment sélectionné et le chromosome correspondant. Avec cette technique, les bons chromosomes seront plus souvent sélectionnés que les mauvais, et un même chromosome pourra avec cette méthode être sélectionné plusieurs fois. Néanmoins, sur des populations de petite taille, il est difficile d'obtenir exactement l'espérance mathématique de sélection à cause du faible nombre de tirages. On aura donc un biais de sélection plus ou moins fort suivant la dimension de la population.
3. *Sélection par tournoi* : Cette technique tire au hasard deux ou plusieurs individus de la population et le plus fort est sélectionné, c'est-à-dire celui ayant le fitness le plus intéressant.
4. La sélection *aléatoire* : comme son nom indique, ce type de sélection choisit le chromosome selon une distribution uniforme.

Dans cette thèse, nous avons choisi la sélection par **tournoi** qui est utilisée le plus souvent par de nombreux auteurs [Goldberg et al. 1992], [Deb et al. 1999].

3.3.6 Reproduction

Afin de faire évoluer la population, nous appliquons des *opérateurs génétiques*. Pour cela, nous définissons les deux opérateurs génétiques fondamentaux :

3.3.6.1 Le croisement (Crossover)

Le croisement permet d'accoupler deux parents pour former deux enfants. L'idée est que les enfants garderont les meilleures caractéristiques de leurs parents. Prenons l'exemple de la fonction $f(x)$ décrite par la figure 3.9. Le principe du croisement consiste à recombinaison les bonnes parties des chromosomes des deux parents P_1 et P_2 pour engendrer deux enfants E_1 et E_2 de meilleure qualité. La difficulté réside dans le fait que la bonne partie du chromosome n'est pas connue et qu'il est alors possible de créer des enfants de plus mauvaise qualité que celles des parents initiaux. L'enfant E_3 est un exemple de cette difficulté dans le cas d'une minimisation de la fonction $f(x)$. Cette difficulté est d'autant plus importante lorsqu'il s'agit d'un codage de type indirect puisque les bonnes parties des solutions parentes sont “masquées” par celui-ci.

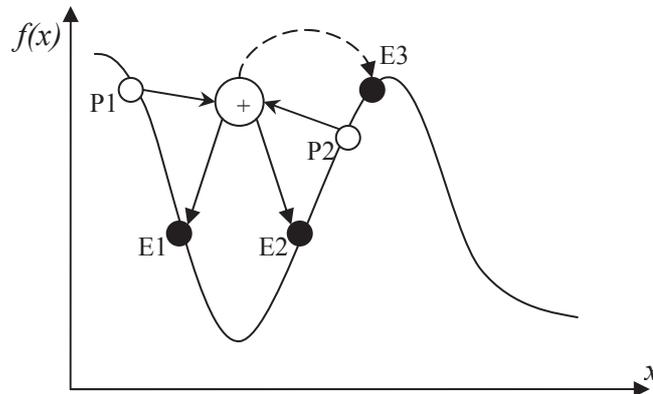


FIG. 3.9 – Principe du croisement

Il existe de nombreuses techniques de croisement, nous allons présenter celles utilisées pour la résolution du problème VRPTW :

IX (1-point) : un point de coupure est déterminé de façon aléatoire sur les gènes des parents. Les enfants sont générés en échangeant les morceaux de parents ainsi constitués [Mitchell 1998]. Nous allons présenter un exemple pour illustrer cet opérateur :

- Considérons par exemple les deux parents :

$$P_1 = 8 \ 2 \ 3 \ 6 \ 5 \ 4 \ 7 \ 1 \ 9$$

$$P_2 = 4 \ 5 \ 2 \ 1 \ 8 \ 7 \ 6 \ 9 \ 3$$

- D’abord, nous choisissons aléatoirement un point de coupure, par exemple, la quatrième position dans les deux parents :

$$P_1 = 8 \ 2 \ 3 \ 6 \ | \ 5 \ 4 \ 7 \ 1 \ 9$$

$$P_2 = 4 \ 5 \ 2 \ 1 \ | \ 8 \ 7 \ 6 \ 9 \ 3$$

- Ensuite, nous construisons les deux enfants comme suit :

$$E_1 = 4 \ 5 \ 2 \ 1 \ | \ x \ x \ x \ x \ x$$

$$E_2 = 8 \ 2 \ 3 \ 6 \ | \ x \ x \ x \ x \ x$$

- Puis, la partie restante dans les deux enfants sera remplie par les clients dans l'ordre d'apparition du parent correspondant et qui n'appartiennent pas à la première partie de l'enfant :

$$P_1 = 8\ 2\ 3\ 6\ 5\ 4\ 7\ 1\ 9 \Rightarrow E_1 = 4\ 5\ 2\ 1\ | \ x\ x\ x\ x\ x$$

$$P_2 = 4\ 5\ 2\ 1\ 8\ 7\ 6\ 9\ 3 \Rightarrow E_2 = 8\ 2\ 3\ 6\ | \ x\ x\ x\ x\ x$$

- Donc les deux enfants deviennent finalement :

$$E_1 = 4\ 5\ 2\ 1\ | \ 8\ 3\ 6\ 7\ 9$$

$$E_2 = 8\ 2\ 3\ 6\ | \ 4\ 5\ 1\ 7\ 9$$

PMX (Partially Mapped Crossover) : Le PMX fut proposé par [Goldberg et al. 1985] dans le cadre de problèmes d'ordonnancement. Le but était de construire un enfant par le choix de sous séquences ordonnancées de l'un des parents et de préserver l'ordre et la position d'autant de sous séquences que possible des autres parents. La sous-séquence de l'ordonnancement est sélectionnée par le choix de deux points de coupure aléatoire, lesquels servent de frontière pour l'opération de substitution. Considérons par exemple les deux parents :

$$P_1 = 123\ | \ 4567\ | \ 89$$

$$P_2 = 452\ | \ 1876\ | \ 93$$

- **Etape 1** : ces deux parents vont produire deux enfants. Dans cette première étape, les segments compris entre les points de coupures sont échangés :

$$E_1 = xxx\ | \ 1876\ | \ xx$$

$$E_2 = xxx\ | \ 4567\ | \ xx$$

Notons qu'en vue de procéder à l'étape 3 de résolution des conflits, il est possible d'établir entre les deux points de coupures une table de correspondances entre les allèles ; cette table est définie comme suit dans notre exemple :

$$1 \leftrightarrow 4, 8 \leftrightarrow 5, 7 \leftrightarrow 6, 6 \leftrightarrow 7$$

- **Etape 2** : les chromosomes "*Enfants*" sont complétés en transmettant les allèles non conflictuelles issus du second parent.

$$E_1 = x23\ | \ 1876\ | \ x9$$

$$E_2 = xx2\ | \ 4567\ | \ 93$$

- **Etape 3** : cette dernière étape consiste à terminer l'élaboration des descendants en résolvant les conflits à l'aide de la table de correspondance :

$$E_1 = 423\ | \ 1876\ | \ 59$$

$$E_2 = 182\ | \ 4567\ | \ 93$$

MX1T/D : Ce croisement a été proposé par [Sushil et al. 1999]. Il est composé de deux parties : le croisement heuristique (**MX1D**) et le croisement de fusion (**MX1T**). Le premier (MX1D) est basé sur les distances entre les clients et le deuxième (MX1T) est basé sur la borne inférieure des fenêtres de temps des clients. Dans ce croisement, deux parents produisent un seul descendant avec le croisement heuristique. Le deuxième descendant est obtenu à partir des mêmes parents avec le croisement de fusion.

1. **MX1T** : Nous allons expliquer ce type de croisement en utilisant l'exemple suivant :

$$P_1 = \underline{2} \ 5 \ 6 \ 1 \ 0 \ 7 \ 3 \ 8 \ 4 \ 9$$

$$P_2 = \underline{4} \ 1 \ 6 \ 9 \ 3 \ 8 \ 2 \ 0 \ 5 \ 7$$

Cet opérateur utilise un vecteur global de précédence. A titre d'exemple, considérons le vecteur suivant :

$$0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8 \ 9$$

Ce vecteur est basé sur la borne inférieure de la fenêtre de temps (e_i) de chaque client. Les clients dans ce vecteur, sont triés en ordre croissant selon leur borne (e_i). Dans ce vecteur, le client 1 peut être servi avant les clients 2, 3 et ainsi de suite. Les clients à l'étude sont soulignés. Le premier client de P_1 s'avère être placé dans le vecteur global de précédence plus tôt que le premier client de P_2 . Ce client est placé comme premier client dans l'enfant E_1 . Pour maintenir la faisabilité de la suite de l'élaboration du chromosome, le premier client de P_2 est permuté. C'est le client 4 qui est permuté avec le client 2 (accentués en gras) :

$$P_1 = 2 \ \underline{5} \ 6 \ 1 \ 0 \ 7 \ 3 \ 8 \ 4 \ 9$$

$$P_2 = \mathbf{2} \ \underline{1} \ 6 \ 9 \ 3 \ 8 \ 4 \ 0 \ 5 \ 7$$

$$E_1 = 2 \ x \ x \ x \ x \ x \ x \ x \ x \ x$$

Le deuxième client de P_2 apparaît dans le vecteur de précédence avant celui de P_1 , ainsi nous le plaçons en deuxième client de E_1 . Nous permutons le deuxième client de P_1 pour maintenir la faisabilité, et nous continuons de cette façon jusqu'à la fin des chromosomes :

$$P_1 = 2 \ \mathbf{1} \ 6 \ \mathbf{5} \ 0 \ 7 \ 3 \ 8 \ 4 \ 9$$

$$P_2 = 2 \ 1 \ 6 \ 9 \ 3 \ 8 \ 4 \ 0 \ 5 \ 7$$

$$E_1 = 2 \ 1 \ x \ x \ x \ x \ x \ x \ x \ x$$

Finalement, l'enfant obtenu est comme suit :

$$E_1 = 2 \ 1 \ 6 \ 5 \ 0 \ 7 \ 3 \ 4 \ 8 \ 9$$

2. **MX1D** : Cet opérateur exploite les distances entre les clients. Par exemple, nous avons les deux parents suivants :

$$P_1 = \underline{2} \ 5 \ 6 \ 1 \ 0 \ 7 \ 3 \ 8 \ 4 \ 9$$

$$P_2 = \underline{4} \ 1 \ 6 \ 9 \ 3 \ 8 \ 2 \ 0 \ 5 \ 7$$

D'abord, nous choisissons aléatoirement l'un des deux premiers clients des parents (par exemple, le client 4 de P_2), et nous initialisons l'enfant E_2 avec ce client :

$$\begin{aligned}
P_1 &= \underline{4} \underline{5} 6 1 0 7 3 8 2 9 \\
P_2 &= \underline{4} \underline{1} 6 9 3 8 2 0 5 7 \\
E_2 &= 4 x x x x x x x x
\end{aligned}$$

Maintenant nous comparons la distance entre le client 4 et le client 5, et la distance entre le client 4 et le client 1. Si $d_{4,5} > d_{4,1}$, nous choisissons alors 1 pour être le prochain client de E_2 . Pour les mêmes raisons que précédemment (faisabilité de la poursuite), on doit permuter le client 1 et le client 5 dans P_1 . On continue de la même manière jusqu'à ce qu'un nouveau chromosome soit entièrement produit E_2 .

$$\begin{aligned}
P_1 &= 4 \underline{1} \underline{6} 5 0 7 3 8 2 9 \\
P_2 &= 4 \underline{1} \underline{6} 9 3 8 2 0 5 7 \\
E_2 &= 4 1 x x x x x x x \\
&\dots \\
E_2 &= 4 1 6 5 3 7 2 8 0 9
\end{aligned}$$

Les parents choisis par l'opérateur de sélection ne seront pas tous croisés. Une probabilité P_{Cross} permet de décider si les parents seront croisés entre eux ou s'ils seront tout simplement copiés. Le croisement est davantage basé sur l'exploitation du voisinage des parents que sur l'exploration de l'espace de recherche.

3.3.6.2 La mutation

La mutation permet d'explorer l'espace de recherche en évitant à l'algorithme de converger trop rapidement vers un optimum local. La figure 3.10 présente un exemple de fonction $f(x)$ à optimiser. Les jetons noirs représentent les chromosomes de la population qui ont tous convergé vers un optimum local. Le principe de la mutation est donc de prendre un de ces chromosomes et de le modifier afin d'explorer une autre partie de l'espace de recherche que l'on espère plus intéressante.

La mutation est traditionnellement considérée comme un opérateur marginal bien qu'elle confère en quelque sorte aux algorithmes génétiques la propriété d'ergodicité (i.e. tous les points de l'espace de recherche peuvent être atteints). Cet opérateur est donc d'une grande importance. Il a de fait un double rôle :

- celui d'effectuer une recherche locale
- et/ou de sortir d'une vallée.

Il existe de nombreuses façons de réaliser la mutation sur un chromosome. Cet opérateur fait appel à la notion de voisinage. Nous allons rappeler trois techniques basiques de mutation ou de voisinage :

1. **1-Opt** (nommé aussi *swap*) : Cet opérateur est inspiré de technique de mutation utilisée pour la résolution de problèmes de voyageur de commerce (TSP) [Grefenstette et al. 1985]. Cet opérateur est le plus répandu. Il consiste à prendre au hasard deux gènes (2 clients) du chromosome, D et B dans l'exemple (voir la figure 3.11) et à les inverser.

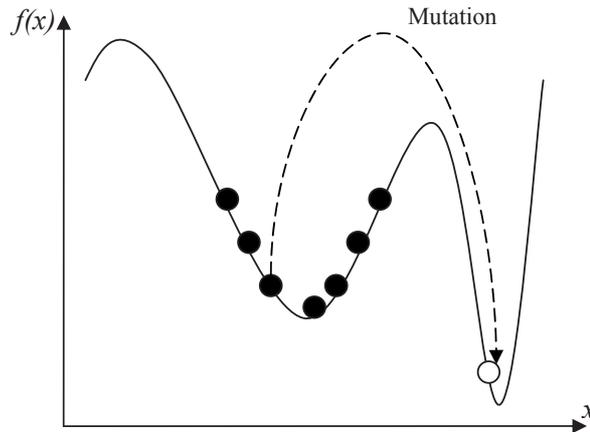


FIG. 3.10 – Principe de la mutation

2. **2-Opt** : Dans la littérature, cet opérateur appartient à la famille des opérateurs $\lambda - Opt$ ($\lambda = 2, 3, 4, \dots, n$) proposée par [Lin 1965], il s’agit de remplacer λ arcs existants (l’arc est la liaison entre deux clients consécutifs dans une tournée) par λ nouveaux arcs. Dans notre cas, il est impossible d’utiliser directement cette famille parce que le chromosome ne permet pas de distinguer clairement les tournées dans le codage indirect que nous utilisons. Donc, nous avons du adapter l’opérateur $2 - Opt$. Celui-ci choisit aléatoirement deux suites de clients dont la taille est égal à deux, puis les échange.
3. **Or-Opt** : Opère de la même manière, mais les tailles des suites choisies peuvent varier entre 0 et 3. Cet opérateur échange deux suites ou il déplace une suite vers une autre position si la taille de l’une des suites est égale à zéro (voir la figure 3.12). L’opérateur *CROSS-exchange* proposé dans [Taillard et al. 1997] est une généralisation de cet opérateur.

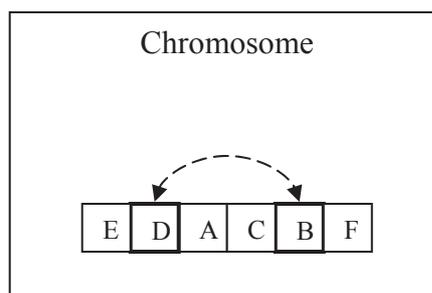


FIG. 3.11 – L’opérateur de mutation 1-Opt

L’opérateur de mutation ne s’applique pas à tous les chromosomes. Chacun a une probabilité P_{Mut} , d’être muté. Cette probabilité est en général très faible. Une fois les chromosomes croisés et mutés, il faut les réévaluer afin de pouvoir sélectionner la prochaine population pour l’itération suivante. Cette évaluation se réalise à l’aide de la fonction d’évaluation, c’est-à-dire la fonction objectif du problème traité.

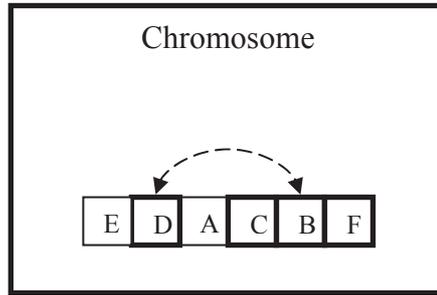


FIG. 3.12 – L'opérateur de mutation Or-Opt

3.3.6.3 L'élitisme

A la création d'une nouvelle population, il y a de grandes chances pour que les meilleurs chromosomes soient perdus après les opérations de croisement et de mutation. Pour éviter cela, on utilise une méthode d'élitisme. Elle consiste à copier un ou plusieurs des meilleurs chromosomes dans la nouvelle génération. Ensuite, on génère le reste de la population selon l'algorithme de reproduction usuel. Cette méthode améliore en général considérablement les algorithmes génétiques, car elle permet de ne pas perdre les meilleures solutions trouvées au cours de l'évolution [Michalewicz 1994].

Dans notre approche, les enfants générés par les opérateurs génétiques, sont copiés dans une population temporaire. Nous obtiendrons donc deux populations ; la première est la population courante contenant les chromosomes parents et la deuxième est la population contenant les chromosomes enfants. Ensuite, nous trions les deux populations en ordre croissant des fitness des chromosomes. Puis, nous construisons la nouvelle population pour la prochaine génération de la manière suivante : nous remplissons un pourcentage de la nouvelle population avec les meilleurs chromosomes de la population triée tandis que l'autre partie de la nouvelle population est remplie par des chromosomes générés aléatoirement (voir la figure 3.13) afin d'introduire de la diversité.

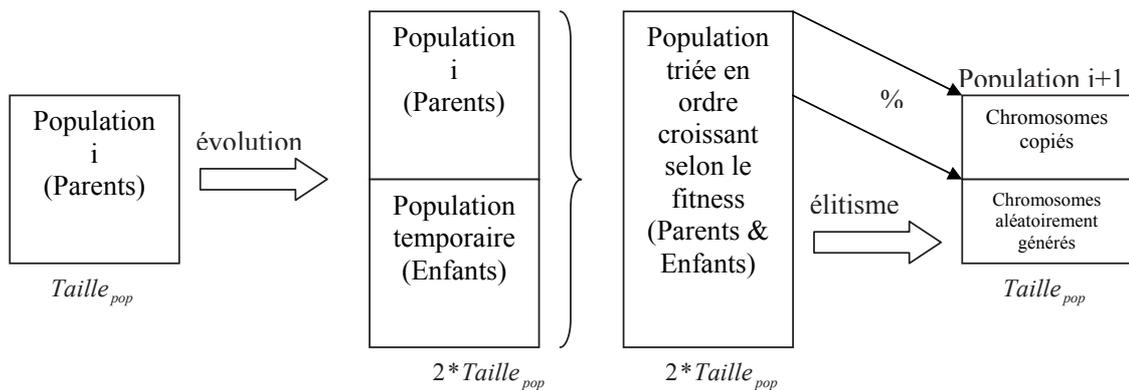


FIG. 3.13 – L'évolution générale de la population

A ce stade, nous avons présenté les différentes étapes de notre approche qui concerne l'adaptation de l'algorithme génétique au problème VRPTW. Nous allons continuer avec la partie de notre approche qui traite les événements dynamiques présents dans le problème DVRPTW. Pour

prendre en compte ce caractère, un gestionnaire événementiel a été couplé à notre algorithme génétique. Pour tester celui-ci en situation, un environnement de simulation a été développé.

3.4 Environnement de Simulation

Pour traiter le problème VRPTW dynamique, nous avons développé un simulateur ayant la capacité de simuler une journée de service. Le simulateur s’appuie sur deux composants principaux qui coopèrent. Un composant **Algorithme Génétique (AG)** qui cherche en tâche de fond à améliorer la population courante selon l’algorithme génétique présenté ci-dessus. Un composant **Gestion d’Événements (GE)** qui attend l’arrivée d’un événement et déclenche les actions appropriées à cet événement. L’ensemble de ces deux composants forme le planificateur (voir la figure 3.14). Dans le planificateur, nous avons aussi le composant “*Graphique*” qui sert à afficher le déroulement de la simulation et les résultats obtenus; et le composant “*Contrôleur*” qui gère le fonctionnement de tous les composants du planificateur.

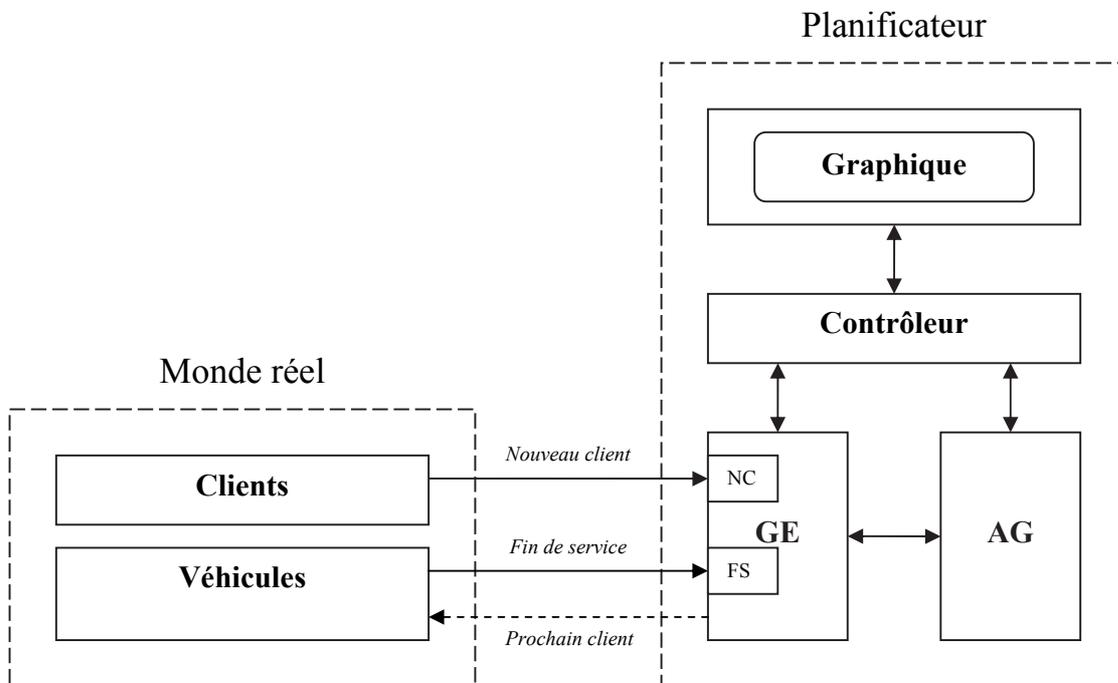


FIG. 3.14 – Simulateur pour le DVRPTW

3.4.1 Le composant Algorithme Génétique (AG)

Cette partie de notre planificateur s’occupe de tout ce qui concerne les opérations génétiques sur la population courante. Comme nous l’avons déjà vu, l’évaluation d’un chromosome passe par une phase de décodage qui utilise une heuristique d’insertion.

3.4.2 Le composant Gestion d’Événements (GE)

Ce composant traite les deux types d’événements (“*arrivée d’un nouveau client*”, “*fin de service d’un client*”) que l’on cherche à générer par notre simulateur. L’événement “*arrivée d’un*

nouveau client” vient en fait du Benchmark (fichier de données) qui décrit le problème dynamique à résoudre. Une procédure appropriée à l’évènement est appliquée sur la population courante.

3.4.3 Le fonctionnement du simulateur

Le temps réel d’ouverture d’une journée de service $[0, T]$ est simulé à l’aide d’une durée de simulation $[0, T_{sim}]$. Cette durée est divisée en petites tranches de temps ayant des tailles variables selon la durée d’une génération (G_k) de l’algorithme génétique. Après chaque génération nous devons calculer le temps écoulé pour vérifier l’apparition éventuelle de nouveaux évènements (figure 3.15). Le temps T_{sim} est beaucoup petit que la durée réelle T d’une journées et permet d’avoir des résultats satisfaisants sur des périodes de simulation courtes.

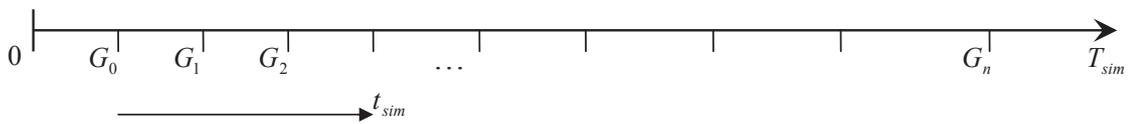


FIG. 3.15 – Durée de la simulation

En l’absence d’évènement, l’algorithme génétique continue à faire évoluer la population au cours du temps. Dans le cas contraire, de nouveaux évènements sont à prendre en compte. Examinons (figure 3.16) l’évolution de la situation quand trois nouvelles demandes apparaissent au cours du temps. Nous avons pris comme hypothèse dans cette figure que la durée prise par l’AG pour produire une nouvelle population est de quatre unités de temps. Le prise en compte d’un nouveau client C_i au temps G_{j+3} entraîne une augmentation du temps nécessaire à l’AG pour produire la prochaine population. En effet l’apparition d’un nouveau client accroît la taille de chaque chromosome de la population courante qui dépend du nombre de clients à servir. Le temps de calcul de l’AG passe alors par exemple à cinq unités de temps. L’application des opérateurs génétiques prend plus de temps. Ce phénomène peut se reproduire à chaque apparition d’un nouveau client (par exemple : C_{i+1} , C_{i+2} , etc).

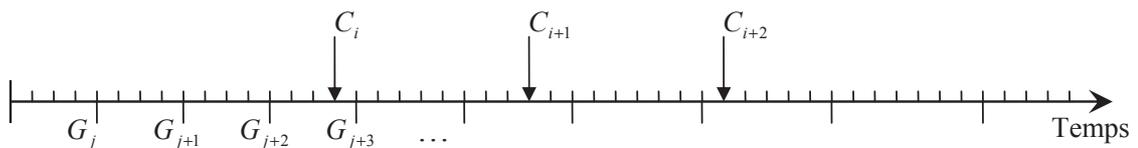
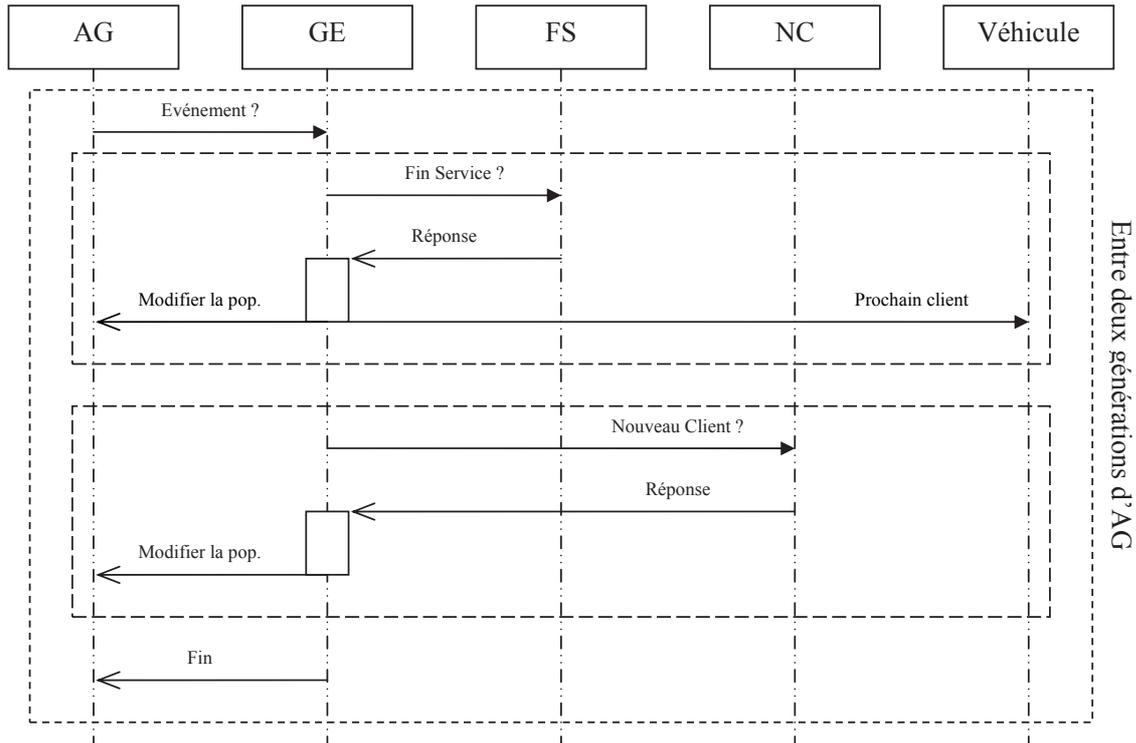


FIG. 3.16 – La simulation de la journée de service

Le planificateur fonctionne en deux étapes décrites comme suit :

1. Dans la première étape, le planificateur utilise l’algorithme génétique pour résoudre le problème VRPTW statique. Dans cette étape nous cherchons à trouver les meilleures solutions pour les clients statiques, ceux-ci représentent les clients connus à l’avance (i.e. avant le début de la journée). A la fin de cette étape nous obtenons une population initiale de solutions pour le problème statique.
2. La deuxième étape de notre planificateur s’occupe de la partie dynamique du problème DVRPTW. Cette étape fonctionne comme suit :
 - (a) Initialiser le temps de simulation t_{sim} à zéro.

- (b) Faire évoluer la population courante et calculer la nouvelle valeur du temps de simulation t_{sim} .
- (c) Vérifier si un ou plusieurs évènements se sont produit pendant la dernière génération (Figure 3.17).
 - Si oui, démarrer le GE correspondant à chaque évènement.
- (d) Retourner en (2b) si le temps écoulé n’est pas arrivé à la fin de la simulation.



- AG : Le composant Algorithme génétique
- GE : Le composant Gestion d’Événements
- FS : La file d’attente « Fin de service »
- NC : La file d’attente « Nouveaux clients »
- Véhicule : Le composant « Flotte de véhicules »

FIG. 3.17 – Diagramme UML montrant la séquence d’interactions entre les composants AG et GE, à la fin d’une génération

La figure 3.17 représente les interactions entre les deux composants AG et GE du planificateur ; cette interaction se fait après chaque génération de l’AG. Elle commence par l’envoi d’un message d’interrogation “Évènement?”. Ce message permet de vérifier l’existence de nouveaux évènements. L’interaction se termine par le message “Fin”. Entre ces deux messages, le composant *GE* vérifie l’apparition de nouveaux évènements ; il vérifie d’abord l’existence de nouveaux évènements “Fin de service”, ensuite il vérifie les nouveaux évènements “Nouveaux clients” et il déclenche les services correspondants.

Dans ce simulateur, nous calculons le temps de simulation écoulé (t_{sim}) de la manière sui-

vante :

$$t_{sim} = t_{cpu} \cdot \frac{T_{sim}}{T}$$

Où, t_{cpu} représente la durée d'utilisation du CPU mesurée en minutes, écoulée depuis le début de la simulation. T est la durée d'ouverture du dépôt (la fin de la journée de service). T_{sim} représente l'horizon de simulation de la journée de service. Par exemple, nous avons pris pour nos simulations $T_{sim} = 30 \text{ min}$. Ainsi, le temps de simulation écoulé (t_{sim}) est variable, il dépend de la durée de l'évolution d'une population. Cette durée dépend toujours de la taille des chromosomes.

Comme nous l'avons énoncé, au cours de la simulation, la taille des chromosomes varie (i.e. elle correspond aux clients restant à servir par les véhicules) mais pour une population fixée, tous les chromosomes ont une taille identique. En fin de simulation (i.e. fin de la journée de service) si chaque chromosome a une taille égale à zéro, cela signifie que tous les clients du problème ont été servis. Dans le cas contraire (i.e. taille différente de zéro), il existe des clients en fin de journée qui n'ont pu être servis par les véhicules. Ces clients pourront être pris en compte la journée suivante et seront considérés dans ce cas comme des clients statiques.

Nous avons déjà vu que pour chaque évènement, nous appliquons une procédure correspondante à cet évènement ; en conséquence nous avons donc dû construire deux procédures répondant aux deux types d'évènements, ce sont :

1. L'évènement "fin de service".
2. L'évènement "arrivée d'un nouveau client".

3.4.3.1 L'évènement "fin de service"

Une procédure est conçue pour traiter l'évènement fin de service d'un client (algorithme 3.2). Le planificateur doit fournir au conducteur du véhicule correspondant, le prochain client à servir. Dans le cas où il n'existe plus de clients à servir par ce véhicule, alors celui-ci retourne au dépôt.

Algorithme 3.2 Algorithme de traitement de l'évènement "fin de service"

Entrée :

Vehicule : Le véhicule qui a généré l'évènement.

Chrom : Meilleur chromosome dans la population.

Tournees : L'ensemble des tournées contenant les clients déjà servis.

Sortie :

Tournees : L'ensemble des tournées contenant les clients déjà servis et le client associé au *Vehicule*.

Début :

Solution : L'ensemble des tournées prévisionnelles

Solution = Décoder le chromosome *Chrom*.

Ajouter le premier client *Client* non encore servi dans la tournée *Vehicule*, à la tournée correspondante dans *Tournees*.

Supprimer le client *Client* de tous les chromosomes de la population.

Fin.

Par exemple, la figure 3.18 montre le déroulement de cette procédure au moment où le véhicule v_3 a fini de servir le client J . Elle commence par la récupération du meilleur chromosome de la population courante (celui ayant le meilleur fitness). Elle décode ensuite celui-ci pour obtenir la solution correspondante. Puis, le planificateur peut simplement décider quel est le prochain client à servir (i.e. le client E dans notre exemple). Finalement, on élimine le client choisi de l'ensemble des chromosomes (vecteur des clients) de la population courante et on ajoute celui-ci dans l'ensemble des tournées de véhicules.

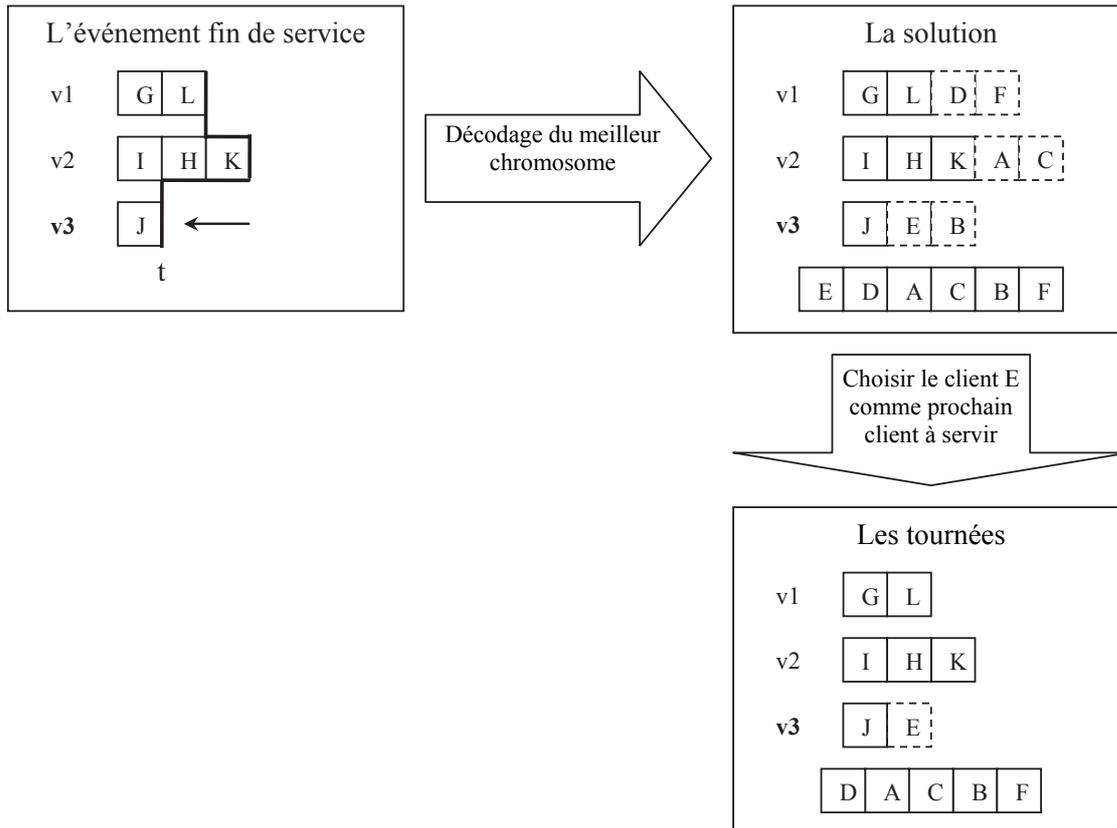


FIG. 3.18 – Procédure d'évènement fin de service

3.4.3.2 L'évènement “arrivée d'un nouveau client”

Une procédure est utilisée pour chaque évènement “arrivée d'un nouveau client” (algorithme 3.3). Elle utilise une heuristique d'insertion pour ajouter le nouveau client dans la meilleure position de chaque chromosome.

La Figure (3.19) montre un exemple d'évènement “arrivée d'un nouveau client” au moment où le client G demande un service. La procédure cherche à insérer ce client dans tous les chromosomes de la population courante. L'opération d'insertion au niveau d'un chromosome consiste à tester l'insertion du client G dans toutes les positions possibles du chromosome considéré. A chaque essai, nous évaluons le fitness du chromosome obtenu, puis nous pouvons choisir la position d'insertion du client G qui donne le meilleur fitness. Cette procédure est simple et sa complexité est en $O(n^2)$ où n représente le nombre de clients.

Algorithme 3.3 Algorithme de traitement de l'évènement "nouveau client"**Entrée :**

Client : le nouveau client
Chrom : le chromosome de taille N

Sortie :

Chrom : le chromosome de taille $N + 1$

Début :

$pos = 0$
 f_{min} = le fitness de *Chrom* prenant en compte le client *Client* à la position pos
 $pos ++$

TantQue ($pos \leq$ la taille de *Chrom*) **Répéter**

f = le fitness de *Chrom* prenant en compte le client *Client* à la position pos

Si ($f < f_{min}$)

$f_{min} = f$

$p_{min} = pos$

FinSi

$pos ++$

FinTantQue

Insérer le client *Client* à la position p_{min} dans *Chrom*

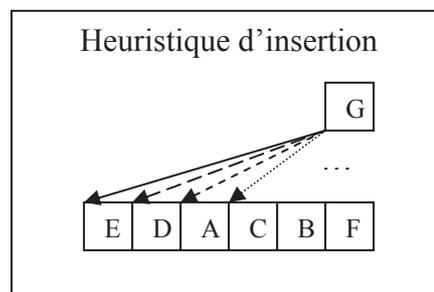
Fin.

FIG. 3.19 – Procédure d'évènement nouveau client

Finalement, il faut rappeler que le planificateur fonctionne *en ligne* ; c’est-à-dire qu’il réagit directement après chaque événement. Comme nous l’avons évoqué, il continue à faire évoluer la population en arrière plan tant qu’il n’y a pas de nouveaux événements à traiter.

3.5 Conclusion

Dans ce chapitre nous avons présenté une méthode dynamique basée sur un *algorithme génétique* pour la résolution du problème **DVRPTW**. Nous avons présenté les différents constituants de notre algorithme et les modifications faites pour que l’algorithme puisse traiter ce type de problème dans un cadre dynamique. Pour évaluer le bon fonctionnement de l’approche, nous avons construit un simulateur capable de simuler une journée de service. La représentation chromosomique est de type *indirecte*, c’est un vecteur de taille variable selon le nombre de clients restant à servir par les véhicules. Pour le décodage des chromosomes et le traitement de l’arrivée d’un nouveau client, nous avons utilisé une heuristique d’insertion. Sur ces chromosomes nous avons appliqué simplement des opérateurs génétiques classiques utilisés précédemment en ordonnancement ou en routage.

Enfin, nous pouvons remarquer que notre approche est la première qui utilise le paradigme algorithme génétique pour la résolution du problème DVRPTW “en ligne”. Elle prend en compte les événements après chaque itération de l’AG, c’est-à-dire avec une réponse quasi-immédiate. Ceci n’est pas le cas de [Larsen 2001] qui collecte les nouvelles demandes pendant une tranche de temps et traite ces demandes dans la prochaine tranche de temps. Avec de petites modifications sur cette approche telle que le changement dans la fonction objectif, nous pouvons utiliser notre approche pour la résolution d’autres problèmes dynamiques de VRP avec ou sans fenêtres de temps et ses extensions (DTRP, DTSPTW, DPDPTW, etc). Notre approche est relativement flexible et nous pensons l’adapter pour qu’elle puisse fonctionner en mode parallèle.

L’inconvénient de notre approche est que nous utilisons l’algorithme génétique comme méthode métaheuristique de résolution. A ce stade, nous avons encore un problème à résoudre, celui du réglage des principaux paramètres de l’algorithme génétique. Ce problème est traité dans le chapitre suivant.

Chapitre 4

Le réglage des paramètres de l’algorithme génétique

Sommaire

| | | |
|------------|-----------------------------------------------------------------------------------|------------|
| 4.1 | Introduction | 79 |
| 4.2 | Revue sur la littérature | 81 |
| 4.2.1 | Réglage de paramètres | 81 |
| 4.2.2 | Contrôle de paramètres | 82 |
| 4.3 | Les plans d’expériences | 84 |
| 4.3.1 | Introduction | 84 |
| 4.3.2 | Terminologie | 85 |
| 4.3.3 | Les types des plans | 85 |
| 4.3.4 | Démarche générale pour utiliser des plans d’expériences | 87 |
| 4.4 | Application des plans d’expériences pour le réglage des paramètres de l’AG | 88 |
| 4.4.1 | Etape 1 - Le plan de criblage | 92 |
| 4.4.2 | Etape 2 - Plans d’expériences complets vs. fractionnaires | 97 |
| 4.4.3 | Etape 3 - Plans d’expériences fractionnaires pour les problèmes dynamiques | 104 |
| 4.5 | Conclusion | 106 |

4.1 Introduction

Dans ce chapitre nous allons présenter la méthodologie que nous avons utilisée pour régler les paramètres de l’algorithme génétique. Celle-ci est illustrée pour la résolution du problème VRPTW dynamique.

Nous avons déjà vu que les algorithmes génétiques sont des méthodes stochastiques de recherche basées sur des principes dérivés de l’évolution naturelle (*Darwinisme*). Elles reposent sur les processus génétiques des organismes biologiques. Beaucoup de chercheurs de ce domaine ont fréquemment employé divers groupes de fonctions mathématiques afin d’étudier la performance des algorithmes génétiques [Jason et al. 2002]. Une grande partie de la recherche concerne le réglage des paramètres et leur ajustement potentiel. Jusqu’à maintenant, aucune conclusion générale n’est disponible concernant la calibration optimale des paramètres tels que la taille de la population, le type de croisement ou le type de mutation.

La modification des valeurs des paramètres ou des composants d'un algorithme génétique peut considérablement affecter les performances de l'algorithme [Davis 1991]. Dans la littérature, nous remarquons que les chercheurs emploient différentes combinaisons des paramètres de l'algorithme génétique pour résoudre le même problème. Ils développent également de nouvelles techniques pour les opérateurs de croisement et de mutation afin d'accélérer ces algorithmes [Mahfouz 1999].

Il existe une pléthore d'études traitant des effets et des interactions entre les différents paramètres d'un AG [Deb et al. 1999]. En effet, les paramètres de l'AG tels que la taille de population, les opérateurs génétiques, les probabilités d'appliquer ces opérateurs, la représentation chromosomique, le taux d'élitisme agissent l'un sur l'autre d'une manière complexe. D'une façon générale, leurs interactions dépendent en grande partie de la fonction à optimiser. Il est donc difficile de réaliser une analyse complète de toutes leurs interactions, c'est pourquoi les chercheurs ont utilisé différents outils d'analyse. Quelques études [Goldberg et al. 1992] traitent précisément des interactions isolées entre deux paramètres importants et ont permis de comprendre leur effet conjugué sur l'exécution de l'AG. Ces études isolées ont fourni des directives utiles pour choisir les valeurs des paramètres de l'AG. Afin d'observer les interactions de plusieurs paramètres de l'AG, des études empiriques ont été également employées [Lindsay et al. 1997]. Pour étudier l'évolution de ces interactions au cours du temps, des modèles stochastiques plus sophistiqués employant des chaînes de Markov ont été également développés et analysés [Chakraborty et al. 1996].

En se basant sur ces études, les observations suivantes peuvent être faites :

- La probabilité de mutation optimale (ou taux de mutation) est dépendante de la représentation utilisée [Tate et al. 1993]. Des travaux ont été également menés relativement à l'opérateur de croisement et des conclusions semblables ont été faites. Ils montrent que la probabilité de croisement optimal dépend en grande partie du codage utilisé [Kargupta et al. 1992], [Radcliffe 1991].
- Il a été suggéré que l'AG puisse correctement fonctionner avec une grande probabilité de croisement (≈ 0.9) et une petite probabilité de mutation (≈ 0.1) [Deb et al. 1999].
- Il a été observé que le meilleur individu de la population peut ne pas produire un nouvel enfant pour la prochaine génération. Une stratégie de sélection élitiste [Davis 1991] élimine ce risque en copiant le meilleur individu de chaque génération dans la génération suivante. En conséquence, une stratégie de sélection élitiste augmente la vitesse de domination d'un très bon individu sur une population. Avec cette stratégie, la performance de l'algorithme génétique semble s'améliorer. L'élitisme fait donc partie du paramétrage de l'AG que nous devons régler pour augmenter efficacement sa performance.
- La taille de la population a un rôle important sur le déroulement de l'algorithme génétique ; selon [Deb et al. 1999], ce paramètre doit avoir une valeur ni trop petite (≈ 10 individus) ni trop grande (≈ 100 individus).
- Enfin, la nature du problème traité affecte largement l'exécution de l'algorithme génétique. Autrement dit, pour chaque problème d'optimisation, nous avons une ou plusieurs combinaisons de réglages optimales ou proches du réglage optimal permettant de résoudre efficacement ce problème.

Comme on vient de le voir, le domaine lié au réglage optimal des paramètres d'un algorithme génétique reste très ouvert. Ce chapitre est divisé comme suit : la section suivante est consacrée aux travaux qui touchent à ce problème. Puis nous présentons la technique des plans d'expériences sur laquelle nous nous sommes appuyés pour présenter dans une dernière partie la démarche que nous avons suivie pour le réglage des paramètres de notre AG.

4.2 Revue sur la littérature

Cette section présente brièvement les travaux des chercheurs qui ont analysé les performances de divers modèles d'AG et leurs paramètres proposés. En fait, dans la littérature, il n'y a pas beaucoup de résultats généralisables. La plupart des chercheurs qui utilisent un algorithme génétique comme méthode d'optimisation métaheuristique pour résoudre leurs problèmes, règlent les paramètres expérimentalement. Autrement dit, ils lancent l'algorithme génétique plusieurs fois en utilisant différentes combinaisons de réglage et choisissent la combinaison qui donne les meilleurs résultats [Jason et al. 2002].

Dans la littérature, nous pouvons distinguer plusieurs méthodes utilisées pour trouver les meilleures valeurs de ces divers paramètres [Eiben et al. 1999]. Ces méthodes sont illustrées dans la figure (4.1).

Dans son étude, [Eiben et al. 1999] a distingué deux principales méthodes de configuration des paramètres :

1. La première (**Réglage de paramètres**) consiste à trouver des bonnes valeurs a priori, c'est-à-dire avant que l'algorithme génétique ne démarre. Ces valeurs restent inchangées pendant toute l'exécution de l'algorithme.
2. La deuxième (**Contrôle de paramètres**) consiste à trouver des valeurs initiales avant le démarrage de l'algorithme, puis à les faire évoluer tant que l'AG s'exécute. Les valeurs des paramètres s'adaptent en fonction du temps ou de l'état de la population courante. Dans ce cas, on parle d'**Algorithme Génétique Adaptatif (AGA)**.

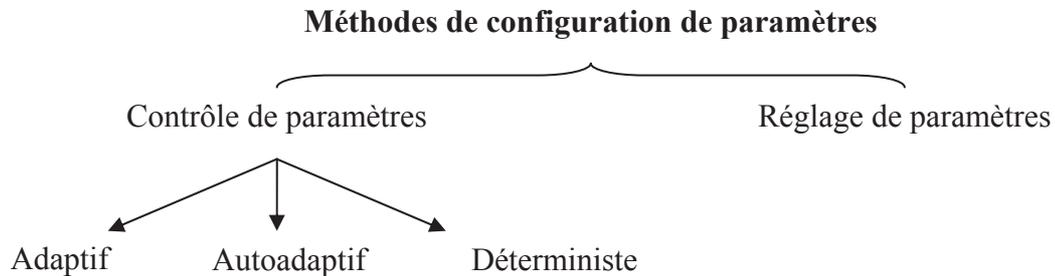


FIG. 4.1 – Les méthodes utilisées pour la configuration de paramètres d'AG

4.2.1 Réglage de paramètres

Dans cette méthode, les auteurs configurent les paramètres de l'algorithme génétique soit en utilisant leurs expériences passées dans ce domaine, soit de manière expérimentale.

Actuellement, le problème de réglage des paramètres des algorithmes génétiques est toujours l'un des secteurs ouverts les plus importants de la recherche relative aux algorithmes génétiques. Les premiers travaux de recherche systématique de la performance d'un AG pour l'optimisation de fonctions, en employant diverses combinaisons des paramètres, ont été entrepris par [DeJong 1975]. [Grefenstette 1986] a formulé le problème du choix des taux de croisement et de mutation comme un problème d'optimisation. Il a recommandé l'utilisation d'un deuxième niveau d'AG pour déterminer une bonne combinaison de ces paramètres. L'inconvénient de sa méthode est le temps de calcul qui peut être assez long. [Friesleben et al. 1993], [Pham 1994]

ont proposé une technique pour le choix des valeurs de ces paramètres en mettant en concurrence plusieurs sous populations qui emploient chacune différentes valeurs pour ces paramètres.

Enfin, [DeJong 1975] a construit un environnement d'essai contenant cinq problèmes à minimiser (i.e. benchmark); ceux-ci sont des fonctions mathématiques. Les fonctions suggérées par De Jong ont certaines caractéristiques, qui les rendent difficiles à optimiser. Il a employé deux mesures de performance pour juger la qualité des algorithmes génétiques. D'abord, il a défini la notion de performance "*En ligne*" qui correspond à la moyenne de tous les fitness observés jusqu'au temps courant (i.e. le fitness cumulé jusqu'à la génération courante). Ceci pénalise les algorithmes ayant une moyenne élevée et favorise ceux ayant une moyenne (la moyenne des fitness) réduite. Ensuite, il a défini la notion de performance "*hors ligne*" qui correspond au meilleur coût (le coût désiré) trouvé jusqu'à la génération courante.

[Deb et al. 1999] a étudié les effets de quatre paramètres sur la performance générale d'un AG. En fait, il a fixé le type de croisement, le type de mutation et le type de la sélection, et il a fait varier les valeurs des autres paramètres : la taille de la population, le taux de croisement et de mutation. L'algorithme génétique étudié a été appliqué sur cinq fonctions mathématiques nommées respectivement : *OneMax*, *Himmelblau*, *Four-peaked*, *Massively multimodal*, *Deceptive*. L'objectif est de minimiser le nombre d'évaluations de la fonction objectif. En se basant sur cette étude, il a fait les conclusions suivantes :

1. La taille de la population doit être supérieure ou égale à 5 individus. Généralement, ce paramètre dépend de la fonction objectif à optimiser.
2. L'AG utilisant conjointement les deux opérateurs de croisement et de mutation a montré un comportement meilleur que l'algorithme se basant uniquement sur le croisement ou uniquement sur la mutation.
3. Un aspect important observé lors de cette étude est le suivant : l'AG utilisant l'opérateur de mutation seul exige une population de taille plus petite comparée à celle de l'AG basé sur l'opérateur de croisement.

4.2.2 Contrôle de paramètres

Dans cette méthode, nous distinguons aussi plusieurs méthodes utilisées. En fait, cette catégorie est basée sur deux critères :

1. Le mécanisme de changement de paramètre.
2. Le paramètre concerné par ce mécanisme.

Nous allons expliquer les trois types d'approches existantes en utilisant l'exemple donné par [Eiben et al. 1999]. Soit la fonction suivante $f(\vec{x}) = f(x_1, \dots, x_n)$ à optimiser, où $l_i \leq x_i \leq u_i$, $i = 1, \dots, n$, où, l_i et u_i sont deux valeurs constantes. Considérons que nous utilisons l'opérateur de mutation Gaussienne. Celui-ci a besoin de deux paramètres, la moyenne (normalement égale à zéro) et l'écart type σ (le pas de la mutation). Donc la mutation est appliquée sur la solution $\vec{x} = (x_1, \dots, x_n)$ comme suit : $x'_i = x_i + N(0, \sigma)$, où N est le générateur Gaussien de nombres aléatoires avec une moyenne égale à zéro, et un écart-type σ .

- Le contrôle de paramètre de type **déterministe** : Dans cette méthode, nous n'avons aucune idée sur l'évolution générale de la population. Ici, les paramètres changent de valeurs aléatoirement ou selon une règle définie au cours du temps (i.e. le changement se fait toutes les n générations). Par exemple, si σ dépend de la génération $\sigma(g) = 1 - 0.9 \cdot \frac{g}{G}$, où g est le

numéro de la génération courante et G est le nombre maximal de génération. Ainsi, σ va varier de 1 (la génération $g = 0$) à 0.1 ($g = G$). Dans ce cas, la probabilité de la mutation variera uniformément au cours de l'évolution. L'utilisateur contrôle complètement la valeur de ce paramètre et celle-ci est prédictible.

- Le contrôle de paramètre **adaptatif** : L'idée de cette méthode est de surveiller l'évolution générale de la population (i.e. environnement exploré de l'espace de recherche) et de changer les valeurs des paramètres concernés selon l'état de la population (i.e. analogie avec une boucle de contre réaction). Si par exemple, la population n'évolue plus (cas d'optimum local) il faut augmenter la probabilité de la mutation pour faire sortir cette population de son état vers une autre vallée ou région [Wang et al. 2004], [Ponnambalam et al. 2002].
- Le contrôle de paramètre **auto-adaptatif** : Nous allons expliquer cette méthode par l'exemple donné par [Eiben et al. 1999]. Dans cet exemple, les paramètres de la mutation sont portés par l'individu. Pour cela, la représentation de l'individu est étendue par $x_1, \dots, x_n, \sigma_1, \dots, \sigma_n$. La réalisation de la mutation est faite par le remplacement de chaque variable à optimiser x_i , et de chaque σ_i par les variables x'_i et σ'_i définies comme suit : $\sigma'_i = \sigma_i \cdot e^{N(0, \tau_0)}$ et $x'_i = x_i + N(0, \sigma'_i)$ où τ_0 est un paramètre de la méthode. Nous voyons bien que chaque variable x_i est mutée par une probabilité différente des autres variables, cette méthode est dite *auto-adaptative*.

[Zhu 2003] a proposé une technique pour maintenir la diversité de la population afin d'améliorer les performances de l'algorithme. Cette technique a été utilisée pour la résolution de problème VRPTW. Il a utilisé la fonction adaptative suivante pour calculer la probabilité de croisement et de mutation et ainsi maintenir la diversité de la population à un niveau désiré :

$$p' = p \left[1 + \frac{\xi(GD_t - GD)}{GD} \right]$$

Où, p est la valeur courante de la probabilité de croisement ou de mutation, p' est la valeur de cette probabilité pour la prochaine génération. Par définition, la valeur de la probabilité de croisement ou de mutation est toujours limitée entre (0 et 1) et GD représente la diversité de la population courante. Cette diversité est égale à la moitié de la somme des distances de **Hamming** de tous les chromosomes pris deux à deux de la population courante. GD_t est la diversité désirée, et ξ est une constante de sensibilité. Les taux de croisement et de mutation sont mis à jour à chaque génération et dépendent chacun des valeurs initiales choisies.

Dans leur travaux [Cicirello et al. 2000] incorporent un réseau de neurones avec l'évaluation de la fonction objectif. Ce réseau est formé pour apprendre les interactions complexes des paramètres de l'algorithme génétique et il permet de prévoir la performance de l'algorithme par rapport aux valeurs de ces paramètres. L'inconvénient de cette méthode est qu'elle introduit aussi des paramètres à régler (i.e. les paramètres du réseau de neurones).

[Tongchim et al. 2002] a ainsi utilisé un algorithme génétique adaptatif parallèle pour optimiser les fonctions mathématiques proposées par [DeJong 1975].

Enfin, nous avons remarqué dans la littérature que les chercheurs ont principalement configuré les paramètres de l'algorithme génétique pour des fonctions mathématiques continues à optimiser. Par conséquent, l'algorithme ne prend en général que quelques minutes pour la résolution. Par contre dans notre cas (le problème VRPTW Dynamique), la résolution est plus difficile car elle fait appel à un simulateur événementiel pour simuler le déroulement d'une journée. L'algorithme génétique a besoin de beaucoup de temps pour le résoudre et il n'est guère faisable d'utiliser les

méthodes employant le réglage de paramètres pour choisir la meilleure combinaison de réglage a priori. De plus, concernant les méthodes adaptives de contrôle de paramètres, bien que très attrayantes sur le papier, elles n'en demeurent pas moins difficiles à mettre en œuvre. Beaucoup de tests sont aussi à réaliser pour régler d'autres paramètres (méta-paramètres) comme le pas de changement de la taille de population, de la probabilité de croisement, de mutation ou de l'élitisme, etc.

Pour les raisons évoquées ci-dessus, nous avons donc choisi d'utiliser une technique expérimentale basée sur "*les plans d'expériences*" pour régler "au mieux" (i.e. en moyenne) les paramètres critiques de notre algorithme génétique. Avec cette méthode, nous avons pu régler et "optimiser" les paramètres de l'algorithme génétique en tenant compte des interactions possibles entre ces paramètres. En fait, cette technique a été déjà implémentée par [Coy et al. 2000] pour régler les paramètres de deux heuristiques utilisées pour la résolution du VRP. [Breedam 1996] a essayé de déterminer les facteurs influents sur la performance de l'algorithme génétique et du recuit simulé utilisés pour résoudre le problème VRP. [Xu et al. 1996] a utilisé les plans d'expériences pour régler cinq paramètres de la méthode de la Recherche Tabou utilisée pour la résolution du problème TSP, ces paramètres concernent l'opérateur de la recherche, la mémoire tabou, etc.

Par ailleurs, cette technique a été utilisée dans d'autres domaines. [Robertson et al. 1998] ont implémenté les plans d'expériences fractionnaires pour régler un réseau de neurones pour une application financière. [Parsons et al. 1997] et [Park et al. 1998] ont réglé les paramètres de l'algorithme génétique et du Recuit Simulé pour plusieurs applications.

Nous allons maintenant présenter cette technique et la démarche que nous avons suivie pour le réglage de ces paramètres.

4.3 Les plans d'expériences

4.3.1 Introduction

La méthode des plans d'expériences n'est pas une technique nouvelle. Elle date en fait du début du *XX* siècle. Elle propose une approche expérimentale pour étudier le comportement d'un système considéré comme une **boîte noire** et en déduire un modèle empirique approximatif à défaut d'un modèle formel plus précis. Ils ont été formalisés par le statisticien *Fischer* puis développés et adaptés plus récemment par *Taguchi* au domaine de la qualité. Leur première utilisation concernait le domaine de l'agronomie.

L'objectif de cette partie est de faire une présentation succincte de la méthode des plans d'expérience afin de pouvoir présenter leur mise en œuvre dans le cadre du réglage des paramètres de l'algorithme génétique. De nombreux ouvrages sont consacrés à la théorie et à la pratique des plans d'expériences [Goupy 2003], [Pillet 1997].

Les plans d'expériences permettent d'organiser, de planifier et d'optimiser la réalisation d'expériences dans un domaine quelconque [Dupas et al. 2003]. Ils répondent au besoin de compréhension et de modélisation d'un système considéré comme une boîte noire, comportant des entrées, appelées *facteurs*, notées X_i sur la figure (4.2), et des sorties, appelées *réponses*, notées Y_i . En l'absence de méthode structurée, l'approche empirique et intuitive consiste à analyser le système en ne faisant varier qu'un seul facteur à la fois.

Enfin, nous pouvons résumer les principaux avantages de cette méthode par rapport aux méthodes traditionnelles d'expérimentation :

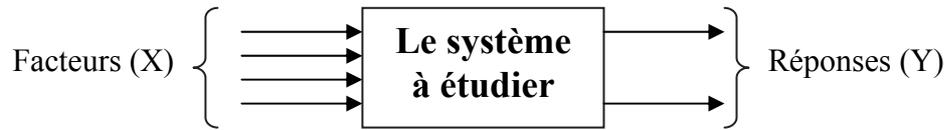


FIG. 4.2 – Système vu comme une boîte noire

1. Les plans permettent une diminution considérable du nombre d'essais et une interprétation rapide et sans équivoque. Ils fournissent des résultats faciles à présenter à des non-spécialistes.
2. Ils permettent d'étudier un très grand nombre de facteurs et de déterminer parmi ceux-ci les facteurs influents.
3. Ils peuvent détecter des interactions éventuelles entre les facteurs.
4. Ils peuvent aussi déterminer des résultats avec une bonne précision.

4.3.2 Terminologie

Avant d'aborder l'étude des plans d'expériences, il convient de préciser le vocabulaire que nous allons utiliser.

- **Facteurs (X)** : dans notre problème, les facteurs sont les paramètres de l'algorithme génétique à régler. Ce sont les entrées du système à étudier.
- **Réponses (Y)** : dans notre cas c'est le fitness de la solution.
- **Niveaux d'un facteur** : les niveaux d'un facteur indiquent les valeurs que prend ce facteur au cours des essais. Ces valeurs sont quantitatives ou qualitatives. Par exemple, le facteur (type de croisement) peut prendre les deux niveaux (PMX , IX), la probabilité de mutation peut avoir trois niveaux (0.05, 0.1, 0.25).
- **L'effet d'un facteur** : l'effet d'un facteur X_i sur une réponse Y , est la variation de Y en fonction des niveaux du facteur. Soit par exemple un facteur X_1 à 2 niveaux notés 1 et 2, l'effet de ce facteur sur une réponse notée Y est représenté en figure (4.3). Nous disons que le facteur est significatif si pour chaque changement de niveau de ce facteur, la valeur de la réponse change. Evidemment, un facteur non significatif sera un facteur qui n'a aucun effet sur cette réponse ou, dont l'effet est trop faible par rapport aux erreurs expérimentales.
- **L'interaction** : Deux facteurs X_1 et X_2 sont en interaction si l'effet de X_1 dépend du niveau de X_2 . A titre d'exemple, l'ajout d'un catalyseur dans une réaction chimique représente une interaction qui engendre un effet additif. La figure (4.4), représente graphiquement une interaction entre deux facteurs X_1 et X_2 à 2 niveaux (1 et 2). Dans le cas (a), les droites représentant les effets sont parallèles, il n'y a donc pas d'interaction entre X_1 et X_2 . Dans le cas (b), ces droites se croisent, les facteurs X_1 et X_2 sont en interaction forte.

4.3.3 Les types des plans

Un plan d'expériences se représente à l'aide d'un tableau ou matrice dont les colonnes représentent les facteurs et les lignes les essais réalisés. La dernière colonne représente la réponse mesurée du système. La table (4.1) représente un plan d'expériences avec 2 facteurs à 2 niveaux, qui comporte 4 essais. Un changement de variables et d'origine a été effectué et fait correspondre

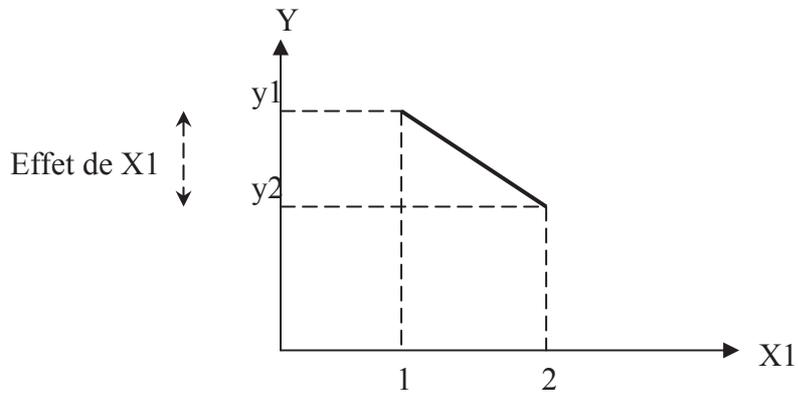


FIG. 4.3 – Effet du facteur X1 sur la réponse Y

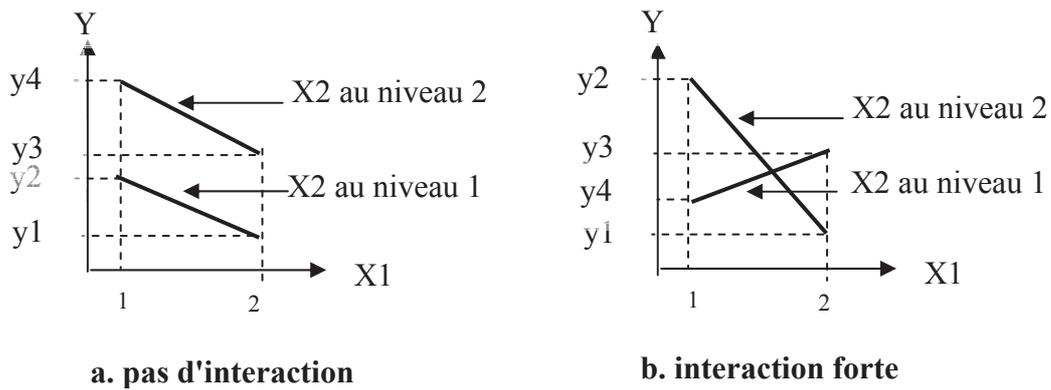


FIG. 4.4 – Interaction entre facteurs

le niveau 1, respectivement 2, du facteur X_1 à la valeur d'origine v_1 , respectivement v_2 ; de même pour le facteur X_2 avec les valeurs v_3 et v_4 . Cette transformation vise à homogénéiser les effets des niveaux des facteurs.

| | Facteur X_1 | Facteur X_2 | Réponse Y |
|----------|---------------|---------------|-------------|
| Essai 1 | 1 | 1 | y_1 |
| Essai 2 | 2 | 1 | y_2 |
| Essai 3 | 1 | 2 | y_3 |
| Essai 4 | 2 | 2 | y_4 |
| | | | |
| Niveau 1 | v_1 | v_3 | |
| Niveau 2 | v_2 | v_4 | |

TAB. 4.1 – Un plan d'expériences complet

Trois types des plans d'expériences sont présentés ci-dessous :

- **Complet** : ils contiennent toutes les combinaisons des facteurs dans tous les niveaux possibles. Le nombre d'essais à réaliser devient rapidement non maîtrisable : il correspond au produit du nombre de niveaux de chaque facteur. Cet inconvénient majeur justifie la théorie de construction des plans fractionnaires.
- **Fractionnaire** : c'est une fraction du plan complet, le nombre d'essais est réduit par rapport au plan complet. En contrepartie, il ne permet d'analyser qu'un sous-ensemble du modèle complet car des effets peuvent être mélangés (notion d'alias).
- **Criblage** : aide à réduire le nombre des niveaux de facteurs, et le nombre de facteurs.

Deux propriétés importantes de ces plans sont énoncées ci-dessous :

- **Orthogonal** : tous les niveaux de chaque facteur sont associés le même nombre de fois à tous les niveaux des autres facteurs. Cette propriété entraîne que chaque effet d'un facteur peut être estimé sans que les autres facteurs ne biaisent cette estimation. La contribution sur la réponse de chaque facteur du plan est indépendante des autres facteurs.
- **Factoriel** : il s'agit d'un plan dans lequel tous les niveaux de tous les facteurs sont représentés.

Un plan complet est nécessairement orthogonal. Cependant un plan d'expériences fractionnaire doit être conçu de manière à respecter la propriété d'orthogonalité pour équilibrer les effets des différents facteurs considérés et permettre ainsi une exploitation non biaisée des expériences réalisées sur le système. Des extensions ont été prévues pour prendre en compte des contraintes de mise en œuvre telles que les essais irréalisables ou dangereux, les erreurs de manipulation (erreurs systématiques).

4.3.4 Démarche générale pour utiliser des plans d'expériences

La mise en œuvre des plans d'expériences nécessite le suivi d'une démarche basée sur les cinq étapes suivantes [Dupas et al. 2003] :

- Etape 1 : *Formaliser le problème* : Cette étape consiste tout d'abord à identifier le problème et les objectifs à atteindre. Les contraintes du système doivent être spécifiées.

- Etape 2 : *Sélectionner les paramètres* : Il s'agit tout d'abord de rechercher l'ensemble des facteurs que l'on suppose influents sur le système, de fixer les niveaux pour chacun de ces facteurs (i.e. modes) et d'identifier les interactions possibles.
- Etape 3 : *Construire le plan* : Les facteurs sont généralement affectés aux colonnes du plan en tenant en compte des interactions et des facteurs difficiles à faire varier pour le système réel. Dans le cas d'un système simulé, l'affectation des facteurs aux colonnes n'est pas une contrainte car il est simple de les faire varier. De plus, le coût d'exécution de ces essais étant moindre, leur nombre peut être plus important. Ceci facilite l'utilisation de plans complets avec de nombreux facteurs. Néanmoins dans certains cas lorsque les temps de simulation sont trop élevés, il convient de limiter le nombre d'essais en recourant à des plans fractionnaires ou en regroupant des facteurs de même nature.
- Etape 4 : *Réaliser les essais* : Effectuer un essai, consiste à affecter aux facteurs du système les niveaux définis dans une ligne du plan d'expériences et à mesurer la réponse du système. Les essais sont réalisés dans l'ordre chronologique. Pour la réalisation de ces essais, il est impératif de mettre en place une organisation qui garantisse que les facteurs non pris en compte dans le plan d'expériences restent à un niveau constant pour l'ensemble des essais. Tout le déroulement de ces essais est généralement consigné dans un journal d'expérimentation. Dans le contexte général de l'utilisation des plans d'expériences sur un système simulé qui correspond au cas de figure que nous traitons, la réalisation des essais est confiée à l'outil informatique qui automatise les exécutions successives et permet donc un plus grand nombre d'essais. Ces essais peuvent aussi être faits sans respecter l'ordre chronologique. Seul, le temps d'exécution du système simulé peut limiter la dimension du plan d'expériences. De plus, la réalisation de ces essais ne nécessite plus de mise en œuvre sur le système réel. Les coûts d'exécution des essais sont donc presque négligeables.
- Etape 5 : *Analyser les résultats* : Cette analyse commence par une phase de dépouillement qui est suivie d'une phase d'interprétation des résultats. Le dépouillement du plan consiste principalement à calculer les coefficients du modèle, établir les graphes des effets, effectuer une étude statistique pour mesurer la variabilité de la réponse. L'interprétation des résultats consiste à : analyser les résultats en fonction des objectifs de départ et sélectionner les niveaux des facteurs qui donnent les résultats optimaux.

4.4 Application des plans d'expériences pour le réglage des paramètres de l'AG

Dans notre cas, le problème à résoudre est le réglage a priori des paramètres de l'algorithme génétique utilisé pour la résolution du problème VRPTW dynamique. L'objectif à atteindre est de trouver la combinaison de ces paramètres qui donne le meilleur comportement pour notre algorithme génétique.

Donc l'algorithme génétique est vu comme une "boite noire" dont les entrées sont les paramètres de l'algorithme. En fait, nous avons choisi parmi l'ensemble des paramètres à régler, six principaux paramètres qui sont :

1. Taille : la taille de la population (X_1).
2. Cr : le type de croisement (X_2).
3. T2Cr : la probabilité de croisement (X_3).

4. Mt : le type de mutation (X_4).
5. T2Mt : la probabilité de mutation (X_5).
6. Elitisme : le taux de l'élitisme (X_6).

En fait, il est prouvé que ces paramètres sont influents sur le comportement des algorithmes génétiques [Jason et al. 2002], [Tavares et al. 2003] et [Deb et al. 1999].

L'opérateur de sélection a été fixé au type "**tournoi**". Nous avons fixé cet opérateur parce que la taille de la population est petite et que la plupart des travaux concernés ont fait ce choix [Deb et al. 1999].

Par ailleurs, pour mesurer les performances de cet algorithme, nous avons dû déterminer les sorties de la boîte noire (i.e. les réponses du système). Pour ce faire, nous avons fixé deux mesures de performances (voir la figure 4.6) qui sont :

1. La distance totale parcourue par les véhicules (Y_1) de la meilleure solution trouvée à la fin de la simulation.
2. Le numéro de la génération (Y_2) correspondant au moment où dans la population, la distance totale parcourue de la meilleure solution est égale à 'K' fois la valeur de la meilleure solution connue (Opt). Dans la pratique le facteur 'K' est expérimentalement fixé à 1.5 ou 1.1 selon les problèmes étudiés (voir la figure 4.5).

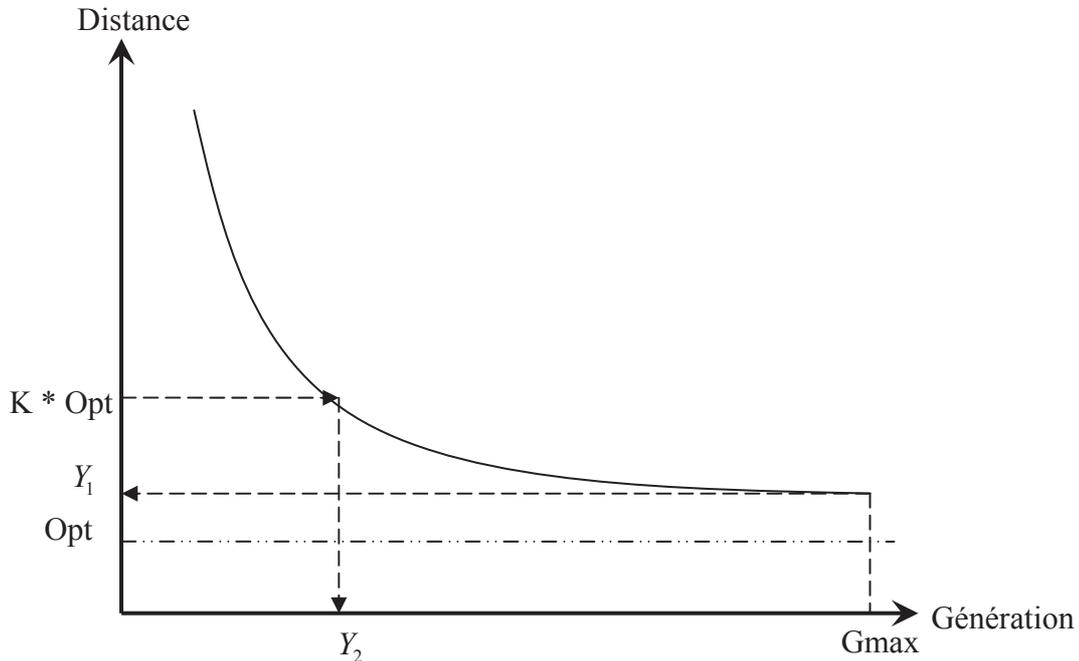


FIG. 4.5 – Les réponses Y_1 et Y_2

Le premier indicateur mesure de manière classique la performance de l'algorithme en fonction de la qualité de la meilleure solution trouvée. Le second indicateur permet d'apprécier la rapidité de convergence de l'algorithme pour un réglage donné des paramètres de l'AG. Ce critère peut être décisif lorsqu'on est plongé dans un contexte dynamique où les contraintes de temps de réponse sont déterminantes.

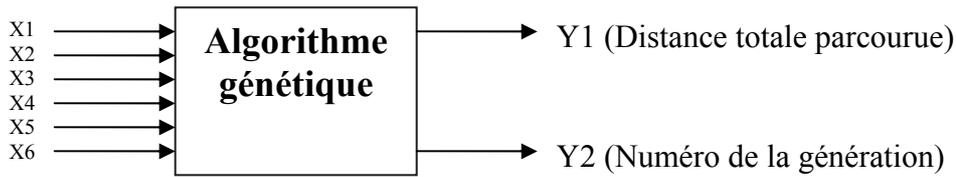


FIG. 4.6 – Le système à analyser

Nous avons utilisé au début de cette étude 3 niveaux pour chaque facteur qui sont donnés dans la table (4.2). En fait, nous avons choisi 3 niveaux parce que les plans d’expériences deviennent vite compliqués si le nombre de facteurs est important et que le nombre de niveaux dépasse 3 [Goupy 1997]. Nous avons choisi ces valeurs à partir de données issues d’ouvrages généraux sur les algorithmes évolutionnistes ou d’articles publiés dans des revues spécialisées portant sur le problème VRPTW [Bräysy et al. 2001], [Blanton et al. 1993].

| Niveaux | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 |
|---------|-------|--------|-------|--------|-------|-------|
| 1 | 50 | PMX | 0.7 | 1-Opt | 0.2 | 50% |
| 2 | 30 | MX1T/D | 0.8 | Or-Opt | 0.05 | 70% |
| 3 | 100 | IX | 0.5 | 2-Opt | 0.5 | 100% |

TAB. 4.2 – Les niveaux des facteurs

Enfin pour exécuter nos plans d’expériences, nous avons utilisé les deux benchmarks suivants :

1. Le benchmark de Solomon [Solomon 1987] qui concerne le problème VRPTW (à 100 clients), ce benchmark contient 56 problèmes statiques (tous les clients sont connus à l’avance).
2. Le benchmark de Solomon modifié par [Gendreau et al. 1999a] : ce benchmark concerne également le problème VRPTW (à 100 clients), mais il a été modifié par [Gendreau et al. 1999a] pour le transformer en un benchmark dynamique.

Maintenant, nous allons présenter la démarche que nous avons suivie pour déterminer la meilleure combinaison des paramètres de l’algorithme génétique.

La première étape que nous avons faite, est d’établir un premier plan d’expériences dit “**plan de criblage**” dont l’objectif était d’une part de vérifier que les facteurs choisis précédemment sont significatifs pour le problème traité ici et d’autre part d’isoler parmi les trois niveaux possibles ceux qui semblent les plus pertinents pour la construction des plans d’expériences suivants. Dans la seconde étape, nous avons mesuré les écarts entre “**plan fractionnaire**” et “**plan complet**” afin de privilégier une approche dans le cas de l’étude de benchmarks dynamiques (troisième étape).

Dans la table (4.3) nous résumons la démarche que nous avons suivie pour atteindre notre objectif.

TAB. 4.3: Le schéma général de la démarche

| Démarche | Problèmes | Plan | Paramètres | Niveaux | Objectif |
|----------|---------------|--------------------------|------------|---------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | 6 statiques | Criblage | 6 | 3 | Découvrir les paramètres influents et réduire le nombre de niveaux |
| 2 | 6 statiques | Complet et fractionnaire | 6 | 2 | Faire une comparaison entre les deux plans complet et fractionnaire pour en privilégier un. |
| 3 | 12 dynamiques | Fractionnaire | 6 | 2 | Découvrir les paramètres influents, trouver la meilleure combinaison pour chaque problème et généraliser cette combinaison sur le groupe ¹ auquel le problème appartient. |

¹ La notion de groupe de problèmes est définie au chapitre 5, elle correspond à un regroupement de problèmes ayant des caractéristiques similaires.

Comme nous le voyons dans ce schéma notre démarche est composée de trois parties. Chaque partie est une étape indépendante. A la fin de ces étapes nous avons pu atteindre notre objectif. Dans les sections suivantes nous présentons notre démarche plus en détail.

4.4.1 Etape 1 - Le plan de criblage

Ce plan consiste à déterminer si, pour chaque facteur envisagé, le passage d'un niveau à un autre provoque une modification significative de la (des) réponse(s) mesurée(s) [Didier et al. 2002].

Nous avons commencé notre démarche par la construction de 6 plans de criblage correspondant aux 6 groupes de problèmes du benchmark de Solomon [Solomon 1987]. A titre indicatif, ces six groupes de problèmes sont notés respectivement : $C1$, $C2$, $R1$, $R2$, $RC1$ et $RC2$. Les problèmes choisis sont $C109$, $C204$, $R101$, $R201$, $RC104$, et $RC201$. Nous avons sélectionné ces problèmes parce qu'ils sont difficiles à résoudre. En effet lors d'une étude préliminaire sur les problèmes statiques, ces instances ont montré une difficulté de résolution⁹ importante.

La table (4.4) montre le plan de criblage ($3^6//16$) [Didier et al. 2002] construit pour résoudre le problème C109. Il contient 6 facteurs à 3 niveaux, nous avons donc besoin d'effectuer 16 essais [Didier et al. 2002]. Chaque ligne de ce plan forme une combinaison différente des paramètres de l'algorithme génétique. Le nombre maximal de génération est égal à 10000. Pour chaque essai (une combinaison), nous avons exécuté notre algorithme génétique cinq fois et nous avons calculé la moyenne de chaque réponse (voir l'annexe A.1 pour les autres plans de criblage).

A partir de ce plan, nous avons pu calculer les effets pour chaque facteur (paramètre) en utilisant l'équation suivante :

$$E_{i,j} = \left(\frac{1}{k_{i,j}} \sum_{l=0}^n (x_{i,j,l} \cdot y_l) \right) - M \quad (4.1)$$

Où :

1. $E_{i,j}$ représente l'effet du facteur i au niveau j ,
2. M est la moyenne générale des réponses,
3. y_l est la réponse correspondante à l'essai l ,
4. n représente le nombre des essais,
5. $k_{i,j}$ est le nombre d'apparitions du facteur i au niveau j dans la colonne du facteur.
6. $x_{i,j,l}$ est égal à 1 si le niveau j du facteur i apparaît dans l'essai l , sinon $x_{i,j,l}$ est égal à zéro.

On peut dire simplement que l'effet d'un facteur i au niveau j ($E_{i,j}$) est l'écart entre la moyenne générale et la moyenne de toutes les réponses issues de l'utilisation du facteur i au niveau j .

Compte tenu du caractère stochastique des algorithmes génétiques, nous avons lancé 5 fois notre AG pour chaque essai. Autrement dit, dans le plan (table 4.4), nous avons effectué au total $16 * 5 = 80$ expériences. Comme nous l'avons dit précédemment, pour mesurer les performances de l'algorithme génétique, on a mesuré à la fin de chaque simulation deux valeurs (réponses Y_1 ,

⁹Le fitness obtenu par l'AG pour chaque problème choisi est loin du meilleur fitness publié.

4.4. Application des plans d'expériences pour le réglage des paramètres de l'AG

| Nb | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | Y ₁ | Y ₂ |
|----------------|---------|----------|---------|---------|---------|---------|----------------|----------------|
| 1 | 50 | MX1T/D | 0,8 | 2-Opt | 0,05 | 100 | 851 | 8 |
| 2 | 100 | PMX | 0,8 | Or-Opt | 0,5 | 70 | 933 | 512 |
| 3 | 100 | IX | 0,7 | Or-Opt | 0,05 | 100 | 1194 | 6990 |
| 4 | 30 | IX | 0,5 | 1-Opt | 0,05 | 70 | 928 | 483 |
| 5 | 100 | MX1T/D | 0,5 | 2-Opt | 0,2 | 70 | 833 | 5 |
| 6 | 30 | IX | 0,8 | 2-Opt | 0,5 | 50 | 1137 | 4609 |
| 7 | 30 | MX1T/D | 0,5 | Or-Opt | 0,5 | 100 | 845 | 11 |
| 8 | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | 829 | 6 |
| 9 | 100 | MX1T/D | 0,8 | 1-Opt | 0,05 | 50 | 837 | 4 |
| 10 | 50 | IX | 0,8 | Or-Opt | 0,2 | 70 | 949 | 647 |
| 11 | 50 | PMX | 0,5 | Or-Opt | 0,05 | 50 | 971 | 1024 |
| 12 | 30 | PMX | 0,7 | 2-Opt | 0,05 | 70 | 1192 | 4585 |
| 13 | 50 | MX1T/D | 0,7 | 1-Opt | 0,5 | 70 | 836 | 5 |
| 14 | 30 | PMX | 0,8 | 1-Opt | 0,2 | 100 | 1019 | 952 |
| 15 | 30 | MX1T/D | 0,7 | Or-Opt | 0,2 | 50 | 829 | 6 |
| 16 | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | 829 | 6 |
| Moyenne | | | | | | | 938 | 1241 |
| Effets | | | | | | | | |
| Y ₁ | 50 | PMX | 0,7 | 1-Opt | 0,2 | 50 | | |
| | -36,69 | 90,21 | 74,51 | -33,19 | -30,79 | 5,21 | | |
| | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | | |
| | 12,79 | -102,06 | -15,24 | -15,81 | 15,66 | -22,09 | | |
| | 100 | IX | 0,5 | 2-Opt | 0,5 | 100 | | |
| | 11,11 | 113,91 | -44,04 | 64,81 | -0,54 | 38,96 | | |
| Y ₂ | 50 | PMX | 0,7 | 1-Opt | 0,2 | 50 | | |
| | -819,85 | 527,40 | 1655,70 | -879,65 | -838,10 | 170,15 | | |
| | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | | |
| | 91,47 | -1234,35 | -397,85 | -90,68 | 397,25 | -459,78 | | |
| | 100 | IX | 0,5 | 2-Opt | 0,5 | 100 | | |
| | 636,90 | 1941,30 | -860,00 | 1061,00 | 43,60 | 749,40 | | |

TAB. 4.4 – Plan de criblage du problème C109 statique

Y_2); Y_1 est la distance totale parcourue par les véhicules de la meilleure solution trouvée et Y_2 est le numéro de la génération contenant comme meilleure distance totale un certain pourcentage ($K = 1.5$ pour le problème C109) de la solution optimale du problème traité.

A partir des calculs approximatifs des effets de chaque facteur pour chaque niveau, nous avons obtenu les graphiques suivants pour chaque réponse (les figures 4.7 et 4.8).

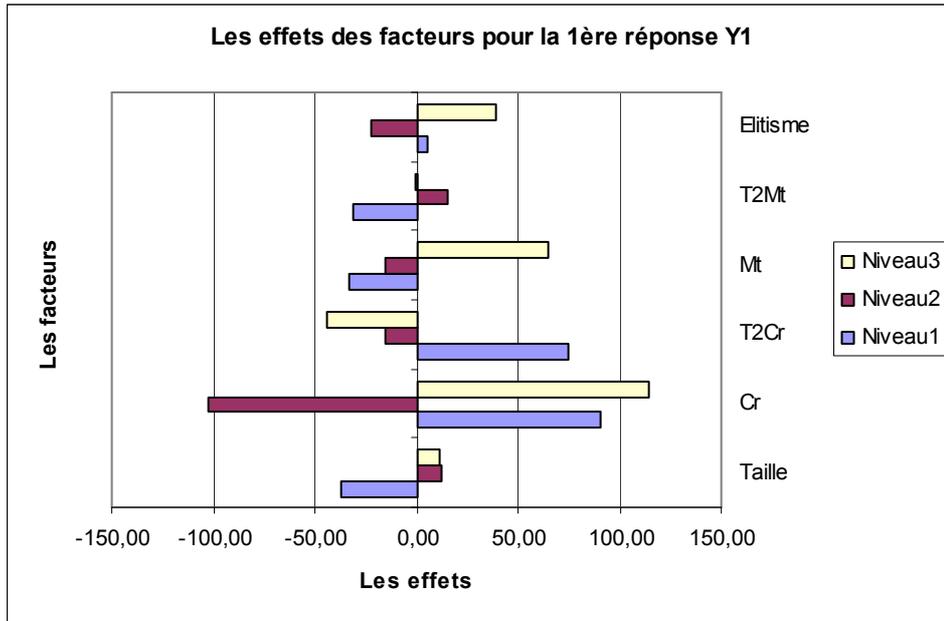


FIG. 4.7 – Effets des facteurs pour la première réponse Y1 (problème C109)

A partir des figures (4.7 et 4.8) nous pouvons déduire que :

- Pour les deux types de réponses, nous avons obtenu sensiblement les mêmes formes des effets sauf que les valeurs des effets des facteurs entre les deux figures sont différentes.
- Le type de croisement (Cr) est plus influent sur les performances de l'algorithme génétique que les autres paramètres car ses effets sur la réponse sont les plus dispersés. Pour les deux réponses, l'opérateur de croisement au niveau 2 (MX1T/D) et au niveau 1 (PMX) fournissent les meilleurs résultats.
- Le deuxième facteur influent est la probabilité de croisement (T2Cr) dont les effets sont également très dispersés.
- Les quatre facteurs restant sont : la taille de la population, le type de mutation, la probabilité de la mutation et le taux d'élitisme ; ceux-ci ont presque tous les mêmes effets sur les performances de l'AG.

Dans la table (4.5), nous donnons les résultats des plans de criblage pour tous les problèmes traités. Dans cette table, chaque problème a 3 niveaux pour chaque facteur ; ces niveaux sont classés selon leurs effets décroissants sur le comportement de l'algorithme génétique.

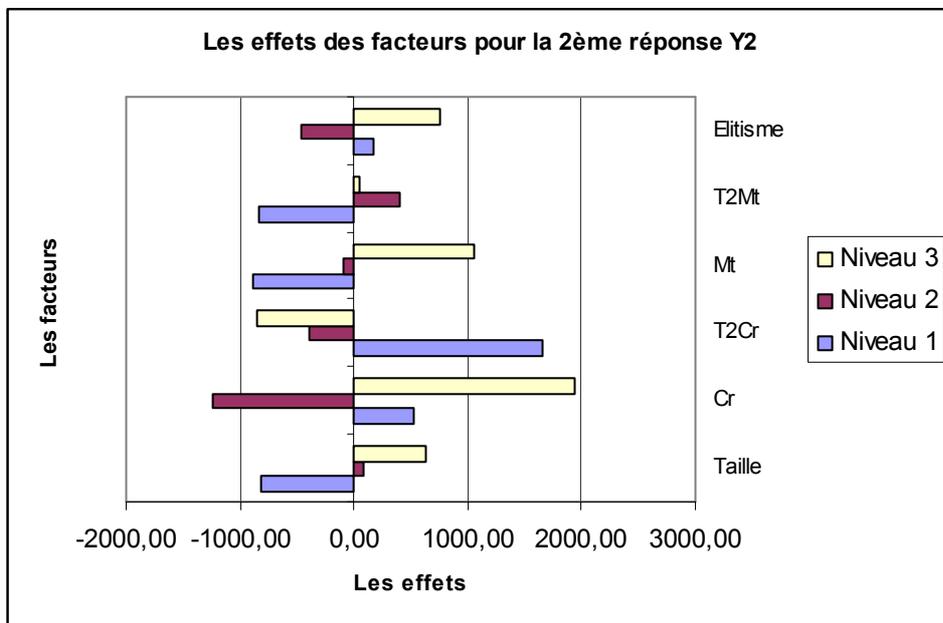


FIG. 4.8 – Effets des facteurs pour la deuxième réponse Y2 (problème C109)

| | C109 | | R101 | | RC104 | | C204 | | R201 | | RC201 | |
|----|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| | Y1 | Y2 |
| X1 | 50 | 50 | 50 | 50 | 30 | 50 | 50 | 50 | 30 | 50 | 50 | 50 |
| | 100 | 30 | 100 | 30 | 50 | 100 | 30 | 30 | 50 | 30 | 30 | 100 |
| | 30 | 100 | 30 | 100 | 100 | 30 | 100 | 100 | 100 | 100 | 100 | 30 |
| X2 | MX1T/D | MX1T/D | MX1T/D | MX1T/D | MX1T/D | MX1T/D | PMX | MX1T/D | MX1T/D | MX1T/D | MX1T/D | MX1T/D |
| | PMX | PMX | PMX | IX | PMX | IX | MX1T/D | PMX | PMX | PMX | PMX | IX |
| | IX | IX | IX | PMX | IX | PMX | IX | IX | IX | IX | IX | PMX |
| X3 | 0,5 | 0,5 | 0,5 | 0,5 | 0,8 | 0,5 | 0,8 | 0,8 | 0,5 | 0,5 | 0,5 | 0,5 |
| | 0,8 | 0,8 | 0,8 | 0,8 | 0,5 | 0,7 | 0,5 | 0,5 | 0,8 | 0,8 | 0,8 | 0,8 |
| | 0,7 | 0,7 | 0,7 | 0,7 | 0,7 | 0,8 | 0,7 | 0,7 | 0,7 | 0,7 | 0,7 | 0,7 |
| X4 | 1-Opt | 1-Opt | 1-Opt | 2-Opt | 1-Opt | 2-Opt | Or-Opt | Or-Opt | 1-Opt | 2-Opt | Or-Opt | Or-Opt |
| | Or-Opt | Or-Opt | Or-Opt | Or-Opt | Or-Opt | Or-Opt | 1-Opt | 1-Opt | Or-Opt | Or-Opt | 1-Opt | 2-Opt |
| | 2-Opt | 2-Opt | 2-Opt | 1-Opt | 2-Opt | 1-Opt | 2-Opt | 2-Opt | 2-Opt | 1-Opt | 2-Opt | 1-Opt |
| X5 | 0,2 | 0,2 | 0,2 | 0,5 | 0,2 | 0,5 | 0,2 | 0,2 | 0,2 | 0,5 | 0,2 | 0,5 |
| | 0,5 | 0,5 | 0,5 | 0,05 | 0,5 | 0,05 | 0,5 | 0,5 | 0,5 | 0,05 | 0,5 | 0,05 |
| | 0,05 | 0,05 | 0,05 | 0,2 | 0,05 | 0,2 | 0,05 | 0,05 | 0,05 | 0,2 | 0,05 | 0,2 |
| X6 | 70 | 70 | 50 | 70 | 50 | 50 | 70 | 70 | 70 | 70 | 70 | 70 |
| | 50 | 50 | 70 | 50 | 70 | 70 | 50 | 100 | 50 | 50 | 50 | 50 |
| | 100 | 100 | 100 | 100 | 100 | 100 | 100 | 50 | 100 | 100 | 100 | 100 |

Tab. 4.5: Plans de criblage des problèmes statiques : classement des niveaux par effets décroissants

A la fin de cette étape nous pouvons réduire le nombre de niveaux des facteurs à deux. En analysant l'ensemble des résultats (la table 4.5), nous avons éliminé les niveaux qui diminuent les performances de l'AG (les niveaux qui ont d'effets négatifs sur la performance de l'AG). Pour la suite de nos travaux, la table (4.2) réduite devient la table (4.6). En réalité, l'utilisation des plans de criblage n'a pas toujours permis de décider quel niveau du facteur X_3 (probabilité de croisement) pouvait être éliminé, c'est pourquoi nous avons sélectionné arbitrairement deux valeurs (petite et grande) pour ce facteur.

| Niveau | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 |
|--------|-------|--------|-------|--------|-------|-------|
| 1 | 50 | PMX | 0.5 | 1-Opt | 0.2 | 50% |
| 2 | 30 | MX1T/D | 0.7 | Or-Opt | 0.5 | 70% |

TAB. 4.6: Les niveaux des facteurs après l'élimination d'un niveau

Dans la deuxième étape, nous avons effectué 6 plans factoriels complets et 6 plans fractionnaires pour les 6 problèmes précédemment choisis pour représenter les différents groupes du benchmark de Solomon (VRPTW statique à 100 clients).

Le temps de réalisation d'un plan complet étant important, il peut être intéressant d'utiliser des plans fractionnaires si la qualité des résultats reste satisfaisante. Pour cela, nous avons comparé les résultats obtenus à partir d'un plan complet avec ceux obtenus à partir d'un plan fractionnaire sur un ensemble de problèmes statiques de Solomon [Solomon87].

4.4.2 Etape 2 - Plans d'expériences complets vs. fractionnaires

4.4.2.1 Le plan factoriel complet

La table (4.7) présente un plan complet du problème C109. Dans cette table nous voyons que les niveaux sont notés (-1, +1), cette notation a été utilisée par Tagushi [Pillet 1997] pour analyser les plans d'expériences ayant deux niveaux.

Cette table est divisée en 4 parties :

1. La première partie est consacrée aux essais, dans ce plan nous avons 64 essais.
2. La deuxième partie rappelle les niveaux utilisés pour chaque facteur.
3. La troisième partie concerne les calculs des moyennes et les écart-types.
4. La quatrième partie est réservée aux effets et aux meilleures combinaisons.

Chapitre 4. Le réglage des paramètres de l'algorithme génétique

Partie I. (Les essais)

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Blit. | I ₁₂ | I ₁₃ | I ₁₄ | I ₁₅ | I ₁₆ | I ₂₃ | I ₂₄ | I ₂₅ | I ₂₆ | I ₃₄ | I ₃₅ | I ₃₆ | I ₄₅ | I ₄₆ | I ₅₆ | Y ₁ D | Y ₂ G | σ ₁ Y ₁ | σ ₂ Y ₂ |
|----|--------------------------|----------------------|------------------------|----------------------|------------------------|-------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|---------------------|---------------------|----------------------------------|----------------------------------|
| | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 835 | 8 | 5 | 3 |
| 2 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 835 | 12 | 7 | 10 |
| 3 | -1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 936 | 793 | 99 | 1045 |
| 4 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 927 | 248 | 84 | 156 |
| 5 | -1 | -1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 836 | 11 | 5 | 10 |
| 6 | 1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 836 | 5 | 8 | 1 |
| 7 | -1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1009 | 490 | 64 | 414 |
| 8 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 921 | 256 | 67 | 156 |
| 9 | -1 | -1 | -1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 830 | 8 | 2 | 2 |
| 10 | 1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 829 | 8 | 0 | 3 |
| 11 | -1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 945 | 1236 | 101 | 1373 |
| 12 | 1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 899 | 685 | 46 | 344 |
| 13 | -1 | -1 | 1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 829 | 7 | 0 | 2 |
| 14 | 1 | -1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 829 | 6 | 0 | 1 |
| 15 | -1 | 1 | 1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 962 | 1203 | 73 | 1284 |
| 16 | 1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 945 | 432 | 65 | 235 |
| 17 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 840 | 10 | 18 | 3 |
| 18 | 1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 831 | 7 | 2 | 2 |
| 19 | -1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 907 | 443 | 116 | 217 |
| 20 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 987 | 1154 | 108 | 1749 |
| 21 | -1 | -1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 832 | 6 | 3 | 1 |
| 22 | 1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 834 | 5 | 5 | 1 |
| 23 | -1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1024 | 2089 | 104 | 2316 |
| 24 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 915 | 339 | 77 | 249 |
| 25 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 829 | 8 | 0 | 2 |
| 26 | 1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 829 | 6 | 0 | 1 |
| 27 | -1 | 1 | -1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1022 | 1088 | 97 | 724 |
| 28 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 902 | 749 | 127 | 359 |
| 29 | -1 | -1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 830 | 6 | 2 | 2 |
| 30 | 1 | -1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 829 | 5 | 0 | 1 |
| 31 | -1 | 1 | 1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 977 | 613 | 47 | 494 |
| 32 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 949 | 480 | 90 | 286 |
| 33 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 838 | 7 | 6 | 1 |
| 34 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 830 | 6 | 1 | 1 |
| 35 | -1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1041 | 2427 | 89 | 3229 |
| 36 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 966 | 1121 | 93 | 812 |
| 37 | -1 | -1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 838 | 6 | 4 | 1 |
| 38 | 1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 835 | 5 | 7 | 0 |
| 39 | -1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 949 | 522 | 96 | 266 |
| 40 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1002 | 323 | 107 | 198 |
| 41 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 829 | 6 | 0 | 1 |
| 42 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 829 | 6 | 0 | 1 |
| 43 | -1 | 1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1015 | 1818 | 64 | 837 |
| 44 | 1 | -1 | -1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 944 | 703 | 92 | 541 |
| 45 | -1 | -1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 830 | 6 | 2 | 1 |
| 46 | 1 | -1 | -1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 830 | 5 | 2 | 1 |
| 47 | -1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 932 | 776 | 89 | 172 |
| 48 | 1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 914 | 1083 | 79 | 593 |
| 49 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 832 | 6 | 4 | 1 |
| 50 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 833 | 6 | 3 | 1 |
| 51 | -1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 904 | 396 | 37 | 215 |
| 52 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 972 | 1478 | 167 | 1409 |
| 53 | -1 | -1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 833 | 6 | 3 | 1 |
| 54 | 1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 997 | 1267 | 70 | 1327 |
| 55 | -1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 940 | 737 | 87 | 653 |
| 56 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 830 | 7 | 1 | 1 |
| 57 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 830 | 7 | 1 | 1 |

A suivre ⇒

4.4. Application des plans d'expériences pour le réglage des paramètres de l'AG

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Eflit. | I ₁₂ | I ₁₃ | I ₁₄ | I ₁₅ | I ₁₆ | I ₂₃ | I ₂₄ | I ₂₅ | I ₂₆ | I ₃₄ | I ₃₅ | I ₃₆ | I ₄₅ | I ₄₆ | I ₅₆ | Y ₁ | | Y ₂ | | σ ₁ Y ₁ | σ ₂ Y ₂ | | | | |
|-------------------------------------------------------|--------------------------|----------------------|------------------------|----------------------|------------------------|--------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|----------------|------|----------------|------|----------------------------------|----------------------------------|--|--|--|--|
| | | | | | | | | | | | | | | | | | | | | | | D | G | D | G | | | | | | |
| 58 | 1 | -1 | -1 | 1 | 1 | 1 | -1 | -1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 829 | 6 | 0 | 0 | | | | | | |
| 59 | -1 | 1 | -1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 946 | 1815 | 65 | 1512 | | | | | | |
| 60 | 1 | 1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 942 | 1696 | 132 | 1421 | | | | | | |
| 61 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 830 | 6 | 1 | 0 | | | | | | |
| 62 | 1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 829 | 5 | 0 | 1 | | | | | | |
| 63 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1012 | 1282 | 88 | 983 | | | | | | |
| 64 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 901 | 930 | 91 | 704 | | | | | | |
| Partie 2. (Les niveaux) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| N. | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | | | | | | | | | | | | | | | | | | | | | | | | |
| | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | | | | | | | | | | | | | | | | | | | | | | | | |
| Partie 3. (Les moyennes et les écart-types) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| M. | | | | | | | | | | | | | | | | | | | | | | | 894 | 483 | 45 | 411 | | | | | |
| σ _{effct} | | | | | | | | | | | | | | | | | | | | | | | | 6 | 6 | 51 | | | | | |
| Partie 4. (Les effets et les meilleures combinaisons) | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| E. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Y ₁ | -9 | 62 | 2 | -4 | -0 | 2 | -8 | -3 | -4 | -0 | 2 | 1 | -2 | 0 | 2 | -2 | 3 | -4 | 3 | -2 | -5 | | | | | | | | | | |
| Y ₂ | -92 | 476 | -79 | 39 | 38 | 95 | -91 | -23 | -5 | 47 | 22 | -78 | 39 | 38 | 95 | -15 | 45 | -63 | -16 | 18 | -12 | | | | | | | | | | |
| C. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Y ₁ | 50 | MX1 | 0,5 | OrOpt | 0,5 | 50% | | | | | | | | | | | | | | | | | | | | | | | | | |
| Y ₂ | 50 | MX1 | 0,7 | 1Opt | 0,2 | 50% | | | | | | | | | | | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | | | | | | | | | 829 | 5 | | | | | | |

Таб. 4.7: Plan complet du problème C109 statique

Comme nous l'avons déjà présenté, ce plan comporte $2^6 = 64$ essais (partie 1 de la table) et pour chaque essai nous avons toujours lancé l'algorithme génétique 5 fois. Dans ce plan et après l'achèvement de tous les essais, nous avons calculé :

1. L'écart-type (σ) de chaque réponse pour chaque ligne (voir les deux dernières colonnes). L'écart-type est donné par la formule :

$$\sigma(y) = \sqrt{\frac{1}{m-1} \sum_i (y_i - \bar{y})^2}$$

Où m représente le nombre de réponses pour chaque essai et $\bar{y} = \frac{1}{m} \sum_i^m y_i$ est la moyenne de toutes les réponses effectuées pour un essai. Dans notre cas, nous avons $m = 5$ réponses.

2. La moyenne de toutes les réponses et l'écart type moyen pour l'ensemble des lignes. Ce dernier représentera l'erreur expérimentale (partie 3 de la table).
3. L'écart-type de l'effet $\sigma(E)$ pour les deux réponses (Y_1, Y_2) (partie 3 de la table).
4. Les effets et les interactions des facteurs pour les deux réponses (Y_1, Y_2). A partir de ce plan, nous avons pu calculer les effets (ou les interactions) pour chaque facteur (paramètre) en utilisant l'équation suivante (partie 4 de la table) :

$$E_i = \left(\frac{1}{n} \sum_{j=1}^n (x_{i,j} \cdot y_j) \right) \quad (4.2)$$

Où :

- (a) E_i représente l'effet (ou l'interaction) du facteur i ,
- (b) y_j est la réponse correspondante à l'essai j ,
- (c) n représente le nombre des essais (le nombre de lignes du plan),
- (d) $x_{i,j}$ est le niveau j du facteur i .

Nous allons maintenant aborder un problème qui mérite une attention particulière : quand peut-on dire qu'un effet est significatif ou non ? La méthode consiste à comparer l'erreur $\sigma(E)$ commise à l'effet E lui-même. Trois cas sont possibles [Goupy 1997].

1. L'effet est bien plus grand que l'erreur : $|E| \gg \sigma(E)$. Dans ce cas, la conclusion est aisée, **l'effet est influent**.
2. L'effet est plus petit que l'erreur : $|E| \ll \sigma(E)$. Dans ce cas, la conclusion est, dans la plupart des situations, que **l'effet est sans influence**.
3. L'effet et l'erreur sont du même ordre de grandeur : $|E| \approx \sigma(E)$. Dans ce cas, la conclusion n'est pas toujours facile : **l'effet peut être sans influence ou légèrement influent**. Il est nécessaire de faire appel à son bon sens, à ses connaissances du phénomène et aux tests statistiques pour donner un avis pertinent. Si l'effet ne joue pas un grand rôle dans l'étude et si une mauvaise décision a peu ou pas de conséquences, on pourra négliger celui-ci.

Pour déduire les meilleures combinaisons pour les deux réponses, nous regardons l'effet de chaque niveau d'un paramètre par rapport à la valeur moyenne de la réponse obtenue. Prenons par exemple l'effet du paramètre "Taille de la population" (deuxième colonne dans la table). Pour

la première réponse celui-ci est égal à (-9) . On peut dire que l'effet d'un changement de niveau du paramètre du niveau bas (ici 30) au niveau haut (ici 50) va diminuer la réponse Y_1 d'une valeur moyenne égale à (9) . Donc il est plus intéressant de prendre le niveau haut du paramètre (50) qui minimise la réponse. Par contre, pour l'effet du paramètre "Type de croisement" (troisième colonne de la table) toujours sur la première réponse, nous avons une variation égale à (62) . Alors dans ce cas, nous pouvons dire que l'effet d'un changement de niveau du paramètre du niveau bas (MX1T/D) au niveau haut (PMX) augmentera la valeur de la réponse. Il est donc plus intéressant de garder le niveau bas du paramètre (MX1T/D). La table 4.8 montre les deux combinaisons déduites du plan complet de la table 4.7 en utilisant le principe précédent.

Nous avons aussi déterminé les facteurs influents de ce plan. Ces facteurs sont soulignés dans la table 4.8. Comme il est mentionné ci-dessus, nous avons comparé l'écart-type de l'effet avec les effets de chaque facteur pour déterminer si la variation observée est significative ou non. Celui-ci est influent si son effet est bien plus grand que l'erreur expérimentale de celui-ci (i.e. écart-type de l'effet). Nous avons fait la même démarche pour les interactions d'ordre 2 puisqu'on est en présence d'un plan complet. On a trouvé une interaction influente pour la réponse Y_1 entre la taille de la population et le type de croisement (l'effet de l'interaction $I_{12} \gg$ de l'écart-type de l'effet σ_{effet} du Y_1). De même pour la réponse Y_2 , on a trouvé deux interactions influentes d'une part entre la taille de la population et le type de croisement, et d'autre part entre le taux d'élitisme et le type de croisement (la table 4.8). On note aussi que les deux combinaisons sont relativement similaires pour les deux réponses Y_1 et Y_2 . En général dans cette table nous remarquons que les facteurs influents sur le comportement de l'AG (valeurs soulignées) sont en interaction.

| | Taille | Cr | T2Cr | Mt | T2Mt | Elitisme | Interactions | Réponse |
|-------|-----------|---------------|------|--------|------|----------|-----------------------------|---------|
| Y_1 | <u>50</u> | <u>MX1T/D</u> | 0,5 | Or-Opt | 0,5 | 50% | Taille / Cr | 829 |
| Y_2 | <u>50</u> | <u>MX1T/D</u> | 0,7 | 1-Opt | 0,2 | 50% | Taille / Cr , Elitisme / Cr | 5 |

TAB. 4.8: Les meilleures combinaisons pour le problème C109

4.4.2.2 Le plan fractionnaire

Parallèlement à la construction des plans complets, nous avons construit des plans fractionnaires pour déterminer s'il était possible d'obtenir les mêmes conclusions mais avec beaucoup moins d'essais ou d'expériences à réaliser. En fait, selon les tables standards de Taguchi [Pillet 1997], nous pouvons utiliser la table L_8 pour étudier un problème de 6 facteurs à 2 niveaux ($2^6//8$). Dans la table 4.9, nous avons étudié le problème (C109). On a fait la même analyse que dans le cas d'un plan complet, sauf qu'ici on ne peut pas étudier les interactions qui se trouvent mélangées aux facteurs. On parle alors de **contrastes**. Dans l'annexe A.3, en regardant l'expression d'un contraste, nous avons montré que la valeur d'un contraste pouvait approximer la valeur de l'effet correspondant. Nous avons effectué une vérification expérimentale pour nous en convaincre.

Dans ce plan, on note que nous avons obtenu deux nouvelles combinaisons légèrement différentes de celles obtenues en utilisant le plan complet. Néanmoins on remarque que les réponses issues d'utilisation de ces combinaisons sont assez comparables. En effet la combinaison issue du plan complet pour Y_1 donne le résultat 829 alors que celle issue du plan fractionnaire donne 831. La différence observée est donc peu significative si on la compare à l'erreur aléatoire observée. De plus, on note que les deux combinaisons de la table 4.10 sont exactement les mêmes pour Y_1 et Y_2 .

| Nb | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | Y_1 | Y_2 | σ_1 | σ_2 |
|------------------|--------|--------|-------|--------|-------|-------|-------|-------|------------|------------|
| | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | D | G | Y_1 | Y_2 |
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 915 | 339 | 77 | 249 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 932 | 776 | 89 | 172 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 830 | 6 | 2 | 2 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 829 | 6 | 0 | 0 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 904 | 396 | 37 | 215 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 835 | 5 | 7 | 0 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 899 | 685 | 46 | 344 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 835 | 8 | 5 | 3 |
| Niveaux | | | | | | | | | | |
| -1 | 30 | MX1T/D | 0,5 | 1-Opt | 0,2 | 50% | | | | |
| 1 | 50 | PMX | 0,7 | Or-Opt | 0,5 | 70% | | | | |
| Moyenne | | | | | | | 872 | 278 | 33 | 123 |
| σ_{effet} | | | | | | | | | 12 | 44 |
| Contrastes | | | | | | | | | | |
| Y_1 | -3 | 40 | 6 | 0 | -3 | 3 | | | | |
| Y_2 | -19 | 271 | 4 | 90 | -91 | 18 | | | | |
| Meilleures | | | | | | | | | | |
| Y_1 | 50 | MX1T/D | 0,5 | 1-Opt | 0,5 | 50% | 831 | | | |
| Y_2 | 50 | MX1T/D | 0,5 | 1-Opt | 0,5 | 50% | | 7 | | |

TAB. 4.9 – Plan fractionnaire du problème C109 statique

| | Taille | Cr | T2Cr | Mt | T2Mt | Elitisme | Réponse |
|-------|--------|--------|------|-------|------|----------|---------|
| Y_1 | 50 | MX1T/D | 0.5 | 1-Opt | 0.5 | 50% | 831 |
| Y_2 | 50 | MX1T/D | 0.5 | 1-Opt | 0,5 | 50% | 7 |

TAB. 4.10 – Les meilleures combinaisons pour le problème C109 en utilisant un plan fractionnaire

En fait, ce n'est pas le cas pour les autres problèmes traités (voir l'annexe A.2 pour tous les plans complets et fractionnaires), c'est-à-dire que nous n'avons pas toujours obtenu les mêmes combinaisons de réponses pour chacun des deux plans (complet et fractionnaire) d'un problème traité (voir les tables 4.8 et 4.10).

C'est pourquoi nous avons étudié le taux de réussite pour les six problèmes qui est reportée dans la table 4.11. PC ($Y_1 - Y_2$) est le taux de ressemblance entre la première réponse Y_1 et la deuxième Y_2 dans le plan complet. De même PF ($Y_1 - Y_2$) est le taux de ressemblance dans le plan fractionnaire, PC (Y_1) - PF (Y_1) est le taux de ressemblance entre la réponse Y_1 issue du plan complet et celle issue du plan fractionnaire. Le taux de ressemblance est défini comme le rapport de nombre de facteurs ayant un réglage identique sur le nombre de facteurs étudiés. Par exemple, le taux de ressemblance entre les deux combinaisons dans la table (4.8) représente le rapport $\frac{3 \text{ (le nombre des facteurs identiques)}}{6 \text{ (le nombre total des facteurs)}}$ qui donne le taux de 50% de ressemblance entre ces deux combinaisons. La ligne "Tous" est la moyenne d'une colonne pour tous les problèmes. Celle-ci varie de 55% à 75%. Donc en regardant les colonnes (2 et 3) nous déduisons que nous pouvons éliminer Y_2 , et en regardant les colonnes (4 et 5) nous pouvons déduire qu'on peut utiliser les plans fractionnaires au lieu des plans complets.

| | PC ($Y_1 - Y_2$) | PF ($Y_1 - Y_2$) | PC (Y_1) - PF (Y_1) | PC (Y_2) - PF (Y_2) |
|------|--------------------|--------------------|-----------------------------|-----------------------------|
| C1 | 66,67% | 100,00% | 66,67% | 33,33% |
| R1 | 50,00% | 33,33% | 83,33% | 66,67% |
| RC1 | 33,33% | 33,33% | 100,00% | 100,00% |
| C2 | 66,67% | 50,00% | 66,67% | 50,00% |
| R2 | 66,67% | 83,33% | 83,33% | 66,67% |
| RC2 | 50,00% | 83,33% | 50,00% | 83,33% |
| Tous | 55,56% | 63,89% | 75,00% | 66,67% |

TAB. 4.11: Tableau de comparaison entre plans fractionnaires et plans complets

4.4.2.3 Plan complet vs. plan fractionnaire

Dans la table 4.12, nous avons effectué une comparaison entre l'utilisation des deux plans pour un problème donné. Nous considérons que pour effectuer un "run" il faut 30 minutes environ.

| | Plan | |
|-----------------|----------------------|---------------------|
| | Complet | Fractionnaire |
| Nombre d'essais | $2^6 = 64$ | 8 |
| Nombre de runs | $64 * 5 = 320$ | $8 * 5 = 40$ |
| Temps | $320 * 30 = 9600min$ | $40 * 30 = 1200min$ |

TAB. 4.12: Comparaison entre les deux plans

Dans ce tableau, pour un problème, nous devons faire 64 expériences pour un plan complet parce que nous avons 6 facteurs à 2 niveaux, alors que dans le cas d'un plan fractionnaire proposé par Tagushi [Pillet 1992], nous devons faire 8 expériences. Dans chaque expérience nous utilisons l'algorithme génétique 5 fois afin de tenir compte du comportement stochastique de l'AG. Ainsi,

nous devons lancer l'AG 320 fois pour un plan d'expériences complet, tandis que 40 lancements sont nécessaires pour un plan fractionnaire.

Bien que le plan fractionnaire soit moins exact que le plan complet, l'erreur¹⁰ reste acceptable en comparant celle-ci avec l'erreur expérimentale. Nous avons calculé la variation de la réponse (la distance) du plan complet à celle du plan fractionnaire¹¹, nous avons obtenu la valeur (1.67%). En raison de cette erreur, la réponse sera augmentée de (1.67%) si nous utilisons le plan fractionnaire au lieu d'un complet. Nous avons également calculé le rapport de l'erreur expérimentale moyenne, due aux phénomènes stochastiques, sur la réponse moyenne, nous avons obtenu la valeur (4.63%). Nous pouvons noter que les deux variations sont du même ordre de grandeur. Ainsi nous pouvons favoriser les plans fractionnaires sans grande perte pour l'étape suivante. Avec un plan fractionnaire on gagnera du temps. Le plan complet prend environ 8 fois plus de temps que le plan fractionnaire pour le même problème. De plus, en regardant la table de comparaison des plans fractionnaires et complets (la table 4.11), nous constatons que la moyenne des taux de ressemblance, entre les deux types de plans est égale à 75% pour la première réponse Y_1 , et est approximativement égale à 67% pour la deuxième réponse Y_2 .

Pour toutes les raisons évoquées ci-dessus, nous pouvons finalement choisir d'utiliser le plan fractionnaire au lieu du plan complet comme méthode pour régler les paramètres de l'algorithme génétique dans le cadre de la résolution du problème statique. Avec les deux étapes précédentes basées respectivement sur le plan de criblage et sur la comparaison entre plans complets et fractionnaires, nous avons donc pu déterminer les niveaux des facteurs à sélectionner et le type idéal de plan à utiliser.

En regardant l'allure générale d'une courbe d'évolution (la figure 4.9 ou l'annexe C) de la réponse Y_1 en fonction du nombre de générations pour le problème dynamique R109, on s'aperçoit que la courbe croît vers la valeur optimale du problème statique au fur et à mesure que les clients dynamiques apparaissent. Cependant la courbe présente de fortes irrégularités à chaque arrivée de client. Le seuil à choisir pour Y_2 est plus difficile à déterminer car il est peut être franchi plusieurs fois. Dans la suite de notre étude, nous avons conservé uniquement la réponse Y_1 . De plus, le nombre de clients traités dans les benches (100 clients) n'est pas critique sur le temps de réponse de l'AG.

4.4.3 Etape 3 - Plans d'expériences fractionnaires pour les problèmes dynamiques

La dernière étape dans notre démarche, objectif initial de ce chapitre, est de trouver les meilleures combinaisons des paramètres de l'algorithme génétique pour résoudre le problème VRPTW dynamique. Pour cela nous avons choisi dans chaque groupe de benchmarks dynamiques, deux problèmes ayant des degrés de dynamisme différents (inférieur à 30% et supérieur à 30%), ce qui donne 12 problèmes à traiter. Les degrés de dynamisme des problèmes choisis sont mentionnés en colonne 2 de la table 4.13. Puis nous avons construit des plans fractionnaires pour chacun de ces problèmes. En utilisant le même type d'analyse que nous avons effectué pour le problème statique, nous présentons sur la table 4.13 les combinaisons résultant de ces plans. Les valeurs soulignées indiquent les valeurs influentes sur les performances générales de l'algorithme (voir l'annexe A.4 pour ces plans fractionnaires).

¹⁰La variation de la réponse du plan complet à celle du plan fractionnaire

¹¹Cette valeur est la moyenne des variations des 6 problèmes statiques : la variation pour un problème est égale à $\frac{Y'_{1,PC} - Y'_{1,PF}}{Y'_{1,PC}}$, où $Y'_{1,PC}$ est la meilleure réponse Y_1 obtenue dans le plan complet et $Y'_{1,PF}$ est la meilleure réponse Y_1 obtenue dans le plan fractionnaire.

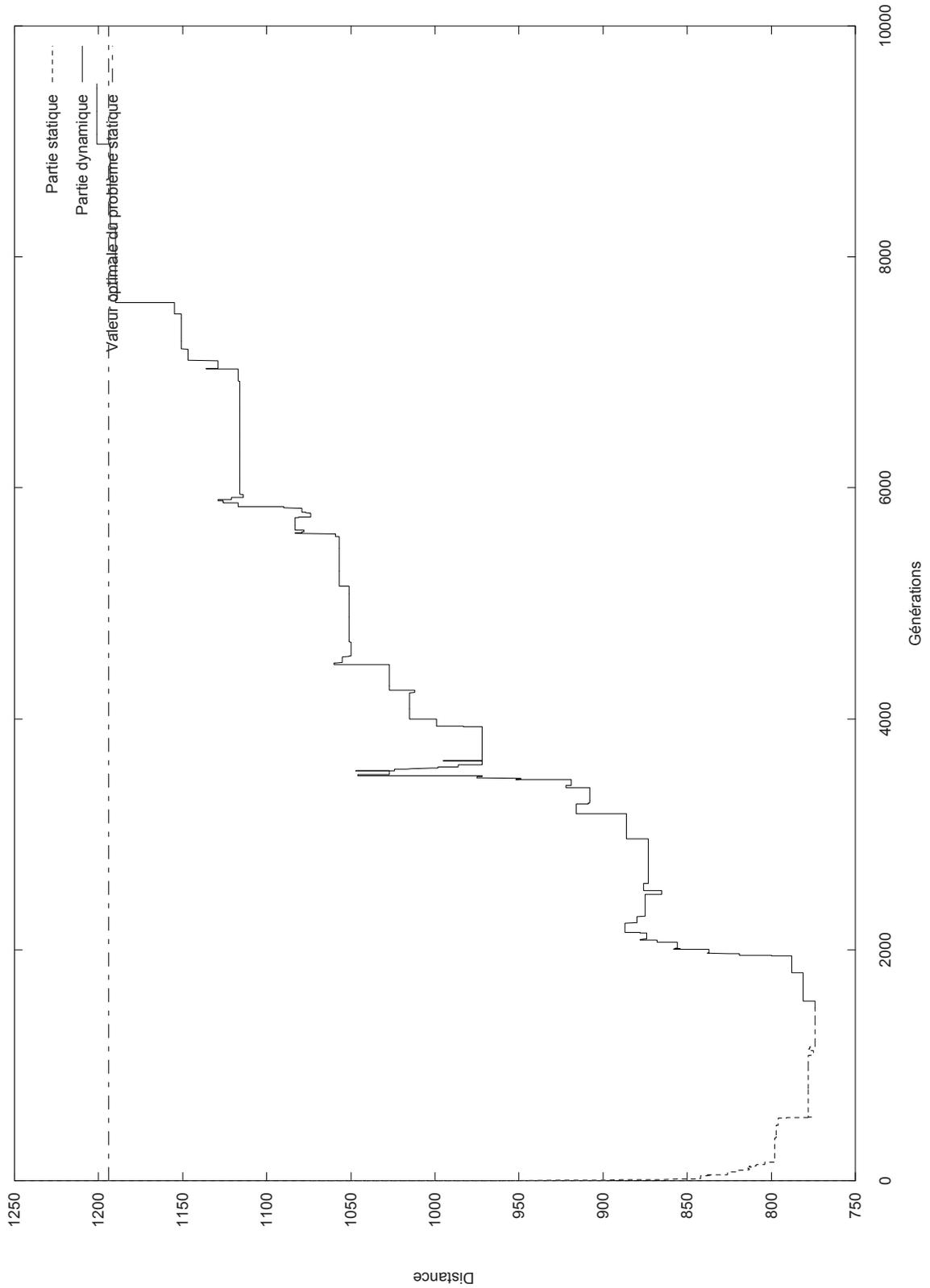


FIG. 4.9 – Courbe d'évolution du Y_1 (Distance) du problème R109 Dynamique

| Problème | | Dyn. (%) | Taille | Cr | T2Cr | Mt | T2Mt | Elitisme |
|----------|-------|----------|-----------|---------------|------|---------------|------------|------------|
| C1 | C104 | 11% | 30 | PMX | 0,5 | Or-Opt | 0,2 | 50% |
| | C108 | 52% | <u>30</u> | <u>MX1T/D</u> | 0,5 | Or-Opt | 0,2 | <u>70%</u> |
| R1 | R104 | 10% | 30 | PMX | 0,5 | 1-Opt | 0,2 | 50% |
| | R109 | 49% | 50 | MX1T/D | 0,7 | Or-Opt | 0,5 | 50% |
| RC1 | RC104 | 10% | 50 | MX1T/D | 0,5 | Or-Opt | 0,5 | 50% |
| | RC106 | 48% | 30 | MX1T/D | 0,7 | Or-Opt | 0,2 | 70% |
| C2 | C204 | 12% | 50 | PMX | 0,5 | Or-Opt | 0,2 | 50% |
| | C207 | 56% | 50 | <u>MX1T/D</u> | 0,7 | <u>Or-Opt</u> | <u>0,2</u> | 50% |
| R2 | R208 | 11% | 50 | PMX | 0,7 | Or-Opt | 0,5 | 70% |
| | R205 | 47% | 50 | MX1T/D | 0,5 | Or-Opt | 0,5 | 50% |
| RC2 | RC204 | 16% | 30 | PMX | 0,7 | Or-Opt | 0,5 | 50% |
| | RC201 | 57% | 50 | <u>MX1T/D</u> | 0,7 | 1-Opt | 0,5 | 70% |

TAB. 4.13: Les meilleurs réglages de chaque groupe des benchmarks dynamiques

A partir de cette table, le choix du paramétrage de l'AG est fait selon le type du problème traité (la dispersion géographique des clients) et le degré de dynamisme du problème ($< 30\%$ ou $\geq 30\%$). Par exemple, pour résoudre un problème de type C1 ayant le degré de dynamisme 50%, nous réglons les paramètres de l'AG en utilisant la deuxième combinaison du C1 dans la table (la combinaison du C108).

Finalement, nous pouvons déduire que :

1. Pour chaque problème à traiter, il existe une combinaison de paramètres qui aide l'algorithme génétique à chercher et à trouver plus rapidement une bonne solution.
2. Dans le problème VRPTW dynamique, l'opérateur de croisement (MX1T/D) est très influent sur la plupart de ces problèmes.
3. Pour l'opérateur de mutation, on constate que le type (Or-Opt) est meilleur dans la plupart des combinaisons.

4.5 Conclusion

Dans ce chapitre, nous avons utilisé la technique "des plans d'expériences" pour régler les paramètres de notre algorithme génétique dédié à la résolution du problème DVRPTW. Les paramètres choisis pour cette étude ont été :

- la taille de la population.
- le type de croisement.
- la probabilité de croisement.
- le type de mutation.
- la probabilité de la mutation.
- le taux d'élitisme.

Nous avons traité deux types de plans d'expériences, les plans complets et les plans fractionnaires. Avec ces plans nous avons pu déterminer les combinaisons aidant l'algorithme génétique à travailler plus efficacement.

Au cours de ce chapitre nous avons découvert l'importance et la fiabilité des plans fractionnaires qui nous ont aidés à économiser du temps en trouvant rapidement une bonne configuration de l'algorithme génétique pour la résolution du problème posé.

Chapitre 5

Plate-forme et résultats

Sommaire

| | | |
|------------|------------------------------------------------|------------|
| 5.1 | Introduction | 109 |
| 5.2 | La plate-forme développée | 110 |
| 5.3 | Le problème DVRPTW | 113 |
| 5.3.1 | Le benchmark utilisé | 113 |
| 5.3.2 | Les résultats | 116 |
| 5.4 | Le problème DVRP sans fenêtres de temps | 119 |
| 5.4.1 | Le benchmark utilisé | 119 |
| 5.4.2 | Les résultats | 119 |
| 5.5 | Le problème DTRP | 120 |
| 5.5.1 | Le benchmark utilisé | 120 |
| 5.5.2 | Les résultats | 122 |
| 5.6 | Le problème DTSPTW | 124 |
| 5.6.1 | Le benchmark utilisé | 124 |
| 5.6.2 | Les résultats | 124 |
| 5.7 | Le problème DPDPTW | 124 |
| 5.7.1 | Le benchmark utilisé | 125 |
| 5.7.2 | Les résultats | 126 |
| 5.8 | Conclusion | 127 |

5.1 Introduction

Ce chapitre est principalement divisé en deux parties : la première partie est consacrée à l'outil que nous avons développé pour la résolution du problème étudié. Dans la deuxième partie, nous allons présenter les jeux de tests dynamiques (les benchmarks) que nous avons utilisés pour évaluer notre approche. Ces benchmarks sont en effet issus d'autres travaux :

1. Le benchmark de Solomon modifié par [Gendreau et al. 1999a] pour la résolution du problème d'élaboration dynamique de tournées de véhicules avec fenêtres de temps (DVRPTW : Dynamic Vehicle Routing Problem with Time Windows).
2. Le benchmark modifié de Taillard, Chrisofides et Fisher pour la résolution du problème DVRP (sans fenêtres de temps) [Kilby et al. 1998a].

3. Le benchmark de Larsen pour le problème dynamique du réparateur (DTRP : Dynamic Traveling Repairman Problem).
4. Le benchmark de Larsen pour le problème dynamique du voyageur de commerce avec fenêtres de temps (DTSPWTW : Dynamic Traveling Salesman Problem with Time Windows).
5. Le benchmark dynamique de Snezana Mitrovic pour le problème dynamique de ramassage et livraison avec fenêtres de temps (DPDPTW : Dynamic Pick-up and Delivery Problem with Time Windows).

En fait, nous avons utilisé notre outil pour la résolution de chacun des jeux de tests mentionnés ci-dessus (les benchmarks) en respectant la fonction objectif de chaque problème. De plus nous avons utilisé les plans d'expériences pour le paramétrage de l'algorithme génétique pour les problèmes dynamiques DVRPTW et DPDPTW. Nous nous sommes focalisés sur ces problèmes de grande taille pour lesquels le réglage est particulièrement important. Ci-après nous présenterons les résultats de chacun de ces problèmes.

5.2 La plate-forme développée

Nous avons développé une plate-forme d'évaluation (i.e. prototype) en Java ayant la capacité de résoudre le problème VRP dynamique et ses principales extensions. Cette plate-forme utilise le modèle de simulation décrit au chapitre 3. Rappelons que celui-ci utilise un composant algorithme génétique dynamique pour résoudre le type de problème dynamique sélectionné au moyen de la boîte liste “**type**” dans la figure 5.1.

Dans cette plate-forme, nous pouvons distinguer trois panneaux, qui sont :

1. Le panneau “**Problème & Simulation**” : Ce panneau permet de visualiser tout ce qui concerne le problème traité et les résultats obtenus (voir la figure 5.1). L'onglet “**Données**” montre toutes les données du problème à résoudre : les clients, les véhicules et les dépôts (figure 5.2). L'onglet “**Résultats**” visualise la meilleure solution trouvée pendant la simulation jusqu'au temps courant ; cette solution est décrite en mode graphique et en mode texte. Le mode texte est montré dans le sous panneau “**Solution**” dans lequel les tournées sont séparées par un tiré (-). La fenêtre intitulée “**graphe**” permet de visualiser à tout moment cette solution en mode graphique (figure 5.1). L'onglet “**Simulation**” visualise ce qui concerne le déroulement de la simulation du problème dynamique, i.e. l'avancement du temps de la simulation et l'apparition des événements (figure 5.3). Cette figure montre un instant particulier de l'avancement de la simulation pour la résolution du problème C201 dynamique. Sur l'onglet “**Moniteur**” on représente les individus de la population et les meilleures solutions trouvées par rapport à chaque critère : distance totale parcourue, retard total chez les clients, etc. L'onglet “**Parallèle**” est en cours d'élaboration ; cet onglet concerne la parallélisation de notre algorithme génétique dynamique. Il est possible de régler la fonction objectif par rapport au problème traité en positionnant des poids sur chacun des critères que l'on cherche à optimiser à l'aide de la zone intitulée “**Fonction Objectif**”. Dès que la simulation démarre nous ne pouvons plus changer ces poids.
2. Le panneau “**Réglage & Résultats**” (figure 5.1) : Ce panneau permet de régler simplement les différents paramètres de l'algorithme génétique à l'aide des deux zones intitulées “**Chromosomes & Evolution**”. Par ailleurs, ce panneau nous permet aussi de surveiller

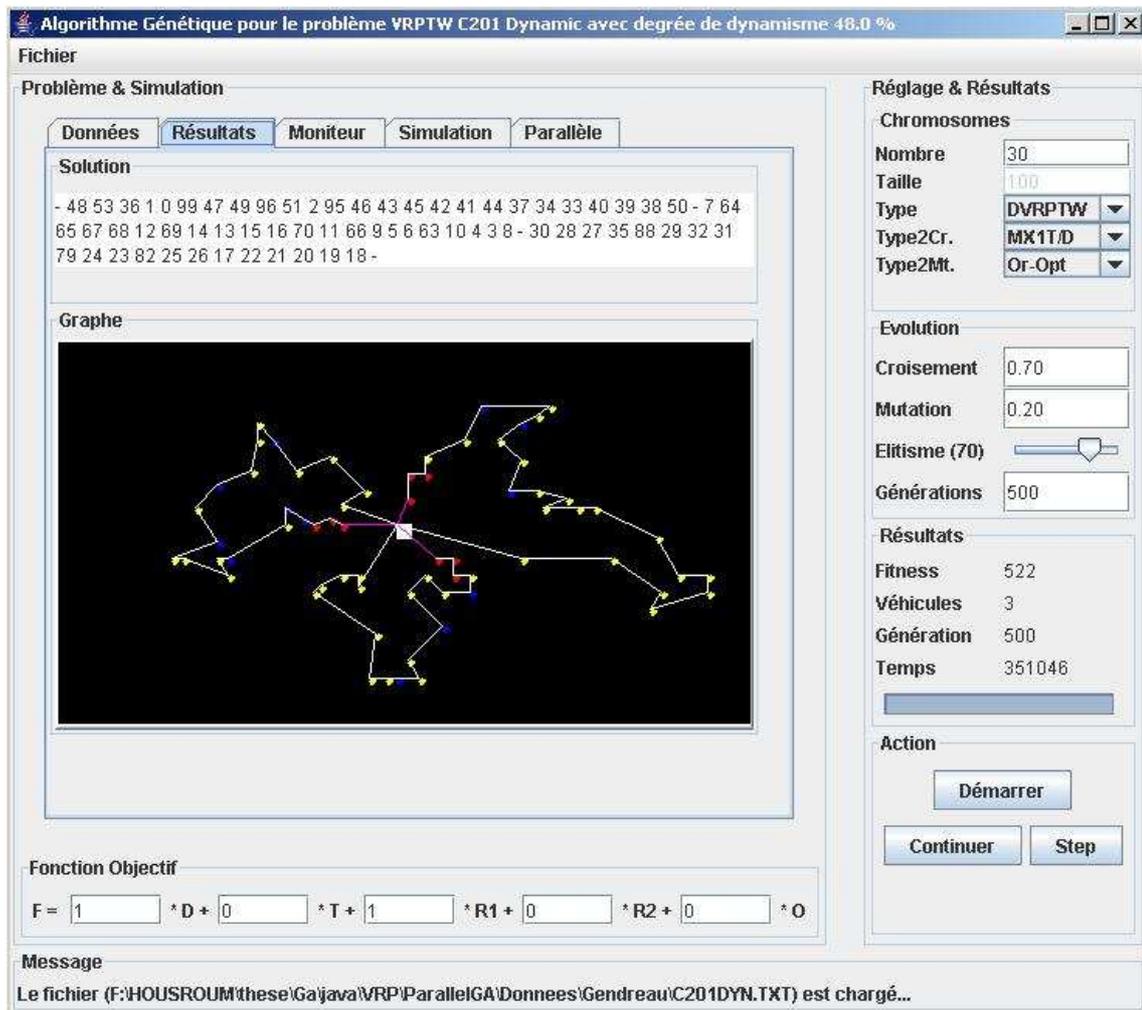


FIG. 5.1 – Plate-forme

le comportement de l’algorithme génétique en regardant l’évolution du fitness à l’aide de la zone intitulée “**Résultat**”. La zone “**Action**” permet de démarrer, d’arrêter la simulation et de passer en mode pas à pas.

3. Le panneau “**Message**” : Ce panneau est dédié aux messages d’erreurs de l’outil, (voir la figure 5.1).

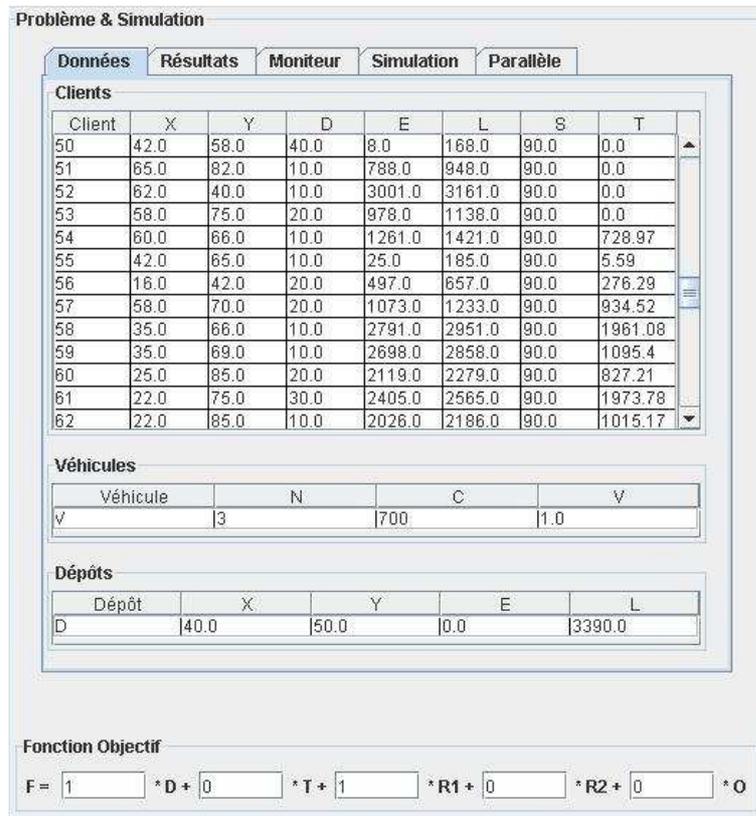


FIG. 5.2 – L’onglet “Données”

Le fichier d’entrée (i.e. benchmark) que nous voulons tester dans la plate-forme, est un fichier de texte ou au format XML qui comporte principalement trois parties :

1. Une partie concernant les données des clients : Cette partie contient tous les attributs de chaque client ; ces attributs sont : la position euclidienne (x, y) , la quantité demandée (d) , la durée de service chez ce client (s) , la borne inférieure de la fenêtre de temps (e) , la borne supérieure de la fenêtre de temps (l) , et le temps d’arrivée de la demande (t) : soit le client est du type statique $(t = 0)$; soit le client est du type dynamique $(t > 0)$ et il demande le service à l’instant t .
2. Une partie concernant les données des véhicules : Cette partie est dédiée aux véhicules disponibles. Normalement, les véhicules pour ce type de problèmes sont identiques et ils ont les mêmes capacités et vitesses ; dans ce cas chaque véhicule est caractérisé par sa capacité (c) et sa vitesse (v) . Le nombre de véhicules disponibles est égal à (N) (voir la figure 5.1).

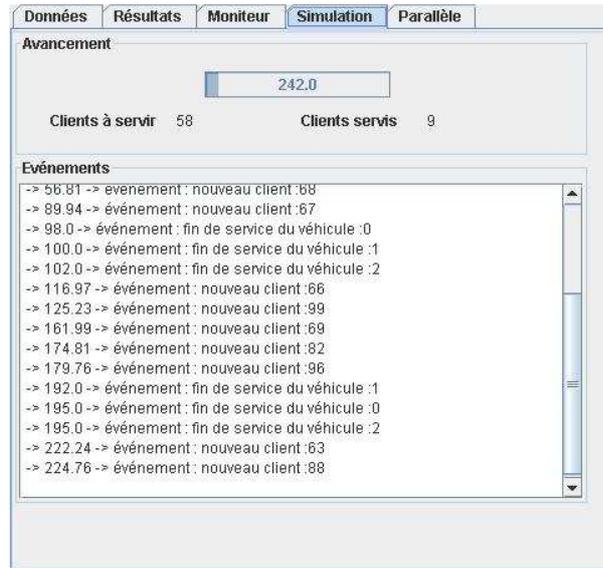


FIG. 5.3 – L’onglet “Simulation”

3. Une partie concernant le dépôt : Cette partie concerne le dépôt. Elle contient les attributs du dépôt, ceux-ci sont la position euclidienne (x, y) , et les deux bornes inférieure et supérieure de la fenêtre de temps (i.e. les horaires d’ouverture et de fermeture du dépôt).

Comme nous l’avons dit, notre outil a la capacité de lire deux types de fichiers de données (i.e. le benchmark dynamique). Ces deux types sont :

1. Fichier texte (voir l’annexe B pour un exemple complet)
2. Fichier XML (voir l’annexe B)

5.3 Le problème DVRPTW

Le but de cette section est de vérifier la performance et la capacité de notre approche évolutionniste pour la résolution du problème DVRPTW en question. A ce stade, nous avons dû comparer notre approche avec les autres approches traitant de ce problème. C’est pourquoi nous avons exploité le benchmark du problème VRPTW statique, ce benchmark a été modifié par M. Gendreau pour qu’il devienne dynamique. Enfin nous avons pu comparer nos résultats avec ceux de [Gendreau et al. 1999a].

5.3.1 Le benchmark utilisé

Nous avons choisi d’utiliser le jeu de problèmes fournis par Solomon [Solomon 1987]. Il présente en effet plusieurs avantages. Le plus important est qu’il constitue implicitement une base de référence pour l’étalonnage des performances de la plupart des méthodes proposées pour le VRPTW.

Cette librairie comprend 56 problèmes au total. Chaque problème contient 100 clients distribués géographiquement dans un carré de 100 sur 100. Le dépôt est situé au centre du carré. On suppose dans notre application que tous les véhicules sont identiques (i.e. flotte homogène)

et que ceux-ci se déplacent à une même vitesse constante. De ce fait, le temps de trajet entre deux clients est directement proportionnel à la distance euclidienne qui sépare ces deux clients. La vitesse des véhicules est supposée égale à 1.0 (unité de distance/unité de temps). La capacité des véhicules est fixée à une valeur importante pour ne pas être contraignante.

Les problèmes de ce benchmark peuvent être regroupés en 6 catégories en fonction de :

1. la loi de distribution des clients dans l'espace (voir la figure 5.4)
 - (a) le type **R** correspond à une distribution uniforme des clients dans l'espace.
 - (b) le type **C** correspond à une distribution des clients dans des clusters deux à deux disjoints.
 - (c) le type **RC** est une combinaison des deux modes de répartition précédents. Une partie des clients est distribuée uniformément dans le carré tandis que l'autre partie est distribuée dans des regroupements (clusters).
2. la taille de la fenêtre de temps sur chaque client
 - (a) le type 1 met en place une fenêtre de temps étroite sur chaque client. Ceci implique un horizon de service court (la fenêtre de temps sur le dépôt central) ainsi que des tournées comportant peu de clients. La capacité des véhicules est également réduite.
 - (b) le type 2 met en place des fenêtres de temps larges. L'horizon de service étant plus étendu, les tournées comprendront en principe plus de clients. En plus la capacité des véhicules est assez large.

Ainsi, pour le problème *R101*, premier problème de type *R1*, les clients sont uniformément répartis dans l'espace, avec des fenêtres de temps réduites, un horizon de service court et une capacité des véhicules faible. Pour le problème *RC205*, cinquième problème de type *RC2*, les clients sont répartis de façon mixte (uniforme et clusters), avec des fenêtres de temps larges, un horizon de service étendu et une capacité de véhicules importante.

Notons que le temps de service auprès de chaque client est fixé à 10 unités de temps pour les problèmes de type *R* et *RC*. Il est de 90 unités de temps pour les problèmes de type *C*.

Dans ce benchmark, chaque fichier d'un problème contient trois parties de données (voir l'exemple dans l'annexe B) :

1. La partie qui concerne les attributs des clients :
 - (x, y) : est la position de client.
 - d : est la quantité de service demandée par le client.
 - s : est la durée de service chez le client.
 - (e, l) : est la fenêtre de temps de client, dans laquelle il souhaite être servi.
2. La partie qui concerne le dépôt :
 - (x, y) : détermine sa position.
 - (e, l) : sont les horaires d'ouverture et de fermeture.
3. La dernière partie concerne les véhicules disponibles :
 - c : la capacité du véhicule.
 - v : la vitesse du véhicule.

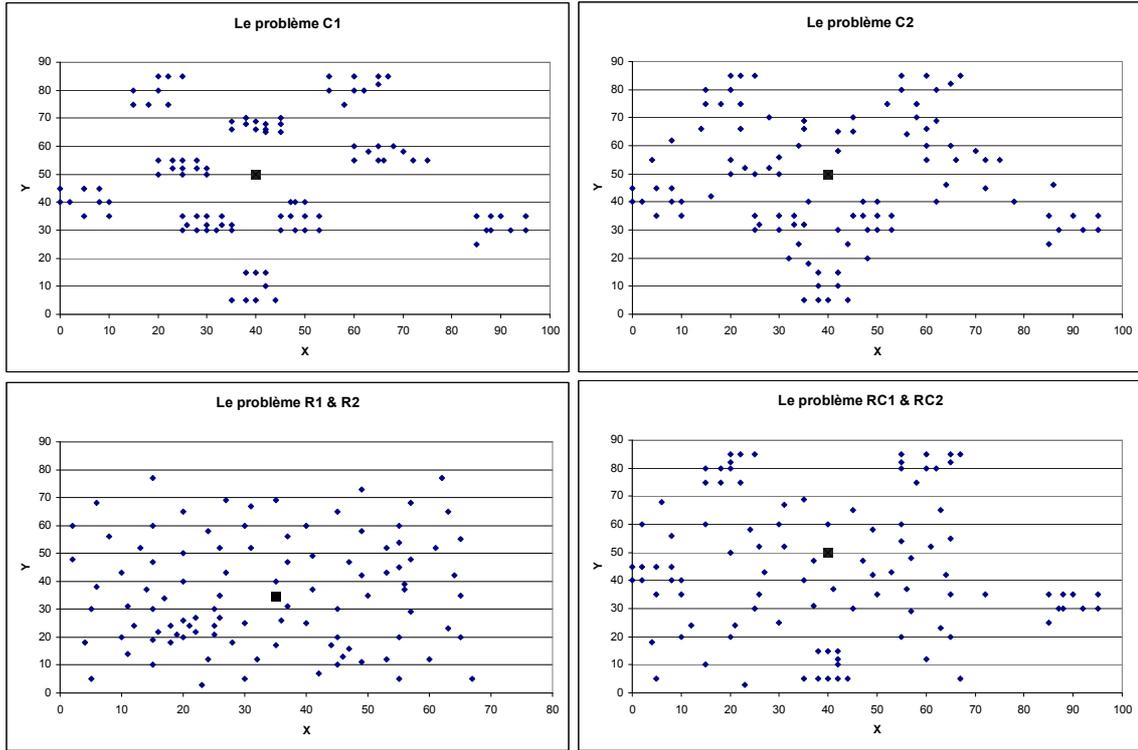


FIG. 5.4 – Les jeux de tests de Solomon pour le VRPTW

Il convient de rappeler que ce benchmark a été conçu initialement pour le problème VRPTW statique. Nous l'avons utilisé pour mettre au point le réglage de nos paramètres génétiques (voir chapitre 4). Il était également intéressant de les utiliser pour mesurer les performances de notre approche dans un milieu statique.

Le benchmark de Salomon a été modifié par [Gendreau et al. 1999a] pour l'adapter au problème VRPTW dynamique. Pour cela, il a ajouté un nouvel attribut à chaque client. Cet attribut, noté (t) , est l'heure d'apparition de la demande. Si $(t = 0)$ pour un client, ce client est déclaré statique c'est-à-dire connu en avance (avant le début d'ouverture de la journée). Dans le cas contraire, il est déclaré dynamique et la valeur de (t) donne l'heure où la demande apparaît dans la journée. Ces clients dynamiques sont aléatoirement choisis parmi l'ensemble des clients.

Donc, l'ensemble des demandes est divisé en deux sous ensembles. Le premier sous ensemble contient toutes les demandes qui sont connues en début de journée ainsi que celles qui ont été reçues la veille du jour courant et qui n'ont pas pu être honorées. Le deuxième sous ensemble contient les demandes qui sont reçues en temps réel durant la journée. Dans ce cas-ci, un instant d'occurrence t est associé à chaque demande. Pour une demande i , cette valeur t est aléatoirement produite dans l'intervalle $[0, \bar{e}_i]$, où :

$$\bar{e}_i = \frac{T}{l_0 - e_0} \cdot \min \{e_i, tdep_{i-1}\}$$

Où, $tdep_{i-1}$ est le temps de fin de service chez le client $(i - 1)$. Ce client est le prédécesseur du client i dans la tournée de la meilleure solution pour le problème statique. T est le temps nécessaire à la simulation d'une journée. e_0 et l_0 sont les horaires d'ouverture et de fermeture du dépôt.

Il est à noter que dans ce benchmark, le nombre de véhicules disponibles n'est pas illimité comme dans la version statique de ce benchmark. En fait, il a été fixé au nombre optimal de véhicules obtenu en résolvant la version statique du problème VRPTW correspondant. Par ailleurs dans notre cas, nous avons fixé à 30 *min* le temps de simulation nécessaire pour simuler une journée d'ouverture. La vitesse des véhicules est aussi égale à 1.0 (unité de distance/unité de temps). La capacité des véhicules des problèmes de type 2, a été aussi modifiée par [Gendreau et al. 1999a] : elle est assez grande pour ne pas engendrer une contrainte de capacité. La capacité des véhicules des problèmes de type 1 est déjà grande.

5.3.2 Les résultats

Nous avons exploité ce benchmark en utilisant notre outil sur une machine sous Windows XP ayant un processeur Pentium 2.4 *GHz*, 256 *Mo* de mémoire vive.

Les résultats globaux obtenus pour l'heuristique d'insertion (voir chapitre 2), l'approche métaheuristique Tabou parallèle utilisée par [Gendreau et al. 1999a], et notre approche sont listés dans la table (5.1). Dans cette table la valeur D représente la distance moyenne parcourue pour tous les problèmes testés d'un groupe, la valeur R représente le retard moyen de tous les problèmes testés d'un groupe, et C est le nombre moyen des clients qui ne sont pas servis à la fin de la journée de service.

La méthode d'insertion insère simplement le nouveau client à la position qui réduit au minimum le coût additionnel de la solution courante. La méthode de la recherche tabou [Gendreau et al. 1999a] a été déjà présentée dans le chapitre 2. Il est à noter que nous avons comparé nos résultats avec ceux de Gendreau en nous limitant à un seul processeur puisque pour le moment notre approche est non parallèle.

Cette table 5.1 illustre une comparaison entre les résultats des trois approches. Comme nous le voyons, nos résultats, issus de l'approche d'algorithme génétique dynamique **AGD**, sont comparables avec les autres approches. Globalement nous pouvons affirmer qu'une approche AG est à la fois viable et peut fournir des performances intéressantes. Pour les deux groupes dynamiques ($R2$, $RC2$) nous avons obtenu de meilleurs résultats que les autres approches. Par contre, en ce qui concerne les autres groupes, nous n'avons obtenu de bons résultats que pour certains critères. Globalement, le comportement général de notre approche est meilleur que celui des deux autres approches : ceci apparaît dans la dernière ligne de la table nommée "**Tous**", qui est la moyenne de tous les groupes du benchmark dynamique. Cependant, en réalité il est difficile de comparer nos résultats avec ceux d'autres approches, car celles-ci ont utilisé des types de machines et de système d'exploitation différents de ceux utilisés par notre approche. De même les temps de simulations pris par les méthodes de résolution sont aussi différents. Donc, nous pouvons raisonnablement dire que nous avons montré dans cette thèse que l'algorithme génétique est efficace pour la résolution des problèmes de transport dynamiques et aussi performant que d'autres approches reconnues (i.e. Tabou).

Comme nous l'avons déjà vu dans le chapitre 4, nous avons utilisé ce benchmark pour régler les paramètres de l'algorithme génétique. Donc pour chaque groupe de ce benchmark nous avons réglé l'algorithme en utilisant une des deux combinaisons possibles selon le degré de dynamisme du problème traité.

Dans un second temps, nous pouvons effectuer une comparaison détaillée pour chaque problème dynamique de l'approche AGD avec la méthode d'insertion (la table 5.2). En effet nous

| Le problème | | Insertion | Recherche Tabou | AGD |
|---------------|---|-----------|-----------------|----------------|
| C1 Dynamique | D | 1234,24 | 857,20 | 840,70 |
| | R | 365,44 | 1,30 | 1,44 |
| | C | 8,40 | 0,00 | 0,00 |
| R1 Dynamique | D | 1297,76 | 1210,80 | 1193,10 |
| | R | 393,88 | 51,70 | 51,07 |
| | C | 10,70 | 0,58 | 0,75 |
| RC1 Dynamique | D | 1393,27 | 1360,30 | 1324,74 |
| | R | 573,19 | 65,90 | 80,02 |
| | C | 8,40 | 0,50 | 0,03 |
| C2 Dynamique | D | 798,81 | 628,70 | 618,63 |
| | R | 282,08 | 0,00 | 0,89 |
| | C | 0,13 | 0,00 | 0,00 |
| R2 Dynamique | D | 1199,84 | 1050,40 | 1047,22 |
| | R | 523,01 | 40,40 | 18,62 |
| | C | 0,51 | 0,45 | 0,00 |
| RC2 Dynamique | D | 1388,48 | 1192,60 | 1192,53 |
| | R | 522,93 | 43,40 | 25,71 |
| | C | 0,58 | 0,00 | 0,00 |
| Tous | D | 1218,73 | 1050,00 | 1036,15 |
| | R | 443,42 | 33,78 | 29,62 |
| | C | 4,78 | 0,26 | 0,13 |

TAB. 5.1 – Comparaison des résultats obtenus par plusieurs méthodes

avons programmé ces deux méthodes et elles ont été exécutées dans des conditions identiques : le type de la machine, le système d'exploitation et le temps de simulation.

| Problème | | Insertion | AGD | Problème | Insertion | AGD | Problème | Insertion | AGD |
|----------|---|-----------|---------|----------|-----------|---------|----------|-----------|---------|
| C101 | D | 1021,75 | 828,94 | R111 | 1284,26 | 1095,46 | R201 | 1446,11 | 1300,64 |
| | R | 323,54 | 0,00 | | 558,01 | 75,74 | | 235,58 | 37,53 |
| | C | 0,00 | 0,00 | | 20,80 | 1,60 | | 0,00 | 0,00 |
| C102 | D | 1284,31 | 833,11 | R112 | 1226,30 | 1044,38 | R202 | 1398,79 | 1237,45 |
| | R | 600,09 | 0,00 | | 547,44 | 6,20 | | 577,41 | 30,82 |
| | C | 3,60 | 0,00 | | 23,20 | 0,00 | | 0,00 | 0,00 |
| C103 | D | 1351,67 | 842,61 | R203 | | | | 1221,35 | 1052,43 |
| | R | 560,32 | 0,00 | | | | | 249,72 | 7,61 |
| | C | 9,60 | 0,00 | | | | | 0,00 | 0,00 |
| C104 | D | 1157,31 | 844,00 | RC101 | 1614,77 | 1483,08 | R204 | 1059,34 | 929,18 |
| | R | 65,46 | 0,00 | | 734,63 | 184,41 | | 740,47 | 30,27 |
| | C | 3,80 | 0,00 | | 11,00 | 0,00 | | 1,00 | 0,00 |
| C105 | D | 944,09 | 825,43 | RC102 | 1437,97 | 1466,26 | R205 | 1308,58 | 1116,90 |
| | R | 118,00 | 0,00 | | 518,22 | 49,79 | | 473,69 | 0,53 |
| | C | 3,60 | 0,00 | | 4,20 | 0,00 | | 0,00 | 0,00 |
| C106 | D | 1107,19 | 835,38 | RC103 | 1345,78 | 1276,25 | R206 | 1208,13 | 1046,64 |
| | R | 334,19 | 0,00 | | 399,06 | 47,51 | | 303,48 | 0,88 |
| | C | 5,00 | 0,00 | | 6,80 | 0,00 | | 0,00 | 0,00 |
| C107 | D | 1316,24 | 824,68 | RC104 | 1205,67 | 1150,15 | R207 | 1090,76 | 928,54 |
| | R | 481,51 | 0,00 | | 188,25 | 18,55 | | 183,61 | 1,14 |
| | C | 8,00 | 0,00 | | 1,20 | 0,00 | | 0,00 | 0,00 |
| C108 | D | 1578,84 | 851,95 | RC105 | 1520,42 | 1398,28 | R208 | 940,51 | 838,58 |
| | R | 618,62 | 12,98 | | 604,13 | 104,40 | | 433,83 | 0,00 |
| | C | 22,00 | 0,00 | | 10,60 | 0,00 | | 1,20 | 0,00 |
| C109 | D | 1346,76 | 880,24 | RC106 | 1366,34 | 1384,75 | R209 | 1184,37 | 973,08 |
| | R | 187,28 | 0,00 | | 929,74 | 76,48 | | 454,20 | 0,00 |
| | C | 20,00 | 0,00 | | 12,60 | 0,20 | | 0,00 | 0,00 |
| RC107 | | | | RC107 | 1352,27 | 1253,76 | R210 | 1231,88 | 1095,69 |
| | | | | | 519,33 | 94,13 | | 361,76 | 0,20 |
| | | | | | 9,80 | 0,00 | | 0,00 | 0,00 |
| R101 | D | 1557,70 | 1583,75 | RC108 | 1302,97 | 1185,36 | R211 | 1108,49 | 1000,29 |
| | R | 301,33 | 66,87 | | 692,18 | 64,87 | | 1739,36 | 95,80 |
| | C | 7,00 | 0,00 | | 11,00 | 0,00 | | 3,40 | 0,00 |
| R102 | D | 1413,76 | 1392,94 | C201 | 634,89 | 591,56 | RC201 | 1692,14 | 1461,35 |
| | R | 250,54 | 59,91 | | 71,79 | 0,00 | | 469,30 | 4,71 |
| | C | 5,60 | 0,00 | | 0,00 | 0,00 | | 4,00 | 0,00 |
| R103 | D | 1266,57 | 1111,88 | C202 | 707,49 | 591,56 | RC202 | 1692,14 | 1461,35 |
| | R | 209,02 | 14,89 | | 37,34 | 0,00 | | 1550,51 | 1252,83 |
| | C | 6,40 | 0,00 | | 0,40 | 0,00 | | 181,18 | 17,29 |
| R104 | D | 1088,02 | 1029,28 | C203 | 1119,23 | 627,07 | RC203 | 1358,40 | 1161,66 |
| | R | 98,04 | 18,96 | | 1565,37 | 0,00 | | 399,56 | 20,22 |
| | C | 0,00 | 0,00 | | 0,00 | 0,00 | | 0,00 | 0,00 |
| R105 | D | 1498,94 | 1381,52 | C204 | 730,86 | 777,42 | RC204 | 1099,34 | 935,17 |
| | R | 450,42 | 77,94 | | 0,00 | 7,10 | | 185,34 | 1,12 |
| | C | 17,00 | 0,00 | | 0,00 | 0,00 | | 0,60 | 0,00 |
| R106 | D | 1328,91 | 1220,29 | C205 | 833,05 | 588,88 | RC205 | 1531,32 | 1372,11 |
| | R | 480,88 | 57,99 | | 177,74 | 0,00 | | 446,41 | 55,02 |
| | C | 5,00 | 0,00 | | 0,00 | 0,00 | | 0,00 | 0,00 |
| R107 | D | 1236,01 | 1079,89 | C206 | 736,50 | 588,49 | RC206 | 1417,05 | 1253,85 |
| | R | 442,98 | 55,37 | | 204,41 | 0,00 | | 902,49 | 12,68 |
| R108 | D | 1076,58 | 1001,49 | | | | | | |
| | R | 362,71 | 52,44 | | | | | | |

A suivre ⇒

| Problème | | Insertion | AGD | Problème | Insertion | AGD | Problème | Insertion | AGD |
|----------|---|-----------|---------|----------|-----------|--------|----------|-----------|---------|
| | C | 0,60 | 0,00 | | 0,00 | 0,00 | | 0,00 | 0,00 |
| R109 | D | 1305,31 | 1208,69 | C207 | 776,95 | 593,40 | RC207 | 1270,59 | 1168,56 |
| | R | 553,75 | 75,74 | | 97,02 | 0,00 | | 1072,52 | 85,63 |
| | C | 18,80 | 0,00 | | 0,60 | 0,00 | | 0,00 | 0,00 |
| R110 | D | 1290,77 | 1167,66 | C208 | 851,52 | 590,68 | RC208 | 1188,47 | 934,69 |
| | R | 471,51 | 50,74 | | 102,99 | 0,00 | | 526,65 | 9,01 |
| | C | 17,00 | 7,40 | | 0,00 | 0,00 | | 0,00 | 0,00 |

TAB. 5.2: AGD vs. l'heuristique d'insertion

Dans cette table nous pouvons vérifier que notre approche est toujours meilleure que la méthode d'insertion simple comme on pouvait s'y attendre.

Globalement, nous pouvons conclure que notre approche est efficace et avec des performances comparables à celles d'autres approches qui sont conçues pour résoudre ce type de problème.

5.4 Le problème DVRP sans fenêtres de temps

Dans cette section quelques résultats sont présentés afin d'évaluer notre algorithme pour la résolution du problème d'élaboration de tournées de véhicules dynamique sans fenêtres de temps (**DVRP**). Ce problème est presque identique au précédent. D'abord nous allons présenter le benchmark utilisé pour ce problème puis nous illustrons les résultats obtenus. L'objectif de ce problème est de minimiser la distance totale parcourue en respectant les contraintes du problème VRP.

5.4.1 Le benchmark utilisé

[Kilby et al. 1998a] a proposé des benchmarks adaptés. Ils sont dérivés des ensembles de benchmarks créés pour le VRP statique : 12 problèmes sont pris de [Taillard 1994], 7 problèmes sont de [Christophides et al. 1984] et 2 problèmes sont pris de [Fisher et al. 1981]. Ces problèmes s'étendent de 50 à 199 clients. Le nombre de clients est indiqué dans le nom de chaque problème (*tai75a* signifie un problème VRP statique issu du benchmark de Taillard, et contenant 75 clients). Kilby a ajouté le concept de durée de journée à tous ces problèmes. À chaque client, il a ajouté le temps d'apparition de la demande client. Celui-ci représente le moment où la demande devient disponible pendant la journée de service. Une durée requise pour servir une demande est aussi ajoutée. Il a également fixé un nombre de véhicules disponibles suffisamment grand pour chaque problème (50 véhicules) ; il s'agit donc d'une contrainte qui n'est donc pas limitative. En plus, nous devons respecter la contrainte de capacité des véhicules et la vitesse des véhicules qui est égale à 1.0 (unité de distance/unité de temps).

5.4.2 Les résultats

La table (5.3) montre une comparaison entre l'implémentation de notre approche avec l'approche utilisée par [Montamenni et al. 2002]. Pour chaque problème, nous avons effectué cinq essais. Les valeurs dans cette table sont pour chaque méthode, la valeur minimale, maximale, et la moyenne de la distance totale parcourue par les véhicules pour les cinq essais. [Montamenni et al. 2003] a proposé une méthode métaheuristique par colonie de fourmis pour la résolution de ce problème. En fait, il a développé deux approches (**MSLS** et **ACS**) : l'algorithme

ACS est simplement la métaheuristique des colonies de fourmis, par contre l'algorithme MSLS (Multi Start Local Search) est une version des colonies de fourmis sans utilisation de phéromone.

Enfin, Montemanni a fixé le temps de la simulation à (1500 secondes) et a utilisé une machine équipée d'un processeur Intel Pentium IV à 1.5 GHz et d'une mémoire de 256 Mo ; l'algorithme est écrit en C ANSI. Pour chaque problème, nous avons démarré l'algorithme génétique dynamique (la simulation) cinq fois, sur une machine équipée d'un processeur Intel Pentium IV à 2.4 GHz et d'une mémoire de 256 Mo pendant aussi 1500 secondes. En plus, nous avons utilisé le réglage suivant :

1. Taille de la population : 50
2. Type de croisement : PMX
3. Probabilité de croisement : 0.7
4. Type de mutation : Or-Opt
5. Probabilité de mutation : 0.2
6. Taux d'élitisme : 50%

En fait, cette combinaison a été construite d'après les valeurs les plus importantes de la table 4.13. Nous avons choisis (PMX) comme opérateur de croisement parce qu'avec les problèmes sans fenêtres de temps on ne peut pas choisir l'opérateur (MX1T/D).

Nous constatons que nos résultats sont comparables et bien souvent meilleurs que les deux autres méthodes.

5.5 Le problème DTRP

Le problème DTRP (Dynamic Traveling Repairman Problem) a été traité par Larsen dans sa thèse [Larsen 2001]. Nous avons déjà présenté ce problème dans le premier et le deuxième chapitre. Ce problème est considéré comme une version simplifiée du problème VRP dynamique, mais dans le cadre de la thèse de Larsen, il n'y a qu'un seul véhicule (celui du réparateur) ; Ce réparateur doit passer chez chaque client qui a déjà demandé le service, par conséquent, nous n'avons ni la contrainte de capacité du véhicule, ni la contrainte de fenêtre de temps. L'objectif de ce problème est de minimiser la distance totale parcourue par le réparateur et de minimiser le temps d'attente des clients : temps écoulé entre l'instant où la demande du client est formulée et l'instant où celui-ci commence à être servi.

5.5.1 Le benchmark utilisé

Nous avons utilisé le benchmark proposé par [Larsen 2001] pour examiner le comportement de notre approche pour la résolution de ce type de problème. Spécifiquement, nous nous sommes concentrés sur l'impact du degré de dynamisme sur la distance parcourue et le temps d'attente des clients. Larsen a généré un ensemble de benchmarks dynamiques pour ce problème. Il a tout d'abord choisi des positions géographiques des clients uniformément distribués dans une région $[0, 10000] \times [0, 10000]$, avec le dépôt situé au centre en position (5000, 5000). La vitesse des véhicules a été fixée à 40 km/heure. Il a employé une échelle de graduation de 100 unités par kilomètre ; la région de service est donc équivalente à [10 kilomètres x 10 kilomètres]. Les temps de service ont été produits selon une distribution logarithmique normale. Ceci signifie que le temps moyen de service était de 3 minutes, tandis que la variance était de 5 minutes. Ces

| Problèmes dynamiques de Kilby | Montemanni | | | | | | AGD | | |
|-------------------------------|------------|-------|---------|-------|-------|---------|-------|-------|---------|
| | MSLS | | | ACS | | | Min. | Max. | Moyenne |
| | Min. | Max. | Moyenne | Min. | Max. | Moyenne | | | |
| tai75a | 1911 | 2141 | 2012 | 1843 | 2044 | 1945 | 1808 | 1896 | 1864 |
| tai75b | 1635 | 1934 | 1782 | 1535 | 1924 | 1704 | 1467 | 1594 | 1531 |
| tai75c | 1606 | 1886 | 1696 | 1575 | 1842 | 1654 | 1490 | 1718 | 1622 |
| tai75d | 1545 | 1642 | 1589 | 1472 | 1647 | 1529 | 1452 | 1541 | 1505 |
| tai100a | 2427 | 2583 | 2510 | 2376 | 2576 | 2428 | 2269 | 2478 | 2422 |
| tai100b | 2303 | 2533 | 2407 | 2284 | 2456 | 2348 | 2167 | 2310 | 2241 |
| tai100c | 1599 | 1801 | 1704 | 1562 | 1804 | 1656 | 1646 | 1853 | 1744 |
| tai100d | 2027 | 2165 | 2110 | 2008 | 2142 | 2061 | 1783 | 2001 | 1908 |
| tai150a | 3788 | 4165 | 3982 | 3645 | 4214 | 3840 | 3634 | 3783 | 3682 |
| tai150b | 3313 | 3656 | 3486 | 3167 | 3452 | 3327 | 3140 | 3450 | 3307 |
| tai150c | 3090 | 3635 | 3253 | 2811 | 3227 | 3016 | 2753 | 3076 | 2914 |
| tai150d | 3159 | 3541 | 3324 | 3059 | 3383 | 3204 | 3110 | 3272 | 3189 |
| c50 | 694 | 757 | 722 | 631 | 756 | 682 | 576 | 628 | 598 |
| c75 | 1067 | 1142 | 1099 | 1009 | 1087 | 1042 | 932 | 999 | 968 |
| c100 | 1080 | 1170 | 1124 | 973 | 1101 | 1066 | 977 | 1078 | 1025 |
| c100b | 978 | 1173 | 1041 | 944 | 1124 | 1024 | 886 | 967 | 931 |
| c120 | 1547 | 1875 | 1752 | 1416 | 1622 | 1525 | 1335 | 1480 | 1430 |
| c150 | 1468 | 1542 | 1493 | 1346 | 1522 | 1456 | 1318 | 1381 | 1343 |
| c199 | 1774 | 1957 | 1898 | 1771 | 1999 | 1845 | 1642 | 1711 | 1681 |
| f71 | 369 | 437 | 390 | 311 | 420 | 349 | 272 | 293 | 283 |
| f134 | 16073 | 17326 | 16867 | 15136 | 17306 | 16084 | 15256 | 16130 | 15721 |

TAB. 5.3 – AGD vs. ACS (DVRP)

valeurs ont été choisies pour ressembler aux temps de ramassage et de livraison dans les services de courrier longue distance.

Deux groupes d'instances dynamiques de **DTRP** ont été exploités. Ces deux groupes sont constitués de 30 clients et de 40 clients. Chaque groupe contient des problèmes ayant des degrés de dynamisme différents $\{0\%, 5\%, 10\%, \dots, 95\%, 100\%\}$; soit 21 sous groupes au total. Chacun de ces sous groupes possède 50 problèmes (instances) différents. Donc chaque groupe contient au total $21 \times 50 = 1050$ problèmes. La journée de service commence à $8h$ et se termine à $16h$: le temps d'ouverture de la journée est égale à 8 heures. Nous avons utilisé (10 minutes) pour simuler la journée de service.

5.5.2 Les résultats

Sur la figure 5.5 nous avons représenté nos résultats graphiquement parce que malheureusement nous n'avons pas pu récupérer les résultats numériques des problèmes de Larsen. L'allure de ces courbes montre les performances de la méthode lorsque le nombre de clients dynamiques augmente. Dans cette figure, les graphiques à gauche représentent respectivement :

1. La distance en kilomètres en fonction du degré de dynamisme pour le groupe de problèmes ayant 30 clients.
2. La distance en kilomètres en fonction du degré de dynamisme pour le deuxième groupe des problèmes ayant 40 clients.
3. Le temps d'attente en minutes en fonction du degré de dynamisme également pour le groupe de 40 clients.

Les graphiques à droite concernent les résultats de plusieurs stratégies de routage implémentées par Larsen ; Ces stratégies ont été déjà présentées dans le chapitre (2). Dans la stratégie FCFS (First Come First Serve), les demandes sont servies dans l'ordre dans lequel elles sont reçues par le planificateur. La stratégie SQM (Stochastic Queue Median) est comme la précédente, mais le réparateur part directement de la médiane de la région de service au prochain client. Une fois le service accompli, le réparateur revient si besoin à la médiane et attend la prochaine demande. Dans la stratégie NN (Nearest Neighbor), après avoir servi un client, le réparateur se dirige vers le client le plus proche non encore servi. Enfin dans la stratégie PART (PARTitioning policy) la région de service est divisée en sous régions dans lesquelles les demandes sont servies en utilisant la stratégie FCFS.

Dans cette figure 5.5, Larsen a utilisé en plus des stratégies mentionnées ci-dessus, des stratégies hybrides, comme la stratégie FCFS-SQM, et la stratégie NN-FCFS.

En comparant les trois graphiques de nos résultats avec ceux de Larsen, nous notons que :

1. Nos résultats sont proches et ont la même forme générale (une forme croissante) que les résultats des meilleures stratégies implémentées par Larsen (i.e. NN-FCFS, NN, PART4 et PART9).
2. La distance parcourue moyenne pour tous les problèmes des deux groupes est en corrélation linéaire avec le degré de dynamisme du problème traité.

Comme nous l'avons déjà signalé, notre but était d'implémenter une approche évolutionniste pour le problème VRP dynamique et ses variantes comme le problème DTRP afin de montrer la faisabilité et l'efficacité de notre approche. Les résultats obtenus montrent que les performances sont comparables aux autres approches utilisées.

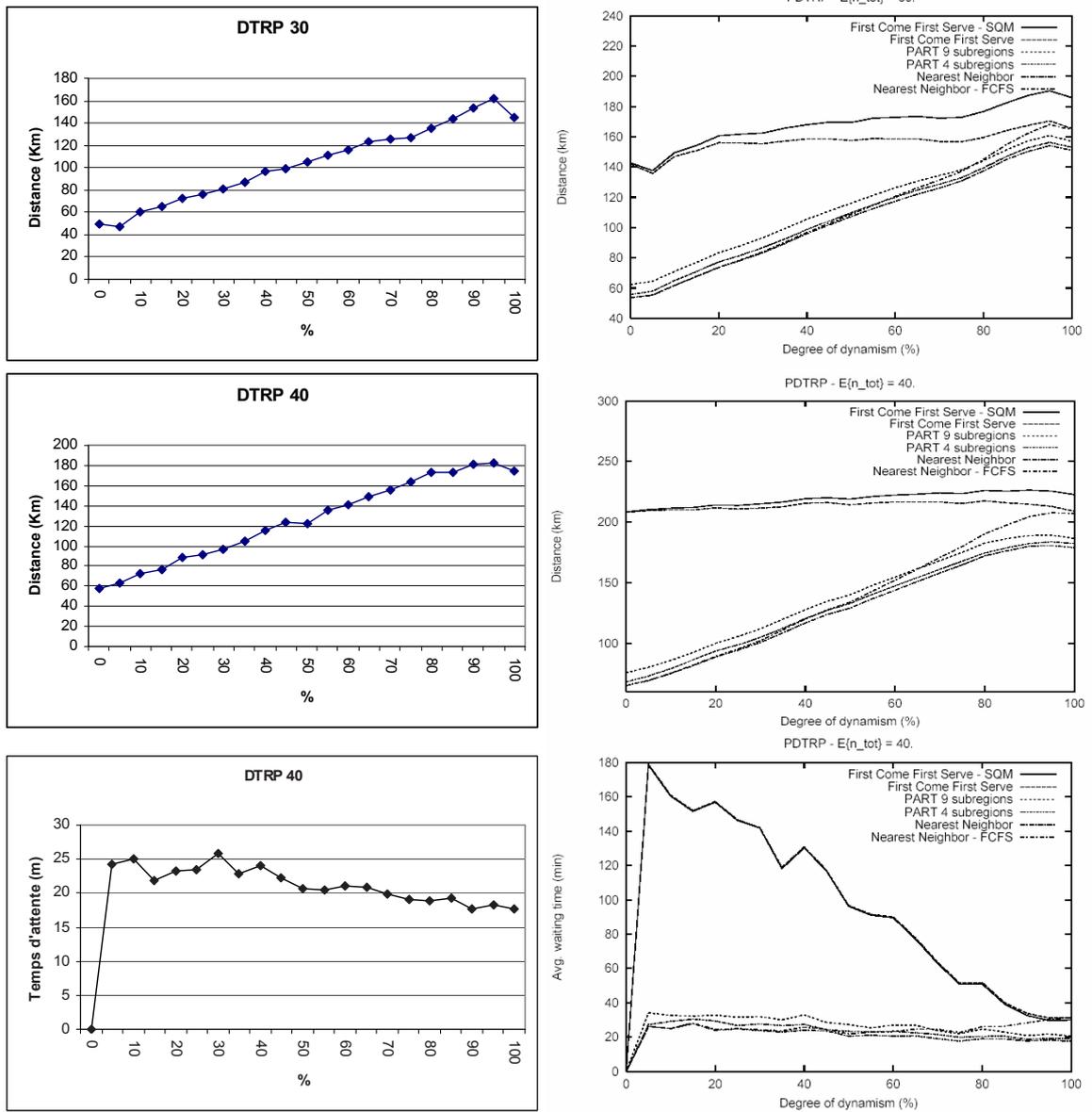


FIG. 5.5 – Les résultats du problème DTRP

5.6 Le problème DTSPTW

Dans cette section, nous discutons des résultats que nous avons obtenus pour le problème dynamique du voyageur de commerce avec fenêtres de temps DTSPTW et nous les comparerons avec ceux obtenus par [Larsen et al. 2004]. En fait, Larsen a proposé une heuristique dynamique efficace pour la résolution de ce problème. Cette heuristique a été déjà présentée dans le chapitre (2). Il a aussi proposé plusieurs stratégies pour le positionnement du véhicule de service dans le cas où il n’y a pas de client à servir (état “*oisif*”); ces stratégies utilisent des informations statistiques sur l’apparition de nouvelles demandes pour pouvoir prévoir les endroits où positionner le véhicule. Ces stratégies ont été aussi présentées dans le chapitre (2).

Ce problème est aussi une extension du problème DVRP; il est proche du problème dynamique du réparateur DTRP : nous avons aussi un véhicule qui doit faire une tournée entre différents clients. L’objectif à satisfaire est de minimiser la distance totale parcourue par ce voyageur en respectant les fenêtres de temps des clients, étant donné qu’il n’y a pas de contrainte de capacité de véhicule.

5.6.1 Le benchmark utilisé

La même méthode, utilisée pour la génération de benchmarks dynamiques du problème DTRP, a été adaptée pour la génération de benchmarks dynamiques du problème DTSPTW. Larsen a généré 21 groupes de problèmes selon des degrés variables de dynamisme $\{0\%, 5\%, 10\%, \dots, 95\%, 100\%\}$. Pour chaque groupe il a généré 100 instances, donc il a généré au total 2100 instances. Chaque instance représente un problème dynamique de type DTSPTW avec 40 clients et la vitesse du véhicule est égale à 40 Km/heure. Nous avons aussi utilisé 10 minutes pour simuler la journée de service du voyageur.

5.6.2 Les résultats

Dans la figure 5.6, en comparant les résultats obtenus par Larsen (les deux derniers graphiques) issus de l’utilisation de plusieurs stratégies, avec ceux obtenus par notre approche (le premier graphique), nous constatons que nos résultats sont meilleurs.

5.7 Le problème DPDPTW

Le problème dynamique de ramassage et de livraison avec fenêtres de temps (DPDPTW) a été traité par [Mitrovic-Minic 2001]; ce problème est similaire au problème DVRPTW, excepté que chaque client doit fournir deux positions géographiques pour que le véhicule puisse ramasser le produit à la première position et le déposer dans la deuxième. Dans ce problème nous devons :

1. Respecter les contraintes de temps. En sachant que les fenêtres de temps des clients sont dures : le véhicule doit arriver chez le client à servir strictement entre les deux limites de temps fournies par ce client.
2. Respecter les contraintes de précedence, cela signifie que pour un seul client la position de ramassage doit toujours précéder celle de livraison.
3. Supposer que la contrainte de capacité des véhicules n’existe pas.
4. Supposer que les véhicules sont homogènes et ont les mêmes vitesses.
5. Supposer que le nombre des véhicules disponibles est illimité.

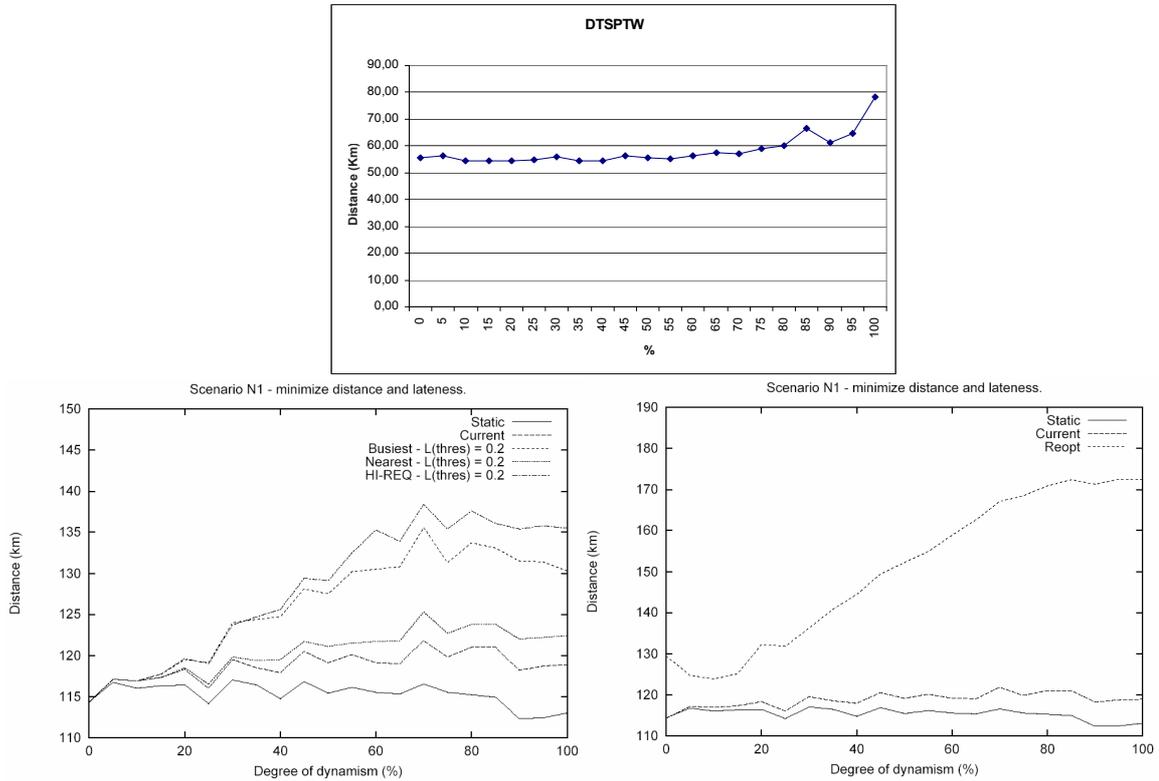


FIG. 5.6 – Les résultats du problème DTSPTW

- Supposer que le dépôt n'existe pas : dans ce cas chaque route commence à une position prédéfinie (au centre de la région de service) et se termine à la position du dernier client servi (la route est ouverte).

L'objectif à satisfaire est de minimiser la distance totale parcourue par les véhicules. Dans ce problème nous traitons aussi les deux événements : arrivée d'un nouveau client et fin de service. Comme nous avons vu dans le chapitre (2), [Mitrovic-Minic et al. 2004b] ont utilisé plusieurs stratégies de positionnement des véhicules dans le cas où il n'existe plus de clients à servir : cas d'attente de nouveaux clients.

5.7.1 Le benchmark utilisé

[Mitrovic-Minic et al. 2004b] ont généré deux ensembles d'instances, mais nous n'avons pu récupérer que le premier. Les expériences ont été exécutées sur cet ensemble d'instances. Cet ensemble contient 30 instances avec 100 clients. Les instances produites sont basées sur des données réelles rassemblées auprès de deux entreprises de courrier fonctionnant à Vancouver au Canada. La période de service est fixée à dix heures, la région de service est $[60 \times 60] \text{ km}^2$, et la vitesse des véhicules est de 60 km/heure. Les demandes se produisent pendant la période de service selon une distribution uniforme continue, et aucune demande n'est connue à l'avance.

Le rejet des demandes et les violations des fenêtres de temps ne sont pas autorisées. Ceci est rendu possible par le fait qu'on suppose que la taille de la flotte de véhicules est illimitée. Les coûts d'utilisation des véhicules sont entièrement à la charge des conducteurs. C'est pourquoi nous n'essayons pas de minimiser le nombre de véhicules. La flotte initiale est égale à 20 véhicules.

La même position de départ à ($x = 20$ kilomètres ; $y = 30$ kilomètres) a été employée pour tous les véhicules.

Toutes les expériences ont été effectuées en utilisant la vitesse de simulation s . Cette vitesse de simulation signifie qu'une heure d'opération réelle est simulée en $\frac{1}{s}$ heures de temps machine ; c'est-à-dire le temps d'ouverture est de s heures et le temps de simulation est de 1 heure, soit un rapport de $\frac{1}{s}$.

5.7.2 Les résultats

Dans ce problème nous avons aussi utilisé les plans d'expériences pour le réglage des paramètres de l'algorithme génétique. Pour cela, nous devons régler les six paramètres que nous avons déjà réglés pour le problème DVRPTW. Dans ce problème nous avons aussi utilisé les mêmes niveaux de paramètres, sauf que les deux niveaux du paramètre "taille de la population" ont été réduits à (10, 20) (voir la table 5.4). Des valeurs plus importantes (30, 50, etc.) vont ralentir la rapidité de l'algorithme génétique, la représentation chromosomique pour ce problème est différente. Chaque client est représenté deux fois dans le chromosome, une fois pour le ramassage et l'autre fois pour la livraison. Dans ce cas la taille du chromosome pour ce problème est doublée. Par conséquent l'étape de décodage du chromosome va prendre plus de temps que dans le cas du problème DVRPTW.

| Niveau | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 |
|--------|-------|--------|-------|--------|-------|-------|
| 1 | 10 | PMX | 0.5 | 1-Opt | 0.2 | 50% |
| 2 | 20 | MX1T/D | 0.7 | Or-Opt | 0.5 | 70% |

TAB. 5.4 – Les 2 niveaux des paramètres de l'AG à régler pour le problème DPDPTW

A ce stade, nous avons choisi trois problèmes parmi les trente de l'ensemble des problèmes du benchmark ; nous avons construit trois plans fractionnaires pour ces trois problèmes. Dans la table 5.5, nous avons mis la meilleure combinaison pour chaque problème correspondant (voir l'annexe A.5 pour les plans détaillés).

| Problème | Taille | Cr | T2Cr | Mt | T2Mt | Elitisme |
|-----------|--------|------------|------|--------------|------|----------|
| 10_100_05 | 10 | MX1T/D | 0,5 | Or-Opt | 0,2 | 70% |
| 10_100_15 | 10 | <u>PMX</u> | 0,5 | <u>1-Opt</u> | 0,2 | 50% |
| 10_100_25 | 20 | PMX | 0,5 | <u>1-Opt</u> | 0,5 | 70% |

TAB. 5.5 – Les résultats des plans fractionnaires pour le problème DPDPTW

Dans cette table, les cellules soulignées sont les niveaux influents des paramètres. En fait nous avons utilisé une seule configuration pour la résolution de tous les problèmes. Cette configuration est calculée à partir de cette table en utilisant les niveaux les plus fréquents dans chaque colonne. En conséquence, la configuration est 10 pour la taille de la population, PMX pour le type de croisement, 0.5 la probabilité de croisement, 1-Opt le type de mutation, 0.2 la probabilité de la mutation et 70% pour le taux d'élitisme.

Dans la table 5.6, nous avons comparé nos résultats avec ceux obtenus par [Mitrovic-Minic et al. 2004a], qui a utilisé une méthode spécifique nommée "double horizon"

pour la résolution du problème DPDPTW. Dans cette méthode, l'horizon de service a été divisé en deux parties : l'horizon à court terme et l'horizon à long terme. L'objectif consiste principalement à trouver une solution pour les demandes dans l'horizon à court terme puis dans la partie de l'horizon à long terme. Sa méthode implémente l'heuristique d'insertion pour trouver la solution. Le problème traité a un degré de dynamisme de 100%, et la méthode d'insertion sert à insérer les nouveaux clients collectés au cours de la tranche de temps précédente (la tranche de temps vaut 15 minutes de la journée de service), dans la meilleure position des tournées existantes. [Mitrovic-Minic et al. 2004a] a ajouté une métaheuristique (Recherche Tabou) pour améliorer la solution jusqu'alors trouvée. La recherche Tabou est appliquée pendant un certain temps après l'opération d'insertion de tous les nouveaux clients.

[Mitrovic-Minic et al. 2004a] a implémenté trois scénarios ($c1$, $c2$, $c3$) ayant des tailles d'horizon différentes, avec deux stratégies (DF : Drive First) et (ADW : Advanced Dynamic Waiting) présentées au chapitre (2), de positionnement des véhicules pour chaque scénario. La stratégie DF consiste lorsque le véhicule a fini son service chez un client, à le faire partir directement et sans retard vers le prochain client à servir. Dans cette table 5.6 nous avons deux valeurs à comparer (la distance totale moyenne parcourue et le nombre des véhicules utilisés). Ici, nous constatons que notre approche utilisant la stratégie DF est meilleure dans tous les cas, du point de vue de la distance. Le temps de simulation est d'une heure du temps de CPU.

| | Double Horizon | | | | | | AGD |
|----------|----------------|---------|---------|---------|---------|---------|----------------|
| | c1 | | c2 | | c3 | | |
| | DF | ADW | DF | ADW | DF | ADW | |
| Distance | 2945.74 | 2742.38 | 2981.03 | 2840.38 | 2922.78 | 2755.70 | 2728.81 |
| m | 19.63 | 19.17 | 18.43 | 18.53 | 19.37 | 19.03 | 19.54 |

TAB. 5.6 – Comparaison entre les résultats

5.8 Conclusion

Dans ce chapitre nous avons présenté les résultats obtenus par notre approche génétique pour traiter le problème dynamique VRP avec ou sans fenêtres de temps. Nous avons montré qu'elle est aussi capable de traiter des extensions de ce problème comme le problème dynamique du réparateur (DTRP) et le problème dynamique du voyageur de commerce avec ou sans fenêtres de temps (DTSPWT).

Grâce à sa nature évolutionniste et sa simplicité, les résultats ont montré que notre approche est performante et efficace pour tous les benchmarks dynamiques traités ici.

Conclusions générales et perspectives

L'objectif de ce travail de cette thèse était d'étudier le potentiel des algorithmes génétiques comme méthode de résolution métaheuristique pour un problème d'optimisation multi-objectif en environnement dynamique. Etant donné son importance pratique, nous avons traité le problème de l'élaboration dynamique des tournées de véhicules avec fenêtres de temps (DVRPTW).

Pour atteindre ce but, nous avons proposé un modèle de résolution approchée "en ligne" basée sur les algorithmes génétiques. Puis pour valider expérimentalement notre approche, nous avons développé un simulateur permettant de traiter et de résoudre ce problème dynamique. Ce simulateur contient deux composants principaux qui collaborent. Ces deux composants sont :

1. Le composant AG (Algorithme Génétique) : Celui-ci traite la résolution du problème en arrière plan pendant la journée de service.
2. Le composant GE (Gestion d'évènements) : Il est dédié à la gestion des évènements issus du problème dynamique. Dans cette thèse nous avons traité deux types d'évènements : l'évènement "arrivée d'un nouveau client" et l'évènement "fin de service chez un client".

Pour évaluer notre approche évolutionniste, nous avons utilisé des benchmarks dynamiques déjà conçus pour plusieurs problèmes dynamiques. Ces benchmarks concernent le problème DVRP et quelques-unes de ses extensions : le problème DVRPTW (Dynamic Vehicle Routing Problem with Time Windows), DTRP (Dynamic Travelling Repairman Problem), DTSPTW (Dynamic Travelling Salesman Problem with Time Windows) et DPDPTW (Dynamic Pick-up and Delivery Problem with Time Windows). Notre outil peut lire les fichiers de données de ces benchmarks selon deux formats : un format texte ou un format XML.

Les résultats obtenus sont très encourageants et comparables avec ceux obtenus avec les meilleures approches utilisées à ce jour. Ces résultats nous permettent de dire que l'algorithme génétique est une méthode alternative, efficace et flexible pour la résolution de ce type de problème dynamique.

Enfin, nous pouvons mentionner les points importants et significatifs de notre travail :

- L'algorithme génétique utilisé est conçu spécialement pour résoudre les problèmes dynamiques et traiter deux types d'évènements.
- Nous utilisons un codage indirect des chromosomes et une heuristique d'insertion simple est nécessaire pour décoder rapidement ces chromosomes.
- La conception d'un simulateur simple utilisant cet algorithme génétique permet la résolution du problème dynamique de DVRPTW. Par ailleurs, ce simulateur permet également de traiter d'autres problèmes dynamiques tels que DPDPTW, DTRP, DTSPTW, etc.

- L’interface graphique développée pour le simulateur permet de changer et de régler les paramètres de l’algorithme génétique, régler les coefficients de la fonction objectif, observer le déroulement de la simulation, surveiller la population courante de l’algorithme génétique et visualiser à tout moment la meilleure solution trouvée.
- Les plans d’expériences ont été mis en oeuvre pour analyser la performance de l’algorithme génétique et régler ses paramètres à des valeurs de compromis.

Les perspectives que nous pouvons donner à ces travaux sont de plusieurs ordres. Sur le plan des applications d’autres problèmes peuvent être facilement adaptés à notre outil comme le problème du bus à la demande, le problème DVRPTW avec une flotte hétérogène de véhicules ou le routage de véhicules d’urgences (i.e. ambulance).

Sur le plan des performances, il est possible de paralléliser l’algorithme génétique en utilisant une stratégie de parallélisation à grain fin, ou à grain gros [Ghiani et al. 2003], [Alba et al. 2004], afin d’accélérer les temps de réponse si le nombre de clients à traiter devenait un facteur critique pour l’obtention de solutions de bonne qualité. Une possibilité intéressante consisterait à permettre à la plate-forme de fonctionner sur un réseau de machines hétérogènes ou un cluster de machines [Nowostawski et al. 1999].

Toujours dans le cadre des améliorations de performances, il serait intéressant d’introduire une recherche locale dans le cadre des opérateurs génétiques comme c’est le cas dans les approches évolutionniste hybride (memetic algorithms) [Berger et al. 2002] pour obtenir des solutions de qualités supérieures.

Sur le plan des fonctionnalités, on pourrait étendre notre outil pour tenir compte de temps de trajet variable selon la période de la journée (i.e. période normale ou période de pointe), ou bien selon la région (i.e. la vitesse des véhicules dans les villes diffère de celle dans les villages) on obtiendrait ainsi une meilleure estimation des temps de trajets réels. De même dans certaines applications (livraisons de fuel, ramassage de lait, . . .) les quantités demandées par les clients sont entachées d’imprécisions qu’il serait possible de prendre en compte avec des méthodes stochastiques ou basées sur la logique floue [Hapke et al. 2003].

Enfin comme nous l’avons déjà vu, notre outil traite uniquement deux événements dynamiques. Dans une approche temps réel, le problème est bien plus compliqué, car nous avons à prendre en compte d’autres événements comme les événements suivants : la panne d’un véhicule, l’annulation d’une demande, le changement de vitesse d’un véhicule suite à un accident, le changement du temps de service chez les clients, etc. Grâce aux technologies actuelles (GPS, systèmes embarqués, SIG) il est possible de suivre directement sur le terrain l’évolution de la flotte de véhicules et de prendre en compte ces événements qui perturbent le prévisionnel pour réévaluer d’autres itinéraires plus appropriés.

Dans cette thèse nous avons traité un problème d’optimisation multi-objectif dynamique, c’est un problème très important dans la vie pratique, il concerne une grande partie des études sur l’optimisation de problèmes des transports dynamiques. Etant donné la grande similitude qui existe entre les problèmes d’ordonnancement et de routage, il est permis de penser que notre approche pourrait être appliquée à d’autres problèmes dynamiques comme ceux rencontrés en ordonnancement dynamique [Beck et al. 2003].

Annexes

Annexe A

Les plans d'expériences

A.1 Plan de criblage

| Nb | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | Y ₁ | Y ₂ |
|----------------|---------|----------|---------|---------|---------|---------|----------------|----------------|
| 1 | 50 | MX1T/D | 0,8 | 2-Opt | 0,05 | 100 | 851 | 8 |
| 2 | 100 | PMX | 0,8 | Or-Opt | 0,5 | 70 | 933 | 512 |
| 3 | 100 | IX | 0,7 | Or-Opt | 0,05 | 100 | 1194 | 6990 |
| 4 | 30 | IX | 0,5 | 1-Opt | 0,05 | 70 | 928 | 483 |
| 5 | 100 | MX1T/D | 0,5 | 2-Opt | 0,2 | 70 | 833 | 5 |
| 6 | 30 | IX | 0,8 | 2-Opt | 0,5 | 50 | 1137 | 4609 |
| 7 | 30 | MX1T/D | 0,5 | Or-Opt | 0,5 | 100 | 845 | 11 |
| 8 | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | 829 | 6 |
| 9 | 100 | MX1T/D | 0,8 | 1-Opt | 0,05 | 50 | 837 | 4 |
| 10 | 50 | IX | 0,8 | Or-Opt | 0,2 | 70 | 949 | 647 |
| 11 | 50 | PMX | 0,5 | Or-Opt | 0,05 | 50 | 971 | 1024 |
| 12 | 30 | PMX | 0,7 | 2-Opt | 0,05 | 70 | 1192 | 4585 |
| 13 | 50 | MX1T/D | 0,7 | 1-Opt | 0,5 | 70 | 836 | 5 |
| 14 | 30 | PMX | 0,8 | 1-Opt | 0,2 | 100 | 1019 | 952 |
| 15 | 30 | MX1T/D | 0,7 | Or-Opt | 0,2 | 50 | 829 | 6 |
| 16 | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | 829 | 6 |
| Moyenne | | | | | | | 938 | 1241 |
| Effets | | | | | | | | |
| Y ₁ | 50 | PMX | 0,7 | 1-Opt | 0,2 | 50 | | |
| | -36,69 | 90,21 | 74,51 | -33,19 | -30,79 | 5,21 | | |
| | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | | |
| | 12,79 | -102,06 | -15,24 | -15,81 | 15,66 | -22,09 | | |
| | 100 | IX | 0,5 | 2-Opt | 0,5 | 100 | | |
| | 11,11 | 113,91 | -44,04 | 64,81 | -0,54 | 38,96 | | |
| Y ₂ | 50 | PMX | 0,7 | 1-Opt | 0,2 | 50 | | |
| | -819,85 | 527,40 | 1655,70 | -879,65 | -838,10 | 170,15 | | |
| | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | | |
| | 91,47 | -1234,35 | -397,85 | -90,68 | 397,25 | -459,78 | | |
| | 100 | IX | 0,5 | 2-Opt | 0,5 | 100 | | |
| | 636,90 | 1941,30 | -860,00 | 1061,00 | 43,60 | 749,40 | | |

TAB. A.1: Plan de criblage du problème C109 statique

Annexe A. Les plans d'expériences

| Nb | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | Y ₁ | Y ₂ |
|----------------|---------|---------|---------|---------|---------|---------|----------------|----------------|
| 1 | 50 | MX1T/D | 0,8 | 2-Opt | 0,05 | 100 | 1317 | 110 |
| 2 | 100 | PMX | 0,8 | Or-Opt | 0,5 | 70 | 1116 | 291 |
| 3 | 100 | IX | 0,7 | Or-Opt | 0,05 | 100 | 1655 | 3972 |
| 4 | 30 | IX | 0,5 | 1-Opt | 0,05 | 70 | 1544 | 2550 |
| 5 | 100 | MX1T/D | 0,5 | 2-Opt | 0,2 | 70 | 1422 | 449 |
| 6 | 30 | IX | 0,8 | 2-Opt | 0,5 | 50 | 1694 | 4174 |
| 7 | 30 | MX1T/D | 0,5 | Or-Opt | 0,5 | 100 | 1359 | 368 |
| 8 | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | 1264 | 266 |
| 9 | 100 | MX1T/D | 0,8 | 1-Opt | 0,05 | 50 | 1445 | 161 |
| 10 | 50 | IX | 0,8 | Or-Opt | 0,2 | 70 | 1163 | 559 |
| 11 | 50 | PMX | 0,5 | Or-Opt | 0,05 | 50 | 1125 | 439 |
| 12 | 30 | PMX | 0,7 | 2-Opt | 0,05 | 70 | 1505 | 2673 |
| 13 | 50 | MX1T/D | 0,7 | 1-Opt | 0,5 | 70 | 1364 | 402 |
| 14 | 30 | PMX | 0,8 | 1-Opt | 0,2 | 100 | 1462 | 743 |
| 15 | 30 | MX1T/D | 0,7 | Or-Opt | 0,2 | 50 | 1078 | 511 |
| 16 | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | 1213 | 590 |
| Moyenne | | | | | | | 1358 | 1141 |
| Effets | | | | | | | | |
| Y ₁ | 50 | PMX | 0,7 | 1-Opt | 0,2 | 50 | | |
| | -115,50 | -56,05 | 42,40 | 95,70 | -76,80 | -22,25 | | |
| | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | | |
| | 32,00 | -50,05 | -23,52 | -111,20 | 25,68 | -34,08 | | |
| | 100 | IX | 0,5 | 2-Opt | 0,5 | 100 | | |
| | 51,50 | 156,15 | 4,65 | 126,70 | 25,45 | 90,40 | | |
| Y ₂ | 50 | PMX | 0,7 | 1-Opt | 0,2 | 50 | | |
| | -763,68 | -104,58 | 748,03 | -177,03 | -575,83 | 180,13 | | |
| | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | | |
| | 343,23 | -783,98 | -279,28 | -266,58 | 204,13 | -168,58 | | |
| | 100 | IX | 0,5 | 2-Opt | 0,5 | 100 | | |
| | 77,23 | 1672,53 | -189,48 | 710,18 | 167,58 | 157,03 | | |

TAB. A.2: Plan de criblage du problème C204 statique

| Nb | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | Y ₁ | Y ₂ |
|----------------|--------|--------|-------|--------|-------|-------|----------------|----------------|
| 1 | 50 | MX1T/D | 0,8 | 2-Opt | 0,05 | 100 | 1613 | 3 |
| 2 | 100 | PMX | 0,8 | Or-Opt | 0,5 | 70 | 1579 | 8 |
| 3 | 100 | IX | 0,7 | Or-Opt | 0,05 | 100 | 1659 | 29 |
| 4 | 30 | IX | 0,5 | 1-Opt | 0,05 | 70 | 1595 | 12 |
| 5 | 100 | MX1T/D | 0,5 | 2-Opt | 0,2 | 70 | 1594 | 3 |
| 6 | 30 | IX | 0,8 | 2-Opt | 0,5 | 50 | 1638 | 12 |
| 7 | 30 | MX1T/D | 0,5 | Or-Opt | 0,5 | 100 | 1602 | 7 |
| 8 | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | 1598 | 4 |
| 9 | 100 | MX1T/D | 0,8 | 1-Opt | 0,05 | 50 | 1577 | 2 |
| 10 | 50 | IX | 0,8 | Or-Opt | 0,2 | 70 | 1590 | 8 |
| 11 | 50 | PMX | 0,5 | Or-Opt | 0,05 | 50 | 1588 | 12 |
| 12 | 30 | PMX | 0,7 | 2-Opt | 0,05 | 70 | 1619 | 10 |
| 13 | 50 | MX1T/D | 0,7 | 1-Opt | 0,5 | 70 | 1583 | 4 |
| 14 | 30 | PMX | 0,8 | 1-Opt | 0,2 | 100 | 1623 | 33 |
| 15 | 30 | MX1T/D | 0,7 | Or-Opt | 0,2 | 50 | 1568 | 3 |
| 16 | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | 1592 | 2 |
| Moyenne | | | | | | | 1601 | 10 |
| Effets | | | | | | | | |
| Y ₁ | 50 | PMX | 0,7 | 1-Opt | 0,2 | 50 | | |
| | -7,48 | 1,47 | 6,22 | -6,48 | -7,33 | -8,23 | | |
| | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | | |
| | 3,15 | -10,30 | 0,10 | -4,18 | 3,97 | -7,40 | | |
| Y ₂ | 100 | IX | 0,5 | 2-Opt | 0,5 | 100 | | |
| | 1,17 | 19,12 | -6,43 | 14,82 | -0,63 | 23,02 | | |
| | 50 | PMX | 0,7 | 1-Opt | 0,2 | 50 | | |
| | -2,75 | 6,00 | 2,05 | 3,35 | 2,15 | -2,25 | | |
| Y ₂ | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | | |
| | 0,90 | -5,85 | -0,47 | -0,30 | -0,27 | -3,08 | | |
| | 100 | IX | 0,5 | 2-Opt | 0,5 | 100 | | |
| | 0,95 | 5,70 | -1,10 | -2,75 | -1,60 | 8,40 | | |

TAB. A.3: Plan de criblage du problème R101 statique

| Nb | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | Y_1 | Y_2 |
|---------|--------|--------|--------|--------|--------|--------|-------|-------|
| 1 | 50 | MX1T/D | 0,8 | 2-Opt | 0,05 | 100 | 1723 | 52 |
| 2 | 100 | PMX | 0,8 | Or-Opt | 0,5 | 70 | 1544 | 60 |
| 3 | 100 | IX | 0,7 | Or-Opt | 0,05 | 100 | 1858 | 187 |
| 4 | 30 | IX | 0,5 | 1-Opt | 0,05 | 70 | 1639 | 84 |
| 5 | 100 | MX1T/D | 0,5 | 2-Opt | 0,2 | 70 | 1538 | 15 |
| 6 | 30 | IX | 0,8 | 2-Opt | 0,5 | 50 | 1762 | 106 |
| 7 | 30 | MX1T/D | 0,5 | Or-Opt | 0,5 | 100 | 1578 | 70 |
| 8 | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | 1467 | 29 |
| 9 | 100 | MX1T/D | 0,8 | 1-Opt | 0,05 | 50 | 1515 | 17 |
| 10 | 50 | IX | 0,8 | Or-Opt | 0,2 | 70 | 1612 | 83 |
| 11 | 50 | PMX | 0,5 | Or-Opt | 0,05 | 50 | 1553 | 72 |
| 12 | 30 | PMX | 0,7 | 2-Opt | 0,05 | 70 | 1747 | 126 |
| 13 | 50 | MX1T/D | 0,7 | 1-Opt | 0,5 | 70 | 1484 | 25 |
| 14 | 30 | PMX | 0,8 | 1-Opt | 0,2 | 100 | 1683 | 205 |
| 15 | 30 | MX1T/D | 0,7 | Or-Opt | 0,2 | 50 | 1464 | 35 |
| 16 | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | 1474 | 33 |
| Moyenne | | | | | | | 1603 | 75 |
| Effets | | | | | | | | |
| Y_1 | 50 | PMX | 0,7 | 1-Opt | 0,2 | 50 | | |
| | -9,38 | 29,27 | 35,83 | -22,68 | -28,33 | -29,02 | | |
| | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | | |
| | -0,93 | -72,28 | -5,15 | -33,78 | 19,45 | -39,50 | | |
| | 100 | IX | 0,5 | 2-Opt | 0,5 | 100 | | |
| | 11,23 | 115,28 | -25,52 | 90,23 | -10,58 | 108,03 | | |
| Y_2 | 50 | PMX | 0,7 | 1-Opt | 0,2 | 50 | | |
| | -16,83 | 40,73 | 18,33 | 7,88 | 9,48 | -17,38 | | |
| | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | | |
| | 10,95 | -40,35 | -2,00 | -3,82 | 0,03 | -18,10 | | |
| | 100 | IX | 0,5 | 2-Opt | 0,5 | 100 | | |
| | -5,08 | 39,98 | -14,33 | -0,22 | -9,52 | 53,58 | | |

TAB. A.4: Plan de criblage du problème R201 statique

| Nb | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | Y_1 | Y_2 |
|---------|--------|--------|-------|--------|--------|--------|-------|-------|
| 1 | 50 | MX1T/D | 0,8 | 2-Opt | 0,05 | 100 | 1305 | 6 |
| 2 | 100 | PMX | 0,8 | Or-Opt | 0,5 | 70 | 1180 | 4 |
| 3 | 100 | IX | 0,7 | Or-Opt | 0,05 | 100 | 1351 | 11 |
| 4 | 30 | IX | 0,5 | 1-Opt | 0,05 | 70 | 1210 | 8 |
| 5 | 100 | MX1T/D | 0,5 | 2-Opt | 0,2 | 70 | 1221 | 4 |
| 6 | 30 | IX | 0,8 | 2-Opt | 0,5 | 50 | 1215 | 5 |
| 7 | 30 | MX1T/D | 0,5 | Or-Opt | 0,5 | 100 | 1264 | 7 |
| 8 | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | 1196 | 6 |
| 9 | 100 | MX1T/D | 0,8 | 1-Opt | 0,05 | 50 | 1190 | 2 |
| 10 | 50 | IX | 0,8 | Or-Opt | 0,2 | 70 | 1178 | 4 |
| 11 | 50 | PMX | 0,5 | Or-Opt | 0,05 | 50 | 1199 | 3 |
| 12 | 30 | PMX | 0,7 | 2-Opt | 0,05 | 70 | 1240 | 5 |
| 13 | 50 | MX1T/D | 0,7 | 1-Opt | 0,5 | 70 | 1185 | 5 |
| 14 | 30 | PMX | 0,8 | 1-Opt | 0,2 | 100 | 1242 | 21 |
| 15 | 30 | MX1T/D | 0,7 | Or-Opt | 0,2 | 50 | 1169 | 4 |
| 16 | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | 1185 | 2 |
| Moyenne | | | | | | | 1221 | 6 |
| Effets | | | | | | | | |
| Y_1 | 50 | PMX | 0,7 | 1-Opt | 0,2 | 50 | | |
| | -4,04 | -5,54 | 15,51 | -13,94 | -18,19 | -27,49 | | |
| | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | | |
| | -5,49 | -6,11 | -9,24 | -5,39 | 13,91 | -21,16 | | |
| | 100 | IX | 0,5 | 2-Opt | 0,5 | 100 | | |
| | 15,01 | 17,76 | 2,96 | 24,71 | -9,64 | 69,81 | | |
| Y_2 | 50 | PMX | 0,7 | 1-Opt | 0,2 | 50 | | |
| | -1,58 | 1,98 | 0,07 | 3,03 | 2,18 | -2,33 | | |
| | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | | |
| | 1,20 | -1,48 | 0,23 | -1,00 | -0,63 | -1,40 | | |
| | 100 | IX | 0,5 | 2-Opt | 0,5 | 100 | | |
| | -0,83 | 0,98 | -0,52 | -1,03 | -0,93 | 5,13 | | |

TAB. A.5: Plan de criblage du problème RC104 statique

| Nb | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | Y ₁ | Y ₂ |
|----------------|--------|--------|--------|--------|--------|--------|----------------|----------------|
| 1 | 50 | MX1T/D | 0,8 | 2-Opt | 0,05 | 100 | 1723 | 52 |
| 2 | 100 | PMX | 0,8 | Or-Opt | 0,5 | 70 | 1544 | 60 |
| 3 | 100 | IX | 0,7 | Or-Opt | 0,05 | 100 | 1858 | 187 |
| 4 | 30 | IX | 0,5 | 1-Opt | 0,05 | 70 | 1639 | 84 |
| 5 | 100 | MX1T/D | 0,5 | 2-Opt | 0,2 | 70 | 1538 | 15 |
| 6 | 30 | IX | 0,8 | 2-Opt | 0,5 | 50 | 1762 | 106 |
| 7 | 30 | MX1T/D | 0,5 | Or-Opt | 0,5 | 100 | 1578 | 70 |
| 8 | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | 1467 | 29 |
| 9 | 100 | MX1T/D | 0,8 | 1-Opt | 0,05 | 50 | 1515 | 17 |
| 10 | 50 | IX | 0,8 | Or-Opt | 0,2 | 70 | 1612 | 83 |
| 11 | 50 | PMX | 0,5 | Or-Opt | 0,05 | 50 | 1553 | 72 |
| 12 | 30 | PMX | 0,7 | 2-Opt | 0,05 | 70 | 1747 | 126 |
| 13 | 50 | MX1T/D | 0,7 | 1-Opt | 0,5 | 70 | 1484 | 25 |
| 14 | 30 | PMX | 0,8 | 1-Opt | 0,2 | 100 | 1683 | 205 |
| 15 | 30 | MX1T/D | 0,7 | Or-Opt | 0,2 | 50 | 1464 | 35 |
| 16 | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | 1474 | 33 |
| Moyenne | | | | | | | 1603 | 75 |
| Effets | | | | | | | | |
| Y ₁ | 50 | PMX | 0,7 | 1-Opt | 0,2 | 50 | | |
| | -9,38 | 29,27 | 35,83 | -22,68 | -28,33 | -29,02 | | |
| | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | | |
| | -0,93 | -72,28 | -5,15 | -33,78 | 19,45 | -39,50 | | |
| | 100 | IX | 0,5 | 2-Opt | 0,5 | 100 | | |
| | 11,23 | 115,28 | -25,52 | 90,23 | -10,58 | 108,03 | | |
| Y ₂ | 50 | PMX | 0,7 | 1-Opt | 0,2 | 50 | | |
| | -16,83 | 40,73 | 18,33 | 7,88 | 9,48 | -17,38 | | |
| | 30 | MX1T/D | 0,8 | Or-Opt | 0,05 | 70 | | |
| | 10,95 | -40,35 | -2,00 | -3,82 | 0,03 | -18,10 | | |
| | 100 | IX | 0,5 | 2-Opt | 0,5 | 100 | | |
| | -5,08 | 39,98 | -14,33 | -0,22 | -9,52 | 53,58 | | |

TAB. A.6: Plan de criblage du problème RC201 statique

A.2 Plans d'expériences complets vs. fractionnaires

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Elit. | I ₁₂ | I ₁₃ | I ₁₄ | I ₁₅ | I ₁₆ | I ₂₃ | I ₂₄ | I ₂₅ | I ₂₆ | I ₃₄ | I ₃₅ | I ₃₆ | I ₄₅ | I ₄₆ | I ₅₆ | Y ₁ | | Y ₂ | | σ ₂ |
|----|--------------------------|----------------------|------------------------|----------------------|------------------------|-------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|----------------|------|----------------|----------------|----------------|
| | | | | | | | | | | | | | | | | | | | | | | D | G | Y ₁ | Y ₂ | |
| 1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 835 | 8 | 5 | 3 |
| 2 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 835 | 12 | 7 | 10 |
| 3 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 936 | 793 | 99 | 1045 |
| 4 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 927 | 248 | 84 | 156 |
| 5 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 836 | 11 | 5 | 10 |
| 6 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 836 | 5 | 8 | 1 |
| 7 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1009 | 490 | 64 | 414 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 921 | 256 | 67 | 156 |
| 9 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 830 | 8 | 2 | 2 |
| 10 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 829 | 8 | 0 | 3 |
| 11 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 945 | 1236 | 101 | 1373 |
| 12 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 899 | 685 | 46 | 344 |
| 13 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 829 | 7 | 0 | 2 |
| 14 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 829 | 6 | 0 | 1 |
| 15 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 962 | 1203 | 73 | 1284 |
| 16 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 945 | 432 | 65 | 235 |
| 17 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 840 | 10 | 18 | 3 |
| 18 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 831 | 7 | 2 | 2 |
| 19 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 907 | 443 | 116 | 217 |
| 20 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 987 | 1154 | 108 | 1749 |
| 21 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 832 | 6 | 3 | 1 |
| 22 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 834 | 5 | 5 | 1 |
| 23 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1024 | 2089 | 104 | 2316 |
| 24 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 915 | 339 | 77 | 249 |
| 25 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 829 | 8 | 0 | 2 |
| 26 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 829 | 6 | 0 | 1 |
| 27 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1022 | 1088 | 97 | 724 |
| 28 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 902 | 749 | 127 | 359 |
| 29 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 830 | 6 | 2 | 2 |
| 30 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 829 | 5 | 0 | 1 |
| 31 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 977 | 613 | 47 | 494 |
| 32 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 949 | 480 | 90 | 286 |
| 33 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 838 | 7 | 6 | 1 |
| 34 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 830 | 6 | 1 | 1 |
| 35 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1041 | 2427 | 89 | 3229 |
| 36 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 966 | 1121 | 93 | 812 |
| 37 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 838 | 6 | 4 | 1 |
| 38 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 835 | 5 | 7 | 0 |
| 39 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 949 | 522 | 96 | 266 |
| 40 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1002 | 323 | 107 | 198 |
| 41 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 829 | 6 | 0 | 1 |
| 42 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 829 | 6 | 0 | 1 |
| 43 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1015 | 1818 | 64 | 837 |
| 44 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 944 | 703 | 92 | 541 |
| 45 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 830 | 6 | 2 | 1 |
| 46 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 830 | 5 | 2 | 1 |
| 47 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 932 | 776 | 89 | 172 |
| 48 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 914 | 1083 | 79 | 593 |
| 49 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 832 | 6 | 4 | 1 |
| 50 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 833 | 6 | 3 | 1 |
| 51 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 904 | 396 | 37 | 215 |
| 52 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 972 | 1478 | 167 | 1409 |
| 53 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 833 | 6 | 3 | 1 |

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Elit. | I ₁₂ | I ₁₃ | I ₁₄ | I ₁₅ | I ₁₆ | I ₂₃ | I ₂₄ | I ₂₅ | I ₂₆ | I ₃₄ | I ₃₅ | I ₃₆ | I ₄₅ | I ₄₆ | I ₅₆ | Y ₁ | | σ ₂ | | | |
|------------------|--------------------------|----------------------|------------------------|----------------------|------------------------|-------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|----------------|------|----------------|----------------|-----|--|
| | | | | | | | | | | | | | | | | | | | | | | D | G | Y ₁ | Y ₂ | | |
| 54 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | 834 | 5 | 5 | 1 | | |
| 55 | -1 | 1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | 997 | 1267 | 70 | 1327 | | |
| 56 | 1 | 1 | 1 | -1 | 1 | 1 | 1 | -1 | 1 | 1 | -1 | 1 | -1 | 1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | 940 | 737 | 87 | 653 | | |
| 57 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | 830 | 7 | 1 | 1 | | |
| 58 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | 829 | 6 | 0 | 0 | | |
| 59 | -1 | 1 | -1 | -1 | 1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 946 | 1815 | 65 | 1512 | | |
| 60 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | -1 | 1 | 1 | 1 | -1 | 1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | 1 | 942 | 1696 | 132 | 1421 | | |
| 61 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | 830 | 6 | 1 | 0 | | |
| 62 | 1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | -1 | 1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1012 | 1282 | 88 | 983 | | |
| 63 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | 1 | 1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 901 | 930 | 91 | 704 | | |
| 64 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | | |
| N. | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | | | | | | | | | | | | | | | | | | | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | | | | | | | | | | | | | | | | | | | | |
| M. | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| σ _{eff} | | | | | | | | | | | | | | | | | | | | | | | 894 | 483 | 45 | 411 | |
| E. | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Y ₁ | -9 | 62 | 2 | -4 | -0 | 2 | -8 | -3 | -4 | -0 | 2 | 1 | -2 | 0 | 2 | -2 | 3 | -4 | 3 | -2 | -5 | | | | | | |
| Y ₂ | -92 | 476 | -79 | 39 | 38 | 95 | -91 | -23 | -5 | 47 | 22 | -78 | 39 | 38 | 95 | -15 | 45 | -63 | -16 | 18 | -12 | | | | | | |
| C. | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Y ₁ | 50 | MX1 | 0,5 | OrOpt | 0,5 | 50% | | | | | | | | | | | | | | | | | | | | | |
| Y ₂ | 50 | MX1 | 0,7 | 1Opt | 0,2 | 50% | | | | | | | | | | | | | | | | | 829 | | | | |

Tab. A.7: Plan complet du problème C109 statique

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Elit. | Y ₁ D | Y ₂ G | σ ₁ Y ₁ | σ ₂ Y ₂ |
|------------------|--------------------------|----------------------|------------------------|----------------------|------------------------|-------------------------|---------------------|---------------------|----------------------------------|----------------------------------|
| | | | | | | | | | | |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 932 | 776 | 89 | 172 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 830 | 6 | 2 | 2 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 829 | 6 | 0 | 0 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 904 | 396 | 37 | 215 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 835 | 5 | 7 | 0 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 899 | 685 | 46 | 344 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 835 | 8 | 5 | 3 |
| Niveaux | | | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | | | |
| Moyenne | | | | | | | 872 | 278 | 33 | 123 |
| σ _{eff} | | | | | | | | | 12 | 44 |
| Contrastes | | | | | | | | | | |
| Y ₁ | -3 | 40 | 6 | 0 | -3 | 3 | | | | |
| Y ₂ | -19 | 271 | 4 | 90 | -91 | 18 | | | | |
| Meilleures | | | | | | | | | | |
| Y ₁ | 50 | MX1 | 0,5 | 1Opt | 0,5 | 50% | 831 | | | |
| Y ₂ | 50 | MX1 | 0,5 | 1Opt | 0,5 | 50% | | 7 | | |

Tab. A.8: Plan fractionnaire du problème C109 statique

A.2. Plans d'expériences complets vs. fractionnaires

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Eflit. | I ₁₂ | I ₁₃ | I ₁₄ | I ₁₅ | I ₁₆ | I ₂₃ | I ₂₄ | I ₂₅ | I ₂₆ | I ₃₄ | I ₃₅ | I ₃₆ | I ₄₅ | I ₄₆ | I ₅₆ | Y ₁ | | Y ₂ | | σ ₁ Y ₁ | σ ₂ Y ₂ |
|----|--------------------------|----------------------|------------------------|----------------------|------------------------|--------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|----------------|------|----------------|-----|----------------------------------|----------------------------------|
| | | | | | | | | | | | | | | | | | | | | | | D | G | D | G | | |
| 1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1426 | 668 | 225 | 796 | |
| 2 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1405 | 296 | 122 | 76 | |
| 3 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1485 | 570 | 85 | 94 | |
| 4 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1406 | 604 | 137 | 280 | |
| 5 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1394 | 288 | 147 | 78 | |
| 6 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1534 | 196 | 168 | 137 | |
| 7 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1343 | 419 | 270 | 137 | |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1320 | 206 | 189 | 81 | |
| 9 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1066 | 349 | 86 | 180 | |
| 10 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1010 | 233 | 207 | 62 | |
| 11 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1118 | 804 | 171 | 406 | |
| 12 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1181 | 563 | 150 | 401 | |
| 13 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1071 | 580 | 107 | 196 | |
| 14 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1035 | 299 | 194 | 79 | |
| 15 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1216 | 827 | 157 | 513 | |
| 16 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1006 | 314 | 214 | 222 | |
| 17 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1369 | 460 | 196 | 235 | |
| 18 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1125 | 200 | 210 | 26 | |
| 19 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1416 | 780 | 128 | 336 | |
| 20 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1281 | 714 | 82 | 617 | |
| 21 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1400 | 212 | 175 | 111 | |
| 22 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1430 | 729 | 125 | 1062 | |
| 23 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1364 | 749 | 229 | 683 | |
| 24 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1223 | 230 | 195 | 93 | |
| 25 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1149 | 488 | 212 | 155 | |
| 26 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1173 | 328 | 126 | 110 | |
| 27 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1075 | 417 | 181 | 207 | |
| 28 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1036 | 590 | 152 | 435 | |
| 29 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1182 | 422 | 109 | 69 | |
| 30 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1192 | 417 | 241 | 358 | |
| 31 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1201 | 633 | 194 | 223 | |
| 32 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1094 | 371 | 179 | 193 | |
| 33 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1453 | 381 | 101 | 141 | |
| 34 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1477 | 693 | 108 | 299 | |
| 35 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1376 | 516 | 66 | 148 | |
| 36 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1341 | 1229 | 302 | 813 | |
| 37 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1535 | 740 | 163 | 692 | |
| 38 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1387 | 450 | 96 | 257 | |
| 39 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1451 | 932 | 163 | 513 | |
| 40 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1287 | 406 | 111 | 193 | |
| 41 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1104 | 323 | 93 | 90 | |
| 42 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1110 | 565 | 171 | 458 | |
| 43 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1122 | 964 | 178 | 1165 | |
| 44 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1167 | 732 | 219 | 837 | |
| 45 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1107 | 501 | 168 | 163 | |
| 46 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1112 | 291 | 97 | 164 | |
| 47 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1230 | 606 | 96 | 300 | |
| 48 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1292 | 810 | 264 | 554 | |
| 49 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1350 | 460 | 173 | 172 | |
| 50 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1352 | 432 | 78 | 207 | |
| 51 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1269 | 488 | 99 | 194 | |
| 52 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1261 | 497 | 247 | 489 | |
| 53 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1405 | 599 | 70 | 602 | |
| 54 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1391 | 223 | 100 | 60 | |
| 55 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1292 | 480 | 167 | 184 | |
| 56 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1220 | 336 | 161 | 128 | |
| 57 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1289 | 756 | 167 | 322 | |
| 58 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1132 | 228 | 199 | 65 | |

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Elt. | I ₁₂ | I ₁₃ | I ₁₄ | I ₁₅ | I ₁₆ | I ₂₃ | I ₂₄ | I ₂₅ | I ₂₆ | I ₃₄ | I ₃₅ | I ₃₆ | I ₄₅ | I ₄₆ | I ₅₆ | Y ₁ | | Y ₂ | | σ ₁ Y ₁ | σ ₂ Y ₂ |
|------------------|--------------------------|----------------------|------------------------|----------------------|------------------------|------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|----------------|-----|----------------|-----|----------------------------------|----------------------------------|
| | | | | | | | | | | | | | | | | | | | | | | D | G | D | G | | |
| 59 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1114 | 850 | 172 | 749 | | |
| 60 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1182 | 442 | 190 | 187 | | |
| 61 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1132 | 588 | 118 | 471 | | |
| 62 | 1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1182 | 470 | 67 | 319 | | |
| 63 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1251 | 482 | 110 | 141 | | |
| 64 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1165 | 550 | 147 | 256 | | |
| N. | 30 | MXI | 0,5 | 1Opt | 0,2 | 50% | | | | | | | | | | | | | | | | | | | | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | | | | | | | | | | | | | | | | | | | | |
| M. | | | | | | | | | | | | | | | | | | | | | | 1254 | 515 | 157 | 20 | 312 | |
| σ _{eff} | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| E. | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Y ₁ | -19 | -11 | 10 | -114 | -14 | 13 | -7 | -2 | 8 | -6 | 6 | -6 | 24 | -14 | -5 | 4 | 7 | 1 | 32 | 15 | -4 | | | | | | |
| Y ₂ | -58 | 82 | -35 | 10 | -27 | 48 | -2 | -29 | -17 | -8 | 17 | -40 | 15 | -32 | 0 | 21 | 15 | 1 | 4 | 0 | -43 | | | | | | |
| C. | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Y ₁ | 50 | PMX | 0,5 | OrOpt | 0,5 | 50% | | | | | | | | | | | | | | | | | | | | | |
| Y ₂ | 50 | MXI | 0,7 | 1Opt | 0,5 | 50% | | | | | | | | | | | | | | | | | | | | | |

Tab. A.9: Plan complet du problème C204 statique

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Elt. | Y ₁ D | Y ₂ G | σ ₁ Y ₁ | σ ₂ Y ₂ |
|------------------|--------------------------|----------------------|------------------------|----------------------|------------------------|------------------------|---------------------|---------------------|----------------------------------|----------------------------------|
| | | | | | | | | | | |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 1230 | 606 | 96 | 300 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 1182 | 422 | 109 | 69 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 1132 | 228 | 199 | 65 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 1269 | 488 | 99 | 194 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 1387 | 450 | 96 | 257 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 1181 | 563 | 150 | 401 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 1426 | 668 | 225 | 796 |
| Niveaux | | | | | | | | | | |
| -1 | 30 | MXI | 0,5 | 1Opt | 0,2 | 50% | | | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | | | |
| Moyenne | | | | | | | 1254 | 457 | 146 | 272 |
| σ _{eff} | | | | | | | | | 52 | 96 |
| Contrastes | | | | | | | | | | |
| Y ₁ | -23 | -28 | 2 | -72 | -52 | 1 | | | | |
| Y ₂ | -89 | 15 | -30 | -2 | -115 | -14 | | | | |
| Meilleures | | | | | | | | | | |
| Y ₁ | 50 | PMX | 0,5 | OrOpt | 0,5 | 50% | 1036 | | | |
| Y ₂ | 50 | MXI | 0,7 | OrOpt | 0,5 | 70% | | 470 | | |

Tab. A.10: Plan fractionnaire du problème C204 statique

A.2. Plans d'expériences complets vs. fractionnaires

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Elit. | I ₁₂ | I ₁₃ | I ₁₄ | I ₁₅ | I ₁₆ | I ₂₃ | I ₂₄ | I ₂₅ | I ₂₆ | I ₃₄ | I ₃₅ | I ₃₆ | I ₄₅ | I ₄₆ | I ₅₆ | Y ₁ | | Y ₂ | | σ ₁ Y ₁ | σ ₂ Y ₂ |
|----|--------------------------|----------------------|------------------------|----------------------|------------------------|-------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|----------------|------|----------------|----|----------------------------------|----------------------------------|
| | | | | | | | | | | | | | | | | | | | | | | D | G | D | G | | |
| 1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1576 | 3 | 18 | 3 | |
| 2 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1575 | 4 | 8 | 3 | |
| 3 | -1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1584 | 31 | 14 | 49 | |
| 4 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1588 | 6 | 10 | 6 | |
| 5 | -1 | -1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1582 | 3 | 19 | 1 | |
| 6 | 1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1583 | 4 | 15 | 0 | |
| 7 | -1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1588 | 10 | 12 | 7 | |
| 8 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1583 | 16 | 26 | 8 | |
| 9 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1592 | 4 | 10 | 1 | |
| 10 | 1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1595 | 5 | 21 | 2 | |
| 11 | -1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1589 | 3 | 8 | 1 | |
| 12 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1592 | 10 | 15 | 10 | |
| 13 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1591 | 3 | 9 | 1 | |
| 14 | 1 | -1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1587 | 2 | 23 | 0 | |
| 15 | -1 | 1 | 1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1593 | 7 | 25 | 4 | |
| 16 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1618 | 12 | 23 | 10 | |
| 17 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1580 | 3 | 18 | 3 | |
| 18 | 1 | -1 | -1 | -1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1578 | 2 | 11 | 1 | |
| 19 | -1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1582 | 13 | 22 | 14 | |
| 20 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1564 | 7 | 10 | 7 | |
| 21 | -1 | -1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1572 | 6 | 9 | 4 | |
| 22 | 1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1587 | 3 | 18 | 1 | |
| 23 | -1 | 1 | 1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1565 | 11 | 9 | 10 | |
| 24 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1564 | 13 | 13 | 4 | |
| 25 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1594 | 6 | 21 | 2 | |
| 26 | 1 | -1 | -1 | -1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1584 | 4 | 14 | 1 | |
| 27 | -1 | 1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1582 | 14 | 18 | 7 | |
| 28 | 1 | -1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1610 | 1 | 15 | 1 | |
| 29 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1578 | 4 | 14 | 2 | |
| 30 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1588 | 4 | 20 | 2 | |
| 31 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1576 | 6 | 14 | 3 | |
| 32 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1591 | 6 | 18 | 3 | |
| 33 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1589 | 5 | 6 | 2 | |
| 34 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1589 | 2 | 7 | 2 | |
| 35 | -1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1598 | 16 | 34 | 23 | |
| 36 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1600 | 16 | 15 | 14 | |
| 37 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1572 | 3 | 18 | 1 | |
| 38 | 1 | -1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1582 | 4 | 7 | 2 | |
| 39 | -1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1591 | 1 | 21 | 1 | |
| 40 | 1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1594 | 13 | 18 | 14 | |
| 41 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1593 | 2 | 19 | 2 | |
| 42 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1574 | 3 | 13 | 1 | |
| 43 | -1 | 1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1614 | 20 | 20 | 23 | |
| 44 | 1 | -1 | -1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1597 | 4 | 10 | 3 | |
| 45 | -1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1574 | 3 | 22 | 1 | |
| 46 | 1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1589 | 4 | 19 | 1 | |
| 47 | -1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1605 | 10 | 21 | 11 | |
| 48 | 1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1598 | 6 | 22 | 3 | |
| 49 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1576 | 4 | 12 | 1 | |
| 50 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1581 | 2 | 16 | 1 | |
| 51 | -1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1594 | 8 | 14 | 9 | |
| 52 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1590 | 14 | 13 | 8 | |
| 53 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1582 | 2 | 23 | 1 | |
| 54 | 1 | -1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1574 | 2 | 16 | 1 | |
| 55 | -1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1567 | 8 | 17 | 5 | |
| 56 | 1 | -1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1581 | 7 | 10 | 8 | |
| 57 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1576 | 2 | 9 | 2 | |
| 58 | 1 | -1 | -1 | -1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1584 | 4 | 20 | 2 | |

A suivre →

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Elit. | I ₁₂ | I ₁₃ | I ₁₄ | I ₁₅ | I ₁₆ | I ₂₃ | I ₂₄ | I ₂₅ | I ₂₆ | I ₃₄ | I ₃₅ | I ₃₆ | I ₄₅ | I ₄₆ | I ₅₆ | Y ₁ D | Y ₂ G | σ ₁ Y ₁ | σ ₂ Y ₂ | | | | | | | | | | | | | | | | | | | | |
|---------------------|--------------------------|----------------------|------------------------|----------------------|------------------------|-------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|---------------------|---------------------|----------------------------------|----------------------------------|--|--|--|--|--|--|--|--|--|--|--|--|--|------|--|---|--|--|--|--|
| 59 | -1 | 1 | -1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1599 | 9 | 25 | 5 | | | | | | | | | | | | | | | | | | | | |
| 60 | 1 | 1 | -1 | 1 | 1 | 1 | 1 | -1 | 1 | 1 | -1 | -1 | 1 | 1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | 1600 | 12 | 15 | 8 | | | | | | | | | | | | | | | | | | | | |
| 61 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1582 | 4 | 13 | 2 | | | | | | | | | | | | | | | | | | | | |
| 62 | 1 | -1 | 1 | 1 | 1 | 1 | -1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1585 | 3 | 17 | 1 | | | | | | | | | | | | | | | | | | | | |
| 63 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1583 | 9 | 2 | 5 | | | | | | | | | | | | | | | | | | | | |
| 64 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1587 | 6 | 11 | 4 | | | | | | | | | | | | | | | | | | | | |
| N. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | IOpt | 0,2 | 50% | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| M. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| σ _{eff} et | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| E. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Y ₁ | 1 | 3 | -2 | 4 | -4 | 1 | 0 | 2 | 1 | 1 | -1 | -1 | 2 | -2 | 3 | 0 | -2 | -1 | 1 | -2 | 0 | | | | | | | | | | | | | | | | | | | | | | | | |
| Y ₂ | -1 | 3 | -1 | -1 | -1 | 0 | 0 | 1 | 0 | 0 | 0 | -1 | -1 | -1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | | | | | | | | | | | | | | | | | | | | | | | |
| C. | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Y ₁ | 30 | MX1 | 0,7 | IOpt | 0,5 | 50% | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 1564 | | | | | | |
| Y ₂ | 50 | MX1 | 0,7 | OrOpt | 0,5 | 70% | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | 1 | | | | |

Tab. A.11: Plan complet du problème R101 statique

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Elit. | Y ₁ D | Y ₂ G | σ ₁ Y ₁ | σ ₂ Y ₂ | |
|---------------------|--------------------------|----------------------|------------------------|----------------------|------------------------|-------------------------|---------------------|---------------------|----------------------------------|----------------------------------|--|
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 1564 | 13 | 13 | 4 | |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 1605 | 10 | 21 | 11 | |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 1578 | 4 | 14 | 2 | |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 1584 | 4 | 20 | 2 | |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 1594 | 8 | 14 | 9 | |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 1582 | 4 | 7 | 2 | |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 1592 | 10 | 15 | 10 | |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 1576 | 3 | 18 | 3 | |
| Niveaux | | | | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | IOpt | 0,2 | 50% | | | | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | | | | |
| Moyenne | | | | | | | | | | | |
| σ _{eff} et | | | | | | | | | | | |
| Contrastes | | | | | | | | | | | |
| Y ₁ | -4 | 4 | -2 | 5 | -4 | 7 | | | | | |
| Y ₂ | 1 | 4 | 1 | 0 | 0 | -1 | | | | | |
| Meilleures | | | | | | | | | | | |
| Y ₁ | 50 | MX1 | 0,7 | IOpt | 0,5 | 50% | 1587 | | | | |
| Y ₂ | 30 | MX1 | 0,5 | IOpt | 0,2 | 70% | | 5 | | | |

Tab. A.12: Plan fractionnaire du problème R101 statique

A.2. Plans d'expériences complets vs. fractionnaires

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Elit. | I ₁₂ | I ₁₃ | I ₁₄ | I ₁₅ | I ₁₆ | I ₂₃ | I ₂₄ | I ₂₅ | I ₂₆ | I ₃₄ | I ₃₅ | I ₃₆ | I ₄₅ | I ₄₆ | I ₅₆ | Y ₁ | Y ₂ | σ ₁ | σ ₂ | |
|----|--------------------------|----------------------|------------------------|----------------------|------------------------|-------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|----------------|----------------|----------------|----------------|----|
| | | | | | | | | | | | | | | | | | | | | | | D | G | Y ₁ | Y ₂ | |
| 1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1322 | 18 | 28 | 13 |
| 2 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1360 | 15 | 35 | 5 |
| 3 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1456 | 132 | 63 | 86 |
| 4 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1464 | 53 | 82 | 20 |
| 5 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1301 | 11 | 31 | 4 |
| 6 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1314 | 10 | 21 | 3 |
| 7 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1409 | 59 | 31 | 14 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1395 | 38 | 70 | 12 |
| 9 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1328 | 16 | 14 | 4 |
| 10 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1290 | 10 | 14 | 3 |
| 11 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1417 | 50 | 48 | 15 |
| 12 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1369 | 26 | 40 | 10 |
| 13 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1328 | 8 | 41 | 2 |
| 14 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1288 | 9 | 31 | 3 |
| 15 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1391 | 47 | 33 | 23 |
| 16 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1425 | 25 | 57 | 11 |
| 17 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1330 | 17 | 39 | 9 |
| 18 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1288 | 10 | 49 | 5 |
| 19 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1380 | 36 | 44 | 15 |
| 20 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1365 | 43 | 24 | 16 |
| 21 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1316 | 9 | 23 | 3 |
| 22 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1341 | 7 | 24 | 3 |
| 23 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1394 | 68 | 35 | 44 |
| 24 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1432 | 58 | 58 | 15 |
| 25 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1305 | 14 | 35 | 8 |
| 26 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1276 | 13 | 18 | 6 |
| 27 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1388 | 39 | 52 | 21 |
| 28 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1432 | 31 | 102 | 24 |
| 29 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1293 | 16 | 25 | 5 |
| 30 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1299 | 12 | 32 | 3 |
| 31 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1461 | 79 | 59 | 39 |
| 32 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1428 | 64 | 16 | 50 |
| 33 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1335 | 13 | 38 | 4 |
| 34 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1315 | 10 | 28 | 2 |
| 35 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1433 | 100 | 70 | 60 |
| 36 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1398 | 61 | 70 | 40 |
| 37 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1314 | 10 | 39 | 1 |
| 38 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1324 | 9 | 47 | 3 |
| 39 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1461 | 69 | 85 | 57 |
| 40 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1398 | 52 | 49 | 43 |
| 41 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1313 | 14 | 49 | 7 |
| 42 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1276 | 11 | 11 | 3 |
| 43 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1489 | 67 | 50 | 36 |
| 44 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1415 | 60 | 79 | 35 |
| 45 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1308 | 12 | 30 | 3 |
| 46 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1296 | 10 | 26 | 1 |
| 47 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1435 | 130 | 78 | 93 |
| 48 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1448 | 20 | 49 | 10 |
| 49 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1363 | 13 | 10 | 4 |
| 50 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1323 | 9 | 21 | 3 |
| 51 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1443 | 99 | 33 | 66 |
| 52 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1398 | 60 | 31 | 17 |
| 53 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1308 | 11 | 24 | 1 |
| 54 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1321 | 9 | 53 | 2 |
| 55 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1443 | 56 | 47 | 17 |
| 56 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1437 | 89 | 18 | 41 |
| 57 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1305 | 15 | 22 | 3 |
| 58 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1303 | 12 | 24 | 4 |

A.2. Plans d'expériences complets vs. fractionnaires

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Elit. | I ₁₂ | I ₁₃ | I ₁₄ | I ₁₅ | I ₁₆ | I ₂₃ | I ₂₄ | I ₂₅ | I ₂₆ | I ₃₄ | I ₃₅ | I ₃₆ | I ₄₅ | I ₄₆ | I ₅₆ | Y ₁ | | Y ₂ | | σ ₁ Y ₁ | σ ₂ Y ₂ |
|----|--------------------------|----------------------|------------------------|----------------------|------------------------|-------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|----------------|------|----------------|----|----------------------------------|----------------------------------|
| | | | | | | | | | | | | | | | | | | | | | | D | G | D | G | | |
| 1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1218 | 6 | 38 | 4 | |
| 2 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1192 | 3 | 40 | 1 | |
| 3 | -1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1180 | 5 | 40 | 1 | |
| 4 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1168 | 3 | 37 | 1 | |
| 5 | -1 | -1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1218 | 3 | 58 | 1 | |
| 6 | 1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1199 | 4 | 41 | 4 | |
| 7 | -1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1221 | 4 | 58 | 4 | |
| 8 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1173 | 4 | 20 | 1 | |
| 9 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1165 | 3 | 23 | 1 | |
| 10 | 1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1173 | 4 | 31 | 2 | |
| 11 | -1 | 1 | 1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1184 | 6 | 13 | 4 | |
| 12 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1184 | 3 | 42 | 1 | |
| 13 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1147 | 4 | 21 | 1 | |
| 14 | 1 | -1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1157 | 2 | 38 | 0 | |
| 15 | -1 | 1 | 1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1165 | 4 | 10 | 3 | |
| 16 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1203 | 5 | 28 | 1 | |
| 17 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1188 | 5 | 43 | 2 | |
| 18 | 1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1186 | 4 | 54 | 1 | |
| 19 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1153 | 10 | 6 | 9 | |
| 20 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1166 | 6 | 33 | 1 | |
| 21 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1204 | 7 | 52 | 7 | |
| 22 | 1 | -1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1180 | 4 | 28 | 1 | |
| 23 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1209 | 6 | 31 | 4 | |
| 24 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1175 | 5 | 41 | 3 | |
| 25 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1178 | 3 | 27 | 1 | |
| 26 | 1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1181 | 5 | 41 | 3 | |
| 27 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1193 | 5 | 40 | 2 | |
| 28 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1170 | 3 | 26 | 2 | |
| 29 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1195 | 3 | 30 | 2 | |
| 30 | 1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1151 | 4 | 18 | 1 | |
| 31 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1164 | 4 | 17 | 3 | |
| 32 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1183 | 3 | 6 | 2 | |
| 33 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1192 | 8 | 23 | 8 | |
| 34 | 1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1186 | 6 | 23 | 2 | |
| 35 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1228 | 10 | 37 | 11 | |
| 36 | 1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1182 | 10 | 26 | 8 | |
| 37 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1191 | 6 | 34 | 4 | |
| 38 | 1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1184 | 4 | 27 | 2 | |
| 39 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1212 | 5 | 43 | 2 | |
| 40 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1203 | 5 | 24 | 1 | |
| 41 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1182 | 5 | 30 | 3 | |
| 42 | 1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1172 | 3 | 30 | 2 | |
| 43 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1154 | 5 | 26 | 3 | |
| 44 | 1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1172 | 3 | 29 | 1 | |
| 45 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1182 | 3 | 29 | 1 | |
| 46 | 1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1168 | 4 | 41 | 1 | |
| 47 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1182 | 6 | 32 | 2 | |
| 48 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1174 | 7 | 35 | 5 | |
| 49 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1210 | 4 | 41 | 1 | |
| 50 | 1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1184 | 3 | 57 | 1 | |
| 51 | -1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1154 | 10 | 28 | 5 | |
| 52 | 1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1188 | 4 | 53 | 1 | |
| 53 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1185 | 4 | 32 | 2 | |
| 54 | 1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1194 | 4 | 57 | 1 | |
| 55 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1165 | 6 | 21 | 4 | |
| 56 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1201 | 4 | 33 | 2 | |
| 57 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1177 | 4 | 35 | 1 | |
| 58 | 1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1163 | 6 | 30 | 2 | |

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Elit. | I ₁₂ | I ₁₃ | I ₁₄ | I ₁₅ | I ₁₆ | I ₂₃ | I ₂₄ | I ₂₅ | I ₂₆ | I ₃₄ | I ₃₅ | I ₃₆ | I ₄₅ | I ₄₆ | I ₅₆ | Y ₁ D | Y ₂ G | σ ₁ Y ₁ | σ ₂ Y ₂ | |
|------------------|--------------------------|----------------------|------------------------|----------------------|------------------------|-------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|---------------------|---------------------|----------------------------------|----------------------------------|---|
| 59 | -1 | 1 | -1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1191 | 7 | 19 | 5 | |
| 60 | 1 | 1 | -1 | 1 | 1 | 1 | -1 | -1 | 1 | 1 | -1 | -1 | 1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1144 | 6 | 22 | 3 | |
| 61 | -1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1170 | 5 | 34 | 1 | |
| 62 | 1 | -1 | 1 | 1 | 1 | 1 | -1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1155 | 3 | 18 | 2 | |
| 63 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1184 | 7 | 31 | 4 | |
| 64 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1169 | 6 | 25 | 3 | |
| N. | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | | | | | | | | | | | | | | | | | | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | | | | | | | | | | | | | | | | | | | |
| M. | | | | | | | | | | | | | | | | | | | | | | | | | | |
| σ _{eff} | | | | | | | | | | | | | | | | | | | | | | | 1182 | 5 | 32 | 3 |
| E. | | | | | | | | | | | | | | | | | | | | | | | | 4 | 0 | |
| Y ₁ | -4 | -1 | 2 | -9 | -3 | 0 | 1 | 0 | 1 | 0 | 0 | 4 | 4 | -2 | 1 | -3 | 0 | 0 | 0 | 3 | -1 | -1 | | | | |
| Y ₂ | -1 | 1 | 0 | -1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | | | | |
| C. | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Y ₁ | 50 | PMX | 0,5 | OrOpt | 0,5 | 70% | | | | | | | | | | | | | | | | | 1144 | | | |
| Y ₂ | 50 | MX1 | 0,7 | OrOpt | 0,2 | 50% | | | | | | | | | | | | | | | | | | 2 | | |

Tab. A.15: Plan complet du problème RC104 statique

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Elit. | Y ₁ D | Y ₂ G | σ ₁ Y ₁ | σ ₂ Y ₂ |
|------------------|--------------------------|----------------------|------------------------|----------------------|------------------------|-------------------------|---------------------|---------------------|----------------------------------|----------------------------------|
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 1175 | 5 | 41 | 3 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 1182 | 6 | 32 | 2 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 1195 | 3 | 30 | 2 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 1163 | 6 | 30 | 2 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 1154 | 10 | 28 | 5 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 1184 | 4 | 27 | 2 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 1184 | 3 | 42 | 1 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 1218 | 6 | 38 | 4 |
| Niveaux | | | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | | | |
| Moyenne | | | | | | | 1182 | 5 | 34 | 3 |
| σ _{eff} | | | | | | | | | | |
| Contrastes | | | | | | | | | | |
| Y ₁ | -5 | -8 | 2 | -1 | -10 | -11 | | | | |
| Y ₂ | -1 | 1 | -1 | -1 | 1 | 1 | | | | |
| Meilleures | | | | | | | | | | |
| Y ₁ | 50 | PMX | 0,5 | OrOpt | 0,5 | 70% | 1144 | | | |
| Y ₂ | 50 | MX1 | 0,7 | OrOpt | 0,2 | 50% | | 2 | | |

Tab. A.16: Plan fractionnaire du problème RC104 statique

A.2. Plans d'expériences complets vs. fractionnaires

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Elit. | I ₁₂ | I ₁₃ | I ₁₄ | I ₁₅ | I ₁₆ | I ₂₃ | I ₂₄ | I ₂₅ | I ₂₆ | I ₃₄ | I ₃₅ | I ₃₆ | I ₄₅ | I ₄₆ | I ₅₆ | Y ₁ | Y ₂ | σ ₁ | σ ₂ | |
|----|--------------------------|----------------------|------------------------|----------------------|------------------------|-------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|----------------|----------------|----------------|----------------|-----|
| | | | | | | | | | | | | | | | | | | | | | | D | G | Y ₁ | Y ₂ | |
| 1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1551 | 49 | 53 | 31 |
| 2 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1487 | 32 | 74 | 7 |
| 3 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1628 | 123 | 65 | 44 |
| 4 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1598 | 98 | 47 | 21 |
| 5 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1541 | 44 | 29 | 27 |
| 6 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1501 | 18 | 26 | 3 |
| 7 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1667 | 82 | 85 | 33 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1613 | 77 | 83 | 26 |
| 9 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1459 | 44 | 45 | 12 |
| 10 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1482 | 20 | 34 | 11 |
| 11 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1716 | 90 | 62 | 27 |
| 12 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1548 | 57 | 66 | 17 |
| 13 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1498 | 35 | 30 | 18 |
| 14 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1478 | 21 | 61 | 4 |
| 15 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1572 | 131 | 26 | 81 |
| 16 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1559 | 52 | 89 | 25 |
| 17 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1509 | 46 | 59 | 26 |
| 18 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1494 | 26 | 20 | 9 |
| 19 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1616 | 116 | 58 | 65 |
| 20 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1626 | 100 | 55 | 16 |
| 21 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1506 | 25 | 48 | 9 |
| 22 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1509 | 20 | 34 | 6 |
| 23 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1617 | 133 | 47 | 50 |
| 24 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1579 | 97 | 57 | 53 |
| 25 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1494 | 29 | 66 | 10 |
| 26 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1485 | 20 | 56 | 8 |
| 27 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1575 | 108 | 75 | 38 |
| 28 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1623 | 96 | 66 | 46 |
| 29 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1486 | 33 | 67 | 11 |
| 30 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1510 | 29 | 56 | 3 |
| 31 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1549 | 85 | 61 | 54 |
| 32 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1520 | 53 | 50 | 33 |
| 33 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1552 | 34 | 33 | 14 |
| 34 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1510 | 33 | 19 | 12 |
| 35 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1644 | 119 | 84 | 42 |
| 36 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1635 | 66 | 112 | 29 |
| 37 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1514 | 45 | 40 | 23 |
| 38 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1490 | 20 | 54 | 8 |
| 39 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1621 | 138 | 73 | 77 |
| 40 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1587 | 82 | 50 | 17 |
| 41 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1466 | 50 | 28 | 9 |
| 42 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1449 | 35 | 27 | 13 |
| 43 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1645 | 177 | 111 | 147 |
| 44 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1562 | 104 | 64 | 51 |
| 45 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1510 | 29 | 50 | 9 |
| 46 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1477 | 24 | 34 | 7 |
| 47 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1601 | 125 | 35 | 60 |
| 48 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1607 | 36 | 88 | 20 |
| 49 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1548 | 29 | 21 | 15 |
| 50 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1509 | 27 | 43 | 1 |
| 51 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1615 | 167 | 40 | 64 |
| 52 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1560 | 78 | 73 | 28 |
| 53 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1492 | 32 | 23 | 17 |
| 54 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1508 | 19 | 39 | 3 |
| 55 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1656 | 206 | 70 | 159 |
| 56 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1669 | 111 | 93 | 33 |
| 57 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1496 | 72 | 37 | 47 |
| 58 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1507 | 28 | 31 | 10 |

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Elit. | I ₁₂ | I ₁₃ | I ₁₄ | I ₁₅ | I ₁₆ | I ₂₃ | I ₂₄ | I ₂₅ | I ₂₆ | I ₃₄ | I ₃₅ | I ₃₆ | I ₄₅ | I ₄₆ | I ₅₆ | Y ₁ D | Y ₂ G | σ ₁ Y ₁ | σ ₂ Y ₂ | |
|--------------------|--------------------------|----------------------|------------------------|----------------------|------------------------|-------------------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|-----------------|---------------------|---------------------|----------------------------------|----------------------------------|--|
| 59 | -1 | 1 | -1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | 1566 | 108 | 63 | 49 | |
| 60 | 1 | 1 | -1 | 1 | 1 | 1 | 1 | -1 | -1 | 1 | 1 | -1 | -1 | 1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | 1609 | 98 | 61 | 35 | |
| 61 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1455 | 28 | 20 | 14 | |
| 62 | 1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1496 | 25 | 40 | 11 | |
| 63 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1588 | 95 | 73 | 32 | |
| 64 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1566 | 143 | 83 | 35 | |
| N. | | | | | | | | | | | | | | | | | | | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | | | | | | | | | | | | | | | | | | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | | | | | | | | | | | | | | | | | | | |
| M. | | | | | | | | | | | | | | | | | | | | | | | | | | |
| σ _{effet} | | | | | | | | | | | | | | | | | | | | | | | | | | |
| E. | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Y ₁ | -10 | 53 | -3 | -16 | -4 | 2 | -4 | 3 | 3 | 9 | 2 | -3 | -1 | -5 | 2 | -3 | 0 | 2 | 0 | 0 | 0 | 3 | | | | |
| Y ₂ | -14 | 36 | -3 | -3 | 3 | 6 | -7 | 0 | 1 | 3 | -3 | 1 | -4 | 4 | 5 | -3 | -3 | 1 | -2 | 2 | 2 | 2 | | | | |
| C. | | | | | | | | | | | | | | | | | | | | | | | | | | |
| Y ₁ | 50 | MX1 | 0,7 | OrOpt | 0,5 | 50% | | | | | | | | | | | | | | | | | 1449 | | | |
| Y ₂ | 50 | MX1 | 0,7 | OrOpt | 0,2 | 50% | | | | | | | | | | | | | | | | | 18 | | | |

Tab. A.17: Plan complet du problème RC201 statique

| Nb | X ₁ Taille | X ₂ Cr | X ₃ T2Cr | X ₄ Mt | X ₅ T2Mt | X ₆ Elit. | Y ₁ D | Y ₂ G | σ ₁ Y ₁ | σ ₂ Y ₂ |
|--------------------|--------------------------|----------------------|------------------------|----------------------|------------------------|-------------------------|---------------------|---------------------|----------------------------------|----------------------------------|
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 1579 | 97 | 57 | 53 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 1601 | 125 | 35 | 60 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 1486 | 33 | 67 | 11 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 1507 | 28 | 31 | 10 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 1615 | 167 | 40 | 64 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 1490 | 20 | 54 | 8 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 1548 | 57 | 66 | 17 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 1551 | 49 | 53 | 31 |
| Niveaux | | | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | | | |
| Moyenne | | | | | | | | | | |
| σ _{effet} | | | | | | | | | | |
| Contrastes | | | | | | | | | | |
| Y ₁ | -16 | 39 | -8 | -12 | 0 | 6 | | | | |
| Y ₂ | -21 | 40 | -3 | -11 | 9 | 13 | | | | |
| Meilleures | | | | | | | | | | |
| Y ₁ | 50 | MX1 | 0,7 | OrOpt | 0,5 | 50% | 1510 | | | |
| Y ₂ | 50 | MX1 | 0,7 | OrOpt | 0,2 | 50% | | 21 | | |

Tab. A.18: Plan fractionnaire du problème RC201 statique

A.3 Le contraste vs. l'effet

Dans cette section nous allons calculer les contrastes de notre plan fractionnaire (la table A.19 ou la table ordonnée A.20).

| Nb | 1 | 2 | 3 | 4 | 5 | 6 |
|----|----|----|----|----|----|----|
| 1 | 1 | 1 | 1 | -1 | 1 | -1 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 |

TAB. A.19 – Plan fractionnaire

| Nb | 1 | 2 | 3 | 4 | 5 | 6 |
|----|----|----|----|----|----|----|
| 8 | -1 | -1 | -1 | -1 | -1 | -1 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 |
| 1 | 1 | 1 | 1 | -1 | 1 | -1 |

TAB. A.20 – Plan fractionnaire ordonné

D'abord, on commence par le plan complet ($2^4//16$) pour 4 facteurs à 2 niveaux (table A.21). Dans celle-ci les colonnes 1, 2, 3 et 4 sont les facteurs, les autres colonnes sont les interactions entre ces facteurs. Maintenant, nous pouvons diviser cette table et obtenir deux tables fractionnaires (i.e. les lignes numérotées sont les lignes de notre plan fractionnaire) et les colonnes (1, 2, 3, 4, 123, et 234) sont les mêmes que notre plan. Donc la colonne 5 dans notre plan fractionnaire est l'interaction entre les facteurs (1, 2, 3), dans ce cas le générateur d'alias est $I_1 = 1235$. De même pour la colonne 6, l'interaction 234 et le générateur d'alias est $I_2 = 2346$. Si l'on multiplie ces deux générateurs d'alias indépendants membre à membre, on obtient un troisième générateur :

$$I.I = 1235.2346$$

$$I = 1456$$

Alors, nous avons obtenu un **groupe de générateurs d'alias** :

$$I = 1235 = 2346 = 1456$$

Nous utilisons ce groupe pour savoir comment les facteurs et les interactions sont aliasés dans les contrastes que l'on calcule avec notre plan fractionnaire. Par exemple, le contraste l_1 sera déterminé en multipliant tous les termes du I par la colonne 1 qui a servi à le calculer.

$$1.I = 1.1235 = 1.2346 = 1.1456$$

$$1 = 235 = 12346 = 456$$

La relation d'équivalence donne les quatre termes du contraste :

$$l_1 = E_1 + E_{235} + E_{12346} + E_{456}$$

Nous constatons que ce contraste l_1 approximativement égal à l'effet E_1 ($l_1 \approx E_1$) parce qu'il est prouvé que les effets des interactions d'ordre supérieur ou égal à trois, sont négligeables [Goupy 1997]. Le calcul est le même pour les autres contrastes.

$$l_2 = E_2 + E_{135} + E_{346} + E_{12456} \Rightarrow l_2 \approx E_2$$

$$l_3 = E_3 + E_{125} + E_{246} + E_{13456} \Rightarrow l_3 \approx E_3$$

$$l_4 = E_4 + E_{12345} + E_{236} + E_{156} \Rightarrow l_4 \approx E_4$$

$$l_5 = E_5 + E_{123} + E_{23456} + E_{146} \Rightarrow l_5 \approx E_5$$

$$l_6 = E_6 + E_{12356} + E_{234} + E_{145} \Rightarrow l_6 \approx E_6$$

| Nb | 1 | 2 | 3 | 4 | 12 | 13 | 14 | 23 | 24 | 34 | 123 | 124 | 234 | 1234 |
|----|----|----|----|----|----|----|----|----|----|----|-----|-----|-----|------|
| 8 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 | -1 | 1 |
| | 1 | -1 | -1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | 1 | 1 | -1 | -1 |
| 5 | -1 | 1 | -1 | -1 | -1 | 1 | 1 | -1 | -1 | 1 | 1 | 1 | 1 | -1 |
| | 1 | 1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 | 1 | -1 | -1 | 1 | 1 |
| | -1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 | 1 | -1 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | -1 | -1 | 1 | -1 | -1 | 1 | 1 | 1 |
| | -1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | -1 | -1 | -1 | 1 | -1 | 1 |
| 1 | 1 | 1 | 1 | -1 | 1 | 1 | -1 | 1 | -1 | -1 | 1 | -1 | -1 | -1 |
| | -1 | -1 | -1 | 1 | 1 | 1 | -1 | 1 | -1 | -1 | -1 | 1 | 1 | -1 |
| 4 | 1 | -1 | -1 | 1 | -1 | -1 | 1 | 1 | -1 | -1 | 1 | -1 | 1 | 1 |
| | -1 | 1 | -1 | 1 | -1 | 1 | -1 | -1 | 1 | -1 | 1 | -1 | -1 | 1 |
| 7 | 1 | 1 | -1 | 1 | 1 | -1 | 1 | -1 | 1 | -1 | -1 | 1 | -1 | -1 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | -1 | -1 | -1 | 1 | 1 | 1 | -1 | 1 |
| | 1 | -1 | 1 | 1 | -1 | 1 | 1 | -1 | -1 | 1 | -1 | -1 | -1 | -1 |
| 2 | -1 | 1 | 1 | 1 | -1 | -1 | -1 | 1 | 1 | 1 | -1 | -1 | 1 | -1 |
| | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |

TAB. A.21 – Plan complet pour 4 facteurs à 2 niveaux

A.4 Plans d'expériences fractionnaires de problèmes dynamiques

| Nb | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | Y_1 | σ_1 |
|------------------|--------|-------|-------|-------|-------|-------|-------|------------|
| | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | D | Y_1 |
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 897 | 34 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 897 | 34 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 892 | 52 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 918 | 45 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 886 | 31 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 918 | 52 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 863 | 16 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 897 | 16 |
| Niveaux | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | |
| Moyenne | | | | | | | 896 | 35 |
| σ_{effet} | | | | | | | | 12 |
| Contrastes | | | | | | | | |
| Y_1 | 3 | -10 | 5 | -3 | 2 | 9 | | |
| Meilleure | | | | | | | | |
| Y_1 | 30 | PMX | 0,5 | OrOpt | 0,2 | 50% | | |

TAB. A.22: Plan fractionnaire du problème C104 dynamique

| Nb | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | Y_1 | σ_1 |
|------------------|--------|-------|-------|-------|-------|-------|-------|------------|
| | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | D | Y_1 |
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 3105 | 938 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 1631 | 683 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 950 | 182 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 1178 | 173 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 1547 | 367 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 1312 | 52 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 2264 | 968 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 1225 | 266 |
| Niveaux | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | |
| Moyenne | | | | | | | 1651 | 454 |
| σ_{effet} | | | | | | | | 160 |
| Contrastes | | | | | | | | |
| Y_1 | 313 | 485 | 98 | -146 | 43 | -234 | | |
| Meilleure | | | | | | | | |
| Y_1 | 30 | MX1 | 0,5 | OrOpt | 0,2 | 70% | | |

TAB. A.23: Plan fractionnaire du problème C108 dynamique

| Nb | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | Y_1 | σ_1 |
|------------------|--------|-------|-------|-------|-------|-------|-------|------------|
| | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | D | Y_1 |
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 860 | 138 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 805 | 135 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 865 | 102 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 747 | 81 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 918 | 309 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 887 | 246 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 737 | 25 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 886 | 191 |
| Niveaux | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | |
| Moyenne | | | | | | | 838 | 153 |
| σ_{effet} | | | | | | | | 54 |
| Contrastes | | | | | | | | |
| Y_1 | -31 | -8 | 16 | -50 | 9 | 1 | | |
| Meilleure | | | | | | | | |
| Y_1 | 50 | PMX | 0,5 | OrOpt | 0,2 | 50% | | |

TAB. A.24: Plan fractionnaire du problème C204 dynamique

| Nb | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | Y_1 | σ_1 |
|------------------|--------|-------|-------|-------|-------|-------|-------|------------|
| | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | D | Y_1 |
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 1782 | 1614 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 664 | 101 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 596 | 5 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 593 | 0 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 5981 | 10737 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 597 | 9 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 614 | 42 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 715 | 165 |
| Niveaux | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | |
| Moyenne | | | | | | | 1443 | 1584 |
| σ_{effet} | | | | | | | | 560 |
| Contrastes | | | | | | | | |
| Y_1 | -546 | 817 | -533 | -826 | 795 | 516 | | |
| Meilleure | | | | | | | | |
| Y_1 | 50 | MX1 | 0,7 | OrOpt | 0,2 | 50% | | |

TAB. A.25: Plan fractionnaire du problème C207 dynamique

A.4. Plans d'expériences fractionnaires de problèmes dynamiques

| Nb | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | Y_1 | σ_1 |
|------------------|--------|-------|-------|-------|-------|-------|-------|------------|
| | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | D | Y_1 |
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 1126 | 38 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 1141 | 32 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 1154 | 21 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 1177 | 45 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 1112 | 33 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 1127 | 22 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 1133 | 24 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 1120 | 16 |
| Niveaux | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | |
| Moyenne | | | | | | | 1136 | 29 |
| σ_{effet} | | | | | | | | 10 |
| Contrastes | | | | | | | | |
| Y_1 | 4 | -8 | 1 | 15 | 6 | 3 | | |
| Meilleure | | | | | | | | |
| Y_1 | 30 | PMX | 0,5 | 1Opt | 0,2 | 50% | | |

TAB. A.26: Plan fractionnaire du problème R104 dynamique

| Nb | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | Y_1 | σ_1 |
|------------------|--------|-------|-------|-------|-------|-------|-------|------------|
| | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | D | Y_1 |
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 1457 | 71 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 1549 | 113 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 1445 | 49 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 1482 | 70 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 1564 | 114 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 1516 | 79 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 1593 | 105 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 1535 | 105 |
| Niveaux | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | |
| Moyenne | | | | | | | 1518 | 88 |
| σ_{effet} | | | | | | | | 31 |
| Contrastes | | | | | | | | |
| Y_1 | -6 | 23 | -26 | 0 | -31 | 10 | | |
| Meilleure | | | | | | | | |
| Y_1 | 50 | MX1 | 0,7 | OrOpt | 0,5 | 50% | | |

TAB. A.27: Plan fractionnaire du problème R109 dynamique

| Nb | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | Y_1 | σ_1 |
|------------------|--------|-------|-------|-------|-------|-------|-------|------------|
| | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | D | Y_1 |
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 888 | 78 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 833 | 36 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 855 | 70 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 858 | 25 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 946 | 108 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 916 | 78 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 892 | 68 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 969 | 155 |
| Niveaux | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | |
| Moyenne | | | | | | | 895 | 77 |
| σ_{effet} | | | | | | | | 27 |
| Contrastes | | | | | | | | |
| Y_1 | -6 | -5 | -22 | -35 | -8 | -6 | | |
| Meilleure | | | | | | | | |
| Y_1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | |

TAB. A.28: Plan fractionnaire du problème R208 dynamique

| Nb | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | Y_1 | σ_1 |
|------------------|--------|-------|-------|-------|-------|-------|-------|------------|
| | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | D | Y_1 |
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 1213 | 124 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 1320 | 128 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 1138 | 42 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 1210 | 58 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 1218 | 144 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 1220 | 81 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 1191 | 60 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 1213 | 57 |
| Niveaux | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | |
| Moyenne | | | | | | | 1215 | 87 |
| σ_{effet} | | | | | | | | 31 |
| Contrastes | | | | | | | | |
| Y_1 | -7 | 20 | 7 | -1 | -21 | 27 | | |
| Meilleure | | | | | | | | |
| Y_1 | 50 | MX1 | 0,5 | OrOpt | 0,5 | 50% | | |

TAB. A.29: Plan fractionnaire du problème R205 dynamique

A.4. Plans d'expériences fractionnaires de problèmes dynamiques

| Nb | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | Y_1 | σ_1 |
|------------------|--------|-------|-------|-------|-------|-------|-------|------------|
| | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | D | Y_1 |
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 1234 | 41 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 1239 | 46 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 1205 | 19 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 1219 | 12 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 1232 | 40 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 1240 | 16 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 1220 | 42 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 1245 | 41 |
| Niveaux | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | |
| Moyenne | | | | | | | 1229 | 32 |
| σ_{effet} | | | | | | | | 11 |
| Contrastes | | | | | | | | |
| Y_1 | -1 | 2 | 0 | -9 | -7 | 3 | | |
| Meilleure | | | | | | | | |
| Y_1 | 50 | MX1 | 0,5 | OrOpt | 0,5 | 50% | | |

TAB. A.30: Plan fractionnaire du problème RC104 dynamique

| Nb | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | Y_1 | σ_1 |
|------------------|--------|-------|-------|-------|-------|-------|-------|------------|
| | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | D | Y_1 |
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 1790 | 230 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 1621 | 78 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 1592 | 185 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 1613 | 218 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 1723 | 160 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 1607 | 152 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 1746 | 94 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 1707 | 192 |
| Niveaux | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | |
| Moyenne | | | | | | | 1675 | 164 |
| σ_{effet} | | | | | | | | 58 |
| Contrastes | | | | | | | | |
| Y_1 | 14 | 45 | -22 | -32 | 5 | -34 | | |
| Meilleure | | | | | | | | |
| Y_1 | 30 | MX1 | 0,7 | OrOpt | 0,2 | 70% | | |

TAB. A.31: Plan fractionnaire du problème RC106 dynamique

| Nb | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | Y_1 | σ_1 |
|------------------|--------|-------|-------|-------|-------|-------|-------|------------|
| | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | D | Y_1 |
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 949 | 30 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 959 | 44 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 930 | 57 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 1012 | 41 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 995 | 38 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 1051 | 83 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 993 | 34 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 1002 | 120 |
| Niveaux | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | |
| Moyenne | | | | | | | 986 | 56 |
| σ_{effet} | | | | | | | | 20 |
| Contrastes | | | | | | | | |
| Y_1 | 15 | -13 | -14 | -13 | -15 | 18 | | |
| Meilleure | | | | | | | | |
| Y_1 | 30 | PMX | 0,7 | OrOpt | 0,5 | 50% | | |

TAB. A.32: Plan fractionnaire du problème RC204 dynamique

| Nb | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | Y_1 | σ_1 |
|------------------|--------|-------|-------|-------|-------|-------|-------|------------|
| | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | D | Y_1 |
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 1642 | 68 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 1766 | 188 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 1583 | 36 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 1612 | 71 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 1682 | 197 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 1545 | 37 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 1826 | 295 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 1599 | 59 |
| Niveaux | | | | | | | | |
| -1 | 30 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | |
| 1 | 50 | PMX | 0,7 | OrOpt | 0,5 | 70% | | |
| Moyenne | | | | | | | 1657 | 119 |
| σ_{effet} | | | | | | | | 42 |
| Contrastes | | | | | | | | |
| Y_1 | -1 | 72 | -23 | 40 | -27 | -6 | | |
| Meilleure | | | | | | | | |
| Y_1 | 50 | MX1 | 0,7 | 1Opt | 0,5 | 70% | | |

TAB. A.33: Plan fractionnaire du problème RC201 dynamique

A.5 Plans d'expériences fractionnaires de problèmes DPDPTW

| Nb | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | Y_1 | σ_1 |
|------------------|--------|-------|-------|-------|-------|-------|-------|------------|
| | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | D | Y_1 |
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 3073 | 108 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 2915 | 57 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 2987 | 83 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 2883 | 33 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 2937 | 78 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 2998 | 128 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 2940 | 54 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 2935 | 46 |
| Niveaux | | | | | | | | |
| -1 | 10 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | |
| 1 | 20 | PMX | 0,7 | OrOpt | 0,5 | 70% | | |
| Moyenne | | | | | | | 2959 | 73 |
| σ_{effet} | | | | | | | | 26 |
| Contrastes | | | | | | | | |
| Y_1 | 15 | 8 | 35 | -27 | 12 | -25 | | |
| Meilleure | | | | | | | | |
| Y_1 | 10 | MX1 | 0,5 | OrOpt | 0,2 | 70% | | |

TAB. A.34: Plan fractionnaire du problème Rnd8-10-100-005 Dynamique

| Nb | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | Y_1 | σ_1 |
|------------------|--------|-------|-------|-------|-------|-------|-------|------------|
| | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | D | Y_1 |
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 2730 | 18 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 2750 | 19 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 2775 | 18 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 2796 | 18 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 2721 | 18 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 2770 | 17 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 2735 | 18 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 2722 | 17 |
| Niveaux | | | | | | | | |
| -1 | 10 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | |
| 1 | 20 | PMX | 0,7 | OrOpt | 0,5 | 70% | | |
| Moyenne | | | | | | | 2750 | 18 |
| σ_{effet} | | | | | | | | 6 |
| Contrastes | | | | | | | | |
| Y_1 | 8 | -16 | 6 | 14 | 6 | 9 | | |
| Meilleure | | | | | | | | |
| Y_1 | 10 | PMX | 0,5 | 1Opt | 0,2 | 50% | | |

TAB. A.35: Plan fractionnaire du problème Rnd8-10-100-015 Dynamique

| Nb | X_1 | X_2 | X_3 | X_4 | X_5 | X_6 | Y_1 | σ_1 |
|------------------|--------|-------|-------|-------|-------|-------|-------|------------|
| | Taille | Cr | T2Cr | Mt | T2Mt | Elit. | D | Y_1 |
| 1 | 1 | 1 | 1 | -1 | 1 | -1 | 2485 | 20 |
| 2 | -1 | 1 | 1 | 1 | -1 | 1 | 2534 | 20 |
| 3 | -1 | -1 | 1 | 1 | 1 | -1 | 2590 | 20 |
| 4 | 1 | -1 | -1 | 1 | 1 | 1 | 2491 | 20 |
| 5 | -1 | 1 | -1 | -1 | 1 | 1 | 2531 | 20 |
| 6 | 1 | -1 | 1 | -1 | -1 | 1 | 2572 | 19 |
| 7 | 1 | 1 | -1 | 1 | -1 | -1 | 2558 | 19 |
| 8 | -1 | -1 | -1 | -1 | -1 | -1 | 2514 | 19 |
| Niveaux | | | | | | | | |
| -1 | 10 | MX1 | 0,5 | 1Opt | 0,2 | 50% | | |
| 1 | 20 | PMX | 0,7 | OrOpt | 0,5 | 70% | | |
| Moyenne | | | | | | | 2534 | 20 |
| σ_{effet} | | | | | | | | 7 |
| Contrastes | | | | | | | | |
| Y_1 | -8 | -7 | 11 | 9 | -10 | -2 | | |
| Meilleure | | | | | | | | |
| Y_1 | 20 | PMX | 0,5 | 1Opt | 0,5 | 70% | | |

TAB. A.36: Plan fractionnaire du problème Rnd8-10-100-025 Dynamique

Annexe B

Exemple : Le problème C101 dynamique

Le problème C101 dynamique est divisé en 3 parties :

1. Les clients : Chaque client est représenté par son identification (la colonne Id), sa position (les colonnes X,Y), les deux limites de temps (les colonnes E,L), le temps de service (la colonne S) et le temps de réception (la colonne Tr).
2. Les véhicules : Les véhicules disponibles (V) sont homogènes et caractérisés par leur nombre et leur capacité.
3. Le dépôt : Il est défini par son identification (Id), sa position (X, Y) et son horaire d'ouverture (E, L).

Tab. B.1: Fichier de données du problème dynamique C101Dyn (Format texte)

| 1. Les clients | | | | | | | |
|----------------|----|----|----|-----|-----|----|----|
| Id | X | Y | D | E | L | S | Tr |
| 0 | 42 | 66 | 10 | 65 | 146 | 90 | 0 |
| 1 | 42 | 68 | 10 | 727 | 782 | 90 | 0 |
| 2 | 42 | 65 | 10 | 15 | 67 | 90 | 0 |
| 3 | 38 | 68 | 20 | 255 | 324 | 90 | 0 |
| 4 | 38 | 70 | 10 | 534 | 605 | 90 | 0 |
| 5 | 35 | 66 | 10 | 357 | 410 | 90 | 0 |
| 6 | 35 | 69 | 10 | 448 | 505 | 90 | 0 |
| 7 | 22 | 75 | 30 | 30 | 92 | 90 | 0 |
| 8 | 20 | 80 | 40 | 384 | 429 | 90 | 0 |
| 9 | 20 | 85 | 40 | 475 | 528 | 90 | 0 |
| 10 | 15 | 80 | 10 | 278 | 345 | 90 | 0 |
| 11 | 30 | 50 | 10 | 10 | 73 | 90 | 0 |
| 12 | 30 | 52 | 20 | 914 | 965 | 90 | 0 |
| 13 | 28 | 55 | 10 | 732 | 777 | 90 | 0 |
| 14 | 25 | 50 | 10 | 65 | 144 | 90 | 0 |
| 15 | 25 | 55 | 10 | 622 | 701 | 90 | 0 |
| 16 | 23 | 52 | 10 | 261 | 316 | 90 | 0 |
| 17 | 20 | 50 | 10 | 358 | 405 | 90 | 0 |
| 18 | 20 | 55 | 10 | 449 | 504 | 90 | 0 |
| 19 | 10 | 35 | 20 | 200 | 237 | 90 | 0 |
| 20 | 10 | 40 | 30 | 31 | 100 | 90 | 0 |
| 21 | 8 | 40 | 40 | 87 | 158 | 90 | 0 |
| 22 | 5 | 35 | 10 | 283 | 344 | 90 | 0 |
| 23 | 5 | 45 | 10 | 665 | 716 | 90 | 0 |
| 24 | 2 | 40 | 20 | 383 | 434 | 90 | 0 |
| 25 | 0 | 40 | 30 | 479 | 522 | 90 | 0 |
| 26 | 0 | 45 | 20 | 567 | 624 | 90 | 0 |
| 27 | 35 | 30 | 10 | 264 | 321 | 90 | 0 |
| 28 | 33 | 35 | 10 | 16 | 80 | 90 | 0 |
| 29 | 30 | 32 | 30 | 448 | 509 | 90 | 0 |
| 30 | 28 | 30 | 10 | 632 | 693 | 90 | 0 |
| 31 | 26 | 32 | 10 | 815 | 880 | 90 | 0 |
| 32 | 42 | 10 | 40 | 186 | 257 | 90 | 0 |
| 33 | 40 | 5 | 30 | 385 | 436 | 90 | 0 |
| 34 | 40 | 15 | 40 | 35 | 87 | 90 | 0 |
| 35 | 38 | 5 | 30 | 471 | 534 | 90 | 0 |
| 36 | 50 | 40 | 50 | 171 | 218 | 90 | 0 |
| 37 | 48 | 30 | 10 | 632 | 693 | 90 | 0 |

A suivre ⇒

Annexe B. Exemple : Le problème C101 dynamique

| Id | X | Y | D | E | L | S | Tr |
|----|----|----|----|------|------|----|--------|
| 38 | 48 | 40 | 10 | 76 | 129 | 90 | 0 |
| 39 | 47 | 35 | 10 | 826 | 875 | 90 | 0 |
| 40 | 47 | 40 | 10 | 12 | 77 | 90 | 0 |
| 41 | 45 | 30 | 10 | 734 | 777 | 90 | 0 |
| 42 | 45 | 35 | 10 | 916 | 969 | 90 | 0 |
| 43 | 95 | 35 | 20 | 293 | 360 | 90 | 0 |
| 44 | 92 | 30 | 10 | 478 | 551 | 90 | 0 |
| 45 | 53 | 35 | 50 | 353 | 412 | 90 | 0 |
| 46 | 85 | 35 | 30 | 47 | 124 | 90 | 0 |
| 47 | 75 | 55 | 20 | 369 | 420 | 90 | 0 |
| 48 | 72 | 55 | 10 | 265 | 338 | 90 | 0 |
| 49 | 66 | 55 | 10 | 173 | 238 | 90 | 0 |
| 50 | 65 | 55 | 20 | 85 | 144 | 90 | 0 |
| 51 | 65 | 60 | 30 | 645 | 708 | 90 | 0 |
| 52 | 60 | 55 | 10 | 20 | 84 | 90 | 0 |
| 53 | 60 | 60 | 10 | 836 | 889 | 90 | 0 |
| 54 | 65 | 85 | 40 | 475 | 518 | 90 | 0 |
| 55 | 60 | 80 | 10 | 95 | 156 | 90 | 0 |
| 56 | 60 | 85 | 30 | 561 | 622 | 90 | 0 |
| 57 | 58 | 75 | 20 | 30 | 84 | 90 | 0 |
| 58 | 55 | 80 | 10 | 743 | 820 | 90 | 0 |
| 59 | 55 | 85 | 20 | 647 | 726 | 90 | 0 |
| 60 | 45 | 68 | 10 | 912 | 967 | 90 | 217,05 |
| 61 | 45 | 70 | 30 | 825 | 870 | 90 | 357,92 |
| 62 | 40 | 69 | 20 | 621 | 702 | 90 | 369,09 |
| 63 | 40 | 66 | 20 | 170 | 225 | 90 | 0,03 |
| 64 | 25 | 85 | 20 | 652 | 721 | 90 | 62,89 |
| 65 | 22 | 85 | 10 | 567 | 620 | 90 | 411,22 |
| 66 | 18 | 75 | 20 | 99 | 148 | 90 | 73,84 |
| 67 | 15 | 75 | 20 | 179 | 254 | 90 | 87,17 |
| 68 | 28 | 52 | 20 | 812 | 883 | 90 | 144,01 |
| 69 | 25 | 52 | 40 | 169 | 224 | 90 | 69,34 |
| 70 | 23 | 55 | 20 | 546 | 593 | 90 | 66,95 |
| 71 | 8 | 45 | 20 | 751 | 816 | 90 | 342,84 |
| 72 | 35 | 32 | 10 | 166 | 235 | 90 | 64,08 |
| 73 | 33 | 32 | 20 | 68 | 149 | 90 | 44,64 |
| 74 | 32 | 30 | 10 | 359 | 412 | 90 | 164,05 |
| 75 | 30 | 30 | 10 | 541 | 600 | 90 | 192,81 |
| 76 | 30 | 35 | 10 | 1054 | 1127 | 90 | 124,72 |
| 77 | 28 | 35 | 10 | 1001 | 1066 | 90 | 240 |
| 78 | 25 | 30 | 10 | 725 | 786 | 90 | 615,99 |
| 79 | 25 | 35 | 10 | 912 | 969 | 90 | 315,89 |
| 80 | 44 | 5 | 20 | 286 | 347 | 90 | 44,19 |
| 81 | 42 | 15 | 10 | 95 | 158 | 90 | 8,72 |
| 82 | 38 | 15 | 10 | 651 | 740 | 90 | 534,18 |
| 83 | 35 | 5 | 20 | 562 | 629 | 90 | 96,41 |
| 84 | 50 | 30 | 10 | 531 | 610 | 90 | 482,17 |
| 85 | 50 | 35 | 20 | 262 | 317 | 90 | 118,58 |
| 86 | 95 | 30 | 30 | 387 | 456 | 90 | 258,24 |
| 87 | 53 | 30 | 10 | 450 | 505 | 90 | 117,07 |
| 88 | 45 | 65 | 20 | 997 | 1068 | 90 | 731,08 |
| 89 | 90 | 35 | 10 | 203 | 260 | 90 | 21,92 |
| 90 | 88 | 30 | 10 | 574 | 643 | 90 | 121,2 |
| 91 | 88 | 35 | 20 | 109 | 170 | 90 | 26,75 |
| 92 | 87 | 30 | 10 | 668 | 731 | 90 | 481,6 |
| 93 | 85 | 25 | 10 | 769 | 820 | 90 | 536,54 |
| 94 | 70 | 58 | 20 | 458 | 523 | 90 | 269,76 |
| 95 | 68 | 60 | 30 | 555 | 612 | 90 | 269,66 |
| 96 | 63 | 58 | 10 | 737 | 802 | 90 | 94,25 |
| 97 | 67 | 85 | 20 | 368 | 441 | 90 | 59,15 |
| 98 | 65 | 82 | 10 | 285 | 336 | 90 | 147,94 |
| 99 | 62 | 80 | 30 | 196 | 239 | 90 | 106,62 |

2. Les véhicules
 Id Nombre Capacité
 V 10 2000

3. Le dépôt
 Id X Y E L
 D 40 50 0 1236

Le format XML d'un fichier de données, peut être comme suivant :

```
<?xml version="1.0" ?>
<!DOCTYPE PROBLEME
[
<!ELEMENT PROBLEME (TYPE?, NOM?, CLIENT+, VEHICULE+, DEPOT+)>
<!ELEMENT TYPE (#PCDATA)>
<!ELEMENT NOM (#PCDATA)>
<!ELEMENT CLIENT (ID, POSITION, SERVICE, Tr)>
<!ELEMENT ID (#PCDATA)>
```

```
<!ELEMENT POSITION (X, Y)>
  <!ELEMENT X (#PCDATA)>
  <!ELEMENT Y (#PCDATA)>
<!ELEMENT SERVICE (FENETRE, D, S)>
  <!ELEMENT FENETRE (E, L)>
    <!ELEMENT E (#PCDATA)>
    <!ELEMENT L (#PCDATA)>
  <!ELEMENT D (#PCDATA)>
  <!ELEMENT S (#PCDATA)>
  <!ELEMENT Tr (#PCDATA)>
<!ELEMENT VEHICULE (ID, N, C, V)>
  <!ELEMENT ID (#PCDATA)>
  <!ELEMENT N (#PCDATA)>
  <!ELEMENT C (#PCDATA)>
  <!ELEMENT V (#PCDATA)>
<!ELEMENT DEPOT (ID, POSITION, HORAIRE)>
  <!ELEMENT ID (#PCDATA)>
  <!ELEMENT POSITION (X, Y)>
    <!ELEMENT X (#PCDATA)>
    <!ELEMENT Y (#PCDATA)>
  <!ELEMENT HORAIRE (E, L)>
    <!ELEMENT E (#PCDATA)>
    <!ELEMENT L (#PCDATA)>
]>
```

```
<PROBLEME>
  <TYPE> DVRPTW </TYPE>
  <NOM> VRPTW C101 DYNAMIC </NOM>
  <CLIENT>
    <ID> 58 </ID>
    <POSITION>
      <X> 55 </X>
      <Y> 80 </Y>
    </POSITION>
    <SERVICE>
      <FENETRE>
        <E> 743 </E>
        <L> 820 </L>
      </FENETRE>
      <D> 10 </D>
      <S> 90 </S>
    </SERVICE>
    <Tr> 0 </Tr>
  </CLIENT>
  <VEHICULE>
    <ID> V </ID>
    <N> 10 </N>
    <C> 2000 </C>
```

```
<V> 1 </V>
</VEHICULE>
<DEPOT>
  <ID> 1 </ID>
  <POSITION>
    <X> 40 </X>
    <Y> 50 </Y>
  </POSITION>
  <HORAIRE>
    <E> 0 </E>
    <L> 1236 </L>
  </HORAIRE>
</DEPOT>
</PROBLEME>
```

La solution du problème DVRPTW C101 Dynamique

```
v0 : 28-73-72-27-74-29-75-30-78-31-79-77-76
v1 : 11-14-69-16-17-18-70-15-13-68-12
v2 : 52-50-49-48-47-94-95-51-96-53
v3 : 20-21-19-22-24-25-26-23-71
v4 : 40-38-36-85-45-87-84-37-41-39-42
v5 : 46-91-89-43-86-44-90-92-93
v6 : 34-81-32-80-33-35-83-82
v7 : 7-66-67-10-8-9-65-64
v8 : 2-0-63-3-5-6-4-62-1-61-60-88
v9 : 57-55-99-98-97-54-56-59-58
```

```
Le degré de dynamisme ..... : 40 %
Fitness ..... : 828.94
La distance totale parcourue .. : 828.94
Le retard total ..... : 0.0
Le nombre de clients non-servis : 0
```

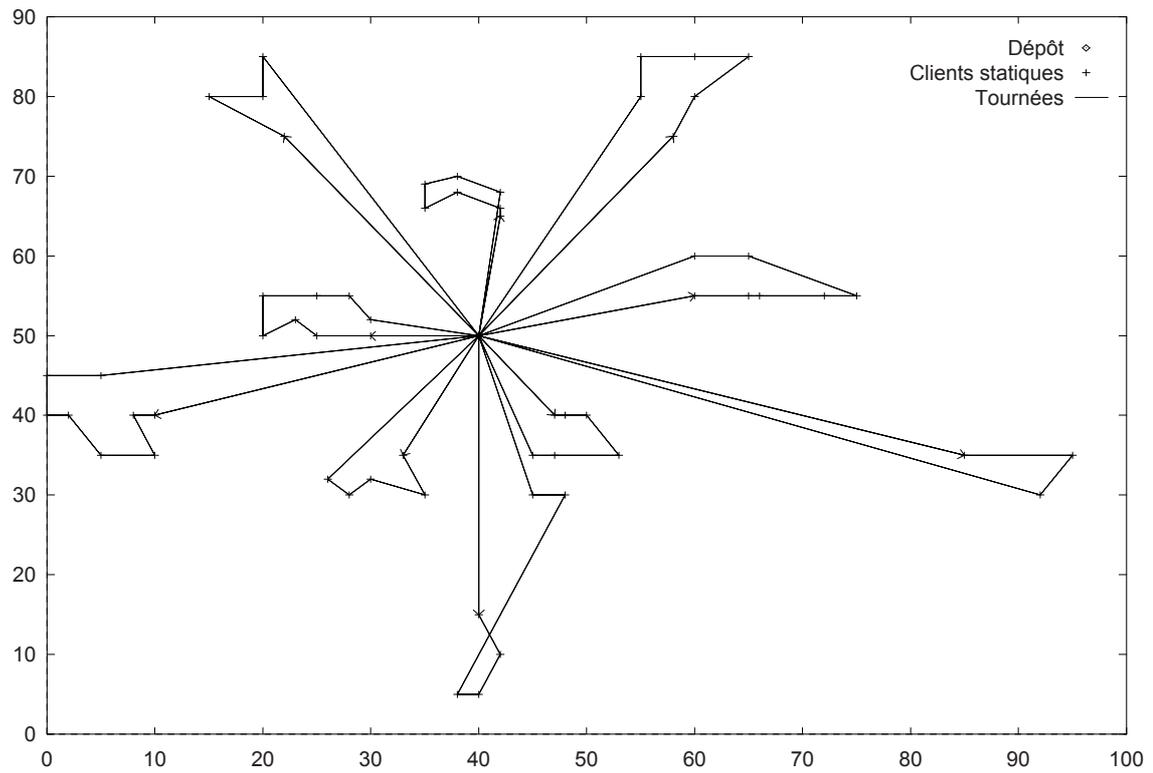


FIG. B.1 – La solution du problème DVRPTW C101 Dyn (Au début de la simulation)

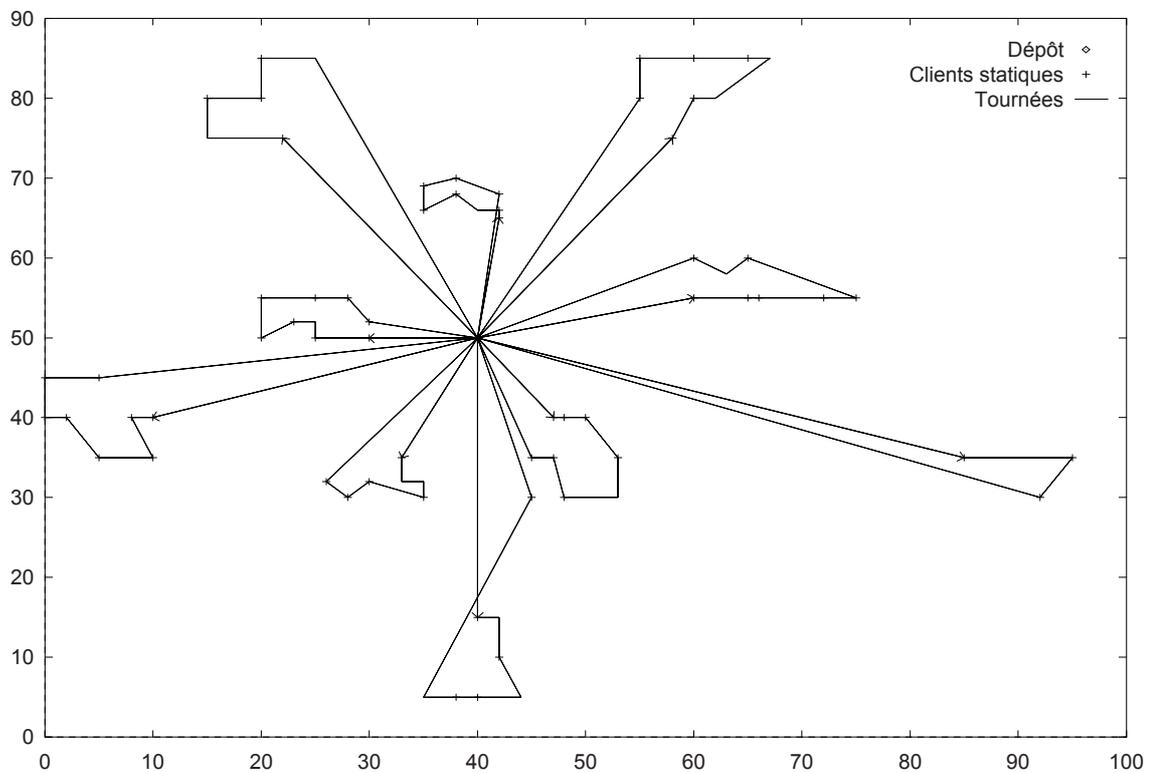


FIG. B.2 – La solution du problème DVRPTW C101 Dyn (Au moment donné t)

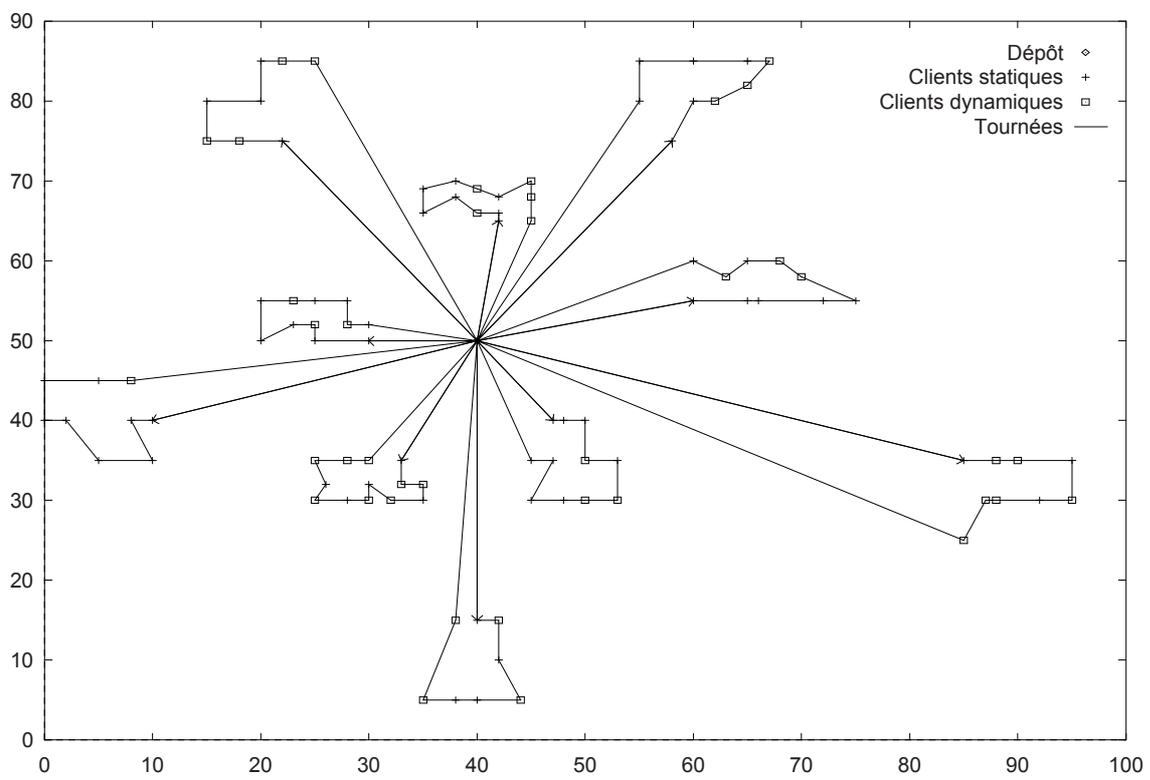


FIG. B.3 – La solution du problème DVRPTW C101 Dyn (A la fin de la simulation)

Annexe C

Courbes d'évolutions du fitness (Distance) de 6 problèmes DVRPTW Dynamiques

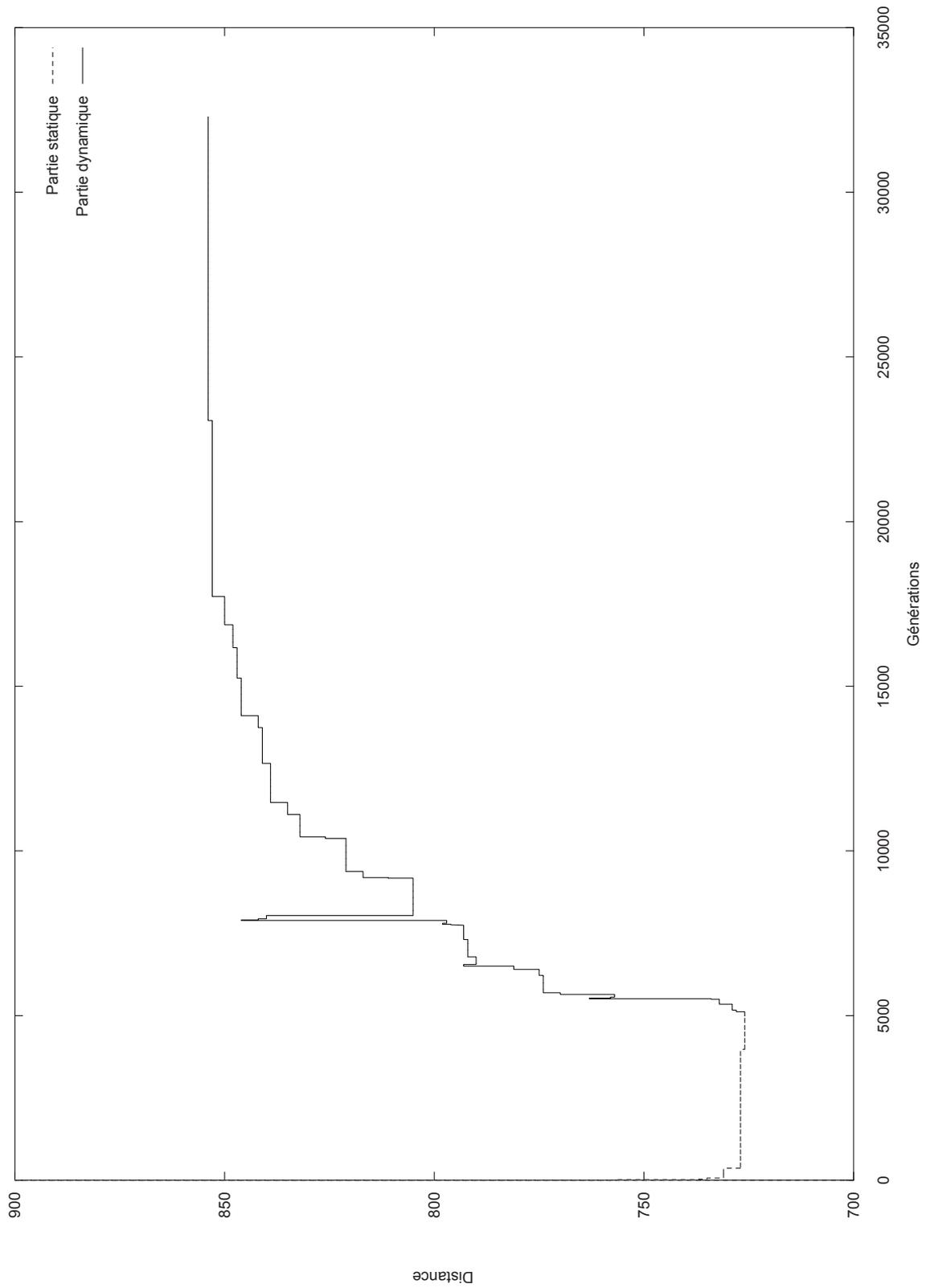


FIG. C.1 – Courbe d'évolution du fitness (Distance) du problème C108 Dynamique

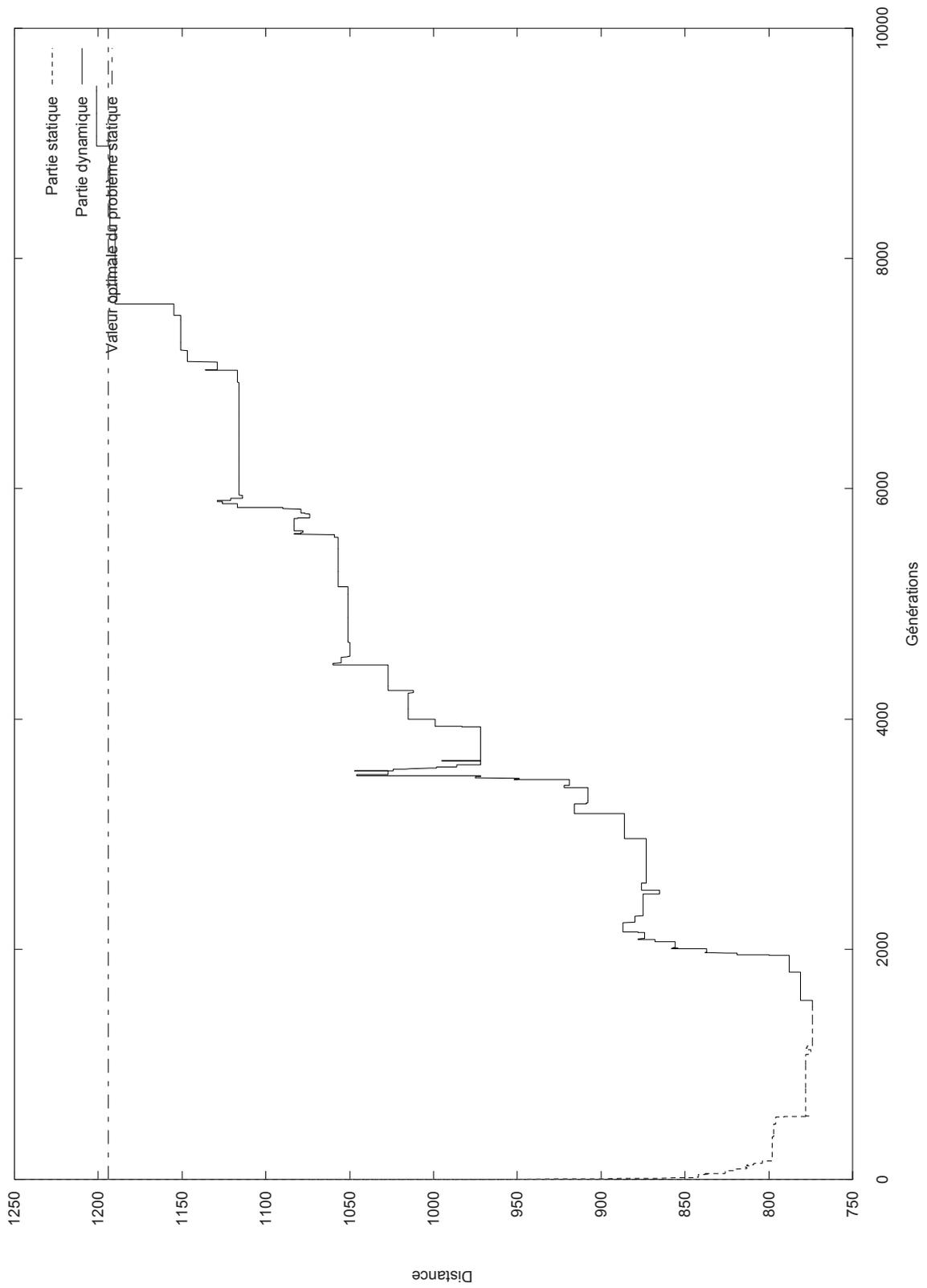


FIG. C.2 – Courbe d'évolution du fitness (Distance) du problème R109 Dynamique

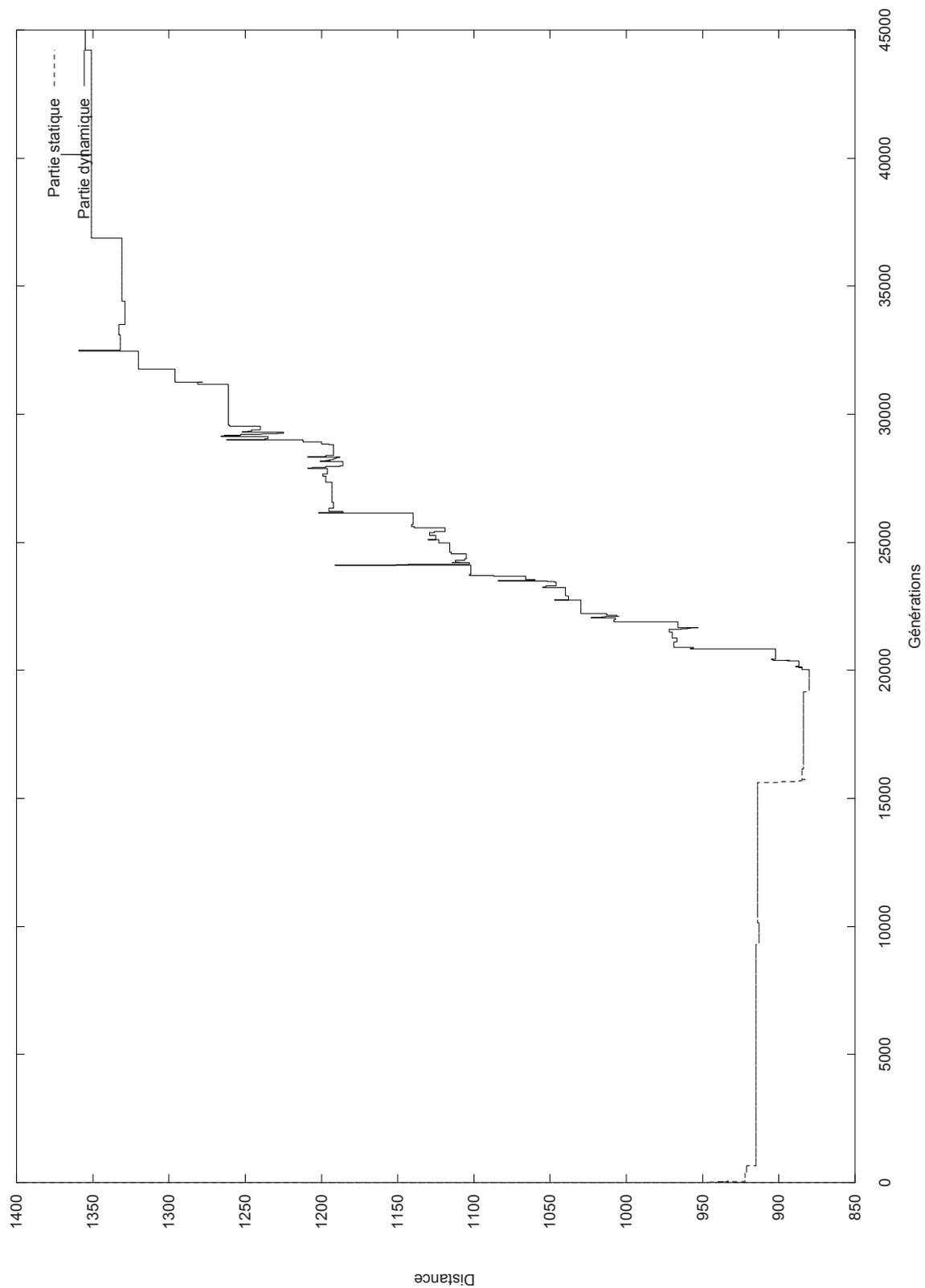


FIG. C.3 – Courbe d'évolution du fitness (Distance) du problème RC106 Dynamique

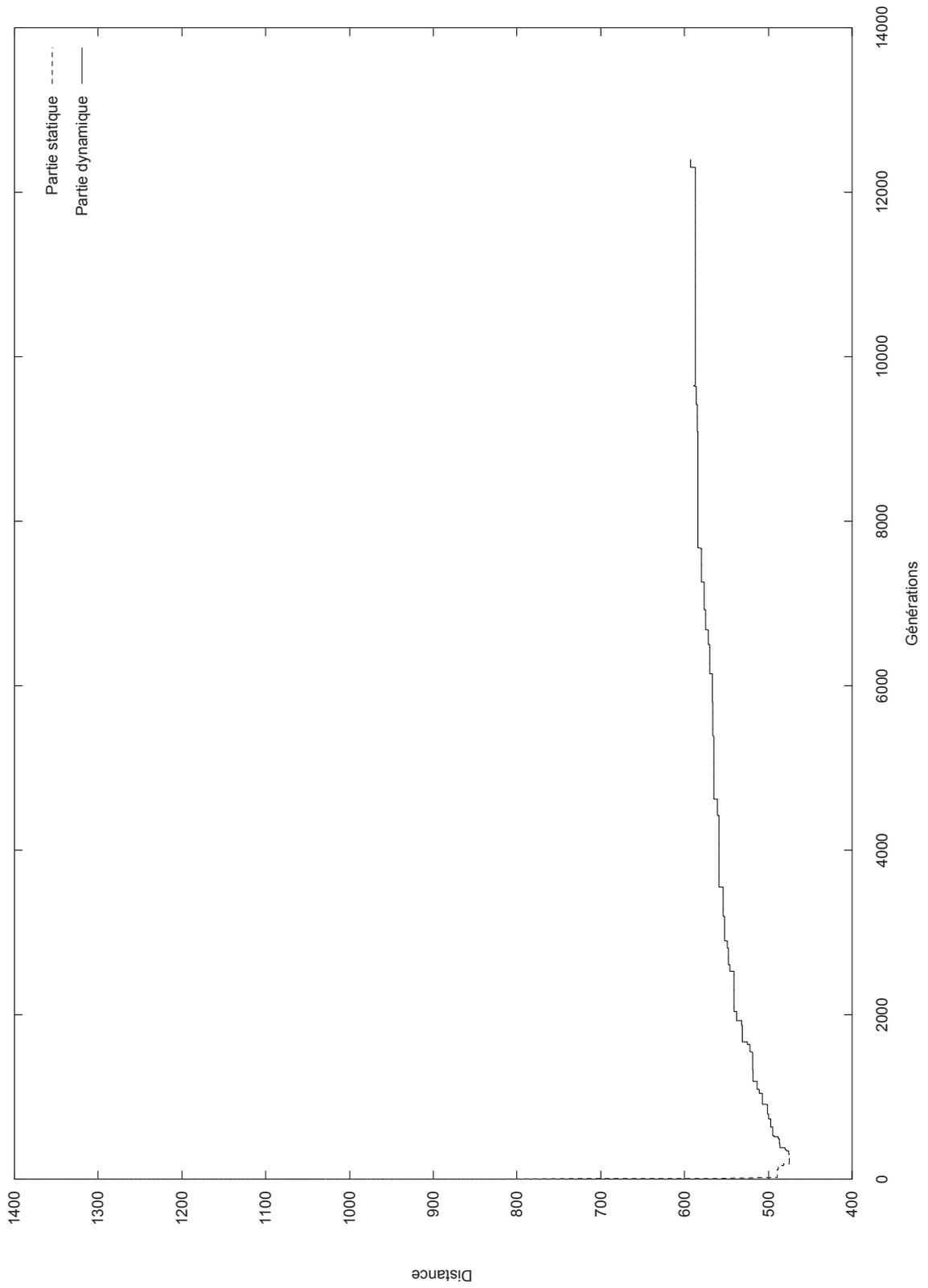


FIG. C.4 – Courbe d'évolution du fitness (Distance) du problème C207 Dynamique

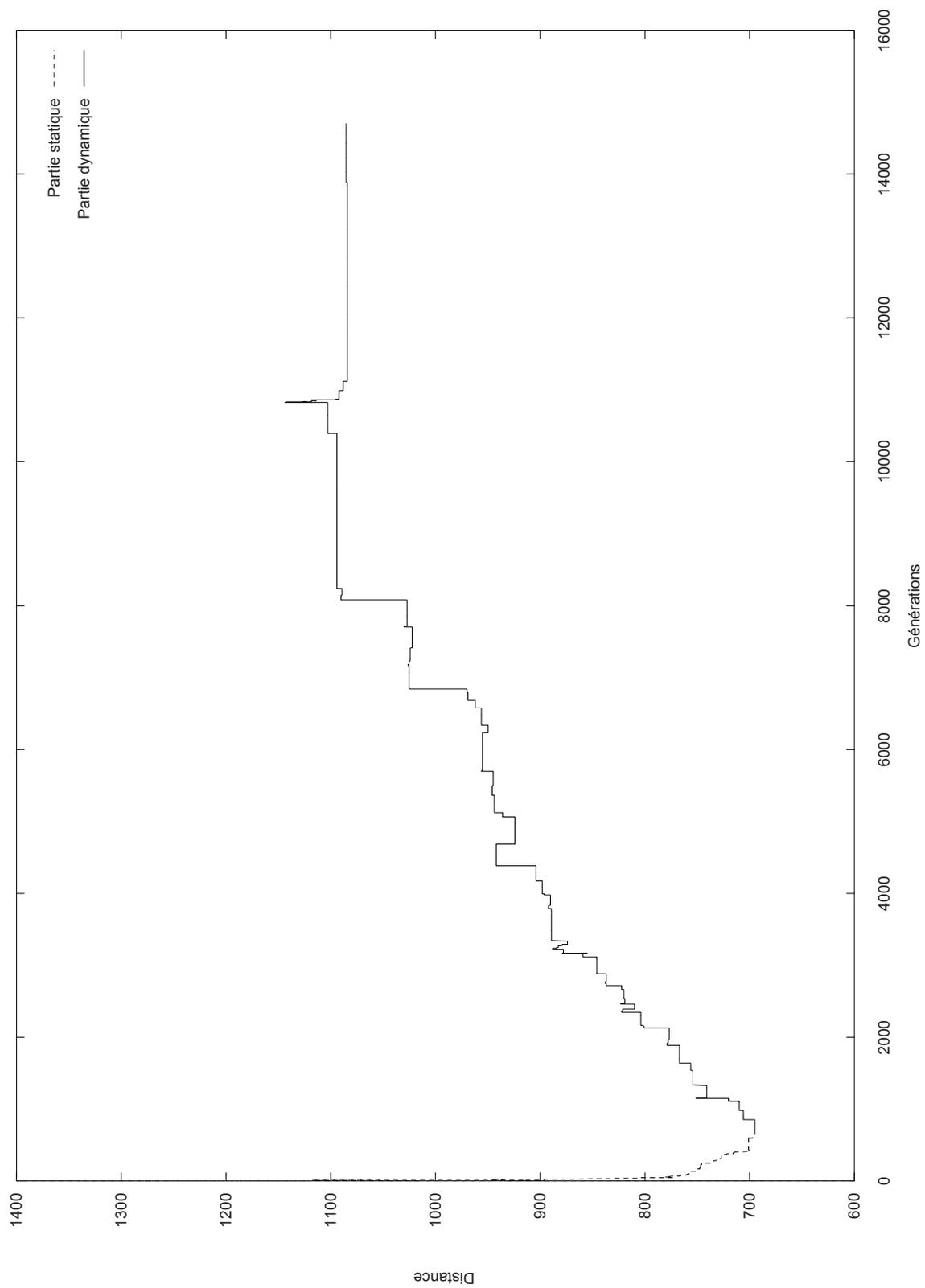


FIG. C.5 – Courbe d'évolution du fitness (Distance) du problème R205 Dynamique

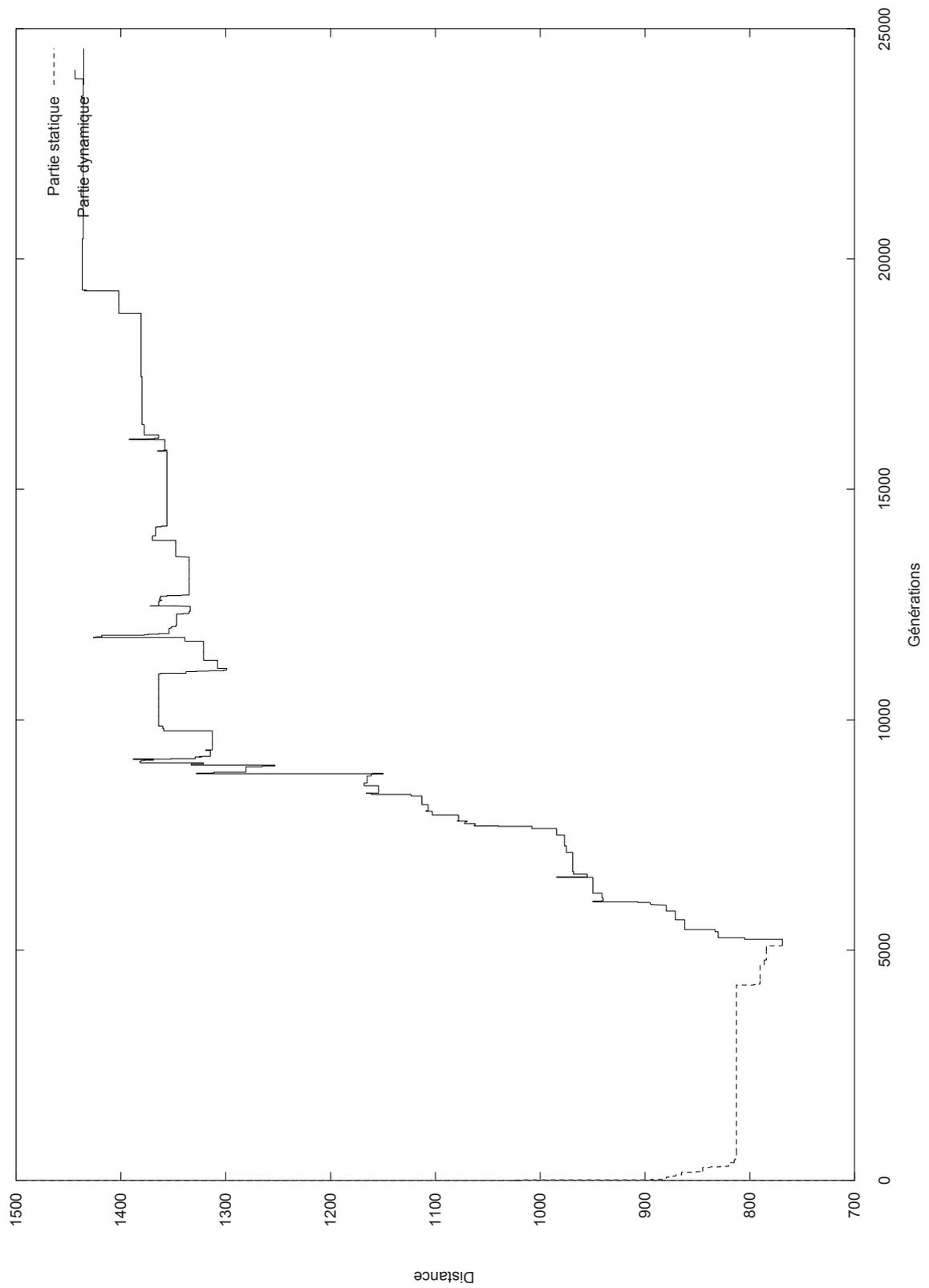


FIG. C.6 – Courbe d'évolution du fitness (Distance) du problème RC201 Dynamique

Bibliographie

- [Alba et al. 2004] E. Alba and B. Dorronsoro . « Solving the Vehicle Routing Problem by Using Cellular Genetic Algorithms ». *Evolutionary Computation in Combinatorial Optimization - EvoCOP 2004*, Jens Gottlieb and Günther R. Raidl, LNCS, Vol. 3004, Coimbra, Portugal, Springer Verlag, Berlin, pages 11–20, 2004.
- [Antes et al. 1995] J. Antes and U. Derigs . « A New Parallel Tour Construction Algorithm for the Vehicle Routing Problem with Time Windows ». *Lehrstuhl für Wirtschaftsinformatik und Operations Research, Universität zu Köln*, March 31 1995.
- [Archetti et al. 2002] C. Archetti , A. Hertz , and M. G. Speranza . « A Tabu Search Algorithm for the Split Delivery Vehicle Routing Problem ». *Mathematics of Information Technology and Complex Systems*, 2002.
- [Aronson 1995] L.D. Aronson . « Algorithms for vehicle routing - A survey ». Technical Report, Faculty of Technical Mathematics and Computer Science, Delft University of Technology, 15 November 1995.
- [Badeau et al. 1997] P. Badeau , M. Gendreau , F. Guertin , J.-Y. Potvin , and E. Taillard . « A Parallel Tabu Search Heuristic for the Vehicle Routing Problem with Time Windows ». *Transportation Research - C 5*, pages 109–122, 1997.
- [Baker et al. 2003] B. M. Baker and M.A. Ayechev . « A genetic algorithm for the vehicle routing problem ». *Computers and Operations Research 30*, pages 787–800, 2003.
- [Beasley 1983] J.E. Beasley . « Route first - cluster second methods for vehicle routing ». *Omega 11(4)*, pages 403–408, 1983.
- [Beck et al. 2003] J. C. Beck , P. Prosser , and E. Selensky . « Vehicle Routing and Job Shop Scheduling : What's the difference? ». *in Proc. of the 13th International Conference on Automated Planning & Scheduling (ICAPS'03), Trento (Italy)*, 2003.
- [Bent et al. 2003] R. Bent and P. Van Hentenryck . « Dynamic Vehicle Routing with Stochastic Requests ». *International Joint Conference on Artificial Intelligence*, pages 1362–1363, 2003.
- [Berger et al. 2002] J. Berger and M. Barkaoui . « A Memetic Algorithm for the Vehicle Routing Problem with Time Windows ». *Presented at the 7th International Command and Control Research and Technology Symposium*, 2002.

- [Berger et al. 2004] J. Berger and M. Barkaoui . « A new hybrid genetic algorithm for the capacitated vehicle routing problem ». *Journal of the Operational Research Society*, 54, pages 1254–1262, 2004.
- [Bertsimas et al. 1991] D.J. Bertsimas and G.V. Ryzin . « A stochastic and dynamic vehicle routing problem in the euclidean plane ». *Operations Research* 39 (4), pages 601–615, 1991.
- [Bertsimas et al. 1993a] D.J. Bertsimas and G.V. Ryzin . « A stochastic and dynamic vehicle routing in the euclidean plane with multiple capacitated vehicles ». *Operations Research* 41 (1), pages 60–76, 1993.
- [Bertsimas et al. 1993b] D.J. Bertsimas and G.V. Ryzin . « A stochastic and dynamic vehicle routing with general demand interarrival time distributions ». *Advanced Applied Probability* 25, pages 947–978, 1993.
- [Bianchi 2000] L. Bianchi . « Notes on Dynamic Vehicle Routing - The State of the Art ». Technical Report, IDSIA-05-01, 20 December 2000.
- [Blanton et al. 1993] J.L. Blanton and R.L. Wainwright . « Multiple Vehicle Routing with Time and Capacity Constraints using Genetic Algorithms ». *Proceedings of the Fifth International Conference on Genetic Algorithms*, pages 452–459, 1993.
- [Bramel et al. 1993] J. Bramel , C.L. LI , and D. Simchi-Levi . « Probabilistic analysis of a Vehicle Routing Problem with Time Windows ». *American Journal of Mathematical and Management Sciences* 13, pages 267–322, 1993.
- [Branke 2001] J. Branke . *Evolutionary Optimization in Dynamic Environments (Genetic Algorithms and Evolutionary Computation)*. Kluwer Academic Publishers, December 2001.
- [Breedam 1996] A. Van Breedam . « An Analysis of the Effect of Local Improvement Operators in Genetic Algorithms and Simulated Annealing for the Vehicle Routing Problem ». *RUCA, Faculty of Applied Economics, University of Antwerp, Antwerp, Belgium*, page Working Paper 96/14, 1996.
- [Brotcorne et al. 2003] L. Brotcorne , G. Laporte , and F. Semet . « Ambulance location and relocation models ». *European Journal of OR* 147, pages 451–463, 2003.
- [Bräysy et al. 2000] O. Bräysy , J. Berger , and M. Barkaoui . « A New Hybrid Evolutionary Algorithm for the Vehicle Routing Problem with Time Windows ». *Presented at the Route 2000 workshop, Danemark*, 2000.
- [Bräysy et al. 2001] O. Bräysy and M. Gendreau . « Metaheuristics for the Vehicle Routing Problem with Time Windows ». *SINTEF Applied Mathematics, Research Council of Norway*, 2001.
- [Caricato et al. 2003] P. Caricato , G. Ghiani , A. Grieco , and E. Guerriero . « Parallel Tabu Search For A Pickup And Delivery Problem Under Track Contention ». *Parallel Computing* 29 (5), pages 631–639, 2003.

-
- [Chakraborty et al. 1996] U. Chakraborty , K. Deb , and M. Chakraborty . « Analysis of selection algorithms : A Markov chain approach ». *Evolutionary Computation*, 4(2), pages 132–167, 1996.
- [Chiang et al. 1996] W. Chiang and R.A. Russell . « Simulated Annealing Metaheuristics for the Vehicle Routing Problem with Time Windows ». *Annals of Operations Research* 63, 3 27, 1996.
- [Chiang et al. 2004] W. Chiang and R.A. Russell . « Scatter search for the vehicle routing problem with time windows ». *European Journal of Operational Research*, forthcoming, 2004.
- [Christofides et al. 1979] N. Christofides , A. Mingozzi , and P. Toth . « The vehicle Routing Problem ». *Combinatorial Optimization*. Wiley, Chichester, pages 315–338, 1979.
- [Christofides et al. 1981] N. Christofides , S. A. Mingozzi , and P. Toth . « State Space Relaxation Procedures for the Computation of Bounds to Routing Problems ». *Networks* 11, pages 145–164, 1981.
- [Christofides et al. 1984] N. Christofides and J. Beasley . « The period routing problem ». *NETWORKS* 14, pages 237–256, 1984.
- [Cicirello et al. 2000] V. Cicirello and S. Smith . « Modeling GA Performance for Control Parameter Optimization ». *GECCO-2000 : Proceedings of the Genetic and Evolutionary Computation Conference*, July 2000.
- [Clarke et al. 1964] G. Clarke and J.W. Wright . « Scheduling of vehicles from a central depot to a number of delivery points ». *Operations Research* 12, pages 568–581, 1964.
- [Coello 1998] C.A. Coello . « An update survey of ga-based multiobjective optimization techniques ». Technical Report, RD-98-08, Laboratorio Nacional de Informatica Avanzada (LANIA), México, Dec. 1998.
- [Coloni et al. 1992] A. Coloni , M. Dorigo , and V. Maniezzo . « Distributed optimization by ant colonies ». *proceedings of the first European Conference on Artificial Life (ECAL 91)*, pages 134–142, 1992.
- [Cordeau et al. 1997] J.-F. Cordeau , M. Gendreau , and G. Laporte . « A tabu search heuristic for periodic and multi-depot vehicle routing problems ». *Networks*, 30, pages 105–119, 1997.
- [Cordeau et al. 2001] J.-F. Cordeau , G. Laporte , and A. Mercier . « A unified tabu search heuristic for vehicle routing problems with time windows ». *Journal of the Operational Research Society* 52, pages 928–936, 2001.
- [Cordeau et al. 2002] J.F. Cordeau , G. Desaulniers , J. Desrosiers , M. Solomon , and F. Soumis . « The VRP with Time Windows ». *SIAM Monographs on Discrete Mathematics and Applications*, 9, P. Toth and D. Vigo (eds.), SIAM, Philadelphia, PA, pages 157–193, 2002.
- [Cordeau et al. 2004] J.-F. Cordeau , M. Gendreau , A. Hertz , G. Laporte , and J.-S. Sormany . « New Heuristics for the Vehicle Routing Problem ». Technical Report, G-2004-33, Université de Montréal, Canada, April 2004.

- [Coy et al. 2000] S.P. Coy , B. Golden , G. Runger , and E. Wasil . « Using Experimental Design to Find Effective Parameter Settings for Heuristics ». *Journal of Heuristics* 7, pages 77–97, 2000.
- [Dantzig et al. 1959] G.B. Dantzig and J.H. Ramser . « The truck dispatching problem ». *Operations Research, Management Sciences*, 6(1), pages 80–91, 1959.
- [Davis 1991] L. Davis . « Handbook of genetic algorithms ». Technical Report, Van Nostrand Reinhold, 1991.
- [Deb et al. 1999] K. Deb and S. Agrawal . « Understanding interactions among genetic algorithm parameters ». *Foundation of Genetic Algorithms*, pages 265–286, 1999.
- [DeJong 1975] K. A. DeJong . « An analysis of the behaviour of a class of genetic adaptive systems ». *University Microfilms No. 76,9381*, 1975.
- [Derigs et al. 1993] U. Derigs and G. Grabenbauer . « A new heuristic approach to the Vehicle Routing Problem with Time Windows with a Bakery Fleet Case ». *American Journal of Mathematical and Management Sciences* 13, pages 249–366, 1993.
- [Desrochers et al. 1992] M. Desrochers , J. Desrosiers , and M. Solomon . « A new Optimization Algorithm for the Vehicle Routing Problem with Time Windows ». *Operations Research* 40, pages 342–354, 1992.
- [Desrosiers et al. 1995] J. Desrosiers , Y. Dumas , M. Solomon , and F. Soumis . « Time Constrained Routing and Scheduling ». *Handbooks in Operations Research and Management Science* 8, North Holland, Amsterdam, pages 35–139, 1995.
- [Didier et al. 2002] M. Didier and P.-T.-L Roger . « Planification d’expériences en formulation : criblage ». Technical Report, Techniques de l’Ingénieur, 2002.
- [Duhamel 2001] C. Duhamel . « *Un Cadre Formel pour les Méthodes par Amélioration Itérative - Application à deux problèmes d’Optimisation dans les Réseaux -* ». PhD thesis, Université Blaise Pascal - Clermont-Ferrand II, Mars 22 2001.
- [Dupas et al. 2003] R. Dupas , D. Fourmaux , and G. Goncalves . « Chapitre 9 : Utilisation des plans d’expériences pour l’évaluation de performances en simulation ». *Evaluation de performances des systèmes de production, Hermès, Traité IC2 série Productique, France, ISBN 274620634X. (sous la direction de C. Tahon)*, 2003.
- [Eglese et al. 2005] R.W. Eglese , Z. Fu , and L. Li . « A tabu search heuristic for the open vehicle routing problem ». *Journal of the Operational Research Society* 56 (3), pages 267–274, 2005.
- [Eiben et al. 1999] A. E. Eiben , R. Hinterding , and Z. Michalewicz . « Parameter Control in Evolutionary Algorithms ». *Evolutionary Computation, IEEE Transactions, Vol 3, Issue 2*, pages 124–141, 1999.
- [Ergun et al. 2003] Ö. Ergun , J.B. Orlin , and A. Steele-Feldman . « Creating very large scale neighborhoods out of smaller ones by compounding

-
- moves : A study on the vehicle routing problem ». *Working paper, Massachusetts Institute of Technology*, 2003.
- [Fischetti et al. 1999] M. Fischetti , A. Lodi , and P. Toth . « A Branch-and-Cut Algorithm for the Multiple Depot Vehicle Scheduling Problem ». *Dipartimento di Elettronica e Informatica, Università di Padova, Italy*, 1999.
- [Fisher et al. 1981] M.L. Fisher and R. Jaikumar . « A Generalized Assignment Heuristic for the Vehicle Routing ». *Networks 11*, pages 109–124, 1981.
- [Fisher et al. 1997] M.L. Fisher , K. J Rnsten , and O.B.G. Madsen . « Vehicle Routing with Time Windows : two optimization algorithms ». *Operations Research 45 (3)*, pages 488–492, 1997.
- [Fisher 1995] M.L. Fisher . « Vehicle Routing ». *Handbooks in Operations Research and Management Science 8, North Holland, Amsterdam*, pages 1–33, 1995.
- [Flood 1956] M. M. Flood . « The Traveling Salesman Problem ». *Operations Research 4*, pages 61–75, 1956.
- [Fogel et al. 1966] L. J. Fogel , A. J. Owens , and M. j. Walsh . « Artificial Intelligence through Simulated Evolution ». *John Wiley*, 1966.
- [Friesleben et al. 1993] B. Friesleben and M. Hartfelder . « Optimisation of genetic algorithms by genetic algorithms ». In R. Albrecht, C. Reeves, and N. Steele, editors, *Artificial Neural Networks and Genetic Algorithms, Springer Verlag*, pages 392–399, 1993.
- [Gambardella et al. 2003] L.M. Gambardella , A.E. Rizzoli , F. Oliveriob , N. Casagrande , A.V. Donati , R. Montemanni , and E. Lucibello . « Ant Colony Optimization for vehicle routing in advanced logistics systems ». *MAS2003 – The International Workshop on Modeling & Applied Simulation*, pages 3–9, 2003.
- [Gendreau et al. 1997] M. Gendreau , G. Laporte , and F. Semet . « Solving an ambulance location model by tabu search ». *Location Sciences 5*, pages 75–88, 1997.
- [Gendreau et al. 1998] M. Gendreau , F. Guertin , J.-Y. Potvin , and R. Séguin . « Neighborhood search heuristics for a dynamic vehicle dispatching problem with pick-ups and deliveries ». Technical Report, CRT-98-10, Centre de Recherche sur les Transports, Université de Montréal, 1998.
- [Gendreau et al. 1999a] M. Gendreau , F. Guertin , J.-Y. Potvin , and E. Taillard . « Parallel Tabu Search for Real-Time Vehicle Routing and Dispatching ». *Transportation Science 33 (4)*, pages 381–390, 1999.
- [Gendreau et al. 1999b] M. Gendreau , G. Laporte , and R. Séguin . « Stochastic Vehicle Routing ». *European Journal of Operational Research 88*, pages 3–12, 1999.
- [Gendreau et al. 2001] M. Gendreau , G. Laporte , and F. Semet . « A dynamic model and parallel tabu search heuristic for real-time ambulance relocation ». *Parallel Computing 27*, pages 1641–1653, 2001.

- [Ghiani et al. 2003] G. Ghiani , F. Guerriero , G. Laporte , and R. Musmanno . « Real-Time Vehicle Routing : Solution Concepts, Algorithms and Parallel Computing Strategies ». *European Journal of Operational Research* 151, pages 1–11, 2003.
- [Gillett et al. 1974] B.E. Gillett and L.R. Miller . « A heuristic algorithm for the Vehicle Dispatch Problem ». *Operations Research* 22, pages 340–349, 1974.
- [Glover 1977] F. Glover . « Heuristics for integer programming using surrogate constraints ». *Decision Science* 13 (5), pages 156–166, 1977.
- [Glover 1986] F. Glover . « Future paths for Integer Programming and Links to Artificial Intelligence ». *Computers and Operations Research*, 5, pages 533–549, 1986.
- [Goldberg et al. 1985] D.E. Goldberg and R. Lingle . « Alleles, loci, and the tsp ». *In Proceedings of the First International Conference on Genetic Algorithms*, pages 154–159, 1985.
- [Goldberg et al. 1992] D. E. Goldberg , K. Deb , and J. H. Clark . « Genetic algorithms, noise, and the sizing of populations ». *Complex Systems*, 6, pages 333–362, 1992.
- [Goldberg 1989] D.E. Goldberg . « Genetic algorithms in search ». *Optimisation and Machine Learning*, 1989.
- [Goss et al. 1989] S. Goss , S. Aron , J.-L. Deneubourg , and J.M. Pasteels . « Self organized shortcuts in the argentine ant ». *Naturwissenschaften*, Vol. 76, pages 579–581, 1989.
- [Goupy 1997] J. Goupy . « Plans d’expériences ». Technical Report, Techniques de l’Ingénieur, 1997.
- [Goupy 2003] J. Goupy . *Introduction aux plans d’expériences*. Dunod - Industrie et technologie, 2003.
- [Grefenstette et al. 1985] J. Grefenstette , R. Gopal , R. Rosmaita , and D. Gucht . « Genetic algorithms for the traveling salesman problem ». *In Proceedings of the Second International Conference on Genetic Algorithms*. Lawrence Erlbaum Associates, Mahwah, NJ, pages 160–165, 1985.
- [Grefenstette 1986] J. J. Grefenstette . « Optimization of control parameters for genetic algorithms ». *IEEE Transactions on Systems, Man, and Cybernetics*, Volume SMC-16, pages 122–128, 1986.
- [Gulay et al. 1999] B. Gulay and D. Ozgur . « A tabu search algorithm for the vehicle routing problem ». *Computers and Operations Research* 26 (3), pages 255–270, 1999.
- [Guntsch et al. 2000] M. Guntsch , J. Branke , and S. Hartmut . « ACO strategies for dynamic TSP ». *Second International Workshop on Ant Algorithms*, pages 59–62, 2000.
- [Hajek 1988] B. Hajek . « Cooling schedules for optimal annealing ». *MOR* 13, pages 311–329, 1988.

-
- [Hapke et al. 2003] M. Hapke and P. Wesolek . « Handling imprecision and flexible constraints in vehicle routing problems : fuzzy approach ». *Raport RA-005/2003, Politechnika Poznanska*, 2003.
- [Hertz et al. 1994] A. Hertz , B. Jaumard , C. Ribeiro , and W. Formosinho Filho . « A multicriteria tabu search approach to cell formation problems in group technology with multiple objectives ». *RAIRO Recherche Opérationnelle 28(3)*, pages 303–328, 1994.
- [Holland 1975] J. Holland . « Adaptation in natural and artificial systems ». *University of Michigan Press, Ann Arbor, Canberra ACT 2601, Australia*, 1975.
- [Hombberger et al. 1999] J. Hombberger and H. Gehring . « Two Evolutionary Metaheuristics for the Vehicle Routing Problem with Time Windows ». *INFOR 37*, pages 297–318, 1999.
- [Housroum et al. 2003] H. Housroum , G. Goncalves , R. Dupas , and T. Hsu . « Une approche évolutionniste du problème de la gestion dynamique de tournées de véhicule avec fenêtres de temps ». *CPI'03, Meknès, Maroc*, Octobre 2003.
- [Housroum et al. 2004] H. Housroum , G. Goncalves , R. Dupas , and T. Hsu . « An hybrid GA approach for solving the Dynamic Vehicle Routing Problem with Time Windows ». *Franco IV, Fribourg, Suisse*, 2004.
- [Ichoua et al. 2003] S. Ichoua , M. Gendreau , and J.-Y. Potvin . « Vehicle dispatching with time-dependent travel times ». *European Journal of OR, 144*, pages 379–396, 2003.
- [Ishibuchi et al. 1998] H. Ishibuchi and T. Murata . « A multi-objective genetic local search algorithm and its application to flowshop scheduling ». *IEEE Transaction on Systems, Man and Cybernetics, Part C*, pages 392–403, August 1998.
- [Jacobs-Blecha et al. 1993] C. Jacobs-Blecha and M. Goetschalckx . « The Vehicle Routing Problem with Backhauls : An Optimization Based Approach ». *Proceedings of the 2nd Industrial Engineering Research Conference*, pages 504–509, 1993.
- [Jason et al. 2002] G. D. Jason and G. M. Konstantinos . « An experimental study of benchmarking functions for Genetic Algorithms ». *International Journal Computer Math, Vol. 79(4)*, pages 403–416, 2002.
- [Jörnsten et al. 1986] K. Jörnsten , O. Madsen , and B. Sorensen . « Exact solution of the Vehicle Routing and Scheduling Problem with Time Windows by Variable Splitting ». *Research Report 5/1986, IMSOR, The Technical University of Denmark, Lyngby, Denmark*, 1986.
- [Kargupta et al. 1992] H. Kargupta , K. Deb , and D. E. Goldberg . « Ordering genetic algorithms and deception ». *Parallel Problem solving from nature*, pages 47–56, 1992.
- [Kilby et al. 1998a] P. Kilby , P. Prosser , and P. Shaw . « Dynamic VRPs : A Study of Scenarios ». *CSIRO, Canberra ACT 2601, Australia*, September 1998.

- [Kilby et al. 1998b] P. Kilby , P. Prosser , and P. Shaw . « Guided Local Search for the Vehicle Routing Problem with Time Windows ». *META-HEURISTICS Advances and Trends in Local Search Paradigms for Optimization*, Kluwer Academic Publishers, Boston, pages 473–486, 1998.
- [Kohl et al. 1997] N. Kohl and O. Madsen . « An optimization algorithm for the Vehicle Routing Problem with Time Windows based on Lagrangean Relaxation ». *Operations Research* 45 (3), pages 395–406, 1997.
- [Kolen et al. 1987] A. Kolen , H. Rinoy Kan , and H. Trienekens . « Vehicle Routing with Time Windows ». *Operations Research* 35, pages 266–273, 1987.
- [Kontoravdis et al. 1992] G. Kontoravdis and J. Bard . « Improved heuristics for the Vehicle Routing Problem with Time Windows ». *Working paper, Operations Research Group, University of Texas at Austin, Austin*, 1992.
- [Kontoravdis et al. 1995] G. Kontoravdis and J. Bard . « A GRASP for the Vehicle Routing Problem with Windows ». *ORSA Journal on Computing* 7, pages 10–23, 1995.
- [Koskosidis et al. 1992] Y.A. Koskosidis , W.B. Powell , and M. Solomon . « An optimization-based heuristic for Vehicle Routing and Scheduling with soft Time Window Constraints ». *Transportation Science* 26, pages 69–85, 1992.
- [Krumke et al. 2003] S.O. Krumke , W.E. de Paepe , D. Poensgen , and L. Stougie . « News from the Online Traveling Repairman ». *Theoretical Computer Science* 295, pages 279–294, 2003.
- [Laporte et al. 1986] G. Laporte , H. Mercure , and Y. Nobert . « A exact algorithm for the asymmetrical capacitated vehicle routing problem ». *Networks*, 16, pages 33–46, 1986.
- [Laporte et al. 1989] G. Laporte , M. Dror , and P. Trudeau . « Vehicle routing with stochastic demands : properties and solution frameworks ». *Transportation Science*, 23, pages 166–176, 1989.
- [Laporte et al. 2000] G. Laporte , M. Gendreau , J-Y. Potvin , and F. Semet . « Classical and modern heuristics for the vehicle routing problem ». *International Transactions in Operational Research* 7, pages 285–300, 2000.
- [Larsen et al. 2004] A. Larsen , O. Madsen , and M. Solomon . « The A-priori Dynamic Traveling Salesman Problem with Time Windows ». *Transportation Science*, 38 (4), pages 459–472, 2004.
- [Larsen 2001] A. Larsen . « *The Dynamic Vehicle Routing Problem* ». PhD thesis, IMM - Danemark, 2001.
- [Lau et al. 2003] H.C. Lau , M. Sim , and K.M. Teo . « Vehicle routing problem with time windows and a limited number of vehicles ». *European Journal of Operational Research* 148, pages 559–569, 2003.

-
- [LeBouthillier 2000] A. LeBouthillier . « Modélisation UML pour une architecture coopérative appliquée au problème de tournées de véhicules avec fenêtres de temps ». Technical Report, Département d’informatique et de recherche opérationnelle, Faculté des arts et des sciences, Université de Montréal, Canada, Avril 2000.
- [Li et al. 2004] F. Li , B.L. Golden , and E.A. Wasil . « Very large-scale vehicle routing : New test problems, algorithms, and results ». *Computers & Operations Research* 32, pages 1165–1179, 2004.
- [Lin 1965] S. Lin . « Computer solutions of the traveling salesman problem ». *Bell System Technical Journal* 44, pages 2245–2269, 1965.
- [Lindsay et al. 1997] R.K. Lindsay , R.L. Riolo , and A. Wu . « Empirical observation on the roles of crossover and mutation ». *Proceedings of the Seventh International Conference on Genetic Algorithms*, pages 362–369, 1997.
- [Luiz et al. 1998] S. Luiz , S.V. Dalessandro , M. Lúcia , and A.O. Victor . « A Parallel Evolutionary Algorithm for the Vehicle Routing Problem with Heterogeneous Fleet ». *Future Generation Computer Systems* 14, pages 285–292, 1998.
- [Lund et al. 1996] K. Lund , O. Madsen , and J. M. Rygaard . « Vehicle Routing Problems with Varying Degrees of Dynamism ». Technical Report, IMM, The Department of Mathematical Modelling, Technical University of Denmark, 1996.
- [Madsen 1990] O.B.G. Madsen . « Lagrangean Relaxation and Vehicle Routing ». *Working paper. IMSOR, The Technical University of Denmark, Lyngby, Denmark*, 1990.
- [Mahfouz 1999] S. Mahfouz . « *Design Optimization of Structural Steelwork* ». PhD thesis, PhD thesis, University of Bradford, 1999.
- [Mester et al. 2005] D. Mester and O. Bräysy . « Active guided evolution strategies for the large scale vehicle routing problems with time windows ». *Computers & Operations Research* 32, pages 1593–1614, 2005.
- [Michalewicz 1994] Z. Michalewicz . *Genetic Algorithms + Data Structures = Evolution Programs*. Berlin : Springer-Verlag, seconde édition, 1994.
- [Miller 1995] D.L Miller . « A matching based exact algorithm for capacitated vehicle routing problems ». *ORSA Journal on Computing*, 7, pages 1–9, 1995.
- [Mitchell 1998] M. Mitchell . *An Introduction to Genetic Algorithms*. MIT Press, 1998.
- [Mitrovic-Minic et al. 2004a] S. Mitrovic-Minic , R. Krishnamurti , and G. Laporte . « Double-horizon based heuristics for the dynamic pickup and delivery problem with time windows ». *Transportation Research* 38, pages 669–685, 2004.
- [Mitrovic-Minic et al. 2004b] S. Mitrovic-Minic and G. Laporte . « Waiting Strategies for the Dynamic Pickup and Delivery Problem with Time Windows ». *Transportation Research Part B*, 38, pages 635–655, 2004.

- [Mitrovic-Minic 2001] S. Mitrovic-Minic . « *The Dynamic Pickup and Delivery Problem with Time Windows* ». PhD thesis, Simon Fraser University, 2001.
- [Mole et al. 1976] R.H. Mole and S.R. Jameson . « A sequential route-building algorithm employing a generalized saving criterion ». *Operational Research 27*, pages 503–511, 1976.
- [Montamenni et al. 2002] R. Montamenni , L.M. Gambardella , A.E. Rizzoli , and A.V. Donati . « A new algorithm for a Dynamic Vehicle Routing Problem based on Ant Colony System ». *IDSIA, Switzerland*, 2002.
- [Montamenni et al. 2003] R. Montamenni , L.M. Gambardella , A.E. Rizzoli , and A.V. Donati . « A new algorithm for a Dynamic Vehicle Routing Problem based on Ant Colony System ». *IDSIA-23-02, presented at ODYSSEUS 2003*, 2003.
- [Nanry et al. 2000] W.P. Nanry and J.W. Barnes . « Solving the Pick-up and Delivery Problem with Time Windows using reactive tabu search ». *Transportation Research 34*, pages 107–121, 2000.
- [Nowostawski et al. 1999] M. Nowostawski and R. Poli . « Parallel genetic algorithm taxonomy ». In *L. C. Jain, editor, Proceedings of the Third International conference on knowledge-based intelligent information engineering systems (KES'99)*, IEEE, pages 88–92, 1999.
- [Or 1976] I. Or . « Travelling salesman-type combinatorial problems and their relation to the logistics of blood banking ». *PhD thesis, Departement of industrial engineering and management sciences, NorthWestern University, Evanston, Il*, 1976.
- [Park et al. 1997] Y.-B. Park and S.-H. Song . « Vehicle Scheduling Problems with Time-Varying Speed ». *Computers & Industrial Engineering 33*, pages 853–856, 1997.
- [Park et al. 1998] M.-W. Park and Y.-D. Kim . « A Systematic Procedure for Setting Parameters in Simulated Annealing Algorithms ». *Computers & Operations Research 24(3)*, pages 207–217, 1998.
- [Park 2000] Y.-B. Park . « A solution of the bicriteria vehicle scheduling problems with time and area-dependent travel speeds ». *Computers and Industrial Engineering 38*, pages 173–187, 2000.
- [Parsons et al. 1997] R. Parsons and M. Johnson . « A Case Study in Experimental Design Applied to Genetic Algorithms with Applications to DNA Sequence Assembly ». *American Journal of Mathematical and Management Sciences 17(3/4)*, pages 369–396, 1997.
- [Pereira et al. 2002] F. Pereira , J. Tavares , P. Machado , and E. Costa . « GVR : a New Genetic Representation for the Vehicle Routing Problem ». *Artificial Intelligence and Cognitive Science : 13th, Irish Conference Proceedings*, pages 95–102, 2002.
- [Pham 1994] Q. T. Pham . « Competitive evolution : A Natural Approach to Operator Selection ». *School of Chemical Engineering and Industrial Chemistry University of New South Wales, Sydney, Australia*, November 1994. Evolutionary Computation Workshop.

-
- [Pillet 1992] M. Pillet . « Introduction aux plans d'expériences par la méthode Taguchi ». *Les Editions d'Organisations*, 1992.
- [Pillet 1997] M. Pillet . *Les plans d'expériences par la méthode TAGUCHI*. Les éditions d'organisation, 1997.
- [Ponnambalam et al. 2002] S. G. Ponnambalam , N. Jawahar , and B. S. Kumar . « Estimation of Optimum Genetic Control Parameters for Job Shop Scheduling ». *International Journal of Advanced Manufacturing Technology* 19, pages 224–234, 2002.
- [Potvin et al. 1993] J.-Y. Potvin and J.-M. Rousseau . « A parallel route building algorithm for the Vehicle Routing and Scheduling Problem with Time Windows ». *European Journal of Operational Research* 66, pages 331–340, 1993.
- [Potvin et al. 1996a] J.-Y. Potvin and S. Bengio . « The Vehicle Routing Problem with Time Windows - part 2 : genetic search ». *INFORMS Journal on Computing* 8, pages 165–172, 1996.
- [Potvin et al. 1996b] J.-Y. Potvin , T. Kervahut , B.L. Garcia , and J.-M. Rousseau . « The Vehicle Routing Problem with Time Windows - part 1 : tabu search ». *ORSA Journal on Computing* 8 (2), pages 158–164, 1996.
- [Prins 2001] C. Prins . « A simple and effective evolutionary algorithm for the vehicle routing problem ». *MIC'2001 (4th Metaheuristics International Conference)*, 143-147, 16-20/07/01, Porto, Portugal, 2001.
- [Prins 2002] C. Prins . « Efficient heuristics for the heterogeneous fleet multi-trip vehicle routing problem ». *Journal of Mathematical Modelling and Algorithms*, 1(2), pages 135–150, 2002.
- [Prins 2004] C. Prins . « A simple and effective evolutionary algorithm for the vehicle routing problem ». *Computers and Operations Research*, 31(12), pages 1985–2002, 2004.
- [Radcliffe 1991] N. J. Radcliffe . « Formal analysis and random respectful recombination ». *Proceedings of the fourth international conference on Genetic Algorithms*, pages 222–229, 1991.
- [Ralphs 2003] T. K. Ralphs . « Parallel branch and cut for capacitated vehicle routing ». *Parallel Computing archive* 29 (5), pages 607–629, 2003.
- [Rego et al. 1994] C. Rego and C. Roucairol . « Le Problème de Tournées de Véhicules : Etude et Résolution Approchée ». Technical Report, inria, Février 1994.
- [Rego 2001] C. Rego . « Node ejection chains for the Vehicle Routing Problem : Sequential and Parallel Algorithms ». *Parallel Computing* 27 (3), pages 201–222, 2001.
- [Reimann et al. 2004] M. Reimann , K. Doerner , and R.F. Hartl . « D-Ants : Savings based ants divide and conquer the vehicle routing problem ». *Computers & Operations Research* 31 (4), pages 563–591, 2004.

- [Robertson et al. 1998] S. Robertson , B. Golden , and E. Wasil . « Neural Network Models for Initial Public Offerings ». *Neurocomputing 18*, pages 165–182, 1998.
- [Rochat et al. 1995] Y. Rochat and E. Taillard . « Probabilistic diversification and intensification in local search for Vehicle Routing ». *Journal of Heuristics 1*, pages 147–167, 1995.
- [Roy et al. 1984] S. Roy , J.-M. Rousseau , G. Lapalme , and J. A. Ferland . « Routing and scheduling of transportation services for disabled : summary report ». Technical Report, Centre de recherche sur les transports, Université de Montréal, June 1984.
- [Russell 1995] R.A. Russell . « Hybrid heuristics for Vehicle Routing Problem with Time Windows ». *Transportation Science 29*, pages 156–166, 1995.
- [Sandgren 1994] E. Sandgren . « Advances in design optimization, chapter Multicriteria design optimization by goal programming ». *Chapman and Hall*, pages 225–265, 1994.
- [Savlesberg et al. 1991] M. Savlesberg and M. Sol . « DRIVE : Dynamic routing of independent vehicles ». *Operations Research, 46(4)*, pages 474–490, 1991.
- [Schaffer 1985] J.D. Schaffer . « Multiple objective optimization with vector evaluated genetic algorithms ». *In Proceedings of the First International Conference on Genetic algorithms*, pages 93–100, 1985.
- [Schneider 2002] J. Schneider . « The time-dependent traveling salesman problem ». *Physica A 314*, pages 151–155, 2002.
- [Siarry et al. 2003] P. Siarry , J. Dréo , A. Pétrowski , and E. Taillard . *Métaheuristiques pour l'optimisation difficile*. Eyrolles, 2003.
- [Solomon et al. 1988] M.M. Solomon and J. Desrosiers . « Time Window Constrained Routing and Scheduling Problems ». *Transportation Science 22*, pages 1–13, 1988.
- [Solomon 1986] M.M. Solomon . « On the worst-case performance of some heuristics for the Vehicle Routing and Scheduling Problem with Time Window Constraints ». *Networks 16*, pages 161–174, 1986.
- [Solomon 1987] M.M. Solomon . « Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints ». *Operations Research 35*, pages 254–265, 1987.
- [Sushil et al. 1999] J. L. Sushil , X. Yin , and Z. Y. Yuan . « Multiple Vehicle Routing with Time Windows Using Genetic Algorithms ». *In Proceedings of IEEE Congress on Evolutionary Computation. Piscataway : IEEE Press*, pages 1804–1808, 29 January 1999.
- [Taillard et al. 1997] E. Taillard , P. Badeau , M. Gendreau , F. Guertin , and J.-Y. Potvin . « A Tabu Search Heuristic for the Vehicle Routing Problem with Soft Time Windows ». *Transportation Science 31*, pages 170–186, 1997.
- [Taillard 1994] E. Taillard . « Parallel iterative search methods for vehicle routing problems ». *NETWORKS 23(8)*, pages 661–673, 1994.

-
- [Taillard 1998] E. Taillard . « *Programmation à mémoire adaptative et algorithmes pseudogloutons : nouvelles perspectives pour les métaheuristiques* ». thèse d'habilitation, Université de Versailles, France, 1998.
- [Taillard 1999] E. Taillard . « A heuristic column generation method for the heterogeneous fleet VRP ». *RAIRO-Operations Research*, 33, pages 1–14, 1999.
- [Talbi 2001] E. Talbi . « Métaheuristiques pour l'optimisation combinatoire multi-objectif : Etat de l'art ». *Equipe OPAC (Optimisation PARallèle Coopérative), Laboratoire d'Informatique Fondamentale de Lille Université de Lille 1, France*, 2001. cite-seer.ist.psu.edu/382692.html.
- [Tan et al. 2001] K.C. Tan , L.H. Lee , and K. Ou . « Hybrid Genetic Algorithms in Solving Vehicle Routing Problems with Time Window Constraints ». *Asia-Pacific Journal of Operational Research* 18, pages 121–130, 2001.
- [Tarantilis et al. 2002] C.-D. Tarantilis and C.T. Kiranoudis . « Bone Route : An adaptive memory-based method for effective fleet management ». *Annals of Operations Research*, 115, pages 227–241, 2002.
- [Tate et al. 1993] D. M. Tate and E. A. Smith . « Expected allele coverage and the role of mutation in genetic algorithms ». *Proceedings of the fifth international conference on Genetic Algorithms*, pages 31–37, 1993.
- [Tavares et al. 2003] J. Tavares , F. B. Pereira , P. Machado , and E. Costa . « Crossover and Diversity : A Study about GVR ». *In Proceedings of the Analysis and Design of Representations and Operators (ADoRo'2003) a bird-of-a-feather workshop at the 2003 Genetic and Evolutionary Computation Conference (GECCO-2003), Chicago, Illinois, USA, 12-16 July 2003*.
- [Thangiah et al. 1994] S.R. Thangiah , I. Osman , and T. Sun . « Hybrid Genetic Algorithm, Simulated Annealing and Tabu Search Methods for Vehicle Routing Problems with Time Windows ». *Working Paper UKC/IMS/OR94/4, Institute of Mathematics and Statistics, University of Kent, Canterbury*, 1994.
- [Thangiah 1995] S.R. Thangiah . « Vehicle Routing with Time Windows using genetic algorithms ». *Handbook of Genetic Algorithms : new frontiers, Vol. 2*, pages 253–277, 1995.
- [Tongchim et al. 2002] S. Tongchim and P. Chongstitvatana . « Parallel genetic algorithm with parameter adaptation ». *Information Processing Letters* 82, pages 47–54, 2002.
- [Toth et al. 2002] P. Toth and D. Vigo . *The Vehicle Routing Problem*. SIAM - Monographs on Discrete Mathematics and Applications, 2002.
- [Toth et al. 2003] P. Toth and D. Vigo . « The granular tabu search and its application to the vehicle routing problem ». *INFORMS Journal on Computing*, 15, pages 333–346, 2003.

- [Wang et al. 2004] L. Wang , L. Zhang , and D.-Z. Zheng . « The ordinal optimisation of genetic control parameters for flow shop scheduling ». *International Journal of Advanced Manufacturing Technology* 2, pages 812–819, 2004.
- [Xu et al. 1996] J. Xu and J. Kelly . « A Network Flow-based Tabu Search Heuristic for the Vehicle Routing Problem ». *Transportation Science* 30, pages 379–393, 1996.
- [Yang et al. 2004] J. Yang , P. Jaillet , and H. S. Mahmassani . « Real-Time Multi-Vehicle Truckload Pick-Up and Delivery Problems ». *Transportation Science*, 38 (2), pages 135–148, 2004.
- [Zhu 1999] K.Q. Zhu . « *Heuristic Methods For Vehicle Routing Problem with Time Windows* ». PhD thesis, National University of Singapore, Departement of Electrical Engineering, May 12 1999.
- [Zhu 2000] K. Q. Zhu . « A new genetic algorithm for VRPTW ». *International Conference on Artificial Intelligence, Las Vegas, USA*, April 13 2000.
- [Zhu 2003] K. Q. Zhu . « A Diversity-controlling Adaptive Genetic Algorithm for the Vehicle Routing Problem with Time Windows ». *15th IEEE International Conference on Tools with Artificial Intelligence ICTAI*, pages 176–183, 2003.

Index

- écart-type, 100
- événements, 75
- 1-Opt, 69
- 2-Opt, 70

- adaptatif, 83
- ADW, 49
- AE, 56
- aléatoire, 65
- algorithme génétique, 55, 56
- Algorithmes génétiques, 17
- arrivée d'un nouveau client, 76
- auto-adaptatif, 83

- BENCH, 47
- benchmark, 113, 119, 120, 124, 125
- boite noire, 84
- Branch (&) Cut, 14
- Branch & Bound, 13
- BUSIEST, 46

- Charles Darwin, 57
- chromosome, 57, 59, 60
- Clarke (&) Wright, 15
- Classification, 35, 41
- colonie de fourmis, 24
- complet, 87, 97
- Complexité, 10
- composant AG, 72
- composant GE, 72
- contrôle, 81
- contraintes, 7, 27
- Criblage, 87
- criblage, 90, 92
- croisement, 19, 66
- CURRENT, 46

- décodage, 62, 64
- démarche, 90
- déterministe, 37, 82
- degré de dynamisme, 39

- DF, 49
- direct, 59
- dispersion, 23
- distance, 89
- DPDPTW, 49, 110
- DTP, 34
- DTRP, 110
- DTSPWT, 46, 110
- DVRP, 11, 35, 43, 109
- DVRPTW, 48, 55, 109
- DW, 49
- dynamique, 34, 37

- effet, 85, 92, 100
- Elitisme, 19, 71

- facteurs, 85
- factoriel, 87
- FCFS, 44
- fin de service, 75
- fitness, 59, 60
- Fonction, 60
- Fonction Objectif, 61, 62
- fonctionnement du simulateur, 73
- Formulation, 8, 27, 39
- fractionnaire, 87, 101

- génération, 89
- GEN, 44

- Heuristiques, 13, 14
- HI-REQ, 46

- indirect, 59
- insertion, 47, 62, 76
- Insertion séquentielle, 15
- interactions, 74, 85
- IX, 66

- m-VRP, 12
- Métaheuristiques, 13, 17
- méthodes approchées, 13, 29

méthodes de résolution, 12
méthodes exactes, 13, 28
MDVRP, 12
mod TSP, 44
multi-objectifs, 61
mutation, 19, 69
MX1T/D, 68
MYOPT, 47

NEAREST, 46
niveaux, 85
NN, 44
NS, 47

objectif, 8
OPTUN, 47
Or-Opt, 70
orthogonal, 87
OVRP, 12

paramètres, 79, 88
PART, 44
PFIH, 65
plans d'expériences, 84
Plate-forme, 109
PMO, 61
PMX, 67
population, 57, 59, 65
Programmation Dynamique, 14
PVRP, 11

réglage, 79, 81
réponses, 85
résultats, 109, 116, 119, 122, 124, 126
ranking, 65
Recherche locale, 16
Recherche Tabou, 22
Recherche Tabou Parallèle, 50
Recuit simulé, 20
REOPT, 47
Reproduction, 65
roulette, 65

Sélection, 19, 65
SE, 47
Simulation, 72
solution, 59
SQM, 44
STATIC, 47

statique, 37
stochastique, 11, 37
Stratégies, 44
SVRP, 11
swap, 69

tournée, 60
tournoi, 65
TRP, 12
TSP, 12, 44

Variantes, 10
Vehicle Routing Problem, 7
VRP, 7, 34
VRPB, 10
VRPHF, 12
VRPPD, 12
VRPSD, 12
VRPTW, 10, 26

WF, 49

Résumé

Ces dernières années les systèmes de transport utilisés pour le ramassage et la distribution de biens ou de services ont fait l'objet de nombreuses études dans la communauté scientifique. De nos jours, la plupart des systèmes de transport doivent pouvoir fonctionner en respectant des contraintes temporelles strictes et ceci en s'adaptant aux aléas du problème. En effet, les clients ou partenaires d'une entreprise exigent de celle-ci une qualité de service garantie (i.e. délais à respecter). De plus, l'environnement dans lequel une entreprise doit évoluer est bien souvent incertain et donc sa réactivité est également un atout important. Ceci a conduit à définir des modèles de pilotage des systèmes de transport dits **dynamiques** dans lesquels une partie des données est considérée comme dépendante du temps.

Le domaine dans lequel s'inscrit nos travaux, concerne le problème classique de l'élaboration de tournées de véhicules (**VRP** : *Vehicle Routing Problem*). Celui-ci consiste à construire des tournées de coût minimal afin que de visiter une fois et une seule fois un ensemble de clients géographiquement distribué. Le travail présenté dans cette thèse traite plus précisément de la résolution du problème de l'élaboration dynamique de tournées de véhicules avec fenêtres de temps (**DVRPTW** : *Dynamic Vehicle Routing Problem with Time Windows*) et de quelques unes de ses extensions.

L'objectif de ce travail était double. D'une part, il s'agissait de montrer qu'une approche de type évolutionniste était utilisable dans un cadre dynamique. D'autre part, il fallait vérifier que les performances que l'on pouvait attendre de ce type d'approche étaient comparables voire supérieures à celles des meilleures techniques utilisées à ce jour.

Pour atteindre ces objectifs, nous avons utilisé la technique des **Algorithmes Génétiques (AG)** pour définir un planificateur dirigé par les événements. Ce planificateur cherche à optimiser dynamiquement le problème après chaque événement significatif (" arrivée d'une nouvelle demande " et " fin de service chez un client ") survenant tout au long de la journée d'ouverture. L'algorithme génétique est basé pour cela sur des chromosomes de taille variable dans le temps permettant de prendre en compte l'arrivée de nouveaux clients pendant l'exécution effective des tournées de véhicules.

L'efficacité des approches AG pose la question délicate du réglage de certains paramètres par rapport au problème à traiter. Nous avons utilisé un réglage " a priori " de ces paramètres en utilisant la technique des plans d'expériences.

Le dernier point de cette thèse porte sur une plate-forme développée en JAVA pour évaluer notre approche et comparer les résultats avec ceux obtenus par d'autres approches.

Mots clés : Problème DVRPTW, Algorithme génétique " en ligne ", Simulation, Plans d'expériences.

