



HAL
open science

Optimisation de plans d'actions multi-objectifs dans le secteur social et médico-social

Brahim Chabane

► **To cite this version:**

Brahim Chabane. Optimisation de plans d'actions multi-objectifs dans le secteur social et médico-social. Modélisation et simulation. Université d'Angers, 2017. Français. NNT : 2017ANGE0035 . tel-01748720

HAL Id: tel-01748720

<https://theses.hal.science/tel-01748720>

Submitted on 29 Mar 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Thèse de Doctorat

Brahim CHABANE

*Mémoire présenté en vue de l'obtention du
grade de Docteur de l'Université d'Angers
sous le sceau de l'Université Bretagne Loire*

École doctorale : Sciences et technologies de l'information, et mathématiques

Discipline : Informatique, section CNU 27

Unité de recherche : Laboratoire d'Étude et de Recherche en Informatique d'Angers (LERIA)

Soutenu le 06 décembre 2017

Optimisation de plans d'actions multi-objectifs dans le secteur social et médico-social

JURY

Rapporteurs :	M. Gilles VENTURINI , Professeur, Université de Tours M^{me} Laetitia JOURDAN , Professeur, Université Lille 1
Examineur :	M. Patrick SIARRY , Professeur, Université Paris-Est
Invité :	M. Rachid NAITALI , Chargé de recherche, GePI-Conseil
Directeur de thèse :	M. Jin-Kao HAO , Professeur, Université d'Angers
Co-directeur de thèse :	M. Matthieu BASSEUR , Maître de conférence HDR, Université d'Angers

Thèse de Doctorat

Brahim CHABANE

*Mémoire présenté en vue de l'obtention du
grade de Docteur de l'Université d'Angers
sous le sceau de l'Université Bretagne Loire*

École doctorale : Sciences et technologies de l'information, et mathématiques

Discipline : Informatique, section CNU 27

Unité de recherche : Laboratoire d'Étude et de Recherche en Informatique d'Angers (LERIA)

Soutenue le 06 décembre 2016

Optimisation de plans d'actions multi-objectifs dans le secteur social et médico-social

JURY

Rapporteurs :	M. Gilles VENTURINI , Professeur, Université de Tours M^{me} Laetitia JOURDAN , Professeur, Université Lille 1
Examineur :	M. Patrick SIARRY , Professeur, Université Paris-Est
Invité :	M. Rachid NAITALI , Chargé de recherche, GePI-Conseil
Directeur de thèse :	M. Jin-Kao HAO , Professeur, Université d'Angers
Co-directeur de thèse :	M. Matthieu BASSEUR , Maître de conférence HDR, Université d'Angers

Remerciements

Mes premiers remerciements s'adressent au *P^r* Jin-Kao Hao et au *D^r* Matthieu Basseur pour avoir encadré ma thèse durant toutes ces années. Je les remercie pour leur présence et leur patience. Sans leur aide, je n'aurais jamais pu terminer cette thèse.

Je tiens à remercier également M. Rachid Naitali pour sa confiance, sa présence et sa patience. Ce travail n'aurait pas été possible sans son soutien et son engagement.

Je remercie chaleureusement tous les membres du jury : *P^r* Patrick Siarry, *P^r* Laetitia Jourdan et *P^r* Gilles Venturini. A tous je leur exprime ma sincère gratitude pour l'attention qu'ils ont porté à mon travail.

Je remercie toutes les personnes avec qui j'ai pris beaucoup de plaisir à travailler durant toutes ces quatre années. Je remercie tous les membres du laboratoire LERIA et de la société GePI pour l'accueil qu'ils m'ont réservé. Je remercie particulièrement Anouck pour sa relecture et ses suggestions.

J'exprime mes remerciements et ma gratitude à toute ma famille. Je suis particulièrement redevable à mes parents, que je ne pourrai jamais remercier assez. Si j'ai pu faire des études c'est d'abord grâce à leur présence, leur engagement et leur sacrifice.

Enfin, je remercie Louiza pour son soutien indéfectible et sa sérénité. Je la remercie pour sa présence quotidienne à mes côtés.

Table des matières

Introduction	7
1 Le contexte pratique et théorique	11
1.1 Introduction	12
1.2 Le secteur social et médico-social (SSMS)	12
1.2.1 Description du SSMS	12
1.2.2 Évolution du SSMS	13
1.2.3 Organisation du SSMS	18
1.2.4 Les principaux acteurs du SSMS	20
1.2.5 Missions du SSMS selon la loi du 2 janvier 2002	23
1.2.6 L'informatique dans le SSMS et le besoin d'outils d'aide à la décision	24
1.2.7 Défis pratiques et scientifiques du SSMS	24
1.3 Optimisation multi-objectif (OMO)	28
1.3.1 Définitions	28
1.3.2 Quelques concepts sur l'optimisation multi-objectif	32
1.4 Conclusion	35
2 Le POPAMO	37
2.1 Introduction	38
2.2 Présentation du POPAMO	39
2.2.1 Les objectifs	41
2.2.2 Les actions	41
2.2.3 Les contraintes	42
2.2.4 Les sorties	42
2.3 Modélisation du POPAMO	42
2.3.1 Les actions (espace décisionnel)	43
2.3.2 Les objectifs (espace objectif)	43
2.3.3 Les plans d'actions	43
2.3.4 Les contraintes	43
2.3.5 Formulation du POPAMO	44
2.4 MSQualité pour le POPAMO	44
2.4.1 Aide à la définition des actions	45
2.4.2 Aide aux choix des actions	46
2.5 État de l'art	46
2.5.1 Problème d'optimisation de plans d'actions	46
2.5.2 Problème du sac-à-dos multi-objectif	47

2.6	Conclusion	52
3	Optimisation MO à base d'indicateurs de qualité	53
3.1	Introduction	54
3.2	Indicateurs de qualité	55
3.2.1	Définition	55
3.2.2	Quelques indicateurs	55
3.3	État de l'art	58
3.3.1	Intégration des préférences du décideur	58
3.3.2	Méthodes à base d'indicateurs de qualité	61
3.4	Approche proposée pour résoudre le POPAMO	63
3.4.1	Motivations	63
3.4.2	R2-IBMOLS	64
3.4.3	Expérimentations	66
3.4.4	Résultats expérimentaux	70
3.5	Conclusion	74
4	La dominance de Lorenz pour l'optimisation MO	77
4.1	Introduction	78
4.2	De l'équité à l'optimisation équitable	79
4.2.1	Définitions	80
4.2.2	État de l'art	81
4.2.3	Dominance de Lorenz d'ordre infini	82
4.3	Approches proposées pour résoudre le POPAMO	84
4.3.1	IOLD-R2-IBMOLS	84
4.3.2	IOLD-BMOLS	87
4.3.3	IOLD-EA	87
4.4	Expérimentations	89
4.4.1	Comparaison avec L-NSGA-II	89
4.4.2	Comparaison avec NSGA-III	90
4.4.3	Protocole d'expérimentation	90
4.4.4	Résultats expérimentaux	91
4.5	Conclusion	96
	Conclusion	97
	Liste des figures	101
	Liste des tableaux	103
	Liste des acronymes	104
	Liste des publications	106
	Bibliographie	107

Introduction

Le travail présenté dans cette thèse a été réalisé dans le cadre d'une Convention Industrielle de Formation par la REcherche (CIFRE) impliquant le laboratoire de recherche LERIA de l'université d'Angers et la société GéPI-conseil.

GéPi-conseil est un éditeur de logiciels dédiés à la gestion de la qualité dans le secteur social et médico-social. Elle a été créée en 2006 pour développer un outil de gestion d'évaluations pour répondre à une obligation légale¹ des établissements sociaux et médico-sociaux d'évaluer continuellement leurs fonctionnements. Aujourd'hui, GePi-conseil propose une suite complète de logiciels (appelée *MSQualité*) permettant de gérer toute la démarche qualité, conformément à la réglementation. Ce travail s'inscrit dans une logique de continuité et d'amélioration de *MSQualité* et vise à intégrer de nouveaux modules d'optimisation de la gestion de la qualité dans ce secteur.

Le but de cette étude est d'apporter une réponse à un problème pratique souvent rencontré par les directeurs des établissements du secteur social et médico-social, qui est l'élaboration de plans d'actions efficaces, dans un contexte budgétaire de plus en plus difficile. De ce fait, notre travail comporte deux volets. Le premier consiste à étudier et comprendre le contexte pratique du problème avec toutes les contraintes législatives, financières et humaines qui l'entourent, puis de le transcrire dans un modèle formel permettant sa résolution avec des algorithmes. Le deuxième volet consiste à proposer de nouvelles méthodes de résolution pour, d'un côté, mettre à disposition des directeurs de ce secteur un outil efficace pour les assister dans l'élaboration de leurs plans d'actions et, d'un autre côté, d'apporter de nouvelles idées et de nouvelles pistes de recherche pour l'optimisation multi-objectif.

Quel que soit le type de l'établissement et les besoins des personnes qui y sont accueillies, le directeur doit agir continuellement sur plusieurs aspects dans la gestion de son établissement et la prise en charge de ses résidents. Il doit mettre en place des plans d'actions améliorant à la fois les "*performances*" de l'établissement et la qualité de vie de ses résidents. De ce fait, son plan d'actions doit inclure des actions de différentes natures (des actions de protection administrative et judiciaire, des actions éducatives, des actions d'aide à l'insertion professionnelle, des actions d'assistance dans les divers actes de la vie, de soutien, de soins et d'accompagnement, des actions pour la prévention des risques sociaux et psychosociaux, etc.). Cependant, souvent les moyens financiers dont disposent les établissements ne permettent pas de mettre en œuvre toutes les actions nécessaires. Par conséquent, le choix des meilleures actions s'impose. De plus, à l'état actuel du fonctionnement de ces structures, certaines actions sont réalisées sans savoir préalablement quels seraient leurs impacts sur la vie de l'établissement, ce qui pourrait causer une perte d'argent et de temps, quand on investit dans des actions peu efficaces.

Le problème d'élaboration d'un plan d'actions optimal consiste donc à identifier les actions

¹Instituée par la loi du 2 janvier 2002

dont l'efficacité est déjà connue. Le fait de savoir préalablement l'impact que pourraient avoir les actions une fois mise en place nous permet d'être sûr que chaque euro dépensé contribuerait à l'amélioration de la qualité globale de l'établissement. Puis, une fois que les actions bénéfiques sont identifiées, il faut sélectionner celles qui permettent d'atteindre au mieux les objectifs du plan tout en respectant certaines contraintes, notamment budgétaires.

La solution que nous développons dans cette thèse a pour but de répondre à ces deux problématiques. Concernant l'identification des solutions efficaces, nous proposons d'intégrer dans *MSQualité* un entrepôt d'actions où seront stockées toutes les actions déjà mises en place dans des établissements et dont l'impact positif a été constaté dans au moins une structure (pour évaluer les résultats des actions, les établissements utilisent souvent des indicateurs). Puis, à l'élaboration d'un nouveau plan d'actions, *MSQualité* propose au décideur des actions suivant les objectifs du plan, son budget et le type de la structure. Pour la deuxième problématique (le choix des actions), nous proposons d'ajouter un nouveau module d'optimisation dans *MSQualité* où nous intégrons deux algorithmes d'optimisation multi-objectif, parmi ceux que nous développons dans le chapitre 3 et le chapitre 4.

Motivations et objectifs

L'idée de développer un système d'aide à la décision pour l'élaboration et l'optimisation de plans d'actions est née de plusieurs difficultés et manques que nous avons constatés dans le secteur social et médico-social. En effet, les trente dernières années, ce secteur a connu de grandes évolutions sur le plan démographique et législatif. Face à l'augmentation du nombre de personnes dites "vulnérables", les gouvernements successifs ont toujours essayé d'apporter des réponses à la hauteur de leurs besoins. Plusieurs textes de loi ont été élaborés pour protéger cette catégorie de personnes et, pendant plusieurs années, l'État et les collectivités locales ont dépensé beaucoup d'argent dans ce secteur. Malheureusement, la conjoncture économique difficile de ces dernières années a fait que les budgets des établissements sont de plus en plus réduits alors que le nombre de bénéficiaires ne cesse d'augmenter (dû principalement au vieillissement de la population et à l'augmentation de l'espérance de vie). Face à cette situation, les responsables de ces structures doivent s'adapter et trouver des solutions innovantes, à la fois techniques et organisationnelles, pour faire mieux avec les moyens qu'ils ont à leur disposition.

Cependant, bien que le secteur médico-social se soit beaucoup modernisé sur le plan législatif, il n'a pas su tirer profit des évolutions qu'a connu (et que connaît encore) l'informatique, notamment ce qui concerne les logiciels. Jusqu'aux années 2009-2010, l'informatique dans le médico-social se limitait aux logiciels de bureautique. Depuis, quelques logiciels de gestion commencent à être utilisés, notamment en ressources humaines, gestion budgétaire et plus récemment le suivi du parcours et du dossier du résident. Mais il y'en a très peu en gestion de la démarche qualité. Concernant l'aide à la décision, les outils existants se limitent uniquement à certains logiciels de gestion de tableaux de bord, basés sur des grilles d'indicateurs. Et face à des problèmes plus complexes, tels que la planification budgétaire, l'optimisation de planning ou de plans d'actions, les logiciels de gestion de tableaux de bord ne sont pas suffisants. De plus, les mutations et les difficultés que connaît le secteur vont rendre la gestion des établissements de plus en plus difficile et conduire sûrement les décideurs à recourir à des outils d'accompagnement et à la mutualisation des moyens et des actions. Malheureusement, à notre connaissance, il y a très peu de recherches engagées dans ce domaine. Notre but est donc de proposer des modèles et des méthodes d'optimisation pour mettre à disposition des responsables un outil de

collaboration et d'aide à la décision et faire de l'élaboration de plans d'actions optimaux une tâche aussi simple qu'efficace.

Organisation de la thèse

Ce manuscrit comporte quatre chapitres dont une brève description est donnée ci-dessous :

Chapitre 1. Nous présentons le contexte pratique et théorique dans lequel s'inscrit le travail de cette thèse. Nous commençons d'abord par fournir quelques éléments de compréhension sur l'évolution, l'organisation et le fonctionnement du secteur social et médico-social en France. Puis, nous introduisons les principales missions des établissements qui sont définies par la loi du 2 janvier 2002 et dans lesquelles s'inscrit le problème d'élaboration de plans d'actions. Ensuite, nous faisons un point de situation de l'informatisation du secteur et son besoin à des outils d'aide à la décision pour relever les défis auxquels il est confronté. Dans la deuxième partie du chapitre, nous fournissons quelques définitions et concepts concernant l'optimisation multi-objectif, utiles pour comprendre les algorithmes présentés dans les chapitres suivants.

Chapitre 2. Dans la première partie de ce chapitre, nous présentons le problème d'optimisation de plans d'actions multi-objectifs ainsi que les éléments pratiques qui le définissent. Puis, nous fournissons une modélisation formelle du problème sous forme d'un problème de sac-à-dos multi-objectif. Dans la deuxième partie, nous présentons la solution que nous proposons d'intégrer dans le logiciel *MSQualité* pour assister le décideur dans l'élaboration et l'optimisation de plans d'actions. Nous terminons par un état de l'art des principaux travaux connus sur ce problème.

Chapitre 3. Nous développons dans ce chapitre une méthode de résolution qui combine la recherche locale itérée avec l'indicateur de qualité $R2$, qui permet d'intégrer facilement les préférences du décideur dans le processus d'optimisation. Avant de présenter notre méthode, nous introduisons d'abord le principe d'optimisation par indicateurs de qualité. Puis, nous fournissons un état de l'art des méthodes de la littérature qui sont développées pour intégrer les préférences du décideur et, en particulier, celles basées sur les indicateurs de qualité. Nous montrons que l'algorithme $R2$ -IBMOLS proposé permet d'obtenir des solutions de bonne qualité mais il nécessite quand même d'être amélioré pour réduire le temps de son exécution (problème résolu dans le chapitre 4).

Chapitre 4. Pour faciliter au décideur le choix de la solution (du plan d'actions) à mettre en œuvre dans son établissement, il est important de lui présenter un nombre réduit de solutions de bonne qualité. Une solution de bonne qualité pour le décideur est celle qui correspond le plus à ses préférences, quand celles-ci sont exprimées, ou celle qui garantit l'équité dans le traitement des différents objectifs. Nous proposons de remplacer la dominance de Pareto par la dominance de Lorenz dans trois méthodes différentes. La première est une version améliorée de $R2$ -IBMOLS qui permet de contrôler le nombre de solutions générées par $R2$ -IBMOLS et réduire son temps d'exécution. Dans les deux autres méthodes, la dominance de Lorenz est adoptée pour orienter la recherche vers le centre de l'espace objectif, où les solutions sont les plus équitables. Un aspect important dans le cas où tous les objectifs ont la même importance pour le décideur.

Enfin, nous terminons cette thèse par une conclusion générale où nous reprenons les différentes contributions de ce travail ainsi que quelques perspectives de recherche.

Le contexte pratique et théorique

Ce chapitre présente le contexte pratique dans lequel s'inscrit le travail de cette thèse. Dans un premier temps, nous présentons quelques éléments de compréhension sur l'évolution, l'organisation, le fonctionnement ainsi que les acteurs du secteur social et médico-social en France. Nous introduisons également les principales missions des établissements du secteur qui sont définies par la loi du 2 janvier 2002 et dans laquelle s'inscrit notre travail. Puis, nous faisons un état de l'informatisation de ce secteur et son besoin d'outils performants d'aide à la décision pour relever les défis auxquels il est confronté. Dans un deuxième temps, nous présentons quelques définitions et concepts concernant l'optimisation multi-objectif sur lesquels portent les chapitres suivants.

Sommaire

1.1	Introduction	12
1.2	Le secteur social et médico-social (SSMS)	12
1.2.1	Description du SSMS	12
1.2.2	Évolution du SSMS	13
1.2.3	Organisation du SSMS	18
1.2.4	Les principaux acteurs du SSMS	20
1.2.5	Missions du SSMS selon la loi du 2 janvier 2002	23
1.2.6	L'informatique dans le SSMS et le besoin d'outils d'aide à la décision	24
1.2.7	Défis pratiques et scientifiques du SSMS	24
1.3	Optimisation multi-objectif (OMO)	28
1.3.1	Définitions	28
1.3.2	Quelques concepts sur l'optimisation multi-objectif	32
1.4	Conclusion	35

1.1 Introduction

Depuis une dizaine d'années, la France, comme beaucoup d'autres pays, se trouve dans un contexte socio-économique difficile pour répondre aux besoins de la population, particulièrement les personnes les plus démunies, et faire face aux nombreux défis qui vont arriver dans un futur proche. Selon plusieurs études et analystes, ce climat socio-économique va certainement s'accroître ces prochaines années, obligeant ainsi les pouvoirs publics et même les organisations privées à repenser leurs modes de gestion et leurs modes d'action afin d'optimiser l'utilisation des moyens disponibles et d'en tirer profit le plus possible.

Le **Secteur Social et Médico-Social (SSMS)** est l'un des secteurs qui est directement concerné par ces nouveaux défis à cause du vieillissement de la population et de l'augmentation de l'espérance de vie des personnes âgées et des personnes handicapées, d'un côté, et la diminution des dotations financières, d'un autre côté. Par conséquent, les établissements **SSMS** ont de plus en plus de difficultés à assurer leurs missions et à répondre aux besoins et aux attentes des personnes qu'ils accueillent ou qu'ils accompagnent, dont le nombre ne cesse d'augmenter. Pour faire face à cette situation, le **SSMS** doit innover et engager des études et des travaux ambitieux, que ce soit sur le plan pratique et législatif que sur le plan technique. En effet, sur le plan pratique et législatif, les choses commencent déjà à bouger : depuis 2002, plusieurs lois et décrets ont été élaborés pour préciser les rôles et les missions des différents types de structures, d'un côté, et définir les mécanismes de contrôles sur tous les aspects de la vie de l'établissement, d'un autre côté. Pour s'adapter aux manques de moyens matériels et financiers, les établissements se regroupent sous forme d'associations et de pôles afin de mutualiser et d'optimiser leurs moyens et d'élaborer des projets communs à tous les niveaux. Par contre, sur le plan technique informatique, il reste beaucoup à faire. L'informatique dans le **SSMS** se résume à quelques logiciels de bureautique et de gestion, les outils d'optimisation et d'aide à la décision sont complètement absents, alors que le besoin de ce type de logiciels est plus que jamais pressant.

Un des défis auxquels sont confrontés les établissements de ce secteur est l'élaboration de plans d'actions multi-objectifs efficaces, même quand les budgets sont réduits. Le but de ce travail est de les aider à relever ce défi en mettant à leur disposition un outil d'aide à la décision pratique et efficace. Avant de présenter le problème d'élaboration et d'optimisation de plans d'actions (chapitre 2) et des solutions que nous apportons dans cette thèse (chapitre 3 et 4), nous allons d'abord commencer par introduire quelques éléments de compréhension du secteur social et médico-social puis donner quelques définitions sur l'optimisation multi-objectif, utiles pour les chapitres suivants.

1.2 Le secteur social et médico-social (SSMS)

1.2.1 Description du SSMS

Le secteur **SSMS** est le fruit d'un modèle français, produit de l'histoire, qui combine les logiques de l'aide, de l'action, de la sécurité et d'inclusion sociales. Il est constitué d'organismes et de services qui accueillent, accompagnent et offrent des services de soins, d'hébergement, d'éducation, de formation, d'accompagnement et d'insertion sociale à différentes catégories de publics fragiles ou en situation marginale, tels que les enfants en âge préscolaire, les enfants ou adolescents qui sont en difficulté, les enfants ou adultes handicapés, les réfugiés, les personnes âgées dépendantes, etc. Les établissements sociaux et médico-sociaux reçoivent des personnes

de tous âges et de toutes catégories, atteintes de toutes sortes de pathologies au sein de structures multiples et spécialisées, permettant une grande diversité de prise en charge.

Cependant, comme l'a très bien dit Jean-François Bauduret¹, nous pourrions considérer qu'il n'est plus nécessaire, en 2017, de détailler l'identité et l'organisation du **SSMS**. Créé officiellement par la loi du 30 juin 1975², entièrement refondé par la loi du 2 janvier 2002³ et complété par une nouvelle gouvernance introduite par la loi du 21 juillet 2009⁴, ce secteur pourrait paraître à ce point connu qu'on ne le présente plus. Mais au contraire, ce n'est pas le cas. Il est plus que jamais nécessaire d'étudier et de communiquer autour de ce secteur, pour au moins trois raisons : (1) une organisation complexe, (2) une architecture mal appréhendée par les acteurs sociaux et médico-sociaux eux-mêmes, (3) une palette de prestations diversifiées largement méconnue des établissements et professionnels du champ sanitaire.

En effet, le champ des structures et services sociaux et médico-sociaux n'est pas simple à comprendre. Avec une nomenclature vaste et diversifiée de plus de 35 000 établissements et services de catégories différentes [Direction des études, statistiques et Prévisions du MASS, 2012] pour environ 2,4 millions de places [Agence Nationale d'Appui à la Performance, 2013] et 1,7 million de salariés [Direction des études, statistiques et Prévisions du MASS, 2012], le **SSMS** devient de plus en plus difficile à saisir. Aussi, de nombreux organismes gestionnaires (publics, associatifs, commerciaux) comportant plusieurs décideurs (préfets, directeurs généraux d'**ARS**, présidents de conseils généraux, etc.) sont impliqués d'une manière directe ou indirecte dans la gestion et la gouvernance des établissements de ce secteur. A cette diversité de structures et d'acteurs s'ajoute une diversité des outils de pilotages (**PRS**, **SROMS**, **PRIAC**, **SDOSMS**, **CPOM**, appels à projets) ainsi que des sources de financement nombreuses (budgets de l'état, de l'Assurance maladie et de la **CNSA**, des départements, sans compter l'importante participation des usagers eux-mêmes en gérontologie).

De plus, le **SSMS** est un secteur en forte croissance, tant au niveau structurel et organisationnel qu'au niveau prestations et services. Depuis sa création en 1975, le nombre d'établissements et de personnes prises en charge s'est multiplié par un facteur de dix pour certains types d'établissements. Et selon toutes les prévisions réalisées sur ce secteur, cette évolution va s'accroître encore plus les prochaines années, conséquence du vieillissement de la population. De ce fait, l'hétérogénéité du **SSMS** permet (et permettra plus encore les prochaines années) de prendre en charge plusieurs catégories de personnes qui sont dans le besoin, mais elle présente également bon nombre de fragmentations et de cloisonnements pouvant constituer autant d'obstacles à son appréhension par les gestionnaires d'établissements comme par les professionnels sociaux et médico-sociaux. Tout cela apporte de nouveaux challenges pratiques, mais aussi académiques, en particulier en optimisation.

1.2.2 Évolution du SSMS

Historiquement, l'action sociale et médico-sociale relevait de l'initiative d'organisations privées à but non lucratif, d'origine caritative ou encore issue des mouvements familiaux. Dès le Moyen

¹Directeur honoraire à la **CNSA** et Vice-président du Conseil scientifique et d'orientation de l'**ANAP**

²<https://www.legifrance.gouv.fr/affichTexte.do?cidTexte=JORFTEXT000000333976>

³<https://www.legifrance.gouv.fr/affichTexte.do?cidTexte=JORFTEXT000000215460>

⁴<https://www.legifrance.gouv.fr/affichTexte.do?cidTexte=JORFTEXT000020879475&categorieLien=id>

Age, des individus (principalement des religieux) se regroupent sous forme d'associations volontaires autour d'un chef charismatique. Au *XII^{me}* siècle, plusieurs confréries regroupaient des personnes autour d'un monastère ou dans un même quartier sur des motifs religieux et de secours mutuel des personnes. Cependant, il faudra attendre la *III^{me}* République pour voir se voter la loi du 2 juillet 1901 relative au contrat d'association.

Par ailleurs, l'évolution économique et sociétale du *XX^{me}* siècle va être l'occasion de la mise en place par l'état français de grandes politiques de protection sociale assise sur le travail et dont la consécration sera la création de la Sécurité Sociale. Les finalités de l'action sociale s'exprimeront autour de concepts tels que l'assistance, la rééducation, l'action sociale, l'insertion, l'accompagnement, la solidarité, etc. Mais avec les crises et difficultés économiques successives, notamment dans le fonctionnement du système de protection sociale, il devient indispensable de changer les méthodes de gestion afin de maîtriser les dépenses et rationaliser l'action publique. Des associations se proposent alors de combler le vide laissé par la disparition des cadres traditionnels de solidarité et d'apporter des réponses à des besoins non couverts par l'Etat. Cela marque la naissance de nouveaux modes d'interactions entre associations et pouvoirs publics qui vont dès lors s'inscrire dans un contexte de raréfaction des ressources.

Au cours du *XX^{me}* siècle, l'évolution du **SSMS** a connu 5 phases principales [Gallinaro, 2011], que nous pouvons résumer comme suit :

1. Avant 1946 : les « œuvres » comme prémices de l'action médico-sociale

Au cours de cette période, la priorité est donnée principalement aux aspects sanitaires. À cette époque, l'intervention de l'État n'existe pas encore, ce sont des structures de bienfaisance connues sous le nom d'"*hospices*" qui se mobilisent sur des actions de bienfaisances et sur la prise en charge sanitaire : apparition des dispensaires et des premiers hôpitaux. Plusieurs œuvres d'origine confessionnelle, puis laïques, sont devenues des associations et sont passées d'établissements à vocation sanitaire à établissements à vocation médico-sociale.

2. De 1946 à 1975/77 : structuration d'un secteur professionnel - des initiatives privées et création des premiers établissements

L'acte majeur et historique marquant cette période est la départementalisation, portée par Raymond Verges et Léon Lepervanche en 1946, avec l'application des lois d'assistance. Puis, en 1949, les premières réalisations médico-sociales sont lancées pour lutter pour la santé au travers de divers programmes : traitement, vaccinations, hygiène sociale, développement de l'hospitalier. Cette période verra également la mise en place de services publics et la création de la **DDASS** en 1964, la **CAF** en 1971 et plusieurs associations.

3. De 1975 à 1985 : avènement du **SSMS**

Le **SSMS** a été consacré par deux lois du 30 juin 1975 :

- la loi n° 75-534 dédiée aux personnes handicapées ;
- la loi n° 75-535 qui concerne les institutions sociales et médico-sociales et qui touche de multiples populations (enfants et familles en difficulté, personnes handicapées, personnes âgées, personnes en situation de précarité ou d'exclusion). Elle sépare le sanitaire du social et organise, pour la première fois, l'offre sociale et médico-sociale comme un ensemble homogène et autonome, soumis à des règles communes.

4. De 1984/86 à 2000 : égalité et reconnaissance sociale

Cette période est marquée par l'application de plusieurs grandes mesures sociales nationales. Plusieurs services de proximité se sont développés pour se rapprocher des familles des bénéficiaires, ce qui a entraîné la délocalisation ou la création d'établissements et de services plus petits. De ce fait, certaines associations sont devenues plus importantes et finissent par créer des sièges afin de contrôler le développement et la gestion de leurs établissements, d'une part, et porter et diffuser le projet associatif, d'autre part.

5. Après 2002 : rénovation et développement

Malgré les nombreux efforts et moyens mis en place depuis 30 ans, plusieurs problématiques médico-sociales persistent encore : des équipements sanitaires de pointe sont là, mais les prises en charge des personnes en situation de handicap restent très tendues, du fait d'un nombre de places insuffisantes. Des problèmes d'alcoolisme, de diabète, de violences faites aux femmes, d'illettrisme peinent à diminuer malgré une mobilisation des acteurs sur ces questions.

La loi du 2 janvier 2002 est arrivée pour moderniser et rénover l'action sociale et préciser la fonction des schémas régionaux d'organisation sociale et médico-sociale (SROSMS). Elle a instauré la mise en place de projets individuels pour les usagers, prenant en compte leurs attentes et leurs besoins. Cette loi a fixé 5 orientations principales :

- affirmer et promouvoir les droits des usagers ;
- diversifier les missions et les offres en adaptant les structures aux besoins ;
- mieux articuler la planification et la programmation, l'allocation de ressources et les évaluations ;
- instaurer une coordination entre les différents acteurs ;
- rénover le statut des établissements publics.

Elle a également favorisé la création de nouveaux types d'établissements et de services ainsi que l'émergence de nouveaux projets expérimentaux. Et pour garantir la continuité de l'amélioration de la prise en charge des personnes, elle a instauré pour les établissements un système obligatoire d'auto-évaluations (avec communication tous les 5 ans) et d'évaluations externes tous les 7 ans (évaluations réalisées par un organisme externe habilité).

Deux autres lois ont suivi apportant chacune des apports considérables tant sur la gestion et la gouvernance des structures que sur la prise en charge des personnes concernées : loi du 11 février 2005⁵ portant sur l'égalité des droits et des chances, la participation et la citoyenneté des personnes handicapées et celle du 22 juillet 2009⁶ portant sur la réforme de l'hôpital et relative aux patients, à la santé et aux territoires. Les apports principaux de ces lois sont présentés dans le tableau 1.1.

Pour accompagner et épauler les établissements dans l'application de ces textes et surtout l'accomplissement de leurs missions, que ce soit sur le plan pratique et législatif que financier, plusieurs organismes ont été créés. Les plus importants sont :

⁵<https://www.legifrance.gouv.fr/affichTexte.do?cidTexte=JORFTEXT000000809647>

⁶<https://www.legifrance.gouv.fr/affichTexte.do?cidTexte=JORFTEXT000020879475&categorieLien=id>

Loi du 2 janvier 2002	Loi du 5 février 2005	Loi du 22 juillet 2009
<ul style="list-style-type: none"> • renforcer les droits des usagers ; • garantir la prise en charge continue des besoin des usagers ; • veiller au bon fonctionnement des établissements par un système d'évaluations internes et externes ; • améliorer l'organisation, la coordination et la coopération des différents acteurs. 	<ul style="list-style-type: none"> • accès à une scolarisation adaptée ; • garantir la plus grande autonomie possible ; • accessibilité des personnes handicapées à toutes les chaines de déplacement ; • accès unifié aux droits et prestations prévues pour les personnes handicapées. 	<ul style="list-style-type: none"> • création d'une Agence Nationale d'Appui à la Performance (ANAP) et définition de critères de mesure de performance des établissements du SSMS ; • articulation renforcée entre les établissements sanitaires et les établissements sociaux et médico-sociaux ; • promouvoir de nouveaux outils pour améliorer et accompagner les établissements.

TABLE 1.1 – Les principaux apports des lois du 2 janvier 2002, du 5 février 2005 et du 22 juillet 2009.

- la **Caisse Nationale de Solidarité pour l'Autonomie (CNSA)** : mise en place en 2005, la **CNSA** est chargée de financer les aides en faveur des personnes âgées en perte d'autonomie et des personnes handicapées, de garantir l'égalité de traitement sur tout le territoire et pour l'ensemble des handicaps et des situations de perte d'autonomie, d'assurer une mission d'information et d'animation de réseau, d'information des personnes âgées, des personnes handicapées et de leurs familles, d'assurer un rôle d'expertise et de recherche sur les questions liées à l'accès à l'autonomie, quels que soient l'âge et l'origine du handicap. Elle dispose pour cela d'un budget de plus de 25 milliards d'euros⁷.
- l'**Agence Nationale de l'Evaluation et de la qualité des établissements et Services Sociaux et Médico-sociaux (ANESM)** : créée en 2007, l'**ANESM** est née de la volonté des pouvoirs publics d'accompagner les **ESSMS** dans la mise en œuvre de l'évaluation interne et externe, instituée par la loi du 2 janvier 2002. Dans le même cadre, la deuxième mission de l'**ANESM** consiste à habilitier des organismes externes qui procéderont à l'évaluation externe à laquelle sont soumis les **ESSMS**⁸.
- l'**Agence Nationale d'Appui à la Performance des établissements de santé et médico-sociaux (ANAP)** : créée en 2009, cette agence vient en appui des **ESSMS** pour améliorer leur performance dans le cadre de la réforme du système de santé en France.

⁷<http://www.cnsa.fr>

⁸http://http://www.anesm.sante.gouv.fr/spip.php?page=article&id_article=
108/

L'ANAP a pour objet d'aider les ESSMS à améliorer le service rendu aux patients et aux usagers, en élaborant et en diffusant des recommandations et des outils dont elle assure le suivi de la mise en œuvre, leur permettant de moderniser leur gestion, d'optimiser leur patrimoine immobilier et de suivre et d'accroître leur performance, afin de maîtriser leurs dépenses. A cette fin, dans le cadre de son programme de travail, elle peut procéder ou faire procéder à des audits de la gestion et de l'organisation de l'ensemble des activités des établissements de santé et médico-sociaux⁹.

- les Agences Régionales de Santé (ARS) : créées en 2010, ces agences visent la mise en œuvre de la politique de santé dans les régions. Elle permettent également de simplifier les démarches de santé en opérant à la fois sur les secteurs sanitaires, médico-sociaux et ambulatoires. Elles sont chargées de l'organisation et du financement des soins au niveau régional (à partir des ressources de la CNSA) et de contrôler l'application des politiques de santé publique afin de renforcer l'efficacité collective et de garantir l'avenir du service public de la santé.

L'évolution et l'amélioration des textes réglementaires accompagnées de la volonté des services publics et privés (lucratif et non lucratif) d'améliorer le quotidien des personnes dépendantes, fragiles ou défavorisées ont largement contribué à l'évolution et au dynamisme du SSMS. Cela a permis d'élargir la gamme d'établissements et de services et de faire bénéficier différentes catégories de publics : les enfants en âge préscolaire, les enfants ou adolescents en difficulté, les enfants ou adultes handicapés, les adultes en difficulté et les personnes âgées. En effet, aujourd'hui le SSMS compte plus de 35 000 établissements qui disposent d'environ 1,5 million de places. Si l'on ajoute le domaine de la petite enfance (accueil collectif, assistantes maternelles, garde à domicile), le secteur représente une capacité d'hébergement ou d'accompagnement de 1,7 million de places permettant ainsi une grande diversité de prise en charge [Ministère de la jeunesse et des solidarités actives, 2010].

Par ailleurs, selon un rapport de la *Direction des études, statistiques et prévisions* datant de 2012 [Direction des études, statistiques et Prévisions du MASS, 2012], l'emploi privé dans le SSMS a progressé de près de 50% entre 2000 et 2010, passant de 653 000 à 976 000 salariés (figure 1.1), soit un taux de croissance annuel moyen de l'ordre de 4,1% (figure 1.2). Cette croissance est bien supérieure à celle de l'ensemble de l'emploi privé, lequel a progressé durant la même période de 0,6% par an en moyenne, et de 6% sur l'ensemble de la période. Avec une telle croissance, le SSMS est un des secteurs qui a contribué le plus au dynamisme de l'emploi. On estime que l'ensemble du secteur emploie, en 2013, environ 1,7 million de salariés, soit environ 6,7% du nombre total d'actifs en France. En termes d'emploi, c'est l'un des plus importants de l'économie. Il dépasse notamment ceux des transports (840 000 emplois), de la restauration (680 000 emplois) ou encore des services financiers (570 000 emplois) et autant que l'enseignement (1,7 million) [Direction des études, statistiques et Prévisions du MASS, 2012].

⁹<http://www.anap.fr/l-anap/qui-sommes-nous/>

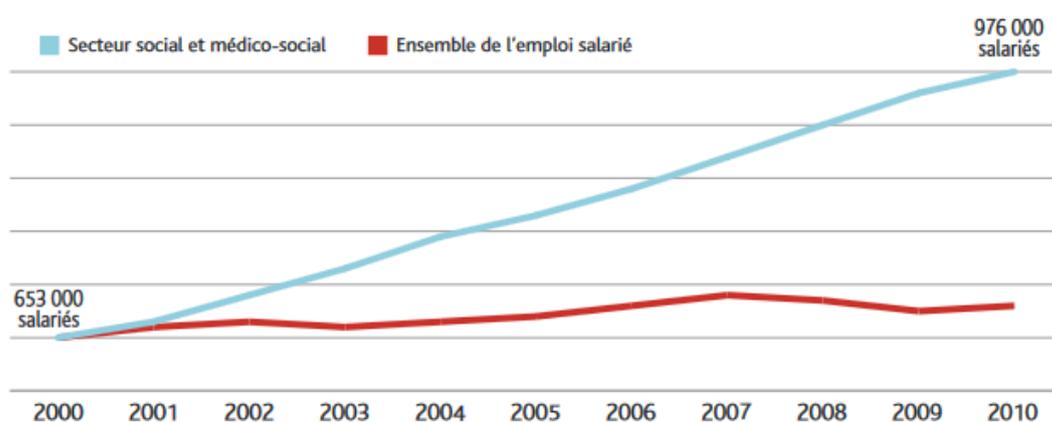


FIGURE 1.1 – Évolution des effectifs salariés dans le SSMS et dans l'ensemble des secteurs

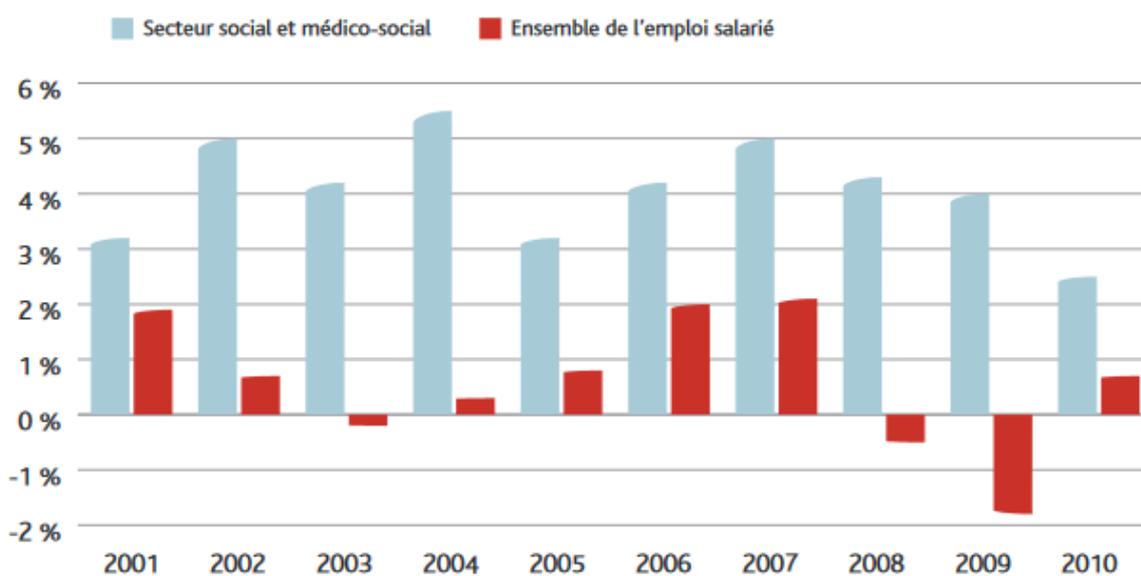


FIGURE 1.2 – Taux de croissance de l'emploi dans le SSMS et dans l'ensemble des secteurs

1.2.3 Organisation du SSMS

Afin d'adapter les services offerts aux besoins de chaque catégorie de personnes et garantir une grande diversité d'accompagnements, les établissements du SSMS sont définis suivant les caractéristiques du public visé et des services rendus. Le SSMS se répartit en 14 domaines principaux selon les champs d'action et, de la même façon, à chacun de ces champs d'action se rattachent différents types d'établissements et services. Ainsi, chaque établissement propose, aux usagers qu'il reçoit, des accompagnements ciblés, adaptés et parfois individualisés, quand c'est nécessaire. La figure 1.3 liste les structures les plus répandues avec l'organe de rattachement et le mode de financement correspondant à chaque structure.

Champs d'action	Nom de l'établissement	Financement	Champ de compétence
Aide sociale à l'enfance	Maison d'enfants à caractère social (MECS)	Aide sociale	Département
	Club de prévention spécialisée		
	Service d'Action éducative en milieu ouvert (AEMO)		
	Service d'Action éducative à domicile (AED)		
Enfance handicapée et inadaptée.	Centres médico-psycho-pédagogiques (CMPP), instituts médico-éducatifs (IME), médico-pédagogiques (IMP), médico-professionnels (IMPRO), instituts de rééducation (ITEP), Service d'éducation spéciale et de soins à domicile (SESSAD)	Assurance maladie	État
Enfance handicapée (0-6 ans)	Centres d'action médico-sociale précoce (CAMSP)	80 % assurance maladie 20 % aide sociale	État-département
Protection judiciaire de la jeunesse (PJJ)	Centres et Foyers d'action éducative (CAE et FAE), Services éducatifs auprès du tribunal (SEAT)	État (ministère de la Justice)	État ou État-département
Handicap mental ou physique, adultes	Établissements et services d'aide par le travail (ESAT, anciennement CAT)	État	État+Activité commerciale
	Centres de rééducation professionnelle, Centre d'orientation professionnelle	Assurance maladie - État	
Personnes âgées	Établissement d'hébergement pour personnes âgées dépendantes (EHPAD)	Assurance maladie - aide sociale - allocation personnalisée d'autonomie (APA)	État-Département
	foyers-logements, maisons de retraite pour personnes non dépendantes (EHPA)	Aide sociale éventuelle	Département
	Service de soins infirmiers à domicile (SSIAD)	Assurance maladie	État
	Services de soins, d'aide et d'accompagnement	Assurance maladie - aide sociale	État-département
	Services d'aide à domicile (SAAD)	Aide sociale -caisses de retraite (action sociale facultative)	Agrément : préfet
Handicap mental, polyhandicap, adultes	Foyers d'hébergement ou foyer de vie	Aide sociale	Département
	Maison d'accueil spécialisée (MAS), Foyer d'accueil médicalisé (FAM), Services de soins, d'aide et d'accompagnement	Assurance maladie - Aide sociale	État-département
	Service d'auxiliaires de vie	État - Aide sociale	État-département
Personnes sans domicile, en errance	Centres d'hébergement et de réinsertion sociale (CHRS) et notamment centres d'aide par la vie active, accueils de jour (boutiques de solidarité ...), veille sociale (Samu social, équipes mobiles, téléphonie sociale « 115 », services d'accueil et d'orientation)	État + contribution le cas échéant des communes et de l'aide sociale départementale	État (en partenariat le cas échéant avec le département et les communes)
Traitement des addictions	Centres d'accueil pour toxicomanes et les Centres d'accueil et d'accompagnement à la réduction de risques pour usagers de drogues (CAARUD).	État	État
	Centres d'accueil pour alcooliques, Appartements de coordination thérapeutique (VIH, maladies chroniques)	Assurance maladie	État
Autres hébergements	Foyers de jeunes travailleurs (FJT)	État-aide sociale -caisse d'allocations familiales	État (conventionnement aide personnalisée au logement, code de la construction et de l'habitation)
Centres de ressources	Centres de ressources: handicap rare, autisme, traumatisme crânien, maladie d'Alzheimer. Autres (centres locaux d'information et de coordination...)	Assurance maladie	État
Demandeurs d'asile	Centre d'accueil de demandeurs d'asile	État	État
Majeurs à protéger	Services mettant en œuvre le mandat spécial au profit de majeurs protégés	Etat + Majeur protégé	État
	Services mettant en œuvre l'aide à la gestion du budget familial		État
Autres	Structures expérimentales dérogeant aux articles L. 162-31 et L. 162-31-1 du code de la sécurité sociale : possibilité de décisions déconcentrées et/ou décentralisées	Toutes possibilités	Selon les cas : État ou département ou État-département
	Lieux de vie non traditionnels	Aide sociale (principalement)	

FIGURE 1.3 – Catégories des établissements du SSMS [OSORIO MONTOYA, 2015]

Cependant, dans un contexte économique de plus en plus difficile, une augmentation continue du nombre de personnes dites "*vulnérables*" et une législation stricte et parfois complexe, les établissements ont beaucoup de mal à assurer leurs missions et garantir une prise en charge de qualité pour les populations qu'ils reçoivent. Pour s'adapter à ces nouvelles contraintes les établissements s'organisent et se regroupent, de plus en plus, sous forme de pôles et d'associations locales, départementales, régionales et nationales. Ces regroupements peuvent être fonctionnels (regroupement par types d'établissements) ou géographiques (différents types d'établissements qui se trouvent dans une même zone géographique). Cela leur permet d'harmoniser leurs pratiques, de mutualiser leurs moyens (financiers, matériels et humains) et de mettre en œuvre des projets de développement communs. Ainsi, les créations d'associations dans le secteur social et sanitaire ont représenté, de 2007 à 2013, environ 12% des nouvelles associations, tous secteurs confondus. Avec une création moyenne annuelle de 3 500 nouveaux services et établissements sur la même période¹⁰.

1.2.4 Les principaux acteurs du SSMS

En France, le système sanitaire, social et médico-social est composé de trois statuts juridiques distincts :

- **le champ public**

Il concerne un ensemble d'hôpitaux et d'établissements médico-sociaux qui sont liés à la fonction publique et territoriale. Ces établissements sont gérés et financés par plusieurs acteurs, tant au niveau national que territorial, jouant ainsi un rôle essentiel dans le financement et le pilotage de l'action sociale. La Tableau 1.2 montre certains d'entre eux selon leur niveau d'intervention [OSORIO MONTROYA, 2015].

D'autres acteurs nationaux placés sous la tutelle des ministères en charge de la santé, de l'aide et de l'action sociale interviennent également dans la prise en charge de certaines catégories de personnes. Nous pouvons citer (à titre d'exemple) le *conseil supérieur du travail social* ; le *conseil national consultatif des personnes handicapées* ; le *conseil pour les droits et devoirs des familles* ; le *conseil national des politiques de lutte contre la pauvreté et l'exclusion* ; le *comité national des retraites et personnes âgées*. Aussi, plusieurs observatoires ont été mis en place, comme l'*observatoire national de l'enfance en danger* (depuis 2004) et l'*observatoire national de la pauvreté et de l'exclusion sociale* [Bazillon, 2013].

- **le champ privé lucratif**

Le secteur privé lucratif concerne principalement des cliniques et maisons de retraites. Depuis la fin des années 1980 et suite à l'évolution démographique et l'augmentation de l'espérance de vie observées ces trois dernières décennies, les acteurs privés ont largement investi le champ d'hébergement et d'accueil des personnes âgées. Ainsi, entre 1986 et 1988, 47% des places d'hébergement de personnes âgées ont été ouvertes avec des fonds privés, par des entreprises commerciales. Bien plus récemment, presque deux tiers des 7 500 places d'accueil, ouvertes grâce aux fonds provenant de la journée de solidarité (instaurée par le gouvernement de Jean-Pierre Raffarin suite à la canicule de 2003), l'ont été au sein de structures privées commerciales [Bazillon, 2013].

¹⁰Source : <http://www.lien-social.com/Un-etat-du-secteur-medico-social>

L'État	L'État possède un rôle d'impulsion en matière de politiques sociales, notamment par l'intermédiaire de deux organismes principaux, la DGCS et l' IGAS . Le premier est chargé de la conception, du pilotage et de la cohérence des politiques publiques sur le territoire national tandis que le deuxième joue un double rôle, d'inspection et de proposition via son rapport annuel. Il y a aussi la CNSA qui garantit le financement à travers des ressources de l'État et pilote la politique en faveur de l'autonomie des personnes âgées et handicapées et à l'égalité du traitement qui leur est réservé.
La région	Les régions sont chargées de contrôler l'application de la législation sociale et de coordonner l'ensemble des actions conduites par les organismes de l'État via les ARS .
Le département	Les départements ont pour mission d'aider les personnes en difficulté à développer leur autonomie. Ces dernières ont accès aux services à travers la MDPH , qui est un guichet d'accueil unique prenant en charge les démarches liées aux différentes situations de handicap et en effectue le suivi. En pratique, le département habilite les établissements publics à mettre en œuvre les actions sociales et médico-sociales pour les personnes âgées dépendantes de plus de 60 ans et pour les personnes handicapées de moins de 60 ans.
La sécurité sociale	La sécurité sociale est en charge d'assurer une personne face à des risques financiers relatifs aux soins suivis (remboursement, paiement des indemnités journalières, avance des frais, etc.) à travers la CPAM dont le financement est pris en charge par l'État.

TABLE 1.2 – Les principaux acteurs du secteur public.

- **Le champ privé non lucratif**

Le secteur privé à but non lucratif concerne des hôpitaux, des cliniques et la plupart des établissements. Il est majoritairement assuré par des associations à but non lucratif, à hauteur de 56% des lits et places dans le médico-social et 15% des lits et places dans le secteur de la santé, où il a plus de mal à faire reconnaître sa place. Peu connu du grand public, le secteur privé non lucratif est souvent défini comme un modèle de gestion alliant les avantages du privé et du public pour le plus grand profit de l'utilisateur. En effet, aujourd'hui le secteur privé non lucratif fait partie intégrante de l'économie sociale. Il est devenu un acteur important de l'action sociale offrant de nombreux services de soins et de l'aide à la personne tant au niveau sanitaire que social et médico-social. Il a fait ses preuves en termes d'efficacité et de performance, à tel point que l'on peut aujourd'hui parler d'un vrai modèle liant souplesse de gestion et dynamisme à la qualité du service et proximité de l'offre de soins à l'adaptation aux besoins de la population [[Lyes, 2008](#)].

La figure 1.4 illustre la répartition du nombre de lits et de places de tous les établissements et services médico-sociaux, entre le secteur public, le secteur privé lucratif et le secteur privé non lucratif. Les figures 1.5, 1.6, 1.7 et 1.8 montrent la répartition des parts d'emplois occupés entre les trois secteurs pour quatre types d'établissements.

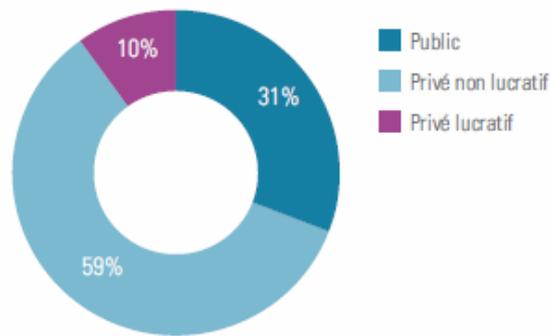


FIGURE 1.4 – Répartition du nombre de lits et de places entre le secteur public, le secteur privé lucratif et le secteur privé non lucratif.

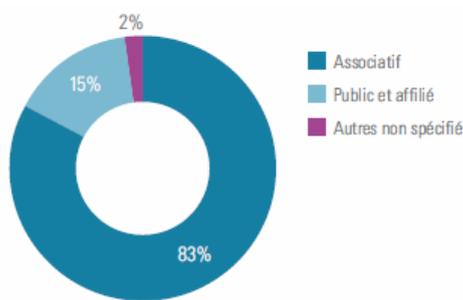


FIGURE 1.5 – Répartition des emplois occupés dans les structures d'accueil et d'hébergement des personnes handicapées entre le secteur public, le secteur privé lucratif et le secteur privé non lucratif [Source : enquête DREES, ES 2006, calculs UNIOPSS].

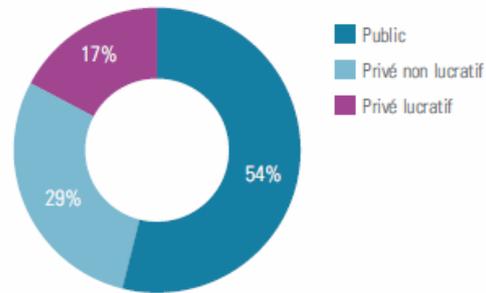


FIGURE 1.6 – Répartition des emplois occupés dans les structures d'accueil et d'hébergement pour personnes âgées entre le secteur public, le secteur privé lucratif et le secteur privé non lucratif [DREES, 2010].

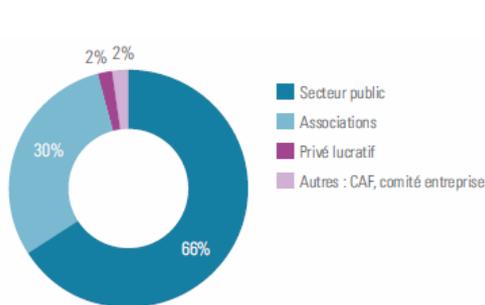


FIGURE 1.7 – Répartition des emplois occupés dans les structures d'accueil de jeunes enfants de moins de 6 ans entre le secteur public, le secteur privé lucratif et le secteur privé non lucratif [Source : enquête PMIDREES 2008, calculs UNIOPSS].

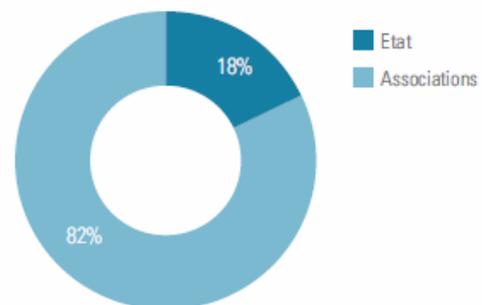


FIGURE 1.8 – Répartition des emplois occupés dans les structures recevant des mineurs entre le secteur public, le secteur privé lucratif et le secteur privé non lucratif [Source : projet de loi de finances 2011, calculs UNIOPSS].

1.2.5 Missions du SSMS selon la loi du 2 janvier 2002

L'objectif du secteur médico-social en France est d'apporter un accompagnement et une prise en charge aux publics vulnérables (personnes en situation de précarité, d'exclusion, de handicap ou de dépendance). Ses missions ont été définies et clarifiées par la loi du 02 janvier 2002 rénovant l'action sociale et médico-sociale. Cette loi définit et structure l'action sociale et médico-sociale destinée à promouvoir, dans un cadre interministériel, l'autonomie et la protection des personnes, la cohésion sociale, à prévenir les exclusions et à en corriger les effets. Elle met en évidence les champs d'interventions et les missions des établissements sociaux et médico-sociaux, dont les principales sont les suivantes [OSORIO MONTOYA, 2015] :

1. Protection administrative et judiciaire de l'enfance et de la famille, de la jeunesse, des personnes handicapées, des personnes âgées ou en difficulté ;
2. Évaluation et prévention des risques sociaux et médico-sociaux, information, investigation, conseil, orientation, formation, médiation et réparation ;
3. Actions d'assistance dans les divers actes de la vie, de soutien, de soins et d'accompagnement, y compris à titre palliatif ;
4. Actions éducatives, médico-éducatives, médicales, thérapeutiques, pédagogiques et de formation adaptées aux besoins de la personne, à son niveau de développement, à ses potentialités, à l'évolution de son état ainsi qu'à son âge ;
5. Actions d'intégration scolaire, d'adaptation et de réadaptation, d'insertion et de réinsertion sociales et professionnelles, d'aide à la vie active, d'information et de conseil sur les aides techniques ainsi que d'aide au travail ;
6. Actions contribuant au développement social et culturel, et à l'insertion par l'activité économique.

D'une manière générale, la loi du 2 janvier 2002 a placé le droit de l'usager au cœur de l'action sociale comme elle a défini et orienté la mission des établissements vers la garantie de ce droit en établissant de nouvelles règles dans la gestion des établissements. Elle a notamment institutionnalisé l'obligation des établissements de réaliser des évaluations internes et externes, de mener des projets et des plans d'actions pluriannuels et de les inscrire dans un processus d'amélioration continue de la gestion de l'établissement et de la prise en charge des usagers.

En effet, avant la loi du 2 janvier 2002, les établissements n'avaient pas d'obligation de faire des évaluations internes ou externes. Par conséquent, les directeurs géraient leurs structures comme ils le conçoivent et il y avait peu de contrôles. Avec cette loi, les établissements sont obligés de faire une évaluation interne tous les 5 ans et une évaluation externe tous les 7 ans suivies d'un projet et d'un plan d'actions pluriannuel. En 2007, un nouveau décret a été publié pour définir tous les aspects de la vie de l'établissement que l'évaluateur externe doit vérifier. Ce dernier construit un référentiel suivant les recommandations de ce décret, l'établissement doit répondre à chaque question de ce référentiel et apporter, dans sa réponse, des éléments prouvant les actions menées dans l'établissement. Par conséquent, le travail réalisé dans cette thèse pour l'optimisation de plans d'actions s'inscrit dans ce cadre pratique et législatif décrit précédemment.

1.2.6 L'informatique dans le SSMS et le besoin d'outils d'aide à la décision

Bien que le **SSMS** se soit beaucoup modernisé sur le plan législatif et pratique dans l'accompagnement et la garantie des droits des personnes en difficulté, il reste très en retard sur le plan informatique et technologique. En effet, même si le **SSMS** est né à peu près à la période où l'informatique commençait à pénétrer dans le monde socio-économique, il n'a pas su tirer profit des évolutions qu'a connu (et que connaît encore) l'informatique, notamment en ce qui concerne les logiciels.

Jusqu'aux années 2009-2010, et contrairement au secteur sanitaire, l'informatique dans le **SSMS** se limitait aux logiciels de bureautique. Depuis, quelques logiciels de gestion commencent à être utilisés, notamment en ressources humaines, gestion budgétaire et plus récemment le suivi du parcours et du dossier du résident, mais très peu en gestion de la démarche qualité. De plus, les outils d'aide à la décision se limitent uniquement à certains logiciels de gestion de tableaux de bord, basés sur des grilles d'indicateurs. En effet, les indicateurs peuvent s'avérer être un outil efficace pour aider le responsable à prendre certains types de décisions. Ils permettent de synthétiser une masse d'information sous forme de quelques données plus lisibles et facilement compréhensibles. Mais face à des problèmes plus complexes, tels que la planification budgétaire, l'optimisation de planning ou de plans d'actions, les logiciels de gestion de tableaux de bord ne sont pas d'une grande utilité. Et malheureusement, à notre connaissance, à part la thèse réalisée, en 2015, par Geovanny Osorio Montoya [[OSORIO MONTOYA, 2015](#)] portant sur la conception d'un système de mesures et de suivi des activités ainsi que l'analyse de ces données de suivi, il y a très peu de recherches engagées dans ce domaine. De plus, les mutations du secteur sur le plan financier, législatif et humain vont rendre la gestion des établissements de plus en plus difficile et conduire les acteurs du secteur à recourir à des outils d'accompagnement et à la mutualisation des moyens et des actions. Le but de ce travail est justement de mettre à disposition des directeurs des établissements un outil à la fois d'aide à la décision et de collaboration, afin de contribuer à leur faciliter la gestion de leurs systèmes qualité.

1.2.7 Défis pratiques et scientifiques du SSMS

1.2.7.1 Défis pratiques

Le **SSMS** se caractérise par un certain nombre de problématiques et d'enjeux communs auxquels il faudrait apporter des réponses et engager des réflexions dès aujourd'hui. En effet, le **SSMS** s'inscrit dans le cadre de la transition épidémiologique, phénomène qui se caractérise par le vieillissement de la population et par une prévalence toujours plus importante des maladies chroniques. Le vieillissement démographique et la croissance des situations de dépendance physique et psychique qui en résultent représentent un défi médical, social, financier et organisationnel majeur, qui fait de la perte d'autonomie une question prioritaire dans l'agenda des politiques publiques françaises et oblige le secteur médico-social à se réformer afin de garantir aux personnes fragiles (pour la plupart vieillissantes) une prise en charge de qualité, répondant à leurs besoins.

Le vieillissement démographique est un phénomène qui touche l'ensemble des sociétés occidentales en raison de l'augmentation de l'espérance de vie et de la baisse parallèle de la fécondité. En 2011, la France compte plus de 12 millions de personnes âgées de plus de 60 ans,

dont plus de 800 000 sont en perte d'autonomie, et 2,3 millions de ces personnes âgées ont plus de 80 ans [ARS Île-de-France, 2016]. Selon une étude de l'INSEE de 2010, ce mouvement de vieillissement va s'accroître encore d'ici 2060, avec une augmentation d'environ 50% pour les personnes âgées de 60 ans et 100% pour les plus de 80 ans. (figure 1.9).

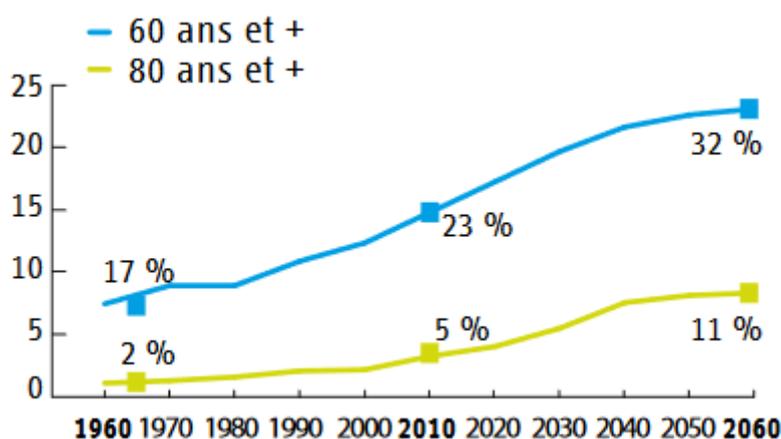


FIGURE 1.9 – Vieillissement de la population française à l'horizon 2060 [Source : INSEE, 2010, projections de population 2007-2060]

De plus, la problématique de la prise en charge des personnes handicapées vieillissantes va se poser avec acuité puisque toutes les projections anticipent, également, une augmentation significative de la population handicapée âgée de 46 à 59 ans. L'évolution épidémiologique et démographique de la population fait du handicap une préoccupation sociale majeure et croissante car, comme le reste de la population, les personnes handicapées connaissent une augmentation spectaculaire de leur espérance de vie. A titre d'exemple, l'espérance de vie d'une personne touchée par une trisomie 21 est passée d'une vingtaine d'années dans les années 1970 à plus de 40 ans aujourd'hui (70% des personnes touchées par cette maladie vivront même au-delà de 50 ans [ARS Île-de-France, 2016]). Par ailleurs, comme le montre une enquête de l'INSEE datant de 2009, chez les personnes de 20 à 59 ans vivant à domicile, la France compte 4,6 millions de personnes qui présentent une forme de handicap, soit 14 % de cette population globale. 2 750 000 personnes souffrent d'au moins une limitation fonctionnelle, dont 420 000 ont aussi une reconnaissance administrative, 420 000 déclarent également un handicap ressenti, et 730 000 ont un risque de handicap avec les trois critères cumulés (figure 1.10).

Sur le plan financier, en 2014, rien que la compensation de la perte d'autonomie des personnes âgées et des personnes handicapées et leur prise en charge ont coûté aux finances publiques (tous organismes confondus) pas moins de 59,3 milliards d'euros (figure 1.11), dont 20,8 milliards pour les personnes âgées dépendantes et 38,5 milliards pour les personnes handicapées (figure 1.12).

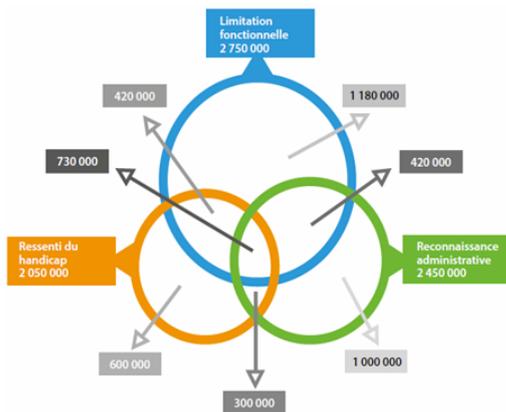


FIGURE 1.10 – Nombre de personnes, âgées de 20 à 59 ans, en situation de handicap [Source : INSEE, 2009, Enquête Handicap-Santé 2008-2009, volet ménages]

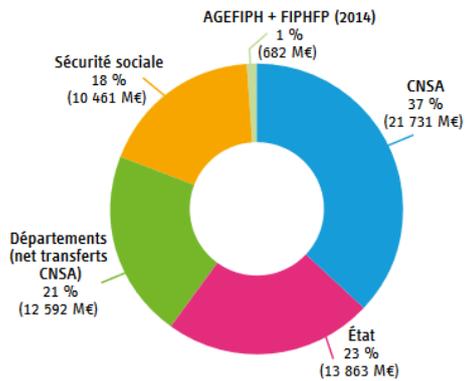


FIGURE 1.11 – Contribution globale des finances publiques à la compensation de la perte d'autonomie des personnes âgées et des personnes handicapées en 2014

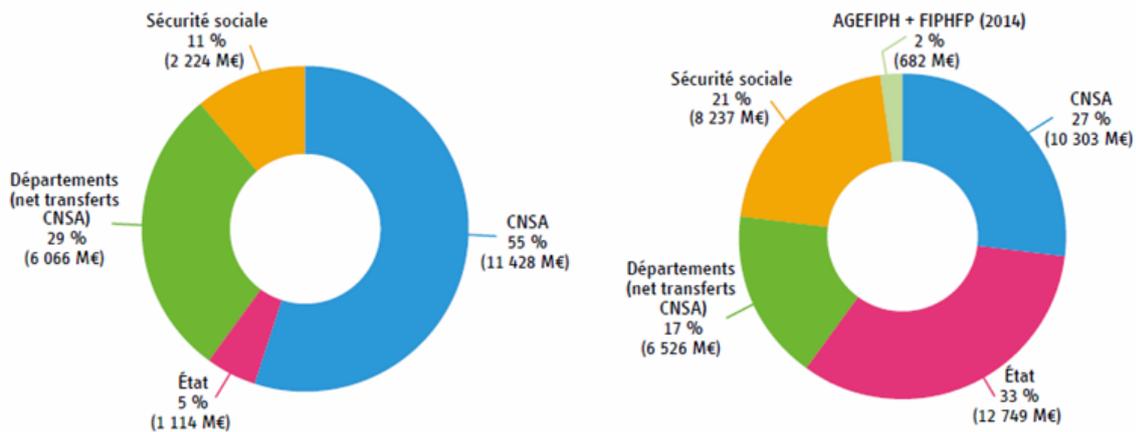


FIGURE 1.12 – Contribution des finances publiques à la compensation de la perte d'autonomie des personnes âgées (gauche) et des personnes handicapées (droite) en 2014.

1.2.7.2 Défis scientifiques

Comme nous l'avons vu dans la section précédente, les dirigeants des **ESSMS** se retrouvent devant deux réalités difficiles :

1. Le vieillissement de la population, l'augmentation de l'espérance de vie, que ce soit pour les personnes âgées ou pour les personnes handicapées, ainsi que l'évolution des situations de précarité et d'exclusion vont engendrer systématiquement une augmentation du nombre de bénéficiaires et l'élargissement de la palette des besoins auxquels il faut répondre et des services qu'il faudra assurer. A cela s'ajoutent de nombreuses contraintes, certaines légales liées à la prise en charge des personnes (garanties des droits et libertés individuelles et collectives, garantie d'une prise en charge personnalisée, l'obligation de

sécurité, l'élaboration de plans de maîtrise sanitaire et des plans d'amélioration continue, etc.), d'autres liées à la responsabilité sociétale et environnementale de l'établissement. De plus, la diversité des établissements et les caractéristiques des populations qui y sont accueillies, souvent des personnes en difficulté ou handicapées (mental ou physique), obligent les dirigeants à mettre tout en œuvre pour assurer une meilleure qualité de service et une bonne prise en charge de ces personnes.

2. La difficulté de la conjoncture économique, d'un côté, et les coûts faramineux nécessaires à la prise en charge des différentes catégories de bénéficiaires, d'un autre côté, ont amené les pouvoirs publics à définir de nouveaux modes de financement pour mieux gérer ce secteur et rationaliser ses dépenses. En effet, le **Contrat Pluriannuel d'Objectifs et de Moyens (CPOM)**, initialement prévu par la loi du 2 janvier 2002, est devenu obligatoire à toutes les structures (depuis le 1^{er} janvier 2016 pour les établissements recevant des personnes handicapées et à compter du 1^{er} janvier 2017 pour les établissements recevant des personnes âgées dépendantes), obligeant ainsi les directeurs d'**ESSMS** à définir des visions sur cinq ans, à faire des choix et à prioriser leurs objectifs et leurs actions. En outre, les organismes gestionnaires sous **CPOM** se trouveront tenus d'élaborer et de soumettre un **État Prévisionnel des Recettes et des Dépenses (EPRD)** ce qui les mène également à faire une gestion budgétaire prévisionnelle sur cinq ans. A cela, s'ajoute la nomenclature **Services et Établissements : Réforme pour une Adéquation des FINancements au Parcours des Personnes Handicapées (SERAFIN-PH)** (dispositif en cours de réalisation) réformant la tarification des **ESSMS** accueillant et accompagnant des personnes handicapées. Aujourd'hui ces établissements sont majoritairement financés sur la base de dotations négociées historiquement, sans que cette allocation de ressources soit réellement objective au regard des besoins des personnes et des prestations apportées. Avec cette nouvelle nomenclature, les établissements seront financés par rapport aux besoins réels des personnes qu'ils accueillent, évalués suivant la grille **SERAFIN-PH** (à un besoin correspond un tarif).

Face à ces difficultés et contraintes dans tous les aspects (ou presque) de la vie de l'établissement, les directeurs sont devant un challenge : **comment faire plus et mieux avec peu de moyens ?**

Pour s'adapter à ces nouvelles contraintes, un mouvement de regroupement des établissements sous forme de pôles, de cités, de territoires et d'associations a été engagé ces dernières années. Ces regroupements s'accompagnent d'une nécessité d'engager des projets de collaboration communs, de plusieurs plans d'actions (plans d'actions sanitaires, plans de prévention des risques professionnels, **CPOM**, plans de formations, etc) et de mutualiser les moyens disponibles à tous les niveaux (pôles, cités, départements, régions, etc).

Par conséquent, il devient primordial (et même urgent) d'engager des études approfondies sur les besoins pratiques et techniques du **SSMS** et de promouvoir l'innovation et les projets d'accompagnement afin d'arriver au développement de systèmes mutualisés, un partage d'informations plus important et une meilleure optimisation des plans d'actions et des moyens existants. En effet, le partage d'informations permet à chaque structure de tirer profit des expériences passées des uns et des autres. Cependant, même si le cadre permettant ce partage d'informations reste à construire, les gestionnaires d'établissements peuvent néanmoins l'anticiper lors du développement de leurs systèmes d'information, et l'investissement dans la recherche demeure une des clés principales susceptible d'améliorer la gestion du **SSMS** et de contribuer à la maîtrise des dépenses futures.

Une des problématiques qui se pose pour le décideur, est comment réaliser des projets ambitieux avec des plans d'actions améliorant la qualité de tous les aspects du fonctionnement de l'établissement et des services qu'il fournit, alors que les budgets se réduisent d'année en année ? Pour relever ce défi, le décideur n'a pas le choix, il doit s'assurer que chaque action mise en place, chaque euro dépensé, doit apporter le maximum possible pour le fonctionnement de l'établissement et pour la qualité de vie de ses résidents. Développer un système d'aide à la décision pour l'optimisation des plans d'actions est une solution qui permettra au décideur d'atteindre ce but. Le problème d'optimisation de plans d'actions est décrit en détail dans le chapitre 2.

1.3 Optimisation multi-objectif (OMO)

L'optimisation est un domaine fondamental de l'aide à la décision dans de nombreux milieux scientifiques et industriels. Au cours des dernières décennies, un très grand nombre de travaux de recherche, à la fois théoriques et pratiques, ont été menés dans ce domaine pour résoudre des problèmes avec différents niveaux de complexité. L'idée de base de la résolution d'un problème d'optimisation consiste à trouver parmi un ensemble de solutions celle qui correspond le plus aux préférences du décideur. La meilleure solution peut être trouvée d'une manière exacte dans le cas où le nombre de solutions réalisables et la complexité du problème ne sont pas trop élevés. Par contre, quand le nombre de solutions possibles est trop élevé (ce qui est le cas dans beaucoup de problèmes), on ne peut pas évaluer toutes les solutions, on se contente alors de trouver une solution acceptable dans un temps raisonnable. D'autant plus que les problèmes pratiques et industriels sont souvent de nature multi-objectif (plusieurs critères à optimiser en même temps), ce qui les rend encore plus difficiles. En effet, la solution à un problème multi-objectif est rarement une solution unique, c'est plutôt un ensemble de solutions de bon compromis (ensemble Pareto optimal). L'une des questions les plus difficiles est donc de déterminer cet ensemble Pareto optimal, ou une approximation de celui-ci, quand le problème est complexe.

Dans la section suivante, nous donnons quelques définitions et notations liées à l'optimisation multi-objectif dont de nombreuses sont tirées du livre de Ehrgott [Ehrgott, 2005], puis quelques concepts sur les méthodes de résolution de tels problèmes (section 1.3.2).

1.3.1 Définitions

1.3.1.1 Formulation d'un problème d'optimisation multi-objectif

Commençons d'abord par définir le problème d'optimisation mono-objectif.

Un problème d'optimisation mono-objectif consiste à trouver, parmi un ensemble de solutions potentielles, une solution optimale, par rapport à un objectif ou un critère. D'une manière plus formelle, à chaque instance d'un tel problème, est associé un ensemble X de solutions potentielles et une fonction de coût f qui associe à chaque solution $x \in X$ une valeur numérique $f(x)$. Résoudre un tel problème d'optimisation consiste à trouver une solution x^* (dite *solution optimale*) qui minimise (ou maximise) la fonction f . Voici une définition tirée du livre de Papadimitriou et Steiglitz [Papadimitriou and Steiglitz, 1982] :

Définition 1. Une instance d'un problème de minimisation est un couple (X, f) où X est un ensemble de solutions ou de configurations potentielles, et f une fonction $f : x \mapsto \mathbb{R}$. Le but est de trouver $x^* \in X$ tel que $f(x^*) \leq f(x)$, pour tout $x \in X$.

Dans un problème d'optimisation multi-objectif, le nombre d'objectifs à minimiser (ou à maximiser) simultanément est supérieur ou égal à 2. Par conséquent, il n'est pas possible de trouver une solution unique qui optimise à la fois tous les objectifs considérés, une solution qui optimise un ou plusieurs objectifs peut détériorer d'autres objectifs.

Définition 2. *Un problème d'optimisation multi-objectif (multiobjective optimization problem, MOP) peut être défini comme suit :*

$$(MOP) = \begin{cases} \text{“min”} & f(x) = (f_1(x), f_2(x), \dots, f_m(x)) \\ \text{tel que} & x \in X \end{cases} \quad (1.1)$$

où m est le nombre d'objectifs ($m \geq 2$), X est l'ensemble des solutions réalisables et $x = (x_1, x_2, \dots, x_n) \in X$ est un vecteur représentant une de ses solutions.

1.3.1.2 Variables de décision et espace décisionnel

Les variables de décision d'un problème d'optimisation multi-objectif sont des variables pour lesquelles il faut définir des valeurs. Cet ensemble de variables est appelé *vecteur de décision*. Pour un problème d'optimisation à n variables de décision, le vecteur de décision correspondant est représenté par :

$$\vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ x_n \end{pmatrix} \quad (1.2)$$

Les différentes valeurs prises par les variables de décision x_i constituent l'ensemble de solutions envisageables. On parle de problèmes d'optimisation continue quand les domaines de définition des variables de décisions sont continus (souvent dans \mathbb{R}) et de problèmes d'optimisation combinatoire quand les domaines de définition des variables de décision sont discrets (souvent booléen ou dans \mathbb{N}). L'espace de dimension n (n étant le nombre de variables de décision) est appelé *espace décisionnel*, noté X .

1.3.1.3 Vecteur objectif et espace objectif

A chaque solution $x \in X$ est associé un vecteur objectif $y \in Y$ constitué d'un vecteur de fonctions $f : X \mapsto Y$ tel que $y = (y_1, y_2, \dots, y_m) = f(x) = (f_1(x), f_2(x), \dots, f_m(x))$. Où $Y = f(X)$ représente l'ensemble des points réalisables dans un espace défini par l'ensemble de définition des fonctions objectifs, appelé *espace objectif* et noté Y . Généralement, l'espace objectif est défini dans \mathbb{R} dans le cas d'une optimisation mono-objectif et dans \mathbb{R}^m dans le cas d'une optimisation m -objectif. Sans perte de généralité, nous supposons dans ce qui suit que $Y \subseteq \mathbb{R}^m$ et que tous les objectifs sont à minimiser.

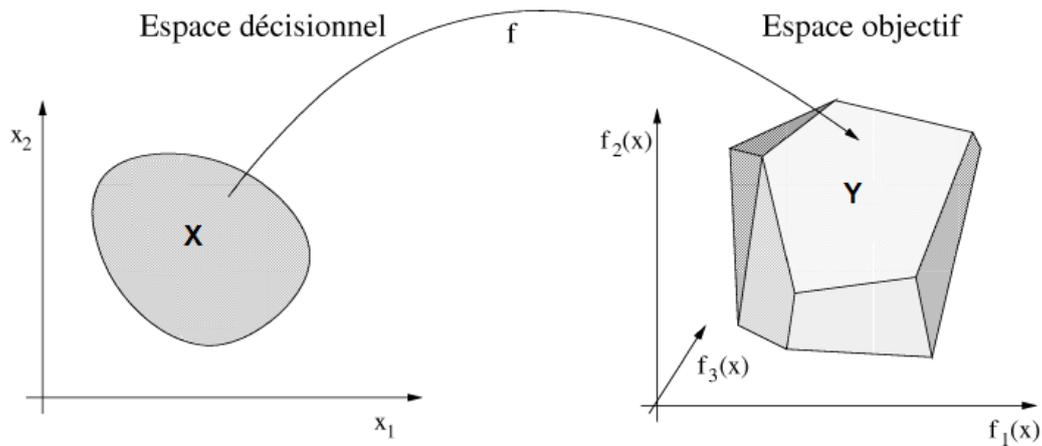


FIGURE 1.13 – Problème d'optimisation multi-objectif : cas avec deux variables de décision et trois fonctions objectifs [Basseur, 2005].

1.3.1.4 Notions de dominance et d'optimalité Pareto

Pour résoudre un problème multi-objectif, le décideur évalue généralement la solution par rapport à chacun des critères. Néanmoins, contrairement aux problèmes mono-objectifs où il existe une relation d'ordre entre les différentes solutions, dans le cas multi-objectif, il n'existe généralement aucune solution qui serait optimale pour tous les objectifs, étant donné le caractère conflictuel de ces derniers.

La notion d'optimalité la plus communément admise est celle introduite par Edgeworth en 1881 puis généralisée par Pareto en 1896, connue sous le nom de l'optimalité de Pareto. Il convient donc de définir la notion de Pareto optimalité et les relations qui en découlent.

Définition 3 (Dominance Pareto). *Un vecteur objectif $y_1 \in Y$ domine un autre vecteur objectif $y_2 \in Y$ ssi $\forall j \in \{1, \dots, m\}, y_1^j \leq y_2^j$ et $\exists k \in \{1, \dots, m\}, y_1^k < y_2^k$. Cette relation sera notée $y_1 \prec y_2$. Par extension, une solution $x_1 \in X$ domine une solution $x_2 \in X$ si $f(x_1) \prec f(x_2)$, on note $x_1 \prec x_2$.*

La relation $x_1 \prec x_2$ signifie que la solution x_1 est meilleure (au sens de l'optimalité Pareto) que la solution x_2 .

Définition 4 (Vecteur non-dominé). *Un vecteur objectif $y_1 \in Y$ est non-dominé ssi $\forall y_2 \in Y, y_2 \not\prec y_1$ (figure 1.14).*

Définition 5 (Solution Pareto optimale). *Une solution $x \in X$ est Pareto optimale si et seulement si il n'existe pas une autre solution $x' \in X$ qui domine x .*

L'ensemble des solutions Pareto optimales forme l'ensemble Pareto optimal, noté X_P . L'image de cet ensemble dans l'espace objectif est le front Pareto. Il sera noté PO .

Définition 6 (Ensemble Pareto optimal). *Pour un problème d'optimisation multi-objectif $F(X)$, l'ensemble Pareto optimal est défini comme suit : $X_P = \{x \in X \mid \nexists x' \in X \prec x\}$*

Trouver le vrai front Pareto n'est pas une tâche facile, en particulier sur les problèmes de grande taille, mais l'approximation de ce front est généralement possible, en particulier avec les métaheuristiques (figure 1.15).

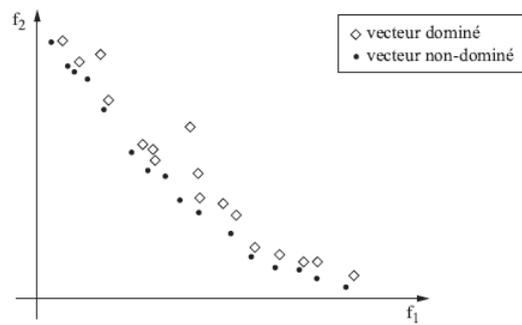


FIGURE 1.14 – Vecteurs dominés et vecteurs non dominés dans l’espace objectif.

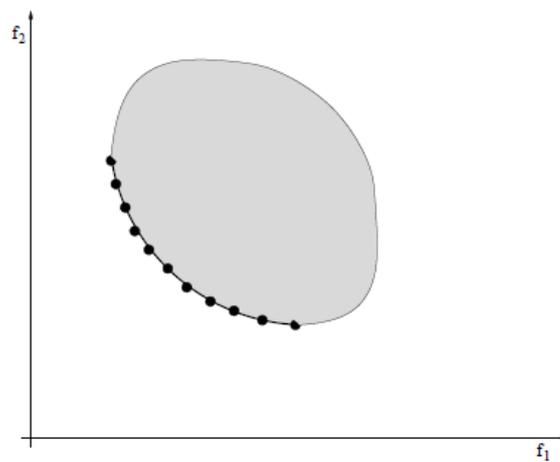


FIGURE 1.15 – Exemple d’approximation du front Pareto.

Il existe d’autres types de dominances différentes de la dominance Pareto, telles que la dominance stricte, la dominance faible et l’ ϵ -dominance. Leurs définitions sont données ci-après.

Définition 7 (Dominance stricte). *Un vecteur objectif $y_1 \in Y$ domine strictement un autre vecteur objectif $y_2 \in Y$ ssi $\forall j \in \{1, \dots, m\}, y_1^j < y_2^j$.*

Définition 8 (Dominance faible). *Un vecteur objectif $y_1 \in Y$ domine faiblement un autre vecteur objectif $y_2 \in Y$ ssi $\forall j \in \{1, \dots, m\}, y_1^j \leq y_2^j$. Elle sera notée \preceq .*

Définition 9 (ϵ -dominance). *Un vecteur objectif $y_1 \in Y$ ϵ -domine un autre vecteur objectif $y_2 \in Y$ ssi $\forall j \in \{1, \dots, m\}, y_1^j \leq \epsilon \cdot y_2^j$, avec $\epsilon > 1$. Elle sera notée \preceq_ϵ .*

1.3.1.5 Point idéal, point nadir et point utopique

En supposant que les optimums correspondant à toutes les fonctions objectifs sont connus, nous pouvons définir les notions de point idéal, point nadir et point utopique, comme suit :

Définition 10 (Point idéal). *Un point idéal $y^* = (y_1^*, y_2^*, \dots, y_m^*)$ est le vecteur qui minimise chaque fonction objectif individuellement, $y_j^* = \min_{x \in X} f(x), \forall j \in \{1, \dots, m\}$.*

Définition 11 (Point nadir). *Un point nadir $y^{nad} = (y_1^{nad}, y_2^{nad}, \dots, y_m^{nad})$ est le vecteur qui maximise chaque fonction objectif parmi les solutions de l'ensemble Pareto optimal, $y_j^{nad} = \max_{x \in X_P} f(x)$, $\forall j \in \{1, \dots, m\}$.*

Définition 12 (Point utopique). *Un point utopique $y^{ut} = (y_1^{ut}, y_2^{ut}, \dots, y_m^{ut})$ est le vecteur irréalisable donné par $y^{nad} - \epsilon$ avec ϵ un vecteur de petites valeurs scalaires strictement positives.*

1.3.2 Quelques concepts sur l'optimisation multi-objectif

Il existe un grand nombre d'approches pour résoudre les problèmes multi-objectifs. Certaines approches non Pareto exploitent des connaissances sur le problème pour définir des préférences sur les différents objectifs et contourner ainsi l'aspect multi-objectif du problème. Avec ce type d'approche, le problème multi-objectif n'est pas traité comme tel, il est plutôt transformé en un ou plusieurs problèmes mono-objectifs. Par opposition aux méthodes non Pareto, les approches Pareto considèrent tous les objectifs avec le même niveau d'importance (une priorisation des objectifs peut être quand même définie par le décideur dans son modèle). Là aussi il existe plusieurs méthodes et approches de résolution. Une synthèse des états de l'art résumant ces approches peut être consulté dans [Hao *et al.*, 1999; Ulungu and Teghem, 1994; Miettinen, 1999; Ehrgott, 2000; Siarry and Collette, 2002]. Nous présentons, ci-dessous, quelques concepts sur les méthodes de résolution Pareto fondées sur des métaheuristiques, la recherche locale et les algorithmes évolutionnaires sur lesquels sont basés les algorithmes présentés dans cette thèse.

1.3.2.1 Voisinage

Dans un contexte de l'optimisation combinatoire, surtout dans le cas multi-objectif, la taille de l'espace de recherche augmente avec la taille de l'instance à résoudre. Ces problèmes d'optimisation deviennent plus difficiles quand les solutions sont nombreuses et non complètement énumérables en un temps raisonnable. La relation de voisinage permet de connecter entre elles les solutions de l'espace de recherche. Un voisinage est une application qui associe à toute solution de l'espace de recherche un ensemble de solutions (définition 13).

Définition 13 (Relation de voisinage). *Soit X l'espace de recherche d'un problème d'optimisation. Une fonction de voisinage \mathcal{N} est une association $\mathcal{N} : X \rightarrow 2^X$, définissant, pour chaque point $x \in X$, un ensemble $\mathcal{N}(x) \subseteq X$ de points proches de x .*

1.3.2.2 Métaheuristiques

Une métaheuristique est un algorithme d'optimisation visant à résoudre des problèmes d'optimisation difficiles que nous ne pouvons pas résoudre d'une manière exacte. Les métaheuristiques sont généralement des algorithmes stochastiques itératifs, qui, partant d'une solution (ou population) initiale, progressent vers un optimum local en tentant d'apprendre les caractéristiques d'un problème afin de trouver une approximation de la meilleure solution. Ainsi, les métaheuristiques permettent de s'attaquer à des problèmes complexes de grande taille en délivrant des solutions satisfaisantes en un temps raisonnable. Il existe un grand nombre de métaheuristiques différentes, allant de la simple recherche locale à des algorithmes complexes de recherche. Ces méthodes sont a priori définies avec un haut niveau d'abstraction, leur permettant d'être adaptables à une large gamme de problèmes différents. Elles sont composées

d'éléments génériques et d'éléments spécifiques au problème considéré, tels que la représentation ou l'évaluation d'une solution. Les métaheuristiques ont reçu, ces dernières années, un intérêt grandissant de la communauté scientifique et ont montré leur efficacité pour de nombreux problèmes industriels [Talbi, 2009].

Nous pouvons distinguer deux types de métaheuristiques : des *métaheuristiques à base de solution unique* qui manipulent et transforment une seule solution durant tout le processus de recherche, telles que les algorithmes de recherche locale, le recuit simulé, la recherche Tabou, etc. Le deuxième type sont des *métaheuristiques à base de population de solutions*, où un ensemble de solutions (la population) évolue simultanément durant tout le processus de recherche. Dans la conception des métaheuristiques à base de population, deux critères sont usuellement considérés : l'exploration de l'espace de recherche (*diversification*) et l'amélioration des solutions déjà trouvées (*intensification*). Plusieurs approches à base de population sont largement utilisées dans l'optimisation multi-objectif, comme les algorithmes évolutionnaires, les algorithmes génétiques, les algorithmes à essaim de particules, etc.

1.3.2.3 Algorithmes évolutionnaires

Les algorithmes évolutionnaires sont des techniques itératives inspirées des concepts de la théorie d'évolution de Darwin en biologie. Un algorithme évolutionnaire simule un processus d'évolution d'une population d'individus dont le but est de les faire évoluer vers des individus de bonne qualité. Nous pouvons distinguer trois classes d'algorithmes évolutionnaires : les algorithmes génétiques, les stratégies d'évolution et la programmation évolutive [Coello et al., 2007]. Ces méthodes se différencient principalement dans leurs manières de représenter l'information et la façon de faire évoluer la population d'une génération à une autre. Un algorithme évolutionnaire est composé de trois éléments de base :

- une *population* : un ensemble d'individus représentant des solutions potentielles au problème considéré à un moment du processus de recherche ;
- un *mécanisme d'évaluation* : permettant d'évaluer la qualité des individus de la population ;
- un *mécanisme d'évolution* : permettant, grâce à certains opérateurs prédéfinis (la *recombinaison*, la *mutation* et la *sélection*), d'éliminer les mauvais individus et de faire évoluer les meilleurs.

La *sélection* a pour objectif de sélectionner des individus qui vont pouvoir se reproduire pour transmettre leurs caractéristiques à la génération suivante. Le *croisement* ou *recombinaison* est un opérateur permettant de construire de nouveaux individus enfants à partir des caractéristiques d'individus parents sélectionnés. La *mutation* effectue de légères modifications de certains individus (crée des perturbations dans la structure des individus).

1.3.2.4 Recherche locale

Les algorithmes utilisant la recherche locale font évoluer la solution courante en la remplaçant par une autre issue de son voisinage, ce changement de solution est couramment appelé *un mouvement*. Ce processus est réitéré jusqu'à ce qu'une solution optimale ait été trouvée ou une condition d'arrêt soit vérifiée.

Toutes les approches de recherche locale ont des méthodes fondées sur une relation de voisinage et sur une procédure exploitant ce voisinage. Un aspect fondamental de ces approches est donc la détermination de ce voisinage. Le voisinage est souvent représenté par une fonction \mathcal{N} qui, à une configuration x , associe un ensemble d'autres configurations $\mathcal{N}(x)$. Il existe une infinité de manières de choisir le voisinage \mathcal{N} , par conséquent, il convient d'adapter ce choix au problème, c'est-à-dire choisir la meilleure fonction \mathcal{N} selon le problème considéré.

1.3.2.5 Valeur de fitness

La valeur de *fitness* d'une solution de l'espace de recherche est la valeur de la fonction objectif pour cette solution. Évaluer une solution consiste à calculer sa valeur de fitness. Dans le cas d'un problème mono-objectif, la valeur de fitness d'une solution est généralement sa valeur objectif unidimensionnelle. Dans le cas multi-objectif, la valeur de fitness d'une solution vise à orienter la recherche vers les solutions Pareto optimales. Quatre classes d'approches peuvent être utilisées pour la définition de la fonction de fitness : approches scalaires, approches basées sur un critère, approches basées sur une relation de dominance et des approches utilisant des indicateurs. Ces différentes approches sont largement décrites dans la thèse d'Arnaud Liefoghe [[Liefoghe, 2009](#)].

1.3.2.6 Optimisation multi-objectif et aide à la décision

Résoudre un problème pratique d'optimisation, revient toujours à répondre à une problématique exprimée par le décideur. Il est donc impératif que l'algorithme fournisse une ou plusieurs solution(s) correspondant aux attentes et aux préférences du décideur. Par conséquent, la coopération entre ce dernier et la méthode de résolution est presque une condition pour trouver les "bonnes" solutions. Cette coopération peut prendre trois formes différentes :

1. **A priori.** Dans ce cas, le décideur fournit ses connaissances sur le problème et ses préférences par rapport aux solutions à chercher avant le processus d'optimisation, ceci en vue d'aider la méthode de résolution à explorer les zones les plus intéressantes de l'espace objectif. Néanmoins, en pratique, l'expression et la modélisation des préférences du décideur ne sont pas toujours faciles, elles doivent donc être validées avant la lancement de la recherche. De plus, le décideur peut ne pas être satisfait des solutions fournies par la méthode et vouloir relancer le processus de résolution avec de nouvelles préférences.
2. **A posteriori.** Dans ce cas, la méthode de résolution vise à trouver l'ensemble des solutions du front Pareto ou une bonne approximation de celui-ci. Le décideur n'a donc qu'à choisir parmi les solutions trouvées celle qui correspond mieux à ces préférences. Ce scénario ne nécessite pas la modélisation des préférences du décideur préalablement. Par contre, il présente deux inconvénients : le fait de chercher toutes les solutions non dominées peut s'avérer très coûteux en temps de calcul, particulièrement pour les grandes instances. Aussi, le nombre de solutions non dominées peut être si grand faisant de leur analyse et le choix de la solution adéquate une tâche difficile.
3. **Interactive.** Avec ce scénario, le décideur coopère d'une manière directe, itérative et progressive avec la méthode de résolution. Ainsi, à partir des connaissances acquises lors de la première résolution, il peut raffiner ses préférences et relancer la résolution. Ce processus est réitéré plusieurs fois jusqu'à ce que le décideur soit satisfait. L'avantage de

cette approche, c'est que le décideur n'est pas obligé d'exprimer ses préférences d'une manière exacte et interchangeable dès le début, il peut les adapter pas à pas. Toutefois, cette approche nécessite sa participation tout au long du processus de résolution.

1.4 Conclusion

Nous avons présenté dans ce chapitre quelques éléments indispensables pour comprendre et appréhender le **SSMS** avec toutes ses spécificités légales, sociales et économiques. Dans un contexte de vieillissement de la population et de l'augmentation des besoins de prise en charge des personnes vulnérables, le **SSMS** se donne pour mission de tendre la main et d'apporter toutes les aides nécessaires à ces catégories de personnes en leur offrant une grande diversité de services adaptés (parfois même personnalisés). Néanmoins, la difficulté de la conjoncture économique fait que pour assurer cette mission sociale et humanitaire, les établissements doivent innover et s'adapter pour continuer à garantir une qualité de service à la hauteur de leurs missions.

En effet, depuis cinq ou six ans, les établissements voient leurs budgets diminués d'année en année alors que les besoins de leurs résidents ne cessent d'augmenter. De plus, pour suivre la politique de modernisation, d'ouverture et d'innovation visée par l'État et les collectivités locales, les décideurs du secteur doivent travailler et collaborer ensemble pour mutualiser leurs moyens et engager des projets de recherche et de développement communs.

Un des challenges auquel sont confrontés les directeurs des structures est d'arriver à élaborer et mettre en œuvre des plans d'actions multi-objectifs efficaces, même dans un contexte budgétaire difficile. Après une description détaillée et une modélisation formelle de ce problème, données dans le chapitre 2, nous présentons, dans les chapitres 3 et 4, plusieurs algorithmes permettant de les aider à relever ce challenge.

Le problème d'optimisation de plans d'actions multi-objectifs (POPAMO)

Dans la première partie de ce chapitre, nous présentons le problème d'optimisation de plans d'actions multi-objectifs ainsi que les éléments pratiques qui le définissent. Puis, nous introduisons notre première contribution, i.e., une modélisation formelle du problème sous forme d'un problème de sac-à-dos multi-objectif. Dans la deuxième partie, nous présentons la solution que nous proposons pour assister le décideur dans l'élaboration de plans d'actions optimaux. Nous terminons par un état de l'art des principaux travaux connus sur ce problème.

Sommaire

2.1	Introduction	38
2.2	Présentation du POPAMO	39
2.2.1	Les objectifs	41
2.2.2	Les actions	41
2.2.3	Les contraintes	42
2.2.4	Les sorties	42
2.3	Modélisation du POPAMO	42
2.3.1	Les actions (espace décisionnel)	43
2.3.2	Les objectifs (espace objectif)	43
2.3.3	Les plans d'actions	43
2.3.4	Les contraintes	43
2.3.5	Formulation du POPAMO	44
2.4	MSQualité pour le POPAMO	44
2.4.1	Aide à la définition des actions	45
2.4.2	Aide aux choix des actions	46

2.5 État de l'art	46
2.5.1 Problème d'optimisation de plans d'actions	46
2.5.2 Problème du sac-à-dos multi-objectif	47
2.6 Conclusion	52

2.1 Introduction

Comme nous l'avons vu dans le chapitre précédent, la loi du 02 janvier 2002 a été définie pour promouvoir l'autonomie et la protection des personnes qui sont dans le besoin et lutter contre leur exclusion sociale. Pour aider les responsables du **SSMS** à atteindre ce but, la loi a précisé les missions et les champs d'action de chaque type de structures. En effet, quel que soit l'établissement et les besoins des personnes qui y sont accueillies, le responsable doit toujours agir continuellement sur plusieurs aspects dans la gestion de son établissement et la prise en charge de ses résidents. Dans les missions et les préconisations qui sont définies par la loi du 2 janvier 2002 (puis complétées par la loi du 5 février 2005 et celle du 22 juillet 2009), le directeur se trouve dans l'obligation de mettre en place des plans d'actions améliorant à la fois les "*performances*" de l'établissement et la qualité de vie des résidents. De ce fait, son plan d'actions doit inclure des actions pour la protection administrative et judiciaire des résidents, des actions éducatives (quand il s'agit d'établissements pour de jeunes enfants) ou d'aide à l'insertion professionnelle adaptées aux besoins de la personne, à son niveau de développement, à ses potentialités, à l'évolution de son état ainsi qu'à son âge, des actions d'assistance dans les divers actes de la vie, de soutien, de soins et d'accompagnement, des actions pour la prévention des risques sociaux et psychosociaux, des actions de formation pour le personnel, des actions d'investissements, etc.

De plus, pour assurer le contrôle et le suivi de l'établissement, la loi du 2 janvier 2002 a instauré un système d'auto-évaluations quinquennales et d'évaluations externes septennales. Le but de ces deux échéances est, suivant un référentiel adapté, de réaliser régulièrement un audit sur la situation de l'établissement et la qualité de la prise en charge qu'il dispense à ses résidents. Suivant les résultats de l'évaluation et les recommandations de l'évaluateur quand il s'agit d'une évaluation externe, le responsable doit mettre en place un projet d'amélioration avec un plan d'actions pluriannuel afin de remédier aux éventuels problèmes et manques qui peuvent être constatés lors de l'audit, d'un côté, et améliorer ce qui fonctionne bien, d'un autre côté. Et dans le but d'une amélioration continue de la qualité, ce cycle est réitéré tout au long de la vie de l'établissement (figure 2.1). A cela, s'ajoute un autre plan d'actions également important, défini dans le cadre du **CPOM**.

Le but de ce chapitre est de présenter le processus d'élaboration de ces différents plans d'actions et leurs éléments, ainsi que les contraintes et les difficultés auxquelles sont confrontés les décideurs, suivi d'une modélisation formelle du problème d'optimisation des plans d'actions sous forme d'un problème de sac-à-dos multi-objectif, notre première contribution de cette thèse. Puis, nous montrons dans la section 2.4 comment le logiciel *MSQualité* peut aider les décideurs dans la définition et l'optimisation de leurs plans d'actions. Enfin, nous terminons le chapitre par un aperçu des principaux travaux déjà publiés sur ce problème.

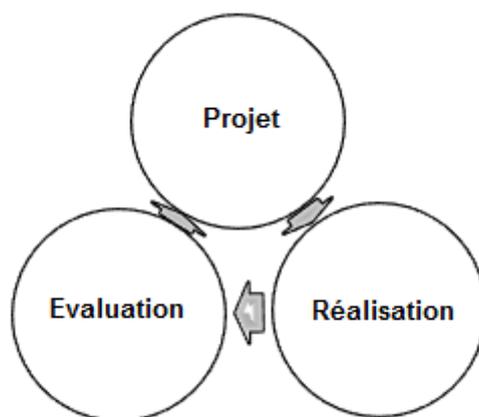


FIGURE 2.1 – Le cycle d'évaluations et de plans d'actions dans le SSMS

2.2 Présentation du POPAMO

Les établissements du SSMS sont continuellement dans une démarche d'amélioration de leurs systèmes qualité. Dès qu'un projet est terminé, ses résultats sont évalués et un nouveau projet démarre pour la période suivante (pour une durée de 5 ans en général). Et comme dans tout projet, le responsable doit définir les objectifs à atteindre, les actions à mener, les éventuelles contraintes à considérer et enfin les moyens qui peuvent être consacrés au projet. Par conséquent, un plan d'actions est composé de quatre éléments constitutifs du projet : les objectifs, les actions, les contraintes et les moyens.

Le problème d'élaboration de plan d'actions consiste donc à définir les meilleures actions à mettre en place dans la structure pour atteindre les objectifs du projet et améliorer ainsi la qualité globale de l'établissement, en tenant compte des moyens disponibles et des contraintes à respecter. Actuellement, pour élaborer un plan d'actions, le décideur procède d'une manière incrémentale, il commence d'abord par définir les objectifs qu'il souhaite atteindre ainsi que les contraintes, puis, dans un second temps, il réfléchit aux éventuelles actions qui peuvent contribuer à l'atteinte de ces objectifs. Vu que chaque action possède un coût de réalisation et que le budget limité ne permet pas de les réaliser dans leur globalité, le décideur n'a d'autre choix que de sélectionner une partie des actions (celles qui lui "semblent" les plus intéressantes selon les objectifs définis), de telle sorte que leur coût total rentre dans son budget (figure 2.2). Nous précisons que dans ce travail, on considère uniquement les moyens financiers parce que c'est le principal obstacle qui fait que le décideur ne peut pas réaliser toutes les actions possibles.

Mais lors de l'élaboration du projet, le décideur est généralement confronté à deux principaux défis :

1. Quelles sont les actions qui peuvent être mises en œuvre pour que tous les objectifs soient atteints et que la qualité globale de la structure soit améliorée, d'autant plus que certains objectifs peuvent être conflictuels ? En effet, à l'état actuel du fonctionnement des établissements, il n'y a pas toujours suffisamment d'éléments permettant de savoir quel serait l'impact d'une action sur la qualité de l'établissement avant qu'elle soit réalisée. Particulièrement les actions qui concernent la qualité de vie des résidents sont difficiles à évaluer, d'autant plus avant la mise en place même des actions. De ce fait, les résultats de

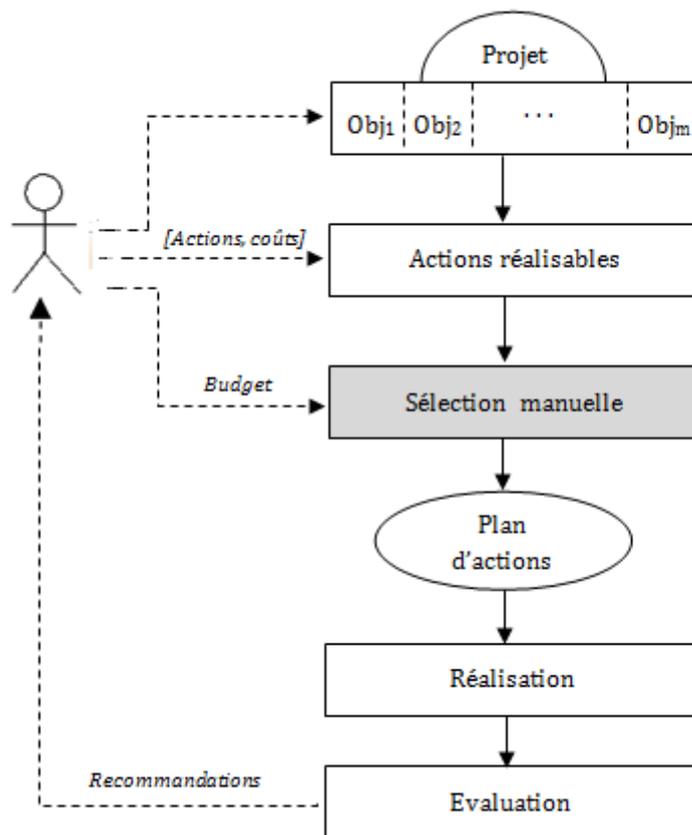


FIGURE 2.2 – Processus d’élaboration des plans d’actions dans les établissements du **SSMS**

certain types d’actions correspondent plus ou moins à ce que le décideur attendait mais pour d’autres types d’actions, le résultat pourrait être tout à fait contraire aux attentes du décideur.

2. Quelles actions faut-il choisir lorsque le budget du projet ne permet pas de mettre en œuvre toutes les actions possibles ? En effet, la majeure partie des actions réalisables ont un coût positif (dans certains types d’établissements, comme les **ESAT**, il peut y avoir des actions qui rapportent de l’argent, dans ce cas, nous considérons que l’action a un coût négatif). Par conséquent, avoir les moyens financiers pour réaliser toutes les actions est presque une mission impossible pour la plupart des établissements, particulièrement dans le contexte budgétaire difficile de ces dernières années. De plus, une même action peut contribuer à l’atteinte d’un (ou plusieurs) objectif(s) mais en même temps elle peut aussi nous éloigner d’un autre objectif. En d’autres termes, le défi consiste à choisir, parmi l’ensemble des actions possibles, un sous-ensemble permettant d’obtenir le plan d’actions le plus efficace possible tout en respectant la limite du budget disponible.

A notre connaissance, il n’y a aucun outil d’aide à la décision dans le **SSMS** permettant d’assister le décideur à relever ces deux défis. Pour définir les actions réalisables pouvant contribuer à l’amélioration de la qualité de l’établissement et du bien-être de ses résidents, le décideur se base sur sa connaissance de l’établissement et des besoins des personnes qui y sont accueillies. Pour le choix des actions, il se base sur ses priorités, son expertise et parfois même son intuition.

Dans la section 2.4, nous décrivons comment, avec le logiciel « *MSQualité* », nous apportons une aide considérable au décideur pour relever ces défis. Mais avant cela, donnons d'abord quelques précisions sur la nature des objectifs, des actions et des contraintes que nous pouvons avoir dans un plan d'actions.

2.2.1 Les objectifs

Les objectifs d'un plan d'actions sont les mêmes que ceux définis dans le projet pour lequel le plan d'actions a été élaboré. La nature des objectifs dépend principalement du type de l'établissement, de ses missions ainsi que de la nature du projet lui-même. Vu que la plupart des projets des établissements sociaux et médico-sociaux visent à améliorer leur qualité globale et assurer une meilleure prise en charge des résidents, les objectifs sont également souvent de nature qualitative. Certains concernent l'établissement lui-même tels que : « *améliorer l'ouverture de l'établissement à son environnement* », « *développer le partenariat* », « *améliorer la sécurité et le niveau de prévention des risques* ». D'autres concernent tout ce qui est en rapport avec la vie et le bien-être des résidents qui y sont accueillis, tels que : « *améliorer la qualité de vie des résidents* », « *améliorer l'autonomie des résidents* », « *assurer les droits individuels et collectifs* », etc. Nous considérons dans ce travail que le décideur est libre de définir ses objectifs, quelles que soient leurs natures. Cependant, même qualitatifs, certains objectifs peuvent être conflictuels. A titre d'exemple, pour améliorer la sécurité dans l'établissement ou la sécurité d'une catégorie de personnes, le décideur peut être amené à mettre en place des actions au détriment de la liberté des personnes concernées. Aussi, le décideur doit pouvoir pondérer ses objectifs en fonction de ses priorités.

2.2.2 Les actions

Les actions que nous trouvons dans un plan d'actions sont aussi variées que les objectifs pour lesquels le plan d'actions est mis en œuvre. Par conséquent, un plan d'actions peut être composé de deux types d'actions

- des actions améliorant la gestion et le fonctionnement de l'établissement comme structure, telles que la rédaction de nouvelles procédures, le recrutement de nouvelles personnes, des actions de formation d'adaptation ou de perfectionnement, acquisition de nouveaux matériels, construction d'un atelier, etc.
- des actions améliorant la qualité de prise en charge des résidents et leur accompagnement sur tous les plans. Ces actions dépendent beaucoup du type des personnes prises en charge par l'établissement. Par exemple, dans un établissement pour personnes âgées, nous trouvons dans le plan d'actions beaucoup d'actions sur la prise en charge sanitaire comme l'acquisition de nouveaux équipements de soins, la formation du personnel sur de nouveaux protocoles de soins, l'organisation d'activités et d'animations de loisirs en interne ou en externe, l'organisation de rencontres familles-résidents, l'organisation de sortie, etc. Dans un établissement pour jeunes enfants nous trouvons beaucoup d'actions d'éducation, de formation et de loisirs.

Cependant, les impacts de ces deux types d'actions ne sont pas complètement séparés en pratique. Quand le décideur met en place des actions pour améliorer la gestion de l'établissement ou pour former le personnel sur de nouveaux outils ou techniques, ces actions auront

forcément un impact, direct ou indirect, sur la qualité de service offert aux résidents. De la même manière, bien prendre en charge ses résidents contribue positivement à l'image de l'établissement et à la satisfaction des résidents, de leurs familles et du personnel, d'un côté, et la satisfaction des autorités, de l'autre côté. De ce fait, une même action peut impacter (positivement ou négativement) tous ou une partie des objectifs du plan d'actions. Aussi, comme cela a été mentionné précédemment, une action peut avoir un coût positif quand la réalisation de l'action coûte de l'argent à l'établissement, un coût négatif quand l'action génère un bénéfice pour l'établissement ou un coût nul (dans ce travail, on ne considère que les coûts qui rentrent dans le budget alloué au projet, pas les coûts de fonctionnement tels que les salaires).

2.2.3 Les contraintes

Le problème d'optimisation de plans d'actions est soumis à deux contraintes dures :

- le coût de réalisation du plan d'actions retenu ne doit pas dépasser le budget alloué au projet ;
- le décideur peut définir un seuil minimal à atteindre pour chacun des objectifs après l'achèvement du plan d'actions. Le plan retenu doit donc garantir qu'après sa réalisation ces seuils soient atteints ou dépassés.

Nous notons également qu'à la réalisation du plan d'actions, il pourrait y avoir d'autres contraintes liées au planning du personnel ou aux disponibilités de certains équipements ou matériels nécessaires à la mise en œuvre de quelques actions. Dans ce travail, nous considérons uniquement les deux contraintes liées à la génération de plans d'actions optimaux, pas celles concernant sa mise en œuvre.

2.2.4 Les sorties

La sortie (donc la solution) du problème est un plan d'actions ayant un certain nombre d'actions à réaliser, un niveau pour chaque objectif supérieur ou égal au seuil défini par le décideur et un coût de réalisation.

2.3 Modélisation du POPAMO

Le POPAMO consiste à élaborer un plan d'actions efficace pour atteindre les objectifs d'un projet tout en respectant ses contraintes. En d'autres termes, l'objectif est de choisir un sous-ensemble d'actions parmi de nombreuses actions possibles tout en optimisant plusieurs objectifs conflictuels et en respectant un ensemble de contraintes. Chaque action a un coût de réalisation et peut influencer (positivement ou négativement) une partie ou la totalité des objectifs. Le POPAMO est exactement un cas pratique du problème du sac-à-dos multi-objectif [Lust and Teghem, 2012; Barichard and Hao, 2003] qui est régulièrement étudié dans la littérature. Par analogie, le plan d'actions représente le sac-à-dos, les objets (les items) à mettre dans le sac-à-dos sont les actions du plan, les coûts et les profits des objets sont respectivement les coûts de réalisation des actions et leurs impacts sur les objectifs du projet. Une modélisation formelle du POPAMO est donnée ci-dessous.

2.3.1 Les actions (espace décisionnel)

Soit A l'ensemble des actions réalisables. Chaque action $a_i \in A$ est une variable de décision. La valeur de la variable a_i est égale à 1 si l'action a_i est sélectionnée pour être réalisée dans le plan d'actions potentiellement solution et $a_i = 0$ dans le cas contraire. Le coût de réalisation de l'action a_i est donné par la variable $\omega_i \in \mathbb{R}$.

2.3.2 Les objectifs (espace objectif)

Chaque objectif j défini pour le plan d'actions est représenté par une fonction objectif f_j qui associe à chaque action $a_i \in A$ un impact sur l'objectif j . Cet impact est évalué par la fonction $f_j(a_i) = v_{ij}$ qui assigne à l'action a_i une valeur entière reflétant la contribution de l'action a_i à l'atteinte de l'objectif j ($v_{ij} > 0$) ou la dégradation de l'action a_i pour l'objectif j ($v_{ij} < 0$). $v_{ij} = 0$ indique que l'action a_i n'a aucun effet sur l'objectif j . L'ensemble des poids que le décideur pourrait définir pour les objectifs est donné par K et le poids de l'objectif j par k_j . L'ensemble des fonctions à maximiser est donné par F .

2.3.3 Les plans d'actions

Dans ce qui suit, un plan d'actions est représenté par la variable x et l'ensemble des plans d'actions possibles formant l'espace de recherche est donné par X . Un plan d'actions x est un sous-ensemble d'actions sélectionnées parmi l'ensemble des actions réalisables A . Il est représenté par un vecteur décisionnel $x = (a_1, a_2, \dots, a_n) \in \{0, 1\}^n$ avec n égal à la taille de A et $a_i = 1$ si l'action $a_i \in x$ et $a_i = 0$ si $a_i \notin x$. L'impact qu'a le plan x sur l'objectif j est obtenu par la somme des impacts des actions qui le composent sur l'objectif j :

$$f_j(x) = \sum_{i=1}^n a_i f_j(a_i) \quad (2.1)$$

Pareillement, le coût de réalisation du plan d'actions x est donné par la somme des coûts de réalisation des actions qui le composent :

$$W(x) = \sum_{i=1}^n a_i \omega_i \quad (2.2)$$

2.3.4 Les contraintes

Nous avons vu dans la section 2.2.3 que toute solution au problème doit respecter deux contraintes :

- le coût de réalisation du plan d'actions solution x retenu ne doit pas dépasser le budget alloué au projet, désigné dans ce qui suit par β ;

$$W(x) \leq \beta \quad (2.3)$$

- le décideur peut définir un seuil minimal à atteindre pour chacun des objectifs. Par conséquent, une contrainte c_j représentant ce seuil est ajoutée pour chaque objectif j .

$$f_j(x) \geq c_j > 0 \quad (2.4)$$

2.3.5 Formulation du POPAMO

Avec la formulation de tous les éléments composant le POPAMO donnée précédemment, nous pouvons définir ce problème comme suit :

$$(POPAMO) \begin{cases} \max & k_j f_j(x) = k_j \sum_{i=1}^n f_{ij} a_i \quad j = 1, \dots, m \\ \text{s.t.} & \sum_{i=1}^n \omega_i a_i \leq \beta \\ & a_i \in \{0, 1\} \quad i = 1, \dots, n \end{cases} \quad (2.5)$$

Ainsi, l'objectif d'optimisation vise à trouver une solution (donc un plan d'actions) $x^* \in \arg \max_{x \in X} K.F(x)$ vérifiant :

$$\begin{cases} x^* \in \{0, 1\}^n \\ \forall_j \in \{1, m\}, f_j(x^*) \geq c_j \\ \sum_{i=1}^n a_i \omega_i \leq \beta \end{cases} \quad (2.6)$$

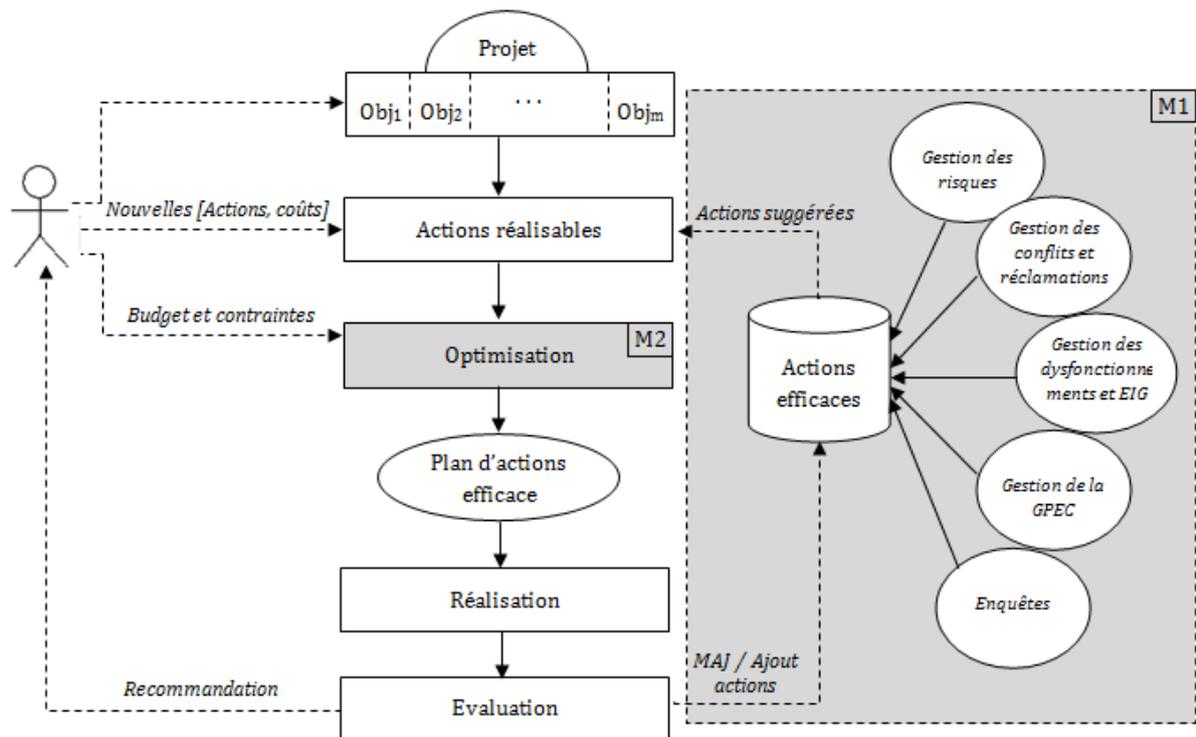
Cependant, vu que nous traitons un problème multi-objectif, x^* n'est quasiment jamais une solution unique. Au lieu de cela, nous obtenons un ensemble de solutions non dominées (au sens de la dominance de Pareto). Le but est donc de se rapprocher le plus possible du front Pareto réel. Dans ce qui suit, l'ensemble des approximations possibles est désigné par le symbole Ω .

2.4 MSQualité pour le POPAMO

Pour élaborer un plan d'actions optimal, le décideur doit d'abord réfléchir aux actions qui pourraient être réalisées dans son établissement et surtout prévoir l'impact qu'aura chaque action sur les objectifs qu'il s'est fixés. Puis, dans un deuxième temps, choisir parmi ces actions ainsi définies celles qui permettent d'atteindre au mieux les objectifs, dans la limite du budget. Cependant, comme cela a été précisé dans la section 2.2, le décideur ne dispose pas toujours de données permettant de prévoir l'impact qu'aurait une action quelconque avant que celle-ci ne soit réalisée. De plus, même en connaissant les impacts des actions sur les objectifs, choisir un sous-ensemble de ces actions est en soit un problème NP-difficile. Le résoudre à la main est tout simplement impossible, d'autant plus que souvent la taille de l'espace de recherche dépasse cinq cents actions pour pas moins de six objectifs à optimiser (dans le cadre pratique du POPAMO présenté ici).

Dans ce travail, nous proposons une solution qui vise à intégrer dans le logiciel *MSQualité* deux modules permettant d'assister le décideur à résoudre ces deux problématiques. Un module qui, en fonction du type de la structure et des besoins des personnes qui y sont accueillies, propose au décideur des actions qui ont déjà été réalisées dans des structures similaires et ayant produit des résultats probants. Et un autre module d'optimisation pour le choix des meilleures actions et la génération de plans d'actions optimaux. La figure 2.3 montre l'intégration des deux modules posés dans le processus d'élaboration des plans d'actions (partie grise). Avant de décrire ces deux modules, disons quelques mots sur le logiciel *MSQualité* lui-même.

Initialement, le logiciel *MSQualité* a été développé dans le cadre de la loi du 2 janvier 2002 pour mettre à disposition des directeurs des établissements un outil leur permettant de réaliser les évaluations internes et gérer les plans d'actions qui en découlent. Le logiciel a été conçu en

FIGURE 2.3 – Processus d’élaboration et d’optimisation des plans d’actions avec *MSQualité*

suivant les recommandations de ladite loi, permettant de construire des référentiels, de réaliser les évaluations et de générer automatiquement les rapports et les plans d’actions correspondants. Puis, plusieurs versions se sont succédées incluant divers autres modules, toujours dans le but de permettre aux établissements d’être conformes aux obligations légales. Nous pouvons citer, à titre d’exemple, un module pour la gestion des risques avec l’élaboration de plans de préventions des risques et l’édition du document unique d’évaluation des risques professionnels (obligatoire pour toutes les organisations, même en dehors du **SSMS**), un module pour l’enregistrement et le suivi des événements indésirables, un module pour la gestion des plans de formation, un module pour la gestion de projets, etc. Actuellement, *MSQualité* dispose d’une trentaine de modules déployés sur plus de trois cents structures.

2.4.1 Aide à la définition des actions

Quel que soit le type de l’établissement et les besoins des populations qu’il reçoit, le responsable est amené à réaliser d’une manière continue différents plans d’actions visant à améliorer les divers aspects de la vie de l’établissement. En effet, nous trouvons dans toute structure un plan d’actions quinquennal résultant de l’évaluation interne, un plan d’actions septennal issu de l’évaluation externe, un plan d’actions annuel pour la prévention des risques, un plan d’actions mis en œuvre dans le cadre du **CPOM**, un plan d’actions mis en place pour la bien-traitance et des plans d’actions décidés tout au long de l’année. A chaque fois qu’une action est réalisée, ses résultats sont évalués, soit en utilisant des indicateurs, soit avec des enquêtes ou d’autres moyens d’évaluation, selon la nature de l’action. Cependant, quels que soient les résultats de l’action, qu’ils soient bénéfiques ou néfastes, ils ne sont connus que par la structure qui l’a mise

en place, car même au sein d'une même association, il y a généralement peu d'échanges sur les expériences des uns et des autres. Par conséquent, il peut y avoir plusieurs établissements qui vont intégrer dans leurs plans d'actions des actions avec des rapports qualité/coût faibles, ce qui pourrait causer une perte d'argent, de temps et d'efforts inutiles. Ou au contraire, il peut y avoir des actions qui sont réalisées dans certaines structures ayant un bon rapport qualité/coût auxquelles les autres établissements n'ont pas pensé.

Pour compenser ce manque d'échange entre les établissements d'un côté, et surtout faire bénéficier les directeurs des expériences passées d'autres structures dans le domaine de la gestion de la qualité, d'un autre côté, nous avons intégré dans *MSQualité* un module composé d'un entrepôt d'actions récupérées des plans d'actions déjà réalisés dans différentes structures (figure 2.3, encadré "M1"). En plus de l'action, nous récupérons également d'autres informations telles que les impacts positifs et les impacts négatifs de l'action, son coût de réalisation, le type de structure où elle a été mise en œuvre, etc. En fonction des objectifs que le décideur définit pour son plan d'actions, le logiciel propose des actions dont l'efficacité sur au moins un des objectifs a déjà été constatée dans d'autres structures similaires. Le décideur est libre de choisir celles qui lui conviennent et de les compléter avec de nouvelles actions. De plus, à chaque évaluation des résultats d'un plan d'actions dans une structure, de nouvelles actions peuvent être intégrées dans l'entrepôt et celles qui y sont déjà peuvent être mises à jour.

2.4.2 Aide aux choix des actions

Le deuxième problème auquel le décideur est confronté est le choix des meilleures actions (en termes d'impacts) quand le budget ne permet pas de toutes les réaliser. Comme nous l'avons précisé dans le chapitre précédent, les besoins et l'obligation de performance ne cessent d'augmenter, pourtant les budgets sont de plus en plus réduits. De ce fait, le décideur se doit de faire le plus possible avec les moyens dont il dispose. Autrement dit, il faut que son plan d'actions soit le plus efficace possible. Pour l'assister dans cette tâche, encore plus difficile que la première, nous avons intégré dans *MSQualité* un deuxième module dédié à la génération de plans d'actions optimaux (figure 2.3, encadré "M2"). Une fois que les actions bénéfiques ou potentiellement bénéfiques sont connues, le défi consiste à résoudre le problème d'optimisation de plans d'actions multi-objectifs (POPAMO) décrit dans la section 2.3. Pour ce faire, plusieurs algorithmes sont proposés dans les chapitres 3 et 4.

2.5 État de l'art

2.5.1 Problème d'optimisation de plans d'actions

A notre connaissance, ce travail est la première étude menée sur le problème d'optimisation de plans d'actions multi-objectifs, tel qu'il a été décrit dans les sections 2.2 et 2.3. Il existe tout de même un projet mené par *THALES Research & Technology* sous le nom de "*Projet Descartes*" sur l'optimisation de la mise en œuvre des plans d'actions de secours pour la gestion de crises [Florence and Savéant, 2010]. Dans ce projet, le but n'était pas de sélectionner un sous-ensemble parmi un ensemble d'actions réalisables, mais plutôt de définir un outil d'ordonnement des opérations qui optimise la mise en œuvre des plans de secours et qui facilite leur adaptation à des situations évolutives. Cependant, même si le problème d'optimisation de plans d'actions multi-objectifs n'était pas étudié auparavant, le problème du sac-à-dos multi-objectif

l'est depuis plusieurs années. Plusieurs méthodes exactes ou d'approximation ont été proposées dans la littérature, dont les principales sont fournies dans la section ci-dessous.

2.5.2 Problème du sac-à-dos multi-objectif

Le problème de sac-à-dos mono-objectif (KP) est l'un des problèmes NP-complet les plus connus et étudiés dans la littérature [Salkin and De Kluyver, 1975; Martello and Toth, 1990; Kellerer *et al.*, 2004]. Sa définition de base consiste à sélectionner une partie d'un ensemble de n éléments (ou objets) ayant une caractéristique donnée ω_i ($i = 1, \dots, n$), telle que le poids, le volume, le coût, etc. ainsi qu'un profit p_i , dans le but de maximiser le bénéfice total du sac-à-dos sans dépasser sa capacité W . Plus formellement :

$$(KP) \begin{cases} \text{“max”} \sum_{i=1}^n x_i p_i \\ \sum_{i=1}^n x_i \omega_i \leq W \\ x_i \in \{0, 1\} \quad i = 1, \dots, n \end{cases} \quad (2.7)$$

où $x_i = 1$ si l'élément i est mis dans le sac, $x_i = 0$ sinon.

A partir de cette formulation, plusieurs variantes du KP sont proposées et étudiées :

- le problème du sac-à-dos multi-dimensionnel (MKP) [Jalali Varnamkhasti, 2012; Kellerer *et al.*, 2004] où le sac-à-dos a c capacités ($c > 1$) et chaque élément possède c caractéristiques ω_i^k ($k = 1, \dots, c$). La solution au problème doit respecter toutes les capacités du sac-à-dos.
- le problème du sac-à-dos multi-objectif (MOKP) [Ben Abdelaziz *et al.*, 1999; Kumar and Banerjee, 2006; Bazgan *et al.*, 2009a] où les éléments génèrent plusieurs profits p_{ij} ($j = 1, \dots, m$) qui doivent tous être maximisés.
- le problème du sac-à-dos multiple [Martello and Toth, 1990; Kellerer *et al.*, 2004], où plusieurs sacs-à-dos, avec différentes capacités, doivent être remplis et optimisés en même temps. En attribuant chaque élément à un seul et unique sac, l'objectif consiste à maximiser le profit total de tous les sacs sans violer leurs contraintes.
- le problème du sac-à-dos à choix multiple (MCKP) [Sinha and Andris A, 1979; Kellerer *et al.*, 2004] où le principe consiste à diviser l'ensemble des éléments en plusieurs classes et à choisir un élément de chaque classe, de telle sorte que le profit total des éléments sélectionnés soit maximisé et la contrainte du sac-à-dos ne soit pas violée.
- le problème du sac-à-dos quadratique (QKP) [Gallo *et al.*, 1980; Pisinger, 2007; Kellerer *et al.*, 2004], dans cette variante chaque élément i possède un poids ω_i et un profit p_i qui est le bénéfice réalisé si l'élément i est sélectionné et un profit P_{ij} dans le cas où l'élément i et l'élément j sont tous les deux sélectionnés. L'objectif étant toujours de maximiser la somme des profits des éléments inclus dans le sac sans que la somme de leurs poids ne dépasse la capacité du sac-à-dos.

Depuis, de nombreux travaux ont traité des problèmes académiques et des problèmes pratiques qui combinent deux ou plusieurs de ces variantes, comme le problème du sac-à-dos multi-objectif multi-dimensionnel (MOMKP) [Barichard and Hao, 2003; Lust and Teghem, 2012;

[Vianna and Vianna, 2013] et le problème du sac-à-dos multi-dimensionnel à choix multiple [Iqbal *et al.*, 2010; Shojaei *et al.*, 2013; Cherfi and Hifi, 2009; Chen and Hao, 2014]. Wilbaut *et al.* [Wilbaut *et al.*, 2008] ont détaillé, dans leur papier "A survey of effective heuristics and their application to a variety of knapsack problems", toutes les variantes du problème du sac-à-dos avec des exemples d'applications réelles sur chacune d'elle.

La version multi-objectif est l'une des variantes les plus difficiles du problème du sac-à-dos du fait que de nombreux objectifs conflictuels doivent être maximisés en même temps. Le MOKP peut être utilisé pour formuler de nombreux problèmes pratiques tels que l'allocation de ressources, les investissements, la gestion de projets et les plans d'actions, d'où l'intérêt de l'étudier. En effet, depuis les années 90, beaucoup d'articles ont été publiés sur le problème, mais la majeure partie de ces papiers ne traite que des instances à deux ou trois objectifs, rarement plus. Or, dans les cas pratiques, on est souvent confronté à des problèmes avec plus d'objectifs à optimiser. Nous listons ci-dessous les principaux travaux de la littérature, une partie de cette liste est tirée de l'état de l'art réalisé par Lust et Teghem en 2012 [Lust and Teghem, 2012].

1. Méthodes exactes

Les méthodes exactes traitant le MOKP peuvent être classées en quatre classes :

- *méthode à deux phases* : proposée initialement par Ulungu et Teghem [Ulungu and Jacques, 1995] pour résoudre des problèmes combinatoires bi-objectifs. La première phase génère les solutions dites "*supportées*" en résolvant une série de problèmes mono-objectifs en utilisant plusieurs agrégations linéaires des fonctions bi-objectifs. La deuxième phase, recherche les solutions dites "*non supportées*" qui se trouvent dans les triangles rectangles définis par deux points supportés consécutifs de l'espace objectif. En 1997 [Ulungu and Teghem, 1997], ils ont appliqué cette approche pour résoudre le problème du sac-à-dos bi-objectif en utilisant la méthode "*branch and bound*" dans la deuxième phase. Cette méthode a été d'abord améliorée par Visée *et al.* en 1998 [Visée *et al.*, 1998] puis par Jorge en 2010 [Jorge, 2010].
- *transformation en problèmes bi-objectifs du plus court chemin* : cette approche utilise une version adaptée de l'algorithme d'étiquetage de Martins et Dos Santos [Martins and Santos, 1999]. Initialement, elle a été proposée par Captivo *et al.* en 2003 [Captivo *et al.*, 2003] puis étendue par Figuera *et al.* en 2006 [Figuera *et al.*, 2006].
- *programmation dynamique* : l'utilisation de la programmation dynamique pour résoudre le MOKP a été suggérée par Klamroth et Wiecek en 2000 [Klamroth and Wiecek, 2000], puis en 2009, Bazgan *et al.* [Bazgan *et al.*, 2009a] ont développé cette idée en ajoutant plusieurs relations de dominance complémentaires pour éliminer les solutions partielles.
- *hybridation* : en 2010, Delort et Spanjaard [Delort and Spanjaard, 2010] ont proposé une hybridation entre la méthode à deux phases et une procédure de programmation dynamique pour résoudre des instances bi-objectifs du problème du sac-à-dos et ils ont obtenu de meilleurs résultats par rapport à ceux obtenus par Bazgan *et al.* [Bazgan *et al.*, 2009a].

2. Algorithmes d'approximation

Nous pouvons répertorier quatre papiers proposant des algorithmes d'approximation :

- en 2002, Erlebach et al. ont présenté dans [Erlebach *et al.*, 2002] "A fully polynomial-time approximation scheme algorithm" appliqué au MOKP et "A polynomial-time approximation scheme problem" appliqué au MOMKP.
- en 2006, Kumar et Banerjee ont proposé un algorithme évolutionnaire restreint appelé REMO (*Restricted Evolutionary Multiobjective Optimizer*) [Kumar and Banerjee, 2006], basé sur un pool de reproduction restreint et une archive séparée pour stocker la population restante.
- en 2008, Gomes da Silva et al. [Gomes da Silva *et al.*, 2008] ont appliqué, pour la première fois, le concept de "cœur du problème" (*core problems*) sur un problème de sac-à-dos bi-objectif. Le concept de "cœur du problème" consiste à résoudre le problème de sac-à-dos ordinaire mais en considérant uniquement un sous-ensemble des items, où la probabilité de trouver une solution optimale est élevée. Puis, la solution optimale trouvée pour le nouveau problème ainsi réduit est considérée comme une borne inférieure à respecter dans la résolution du problème principal. Gomes da Silva et al. ont montré que, en moyenne, le nombre d'items utilisés pour le problème réduit correspond à un très petit pourcentage du nombre total d'items.
- en 2009, suivant la même idée que celle qu'ils ont développée dans [Bazgan *et al.*, 2009a], Bazgan et al. ont proposé "A fully polynomial-time approximation scheme algorithm" et les résultats qu'ils ont obtenus sont bien meilleurs que ceux obtenus par Erlebach et al. [Erlebach *et al.*, 2002].

3. Méthodes heuristiques

Plusieurs algorithmes ont été proposés pour approximer les fronts Pareto induits par les instances de MOKP :

- approches basées sur le recuit simulé : nous pouvons en citer l'algorithme "Pareto simulated annealing" (PSA) [Czyzjak and Jaskiewicz, 1998] proposé par Czyzjak et Jaskiewicz en 1998 et la méthode "multi-objective simulated annealing" (MOSA) [Ulungu and Teghem, 1999] proposée par Ulungu et al. en 1999. La première est utilisée pour résoudre des instances aléatoires bi-objectifs avec 500 items, la seconde pour résoudre des instances également aléatoires allant jusqu'à 4 objectifs et 800 items.
- approches basées sur la recherche Tabu comme l'algorithme "TS-based method" [Ben Abdelaziz and Krichen, 1997] proposé par Ben Abdelaziz et Krichen en 1997 puis étendu en 1999 pour intégrer la recherche Tabu dans un algorithme génétique [Ben Abdelaziz *et al.*, 1999] et la méthode "multi-objective Tabu search" (MOTS) [Gandibleux and Freville, 2000] proposée par Gandibleux et Freville en 2000, où un accroissement de la fonction de scalarisation Tchebycheff pondérée est utilisé pour sélectionner le meilleur voisin.
- en 2010, Lust et Teghem ont proposé une adaptation de la méthode à deux phases [Lust and Teghem, 2010] pour résoudre, dans la deuxième phase de la méthode, le MOMKP couplé avec VLSN (*Very Large Scale Neighborhood*). La première phase de 2PPLS génère une bonne approximation des solutions efficaces en utilisant des sommes pondérées et des solveurs mono-objectifs. En utilisant la méthode PLS [Paquete *et al.*, 2004], la deuxième phase améliore l'approximation en générant des solutions efficaces non supportées. Les auteurs ont montré que sur des cas bi-objectifs,

2PPLS combinée avec VLSN donne de meilleurs résultats que d'autres méthodes de la littérature, mais sur des problèmes avec trois objectifs, le temps de convergence de 2PPLS est trop élevé.

- en 2006, Gomes da Silva et al. [Gomes da Silva et al., 2006] ont proposé une méthode de recherche de dispersion pour résoudre des instances aléatoires du problème de sac-à-dos bi-objectif, ayant jusqu'à 6 000 items. En 2007, ils ont proposé une nouvelle approche hybride [Gomes da Silva et al., 2007] qui combine une recherche exacte avec une autre heuristique afin de réduire l'imprécision observée dans la méthode de recherche de dispersion.
- pour réduire la taille de l'espace de recherche d'un problème du sac-à-dos bi-objectif, Gandibleux et Klamroth [Gandibleux and Klamroth, 2006] ont étudié les cardinalités des bornes du problème en utilisant des scalarisations de sommes pondérées. En 2007, Ehrgott et Gandibleux [Ehrgott and Gandibleux, 2007] ont généralisé la notion de valeur bornée à celle d'un ensemble borné. En appliquant ces concepts au MOKP, ils ont obtenu des ensembles bornés supérieurs en utilisant la relaxation linéaire du MOKP et des ensembles bornés inférieurs en utilisant un algorithme glouton simple.
- en 2013, Vianna et Vianna [Vianna and Vianna, 2013] ont développé les algorithmes "Multiobjective-GRASP" (MGRASP) et "MultiObjective-ILS" (MILS) pour générer une bonne approximation de l'ensemble Pareto optimal du MOKP. Dans leurs expérimentations, ils ont comparé leurs méthodes avec les algorithmes MOTGA [Alves and Almeida, 2007], MOGLS [Jaszkiwicz, 2002] et SPEAII [Zitzler et al., 2012], sur des instances avec 250, 500 et 750 éléments, et de 2 à 4 objectifs. L'algorithme MGRASP génère plus de solutions non dominées que MOTGA, MOGLS et SPEAII, dans 7 instances sur un total de 9 et l'algorithme MILS génère plus de solutions non dominées que MOTGA, MOGLS et SPEAII pour toutes les instances considérées. Les auteurs ont également noté que les solutions obtenues par les algorithmes MGRASP et MILS sont mieux distribuées que celles obtenues par les autres algorithmes et lorsqu'ils ont comparé MILS à MGRASP, MILS converge mieux que MGRASP.
- dans [Figueira et al., 2013], Figueira et al. ont proposé trois variantes de l'algorithme DP de Bazgan et al. [Bazgan et al., 2009b] pour résoudre le problème du sac-à-dos bi-objectif. En proposant de nouvelles définitions des bornes inférieures, des bornes supérieures et des ensembles bornés d'un côté, et en utilisant la recherche dichotomique et des algorithmes simplexes bi-objectifs, d'un autre côté, les variantes proposées ont considérablement amélioré les résultats obtenus jusqu'alors avec des algorithmes basés sur la programmation dynamique de la littérature (en termes de mémoire utilisée et temps CPU). Sur les grandes instances, ils ont réduit d'environ 20% le temps de calcul et de 15% la mémoire utilisée par rapport à l'état de l'art.
- Aisopos et al. [Aisopos et al., 2013] ont utilisé le MOKP pour modéliser un problème pratique d'allocation de ressources dans un environnement SaaS, optimisant à la fois l'utilisation des ressources, la qualité de service et le profit du fournisseur.
- en 2014, Vianna et al. [Vianna et al., 2014] ont proposé un nouvel algorithme, appelé "Memory-based Multiobjective GRASP" (MMGRASP). Dans un algorithme

GRASP traditionnel, chaque itération est indépendante, c'est-à-dire lorsqu'une itération se termine, les informations trouvées ne sont pas utilisées dans les itérations suivantes. Dans MMGRASP, l'itération $k + 1$ utilise, comme point de départ, la meilleure solution trouvée pendant l'itération k .

- dans [Ishibuchi *et al.*, 2014], Ishibuchi et al. ont étudié la performance des algorithmes évolutionnaires quand on utilise des parents totalement différents ou n'échangeant qu'un petit nombre de gènes entre deux parents (croisement avec une très petite probabilité d'échange de gènes). Pour examiner ces deux schémas, ils ont appliqué NSGA-II avec différents paramètres de croisement à un MOKP avec 500 items et de 2 à 10 objectifs, puis ils ont mesuré la distance parent/parent et la distance parent/enfants dans chaque cas. Les résultats expérimentaux montrent que NSGA-II est plus efficace et la maintenance de la diversité est meilleure lorsque la distance parent/enfants est faible.
- en 2015, Cerqueus [Cerqueus, 2015], a proposé une nouvelle méthode "*branch-and-bound*" pour un problème qui teste plusieurs stratégies à chaque séparation et choisit la meilleure selon une mesure de qualité préalablement définie. La méthode réduit considérablement les arbres de recherche résultants, mais son coût de calcul est supérieur à celui obtenu par une stratégie statique. En utilisant des méthodes d'apprentissage renforcées pour sélectionner dynamiquement une stratégie de ramification, elle a pu réduire le temps de calcul sur 58% des instances, par rapport à une stratégie statique. Elle a également introduit deux algorithmes exacts et une version heuristique pour calculer la borne supérieure optimale pour résoudre des problèmes de sac-à-dos bi-objectifs bi-dimensionnels.
- Ben Mansour et Alaya [Ben Mansour and Alaya, 2015], ont proposé l'algorithme "*indicator-based ant colony optimization algorithm*" (IBACO) pour le MOKP. Dans leurs expérimentations, elles ont utilisé les indicateurs ϵ et HV (hyper-volume) pour comparer IBACO avec les algorithmes IBEA, SPEA2 et m-ACO sur des instances avec 250, 500 et 750 éléments et de 2 à 4 objectifs. Les résultats expérimentaux montrent que IBACO surpasse significativement les autres algorithmes considérés.
- en 2017, Ben Mansour et al. [Ben Mansour *et al.*, 2017] ont proposé une approche de résolution basée sur un algorithme de recherche locale Min-Max Tchebycheff itéré (Min-Max TLS). Pour explorer les régions les plus prometteuses de l'espace de recherche, elles ont proposé une nouvelle structure de voisinage appelée *Min-Max $\mathcal{N}(s)$* qui combine deux algorithmes "*Min-Extraction-Item*" et "*Max-Insertion-Item*". Suivant la qualité des items, le premier algorithme (c'est-à-dire "*Min-Extraction-Item*") les classe dans une liste l_1 dans l'ordre ascendant et le deuxième les classe dans une autre liste l_2 dans l'ordre descendant. Pour générer un nouveau voisin, la procédure *Min-Max $\mathcal{N}(s)$* supprime d'abord un item de la liste l_1 puis sélectionne les premiers items de la liste l_2 jusqu'à violation d'une des contraintes, après recalcule à nouveau le contenu de l_2 , supprime un autre item de l_1 et sélectionne les premiers items de l_2 , ainsi de suite. Comparée aux algorithmes MOMAD, 2PPLS et MOTGA sur le MOMKP, leur méthode a obtenu les meilleurs résultats sur toutes les instances considérées.
- Zouache et al. [Zouache *et al.*, 2017] ont proposé l'hybridation de deux méthodes à intelligence d'essaim, à savoir l'algorithme de luciole et l'optimisation des essaims

de particules dans un algorithme appelé "*Multi-objective Firefly algorithm with Particle swarm optimization*" (MO-FPA). Ils ont testé leur méthode sur une douzaine d'instances du MOKP avec un nombre d'items compris entre 250 et 750 et de 2 à 4 objectifs. L'algorithme MOFPA a été comparé à trois autres algorithmes évolutionnaires : NSGA-II, SPEA-II et MOEA/D. Leurs expérimentations ont montré que l'algorithme proposé a fourni des solutions qui dominent la plupart des solutions générées par les autres algorithmes.

2.6 Conclusion

Nous avons présenté dans ce chapitre le processus d'élaboration des plans d'actions dans les établissements du SSMS et la nature des objectifs, des actions et des contraintes qui le définissent. Nous avons également montré que, dans le but d'une amélioration continue de la gestion de l'établissement et de la qualité de service rendu aux résidents, l'élaboration de plans d'actions optimaux est un problème récurrent qui revient tous les cinq ans. De plus, dans un contexte de restriction budgétaire continue, le décideur se trouve dans l'obligation de faire des choix quant aux actions qu'il peut réaliser, parce que les moyens financiers dont il dispose ne lui permettent pas de toutes les réaliser. Par conséquent, il doit s'assurer que, même avec un budget limité, le plan d'actions qu'il va mettre en place permettrait d'améliorer au maximum tous les aspects de l'établissement et de la vie de ses résidents.

Dans la troisième section du chapitre, nous avons présenté notre première contribution de cette thèse, à savoir une modélisation formelle du problème d'optimisation de plans d'actions sous forme d'un problème de sac-à-dos multi-objectif où le plan d'actions représente le sac-à-dos, les actions représentent les items, les poids et les profits des items sont respectivement les coûts de réalisation des actions et les impacts qu'elles ont sur les objectifs. Puis, nous avons décrit, dans la quatrième section, la solution que nous proposons d'intégrer dans le logiciel *MSQualité* pour assister le décideur à la fois dans la définition des actions efficaces et le choix de celles qui seront réellement réalisées, dans la limite de son budget et le respect de ses contraintes. Nous avons terminé le chapitre par un état de l'art sur le problème du sac-à-dos multi-objectif et le problème d'optimisation de plans d'actions où un seul article, d'ailleurs n'étant que relativement proche, a été publié sur l'ordonnancement des actions.

Dans les chapitres suivants, nous décrivons en détail les solutions que nous proposons pour la génération de plans d'actions optimaux. Dans le chapitre 3, nous proposons l'algorithme *R2-IBMOLS* qui combine les performances de l'algorithme *IBMOLS* [Basseur and Burke, 2007] avec l'indicateur *R2* [Brockhoff et al., 2012]. Pour réduire le nombre de solutions non dominées générées et le temps de calcul constatés avec *R2-IBMOLS*, nous proposons, dans le chapitre 4, de remplacer la dominance Pareto par la dominance de Lorenz dans trois algorithmes différents.

Optimisation multi-objectif à base d'indicateurs de qualité

Après avoir modélisé le POPAMO sous forme d'un MOKP dans le chapitre 2, nous présentons dans ce chapitre une méthode de résolution qui combine la recherche locale itérée avec l'utilisation de l'indicateur de qualité $R2$ dans le calcul de la valeur de fitness des solutions. L'indicateur $R2$ est à la fois un outil efficace pour évaluer la qualité d'une solution et un moyen qui permet d'intégrer les préférences du décideur dans le processus d'optimisation. Avant de décrire ladite méthode, nous présentons d'abord le principe de l'optimisation à base d'indicateurs de qualité suivi d'un état de l'art des méthodes permettant d'intégrer les préférences du décideur en général et celles basées sur les indicateurs de qualité en particulier. Nous terminons le chapitre par quelques résultats expérimentaux sur des instances ayant de 2 à 8 objectifs et jusqu'à 500 actions ainsi que quelques perspectives. Le travail présenté dans ce chapitre a fait l'objet de deux articles : un article de conférence [Chabane *et al.*, 2015] (papier court) et un article de revue [Chabane *et al.*, 2017].

Sommaire

3.1	Introduction	54
3.2	Indicateurs de qualité	55
3.2.1	Définition	55
3.2.2	Quelques indicateurs	55
3.3	État de l'art	58
3.3.1	Intégration des préférences du décideur	58
3.3.2	Méthodes à base d'indicateurs de qualité	61
3.4	Approche proposée pour résoudre le POPAMO	63
3.4.1	Motivations	63
3.4.2	$R2$ -IBMOLS	64
3.4.3	Expérimentations	66

3.4.4 Résultats expérimentaux	70
3.5 Conclusion	74

3.1 Introduction

Résoudre des problèmes d'optimisation multi-objectif n'est pas toujours une tâche facile, particulièrement quand nous avons à faire à un problème de plus de quatre objectifs. La tâche est encore plus difficile (mais à mon avis plus intéressante) quand nous traitons un problème réel où, comme nous l'avons vu dans le chapitre 2, il faut prendre en compte les préférences du décideur et plusieurs de ses contraintes pratiques.

Une des techniques qui a montré son efficacité sur cette catégorie de problèmes est celle qui consiste à intégrer, dans des algorithmes évolutionnaires, génétiques ou de recherche locale, des indicateurs de qualité qui permettent d'évaluer la qualité de chaque individu de la population ou de chaque voisin d'une solution donnée et d'en choisir la (les) solution(s) ayant la meilleure qualité par rapport à l'indicateur considéré. En effet, depuis la publication de l'algorithme *Indicator-based Selection in Multiobjective Search* (IBEA) [Zitzler and Künzli, 2004] en 2004, beaucoup de chercheurs s'intéressent de plus en plus à cette technique et plusieurs travaux ont été publiés.

Dans ce chapitre, nous proposons de combiner l'algorithme *Indicator-based Multiobjective Local Search* (IBMOLS) [Basseur and Burke, 2007] avec l'indicateur Epsilon (I_ϵ) [Zitzler and Künzli, 2004] et l'indicateur $R2$ (I_{R2}) [Hansen and Jaszkiewicz, 1998; Brockhoff et al., 2012] pour résoudre le POPAMO. Le but est de montrer que l'association de l'algorithme IBMOLS avec l'indicateur $R2$ permet d'obtenir des solutions non dominées de très bonne qualité même sur des problèmes de grande taille. En effet, même si l'algorithme IBMOLS a déjà été testé sur des problèmes bi-objectifs avec l'indicateur Epsilon et l'indicateur hyper-volume (I_{HV}) [Basseur et al., 2012a], c'est la première fois qu'il est associé avec l'indicateur $R2$ pour résoudre un problème ayant plus de trois objectifs. Notre recours à l'utilisation de l'indicateur $R2$ a été motivé, d'un côté, par une étude préliminaire [Chabane et al., 2015] que nous avons réalisée sur l'indicateur Epsilon qui a montré que cet indicateur ne maintient pas toujours la diversité de la population dans l'espace objectif. D'un autre côté, par le fait que le temps de calcul de l'indicateur HV augmente d'une manière exponentielle suivant le nombre d'objectifs [Beume et al., 2009], ce qui le rend inapproprié à la résolution de notre problème. Pour résoudre le POPAMO, nous proposons donc l'algorithme $R2$ -IBMOLS qui a prouvé, par les expérimentations que nous avons réalisées sur une trentaine d'instances de différentes tailles, qu'il est très efficace comparé à ϵ -IBMOLS et NSGA-II [Deb et al., 2002], sur toutes les instances utilisées.

Ce chapitre est organisé de la façon suivante : dans la section 3.2, nous présentons le principe d'indicateurs de qualité pour l'optimisation multi-objectif et nous fournissons les définitions des indicateurs Epsilon et $R2$ utilisés dans l'approche proposée dans ce chapitre. Dans la première partie de la section 3.3, nous présentons un recueil des méthodes permettant d'intégrer les préférences du décideur. Puis, dans la deuxième partie, nous présentons celles qui utilisent les indicateurs de qualité. Dans la section 3.4, nous décrivons la solution que nous proposons et dans les sections 3.4.3, nous introduisons les instances du problème ainsi que le protocole expérimental utilisé. Enfin, dans la section 3.4.4, nous fournissons nos résultats expérimentaux et une étude comparative avec NSGA-II.

3.2 Indicateurs de qualité

3.2.1 Définition

Les indicateurs binaires de qualité [Zitzler *et al.*, 2003] représentent une extension naturelle de la relation de dominance Pareto sur l'ensemble des vecteurs objectifs, où une fonction $I : \Omega \times \Omega \rightarrow \mathbb{R}$ quantifie la différence de qualité entre deux approximations A et B . Si R est un ensemble de référence de solutions non dominées qui est fixe (par exemple, l'ensemble des solutions Pareto optimales), la fonction I devient un indicateur de qualité unaire qui attribue à chaque ensemble de vecteurs objectifs A un nombre réel représentant sa "distance" de la référence R . Résoudre le problème d'optimisation revient à minimiser cette distance pour que l'approximation soit aussi proche que possible de la référence R . Avec ce principe, l'objectif d'optimisation est transformé en l'identification d'une approximation qui minimise la fonction I . En outre, l'indicateur de qualité peut être également utilisé pour comparer la qualité de deux solutions individuelles ou une solution unique par rapport au reste de la population. Cette comparaison pourrait être utilisée comme moyen de sélection dans le processus de sélection des algorithmes évolutionnaires. En effet, lors du processus de sélection, les solutions à supprimer de la population sont celles qui possèdent les plus mauvaises valeurs pour l'indicateur utilisé (ou choisir celles qui ont les meilleures valeurs pour l'indicateur). En d'autres termes, le but du processus de sélection consiste à éliminer les solutions qui induisent la plus faible dégradation de la qualité globale de la population, vis à vis de l'indicateur considéré. Cependant, comme cela a été défini dans [Zitzler and Künzli, 2004], l'indicateur binaire doit vérifier la propriété de préservation de la dominance (définition 14).

Définition 14. *L'indicateur binaire I vérifie la préservation de la dominance ssi :*

$$\forall x_1, x_2 \in X, x_1 \prec x_2 \Rightarrow I(x_1, x_2) < I(x_2, x_1) \text{ et}$$

$$\forall x_1, x_2, x_3 \in X, x_1 \prec x_2 \Rightarrow I(x_3, x_1) < I(x_3, x_2)$$

Il existe plusieurs indicateurs de qualité différents dans la littérature. Les plus utilisés sont l'indicateur Epsilon, l'indicateur HV et plus récemment l'indicateur $R2$. Par ailleurs, le décideur pourrait bien exprimer ses préférences sous forme d'un indicateur de qualité qui pourrait être utilisé aussi dans le calcul de la valeur de fitness des solutions et dans le processus de sélection des algorithmes.

3.2.2 Quelques indicateurs

Dans cette section, nous donnons les définitions des trois indicateurs les plus utilisés, à savoir I_ϵ , I_{HV} et I_{R2} . Ces définitions sont tirées en partie de [Zitzler and Künzli, 2004; Zitzler *et al.*, 2003] pour I_ϵ et I_{HV} et de [Brockhoff *et al.*, 2012; Hansen and Jaszkievicz, 1998] pour I_{R2} .

3.2.2.1 Indicateur Epsilon

$I_\epsilon(A, B)$ correspond à la distance minimale par laquelle l'approximation A doit être translatée dans chaque dimension de l'espace objectif de sorte que l'approximation B soit dominée par A . Plus formellement, l'indicateur Epsilon peut être défini comme suit :

$$I_\epsilon(A, B) = \min_\epsilon \{ \forall x_2 \in B, \exists x_1 \in A : f_j(x_1) - \epsilon \leq f_j(x_2) \} \quad (3.1)$$

$$\forall j \in \{1, \dots, m\}$$

Pour comparer la différence de qualité de deux solutions x_1 et x_2 , I_ϵ peut être utilisé comme suit :

$$I_\epsilon(x_1, x_2) = \max\{f_j(x_1) - f_j(x_2)\} \quad \forall j \in \{1, \dots, m\} \quad (3.2)$$

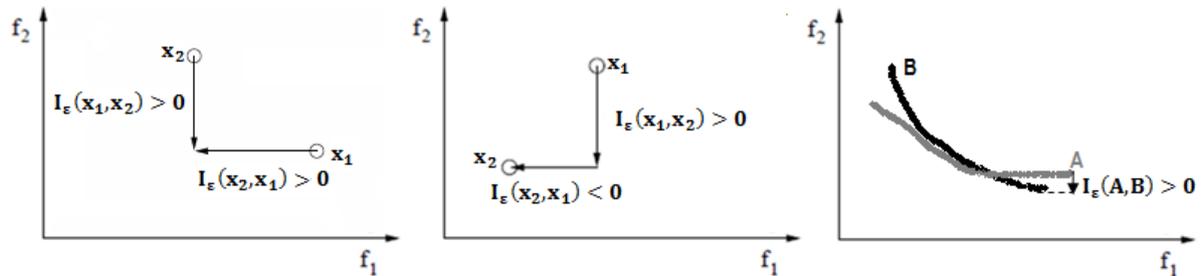


FIGURE 3.1 – Illustration de l'indicateur Epsilon appliqué à deux solutions non dominées x_1 et x_2 (gauche), $x_2 \prec x_1$ (centre) et à deux approximations A et B (droite).

3.2.2.2 Indicateur hyper-volume

$I_{HV}(x)$ correspond à l'hyper-volume dominé par la solution x dans l'espace objectif. $I_{HV}(x_1, x_2)$ désigne l'hyper-volume de l'espace objectif dominé par la solution x_2 et qui n'est pas dominé par la solution x_1 (figure 3.2). Cependant, pour calculer la valeur de I_{HV} , il faut définir un point de référence qui sert à délimiter l'espace objectif. Selon que les solutions x_1 et x_2 soient dominées ou pas, l'hyper-volume est calculé comme suit :

$$I_{HV}(x_1, x_2) = \begin{cases} I_{HV}(x_1) - I_{HV}(x_2) & \text{si } x_1 \prec x_2 \text{ ou } x_2 \prec x_1 \\ I_{HV}(x_1 + x_2) - I_{HV}(x_1) & \text{sinon} \end{cases} \quad (3.3)$$

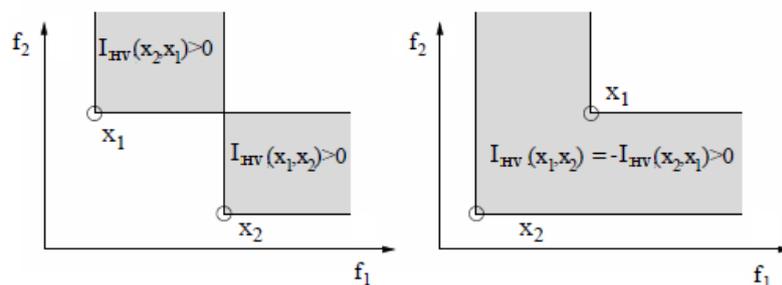


FIGURE 3.2 – Illustration de l'indicateur HV appliqué à deux solutions non dominées x_1 et x_2 (à gauche) et $x_2 \prec x_1$ (à droite) [Zitzler and Künzli, 2004].

3.2.2.3 Indicateur R2

L'indicateur $R2$ [Brockhoff *et al.*, 2012; Hansen and Jaszkiewicz, 1998] et l'indicateur HV sont les deux approches recommandées [Knowles *et al.*, 2006] qui évaluent simultanément tous les

aspects souhaités d'une approximation du front Pareto. Cependant, l'indicateur $R2$ est considéré comme une alternative à l'indicateur HV pour deux raisons : (i) le temps de calcul de la contribution d'une solution à l'hyper-volume global de la population augmente de façon exponentielle avec le nombre d'objectifs. (ii) les distributions obtenues avec l'indicateur HV se situent souvent autour des régions convexes du front Pareto [Brockhoff *et al.*, 2014].

Comme cela à été décrit par Hansen et Jaszkiwicz dans [Hansen and Jaszkiwicz, 1998], I_{R2} appartient à une famille d'indicateurs basée sur l'utilisation de fonctions d'utilité qui associent à chaque vecteur $z \in \mathbb{R}^m$ une valeur scalaire $v \in \mathbb{R}$ afin d'évaluer la qualité relative de deux approximations différentes du front Pareto. La définition 15 montre comment évaluer avec I_{R2} la différence de qualité entre une approximation A et un ensemble de référence R .

Définition 15. Pour un ensemble fini de fonctions d'utilité U , une distribution uniforme p sur U , et un ensemble de référence R , la valeur de l'indicateur $R2$ pour une approximation A est donnée par :

$$I_{R2}(R, A, U) = \frac{1}{|U|} \sum_{u \in U} \left(\max_{r \in R} \{u(r)\} - \max_{a \in A} \{u(a)\} \right) \quad (3.4)$$

Quand la référence R est constante, I_{R2} devient un indicateur unaire, défini comme suit :

$$I_{R2}(A, U) = -\frac{1}{|U|} \sum_{u \in U} \max_{a \in A} \{u(a)\} \quad (3.5)$$

Comme l'ont suggéré Hansen et Jaszkiwicz [Hansen and Jaszkiwicz, 1998] et utilisé par Brockhoff *et al.* dans [Brockhoff *et al.*, 2012] et [Brockhoff *et al.*, 2014], nous utilisons également, dans la suite de ce chapitre, la fonction Tchebycheff pondérée $u(z) = u_\lambda(z) = -\max_{j \in \{1, \dots, m\}} \lambda_j |z_j^* - z_j|$ avec l'indicateur $R2$ tel qu'il est défini dans l'équation 3.5, où $\lambda = (\lambda_1, \dots, \lambda_m) \in \Lambda$ est un vecteur de poids donné et z^* est un point utopique.

Dans [Brockhoff *et al.*, 2012; Brockhoff *et al.*, 2014], Brockhoff *et al.* ont étudié l'influence du nombre de vecteurs de poids ainsi que leur distribution sur la distribution optimale des solutions. Ils ont conclu que :

- pour les problèmes bi-objectifs, l'emplacement optimal d'un point par rapport à l'indicateur $R2$ dépend uniquement de ses voisins et seulement d'un sous-ensemble de vecteurs de poids ;
- pour un ensemble de solutions de taille μ , les μ -distributions¹ pour l'indicateur $R2$ tend à contenir les points d'intersection des rayons correspondant aux vecteurs de poids avec le front Pareto lorsque $\mu > |\Lambda|$. Il existe une seule μ -distribution optimale lorsque $\mu = |\Lambda|$ et la μ -distribution optimale n'est pas toujours unique dans le cas où $\mu < |\Lambda|$;
- même dans les scénarios dans lesquels la répartition optimale change avec l'augmentation du nombre de vecteurs de poids, cette distribution se stabilise dès qu'un nombre spécifique $|\Lambda|_{min}$ est dépassé. Pour le cas bi-objectif, $|\Lambda|_{min} = 10\mu$.

Les mêmes auteurs ont montré dans [Wagner *et al.*, 2013] que les μ -distributions optimales peuvent être affectées en changeant la position du point de référence ou en rétrécissant la distribution des vecteurs de poids dans l'espace objectif (figure 3.3). Ces deux mécanismes peuvent

¹Ensembles de solutions optimales de taille μ .

être utilisés pour orienter la recherche vers une zone de l'espace objectif qui intéresserait plus le décideur. De plus, au moyen du point de référence et des vecteurs de poids, l'indicateur $R2$ permet également d'intégrer directement dans le processus de recherche les préférences du décideur.

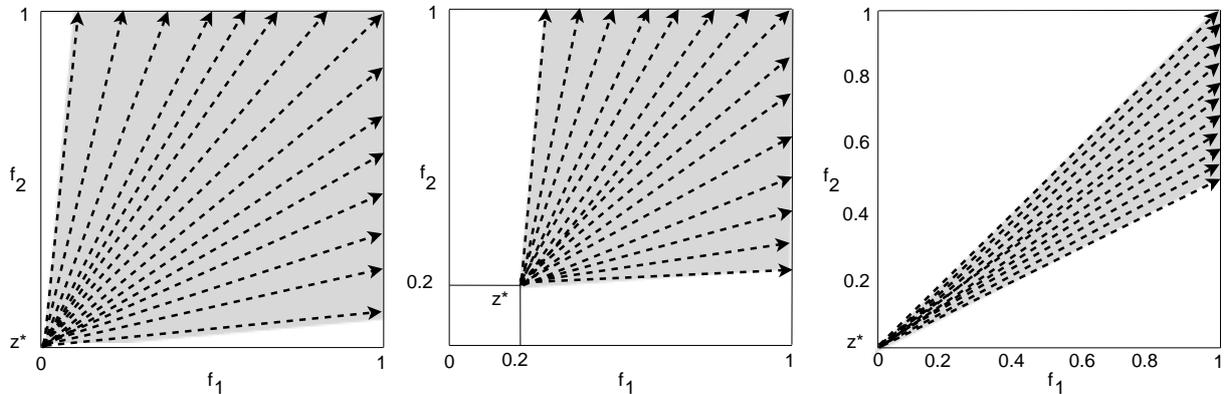


FIGURE 3.3 – Illustration de la réduction de l'espace objectif (espace gris) en changeant la position du point de référence, de $z^* = (0, 0)$ (gauche) à $z^* = (0.2, 0.2)$ (centre) ou en agissant sur la distribution des vecteurs de poids normalisés (droite). Les flèches pointillées correspondent aux directions définies par les vecteurs de poids.

3.3 État de l'art

Depuis la publication de l'algorithme IBEA [Zitzler and Künzli, 2004] en 2004, le recours à l'utilisation des indicateurs de qualité dans les algorithmes multi-objectifs est en augmentation continue pour deux raisons : (i) les algorithmes à base d'indicateurs de qualité ont montré leur efficacité sur différents types de problèmes pratiques et académiques. (ii) la définition de certains indicateurs (comme I_{R2} et I_{HV}) permet d'intégrer, complètement ou partiellement, les préférences du décideur dans le processus de recherche de l'algorithme, ce qui augmente considérablement la probabilité d'acceptation de la solution par le décideur.

Avant de présenter l'approche que nous proposons pour résoudre le POPAMO (section 3.4), nous présentons d'abord, dans cette section, quelques travaux représentatifs sur les méthodes permettant d'intégrer les préférences du décideur ainsi que celles basées sur l'utilisation d'indicateurs de qualité.

3.3.1 Intégration des préférences du décideur

Le but final des algorithmes d'optimisation est d'aider le décideur à choisir une solution finale qui correspond le mieux à ses préférences. Étant donné que le nombre de solutions non dominées générées par les évolutionnaires est souvent élevé, il devient difficile de choisir la solution à mettre en œuvre. Afin de faciliter cette tâche au décideur, il devient plus approprié d'intégrer ses préférences directement dans le processus de recherche. Pour cette raison, plusieurs approches basées sur des techniques différentes sont développées ces trois dernières décennies. Quelques-unes de ces méthodes sont listées ci-dessus, pour un état de l'art plus détaillé consulter [Rachmawati and Srinivasan, 2006] et [Bechikh *et al.*, 2015].

- **approches utilisant le classement des solutions** : dans [Greenwood *et al.*, 1996], pour classer les objectifs, les auteurs ont proposé une approche avec une somme pondérée basée sur une fonction de fitness multi-critère imprécise, où le classement des objectifs est implicitement dérivé du classement de certaines solutions candidates. La fonction de fitness utilisée est dite "imprécise" parce que les poids n'ont pas de valeurs spécifiques, mais ils sont générés à partir des préférences du décideur. Deb *et al.* [Deb *et al.*, 2010] ont proposé un algorithme évolutionnaire multi-objectif qui interagit avec le décideur d'une manière progressive. Pour progresser vers la solution la plus proche des préférences du décideur, l'algorithme intègre progressivement (après un certain nombre d'itérations) les informations sur ses préférences. Pour ce faire, un échantillon de solutions est présenté au décideur en lui demandant de les classer suivant ses préférences. Puis, ces informations sont intégrées dans la définition d'une fonction d'évaluation qui est utilisée pour les itérations suivantes de l'algorithme. Köksalan et Karahan [Köksalan and Karahan, 2010] ont proposé *an Interactive Territory Defining Evolutionary Algorithm : iTDEA* qui interagit avec le décideur au cours du processus d'optimisation pour intégrer ses préférences. L'idée de territoires consiste à définir autour de chaque solution non dominée une zone de l'espace objectif où aucun voisin se trouvant dans cette zone n'est accepté. À chaque itération, l'algorithme demande au décideur de choisir sa meilleure solution parmi un ensemble de solutions représentatives. Pour concentrer la recherche sur le voisinage de la solution sélectionnée, les auteurs proposent de réduire la taille des territoires des solutions qui sont proches de cette solution, ce qui permet d'augmenter la densité des solutions non dominées dans la zone contenant la solution choisie. Pour les autres solutions qui sont loin de cette zone, des territoires plus larges sont définis afin de réduire leur nombre (car elles sont moins intéressantes pour le décideur). Battiti et Passerini [Battiti and Passerini, 2010] ont intégré le paradigme "*learning while optimizing*" dans un algorithme évolutionnaire multi-objectif. Leur approche se caractérise par sa capacité à construire, à partir des jugements du décideur exprimés sur deux solutions différentes (en les classant suivant ses préférences), des modèles incrémentaux robustes de la fonction d'évaluation. Avec cet algorithme, il n'est pas nécessaire d'intégrer les préférences du décideur au départ vu que la fonction d'évaluation est construite progressivement en impliquant le décideur.
- **approches utilisant le classement des objectifs** : dans [Jin and Sendhoff, 2002], les auteurs ont utilisé dans un algorithme évolutionnaire une agrégation pondérée aléatoire et des techniques d'agrégation pondérée dynamique pour transformer des préférences floues en intervalles de poids. Concrètement, le décideur est invité à faire des comparaisons par paires sur l'ensemble des objectifs selon leurs importances en utilisant des énoncés linguistiques tels que "l'objectif f_1 est plus important que l'objectif f_2 ". La méthode développée par les auteurs convertit ces préférences floues en poids où chaque poids indique l'importance d'un objectif. Cette approche convertit le problème multi-objectif en un problème mono-objectif par une agrégation pondérée, mais varie de façon dynamique les poids pendant l'exécution. Cvetkovic et Parmee [Cvetkovic and Parmee, 2002] ont développé une approche similaire à celle développée dans [Jin and Sendhoff, 2002]. Pour intégrer des préférences floues dans des algorithmes évolutionnaires, ils ont proposé de générer une pondération des objectifs à partir de variables linguistiques exprimées par le décideur. Dans cette approche, le décideur est invité à faire des comparaisons par paires entre les objectifs en utilisant des expressions linguistiques telles que "plus important", "beaucoup moins important" et "ne se soucie pas". Cependant, le nombre de comparai-

sons par paires à réaliser peut devenir une tâche fastidieuse pour le décideur quand le nombre d'objectifs est élevé. Pour remédier à cet inconvénient, les auteurs ont proposé d'utiliser des relations transitives. Sur la base du vecteur de poids obtenu exprimant l'importance relative de chaque objectif, une nouvelle relation de dominance pondérée a été proposée. Dans [Rachmawati and Srinivasan, 2010], le décideur est invité à exprimer un ordre total ou partiel sur l'ensemble des objectifs par rapport à leur importance relative. Au lieu d'utiliser une pondération, le décideur exprime la différence d'importance entre chaque paire d'objectifs sous forme d'expressions telles que "l'objectif f_1 est préféré sur l'objectif f_2 ", "les objectifs f_1 et f_2 ont la même importance" ou "l'objectif f_1 et l'objectif f_2 sont incomparables". Par défaut, tous les objectifs sont incomparables mais les auteurs ont intégré un mécanisme interactif permettant au décideur de classer les objectifs tout au long du processus de recherche et obtenir une préférence globale cohérente. Cette approche a été incorporée dans NSGA-II de trois manières différentes : (1) les préférences sont considérées comme des contraintes, (2) les préférences sont utilisées dans la définition des fronts et (3) les préférences sont incluses dans le calcul de la distance *crowding*. Pour plus de détails sur ces trois utilisations, consulter [Rachmawati and Srinivasan, 2010].

- **approches utilisant la pondération** : Deb [Deb, 1999] intègre dans le calcul de la distance euclidienne de NSGA (*Nondominated Sorting Genetic Algorithm*) l'importance relative de chaque objectif sous forme d'une pondération des fonctions objectifs. Dans [Branke and Deb, 2005], Branke et Deb ont intégré dans le calcul de la distance euclidienne de NSGA-II une autre distance définie par la distance qui sépare chaque solution d'un hyper-plan qui représente la direction de l'espace de recherche souhaitée par le décideur. Zitzler et al. [Zitzler et al., 2007] ont proposé une version pondérée de l'indicateur hyper-volume pour guider la recherche vers une zone de l'espace objectif donnée par des coefficients de pondération et un point de référence.
- **approches utilisant des points de référence** : il existe dans la littérature beaucoup de travaux qui utilisent des points de référence pour intégrer les préférences du décideur. Dans [Fonseca and Fleming, 1993] et [Fonseca and Fleming, 1998], les auteurs ont modélisé les préférences du décideur sous forme d'un vecteur de buts à atteindre, le vecteur lui-même est représenté avec un point de référence. Entre deux solutions non dominées, celle qui atteint plus de buts est préférée sur l'autre. Dans le cas où les deux solutions atteignent les mêmes buts, la dominance Pareto classique est adoptée. Une approche similaire a été développée dans [Tan et al., 2003; Tan et al., 1999] où les auteurs ont proposé une variante de la dominance Pareto capable de combiner toute combinaison de buts et d'informations sur la priorisation des objectifs. Pour guider la recherche vers les zones du front Pareto correspondant aux préférences du décideur, Deb et al. [Deb et al., 2006] ont proposé R-NSGA-II, une version modifiée de NSGA-II [Deb et al., 2002] qui utilise des points de références fournis par le décideur pour exprimer ses préférences. Dans cette version, la distance *Crowding* de NSGA-II est modifiée. Pour chaque point de référence, la distance euclidienne normalisée de chaque solution est calculée puis les solutions sont triées par ordre croissant de cette distance. Un rang de un est attribué à la solution la plus proche du point de référence, la deuxième solution la plus proche reçoit un rang de deux et ainsi de suite. Une fois que les rangs des solutions sont définis pour tous les points de références, la distance *Crowding* d'une certaine solution est don-

née par le plus petit rang qu'elle a obtenu par rapport à tous les points de références. Ainsi, les solutions ayant la plus petite valeur pour la distance *Crowding* sont les plus proches des points de références (elles constituent donc le premier front), les solutions ayant la deuxième plus petite valeur forment le deuxième front, etc. La phase de sélection reste la même que sur NSGA-II, les solutions du premier front sont d'abord sélectionnées, puis celles du second front, puis le troisième, et ainsi de suite. A noter que ce même principe est utilisé par Bechikh et al. [Bechikh et al., 2010a; Bechikh et al., 2011; Bechikh et al., 2010b] pour détecter les régions concaves du front Pareto dans le cas où le décideur n'a pas exprimé ses préférences. Pour trouver la meilleure solution (une solution unique, la plus proche des préférences du décideur), Deb et Kumar [Deb and Kumar, 2007] ont combiné la méthode à direction de référence [Korhonen and Laakso, 1986] avec NSGA-II. Une direction de référence est définie par deux points : un point de départ et un point de référence. Dans cette approche, l'utilisateur fournit une ou plusieurs directions de référence dans l'espace objectif. Puis, l'algorithme cherche un ensemble de solutions efficaces correspondant à un certain nombre de points représentatifs le long de chaque direction. En utilisant une fonction de fitness, une solution unique est choisie pour une analyse plus approfondie. Cette procédure se poursuit jusqu'à ce qu'il n'y ait plus d'amélioration possible. Thiele et al. [Thiele et al., 2009] ont proposé l'algorithme *reference-based IBEA*, une version de l'algorithme IBEA [Zitzler and Künzli, 2004] qui intègre dans la définition de l'indicateur de qualité les préférences du décideur exprimées également sous forme de points de référence. Enfin, Liu et al. [Liu et al., 2017] ont étudié le potentiel des points de référence pour pousser la recherche vers des parties précises du front Pareto tout en garantissant la diversité des solutions dans les régions ciblées. Pour ce faire, ils ont proposé une méthode qui fonctionne en deux étapes : (1) générer de manière adaptative une série de points de référence avec une bonne convergence et une bonne distribution basée sur l'évolution de la population ; (2) focaliser la sélection sur des parties du front Pareto en calculant les distances entre les points de référence et les individus de la population.

3.3.2 Méthodes à base d'indicateurs de qualité

De nombreux algorithmes d'optimisation multi-objectif de la littérature sont basés sur l'évolution d'une population de solutions pour trouver un bon ensemble de solutions de compromis. L'utilisation des indicateurs binaires de qualité dans le processus de sélection a d'abord été proposé, en 2004, dans un algorithme évolutionnaire multi-objectif (EMO) [Zitzler and Künzli, 2004]. Depuis, plusieurs autres méthodes et études utilisant cette technique sont apparues telles que : *an EMO algorithm using the hypervolume measure as selection criterion* [Emmerich et al., 2005] (en 2005), *handling uncertainty in indicator-based multiobjective optimization* [Basseur and Zitzler, 2006] (en 2006) et *improving hypervolume-based EMO algorithms by using objective reduction methods* [Brockhoff and Zitzler, 2007] (en 2007). En 2011, Bader et Zitzler [Bader and Zitzler, 2011] ont proposé l'algorithme *HypE*, un algorithme d'estimation de l'hyper-volume pour l'optimisation multi-objectif qui utilise la simulation *Monte Carlo* pour approximer les valeurs exactes de l'hyper-volume. Leurs résultats expérimentaux indiquent que *HypE* est très efficace pour de nombreux problèmes ayant beaucoup d'objectifs (plus de quatre). En 2013, Phan et Junichi ont combiné l'indicateur *R2* avec l'algorithme IBEA dans un nouvel algorithme appelé *R2-IBEA* [Phan and Junichi, 2013]. La même année, Trautmann et al.

[Trautmann *et al.*, 2013] ont proposé l'algorithme *R2-EMOA*, un algorithme évolutionnaire utilisant l'indicateur *R2* dans la fonction d'évaluation. En 2014, Brockhoff *et al.* [Brockhoff *et al.*, 2014] ont également utilisé l'indicateur *R2* dans un autre algorithme multi-objectif (appelé *R2 indicator-based multiobjective search*). Phan et Junichi [Phan and Junichi, 2014] ont proposé dans un algorithme évolutionnaire l'opérateur U_{R2} – *dominance* pour évaluer la qualité des solutions pour des problèmes multi-objectifs qui contiennent du bruit dans leurs fonctions objectifs. Pour déterminer les meilleures solutions dans une population, l'opérateur U_{R2} utilise l'indicateur *R2* pour évaluer la qualité de chaque solution et un test de mesure statistique appelé *Mann-Whitney U-test* qui estime l'impact du bruit sur les valeurs des solutions pour l'indicateur *R2*. U_{R2} détermine ainsi quelles sont les solutions de la population qui sont statistiquement les meilleures. Plus récemment (2015), Jiang *et al.* [Jiang *et al.*, 2015] ont introduit un algorithme évolutionnaire (appelé *FV-MOEA*) utilisant une version simple et rapide de l'indicateur hyper-volume pour mettre à jour rapidement la contribution exacte de chaque solution à l'hyper-volume global de la population. Ils ont testé leur algorithme sur 44 instances avec de 2 à 5 objectifs et ils ont démontré que *FV-MOEA* obtient des valeurs élevées pour l'hyper-volume en un temps d'exécution plus court par rapport à d'autres MOEA utilisant aussi l'indicateur hyper-volume. La même année, Ben Mansour et Alaya [Ben Mansour and Alaya, 2015] ont utilisé les indicateurs de qualité dans un algorithme d'optimisation de colonies de fourmis pour un problème de sac à dos multi-objectif. Li *et al.* [Li *et al.*, 2015] ont intégré l'indicateur *R2* dans un algorithme multi-objectif à essaims de particules appelé *R2-MOPSO*. *R2-MOPSO* utilise la contribution à la valeur de l'indicateur *R2* des solutions archivées pour sélectionner les leaders qui seront utilisés pour mettre à jour l'essaim. En 2016, Falcoén-Cardona et Coello [Falcoén-Cardona and Coello, 2016] ont développé un nouvel algorithme multi-objectif de colonies de fourmis pour les espaces de recherche continus ($iMOACO_{\mathbb{R}}$), qui est basé sur les indicateurs $ACO_{\mathbb{R}}$ et *R2*. La même année, Díaz-Manríquez *et al.* [Díaz-Manríquez *et al.*, 2016] ont également combiné l'indicateur *R2* avec un algorithme d'optimisation à essaims de particules pour résoudre des problèmes multi-objectifs de différentes tailles.

Par ailleurs, les algorithmes de recherche locale sont également connus pour être très efficaces pour de nombreuses applications réelles, particulièrement sur des problèmes de grande taille. IBMOLS [Basseur and Burke, 2007] est un algorithme multi-objectif combinant à la fois l'utilisation des indicateurs de qualité avec la performance de la recherche locale. Contrairement à la plupart des algorithmes de recherche locale multi-objectif de la littérature qui reposent soit sur la relation de domination Pareto entre les solutions, soit sur les méthodes d'agrégation, IBMOLS utilise des indicateurs de qualité pour calculer la valeur de fitness de chaque solution. Il ne nécessite aucun mécanisme spécifique de préservation de la diversité, car cet aspect devrait être considéré directement dans l'indicateur lui-même. De plus, IBMOLS traite une population de taille fixe, ce qui permet à l'algorithme de trouver dans une seule exécution plusieurs solutions non dominées, sans aucun mécanisme pour contrôler le nombre de solutions non dominées au cours du processus de recherche locale (problème rencontré avec les algorithmes multi-objectifs de recherche locale classique [Paquete *et al.*, 2004]). Enfin, IBMOLS ne nécessite qu'un petit nombre de paramètres.

Plusieurs études de la littérature ont montré l'efficacité de la méthode IBMOLS sur différents problèmes multi-objectifs. En 2008, pour résoudre un cas bi-objectif du *Ring star problem*, Liefoghe *et al.* [Liefoghe *et al.*, 2008] ont comparé IBMOLS aux algorithmes IBEA [Zitzler and Künzli, 2004], NSGA-II [Deb *et al.*, 2002] et un autre algorithme qu'ils ont proposé, appelé *simple elitist evolutionary algorithm* (SEEA). Ils ont conclu que la version itérée d'IBMOLS

était, en général, significativement meilleure que tous les autres algorithmes considérés. Dans la seconde partie de leur papier, ils ont proposé deux schémas de coopération entre SEEA et la version non itérée d'IBMOLS. Les deux consistent à utiliser SEEA comme principal processus de recherche et à lancer IBMOLS régulièrement pour intensifier la recherche. Dans le premier schéma, une procédure *periodic cooperative search* (PCS) lance IBMOLS systématique à chacune de ses étapes. Dans le second schéma, une procédure *auto-adaptive cooperative search* (ACS), lance IBMOLS à une étape spécifique de l'algorithme et uniquement si une certaine condition est vérifiée. En comparaison à des métaheuristiques autonomes, ces deux méthodes de recherche hybrides améliorent statistiquement les résultats sur un grand nombre d'instances, en particulier celles de grande taille. En 2012, Basseur et al. [Basseur et al., 2012a] ont montré l'efficacité d'IBMOLS sur trois problèmes différents (*Flow-Shop*, *Ring Star* et *Nurse Rostering*), en utilisant I_{HV} et I_{ϵ} . La même année, Basseur et al. [Basseur et al., 2012b] ont proposé l'algorithme *HBMOLS* qui utilise dans le processus de sélection l'indicateur unaire I_{HC} qui sert à évaluer la contribution hyper-volume de chaque solution à l'hyper-volume globale de la population. Puis, en 2013, pour résoudre un problème d'ordonnancement bi-objectif, ils ont combiné HBMOLS avec un autre algorithme *Path Relinking* (MOPR) [Zeng et al., 2013], où HBMOLS est utilisé pour améliorer la qualité de la population et l'algorithme MOPR choisit les deux meilleures solutions d'une archive contenant les solutions non dominées puis sélectionne un ensemble de solutions parmi celles qui se trouvent sur le chemin généré entre les deux solutions choisies. Cet ensemble ainsi construit est utilisé pour initialiser la nouvelle population de HBMOLS. En 2015, Tangpattanakul et al. [Tangpattanakul et al., 2015] ont appliqué IBMOLS avec I_{HV} pour résoudre un problème d'optimisation multi-objectif associé à la sélection et à la planification des observations d'un satellite. Comparé à l'algorithme BRKGA [Gonçalves and Resende, 2011] sur différentes instances, IBMOLS a obtenu de bien meilleurs résultats. Enfin, Tangpattanakul [Tangpattanakul, 2015] a également montré l'efficacité d'IBMOLS avec I_{HV} sur un problème p-médian bi-objectif.

3.4 Approche proposée pour résoudre le POPAMO

Après une première étude que nous avons menée sur l'efficacité de ϵ -IBMOLS pour résoudre le POPAMO [Chabane et al., 2015] qui a montré que ϵ -IBMOLS ne garantit pas toujours la diversité des solutions générées, nous proposons dans cette section (pour la première fois) de combiner les performances de la méthode IBMOLS avec l'indicateur $R2$ tel qu'il est présenté dans la section 3.2.2.3. Avant de décrire la méthode IBMOLS (section 3.4.2), précisons d'abord les raisons qui nous ont motivé à proposer cette approche.

3.4.1 Motivations

Le choix d'un indicateur de qualité approprié au problème à résoudre, est primordial pour obtenir une bonne approximation du front Pareto. De ce fait, plusieurs raisons ont motivé notre choix de combiner la méthode IBMOLS avec l'indicateur $R2$:

- à notre connaissance, tous les travaux et les expérimentations réalisés avec l'algorithme IBMOLS sont réalisés sur des instances qui ne dépassent pas trois objectifs. Or, nous savons que parfois une méthode efficace sur des instances de deux ou trois objectifs peut montrer ses limites dès qu'on augmente le nombre d'objectifs ;

- très peu d'algorithmes sont développés avec l'indicateur $R2$ et, à notre connaissance, cet indicateur n'a jamais été utilisé avec l'algorithme IBMOLS.
- dans [Basseur *et al.*, 2012a], l'efficacité des deux indicateurs I_ϵ et I_{HV} a été comparée aux indicateurs I_{Ben} , I_{Fon} et I_{Sri} , qui sont dérivés des méthodes de classement classiques basées sur la relation de la domination Pareto. Les auteurs ont conclu que, pour le cas bi-objectif, les indicateurs Epsilon et HV ont tendance à surpasser les indicateurs basés sur la domination Pareto, surtout lorsque la taille du problème augmente, I_ϵ étant globalement plus efficace. D'un autre côté, les expérimentations menées sur HBMOLS [Basseur *et al.*, 2012b] (où l'indicateur I_{HC} défini par rapport à la contribution de chaque solution à l'hyper-volume globale de la population est utilisé) ont montré la supériorité de HBMOLS sur différentes versions d'IBMOLS. Néanmoins, on sait que I_{HC} est très coûteux à calculer lorsque le nombre d'objectifs est supérieur à deux ;
- dans les expérimentations que nous avons réalisées avec $\epsilon - IBMOLS$ pour résoudre le POPAMO [Chabane *et al.*, 2015], nous avons constaté que la diversité des solutions obtenues est réduite avec I_ϵ ;
- il est rapporté dans [Wessing and Naujoks, 2010], qu'en pratique, les indicateurs I_{HV} et I_{R2} montrent un comportement corrélé en termes de performances.

Pour toutes ces raisons, nous proposons donc de combiner IBMOLS avec l'indicateur $R2$, ce qui nous permet, d'un côté, d'avoir des résultats plus ou moins proches de ceux qu'on pourrait obtenir avec l'indicateur HV mais en beaucoup moins de temps. D'un autre côté, de vérifier l'efficacité de l'algorithme IBMOLS sur un problème pratique pouvant avoir jusqu'à huit objectifs.

3.4.2 $R2$ -IBMOLS

Soit P la population courante de l'algorithme IBMOLS et E une archive de solutions non dominées (au sens de la dominance Pareto). Une procédure de recherche locale est appliquée sur P comme décrit ci-dessous.

Une étape de recherche locale de l'algorithme IBMOLS correspond à une étape de recherche locale appliquée sur chaque solution x de P . Un voisin x^* est accepté si sa valeur pour l'indicateur considéré I est meilleure que la pire solution dans P . La génération du voisinage de x s'arrête lorsque tout son voisinage est exploré ou une fois qu'un voisin améliorant est trouvé (premier voisin qui améliore la qualité de P par rapport à l'indicateur I). En tant que tel, le voisinage n'est pas exploré entièrement pour la recherche du meilleur voisin. Deux raisons principales ont guidé ce choix : (i) il permet souvent d'accélérer la convergence de la population, car la plupart du temps, seule une petite partie du voisinage est générée. (ii) la sélection du meilleur voisin conduit à des étapes de recherche locale déterministes (un seul chemin possible pour passer d'une solution initiale à un ensemble d'optima locaux). Le fait de choisir des mouvements aléatoires nous permet d'atteindre différents optima locaux (au sens de l'optimisation multi-objectif) à partir d'une seule solution initiale. La recherche locale d'IBMOLS est terminée lorsque l'archive E n'a reçu aucune nouvelle solution lors d'une étape de recherche locale complète.

En outre, nous considérons que toutes les valeurs des fonctions objectifs de toutes les solutions de P sont normalisées. Pour ce faire, la valeur minimale m_j et la valeur maximale M_j de chaque fonction objectif f_j dans la population P sont calculées en premier :

$$\begin{cases} m_j = \min_{x \in P}(f_j(x)) \\ M_j = \max_{x \in P}(f_j(x)) \end{cases} \quad (3.6)$$

Puis la valeur de chaque fonction objectif j pour tout individu $x \in P$ est normalisée comme suit :

$$NF_j(x) = \frac{f_j(x) - m_j}{M_j - m_j} \quad (3.7)$$

avec $NF_j(x)$ est la valeur de la j^{me} fonction objectif normalisée pour l'individu x . Les valeurs extrêmes de la population sont recalculées après l'initialisation de la population et après chaque étape de recherche locale (voir l'algorithme 1). Pour calculer la différence de qualité entre deux solutions x_1 et x_2 , c'est-à-dire, $I(x_1, x_2)$, les valeurs normalisées des fonctions objectifs $NF_j(x)$ sont utilisées. La description détaillée d'IBMOLS est donnée dans l'algorithme 1.

Algorithm 1 Algorithme IBMOLS

Input : P (initial population of size N); I_{R2} (binary indicator)

Output : E (Pareto approximation set)

Step 1 : $E \leftarrow \text{nondominatedSolutions}(P)$

Step 2 - fitness assignment : Calculate fitness values of individual x in P , i.e., $fit(x) = I_{R2}(P \setminus \{x\}, x)$

Step 3 - local search step : For all $x \in P$ do :

update, for each objective function f_j , the minimal m_j and maximal M_j values in P (for objective functions normalisation)

repeat

1) $x^* \leftarrow$ one unexplored neighbor of x

2) $P \leftarrow P \cup x^*$

3) compute x^* fitness : $I_{R2}(P \setminus \{x^*\}, x^*)$

4) update all $z \in P$ fitness values : $fit(z) + = I_{R2}(x^*, z)$

5) $w \leftarrow$ the worst individual in P

6) remove w from P

7) update all $z \in P$ fitness values : $fit(z) - = I_{R2}(w, z)$

until all neighbors are explored or $w \neq x^*$

Step 4 - termination : $E \leftarrow \text{nondominatedSolutions}(E \cup P)$

if E does not change **then**

return E

else

perform another local search step

end if

Dans la version itérée d'IBMOLS (algorithme 2), une approximation PO du Front Pareto est maintenue et mise à jour après chaque itération d'IBMOLS. Après chaque recherche locale, une nouvelle population initiale est créée pour la prochaine itération d'IBMOLS en utilisant

la fonction *initWalk*. Même si la population initiale de la première itération est entièrement créée d'une manière aléatoire, lorsque nous itérons le processus de recherche locale, la fonction *initWalk* génère une nouvelle population P pour la prochaine itération en utilisant des informations sur les bonnes solutions obtenues lors des itérations précédentes. En effet, la fonction *initWalk* applique des mutations aléatoires sur N solutions sélectionnées au hasard de PO (chaque solution de PO ne peut être sélectionnée qu'une fois). Pour chaque solution sélectionnée, une mutation est appliquée avec une probabilité de $1/n$ (où n est le nombre d'actions) et la solution mutée est ajoutée à la population si elle n'y était pas déjà et si elle vérifie la contrainte budgétaire β et les seuils définis pour les objectifs (la contrainte c_j dans notre modèle). Quand $|PO| < N$, toutes les solutions de PO sont sélectionnées et les individus manquants de P sont complétés avec de nouvelles solutions aléatoires.

Algorithm 2 Version itérée de l'algorithme IBMOLS

Input : N (population size) ; I_{R2} (binary indicator)
Output : PO (Pareto approximation set)
Step 1 : $PO \leftarrow \emptyset$
Step 2 :
while stopping condition not achieved **do**
 1) $P \leftarrow \text{initWalk}(PO, N)$
 2) $E \leftarrow \text{IBMOLS}(P, I_{R2})$
 3) $PO \leftarrow \text{nondominatedSolutions}(PO \cup E)$
end while
Step 3 : Return PO

On note qu'IBMOLS n'est défini que par deux paramètres principaux (la taille de la population N et l'indicateur binaire I) ainsi que la fonction *initWalk* qui initialise la population. Ces paramètres peuvent être réglés dynamiquement pendant la recherche ou fixés préalablement en fonction de l'instance et du problème considéré.

La taille de la population d'IBMOLS est fixée et ne dépend pas du problème, des fonctions objectifs ou des dimensions de l'espace considéré. Basseur et al. [Basseur et al., 2012a] ont montré sur trois problèmes différents qu'IBMOLS est efficace particulièrement lorsqu'il est utilisé avec une population de petite taille (pas plus de 15 individus). Sur la base de cette conclusion, nous avons utilisé des populations de 10 solutions dans nos expérimentations.

3.4.3 Expérimentations

Dans cette section, nous présentons un ensemble d'expérimentations visant à permettre une comparaison des résultats obtenus avec $R2$ -IBMOLS et ϵ -IBMOLS et un autre algorithme de référence NSGA-II [Deb et al., 2002] appliqués au problème d'optimisation du plan d'actions multi-objectif. Avant de présenter le protocole d'expérimentation et de fournir les résultats (section 3.4.4), nous décrivons d'abord comment les instances utilisées ont été générées.

3.4.3.1 Instances

Sur la base du modèle formel du POPAMO donné dans la section 2.3 du chapitre 2 et une étude d'une dizaine de plans d'actions réels, nous avons généré plusieurs instances partiellement structurées et de différentes tailles : un nombre d'actions $n \in \{50, 100, 150, 250, 500\}$ et

un nombre d'objectifs $m \in \{2, 3, 4, 5, 6, 8\}$. Pour être aussi proche que possible des plans d'actions réels, pour chaque fonction objectif, une action a une probabilité de 50% d'être neutre, 40% d'avoir un impact positif et 10% d'avoir un impact négatif. De plus, le coût de 40% des actions est mis à zéro. Les valeurs non nulles des impacts des actions sont prises uniformément dans l'intervalle $[-100, 100]$. Les coûts des actions non nuls sont pris uniformément dans l'intervalle $[-10^4, 10^4]$. Les instances utilisées dans nos expérimentations sont disponibles à cette adresse : <http://www.info.univ-angers.fr/pub/ha0/gepiplanning/R2-IBMOLS.zip>.

3.4.3.2 Comparaison avec NSGA-II

De manière générale, les algorithmes évolutionnaires multi-objectifs attribuent un score à chaque individu de la population courante selon qu'il est dominé ou non par d'autres individus de la population ou que lui-même domine d'autres individus. Ce score détermine principalement la chance de l'individu d'être propagé sur les descendants de la population ou pas. NSGA-II [Deb et al., 2002] est un exemple classique de cette classe d'algorithmes.

NSGA-II est un algorithme génétique qui utilise une méthode de *ranking* qui attribue aux solutions non dominées de la population courante le meilleur score les classant ainsi sur un front de niveau 1. Puis les solutions qui ne sont dominées que par les solutions du premier front, reçoivent le second meilleur score et constituent le front de niveau 2 de la population, et ainsi de suite. De cette manière, la population est organisée en couches où chaque couche contient des solutions non comparables entre elles. Pour choisir les individus qui seront utilisés pour générer la population enfant, NSGA-II sélectionne d'abord ceux du premier front, puis le second front, etc. Jusqu'à ce que la taille de la population soit atteinte. Parfois, on ne peut pas choisir tous les individus du dernier front sélectionné, sinon on dépasserait la taille de la population. Dans ce cas, NSGA-II utilise une seconde mesure, la distance de *crowding*, visant à choisir uniquement les individus du dernier front qui améliorent la diversité de la population dans l'espace objectif.

3.4.3.3 Protocole d'expérimentation

En utilisant les instances décrites à la section 3.4.3.1, nous avons testé *R2-IBMOLS*, ϵ -*IBMOLS* et NSGA-II avec les paramètres les plus couramment utilisés dans la littérature. Pour NSGA-II, nous avons utilisé une population de taille 100, une probabilité de mutation de $1/n$ (où n est le nombre d'actions). Pour ϵ -*IBMOLS* et *R2-IBMOLS*, nous avons utilisé la version itérée d'*IBMOLS* avec une population d'une taille de 10 individus. Pour une population courante P , un point de référence z^* et un ensemble de vecteurs de poids Λ , la valeur de fitness de chaque solution x dans P est évaluée, par rapport au reste de la population, en utilisant la formule 3.8 pour ϵ -*IBMOLS* et la formule 4.6 pour *R2-IBMOLS*.

$$I_\epsilon(P \setminus \{x\}, x) = \min_{z \in P \setminus \{x\}} (I_\epsilon(z, x)) \quad (3.8)$$

$$I_{R2}(x, \Lambda, z^*) = I_{R2}(P, \Lambda, z^*) - I_{R2}(P \setminus \{x\}, \Lambda, z^*) \quad (3.9)$$

R2-IBMOLS est utilisé avec le point de référence $z^* = (2, 2, \dots, 2)$ (après normalisation des valeurs des fonctions objectifs) et 100 vecteurs ($|\Lambda| = 100$) distribués uniformément dans l'espace objectif. La méthode de génération de ces vecteurs est décrite dans la section 3.4.3.5.

De plus, une population de taille fixe est utilisée et la stratégie de sélection suivante est adoptée : un voisin aléatoire de chaque individu de la population courante est sélectionné pour

être membre de la population enfant dans NSGA-II ou pour intégrer la population courante d'IBMOLS. La génération du voisinage est la même pour les trois méthodes : le i^{me} voisin de la solution $x = (a_1, a_2, \dots, a_n)$ est obtenu en modifiant la valeur de a_i et uniquement les voisins qui vérifient la contrainte de coût β et les seuils des objectifs sont considérés comme candidats (si le le voisin courant viole une des contraintes, un autre voisin est généré). Pour toutes les instances, la contrainte budgétaire β est fixée à un million et les seuils ainsi que les poids des objectifs sont tous fixés à 1 ($c_j \geq 1$ et $k_j = 1 \forall j \in \{1, \dots, m\}$).

Pour la mesure de performance des trois méthodes, nous avons effectué 30 exécutions de chaque algorithme sur chaque instance. La condition d'arrêt de chaque exécution correspond à $200 \times n \times m$ solutions évaluées (où n est le nombre d'actions et m est le nombre d'objectifs). Les expérimentations sont réalisées sur une machine avec une CPU Intel Core i5-2400 de 2×3.10 GHz et 16 Gb de RAM.

Pour l'analyse statistique, nous avons utilisé le package PISA, décrit dans [Knowles *et al.*, 2006]. Nous avons effectué une comparaison par paires des algorithmes pour chaque instance en utilisant le test de Mann-Whitney. La correction de Bonferroni est également appliquée pour ajuster les niveaux individuels de signification statistique. Avec la correction de Bonferroni, pour atteindre un niveau de signification l , les tests en paires devraient être effectués avec un niveau de $\frac{l}{h}$, où h est le nombre de paires comparées, qui est de trois dans notre cas. À l'aide des indicateurs R et Epsilon, nous obtenons une valeur p value correspondant à l'hypothèse "*le premier algorithme est plus performant que le second*". La valeur p value correspond au niveau de signification statistique le plus faible pour lequel l'hypothèse nulle est rejetée. Dans nos expérimentations, nous considérons que l'algorithme A_1 surpasse l'algorithme A_2 si le test de Mann-Whitney fournit la conclusion " *A_1 est plus performant que A_2* " avec un niveau de confiance supérieur à 95% (p value ≤ 0.05), ce qui correspond à un niveau individuel de signification statistique de 0.0167, après application de la correction Bonferroni. La procédure de test et le package PISA sont disponibles à l'URL suivante : <http://www.tik.ee.ethz.ch/sop/pisa/?page=assessment.php>.

3.4.3.4 Génération de la population initiale

Pour les trois algorithmes ($R2$ -IBMOLS, ϵ -IBMOLS et NSGA-II), la population initiale est générée de manière aléatoire tout en respectant les contraintes du problème : i) les coûts des individus de la population ne dépassent pas le budget β alloué au plan d'actions ; ii) chaque individu accepté x doit améliorer tous les objectifs (c'est-à-dire $f_j(x) > 0 \forall j \in \{1, \dots, m\}$). La procédure de génération de la population initiale est décrite dans l'algorithme 3.

3.4.3.5 Génération des vecteurs

Pour que les vecteurs de poids générés soit uniformément distribués dans l'espace objectif, nous avons utilisé l'algorithme 4, initialement proposé par Suzuki et Phan [Phan and Junichi, 2013].

L'algorithme 4 utilise l'indicateur HV qui maximise l'hyper-volume global dominé par l'ensemble des vecteurs dans l'espace objectif. Cette méthode est intéressante car elle ne dépend pas de la dimension de l'espace objectif et fonctionne de la même manière, que ce soit pour les espaces à petite ou à grande dimension. Cependant, vu que l'algorithme 4 utilise l'indicateur HV , il peut s'avérer très coûteux en temps d'exécution si les vecteurs sont générés à chaque itération d'IBMOLS. D'autant plus que nous avons à traiter un problème avec beaucoup

Algorithm 3 Génération de la population initiale

Input : N (population size); β (budget)**Output** : P (initial population) $P = \emptyset$ **while** $|P| < N$ **do** $x = \text{randomSolution}()$ **while** $\text{cost}(x) > \beta$ **do** $x = x$ less one random action with positive cost **end while** **for** each objective function f_j **do** **while** $f_j(x) \leq 0$ **do** $x = x$ less one random action with negative value for f_j **end while** **end for** $P = P \cup \{x\}$ **end while**Return P

Algorithm 4 Génération des vecteurs de poids

Input : t_{max} , the maximum number of iterations**Input** : l , the number of weight vectors to be generated**Input** : m , the number of objectives**Output** : Λ , the set of generated weight vectors $t = 0$ $\Lambda = \emptyset$ **while** $t < t_{max}$ **do** Randomly choose a vector σ in $[0, 1]^{m-1}$: $\sigma = (\sigma_1, \sigma_2, \dots, \sigma_{m-1})$ Sort σ_j in σ ascending order, such that $\sigma_1 \leq \sigma_2 \leq \dots \leq \sigma_{m-1}$ Create a vector $\lambda = (\lambda_1, \lambda_2, \dots, \lambda_{m-1}, \lambda_m) = (\sigma_1, \sigma_2 - \sigma_1, \dots, \sigma_{m-1} - \sigma_{m-2}, 1 - \sigma_{m-1})$ $\Lambda = \Lambda \cup \{\lambda\}$ **if** $|\Lambda| > l$ **then** Calculate hypervolume contribution of each λ in Λ : $I_{HC}(\lambda) = HV(\Lambda) - HV(\Lambda \setminus \{\lambda\})$ $\lambda^* = \text{argmin}_{\lambda \in \Lambda} I_{HC}(\lambda)$ $\Lambda = \Lambda \setminus \{\lambda^*\}$ **end if** $t = t + 1$ **end while**

d'objectifs. Pour éviter cet inconvénient, dans nos expérimentations, les vecteurs de poids sont générés une fois et restent les mêmes pendant toutes les itérations d'IBMOLS.

3.4.4 Résultats expérimentaux

Dans cette section, nous présentons les résultats expérimentaux obtenus sur les instances et le protocole expérimental décrits précédemment. Puis, dans la section suivante, nous introduisons les apports pratiques que l'approche proposée pourrait avoir pour assister le décideur dans l'élaboration de plans d'actions efficaces.

Instance	Évaluation avec l'indicateur ϵ			Évaluation avec l'indicateur R		
	NSGA-II	ϵ -IBMOLS	R2-IBMOLS	NSGA-II	ϵ -IBMOLS	R2-IBMOLS
2_50	<u>0.323</u>	0.567	0.144	<u>0.118</u>	0.310	0.027
2_100	<u>0.314</u>	0.549	0.105	<u>0.132</u>	0.303	0.018
2_150	<u>0.346</u>	0.577	0.124	<u>0.144</u>	0.279	0.023
2_250	<u>0.335</u>	0.524	0.126	<u>0.144</u>	0.279	0.028
2_500	<u>0.325</u>	0.619	0.116	<u>0.149</u>	0.334	0.030
3_50	<u>0.296</u>	0.425	0.171	<u>0.068</u>	0.080	0.022
3_100	<u>0.315</u>	0.433	0.175	<u>0.085</u>	0.117	0.026
3_150	<u>0.365</u>	0.413	0.181	<u>0.090</u>	0.139	0.030
3_250	0.432	<u>0.376</u>	0.118	0.145	<u>0.074</u>	0.021
3_500	0.532	<u>0.261</u>	0.155	0.148	<u>0.075</u>	0.032
4_50	<u>0.261</u>	0.350	0.174	<u>0.044</u>	0.059	0.016
4_100	0.466	<u>0.192</u>	0.086	0.083	<u>0.034</u>	0.009
4_150	0.497	<u>0.248</u>	0.191	0.080	<u>0.050</u>	0.026
4_250	0.408	<u>0.118</u>	0.086	0.035	<u>0.021</u>	0.009
4_500	0.571	<u>0.122</u>	0.111	0.108	<u>0.021</u>	0.014
5_50	<u>0.227</u>	0.366	0.135	<u>0.032</u>	0.042	0.011
5_100	0.399	<u>0.208</u>	0.169	0.052	<u>0.030</u>	0.014
5_150	0.446	<u>0.191</u>	0.129	0.068	<u>0.028</u>	0.012
5_250	0.505	<u>0.089</u>	0.039	0.096	<u>0.011</u>	0.003
5_500	0.381	<u>0.068</u>	0.032	0.074	<u>0.009</u>	0.003
6_50	<u>0.232</u>	0.348	0.167	<u>0.031</u>	0.052	0.014
6_100	0.486	<u>0.144</u>	0.092	0.057	<u>0.012</u>	0.004
6_150	0.510	<u>0.127</u>	0.079	0.084	<u>0.013</u>	0.004
6_250	0.514	<u>0.091</u>	0.054	0.057	<u>0.009</u>	0.003
6_500	0.556	<u>0.153</u>	0.065	0.089	<u>0.015</u>	0.006
8_50	<u>0.158</u>	0.286	0.122	<u>0.013</u>	0.015	0.005
8_100	0.227	<u>0.105</u>	0.084	0.008	<u>0.006</u>	0.002
8_150	0.246	<u>0.077</u>	0.062	0.005	<u>0.004</u>	0.001
8_250	0.271	<u>0.05</u>	0.041	0.004	<u>0.003</u>	0.001
8_500	0.436	<u>0.225</u>	0.076	0.073	<u>0.021</u>	0.009

TABLE 3.1 – Comparaison des valeurs moyennes des indicateurs ϵ et R obtenues avec PISA pour les algorithmes NSGA-II, ϵ -IBMOLS et R2-IBMOLS.

Le tableau 3.1 rapporte la comparaison entre NSGA-II, ϵ -IBMOLS et $R2$ -IBMOLS en termes de valeurs moyennes obtenues pour les indicateurs R et Epsilon sur 30 exécutions, en utilisant un ensemble de 30 instances de différentes tailles. La première colonne présente le nom de l'instance, indiqué par ses principales caractéristiques : m et n correspondent respectivement au nombre d'objectifs et au nombre d'actions de l'instance. Les valeurs en gras signifient que l'algorithme correspondant est statistiquement plus performant aux deux autres pour l'instance et l'indicateur correspondant avec un niveau de confiance supérieur ou égal à 95%. Les valeurs soulignées signifient que l'algorithme correspondant est statistiquement plus performant que l'algorithme ayant les valeurs dans le style normal, mais il est moins performant que l'algorithme ayant les valeurs en caractère gras, pour l'instance et l'indicateur considérés (également avec un niveau de confiance supérieur ou égal à 95%).

Du tableau 3.1, nous pouvons conclure que $R2$ -IBMOLS est plus performant sur toutes les instances considérées du POPAMO. NSGA-II converge mieux que ϵ -IBMOLS dans le cas des petites instances (toutes les instances ayant 2 objectifs ou 50 actions ainsi que les instances 3_100 et 3_150). Par contre, ϵ -IBMOLS converge mieux que NSGA-II dès qu'on dépasse quatre objectifs.

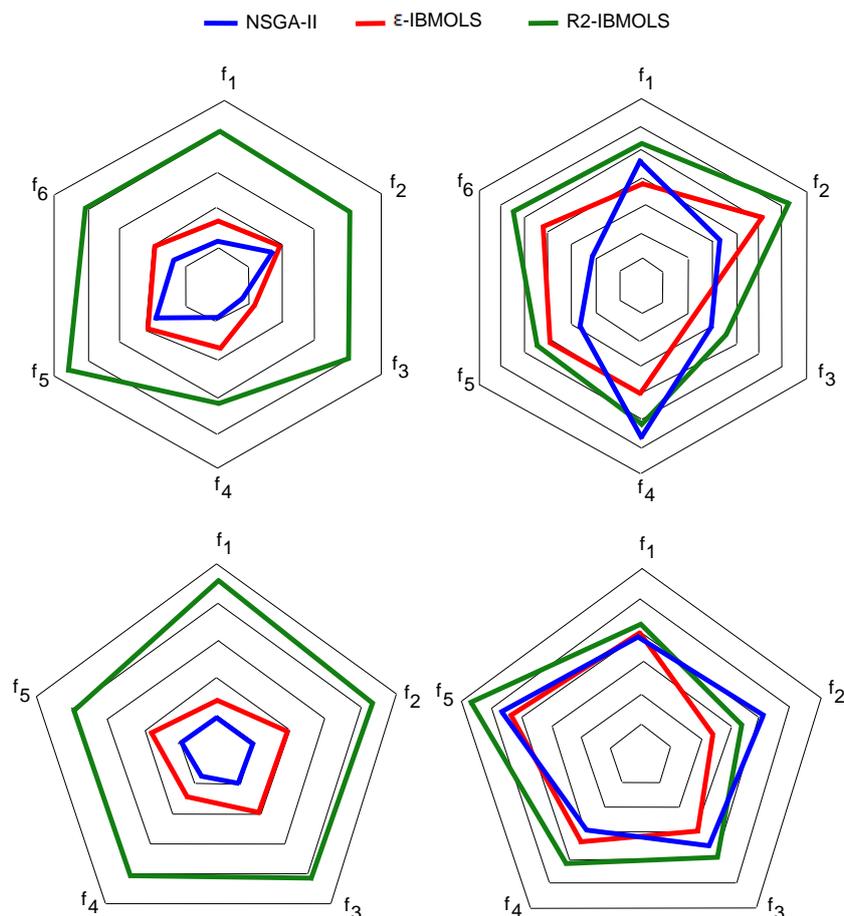


FIGURE 3.4 – Valeurs maximales obtenues avec NSGA-II, ϵ -IBMOLS et $R2$ -IBMOLS sur 30 exécutions pour les objectifs de quatre instances représentatives, avec 50 ou 500 actions et 5 ou 6 objectifs "6_500" (haut gauche), "6_50" (haut droite), "5_500" (bas gauche) et "5_50" (bas droite).

La figure 3.4 montre les valeurs maximales obtenues pour chacune des fonctions objectifs avec NSGA-II, ϵ -IBMOLS et $R2$ -IBMOLS sur les 30 exécutions effectuées pour quatre instances représentatives du POPAMO (instances avec 5 et 6 objectifs). En effet, dans les plans d'actions réels, le nombre d'actions peut varier de 50 actions à 500 actions pour quatre à six objectifs. Il est clair de la figure 3.4 que, comparé à NSGA-II et ϵ -IBMOLS, $R2$ -IBMOLS atteint mieux les solutions extrêmes maximisant ainsi les fonctions objectifs du problème. Sur les quatre instances considérées, les valeurs maximales de toutes les fonctions objectifs obtenues avec $R2$ -IBMOLS sont plus élevées que celles obtenues avec NSGA-II et ϵ -IBMOLS, à l'exception de l'objectif f_2 de l'instance "2_50" (bas droite de la figure 3.4) et l'objectif f_4 de l'instance 6_500 (haut droite de la figure 3.4).

Souvent dans les cas pratiques de problèmes multi-objectifs, choisir la solution finale parmi toutes celles qui ne sont pas dominées n'est pas toujours une tâche facile pour le décideur. Par

Instance	NSGA-II	ϵ -IBMOLS	$R2$ -IBMOLS
2_50	2 150	47	32
2_100	1 529	66	48
2_150	1 450	66	67
2_250	1 300	86	89
2_500	1 279	119	135
3_50	2 528	210	333
3_100	2 564	270	629
3_150	3 061	335	845
3_250	3 947	495	1,696
3_500	5 165	703	2,856
4_50	585	87	358
4_100	1 009	164	671
4_150	1 131	235	905
4_250	1 143	305	2 348
4_500	1 769	663	3 558
5_50	1 148	215	522
5_100	1 979	426	1 447
5_150	2 114	502	1 949
5_250	3 046	598	3 266
5_500	4 243	1 058	9 501
6_50	1 542	341	651
6_100	2 828	697	2 097
6_150	3 952	908	3 725
6_250	5 075	1 210	7 727
6_500	7 687	2 492	18 900
8_50	3 991	965	3 989
8_100	7 988	2 151	4 918
8_150	11 989	3 080	5 317
8_250	19 991	4 919	19 176
8_500	39 996	9 325	45 419

TABLE 3.2 – Nombres moyens de solutions non dominées générées par NSGA-II, ϵ -IBMOLS et $R2$ -IBMOLS sur 30 exécutions.

conséquent, pour la lui faciliter, il est recommandé de lui présenter un nombre réduit de solutions non dominées. Mais il faut que ces solutions soient les plus proches possibles de ses préférences. En effet, dans le problème étudié dans cette thèse (le POPAMO), une solution pourrait avoir plusieurs dizaines d'actions différentes et le décideur devrait pouvoir choisir facilement le plan d'actions le plus adéquat pour son projet d'amélioration. La table 3.2 montre le nombre moyen de solutions non dominées obtenues avec chacun des algorithmes sur les 30 exécutions effectuées.

Nous constatons (du tableau 3.2) que le nombre de solutions obtenues est très variable pour les trois algorithmes, allant de plusieurs dizaines de solutions pour les instances de petite taille à plusieurs milliers de solutions pour les instances de grande taille. Même pour la plus petite instance ("2_50"), le nombre de solutions est élevé pour que le décideur puisse en choisir une facilement (32 solutions pour *R2-IBMOLS*, 47 solutions pour ϵ -*IBMOLS* et 2 150 solutions pour *NSGA-II*). Par conséquent, il serait intéressant de développer d'autres techniques afin de réduire le nombre de solutions non dominées. L'utilisation de la dominance de Lorenz à la place de la dominance Pareto est un moyen efficace pour définir des préférences même entre des solutions non dominées. Cette idée est largement développée dans le chapitre 4 où plusieurs algorithmes sont proposés.

Instance	NSGA-II	ϵ -IBMOLS	R2-IBMOLS
2_50	3	0.18	0.08
2_100	11	0.90	0.38
2_150	28	2	1
2_250	79	9	4
2_500	368	60	31
3_50	6	1	1
3_100	24	9	26
3_150	52	29	106
3_250	153	157	1 035
3_500	661	1 279	12 193
4_50	284	1	3
4_100	1 196	7	41
4_150	2 433	27	165
4_250	6 790	101	2 677
4_500	28 709	1 481	20 392
5_50	417	3	8
5_100	1 458	33	183
5_150	3 251	99	792
5_250	8 696	361	5 405
5_500	36 078	4 141	152 636
6_50	513	10	14
6_100	13 597	114	494
6_150	4 439	385	2 987
6_250	36 056	1 629	29 026
6_500	45 444	39 494	690 734

TABLE 3.3 – Temps d'exécution moyens de *NSGA-II*, ϵ -*IBMOLS* et *R2-IBMOLS* sur 30 exécutions.

Pour le problème d'optimisation de plans d'actions étudié dans cette thèse, le temps d'exécution des algorithmes n'est pas nécessairement un aspect principal pour évaluer leurs performances, puisque le plan d'actions s'inscrit dans un projet qui se fait sur cinq ans. Néanmoins, nous notons, à partir de la table 3.3 qui montre les temps d'exécution moyens (en secondes) obtenus pour chacun des trois algorithmes testés, que même si $R2$ -IBMOLS est plus performant que NSGA-II et ϵ -IBMOLS (voir table 3.1), son temps d'exécution devrait être réduit, en particulier pour les instances de grande taille où $R2$ -IBMOLS prend beaucoup de temps. Les temps d'exécution obtenus avec $R2$ -IBMOLS pour certaines instances de grandes tailles sont : 12 193 pour l'instance "3_500", 20 392 pour l'instance "4_500", 152 636 pour l'instance "5_500", 29 026 pour l'instance "6_250" et 690 734 pour l'instance "6_500". En effet, pour toutes les instances avec plus de 2 objectifs, le temps d'exécution moyen de $R2$ -IBMOLS est significativement plus élevé que celui de ϵ -IBMOLS et NSGA-II et dépend essentiellement du nombre de vecteurs de poids utilisés. Cependant, il pourrait être réduit en exploitant une propriété de l'indicateur $R2$ qui indique que le placement optimal des solutions dans l'espace objectif dépend uniquement d'un sous-ensemble des vecteurs de poids (les vecteurs autour de chaque solution) [Brockhoff *et al.*, 2012].

3.5 Conclusion

Pour résoudre le problème d'optimisation de plans d'actions multi-objectif, nous avons proposé de combiner la méthode IBMOLS avec l'indicateur $R2$. Afin d'évaluer les performances de notre approche, nous avons effectué des études comparatives entre la méthode IBMOLS, utilisée avec deux indicateurs différents Epsilon et $R2$ et NSGA-II, un algorithme très populaire. Nos expérimentations sont réalisées sur des instances de tailles variées, avec 50 à 500 actions et de 2 à 8 objectifs.

En se basant sur ce travail, nous pouvons tirer deux conclusions principales. Tout d'abord, l'approche IBMOLS utilisée avec l'indicateur $R2$ s'avère être une méthode viable pour trouver des solutions de compromis de haute qualité pour le problème d'optimisation du plan d'actions dans les structures sociales et médico-sociales. En tant que tel, son intégration dans le logiciel d'aide à la décision *MSQualité* constitue un moyen précieux qui aide les décideurs à élaborer des plans d'actions plus efficaces leur permettant d'améliorer continuellement la qualité de leurs structures. Deuxièmement, grâce à la plate-forme partagée offerte par *MSQualité*, les décideurs de différentes structures peuvent collaborer entre eux et bénéficier mutuellement des expériences des uns et des autres.

Ce travail ouvre également la voie à de futures recherches. Tout d'abord, l'indicateur $R2$ offre certaines propriétés pour intégrer les préférences du décideur. En effet, nous pouvons utiliser la position du point de référence et l'orientation des vecteurs de poids de l'indicateur pour réduire l'espace de recherche et orienter le processus de recherche vers une région de l'espace objectif qui intéresserait plus le décideur. Cela contribuerait également à réduire le nombre de solutions générées et à faciliter ainsi le choix de la solution (donc du plan d'actions) à mettre en œuvre. Un autre moyen qui permet de réduire le nombre de solutions générées consiste à définir une relation de préférence entre les solutions non dominées dans l'espace objectif. Dans le chapitre 4, nous développons cet aspect plus en détail en utilisant la dominance de Lorenz au lieu de la dominance Pareto. Deuxièmement, dans nos expérimentations, nous avons remarqué que $R2$ -IBMOLS est plus coûteux en temps d'exécution que ϵ -IBMOLS et NSGA-II sur les grandes instances et son temps d'exécution dépend du nombre de vecteurs de poids utilisés. Ce

problème peut être atténué en prenant en compte le fait que le placement optimal des solutions selon l'indicateur $R2$ dépend uniquement d'un sous-ensemble des vecteurs de poids (une propriété de l'indicateur $R2$). Enfin, ce travail a montré que l'approche proposée fonctionne bien pour le POPAMO dont le nombre d'objectifs est limité à huit. Un travail également intéressant serait d'étudier $R2$ -IBMOLS sur des problèmes avec plus d'objectifs et une plus grande complexité et comparer ses performances avec d'autres approches multi-objectifs compétitives telles que HypE [Bader and Zitzler, 2011] et MOEA/D [Qingfu Zhang and Hui Li, 2007].

La dominance de Lorenz pour l'optimisation multi-objectif

Nous avons vu dans le chapitre précédent qu'à cause du grand nombre d'objectifs que peuvent avoir les plans d'actions, le nombre de solutions non dominées générées par les algorithmes testés est trop élevé pour que le décideur choisisse aisément le plan d'actions à mettre en œuvre. Dans ce chapitre, nous allons montrer qu'en utilisant la dominance de Lorenz à la place de la dominance de Pareto peut être un moyen efficace pour remédier à cet inconvénient. En effet, la dominance de Lorenz est un raffinement de la dominance de Pareto qui intègre l'équité dans l'optimisation multi-objectif lorsque les objectifs sont considérés comme égaux. En introduisant un ordre partiel parmi un ensemble de solutions non dominées, la dominance de Lorenz réduit la taille du front en conservant seulement les solutions dites "équitables". Dans ce chapitre, nous proposons une nouvelle version de *R2-IBMOLS* qui intègre ce concept dans la sélection des solutions à archiver et celles à présenter au décideur. Dans le cas où tous les objectifs ont la même importance pour le décideur, nous proposons d'intégrer la dominance de Lorenz dans deux autres algorithmes qui orientent la recherche vers le centre de l'espace objectif, où les solutions sont les plus équitables. Le travail présenté dans ce chapitre a fait l'objet d'un article en cours de soumission.

Sommaire

4.1	Introduction	78
4.2	De l'équité à l'optimisation équitable	79
4.2.1	Définitions	80
4.2.2	État de l'art	81
4.2.3	Dominance de Lorenz d'ordre infini	82
4.3	Approches proposées pour résoudre le POPAMO	84
4.3.1	IOLD-R2-IBMOLS	84
4.3.2	IOLD-BMOLS	87

4.3.3	IOLD-EA	87
4.4	Expérimentations	89
4.4.1	Comparaison avec L-NSGA-II	89
4.4.2	Comparaison avec NSGA-III	90
4.4.3	Protocole d'expérimentation	90
4.4.4	Résultats expérimentaux	91
4.5	Conclusion	96

4.1 Introduction

La plupart des métaheuristiques traitant des problèmes multi-objectifs visent à atteindre un certain équilibre entre la diversification et l'intensification, afin d'obtenir les solutions les plus optimales et les mieux réparties dans l'espace objectif. Cependant, dans certains problèmes pratiques, tels que l'optimisation de plans d'actions, cette stratégie n'est pas toujours la mieux adaptée pour deux raisons principales : i) le nombre de solutions non dominées générées par ces algorithmes peut s'avérer trop élevé pour que le décideur puisse choisir la solution qui lui convienne aisément. Particulièrement, quand on traite un problème de grande taille. ii) le décideur peut avoir une préférence sur une zone de l'espace objectif plutôt que d'autres. Par conséquent, il n'y a aucun intérêt à explorer des zones de recherche où les potentielles solutions qu'on y trouve seront sûrement rejetées par le décideur.

Dans le chapitre 3, pour résoudre le POPAMO, nous avons proposé *R2-IBMOLS* qui combine l'efficacité de la méthode IBMOLS [Basseur and Burke, 2007; Basseur *et al.*, 2012a] avec l'indicateur de qualité *R2* [Brockhoff *et al.*, 2012]. Nous avons montré que cette combinaison nous permet, à la fois, d'obtenir des solutions de bonne qualité (donc des plans d'actions plus efficaces) et d'intégrer facilement les préférences du décideur dans le processus de recherche d'IBMOLS. Malheureusement, nous avons également constaté que la méthode *R2-IBMOLS* souffre de deux inconvénients pour qu'elle soit intégrée telle quelle dans le logiciel *MSQualité* : i) le nombre de solutions non dominées est trop élevé (plusieurs milliers pour les instances de grande taille), ce qui rend le choix du plan d'actions à mettre en œuvre une tâche difficile pour le décideur. D'autant plus qu'un seul plan d'actions peut avoir plus de trois cents actions ; ii) le temps d'exécution moyen de *R2-IBMOLS* reste très élevé, particulièrement pour les instances de grande taille (plus d'une journée). Laisser tourner l'algorithme pendant toute une journée n'est pas un obstacle à son utilisation lorsque le plan d'actions est défini pour une durée de cinq ans. Mais si nous appliquons cette méthode dans un autre contexte ou pour un projet avec plan d'actions défini pour une durée plus courte, le temps d'exécution de l'algorithme devient problématique. Par ailleurs, le décideur pourrait bien accorder la même importance à tous les objectifs de son plan d'actions, auquel cas garantir une équité des solutions devient aussi important pour lui que l'optimalité du plan.

Le but du travail présenté dans ce chapitre est double : (1) réduire le temps d'exécution de l'algorithme *R2-IBMOLS* et le nombre de solutions non dominées qu'il génère, dans le cadre de l'intégration des préférences du décideur. (2) Garantir une équité des solutions proposées au décideur lorsque tous les objectifs sont considérés comme égaux. Pour le premier but, nous proposons d'intégrer dans *R2-IBMOLS* un paramètre T qui permet au décideur de contrôler le nombre de solutions gardées dans l'archive de *R2-IBMOLS* et le nombre de solutions qui

lui sont proposées à la fin de la recherche. Pour sélectionner ces solutions, nous utilisons la dominance de Lorenz d'ordre infini (*infinite order Lorenz dominance*) [Perny and Golden, 2010] qui définit un ordre partiel dans un ensemble de solutions non dominées. Pour le cas où tous les objectifs ont la même importance pour le décideur, la même dominance est utilisée dans deux autres algorithmes afin de guider la recherche vers le centre de l'espace objectif et garantir ainsi l'équité des solutions générées.

Nous montrons que l'utilisation de la dominance de Lorenz dans $R2$ -IBMOLS, à la place de la dominance de Pareto, nous permet de réduire de manière significative le nombre de solutions non dominées ainsi que le temps d'exécution de $R2$ -IBMOLS sans pour autant réduire son efficacité. En effet, la dominance de Lorenz (également appelée *dominance équitable*) est un raffinement de la dominance de Pareto, qui établit, sur un ensemble de solutions non dominées, une relation de préférence basée sur l'équité, conduisant ainsi à des solutions qui sont mieux distribuées qu'on appelle "front optimal de Lorenz". Nous entendons par "solutions mieux distribuées" les solutions les plus proches du point de référence lorsque les préférences du décideur sont connues et les solutions situées au centre de l'espace objectif lorsque ses préférences ne sont pas connues ou que tous les objectifs ont la même importance pour lui.

4.2 De l'équité à l'optimisation équitable

Le concept de l'équité a été étudié depuis le début du vingtième siècle dans divers domaines, tels que l'économie [Dalton, 1920; Sen, 1973], la formulation de modèles mathématiques abstraits [Steinhaus, 1949; Brams and Taylor, 1996] et les télécommunications [Ogryczak *et al.*, 2008]. L'équité est, essentiellement, un concept socio-économique de justice distributive qui vise à garantir l'égalité et l'impartialité dans la distribution des biens et des revenus [Ogryczak *et al.*, 2014]. Dans une répartition économique équitable, l'objectif est de garantir que 50% des biens sont possédés par 50% de la population, ou au moins minimiser l'écart avec cet objectif.

Au cours des dernières années, certaines approches intégrant le concept d'équité ont été proposées dans le domaine de l'optimisation multi-objectif (nous parlons d'une optimisation équitable). En effet, une optimisation équitable nécessite une impartialité dans le traitement des objectifs, en focalisant l'évaluation juste sur la distribution des solutions. Cela signifie que nous nous intéressons à un ensemble de solutions sans tenir compte de la solution qui prend une valeur spécifique pour un des objectifs. Pour arriver à cette fin, les méthodes d'optimisation équitable utilisent souvent la dominance de Lorenz. En outre, dans certains problèmes multi-objectifs, particulièrement les problèmes de grande taille, le nombre de solutions non dominées générées par les algorithmes d'optimisation peut être trop élevé, rendant le choix de la solution à mettre en œuvre une tâche difficile pour le décideur. L'utilisation de la dominance de Lorenz à la place de la dominance de Pareto peut être une bonne solution pour remédier à ce problème.

La dominance de Lorenz [Perny *et al.*, 2006] est un raffinement de la dominance de Pareto qui crée un ordre partiel (une relation de préférence) entre toutes les solutions non dominées, permettant ainsi, d'un côté, de réduire leur nombre et, d'un autre côté, de minimiser l'inégalité de traitement des différents objectifs. La figure 4.1 montre, pour une solution y d'un problème de minimisation, la différence entre les espaces objectifs dominés par y (couleur grise) dans le cas d'une dominance de Pareto et d'une dominance de Lorenz. Étant donné que l'espace de recherche résultant est réduit, la dominance de Lorenz permet de trouver des solutions plus efficaces et bien réparties que la dominance de Pareto. Ce fait est largement décrit dans [Dugardin *et al.*, 2010a].

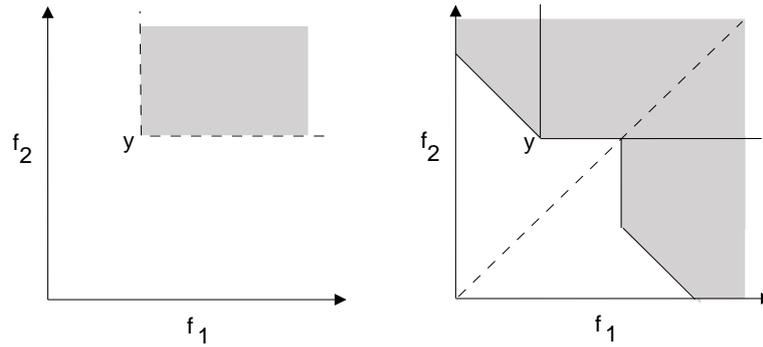


FIGURE 4.1 – Espace objectif dominé par y dans le cas d'une dominance de Pareto (gauche) et d'une dominance de Lorenz (droite).

4.2.1 Définitions

Avant de définir formellement la dominance de Lorenz, il est indispensable d'introduire d'abord certains concepts utiles.

4.2.1.1 La relation de préférence

Comme cela a été indiqué dans [Perny and Golden, 2010], afin de choisir entre deux solutions non dominées, nous devons définir une relation de préférence \succsim sur les vecteurs objectifs. La relation $y \succsim z$ signifie que la solution correspondante au vecteur y est préférable à la solution correspondante au vecteur z . De plus, pour formaliser le fait que tous les objectifs sont traités de manière équivalente, nous définissons l'axiome suivant :

Définition 16. Pour un vecteur $y \in Y$ et une permutation quelconque π de $\{1, \dots, m\}$, $(y_{\pi(1)}, \dots, y_{\pi(m)}) \sim (y_1, \dots, y_m)$, où \sim est une relation d'indifférence définie comme étant la partie symétrique de \succsim .

4.2.1.2 Le principe de transfert

Pour qu'une optimisation soit équitable, elle doit satisfaire le *principe de transfert* de Pigou-Dalton [Sen, 1973]. Ce principe stipule qu'un transfert d'une petite quantité d'un vecteur vers un autre vecteur, relativement moins bon, tout en préservant la moyenne des valeurs objectifs, pourrait produire un vecteur mieux distribué dans l'espace objectif. En tant que propriété de la relation de préférence \succsim , le principe de transfert est défini comme suit :

Définition 17. Pour un vecteur $y \in Y$ tel que $y_i > y_j$ et tout ϵ tel que $0 < \epsilon < y_i - y_j$, $y - \epsilon e_i + \epsilon e_j \prec y$ où e_i et e_j sont respectivement le i^{me} et le j^{me} vecteur unitaire.

Par exemple, les deux vecteurs $y = (7, 3, 4)$ et $z = (5, 3, 6)$ sont tous les deux non dominés (au sens de la dominance de Pareto). Le principe de transfert stipule que z est préférable à y ($z \succsim y$) parce qu'il existe un transfert de taille $\epsilon = 2$ pour passer de z à y .

4.2.1.3 Vecteur de Lorenz généralisé

Pour identifier ces vecteurs qui peuvent être comparés en utilisant le principe du transfert, nous rappelons la définition du *vecteur de Lorenz généralisé*, sur lequel la dominance de Lorenz est basée.

Définition 18. Pour tout vecteur $y \in Y$, le vecteur de Lorenz généralisé de y est le vecteur $L(y) = (y_1, y_1 + y_2, \dots, y_1 + y_2 + \dots + y_m)$, où $y_1 \geq y_2 \geq \dots \geq y_m$ représentent les composantes de y ordonnées dans un ordre descendant (ascendant pour un problème de maximisation). La j^{me} composante de $L(y)$ est $L(y)_j = \sum_{i=1}^j y_i$.

4.2.1.4 Dominance de Lorenz généralisée

A présent, nous pouvons définir la dominance de Lorenz généralisée comme suit :

Définition 19. La dominance de Lorenz généralisée est définie par :

$$\forall y, z \in Y, y \prec_L z \Leftrightarrow L(y) \prec L(z).$$

Avec cette définition, on considère qu'un vecteur y est *L-dominé* ("L" pour Lorenz) s'il existe un autre vecteur $z \in Y$ tel que $z \prec_L y$. Sinon, y est dit *L-non dominé*.

Afin d'établir le lien entre la dominance de Lorenz généralisée et le principe de transfert, nous rappelons le théorème de Chong [Chong, 1976] :

Théorème 1. Pour tous vecteurs y et $z \in Y$, si $y \prec z$ ou y est obtenu à partir de z par le transfert de Pigou-Dalton, alors $y \prec_L z$. Par conséquent, si $y \prec_L z$ alors il existe une séquence de transferts admissibles et/ou d'améliorations Pareto pour transformer z en y .

4.2.2 État de l'art

Initialement, la dominance de Lorenz a été proposée par Kostreva et Ogryczak [Kostreva and Ogryczak, 1999b] en 1999 pour résoudre des problèmes linéaires d'optimisation multi-critère. Puis, ils l'ont appliquée à d'autres types de problèmes, tels que les problèmes d'allocation [Kostreva and Ogryczak, 1999a; Ogryczak, 2000a] et de portfolio [Ogryczak, 2000b]. Dans la littérature, il existe quelques études qui développent des méthodes d'optimisation multi-objectif en utilisant la dominance de Lorenz mais pas autant que celles basées sur la dominance de Pareto. La liste des travaux que nous avons pu recenser est présentée ci-dessous.

- en 2004, Kostreva et al. [Kostreva et al., 2004] ont proposé l'utilisation de la moyenne pondérée ordonnée (*Ordered Weighted Averaging*) pour obtenir des solutions équitablement efficaces aux problèmes d'optimisation multi-critères, linéaires et non linéaires.
- en 2006, Perny et al. [Perny et al., 2006] ont introduit un nouveau cadre de travail pour définir une robustesse dans des problèmes combinatoires. Pour ce faire, ils ont utilisé la dominance de Lorenz pour comparer des solutions dans plusieurs scénarios différents. Ils ont également proposé une nouvelle approche pour rechercher des solutions efficaces, au sens de la dominance de Lorenz, pour des problèmes robustes du plus court chemin et des problèmes d'arbres couvrants. Dans leur approche, ils ont introduit dans la définition de la dominance Lorenz la moyenne pondérée ordonnée comme mesure de robustesse.

- en 2010, Dugardin et al. [Dugardin et al., 2010b] ont publié une version de NSGA-II [Deb et al., 2002] qu'ils ont appelé "*L-NSGA-II*", où ils ont remplacé la dominance de Pareto par la dominance de Lorenz dans la définition des fronts. Ils ont comparé L-NSGA-II à NSGA-II et SPEA2 [Zitzler et al., 2012] sur deux groupes de 24 instances d'un problème d'ordonnancement bi-objectif, maximisant le taux d'utilisation des machines et minimisant le temps maximal de réalisation des tâches. Leurs résultats montrent que les meilleures solutions sont toujours obtenues avec L-NSGA-II. Dans un deuxième jeu de tests, ils ont comparé L-NSGA-II à un front optimal obtenu avec une méthode exacte sur 16 instances de petites tailles. Les résultats de L-NSGA-II sont très proches de ceux obtenus avec la méthode exacte.

La même année, Perny et Golden [Perny and Golden, 2010] ont introduit la notion de dominance de Lorenz d'ordre infini (*infinite order Lorenz dominance*). Ils ont montré qu'en utilisant la moyenne pondérée ordonnée, il est possible de reformuler le problème de dominance sous forme d'un problème d'optimisation mono-objectif. En utilisant la formulation qu'ils ont proposée, ils ont montré l'efficacité de leur approche sur plusieurs instances du problème d'affectation de papiers. Justement, grâce à l'efficacité de cette approche, les algorithmes que nous proposons dans ce chapitre sont basés sur l'utilisation de cette formulation de la dominance de Lorenz d'ordre infini, que nous détaillons dans la prochaine section.

- en 2011, Dugardin et al. [Dugardin et al., 2011] ont utilisé la dominance de Lorenz dans une nouvelle méthode d'optimisation à colonie de fourmis pour résoudre un problème d'ordonnancement bi-objectif, minimisant à la fois le temps de réalisation des tâches et le retard global de réalisation. Moghaddam et al. [Moghaddam et al., 2011] ont intégré la dominance de Lorenz dans un algorithme de recuit simulé bi-objectif (L-MOSA). Dans leurs expérimentations, ils ont appliqué L-MOSA également à un problème d'ordonnancement à une machine en considérant la minimisation de deux objectifs : la somme pondérée des temps de réalisation des tâches acceptées et la somme des pénalités relatives aux tâches rejetées. En comparant, sur plusieurs instances, les performances de L-MOSA à un recuit simulé multi-objectif utilisant la dominance de Pareto, ils ont montré que l'application de la dominance de Lorenz a réduit considérablement le nombre de solutions non dominées (une réduction de 70% pour certaines instances). De plus, dans certains cas, il était possible de trouver de meilleures solutions que celles obtenues avec la dominance de Pareto en moins de temps de calcul.
- plus récemment (2015), pour générer des solutions Lorenz optimales, Galand et Lust [Galand and Lust, 2015a; Galand and Lust, 2015b] ont proposé deux adaptations efficaces de la méthode à deux phases classique. Dans leurs expérimentations, ils ont appliqué la première méthode [Galand and Lust, 2015a] à un problème bi-objectif de couverture par ensembles et à un problème bi-objectif du plus court chemin. La seconde méthode est appliquée [Galand and Lust, 2015b] à un problème de sac-à-dos bi-agent. Les deux méthodes ont obtenu de bons résultats comparées à d'autres méthodes de la littérature.

4.2.3 Dominance de Lorenz d'ordre infini

Lorsque deux vecteurs objectifs y et z ne peuvent être comparés avec la dominance de Pareto (ils sont tous les deux non dominés), nous calculons les vecteurs de Lorenz généralisés

correspondants $L(y)$ et $L(z)$ puis nous comparons $L(y)$ à $L(z)$ afin de définir une préférence entre les deux vecteurs. Mais, il se peut qu'il n'y ait pas non plus de dominance entre $L(y)$ et $L(z)$. Dans ce cas, l'indétermination pourrait être résolue en comparant $L^2(y) = L(L(y))$ à $L^2(z) = L(L(z))$. Pour réduire l'incomparabilité, ce processus peut être itéré à des degrés plus élevés. Perny et Golden [Perny and Golden, 2010] ont introduit la notion de vecteur de Lorenz d'ordre k correspondant à $L^k(y)$ et défini comme suit :

$$L^k(y) = \begin{cases} y & \text{si } k = 0 \\ L(L^{k-1}(y)) & \text{si } k \geq 1 \end{cases} \quad (4.1)$$

Ainsi, nous pouvons définir la dominance de Lorenz d'ordre k comme suit :

$$\forall y, z \in Y, y \prec_L^k z \Leftrightarrow L^k(y) \prec L^k(z) \quad (4.2)$$

A partir des définitions 4.1 et 4.2, Perny et Golden ont proposé la dominance stricte de Lorenz d'ordre infini (L^∞ -dominance stricte), définie par :

$$\prec_L^\infty = \bigcup_{k \geq 1} \prec_L^k \quad (4.3)$$

E pour appliquer la \prec_L^∞ à deux vecteurs $y, z \in Y$, ils ont proposé l'algorithme 5 suivant :

Algorithm 5 L^∞ -dominance stricte

```

 $y' \leftarrow y$ 
 $z' \leftarrow z$ 
while not( $y' \prec z'$  or  $z' \prec y'$ ) do
   $y' \leftarrow L(y')$ 
   $z' \leftarrow L(z')$ 
end while
if ( $y' \prec z'$ ) then
   $y \prec_L^\infty z$ 
end if
if ( $z' \prec y'$ ) then
   $z \prec_L^\infty y$ 
end if

```

L'algorithme 5 tente de faire un choix entre des vecteurs qui ne sont pas discriminés par la dominance de Lorenz. Cependant, rien ne prouve qu'il se termine pour n'importe quelle paire de vecteurs. Pour résoudre ce problème, Perny et Golden ont reformulé la recherche de solutions non dominées réalisée par l'algorithme 5 sous forme d'un problème d'optimisation mono-objectif. Ils ont fourni une définition mathématique stricte de la L^∞ -dominance (définition 20), ce qui rend possible la comparaison de n'importe quelle paire de vecteurs.

Définition 20. La L^∞ -dominance stricte a une représentation numérique stricte utilisant la moyenne pondérée ordonnée suivante :

$$\mathcal{W}(y) = \sum_{k=1}^m \sin\left(\frac{(m+1-k)\pi}{2m+1}\right) y_k \quad (4.4)$$

Cette représentation est donnée par la propriété suivant :

$$\forall y, z \in Y, y \prec_L^\infty z \iff \mathcal{W}(y) < \mathcal{W}(z) \quad (4.5)$$

Pour la démonstration de cette propriété ainsi que l'équation 4.4, consulter [Perny and Golden, 2010].

A présent, pour comparer n'importe quelle paire de vecteurs y et z avec la L^∞ -dominance stricte, il n'est plus nécessaire d'exécuter l'algorithme 5. Il suffit de calculer et de comparer $\mathcal{W}(y)$ et $\mathcal{W}(z)$. Lorsque $\mathcal{W}(y) \neq \mathcal{W}(z)$, le vecteur ayant la plus petite valeur pour \mathcal{W} L^∞ -domine strictement l'autre. Par conséquent, l'algorithme 5 s'arrêterait sûrement après un certain nombre d'itérations. Et lorsque $\mathcal{W}(y) = \mathcal{W}(z)$, il n'y a pas de relation de dominance entre les deux vecteurs quel que soit l'ordre auquel on applique la dominance de Lorenz. Dans ce cas, l'algorithme 5 ne s'arrêterait jamais.

L'équation 4.4 présente deux avantages principaux. Premièrement, il est très facile de l'intégrer dans un algorithme évolutionnaire. En effet, avec quelques paramètres, elle permet de sélectionner des individus à un haut niveau de comparaison, comme elle permet de détecter une relation de dominance (ou de préférence) même lorsque la différence est minimale entre les individus. Deuxièmement, l'utilisation de l'équation 4.4 dans un processus de sélection d'un algorithme évolutionnaire permet une réduction significative du temps d'exécution de l'algorithme, car il n'est plus nécessaire de vérifier la dominance entre les individus de la population. Cette vérification est directement intégrée dans la définition de l'équation elle-même.

4.3 Approches proposées pour résoudre le POPAMO

Pour résoudre le problème d'optimisation de plans d'actions multi-objectifs, nous proposons, dans cette section, d'utiliser la L^∞ -dominance stricte donnée par l'équation 4.4 dans trois algorithmes différents. Dans le premier algorithme, elle est utilisée pour établir une préférence entre les solutions générées par l'algorithme $R2$ -IBMOLS présenté dans le chapitre 3. En plus de la réduction du nombre de solutions qui sont présentées au décideur, elle permet aussi de réduire considérablement le temps d'exécution de $R2$ -IBMOLS, un inconvénient de l'algorithme non négligeable. Dans les deux autres méthodes proposées, la L^∞ -dominance stricte est utilisée comme fonction d'évaluation du fitness des solutions. Dans les trois algorithmes proposés, nous utilisons *IOLD* pour *infinite order Lorenz dominance*.

Nous rappelons que, dans ce qui suit, P désigne la population courante de l'algorithme et que toutes les valeurs objectifs sont normalisées avec la même méthode que celle présentée dans la section 3.4.2 du chapitre 3.

4.3.1 IOLD-R2-IBMOLS

L'algorithme *IOLD-R2-IBMOLS* (algorithme 6) est une combinaison des performances de l'algorithme $R2$ -IBMOLS, qui offre deux moyens pour intégrer les préférences du décideur (point de référence et un ensemble de vecteurs de poids), avec la L^∞ -dominance stricte qui permet une comparaison entre les solutions non dominées à un degré élevé de dominance. En effet, nous avons vu dans le chapitre 3 que malgré que $R2$ -IBMOLS fournisse des solutions de bonne qualité, leur nombre est tellement élevé qu'il est difficile de l'utiliser en pratique comme tel. L'idée développée dans *IOLD-R2-IBMOLS* consiste à ne garder dans l'archive

qu'un certain nombre T de solutions non dominées. Les T solutions à garder sont sélectionnées suivant leurs valeurs pour \mathcal{W} donnée par l'équation 4.4.

IOLD-R2-IBMOLS commence par sélectionner les solutions non dominées de la population initiale P . Pour ce faire, nous utilisons la fonction *infiniteOrderLorenzNonDominatedSolutions* (ayant deux paramètres P et T). Suivant la valeur de chaque solution de P pour \mathcal{W} , cette fonction sélectionne, au plus, les T premières solutions L-non dominées de P (voir algorithme 7). Pour choisir les solutions les plus proches du point de référence, \mathcal{W} est utilisée avec les mêmes poids utilisés pour définir le point de référence z^* (ces poids sont donnés par le décideur). Le paramètre T permet au décideur de contrôler le nombre de solutions stockées dans l'archive et le nombre de solutions retournées par l'algorithme. Dans nos expérimentations, nous avons utilisé la même valeur de T pour les deux, mais on peut bien sûr l'utiliser avec deux valeurs différentes. Ensuite, le fitness de chaque solution x de P est évalué avec l'indicateur $R2$, de la même manière que dans *R2-IBMOLS* (formule 4.6)

$$I_{R2}(x, \Lambda, z^*) = I_{R2}(P, \Lambda, z^*) - I_{R2}(P \setminus \{x\}, \Lambda, z^*) \quad (4.6)$$

Ensuite, l'algorithme *IOLD-R2-IBMOLS* applique une étape de recherche locale pour explorer le voisinage de chaque solution de P . Un voisin est accepté si sa valeur pour l'indicateur $R2$ est meilleure que la pire solution dans P . La génération du voisinage d'une solution s'arrête lorsque tout son voisinage est exploré ou une fois qu'un voisin améliorant est trouvé (premier voisin qui améliore la qualité de P par rapport à l'indicateur $R2$). Pour les mêmes raisons évoquées pour *R2-IBMOLS* (voir la section 3.4.2 du chapitre 3), le voisinage n'est pas entièrement exploré. La recherche locale de *IOLD-R2-IBMOLS* est terminée lorsque l'archive E de solutions L-non dominées n'a reçu aucune nouvelle solution durant toute une étape de recherche locale complète.

La principale différence entre *IOLD-R2-IBMOLS* et *R2-IBMOLS* est l'introduction de la fonction *infiniteOrderLorenzNonDominatedSolutions*. En effet, d'une part, cette fonction nous permet de réduire le nombre de solutions non dominées à présenter au décideur en introduisant le paramètre T et en veillant à ce que les solutions sélectionnées soient les mieux distribuées. D'autre part, la limitation de la taille de l'archive permet de réduire de manière significative le temps d'exécution de l'algorithme (voir les résultats expérimentaux présentés dans la section 4.4.4).

Dans la version itérée de *IOLD-R2-IBMOLS* (algorithme 8), une approximation courante AS est maintenue et potentiellement mise à jour après chaque itération de l'algorithme. Après chaque phase de recherche locale, la fonction *initWalk* crée une nouvelle population initiale pour la prochaine itération de *IOLD-R2-IBMOLS*. Même si la population initiale est entièrement créée d'une manière aléatoire pour la première itération, lorsque le processus de recherche locale est itéré, la fonction *initWalk* génère une nouvelle population P pour la prochaine itération en utilisant des informations sur les bonnes solutions obtenues lors des itérations précédentes. En effet, la fonction *initWalk* applique des mutations aléatoires sur N solutions sélectionnées aléatoirement de AS (chaque solution de AS ne peut être sélectionnée qu'une seule fois). Pour chaque solution sélectionnée, une mutation est appliquée avec une probabilité de $1/n$ (où n correspond au nombre d'actions). La solution mutée est ajoutée à la population si elle n'y est pas déjà présente et uniquement si elle vérifie la contrainte budgétaire β et les seuils définis pour les objectifs. Quand $|AS| < N$, toutes les solutions de AS sont sélectionnées et les individus manquants de P sont complétés avec de nouvelles solutions générées aléatoirement. Après chaque itération de *IOLD-R2-IBMOLS*, la fonction *infiniteOrderLorenzNonDomina-*

Algorithm 6 *IOLD-R2-IBMOLS* algorithm

Require: P (initial population of size N); I_{R2} (binary indicator); T (size of the archive)**Ensure:** E (approximation set ($|E| \leq T$)) $E \leftarrow \text{infiniteOrderLorenzNonDominatedSolutions}(P, T)$ */* Compute fitness values of individual x in P */***for all** $x \in P$ **do** $\text{fit}(x) = R2(P, \Lambda, z^*) - R2(P \setminus \{x\}, \Lambda, z^*)$ **end for***/* Local search step */***for all** $x \in P$ **do****for all** $j \in \{1, \dots, m\}$ **do**updateMinMax(j) */* Update minimal m_j and maximal M_j (for objective functions normalization) */***end for****repeat** $x^* \leftarrow$ one unexplored neighbor of x $P \leftarrow P \cup x^*$ $\text{fit}(x^*) = R2(P, \Lambda, z^*) - R2(P \setminus \{x^*\}, \Lambda, z^*)$ */* Compute x^* fitness */*Update all $z \in P$ fitness values $w \leftarrow$ the worst individual in P remove w from P Update all $z \in P$ fitness values**until** all neighbors are explored or $w \neq x^*$ **end for** $E \leftarrow \text{infiniteOrderLorenzNonDominatedSolutions}(E \cup P, T)$ **if** E does not change **then**return E **else**

perform another local search step

end if

Algorithm 7 infiniteOrderLorenzNonDominatedSolutions(P, T) procedure

Require: P (set of solutions); T (number of L-nondominated solutions to be selected)**Ensure:** S (set of L-nondominated solutions ($|S| \leq T$))**for all** $x \in P$ **do**Compute $\mathcal{W}(x)$ **end for**Sort solutions of P in increasing order of \mathcal{W} values */* for maximization problem, sort in non-increasing order */* $S' \leftarrow T$ first solutions in P $S \leftarrow$ L-nondominated solutions in S'

tedSolutions ordonne les nouvelles solutions générées avec celles déjà contenues dans l'archive *AS*, puis sélectionne les *T* meilleures solutions.

Algorithm 8 Iterated *IOLD-R2-IBMOLS* algorithm

Require: *N* (population size); *T* (size of the archive)

Ensure: *AS* (approximation set)

AS $\leftarrow \emptyset$

while stopping condition not achieved **do**

P $\leftarrow \text{initWalk}(AS, N)$

E $\leftarrow \text{IOLD-R2-IBMOLS}(P, T)$ /* Local search step */

AS $\leftarrow \text{infiniteOrderLorenzNonDominatedSolutions}(AS \cup E, T)$

end while

Return *AS*

4.3.2 IOLD-BMOLS

Comme cela a été mentionné dans la section 4.2.3, nous pouvons utiliser l'équation 4.4 pour comparer deux solutions *y* et *z* à un niveau élevé de dominance de Lorenz et la solution ayant le plus petit score (le plus élevé dans le cas d'une maximisation) pour \mathcal{W} est préférable à l'autre. Ainsi, dans l'algorithme *IOLD-BMOLS* (algorithme 9), nous utilisons \mathcal{W} pour évaluer le fitness de chaque solution *x* de la population *P* et dans le processus de sélection.

Comme *IOLD-R2-IBMOLS*, *IOLD-BMOLS* utilise la fonction *infiniteOrdredLorenzNonDominatedSolutions* (algorithme 7) pour sélectionner toutes les solutions L-non dominées d'une population initiale donnée *P*. Ensuite, il calcule la valeur de fitness de chaque solution *x* dans *P*, donnée par \mathcal{W} telle qu'elle est définie dans l'équation 4.4. Une étape de recherche locale est appliquée dans *IOLD-BMOLS*, comme dans *IOLD-R2-IBMOLS*. La différence entre les deux algorithmes réside dans la suppression du pire individu de la population courante. Dans *IOLD-R2-IBMOLS*, le pire individu est identifié par rapport à sa valeur pour l'indicateur *R2*, alors que dans *IOLD-BMOLS*, l'indicateur *R2* est complètement remplacé par \mathcal{W} . Par conséquent, l'individu à supprimer est sélectionné par rapport à son score pour \mathcal{W} .

IOLD-BMOLS a également une version itérée (algorithme 10) avec une structure similaire à la version itérée de *IOLD-R2-IBMOLS*. Nous remplaçons simplement *IOLD-R2-IBMOLS* par *IOLD-BMOLS* dans l'étape de recherche locale.

4.3.3 IOLD-EA

Dans cette section, nous proposons d'utiliser \mathcal{W} dans le processus de sélection d'un algorithme évolutionnaire basé sur *IOLD* appelé *IOLD-EA* (algorithme 11). Dans cet algorithme, la formule \mathcal{W} est utilisée pour calculer la valeur de fitness des solutions de la population courante et à explorer le voisinage de chaque solution à chaque nouvelle génération de l'algorithme *IOLD-EA*. En effet, les algorithmes évolutionnaires classiques basés sur le calcul de la dominance de Pareto dans le processus de sélection, tel que NSGA-II [Deb et al., 2002] et NSGA-III [Deb and Jain, 2014], pourrait coûter cher en temps de calcul (particulièrement sur des problèmes de grande taille). Par contre, le fait d'utiliser \mathcal{W} dans le processus de sélection nous permet de réduire considérablement le temps d'exécution de l'algorithme (puisque nous n'avons plus besoin de vérifier la dominance entre les individus de la population).

Algorithm 9 IOLD-BMOLS algorithm

Require: P (initial population of size N); T (size of the archive)**Ensure:** E (Pareto approximation set ($|E| \leq T$)) $E \leftarrow \text{infiniteOrderLorenzNonDominatedSolutions}(P, T)$ /* Calculate fitness values of individual x in P */**for all** $x \in P$ **do** $\text{fit}(x) = \mathcal{W}(x)$ **end for**

/* Local search step */

for all $x \in P$ **do****for all** $j \in \{1, \dots, m\}$ **do**updateMinMax(j) /* Update minimal m_j and maximal M_j (for objective functions normalization) */**end for****repeat** $x^* \leftarrow$ one unexplored neighbor of x $P \leftarrow P \cup x^*$ $\text{fit}(x^*) = \mathcal{W}(x^*)$ /* Compute x^* fitness */ $w \leftarrow$ the worst individual in P /* Individual with the smallest value of \mathcal{W} */remove w from P **until** all neighbors are explored or $w \neq x^*$ **end for** $E \leftarrow \text{infiniteOrderLorenzNonDominatedSolutions}(E \cup P, T)$ **if** E does not change **then**return E **else**

perform another local search step

end if

Algorithm 10 Iterated IOLD-BMOLS algorithm

Require: N (population size); T (size of the archive)**Ensure:** AS (approximation set) $AS \leftarrow \emptyset$ **while** stopping condition not achieved **do** $P \leftarrow \text{initWalk}(AS, N)$ $E \leftarrow \text{IOLD-BMOLS}(P, T)$ /* Local search step */ $AS \leftarrow \text{infiniteOrderLorenzNonDominatedSolutions}(AS \cup E, T)$ **end while**Return AS

IOLD-EA est défini par trois paramètres : une population initiale P de taille N , un nombre de générations à effectuer $nbGen$, et le nombre de solutions T à retourner au décideur. Ensuite, pour une génération t et une population parent P_t , la fonction $child(P_t)$ crée une population enfant Q_t de N individus, par recombinaison et mutation de P_t . Puis, la formule \mathcal{W} est utilisée pour calculer la valeur de fitness de chaque individu de la population R_t contenant les individus de la population parent et ceux de la population enfant, $R_t = P_t \cup Q_t$ (de taille $2N$). Pour sélectionner les individus de la génération P_{t+1} , la fonction *infiniteOrdredLorenzSelection* (algorithme 7) sélectionne les N meilleurs individus de R_t en fonction de leurs scores pour \mathcal{W} . A la fin de l'exécution de l'algorithme, la fonction *infiniteOrderLorenzNonDominatedSolutions* retourne au décideur les T meilleures solutions de la dernière génération P_{nbGen} . Une description détaillée de *IOLD*-EA est fournie dans l'algorithme 11.

Algorithm 11 *IOLD*-EA algorithm

Require: P (initial population of size N); $nbGen$ (number of generations); T (number of returned solutions)

Ensure: E (Pareto approximation set)

$t = 1$

$P_t = P$

while $t \leq nbGen$ **do**

$Q_t = child(P_t)$

$R_t = P_t \cup Q_t$

for all $x \in R_t$ **do**

$fit(x) = \mathcal{W}(x)$

end for

$t = t + 1$

$P_t \leftarrow infiniteOrdredLorenzSelection(R_t, N)$

end while

$E \leftarrow infiniteOrdredLorenzSelection(P_t, T)$

4.4 Expérimentations

Dans cette section, nous présentons un ensemble d'expérimentations visant à permettre une comparaison des résultats obtenus avec les algorithmes que nous avons proposés (*IOLD*-R2-IBM-OLS, *IOLD*-BMOLS et *IOLD*-EA) avec trois autres algorithmes de la littérature : L-NSGA-II [Dugardin *et al.*, 2010b], NSGA-III [Deb and Jain, 2014] et R2-IBMOLS [Chabane *et al.*, 2017]. C'est également suivant les résultats de ces expérimentations que sont choisis les algorithmes qui seront intégrés dans le logiciel *MSQualité*.

4.4.1 Comparaison avec L-NSGA-II

L-NSGA-II [Dugardin *et al.*, 2010b] a la même conception que NSGA-II. La seule différence réside dans le fait que L-NSGA-II utilise la dominance de Lorenz dans le classement des solutions sur des fronts alors que NSGA-II utilise la dominance de Pareto. Nous avons choisi cet algorithme pour l'évaluation de nos résultats parce qu'il est basé aussi sur la dominance

de Lorenz et il a permis d'obtenir de bons résultats sur d'autres problèmes [Dugardin *et al.*, 2010b].

4.4.2 Comparaison avec NSGA-III

NSGA-III [Deb and Jain, 2014] est une autre version de NSGA-II avec des changements significatifs dans son mécanisme de sélection. Le maintien de la diversité parmi les membres de la population dans NSGA-III est assuré par la définition et la mise à jour adaptée d'un certain nombre de points de référence bien répartis. Tandis que NSGA-II utilise la distance de *crowding* pour sélectionner les solutions du dernier front qui maximisent la diversité de la population. Dans NSGA-III, l'opérateur *crowding* de NSGA-II est remplacé par une approche basée sur des points de référence. Pour identifier les fronts de NSGA-III, les distances qui séparent les solutions de la population de chaque point de référence sont d'abord calculées. Puis pour chaque point de référence, les solutions sont ordonnées dans un ordre ascendant suivant leurs distances du point de référence considéré. Ainsi, chaque solution la plus proche d'au moins un point de référence est assignée au front 1, puis celles qui sont classées deuxièmes par rapport à au moins un point de référence sont assignées au front 2, et ainsi de suite. Comme NSGA-II, NSGA-III est un algorithme qui a montré son efficacité sur beaucoup de problèmes multi-objectifs, d'où notre choix pour cet algorithme comme élément de comparaison.

4.4.3 Protocole d'expérimentation

En utilisant les mêmes instances décrites dans la section 3.4.3.1 du chapitre 3, nous avons testé *IOLD-R2-IBMOLS*, *IOLD-BMOLS*, *IOLD-EA*, *R2-IBMOLS*, L-NSGA-II et NSGA-III avec les paramètres suivants : pour L-NSGA-II, NSGA-III et *IOLD-EA*, nous avons utilisé une population de 100 individus et une probabilité de mutation de $1/n$ (où n correspond au nombre d'actions). Pour *IOLD-R2-IBMOLS*, *IOLD-BMOLS* et *R2-IBMOLS*, nous avons utilisé les versions itérées avec une population de taille fixe de 10 individus. Le choix d'une population de 10 solutions pour ces algorithmes est basé sur les recommandations données dans [Basseur *et al.*, 2012a], qui montre que IBMOLS converge mieux avec une population de petite taille. Pour chaque algorithme, la population initiale est générée d'une manière aléatoire, en s'assurant que ces contraintes sont bien respectées : i) le coût de réalisation de chaque solution ne dépasse pas le budget β alloué au plan d'actions ; ii) toute solutions x de la population doit améliorer tous les objectifs du plan ($f_j(x) > 0 \forall j \in \{1, \dots, m\}$). La procédure de génération de la population initiale est décrite dans l'algorithme 3 du chapitre 3.

De plus, la stratégie de sélection suivante est adoptée : un seul voisin aléatoire de chaque individu de la population courante est sélectionné pour être membre de la population enfant dans L-NSGA-II, NSGA-III et *IOLD-EA* ou intégrer la population courante de *IOLD-BMOLS*, *IOLD-R2-IBMOLS* et *R2-IBMOLS*. La même procédure de génération du voisinage est utilisée pour tous les algorithmes : le i^{me} voisin de la solution $x = (a_1, a_2, \dots, a_n)$ est obtenu en modifiant la valeur de a_i (remplacer 0 par 1 et vice versa). Uniquement les voisins vérifiant la contrainte de budget β et les seuils des objectifs sont considérés comme candidats (quand une des contraintes n'est pas vérifiée, un nouveau voisin est généré). Pour toutes les instances, le budget β est fixé à un million et les seuils des objectifs sont fixés à 1 (c'est-à-dire $c_j \geq 1 \forall j \in \{1, \dots, m\}$).

Pour calculer la valeur de l'indicateur *R2* de chaque solution x de *IOLD-R2-IBMOLS*

et *R2-IBMOLS*, l'équation 4.6 est utilisée avec le point de référence $z^* = (2, 2, \dots, 2)$ et 100 vecteurs de poids fixes ($|\Lambda| = 100$), uniformément répartis dans l'espace objectif. Pour générer ces vecteurs, nous avons utilisé l'algorithme de Phan et al. proposé dans [Phan and Junichi, 2013] (voir l'algorithme 4 du chapitre 3). Cependant, dans [Phan and Junichi, 2013], le premier vecteur de poids est généré de manière aléatoire, dans nos expérimentations, il est défini suivant le point de référence z^* , qui est également utilisé dans NSGA-III. Nous notons que les points de référence de NSGA-III pourraient être prédéfinis de manière structurée ou fournis par le décideur et que leur nombre pourrait être important. Dans nos expérimentations, nous nous intéressons uniquement à la région de l'espace objectif donnée par z^* . Enfin, nous avons fixé le paramètre T de *IOLD-BMOLS*, *IOLD-R2-IBMOLS* et *IOLD-EA* à 10 solutions.

Pour la mesure de performance, nous avons effectué 30 exécutions de chacun des algorithmes sur chaque instance. La même condition d'arrêt est utilisée pour tous les algorithmes, qui est de $200 * n * m$ recherches locales pour *IOLD-BMOLS*, *IOLD-R2-IBMOLS* et *R2-IBMOLS* et $200 * n * m$ générations pour *IOLD-EA*, L-NSGA-II et NSGA-III (où n et m correspondent respectivement au nombre d'actions et au nombre d'objectifs). Les expérimentations sont réalisées sur une machine Intel core i5-2400 de 2x3.10Ghz et 16Gb de RAM.

4.4.4 Résultats expérimentaux

Dans cette section, nous présentons les résultats expérimentaux obtenus sur les instances décrites dans la section 3.4.3.1 du chapitre 3 avec le protocole expérimental décrit précédemment.

Les figures 4.2 et 4.3 montrent les ensembles d'approximations obtenus avec les algorithmes proposés (*IOLD-BMOLS*, *IOLD-R2-IBMOLS* et *IOLD-EA*) sur 30 exécutions, comparés à NSGA-III, L-NSGA-II et *R2-IBMOLS* sur trois instances représentatives : une instance avec 2 objectifs et 50 ou 500 actions (figure 4.2) et une autre avec 4 objectifs et 500 actions (figure 4.3).

D'après les figures 4.2 et 4.3, il est clair que toutes les solutions obtenues avec *IOLD-BMOLS*, *IOLD-R2-IBMOLS* et *IOLD-EA* ne sont pas dominées par celles obtenues par NSGA-III, L-NSGA-II ou *R2-IBMOLS*. De plus, par rapport à la direction de préférence (flèche grise) donnée par le point de référence $z^* = (2, \dots, 2)$, les solutions obtenues avec *IOLD-BMOLS*, *IOLD-R2-IBMOLS* et *IOLD-EA* sont mieux distribuées dans l'espace bi-objectif de la figure 4.2 et aussi mieux distribuées pour trois sur quatre objectifs de la figure 4.3.

Pour la mesure de performance, nous avons choisi la distance euclidienne du point de référence z^* appliqué aux ensembles d'approximations obtenus par les différents algorithmes sur un ensemble de 30 instances de différentes tailles. Nous avons calculé les distances minimales, médianes et maximales du point de référence aux ensembles d'approximation obtenus par chaque algorithme.

La table 4.1 montre une comparaison entre *IOLD-BMOLS*, *IOLD-R2-IBMOLS*, *IOLD-EA*, NSGA-III, L-NSGA-II et *R2-IBMOLS* en termes de valeurs moyennes obtenues pour les distances minimales, médianes et maximales sur 30 exécutions (l'approximation ayant la plus petite distance du point de référence est la meilleure). La première colonne contient le nom de l'instance indiqué par sa taille m et n , correspondant respectivement au nombre d'objectifs et au nombre d'actions de l'instance. Chaque cellule de la table 4.1 contient trois valeurs : la distance minimale, la distance médiane et la distance maximale du point de référence (dans cet ordre). Les valeurs écrites dans un style gras signifient que l'algorithme correspondant est meilleur par rapport aux autres algorithmes, pour l'instance et la distance correspondantes. Les

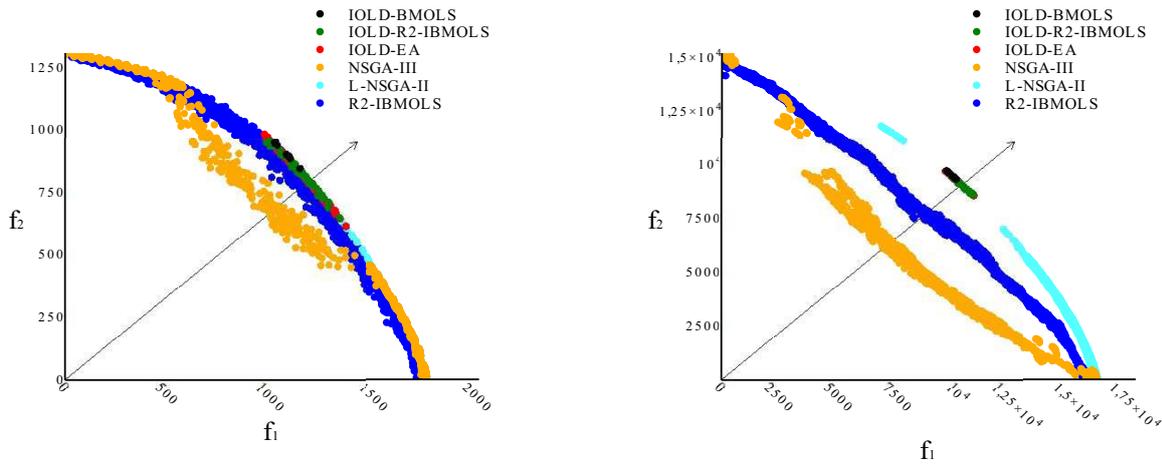


FIGURE 4.2 – Solutions obtenues avec *IOLD-BMOLS*, *IOLD-R2-IBMOLS*, *IOLD-EA*, *NSGA-III*, *L-NSGA-II* et *R2-IBMOLS* pour l'instance "2-50" (gauche) et "2-500" (droite).

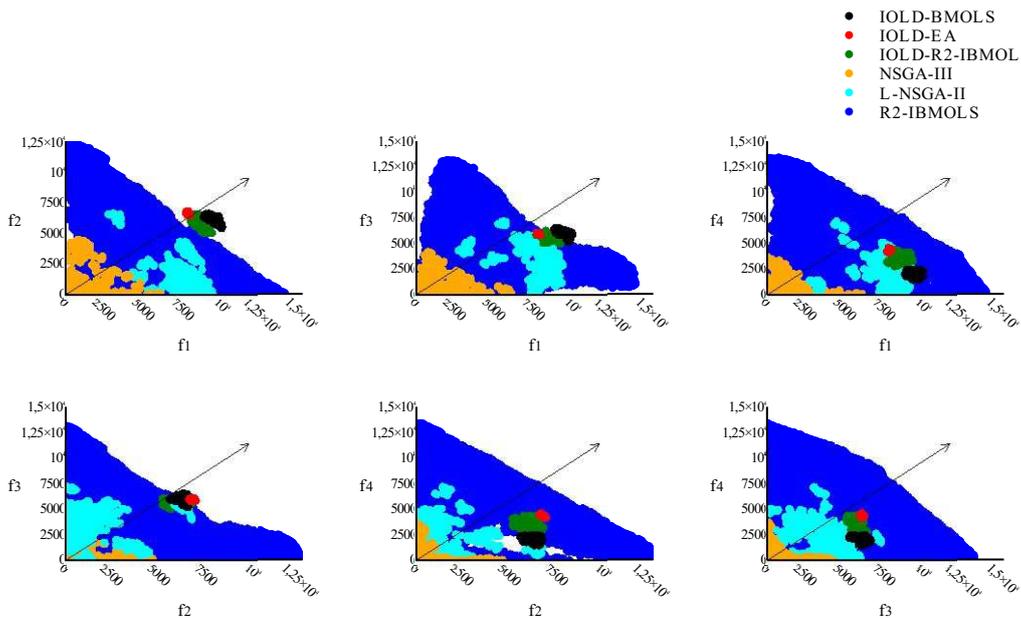


FIGURE 4.3 – Solutions obtenues avec *IOLD-BMOLS*, *IOLD-R2-IBMOLS*, *IOLD-EA*, *NSGA-III*, *L-NSGA-II* et *R2-IBMOLS* pour l'instance "4-500".

valeurs écrites en italique signifient que l'algorithme correspondant est meilleur que ceux ayant les valeurs dans un style normal mais il est moins performant que l'algorithme ayant la valeur écrite dans un style gras, pour l'instance et la distance correspondantes (c'est-à-dire c'est le 2^{me} meilleur algorithme).

Instance	<i>IOLD</i> -BMOLS	<i>IOLD</i> -R2-IBMOLS	<i>IOLD</i> -EA	NSGA-III	L-NSGA-II	R2-IBMOLS
2_50	2.096	2.050	1.999	2.080	2.049	1.912
	2.201	2.070	2.087	2.105	2.088	2.019
	2.249	2.241	2.238	2.250	2.236	2.247
2_100	2.111	2.078	2.090	2.109	2.018	2.003
	2.121	2.105	2.105	2.157	2.065	2.071
	2.236	2.236	2.245	2.238	2.243	2.247
2_150	2.118	2.080	2.087	2.140	1.911	1.937
	2.152	2.121	2.112	2.170	2.054	2.060
	2.236	2.338	2.236	2.238	2.236	2.244
2_250	2.120	2.104	2.097	2.142	2.034	1.991
	2.130	2.153	2.114	2.192	2.075	2.076
	2.236	2.269	2.236	2.237	2.243	2.251
2_500	2.119	2.097	2.009	2.155	2.097	2.032
	2.129	2.139	2.095	2.229	2.112	2.102
	2.236	2.238	2.238	2.259	2.236	2.241
3_50	2.531	2.394	2.469	2.729	2.540	2.567
	2.588	2.444	2.565	2.975	2.600	2.719
	2.749	2.703	2.812	3.099	2.884	2.928
3_100	2.389	2.459	2.413	2.774	2.493	2.521
	2.452	2.500	2.475	2.916	2.664	2.521
	2.655	2.654	2.690	3.201	2.962	2.997
3_150	2.552	2.377	2.449	2.734	2.486	2.579
	2.631	2.437	2.546	2.917	2.616	2.716
	2.779	2.728	2.860	3.223	2.966	3.003
3_250	2.476	2.472	2.516	2.825	2.462	2.697
	2.520	2.527	2.642	3.043	2.713	2.815
	2.627	2.840	2.856	3.377	2.912	3.013
3_500	2.481	2.482	2.400	2.672	2.429	2.656
	2.527	2.592	2.478	2.787	2.589	2.800
	2.785	2.994	2.743	3.196	2.968	3.007
4_50	2.864	2.849	2.669	3.351	2.981	2.992
	2.932	2.984	2.993	3.519	3.208	3.311
	3.275	3.212	3.263	3.711	3.556	3.578
4_100	2.910	2.884	2.784	3.452	3.101	2.768
	2.984	3.013	2.997	3.642	3.325	3.090
	3.117	3.246	3.236	3.766	3.596	3.439
4_150	2.805	2.895	2.737	3.353	2.763	3.035
	2.913	3.047	2.989	3.596	3.157	3.326
	3.134	3.284	3.221	3.805	3.472	3.595
4_250	3.025	2.664	2.713	3.339	2.969	3.046
	3.075	2.965	2.970	3.721	3.180	3.337
	3.194	3.227	3.241	3.816	3.482	3.589
4_500	3.061	2.926	2.810	3.480	2.994	3.129
	3.118	3.056	3.005	3.793	3.166	3.376
	3.248	3.246	3.282	3.922	3.490	3.600
5_50	3.117	3.208	3.194	3.814	3.237	3.566
	3.384	3.391	3.433	4.001	3.622	3.819
	3.803	3.599	3.719	4.161	4.041	4.100
5_100	3.214	3.158	3.033	3.919	3.416	3.655
	3.376	3.333	3.297	4.100	3.654	3.898
	3.566	3.564	3.648	4.244	4.005	4.115
5_150	3.301	3.106	3.082	3.728	3.246	3.640
	3.503	3.364	3.294	4.116	3.529	3.882
	3.628	3.566	3.555	4.258	3.923	4.124
5_250	3.324	3.171	2.930	3.915	3.533	3.709
	3.504	3.342	3.242	4.091	3.712	4.007
	3.689	3.632	3.525	4.211	4.016	4.245
5_500	3.217	3.149	3.144	3.625	3.243	3.742
	3.312	3.338	3.395	4.089	3.615	4.001
	3.405	3.569	3.683	4.218	3.967	4.219
6_50	3.467	3.598	3.496	4.220	3.378	3.699
	3.554	3.724	3.718	4.394	3.865	4.042
	3.971	3.947	3.956	4.583	4.312	4.483
6_100	3.699	3.582	3.509	4.091	3.610	4.035
	3.757	3.814	3.704	4.529	4.040	4.351
	3.935	4.095	3.912	4.667	4.484	4.580
6_150	3.631	3.569	3.358	4.206	3.767	3.931
	3.703	3.795	3.669	4.479	4.039	4.290
	3.860	4.311	3.927	4.625	4.413	4.563
6_250	3.530	3.591	3.328	4.260	3.535	3.951
	3.688	3.727	3.602	4.444	3.944	4.266
	3.847	3.950	3.866	4.585	4.412	4.554
6_500	3.626	3.517	3.308	3.807	3.649	3.959
	3.742	3.668	3.542	4.650	3.901	4.411
	3.920	3.907	3.840	4.747	4.200	4.665
8_50	4.182	4.127	4.005	5.032	4.181	4.263
	4.313	4.251	4.164	5.140	4.549	5.408
	4.551	4.568	4.445	5.326	5.035	5.656
8_100	4.076	4.175	4.079	5.058	4.121	4.433
	4.322	4.358	4.281	5.218	4.687	5.623
	4.526	4.655	4.510	5.370	5.131	5.656
8_150	4.085	4.024	4.056	4.594	4.139	4.498
	4.207	4.214	4.269	4.900	4.647	5.580
	4.440	4.454	4.531	5.278	5.096	5.656
8_250	4.109	4.079	3.910	4.381	4.347	4.582
	4.212	4.257	4.137	4.718	4.689	5.599
	4.387	4.432	4.418	5.290	4.996	5.656
8_500	4.170	4.051	3.986	4.445	4.272	4.573
	4.273	4.294	4.170	4.728	4.616	5.698
	4.432	4.421	4.415	5.131	4.941	5.743

TABLE 4.1 – Moyenne des distances euclidienne minimales, médianes, maximales du point de référence z^* des solutions obtenues par *IOLD*-BMOLS, *IOLD*-R2-IBMOLS, *IOLD*-EA, NSGA-III, L-NSGA-II et R2-IBMOLS sur 30 exécutions.

De la table 4.1, nous pouvons conclure que *IOLD*-BMOLS, *IOLD*-R2-IBMOLS et *IOLD*-EA sont plus efficaces que NSGA-III, L-NSGA-II et R2-IBMOLS et, en général, les solutions obtenues avec *IOLD*-BMOLS, *IOLD*-R2-IBMOLS et *IOLD*-EA sont plus proches de la région ciblée (donnée par le point de référence z^*). En effet, si nous regardons, par exemple, la distance minimale au point de référence, R2-IBMOLS obtient le meilleur score juste pour 4 instances ("2_50", "2_100", "2_250" et "4_100") et L-NSGA-II obtient le meilleur score pour 3 instances uniquement ("2_150", "3_250" et "6_50"). Quand à NSGA-III, il n'a obtenu aucun meilleur score pour toutes les instances considérées. Si nous considérons uniquement les algorithmes que nous avons proposés, la table 4.1 montre que *IOLD*-EA est plus efficace que *IOLD*-BMOLS et *IOLD*-R2-IBMOLS.

En pratique, il est préférable de présenter au décideur un nombre réduit de solutions non dominées mais de grande qualité. En effet, dans le cas du POPAMO, une seule solution pourrait avoir plusieurs dizaines d'actions. Ce qui fait que le choix du plan d'actions à mettre en œuvre peut s'avérer un peu difficile. Il est donc recommandé de réduire au maximum le nombre de solutions générées, d'où l'introduction du paramètre T dans nos algorithmes.

La table 4.2 montre le nombre moyen de solutions non dominées générées avec chacun des algorithmes sur les 30 exécutions effectuées ainsi que le temps moyen d'exécution correspondant (respectivement $nbSol$ et t). Nous notons de la table 4.2 que le nombre moyen de solutions générées est très variable pour les différents algorithmes. Le paramètre T des algorithmes *IOLD*-BMOLS, *IOLD*-R2-IBMOLS et *IOLD*-EA permet au décideur de fixer lui-même le nombre de solutions non dominées qu'il désire avoir. Dans nos expérimentations, nous avons fixé le paramètre T à 10 ($T = 10$) pour *IOLD*-BMOLS, *IOLD*-R2-IBMOLS et *IOLD*-EA. Par conséquent, on voit que tous les algorithmes que nous avons proposés fournissent tous un nombre de solutions égal à la valeur définie pour T . Le nombre moyen de solutions obtenues avec NSGA-III est égal à la taille de la population utilisée (100 solutions dans nos expérimentations). Ce qui indique que sur les dernières générations de NSGA-III, tous les individus de la population sont tous non dominés. La table 4.2 montre aussi que même pour la plus petite instance ("2_50"), NSGA-III, L-NSGA-II et R2-IBMOLS génèrent beaucoup de solutions, rendant le choix de la solution finale difficile. Cependant, même pour ces algorithmes, nous pouvons utiliser la fonction *infiniteOrderLorenzNonDominatedSolutions* (algorithme 7) pour ordonner les solutions générées et sélectionner les plus intéressantes pour le décideur.

Concernant le temps d'exécution des algorithmes, la table 4.2 montre que pour toutes les instances utilisées, les algorithmes proposés (*IOLD*-EA, *IOLD*-BMOLS et *IOLD*-R2-IBMOLS) sont moins coûteux comparés à NSGA-III, L-NSGA-II et R2-IBMOLS. Nous notons aussi que l'utilisation de la dominance de Lorenz d'ordre infini et l'introduction du paramètre T dans l'algorithme R2-IBMOLS ont permis de réduire considérablement le temps d'exécution de R2-IBMOLS (le temps d'exécution est divisé par un facteur de 275 pour l'instance "3_500" et un facteur de 106 pour l'instance "6_500").

Instance	IOLD-BMOLS		IOLD-R2-IBMOLS		IOLD-EA		NSGA-III		L-NSGA-II		R2-IBMOLS	
	nbSol	t	nbSol	t	nbSol	t	nbSol	t	nbSol	t	nbSol	t
2_50	10	2.35×10^{-1}	10	1.56×10^{-1}	10	6.8×10^{-2}	100	3.16	44	3.74	95	3.22×10^{-1}
2_100	10	4.68×10^{-1}	10	5.40×10^{-1}	10	1.85×10^{-1}	100	1.09×10^1	64	1.18×10^1	151	2.46
2_150	10	1.33	10	1.18	10	3.48×10^{-1}	100	2.48×10^1	100	2.44×10^1	175	6.53
2_250	10	5.23	10	3.62	10	7.50×10^{-1}	100	7.04×10^1	100	6.64×10^1	264	3.35×10^1
2_500	10	3.81×10^1	10	1.88×10^1	10	2.48	100	3.11×10^2	100	2.70×10^2	473	3.51×10^2
3_500	10	1.61×10^{-1}	10	8.46×10^{-1}	10	1.24×10^{-1}	100	5.76	100	8.52	820	1.30×10^1
3_1000	10	8.48×10^{-1}	10	4.12	10	3.25×10^{-1}	100	1.94×10^1	100	2.68×10^1	1160	9.38×10^1
3_150	10	2.6	10	9.45	10	6.41×10^{-1}	100	4.43×10^1	100	5.62×10^1	1748	4.49×10^2
3_250	10	1.09×10^1	10	5.80×10^1	10	1.36	100	1.26×10^2	100	1.39×10^2	3369	4.17×10^3
3_500	10	7.42×10^1	10	3.21×10^2	10	2.83	100	5.37×10^2	100	5.98×10^2	7635	8.84×10^4
4_50	10	1.15	10	1.19	10	0.34	100	8.91	20	1.61×10^1	381	7.48×10^1
4_100	10	3.01	10	1.77×10^1	10	1.12	100	3.31×10^1	28	4.96×10^1	666	7.99×10^1
4_150	10	4.47	10	7.44×10^1	10	1.16	100	7.06×10^1	52	1.02×10^2	1551	6.88×10^2
4_250	10	1.66×10^1	10	2.67×10^2	10	3.39	100	1.88×10^2	50	2.60×10^2	2385	3.45×10^3
4_500	10	1.31×10^2	10	2.51×10^4	10	1.33×10^{-1}	100	7.52×10^2	71	1.06×10^3	7591	1.03×10^5
5_50	10	2.50	10	1.18	10	9.50×10^{-1}	100	1.37×10^1	15	2.68×10^1	667	1.84×10^1
5_100	10	5.57	10	4.22×10^1	10	1.12	100	4.21×10^1	65	7.78×10^1	1992	5.14×10^2
5_150	10	6.69	10	1.38×10^1	10	2.25	100	8.63×10^1	73	1.59×10^2	3239	2.59×10^3
5_250	10	1.82×10^2	10	5.37×10^2	10	9.93	100	3.05×10^2	89	3.99×10^2	5257	1.58×10^4
5_500	10	2.66×10^2	10	4.53×10^3	10	2.44×10^2	100	1.65×10^3	96	1.59×10^3	15,898	4.77×10^5
6_50	10	5.51	10	4.49	10	1.01	100	1.52×10^1	27	3.71×10^1	799	2.45×10^1
6_100	10	9.99	10	5.66×10^1	10	1.16	100	9.16×10^2	60	4.81×10^2	1860	3.24×10^2
6_150	10	1.29×10^1	10	3.40×10^2	10	3.30	100	1.88×10^3	96	2.30×10^2	4971	5.39×10^3
6_250	10	4.99×10^1	10	1.54×10^3	10	1.21×10^1	100	6.63×10^3	87	1.53×10^2	10,658	5.46×10^4
6_500	10	2.46×10^2	10	1.23×10^4	10	2.96×10^1	100	2.06×10^4	100	2.38×10^3	31,068	1.32×10^6
8_50	10	2.20×10^1	10	2.27×10^1	10	1.44	100	2.46×10^1	87	6.60×10^1	2147	2.13×10^3
8_100	10	8.78×10^1	10	1.87×10^2	10	1.98	100	1.44×10^3	80	1.30×10^4	4052	2.19×10^4
8_150	10	5.56×10^1	10	1.38×10^3	10	3.63	100	2.88×10^3	100	1.31×10^5	10,588	3.08×10^5
8_250	10	6.13×10^2	10	2.86×10^3	10	1.60×10^1	100	1.01×10^4	100	1.97×10^5	40,404	2.12×10^6
8_500	10	4.08×10^3	10	2.20×10^4	10	3.43×10^1	100	3.16×10^4	100	2.03×10^5	58,413	2.89×10^6

TABLE 4.2 – Comparaison du nombre moyen de solutions générées et du temps moyen d'exécution pour IOLD-BMOLS, IOLD-R2-IBMOLS, IOLD-EA, NSGA-III, L-NSGA-II et R2-IBMOLS sur 30 exécutions.

4.5 Conclusion

Nous avons proposé dans ce chapitre trois algorithmes basés sur la dominance de Lorenz pour résoudre le problème d'optimisation de plans d'actions multi-objectifs. Nous avons proposé d'utiliser la dominance de Lorenz d'ordre infini, au lieu de la dominance de Pareto, dans trois algorithmes différents : *IOLD-BMOLS*, *IOLD-R2-BMOLS* et *IOLD-EA*. En effet, la dominance de Lorenz nous permet d'introduire une relation de préférence entre les solutions non dominées au sens de Pareto, en particulier lorsque leur nombre est élevé. Cependant, dans certains cas, l'application de la dominance de Lorenz de base ne garantit pas la détection d'une solution préférable entre deux solutions non dominées et, dans ce cas, la dominance de Lorenz d'ordre infini devient un bon moyen pour résoudre ce problème.

Nous avons évalué les méthodes que nous avons proposées sur des données simulées avec 50 à 500 actions et 2 à 8 objectifs. Nous avons également réalisé des études comparatives des algorithmes *IOLD-BMOLS*, *IOLD-R2-BMOLS*, *IOLD-EA* proposés avec trois autres algorithmes de référence (L-NSAGA-II, NSGA-III et *R2-IBMOLS*). Sur la base de ce travail, nous pouvons tirer deux conclusions principales. Premièrement, l'utilisation de la dominance de Lorenz d'ordre infini dans les algorithmes basés sur les indicateurs de qualité ou la recherche locale ou dans des algorithmes évolutionnaires pourrait s'avérer une méthode viable pour trouver des solutions de compromis de très bonne qualité pour le problème d'optimisation de plans d'actions. En outre, l'introduction du paramètre T nous a permis, d'une part, de réduire considérablement le temps d'exécution de *IOLD-R2-IBMOLS* par rapport à *R2-IBMOLS* et, d'autre part, de fournir au décideur un moyen de contrôler le nombre de solutions qui lui sont présentées.

Considérant les trois méthodes proposées, l'algorithme *IOLD-EA* est plus efficace que *IOLD-BMOLS* et *IOLD-R2-IBMOLS*. Il a obtenu les meilleurs résultats pour 16 instances sur les 30 instances testées (voir table 4.1). D'un autre côté, *IOLD-R2-IBMOLS* offre deux mécanismes qui permettent d'intégrer facilement les préférences du décideur : un point de référence et un ensemble de vecteurs de poids. Par conséquent, nous pouvons conclure que lorsque les préférences du décideur sont connues, il est plus approprié d'utiliser *IOLD-R2-IBMOLS* même s'il est un peu plus coûteux en temps de calcul. Par contre, quand les préférences du décideur sont inconnues ou que tous les objectifs ont la même importance pour le décideur, *IOLD-EA* devient le meilleur algorithme, car il permet d'obtenir en moins de temps des solutions équitables et de grande qualité (en terme de convergence). Deuxièmement, l'intégration des algorithmes *IOLD-R2-IBMOLS* et *IOLD-EA* dans le logiciel *MSQualité* constituerait un moyen considérable qui aiderait les directeurs des établissements du [SSMS](#) dans l'élaboration de plans d'actions efficaces. Un projet avec un plan d'actions optimal permet au responsable de réussir sa démarche d'amélioration du fonctionnement de son établissement et d'améliorer la qualité de vie de ses résidents.

Enfin, puisque ce travail a montré que les approches proposées fonctionnent bien pour le problème d'optimisation de plans d'actions, une perspective intéressante serait d'étudier les méthodes proposées sur d'autres problèmes multi-objectifs avec plus d'objectifs et une plus grande complexité.

Conclusion

Dans cette thèse, nous avons étudié un cas pratique du problème du sac-à-dos multi-objectif qui concerne l'optimisation de plans d'actions multi-objectifs dans un contexte budgétaire limité. Depuis la loi du 02 janvier 2002, qui modernise le fonctionnement et le contrôle du secteur social et médico-social, les responsables des établissements se confrontent au problème d'élaboration de plans d'actions au moins une fois tous les cinq ans. En effet, cette loi a instauré pour les établissements une obligation d'élaborer un CPOM quinquennal incluant obligatoirement un plan d'actions. Elle a également introduit un système d'évaluations internes et d'évaluations externes visant à installer une logique d'amélioration continue du fonctionnement des établissements et de la qualité de prise en charge des bénéficiaires.

Dans la première partie de notre travail (chapitre 1), nous avons étudié le contexte pratique dans lequel s'inscrit le problème d'élaboration de plans d'actions. Nous avons notamment présenté les différentes spécificités et contraintes légales, sociales et économiques qui régissent le quotidien des responsables des établissements sociaux et médico-sociaux. Nous avons vu que dans un contexte où, d'un côté, le nombre de personnes qui ont besoin d'assistance augmente et, d'un autre côté, les moyens financiers baissent, les établissements doivent innover et s'adapter pour continuer à garantir une qualité de service à la hauteur de leurs missions.

Dans la deuxième partie (chapitre 2), nous avons décrit le processus d'élaboration des plans d'actions ainsi que la nature des objectifs, des actions et des contraintes qui le définissent. Ensuite, nous avons fourni une modélisation formelle du problème sous forme d'un problème d'optimisation de sac-à-dos multi-objectif. Enfin, nous avons présenté la solution que nous intégrons dans le logiciel *MSQualité* pour assister le décideur à la fois dans la définition des actions efficaces et la sélection de celles qui intégreront son plan d'actions réel.

Dans la troisième partie (chapitre 3), pour résoudre le problème d'optimisation de plans d'actions, nous avons proposé la méthode *R2-IBMOLS* qui combine la méthode IBMOLS [Basseur and Burke, 2007] avec l'indicateur *R2* [Hansen and Jaszkiwicz, 1998; Brockhoff et al., 2012]. L'objectif de cette approche est d'exploiter les performances de la méthode IBMOLS avec un indicateur qui nous permet d'intégrer aisément les préférences du décideur. Nous avons montré, par nos expérimentations, que l'approche IBMOLS utilisée avec l'indicateur *R2* est une méthode viable pour trouver de bonnes solutions pour le problème étudié. Cependant, nos expérimentations montrent également que l'algorithme *R2-IBMOLS* est coûteux en temps de calcul et le nombre de solutions qu'il génère est trop élevé. Par conséquent, pour qu'il soit utilisé dans *MSQualité*, il nécessite d'être amélioré.

Dans la quatrième partie (chapitre 4), pour réduire le nombre de solutions générées par *R2-IBMOLS* et son temps d'exécution, nous avons proposé une nouvelle méthode que nous avons appelée *IOLD-R2-BMOLS*, où la dominance de Pareto de *R2-IBMOLS* est remplacée par la dominance de Lorenz d'ordre infini [Perny and Golden, 2010]. Nous avons également utilisé cette nouvelle dominance dans deux autres nouvelles méthodes (*IOLD-BMOLS* et *IOLD-EA*)

pour orienter la recherche vers le centre de l'espace objectif, où se trouvent les solutions les plus équitables. En effet, dans un contexte où les préférences du décideur ne sont pas connues ou que tous les objectifs ont la même importance pour lui, l'équité du traitement des objectifs devient aussi importante que l'optimalité de la solution. Par ailleurs, la dominance de Lorenz introduit une relation de préférence entre les solutions non dominées, ce qui permet de discriminer les solutions les moins intéressantes. Avec la dominance de Lorenz d'ordre infini, cette relation de préférence est appliquée à des degrés élevés et de manière itératives, raffinant ainsi le concept de la dominance. De plus, l'introduction du paramètre T dans nos méthodes nous a permis, d'une part, de remédier au problème du temps d'exécution de *R2-IBMOLS* et, d'autre part, de fournir au décideur un moyen pour contrôler le nombre de solutions qui lui sont présentées.

Impact pratique sur l'élaboration de plans d'actions

En tant que composant clé de *MSQualité*, la solution présentée dans ce travail fournit une aide précieuse au décideur sur les aspects suivants :

- chaque directeur d'une structure peut être au courant des actions déjà mises en œuvre dans d'autres structures, particulièrement celles qui contribuent à l'amélioration de leur fonctionnement et de la qualité de prise en charge de leurs résidents. Cette information aide le décideur à bien choisir les actions à réaliser dans son propre établissement ;
- pour évaluer l'impact réel des actions réalisées sur les objectifs, les décideurs utilisent souvent des indicateurs qui peuvent être nombreux et variés, selon la nature des actions. Cependant, avoir une plateforme commune et partagée dans *MSQualité* permet à chaque décideur d'exploiter et de bénéficier des grilles d'indicateurs construites par d'autres structures pour évaluer son propre plan d'actions ;
- comme cela a été mentionné dans la section 2.2 du chapitre 2, un des défis auquel est confronté le décideur consiste à définir et choisir les actions à mettre en œuvre pour atteindre les objectifs de son plan d'actions. Avec la solution proposée dans ce travail, cette tâche devient plus facile puisque le décideur n'a qu'à définir les objectifs de son projet, et *MSQualité* lui suggère des actions dont les impacts positifs ont déjà été prouvés dans le passé, dans au moins une structure similaire à la sienne. Dans les plans d'actions réels, environ 60% à 80% des actions pourraient être proposées par *MSQualité*. Ces taux correspondent aux actions qui sont directement liées aux missions de chaque type de structures ;
- avec la solution proposée, nous pensons que, pour un budget égal, les décideurs pourraient élaborer des plans d'actions qui sont 40% à 50% plus efficaces que ceux élaborés manuellement. En effet, 40% à 50% des actions de plans d'actions réels sont généralement mises en œuvre sans savoir préalablement si les actions seront efficaces ou non, ou dans quelle mesure elles le seront. Avec la méthode proposée, ces actions pourraient être remplacées par des actions dont les impacts positifs sont déjà connus. Car lorsqu'une action donnée a déjà été réalisée et évaluée dans n'importe quelle structure, les résultats de cette évaluation sont accessibles à tous les décideurs, ce qui leur permet d'élaborer des plans d'actions plus efficaces.

Par ailleurs, notre approche peut être améliorée sur quelques aspects :

- dans nos expérimentations, nous avons utilisé un voisinage relativement simple. Pour générer un voisin d'une solution donnée s , soit on ajoute une action à la solution s , soit on en supprime une. Il serait intéressant de tester d'autres voisinages plus évolués. Nous pouvons, par exemple, classer les actions d'une solution selon leurs qualités individuelles. La qualité d'une action peut être évaluée soit en utilisant un indicateur de qualité, soit en évaluant son impact (nombre d'objectifs améliorés par l'action, nombre d'objectifs détériorés et son coût), soit avec une formule définie par le décideur lui-même. A la génération d'un nouveau voisin, au lieu de remplacer une action quelconque d'une manière aléatoire, on remplace des actions de moins bonne qualité. La même méthode peut être également utilisée lors de la génération de la population initiale.
- comme cela a été indiqué dans le chapitre 3, l'algorithme $R2$ -IBMOLS est très coûteux en temps de calcul. Nous l'avons réduit considérablement dans l'algorithme $IOLD$ - $R2$ -IBMOLS en réduisant le nombre de solutions archivées (le paramètre T) et en remplaçant la dominance de Pareto par la dominance de Lorenz. Cependant, si on considère uniquement les vecteurs autour d'une solution lorsque on évalue sa qualité avec l'indicateur $R2$, le temps d'exécution de $IOLD$ - $R2$ -IBMOLS peut être réduit encore davantage.
- pour tester nos algorithmes, nous avons généré les instances utilisées de telle sorte à ce qu'elles correspondent le mieux possible aux plans d'actions réels. Mais pour avoir une évaluation précise de leur apport en pratique, il faudrait les utiliser pour l'élaboration d'un nouveau plan dans un établissement qui a déjà mis en place un plan d'actions et évalué ses résultats. Cela nous permettrait de comparer les résultats du plan d'actions élaboré manuellement avec celui élaboré avec *MSQualité*. Nous n'avons pas pu faire cette comparaison durant la durée de nos travaux car les plans d'actions sont souvent réalisés pour une période de cinq ans.

Pour conclure, cette étude montre que les systèmes informatisés d'aide à la décision ont beaucoup à offrir dans le secteur social et médico-social qui est en perpétuelle évolution. Les logiciels d'aide à la décision peuvent s'avérer très efficaces pour assister les décideurs à identifier et adopter des choix appropriés pour améliorer continuellement la qualité globale de leurs structures. Avec la solution proposée dans ce travail, nous fournissons aux directeurs des structures médico-sociales un outil viable pour optimiser leurs plans d'actions.

Enfin, puisque ce travail a montré que les approches proposées fonctionnent bien pour le problème d'optimisation de plans d'actions, une perspective intéressante serait de les évaluer sur d'autres problèmes multi-objectifs avec plus d'objectifs et une plus grande complexité. D'autant plus que l'on constate que les travaux liés à l'optimisation de Lorenz sont très limités par rapport au grand nombre d'études réalisées sur l'optimisation avec la dominance de Pareto.

Table des figures

1.1	Évolution des effectifs salariés dans le SSMS et dans l'ensemble des secteurs	18
1.2	Taux de croissance de l'emploi dans le SSMS et dans l'ensemble des secteurs	18
1.3	Catégories des établissements du SSMS [OSORIO MONTROYA, 2015]	19
1.4	Répartition du nombre de lits et de places entre le secteur public, le secteur privé lucratif et le secteur privé non lucratif.	22
1.5	Répartition des emplois occupés dans les structures d'accueil et d'hébergement des personnes handicapées entre le secteur public, le secteur privé lucratif et le secteur privé non lucratif [Source : enquête DREES, ES 2006, calculs UNIOSS].	22
1.6	Répartition des emplois occupés dans les structures d'accueil et d'hébergement pour personnes âgées entre le secteur public, le secteur privé lucratif et le secteur privé non lucratif [DREES, 2010].	22
1.7	Répartition des emplois occupés dans les structures d'accueil de jeunes enfants de moins de 6 ans entre le secteur public, le secteur privé lucratif et le secteur privé non lucratif [Source : enquête PMI-DREES 2008, calculs UNIOSS].	22
1.8	Répartition des emplois occupés dans les structures recevant des mineurs entre le secteur public, le secteur privé lucratif et le secteur privé non lucratif [Source : projet de loi de finances 2011, calculs UNIOSS]	22
1.9	Vieillesse de la population française à l'horizon 2060 [Source : INSEE, 2010, projections de population 2007-2060]	25
1.10	Nombre de personnes, âgées de 20 à 59 ans, en situation de handicap [Source : INSEE, 2009, Enquête Handicap-Santé 2008-2009, volet ménages]	26
1.11	Contribution globale des finances publiques à la compensation de la perte d'autonomie des personnes âgées et des personnes handicapées en 2014	26
1.12	Contribution des finances publiques à la compensation de la perte d'autonomie des personnes âgées (gauche) et des personnes handicapées (droite) en 2014.	26
1.13	Problème d'optimisation multi-objectif : cas avec deux variables de décision et trois fonctions objectifs [Basseur, 2005].	30
1.14	Vecteurs dominés et vecteurs non dominés dans l'espace objectif.	31
1.15	Exemple d'approximation du front Pareto.	31
2.1	Le cycle d'évaluations et de plans d'actions dans le SSMS	39
2.2	Processus d'élaboration des plans d'actions dans les établissements du SSMS	40
2.3	Processus d'élaboration et d'optimisation des plans d'actions avec <i>MSQualité</i>	45
3.1	Illustration de l'indicateur Epsilon appliqué à deux solutions non dominées x_1 et x_2 (gauche), $x_2 \prec x_1$ (centre) et à deux approximations A et B (droite).	56

3.2	Illustration de l'indicateur HV appliqué à deux solutions non dominées x_1 et x_2 (à gauche) et $x_2 \prec x_1$ (à droite) [Zitzler and Künzli, 2004].	56
3.3	Illustration de la réduction de l'espace objectif (espace gris) en changeant la position du point de référence, de $z^* = (0, 0)$ (gauche) à $z^* = (0.2, 0.2)$ (centre) ou en agissant sur la distribution des vecteurs de poids normalisés (droite). Les flèches pointillées correspondent aux directions définies par les vecteurs de poids.	58
3.4	Valeurs maximales obtenues avec NSGA-II, ϵ -IBMOLS et $R2$ -IBMOLS sur 30 exécutions pour les objectifs de quatre instances représentatives, avec 50 ou 500 actions et 5 ou 6 objectifs "6_500" (haut gauche), "6_50" (haut droite), "5_500" (bas gauche) et "5_50" (bas droite).	71
4.1	Espace objectif dominé par y dans le cas d'une dominance de Pareto (gauche) et d'une dominance de Lorenz (droite).	80
4.2	Solutions obtenues avec $IOLD$ -BMOLS, $IOLD$ - $R2$ -IBMOLS, $IOLD$ -EA, NSGA-III, L-NSGA-II et $R2$ -IBMOLS pour l'instance "2-50" (gauche) et "2-500" (droite).	92
4.3	Solutions obtenues avec $IOLD$ -BMOLS, $IOLD$ - $R2$ -IBMOLS, $IOLD$ -EA, NSGA-III, L-NSGA-II et $R2$ -IBMOLS pour l'instance "4-500".	92

Liste des tableaux

1.1	Les principaux apports des lois du 2 janvier 2002, du 5 février 2005 et du 22 juillet 2009.	16
1.2	Les principaux acteurs du secteur public.	21
3.1	Comparaison des valeurs moyennes des indicateurs ϵ et R obtenues avec PISA pour les algorithmes NSGA-II, ϵ -IBMOLS et $R2$ -IBMOLS.	70
3.2	Nombres moyens de solutions non dominées générées par NSGA-II, ϵ -IBMOLS et $R2$ -IBMOLS sur 30 exécutions.	72
3.3	Temps d'exécution moyens de NSGA-II, ϵ -IBMOLS et $R2$ -IBMOLS sur 30 exécutions.	73
4.1	Moyenne des distances euclidienne minimales, médianes, maximales du point de référence z^* des solutions obtenues par $IOLD$ -BMOLS, $IOLD$ - $R2$ -IBMOLS, $IOLD$ -EA, NSGA-III, L-NSGA-II et $R2$ -IBMOLS sur 30 exécutions.	93
4.2	Comparaison du nombre moyen de solutions générées et du temps moyen d'exécution pour $IOLD$ -BMOLS, $IOLD$ - $R2$ -IBMOLS, $IOLD$ -EA, NSGA-III, L-NSGA-II et $R2$ -IBMOLS sur 30 exécutions.	95

Liste des acronymes

ANAP Agence Nationale d'Appui à la Performance des établissements de santé et médico-sociaux. [13](#), [16](#), [17](#)

ANESM Agence Nationale de l'Evaluation et de la qualité des établissements et Services Sociaux et Médico-sociaux. [16](#)

ARS Agences Régionales de Santé. [13](#), [17](#), [21](#)

CAF Caisse d'Allocations Familiales. [14](#)

CNSA Caisse Nationale de Solidarité pour l'Autonomie. [13](#), [16](#), [17](#), [21](#)

CPAM Caisse Primaire d'Assurance Maladie. [21](#)

CPOM Contrat Pluriannuel d'Objectifs et de Moyens. [13](#), [27](#), [38](#), [45](#), [95](#)

DDASS Direction Départementale des Affaires Sanitaires et Sociales. [14](#)

DGCS Direction Générale de la Cohésion Sociale. [21](#)

EPRD État Prévisionnel des Recettes et des Dépenses. [27](#)

ESAT Établissement et Service d'Aide par le Travail. [40](#)

ESSMS Établissements et Services Sociaux et Médico-Sociaux. [16](#), [17](#), [26](#), [27](#)

IGAS Inspection Générale des Affaires Sociales. [21](#)

INSEE Institut National de la Statistique et des Études Économiques. [25](#)

MDPH Maison Départementale des Personnes Handicapées. [21](#)

PRIAC Programme Régional Interdépartemental d'Accompagnement des handicaps et de la perte d'autonomie. [13](#)

PRS Projet Régional de Santé. [13](#)

SDOSMS Schéma Départemental d'Organisation Sociale et Médico-sociale, appels à projets. [13](#)

SERAFIN-PH Services et Établissements : Réforme pour une Adéquation des FINancements au Parcours des Personnes Handicapées. [27](#)

SROMS Schéma Régional d'Organisation Médico-Sociale. [13](#)

SSMS Secteur Social et Médico-Social. [12–14](#), [16–19](#), [24](#), [27](#), [35](#), [38–40](#), [45](#), [52](#), [94](#), [99](#)

Liste des publications

Revue internationale

- B. Chabane, M. Basseur, and J.-K. Hao, *R2-IBMOLS applied to a practical case of the multiobjective knapsack problem*, Expert Systems with Applications, vol. 71, pp. 457–468, 2017.
- B. Chabane, M. Basseur, and J.-K. Hao, *Lorenz dominance based algorithms to solve a practical case of the multiobjective knapsack problem*. (soumis).

Conférences internationales avec comité de sélection

- B. Chabane, M. Basseur, and J.-K. Hao, *A practical case of the multiobjective knapsack problem : Design, modelling, tests and analysis*, in Learning and Intelligent Optimization (C. Dhaenens, L. Jourdan, and M.-E.Marmion, eds.), no. 8994 in Lecture Notes in Computer Science, pp. 249–255, Springer International Publishing, 2015. (papier court).

Conférences nationales avec actes de résumés

- B. Chabane, M. Basseur, and J.-K. Hao, *R2-IBMOLS : appliqué à un problème pratique du sac-à-dos multiobjectif*, in Recherche Opérationnelle et Aide à la DEcision Française (ROADEF'16) proceedings, Compiègne, France, February 2016.
- B. Chabane, M. Basseur, and J.-K. Hao, *Cas pratique pour le problème du sac-à-dos multiobjectif : Conception, modélisation, tests et analyse*, in Recherche Opérationnelle et Aide à la DEcision Française (ROADEF'15) proceedings, Marseille, France, February 2015.

Bibliographie

- [Agence Nationale d'Appui à la Performance, 2013] cité page 13
Agence Nationale d'Appui à la Performance. Anticiper et comprendre. In *Le secteur médico-social : comprendre pour agir mieux*. Agence Nationale d'Appui à la Performance, 2013.
- [Aisopos *et al.*, 2013] cité page 50
Fotis Aisopos, Konstantinos Tserpes, and Theodora Varvarigou. Resource management in software as a service using the knapsack problem model. *International Journal of Production Economics*, 141(2) :465–477, 2013.
- [Alves and Almeida, 2007] cité page 50
Maria João Alves and Marla Almeida. MOTGA : A multiobjective tchebycheff based genetic algorithm for the multidimensional knapsack problem. *Computers & Operations Research*, 34(11) :3458–3470, 2007.
- [ARS Île-de-France, 2016] cité page 24, 25
ARS Île-de-France. Schéma régional d'organisation médico-sociale. Document de travail, ARS Île-de-France, 2016.
- [Bader and Zitzler, 2011] cité page 61, 74
Johannes Bader and Eckart Zitzler. HypE : An algorithm for fast hypervolume-based many-objective optimization. *Evolutionary Computation*, 19(1) :45–76, 2011.
- [Barichard and Hao, 2003] cité page 42, 47
Vincent Barichard and Jin-Kao Hao. Genetic tabu search for the multi-objective knapsack problem. *Journal of Tsinghua Science and Technology*, 8(1) :8–13, 2003.
- [Basseur and Burke, 2007] cité page 52, 54, 62, 78, 97
Matthieu Basseur and Edmund Burke. Indicator-based multi-objective local search. In *IEEE Congress on Evolutionary Computation. (CEC 2007)*, pages 3100–3107, 2007.
- [Basseur and Zitzler, 2006] cité page 61
Matthieu Basseur and Eckart Zitzler. Handling uncertainty in indicator based multiobjective optimization. *International Journal of Computational Intelligence Research*, 2(3) :255–272, 2006.
- [Basseur *et al.*, 2012a] cité page 54, 62, 64, 66, 78, 90
Matthieu Basseur, Arnaud Liefooghe, Khoi Lee, and Edmund Burke. The efficiency of indicator-based local search for multi-objective combinatorial optimisation problems. *Journal of Heuristics*, 18(2) :263–296, 2012.

- [Basseur *et al.*, 2012b] cité page 62, 64
Matthieu Basseur, Rong-Qiang Zeng, and Jin-Kao Hao. Hypervolume-based multi-objective local search. *Neural Computing and Applications*, 21(8) :1917–1929, 2012.
- [Basseur, 2005] cité page 29, 29, 101
Matthieu Basseur. *Conception d'algorithmes coopératifs pour l'optimisation multi-objectif : application aux problèmes d'ordonnancement de type flow-shop*. Thèse, Université Lille 1 - Sciences et technologies, France, 2005.
- [Battiti and Passerini, 2010] cité page 58
Roberto Battiti and Andrea Passerini. Brain-computer evolutionary multiobjective optimization : A genetic algorithm adapting to the decision maker. *IEEE Transactions on Evolutionary Computation*, 14(5) :671–687, 2010.
- [Bazgan *et al.*, 2009a] cité page 47, 48, 48, 49
Cristina Bazgan, Hadrien Hugot, and Daniel Vanderpooten. Implementing an efficient fpts for the 0–1 multi-objective knapsack problem. *European Journal of Operational Research*, 198(1) :47–56, 2009.
- [Bazgan *et al.*, 2009b] cité page 50
Cristina Bazgan, Hadrien Hugot, and Daniel Vanderpooten. Solving efficiently the 0–1 multi-objective knapsack problem. *Computers & Operations Research*, 36(1) :260–279, 2009.
- [Bazillon, 2013] cité page 20, 20
Bernard Bazillon. Secteur social et médico-social : Regards croisés, enjeux et perspectives. Rapport, KPMG, 2013.
- [Bechikh *et al.*, 2010a] cité page 60
Slim Bechikh, Lamjed Ben Said, and Khaled Ghédira. Searching for knee regions in multi-objective optimization using mobile reference points. In *Proceedings of the 25th ACM Symposium on Applied Computing, SAC '10*, pages 1118–1125, New York, NY, USA, 2010. ACM.
- [Bechikh *et al.*, 2010b] cité page 60
Slim Bechikh, Lamjed Ben Said, and Khaled Ghedira. The r-dominance : A new dominance relation for interactive evolutionary multicriteria decision making. *IEEE Transactions on Evolutionary Computation*, 14(5) :801–818, 2010.
- [Bechikh *et al.*, 2011] cité page 60
Slim Bechikh, Lamjed Ben Said, and Khaled Ghédira. Searching for knee regions of the pareto front using mobile reference points. *Soft Computing*, 15(9) :1807–1823, 2011.
- [Bechikh *et al.*, 2015] cité page 58
Slim Bechikh, Marouane Kessentini, Lamjed Ben Said, and Khaled Ghédira. Preference incorporation in evolutionary multiobjective optimization : A survey of the state-of-the-art. In Ali R. Hurson, editor, *Advances in Computers*, volume 98, pages 141–207. Elsevier, 2015.
- [Ben Abdelaziz and Krichen, 1997] cité page 49
Foued Ben Abdelaziz and Saoussen Krichen. A tabu search heuristic for multiobjective

- knapsack problems. Technical Report RRR 28-97, Rutgers Center for Operations Research, New Jersey, USA, New Jersey, USA, 1997.
- [Ben Abdelaziz *et al.*, 1999] cité page 47, 49
Foued Ben Abdelaziz, Saoussen Krichen, and Jouhaina Chaouachi. A hybrid heuristic for multiobjective knapsack problems. In *Meta-heuristics : advances and trends in local search paradigms for optimization*, pages 205–212. Springer, 1999.
- [Ben Mansour and Alaya, 2015] cité page 51, 61
Imen Ben Mansour and Ines Alaya. Indicator based ant colony optimization for multi-objective knapsack problem. *Procedia Computer Science*, 60 :448–457, 2015.
- [Ben Mansour *et al.*, 2017] cité page 51
Imen Ben Mansour, Ines Alaya, and Moncef Tagina. A min-max tchebycheff based local search approach for MOMKP. In *Proceedings of the 12th International Conference on Software Technologies*, pages 140–150, Madrid, Spain, 2017.
- [Beume *et al.*, 2009] cité page 54
N. Beume, C. M. Fonseca, M. Lopez-Ibanez, L. Paquete, and J. Vahrenhold. On the complexity of computing the hypervolume indicator. *IEEE Transactions on Evolutionary Computation*, 13(5) :1075–1082, 2009.
- [Brams and Taylor, 1996] cité page 79
Steven J. Brams and Alan D. Taylor. *Fair Division : From Cake-Cutting to Dispute Resolution*. Cambridge University Press, 1996.
- [Branke and Deb, 2005] cité page 60
Jürgen Branke and Kalyanmoy Deb. Integrating user preferences into evolutionary multi-objective optimization. In Yaochu Jin, editor, *Knowledge Incorporation in Evolutionary Computation*, volume 167, pages 461–477. Springer Berlin Heidelberg, Berlin, Heidelberg, 2005.
- [Brockhoff and Zitzler, 2007] cité page 61
Dimo Brockhoff and Eckart Zitzler. Improving hypervolume-based multiobjective evolutionary algorithms by using objective reduction methods. In *Congress on Evolutionary Computation (CEC 2007)*, pages 2086–2093, 2007.
- [Brockhoff *et al.*, 2012] cité page 52, 54, 55, 56, 57, 57, 73, 78, 97
Dimo Brockhoff, Tobias Wagner, and Heike Trautmann. On the properties of the r2 indicator. In *Proceedings of the 14th Annual Conference on Genetic and Evolutionary Computation, GECCO '12*, pages 465–472, New York, NY, USA, 2012. ACM.
- [Brockhoff *et al.*, 2014] cité page 56, 57, 57, 61
Dimo Brockhoff, Tobias Wagner, and Heike Trautmann. R2 indicator-based multiobjective search. *Evolutionary Computation*, 23(3) :369–395, 2014.
- [Captivo *et al.*, 2003] cité page 48
M. Eugénia Captivo, Joao Climaco, José Figueira, Ernesto Martins, and Jos Luis Santos. Solving bicriteria 0–1 knapsack problems using a labeling algorithm. *Computers & Operations Research*, 30(12) :1865–1886, 2003.

- [Cerqueus, 2015] cité page 51
Audrey Cerqueus. *Bi-objective branch-and-cut algorithms applied to the binary knapsack problem*. Thèse, Université de Nantes, France, 2015.
- [Chabane *et al.*, 2015] cité page 53, 54, 63, 64
Brahim Chabane, Matthieu Basseur, and Jin-Kao Hao. A practical case of the multiobjective knapsack problem : Design, modelling, tests and analysis. In Clarisse Dhaenens, Laetitia Jourdan, and Marie-Éléonore Marmion, editors, *Learning and Intelligent Optimization*, number 8994 in Lecture Notes in Computer Science, pages 249–255. Springer International Publishing, 2015.
- [Chabane *et al.*, 2017] cité page 53, 89
Brahim Chabane, Matthieu Basseur, and Jin-Kao Hao. R2-IBMOLS applied to a practical case of the multiobjective knapsack problem. *Expert Systems with Applications*, 71 :457–468, 2017.
- [Chen and Hao, 2014] cité page 47
Yuning Chen and Jin-Kao Hao. A “reduce and solve” approach for the multiple-choice multidimensional knapsack problem. *European Journal of Operational Research*, 239(2) :313–322, 2014.
- [Cherfi and Hifi, 2009] cité page 47
Nawal Cherfi and Mhand Hifi. Hybrid algorithms for the multiple-choice multi-dimensional knapsack problem. *International Journal of Operational Research*, 5(1) :89–109, 2009.
- [Chong, 1976] cité page 81
Kong Ming Chong. An induction theorem for rearrangements. *Canadian Journal of Mathematics*, 28(1) :154–160, 1976.
- [Coello *et al.*, 2007] cité page 33
Carlos Coello Coello, Gary B. Lamont, and David A Van Veldhuizen. *Evolutionary Algorithms for Solving Multi-Objective Problems*. Springer, second edition, 2007.
- [Cvetkovic and Parmee, 2002] cité page 59
D. Cvetkovic and I. C. Parmee. Preferences and their application in evolutionary multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 6(1) :42–57, 2002.
- [Czyżżak and Jaskiewicz, 1998] cité page 49
Piotr Czyżżak and Adrezej Jaskiewicz. Pareto simulated annealing—a metaheuristic technique for multiple-objective combinatorial optimization. *Journal of Multi-Criteria Decision Analysis*, 7(1) :34–47, 1998.
- [Dalton, 1920] cité page 79
Hugh Dalton. The measurement of the inequality of incomes. *Economic Journal*, 30 :348—361, 1920.
- [Díaz-Manríquez *et al.*, 2016] cité page 61
Alan Díaz-Manríquez, Gregorio Toscano, Jose Hugo Barron-Zambrano, and Edgar Tello-Leal. R2-based multi/many-objective particle swarm optimization. *Computational Intelligence and Neuroscience*, 2016 :10, 2016.

- [Deb and Jain, 2014] cité page 87, 89, 90
Kalyanmoy Deb and H. Jain. An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i : Solving problems with box constraints (NSGA-III). *IEEE Transactions on Evolutionary Computation*, 18(4) :577–601, 2014.
- [Deb and Kumar, 2007] cité page 60
Kalyanmoy Deb and Abhishek Kumar. Interactive evolutionary multi-objective optimization and decision-making using reference direction method. In *Proceedings of GECCO 2007 : Genetic and Evolutionary Computation Conference*, pages 781–788, London, England, 2007.
- [Deb *et al.*, 2002] cité page 54, 60, 62, 66, 67, 81, 87
Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T Meyarivan. A fast and elitist multiobjective genetic algorithm : NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2) :182–197, 2002.
- [Deb *et al.*, 2006] cité page 60
Kalyanmoy Deb, J. Sundar, Bhaskara Rao Udaya, and Shamik Chaudhuri. Reference point based multi-objective optimization using evolutionary algorithms. In *International Journal of Computational Intelligence Research*, pages 635–642. Springer-Verlag, 2006.
- [Deb *et al.*, 2010] cité page 58
Kalyanmoy Deb, Ankur Sinha, Pekka. J. Korhonen, and Jyrki Wallenius. An interactive evolutionary multiobjective optimization method based on progressively approximated value functions. 14(5) :723–739, 2010.
- [Deb, 1999] cité page 60
Kalyanmoy Deb. Multi-objective evolutionary algorithms : Introducing bias among pareto-optimal solutions. *Evolutionary Computation*, 8 :173–195, 1999.
- [Delort and Spanjaard, 2010] cité page 48
Charles Delort and Olivier Spanjaard. Using bound sets in multiobjective optimization : Application to the biobjective binary knapsack problem. In *Proceedings of the 9th International Conference on Experimental Algorithms (SEA'10)*, Italy, 2010.
- [Direction des études, statistiques et Prévisions du MASS, 2012] cité page 13, 13, 17, 17
Direction des études, statistiques et Prévisions du MASS. Le champ social et medico-social : une activité en forte croissance, des métiers qui se développent et se diversifient. *Repères et Analyses*, 44, 2012.
- [DREES, 2010] cité page 21, 21, 101
DREES. Les établissements d'hébergement pour personnes âgées au 31/12/2007. Document de travail, DREES, 2010.
- [Dugardin *et al.*, 2010a] cité page 79
Frédéric Dugardin, Farouk Yalaoui, and Lionel Amodeo. New multi-objective method to solve reentrant hybrid flow shop scheduling problem. *European Journal of Operational Research*, 203(1) :22–31, 2010.

- [Dugardin *et al.*, 2010b] cité page 81, 89, 89, 89
 Frédéric Dugardin, Farouk Yalaoui, and Lionel Amodeo. New multi-objective method to solve reentrant hybrid flow shop scheduling problem. *European Journal of Operational Research*, 203(1) :22–31, 2010.
- [Dugardin *et al.*, 2011] cité page 82
 F. Dugardin, L. Amodeo, and F. Yalaoui. Fuzzy lorenz ant colony system to solve multiobjective reentrant hybride flowshop scheduling problem. In *2011 International Conference on Communications, Computing and Control Applications (CCCA)*, pages 1–6, 2011.
- [Ehr Gott and Gandibleux, 2007] cité page 50
 Matthias Ehr Gott and Xavier Gandibleux. Bound sets for biobjective combinatorial optimization problems. *Computers & Operations Research*, 34(9) :2674–2694, 2007.
- [Ehr Gott, 2000] cité page 32
 Matthias Ehr Gott. *Multicriteria Optimization*, volume 491 of *Lecture Notes in Economics and Mathematical Systems*. Springer, 2000.
- [Ehr Gott, 2005] cité page 28
 Matthias Ehr Gott. *Multicriteria Optimization*. Springer, 2nd edition, 2005.
- [Emmerich *et al.*, 2005] cité page 61
 Michael Emmerich, Nicola Beume, and Boris Naujoks. An EMO algorithm using the hypervolume measure as selection criterion. In Carlos A. Coello Coello, Arturo Hernández Aguirre, and Eckart Zitzler, editors, *Evolutionary Multi-Criterion Optimization*, number 3410 in *Lecture Notes in Computer Science*, pages 62–76. Springer, 2005.
- [Erlebach *et al.*, 2002] cité page 48, 49
 Thomas Erlebach, Hans Kellerer, and Ulrich Pferschy. Approximating multiobjective knapsack problems. *Management Science*, 48(12) :1603–1612, 2002.
- [Falcón-Cardona and Coello, 2016] cité page 61
 Jesús Guillermo Falcón-Cardona and Carlos A. Coello Coello. iMOACO_{mathbb{R}} : A new indicator-based multi-objective ant colony optimization algorithm for continuous search spaces. In Julia Handl, Emma Hart, Peter R. Lewis, Manuel López-Ibáñez, Gabriela Ochoa, and Ben Paechter, editors, *Parallel Problem Solving from Nature – PPSN XIV*, number 9921 in *Lecture Notes in Computer Science*, pages 389–398. Springer, 2016.
- [Figueira *et al.*, 2013] cité page 50
 José Figueira, Luís Paquete, Marco Simões, and Daniel Vanderpooten. Algorithmic improvements on dynamic programming for the bi-objective $\{0,1\}$ knapsack problem. *Computational Optimization and Applications*, 56(1) :97–111, 2013.
- [Figuera *et al.*, 2006] cité page 48
 J.R Figuera, M Wiecek, and J Tavares. Multiple criteria knapsack problems : network models and computational results. In *International Conference on Multi-Objective Programming and Goal Programming (MOPGP '06)*, Tours, France, 2006.

- [Florence and Savéant, 2010] cité page 46
Aligne Florence and Pierre Savéant. Gestion de crise : optimisation de la mise en œuvre des plans de secours. In *4ème Workshop Interdisciplinaire sur la Sécurité Globale (WISG 2010)*, 2010.
- [Fonseca and Fleming, 1993] cité page 60
Carlos M. Fonseca and Peter J. Fleming. Genetic algorithms for multiobjective optimization : Formulation, discussion and generalization. In *Proceedings of the 5th International Conference on Genetic Algorithms (ICGA'93)*, pages 416–423, Urbana-Champaign, USA, 1993. Morgan Kaufmann.
- [Fonseca and Fleming, 1998] cité page 60
C. M. Fonseca and P. J. Fleming. Multiobjective optimization and multiple constraint handling with evolutionary algorithms. I. A unified formulation. *IEEE Transactions on Systems, Man, and Cybernetics - Part A : Systems and Humans*, 28(1) :26–37, 1998.
- [Galand and Lust, 2015a] cité page 82, 82
Lucie Galand and Thibaut Lust. Exact methods for computing all lorenz optimal solutions to biobjective problems. In *Proceedings of the 4th International Conference on Algorithmic Decision Theory*, volume 9346, pages 305–321, New York, NY, USA, 2015.
- [Galand and Lust, 2015b] cité page 82, 82
Lucie Galand and Thibaut Lust. Multiagent fair optimization with lorenz dominance. In *Proceedings of the 2015 International Conference on Autonomous Agents and Multiagent Systems*, pages 1895–1896, Istanbul, Turkey, 2015.
- [Gallinaro, 2011] cité page 14
Danièle Gallinaro. Evolution du secteur social et médico-social du à la réunion de 1946 à 2011. Rapport, IRTS Réunion, 2011.
- [Gallo *et al.*, 1980] cité page 47
G. Gallo, P. L. Hammer, and B. Simeone. Quadratic knapsack problems. In M. W. Padberg, editor, *Combinatorial Optimization*, number 12 in Mathematical Programming Studies, pages 132–149. Springer Berlin Heidelberg, 1980.
- [Gandibleux and Freville, 2000] cité page 49
Xavier Gandibleux and Arnaud Freville. Tabu search based procedure for solving the 0-1 MultiObjective knapsack problem : The two objectives case. *Journal of Heuristics*, 6(3) :361–383, 2000.
- [Gandibleux and Klamroth, 2006] cité page 50
Xavier Gandibleux and Kathrin Klamroth. Cardinality bounds for multiobjective knapsack problems based on weighted sums scalarizations. In *Proceedings of the Multi-Objective Programming and Goal Programming Conference (MOPGP'06)*, Tours, France, 2006.
- [Gomes da Silva *et al.*, 2006] cité page 50
Carlos Gomes da Silva, João Clímaco, and José Figueira. A scatter search method for bi-criteria $\{0, 1\}$ -knapsack problems. *European Journal of Operational Research*, 169(2) :373–391, 2006.

- [Gomes da Silva *et al.*, 2007] cité page 50
Carlos Gomes da Silva, José Figueira, and João Clímaco. Integrating partial optimization with scatter search for solving bi-criteria $\{0, 1\}$ -knapsack problems. *European Journal of Operational Research*, 177(3) :1656–1677, 2007.
- [Gomes da Silva *et al.*, 2008] cité page 49
Carlos Gomes da Silva, João Clímaco, and José Rui Figueira. Core problems in bi-criteria -knapsack problems. *Computers & Operations Research*, 35(7) :2292–2306, 2008.
- [Gonçalves and Resende, 2011] cité page 62
José Fernando Gonçalves and Mauricio G. C. Resende. Biased random-key genetic algorithms for combinatorial optimization. *Journal of Heuristics*, 17(5) :487–525, 2011.
- [Greenwood *et al.*, 1996] cité page 58
Garrison W. Greenwood, Xiaobo Hu, and Joseph G. D’Ambrosio. Fitness functions for multiple objective optimization problems : Combining preferences with pareto rankings. In *Proceedings of the 4th Workshop on Foundations of Genetic Algorithms. San Diego, CA, USA, August 5 1996*, pages 437–455, 1996.
- [Hansen and Jaszkiwicz, 1998] cité page 54, 55, 56, 57, 57, 97
Michael Pilegaard Hansen and Andrzej Jaszkiwicz. Evaluating the quality of approximations to the non-dominated set. Technical Report IMM-REP-1998-7, Institute of Mathematical Modelling Technical University of Denmark, Denmark, 1998.
- [Hao *et al.*, 1999] cité page 32
Jin-Kao Hao, Philippe Galinier, and Michel Habib. Métaheuristiques pour l’optimisation combinatoire et l’affectation sous contraintes. *Revue d’Intelligence Artificielle*, 13(2) :283–324, 1999.
- [Iqbal *et al.*, 2010] cité page 47
Shahreaz Iqbal, Md Faizul Bari, and M. Sohel Rahman. Solving the multi-dimensional multi-choice knapsack problem with the help of ants. In *Swarm Intelligence*, number 6234 in Lecture Notes in Computer Science, pages 312–323. 2010.
- [Ishibuchi *et al.*, 2014] cité page 51
Hisao Ishibuchi, Yuki Tanigaki, Hiroyuki Masuda, and Yusuke Nojima. Distance-based analysis of crossover operators for many-objective knapsack problems. In Thomas Bartz-Beielstein, Jürgen Branke, Bogdan Filipič, and Jim Smith, editors, *Parallel Problem Solving from Nature – PPSN XIII*, number 8672 in Lecture Notes in Computer Science, pages 600–610. Springer International Publishing, 2014.
- [Jalali Varnamkhasti, 2012] cité page 47
Malik Jalali Varnamkhasti. Overview of the algorithms for solving the multidimensional knapsack problems. *Advanced Studies in Biology*, 4(1) :37–47, 2012.
- [Jaszkiwicz, 2002] cité page 50
Anderzej Jaszkiwicz. On the performance of multiple-objective genetic local search on the 0/1 knapsack problem - a comparative experiment. *IEEE Transactions on Evolutionary Computation*, 6(4) :402–412, 2002.

- [Jiang *et al.*, 2015] cité page 61
Siwei Jiang, Jie Zhang, Yew-Soon Ong, Allan Nengsheng Zhang, and Puay Siew Tan. A simple and fast hypervolume indicator-based multiobjective evolutionary algorithm. *IEEE Transactions on Cybernetics*, 45(10) :2202–2213, 2015.
- [Jin and Sendhoff, 2002] cité page 59, 59
Yaochu Jin and Bernhard Sendhoff. Fuzzy preference incorporation into evolutionary multi-objective optimization. In *Proceedings of the 4th Asia-Pacific Conference on Simulated Evolution and Learning*, volume 1, pages 26–30, Nanyang, Singapore, 2002.
- [Jorge, 2010] cité page 48
Julien Jorge. *Nouvelles propositions pour la résolution exacte du sac à dos multi-objectif unidimensionnel en variables binaires*. Thèse, Université de Nantes, 2010.
- [Kellerer *et al.*, 2004] cité page 47, 47, 47, 47, 47
Hans Kellerer, Ulrich Pferschy, and David Pisinger. *Knapsack Problems*. Springer Science & Business Media, 2004.
- [Klamroth and Wiecek, 2000] cité page 48
Kathrin Klamroth and Margaret M. Wiecek. Dynamic programming approaches to the multiple criteria knapsack problem. *Naval Research Logistics (NRL)*, 47(1) :57–76, 2000.
- [Knowles *et al.*, 2006] cité page 56, 68
Joshua D. Knowles, Lothar Thiele, and Eckart Zitzler. A tutorial on the performance assessment of stochastic multiobjective optimizers. Technical Report 214, Computer Engineering and Networks Laboratory (TIK), ETH Zurich, 2006.
- [Koksalan and Karahan, 2010] cité page 58
Murat Koksalan and Ibrahim Karahan. An interactive territory defining evolutionary algorithm : iTDEA. *IEEE Transactions on Evolutionary Computation*, 14(5) :702–722, 2010.
- [Korhonen and Laakso, 1986] cité page 60
Pekka J. Korhonen and Jukka Laakso. A visual interactive method for solving the multiple criteria problem. *European Journal of Operational Research*, 24(2) :277–287, 1986.
- [Kostreva and Ogryczak, 1999a] cité page 81
Michael M. Kostreva and Włodzimierz Ogryczak. *Equitable Approaches to Location Problems*. Ashgate, Brookfield, 1999.
- [Kostreva and Ogryczak, 1999b] cité page 81
Michael M. Kostreva and Włodzimierz Ogryczak. Linear optimization with multiple equitable criteria. *RAIRO-Operations Research*, 33(3) :275–297, 1999.
- [Kostreva *et al.*, 2004] cité page 81
Michael M. Kostreva, Włodzimierz Ogryczak, and Adam Wierzbicki. Equitable aggregations and multiple criteria analysis. *European Journal of Operational Research*, 158(2) :362–377, 2004.
- [Kumar and Banerjee, 2006] cité page 47, 49
Rajeev Kumar and Nilanjan Banerjee. Analysis of a multiobjective evolutionary algorithm on the 0–1 knapsack problem. *Theoretical Computer Science*, 358(1) :104–120, 2006.

- [Li *et al.*, 2015] cité page 61
Fei Li, Jianchang Liu, Shubin Tan, and Xia Yu. R2-MOPSO : A multi-objective particle swarm optimizer based on r2-indicator and decomposition. In *2015 IEEE Congress on Evolutionary Computation (CEC)*, pages 3148–3155, 2015.
- [Liefoghe *et al.*, 2008] cité page 62
Arnaud Liefoghe, Laetitia Jourdan, and El-Ghazali Talbi. Metaheuristics and their hybridization to solve the bi-objective ring star problem : a comparative study. Rapport 6515, INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE, France, 2008.
- [Liefoghe, 2009] cité page 34
Arnaud Liefoghe. *Métaheuristiques pour l'optimisation multiobjectif : approches coopératives, prise en compte de l'incertitude et application en logistique*. Thèse, Université Lille 1 - Sciences et technologies, France, 2009.
- [Liu *et al.*, 2017] cité page 60
Yiping Liu, Dunwei Gong, Xiaoyan Sun, and Yong Zhang. Many-objective evolutionary optimization based on reference points. *Applied Soft Computing*, 50 :344–355, 2017.
- [Lust and Teghem, 2010] cité page 49
Thibaut Lust and Jacques Teghem. Two-phase pareto local search for the biobjective traveling salesman problem. *Journal of Heuristics*, 16(3) :475–510, 2010.
- [Lust and Teghem, 2012] cité page 42, 47, 48
Thibaut Lust and Jacques Teghem. The multiobjective multidimensional knapsack problem : a survey and a new approach. *International Transactions in Operational Research*, 19(4) :495–520, 2012.
- [Lyes, 2008] cité page 20
Valérie Lyes. De la gouvernance des établissements à la gouvernance des territoires. Rapport, IRTS de Basse-Normandie – France, 2008.
- [Martello and Toth, 1990] cité page 47, 47
Silvano Martello and Paolo Toth. *Knapsack Problems : Algorithms and Computer Implementations*. John Wiley & Sons, Inc., 1990.
- [Martins and Santos, 1999] cité page 48
Ernesto Martins and José Santos. The labeling algorithm for the multiobjective shortest path problem. Technical Report 99/005, University of Coimbra, 1999.
- [Miettinen, 1999] cité page 32
Kaisa Miettinen. *Nonlinear Multiobjective Optimization*. Kluwer Academic Publishers, 1999.
- [Ministère de la jeunesse et des solidarités actives, 2010] cité page 17
Ministère de la jeunesse et des solidarités actives. Des activités et des métiers du secteur social, médico-social et petite enfance. Rapport, Ministère de la jeunesse et des solidarités actives, 2010.

- [Moghaddam *et al.*, 2011] cité page 82
Atefeh Moghaddam, Farouk Yalaoui, and Lionel Amodeo. Lorenz versus pareto dominance in a single machine scheduling problem with rejection. In *Proceedings of the 6th International Conference on Evolutionary Multi-criterion Optimization*, pages 520–534, Berlin, Heidelberg, 2011.
- [Ogryczak *et al.*, 2008] cité page 79
Włodzimierz Ogryczak, Adam Wierzbicki, and Marcin Milewski. A multi-criteria approach to fair and efficient bandwidth allocation. *Omega*, 36(3) :451–463, 2008.
- [Ogryczak *et al.*, 2014] cité page 79
Włodzimierz Ogryczak, Hanan Luss, Michal Pióro, Dritan Nace, and Artur Tomaszewski. Fair optimization and networks : A survey. *Journal of Applied Mathematics*, 2014 :1–26, 2014.
- [Ogryczak, 2000a] cité page 81
Włodzimierz Ogryczak. Inequality measures and equitable approaches to location problems. *European Journal of Operational Research*, 122(2) :374–391, 2000.
- [Ogryczak, 2000b] cité page 81
Włodzimierz Ogryczak. Multiple criteria linear programming model for portfolio selection. *Annals of Operations Research*, 97(1) :143–162, 2000.
- [OSORIO MONTOYA, 2015] cité page 19, 20, 23, 24, 101
Geovanny OSORIO MONTOYA. *Conception des outils pour le suivi des activités et l'aide au pilotage dans le secteur médico-social*. Thèse, Institut national des sciences appliquées de Lyon, Lyon, 2015.
- [Papadimitriou and Steiglitz, 1982] cité page 28
Christos H Papadimitriou and Kenneth Steiglitz. *Combinatorial optimization : algorithms and complexity*. Prentice Hall, 1982.
- [Paquete *et al.*, 2004] cité page 49, 62
Luis Paquete, Marco Chiarandini, and Thomas Stützle. Pareto local optimum sets in the biobjective traveling salesman problem : An experimental study. In Xavier Gandibleux, Marc Sevaux, Kenneth Sörensen, and Vincent T'kindt, editors, *Metaheuristics for Multiobjective Optimisation*, number 535 in Lecture Notes in Economics and Mathematical Systems, pages 177–199. Springer, 2004.
- [Perny and Golden, 2010] cité page 78, 80, 82, 82, 84, 97
Patrice Perny and Boris Golden. Infinite order lorenz dominance for fair multiagent optimization. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems*, volume 1, pages 383–390, Richland, SC, 2010.
- [Perny *et al.*, 2006] cité page 79, 81
Patrice Perny, Olivier Spanjaard, and Louis-Xavier Storme. A decision-theoretic approach to robust optimization in multivalued graphs. *Annals of Operations Research*, 147(1) :317–341, 2006.

- [Phan and Junichi, 2013] cité page 61, 68, 90, 90
 H.Dung Phan and Suzuki Junichi. R2-IBEA : R2 indicator based evolutionary algorithm for multiobjective optimization. In *Congress on Evolutionary Computation*, pages 1836–1845, Mexico, 2013.
- [Phan and Junichi, 2014] cité page 61
 H.Dung Phan and Suzuki Junichi. R2-BEAN : R2 indicator based evolutionary algorithm for noisy multiobjective optimization. In *Proceedings of the 2014 Seventh IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA)*, pages 1–8, 2014.
- [Pisinger, 2007] cité page 47
 David Pisinger. The quadratic knapsack problem—a survey. *Discrete Applied Mathematics*, 155(5) :623–648, 2007.
- [Qingfu Zhang and Hui Li, 2007] cité page 74
 Qingfu Zhang and Hui Li. MOEA/d : A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on Evolutionary Computation*, 11(6) :712–731, 2007.
- [Rachmawati and Srinivasan, 2006] cité page 58
 L. Rachmawati and D. Srinivasan. Preference incorporation in multi-objective evolutionary algorithms : A survey. In *IEEE International Conference on Evolutionary Computation*, pages 962–968, 2006.
- [Rachmawati and Srinivasan, 2010] cité page 59, 59
 Lily Rachmawati and Dipti Srinivasan. Incorporating the notion of relative importance of objectives in evolutionary multiobjective optimization. *IEEE Transactions on Evolutionary Computation*, 14(4) :530–546, 2010.
- [Salkin and De Kluyver, 1975] cité page 47
 Harvey M Salkin and Cornelis A De Kluyver. The knapsack problem : A survey. *Naval Research Logistics Quarterly*, 22(1) :127 – 144, 1975.
- [Sen, 1973] cité page 79, 80
 Amartya Sen. *On Economic Inequality*. Clarendon Press, Oxford, 1973.
- [Shojaei et al., 2013] cité page 47
 Hamid Shojaei, Twan Basten, Marc Geilen, and Azadeh Davoodi. A fast and scalable multi-dimensional multiple-choice knapsack heuristic. *ACM Transactions on Design Automation of Electronic Systems*, 18(4) :51 :1–51 :32, 2013.
- [Siarry and Collette, 2002] cité page 32
 Patrick Siarry and Yann Collette. *Optimisation multiobjectif*. Eyrolles, 2002.
- [Sinha and Andris A, 1979] cité page 47
 Prabhakant Sinha and Zoltners Andris A. The multiple-choice knapsack problem. *Operations Research*, 27(3) :503–515, 1979.
- [Steinhaus, 1949] cité page 79
 Hugo Steinhaus. Sur la division pragmatique. *Econometrica*, 17 :315—319, 1949.

- [Talbi, 2009] cité page 32
El-Ghazali Talbi. *Metaheuristics : From Design to Implementation*. John Wiley & Sons, 2009.
- [Tan *et al.*, 1999] cité page 60
K. C. Tan, T. H. Lee, and E. F. Khor. Evolutionary algorithms with goal and priority information for multi-objective optimization. In *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406)*, volume 1, pages 106–113, Washington, DC, USA, 1999.
- [Tan *et al.*, 2003] cité page 60
K.C. Tan, E.F. Khor, T.H. Lee, and R. Sathikannan. An evolutionary algorithm with advanced goal and priority specification for multi-objective optimization. *Journal of Artificial Intelligence Research*, 18 :183–215, 2003.
- [Tangpattanakul *et al.*, 2015] cité page 62
Panwadee Tangpattanakul, Nicolas Jozefowicz, and Pierre Lopez. A multi-objective local search heuristic for scheduling earth observations taken by an agile satellite. *European Journal of Operational Research*, 245(2) :542–554, 2015.
- [Tangpattanakul, 2015] cité page 62
Panwadee Tangpattanakul. Genetic algorithm and local search comparison for solving bi-objective p-median problem. In *Proceedings of the 2015 International Conference on Informatics, Electronics Vision (ICIEV)*, pages 1–5, 2015.
- [Thiele *et al.*, 2009] cité page 60
Lothar Thiele, Kaisa Miettinen, Pekka J. Korhonen, and Julian Molina. A preference-based evolutionary algorithm for multi-objective optimization. *Evolutionary Computation*, 17(3) :411–436, 2009.
- [Trautmann *et al.*, 2013] cité page 61
Heike Trautmann, Tobias Wagner, and Dimo Brockhoff. R2-EMOA : Focused multiobjective search using r2-indicator-based selection. In *7th International Learning and Intelligent Optimization Conference*, Lecture Notes in Computer Science, pages 70–74. Springer Berlin Heidelberg, Catania, Italy, 2013.
- [Ulungu and Jacques, 1995] cité page 48
Ekunda Lukata. Ulungu and Teghem Jacques. The two phase method : An efficient procedure to solve bi-objective combinatorial optimization problems. *Foundations of Computing and Decision Sciences*, 20(2) :149–165, 1995.
- [Ulungu and Teghem, 1994] cité page 32
E. L. Ulungu and J. Teghem. Multi-objective combinatorial optimization problems : A survey. *Journal of Multi-Criteria Decision Analysis*, 3(2) :83–104, 1994.
- [Ulungu and Teghem, 1997] cité page 48
Ekunda Lukata Ulungu and Jacques Teghem. Solving multi-objective knapsack problem by a branch-and-bound procedure. In *Multicriteria Analysis*, pages 269–278. Springer, Berlin, 1997.

- [Ulungu and Teghem, 1999] cité page 49
Ekunda Lukata Ulungu and Jacques Teghem. MOSA method : A tool for solving multiobjective combinatorial optimization problems. *Journal of Multi-Criteria Decision Analysis*, 8(4) :221–236, 1999.
- [Vianna and Vianna, 2013] cité page 47, 50
Dalessandro Soares Vianna and Marcilene de Fátima Dianin Vianna. Local search-based heuristics for the multiobjective multidimensional knapsack problem. *Production*, 23(3) :478–487, 2013.
- [Vianna *et al.*, 2014] cité page 50
Dalessandro Soares Vianna, Fermin Alfredo Tang Montané, Edwin Benito Mitacc Meza, and Carlos Bazilio Martins. A memory-based GRASP heuristic for the multiobjective multidimensional knapsack problem. *International Journal of Latest Research in Science and Technology*, 3(4) :186–194, 2014.
- [Visée *et al.*, 1998] cité page 48
M. Visée, Jacques. Teghem, M. Pirlot, and Ekunda Lukata. Ulungu. Two-phases method and branch and bound procedures to solve the bi-objective knapsack problem. *Journal of Global Optimization*, 12(2) :139–155, 1998.
- [Wagner *et al.*, 2013] cité page 57
Tobias Wagner, Heike Trautmann, and Dimo Brockhoff. Preference articulation by means of the r2 indicator. In *Evolutionary Multi-Criterion Optimization*, number 7811 in Lecture Notes in Computer Science, pages 81–95. Springer Berlin Heidelberg, 2013.
- [Wessing and Naujoks, 2010] cité page 64
Simon Wessing and Boris Naujoks. Sequential parameter optimization for multi-objective problems. In *IEEE Congress on Evolutionary Computation*, pages 1–8, 2010.
- [Wilbaut *et al.*, 2008] cité page 47
Christophe Wilbaut, Said Hanafi, and Said Salhi. A survey of effective heuristics and their application to a variety of knapsack problems. *IMA Journal of Management Mathematics*, 19(3) :227–244, 2008.
- [Zeng *et al.*, 2013] cité page 62
Rong-Qiang Zeng, Matthieu Basseur, and Jin-Kao Hao. Solving bi-objective flow shop problem with hybrid path relinking algorithm. *Applied Soft Computing*, 13(10) :4118–4132, 2013.
- [Zitzler and Künzli, 2004] cité page 54, 54, 54, 55, 56, 56, 58, 60, 61, 62, 101
Eckart Zitzler and Simon Künzli. Indicator-based selection in multiobjective search. *8th International Conference on Parallel Problem Solving from Nature (PPSN VIII)*, 3242 :832–842, 2004.
- [Zitzler *et al.*, 2003] cité page 54, 55
Eckart Zitzler, Laetitia Thiele, M. Laumanns, C.M. Fonseca, and V.G. da Fonseca. Performance assessment of multiobjective optimizers : an analysis and review. *IEEE Transactions on Evolutionary Computation*, (2) :117–132, 2003.

- [Zitzler *et al.*, 2007] cité page 60
Eckart Zitzler, Dimo Brockhoff, and Lothar Thiele. The hypervolume indicator revisited : On the design of pareto-compliant indicators via weighted integration. In Shigeru Obayashi, Kalyanmoy Deb, Carlo Poloni, Tomoyuki Hiroyasu, and Tadahiko Murata, editors, *Evolutionary Multi-Criterion Optimization*, number 4403 in Lecture Notes in Computer Science, pages 862–876. Springer Berlin Heidelberg, 2007.
- [Zitzler *et al.*, 2012] cité page 50, 81
E. Zitzler, M. Laumanns, and L. Thiele. SPEA2 : Improving the strength pareto evolutionary algorithm for multiobjective optimization. In *Evolutionary Methods for Design, Optimisation, and Control with Application to Industrial Problems*, pages 95–100. 2012.
- [Zouache *et al.*, 2017] cité page 51
Djaafar Zouache, Abdelouahab Moussaoui, and Fouad Ben Abdelaziz. A cooperative swarm intelligence algorithm for multi-objective discrete optimization with application to the knapsack problem. *European Journal of Operational Research*, 2017.

Thèse de Doctorat

Brahim CHABANE

Optimisation de plans d'actions multi-objectifs dans le secteur social et médico-social

Multiobjective action plan optimization in social and medico-social sector

Résumé

Depuis le début des années 2000, le secteur social et médico-social connaît des évolutions et des mutations importantes. D'un côté, le nombre de personnes prises en charge est en perpétuelle augmentation. D'un autre côté, les finances et les budgets mis à disposition des établissements ne cessent de se réduire, ce qui oblige les décideurs à s'adapter et à trouver de nouvelles solutions pour faire plus avec moins de moyens. Dans cette thèse, nous étudions un problème pratique auquel sont souvent confrontés les directeurs des établissements qui est l'élaboration de plans d'actions optimaux. Un plan d'actions est un ensemble d'actions qui sont mises en place afin d'améliorer à la fois les performances de l'établissement et la qualité de prise en charge de ses résidents. Élaborer un plan d'actions optimal consiste à identifier et choisir les meilleures actions qui améliorent tous les objectifs du plan tout en respectant quelques contraintes. Après la présentation du contexte pratique et théorique, nous fournissons une modélisation formelle du problème sous forme d'un problème de sac-à-dos multi-objectif. Puis nous présentons quelques méthodes de résolution à base d'indicateurs de qualité et de la dominance de Lorenz. Nous montrons que la méthode IBMOLS combinée avec l'indicateur de qualité $R2$ permet d'obtenir des solutions efficaces et d'intégrer facilement les préférences du décideur. Nous montrons également que dans un contexte où les préférences du décideur sont inconnues ou les objectifs ont tous la même importance, la dominance de Lorenz est un outil très efficace qui permet, d'un côté, d'intégrer l'équité dans le processus de recherche et, d'un autre côté, de réduire le nombre de solutions non dominées ainsi que le temps d'exécution.

Mots clés

Problème de sac-à-dos, optimisation multi-objectif, indicateurs de qualité, dominance de Lorenz.

Abstract

Since the early 2000s, the social and medico-social sector is experiencing significant evolutions and mutations. On the one hand, the number of persons taken over is constantly increasing. On the other hand, the finances and budgets available to the structures are constantly decreasing. This forces decision-makers to adapt and find new solutions to do more with fewer resources. In this thesis, we study a practical problem that is often faced by the decision-makers, which is the elaboration of optimal action plans. An action plan is a set of actions that are realized to improve both the performance of the structure and the quality of service offered to its residents. Elaborating an optimal action plan consists of identifying and selecting the best actions that improve all the objectives of the plan while respecting some constraints. After presenting the practical and theoretical context, we provide a formal modeling of the problem as a multi-objective knapsack problem. Then, we present a number of solution methods based on quality indicators and Lorenz dominance. We show that combining IBMOLS method with $R2$ indicator allows obtaining efficient solutions and easily integrating the decision-maker preferences. We also show that in a context where decision-maker preferences are not known or all the objectives are considered equals, Lorenz dominance is a very efficient tool to incorporate equity into the search process and reduce the number of non-dominated solutions as well as the algorithm runtime.

Key Words

Knapsack problem, multiobjective optimization, quality indicators, Lorenz dominance.