



HAL
open science

De la vision métier à la génération assistée de plannings pour la coordination centralisée de services de soins à domicile

Liwen Zhang

► **To cite this version:**

Liwen Zhang. De la vision métier à la génération assistée de plannings pour la coordination centralisée de services de soins à domicile. Autre [cs.OH]. Ecole des Mines d'Albi-Carmaux, 2021. Français. NNT : 2021EMAC0004 . tel-03405854

HAL Id: tel-03405854

<https://theses.hal.science/tel-03405854v1>

Submitted on 27 Oct 2021

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

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

délivré par

IMT – École Nationale Supérieure des Mines d'Albi-Carmaux

présentée et soutenue par

Liwen ZHANG

le 5 juillet 2021

Titre :

**De la vision métier à la génération assistée de
plannings pour la coordination centralisée de services
de soins à domicile**

École doctorale et discipline ou spécialité :

EDSYS : Génie Industriel et Informatique

Unité de recherche :

Centre Génie Industriel, IMT Mines Albi

Directeurs de thèse :

Hervé PINGAUD, Professeur, INU Champollion Campus d'Albi

Franck FONTANILI, Maître assistant, IMT Mines Albi

Autres membres du jury :

Virginie GOEPP, Maître de conférences, INSA de Strasbourg, Rapporteur

Christophe MERLO, Professeur, ESTIA Recherche, Bidart, Rapporteur

Sondes CHAABANE, Maître de conférences, INSA Hauts de France, Examinatrice

Maria DI MASCOLO, Directrice de recherche CNRS, G-SCOP, Grenoble, Examinatrice

Jean Pierre BOUREY, Professeur, Ecole Centrale de Lille, Président du jury

Elyes LAMINE, Maître de conférences, IMT Mines Albi et INU Champollion, Encadrant

Christophe BORTOLASO, Responsable de la Recherche, Berger-Levrault, Labège, Invité

RESUME

Aujourd'hui, qu'il s'agisse du développement du post-opératoire en ambulatoire, de la prise en charge de handicap, d'affectations de longue durée ou des personnes âgées fragiles en perte d'autonomie, la Prise en charge A Domicile (PAD) par des structures médico-sociales se développe. Le plus souvent, l'organisation de ces services à la personne a un caractère artisanal, mobilisant peu de méthodes et d'outils pour soutenir le pilotage d'un tel système de PAD. Lors de la prise en charge de services de soins et d'accompagnement, la demande est distribuée sur un territoire. Dès lors, de nouvelles capacités sont requises dans l'organisation de la prise en charge et, tout particulièrement, dans la réalité de la coordination des intervenants qui se déplacent. Face à la montée en charge rapide de la demande et de la diversité des structures de PAD, se manifeste un double besoin : (1) celui de pouvoir spécifier leur propre problème de coordination de PAD, et (2) celui de générer des plannings adaptés à ce besoin spécifique. Nous proposons donc un cadre abordant ces deux aspects pour cerner un sujet complexe. Notre étude est basée sur un environnement d'aide à la prise de décision utilisant une approche d'Ingénierie Dirigée par les Modèles (IDM). Sur la base d'un bilan de connaissances de ces PAD, les travaux de recherche visent d'abord le développement d'un langage de modélisation spécifique au domaine pour capturer le besoin de coordination. Ensuite, nous avons développé des méthodes de planification des activités selon deux formulations du problème de coordination (dit HHCRRSP - Home Health Care Routing and Scheduling Problem) : une méthode exacte en Programmation Linéaire en Nombre Mixte (PLNM) résolue par le solveur CPLEX, et une méthode approchée en satisfaction de contraintes couplée à une méta-heuristique, apte à traiter des problèmes de grande taille et résolue par OptaPlanner. Nous les avons comparées. Enfin, nous avons bâti un prototype d'une fabrique numérique de la coordination en PAD réunissant les résultats acquis, appelé *BL.Autom*. Sa conception suit les principes de l'architecture dirigée par les modèles, et son développement a été réalisé à l'aide de l'outil de méta-modélisation ADOxx. Ce prototype facilite l'exercice du métier du coordinateur en lui procurant la faculté de modifier au fil de l'eau ses orientations en termes de performance et ses exigences. Dans son usage, cette fabrique peut mettre l'accent sur la recherche de satisfaction des parties-prenantes (bénéficiaire et intervenant) dans ce prisme d'un service à la personne.

Keywords : Ingénierie dirigée par les modèles, Langage de modélisation spécifique au domaine, Optimisation, Satisfaction des parties prenantes, Prise en charge à domicile, Coordination

ABSTRACT

Nowadays, with the development of post-operative care, the management of disabilities, long duration disease or the frail elderly losing their autonomy, Home Health Care (HHC) services are developed increasingly by medical and social institutions. Mostly, these organizations have a lack of methods and tools to support the management of such a system of HHC. When providing care and assistance services, the demand of care beneficiaries is distributed over a large area, which means that there is a requirement for new capacities in the HHC services organization and in the real-life coordination of the caregivers who are mobilized to perform the care services. Due to the rapid increase in demand and the diversity of HHC institutions, a twofold need arises: (1) the ability to specify their own HHC coordination problem, and (2) the capacity to generate schedules adapted to their specific needs. Therefore, we propose a framework addressing these two aspects, based on a decision-support environment using a Model Driven Engineering (MDE) approach. On the basis of a state of the art regarding the HHC, the research work firstly aims at developing a Domain Specific Modeling Language (DSML) to capture the requirements of coordination. Then, we developed two methods for planning activities according to two formulations of the coordination problem in HHC (known as the Home Health Care Routing and Scheduling Problem - HHCRSP): one is expressed as a Mixed Number Linear Programming (MILP) formulation and solved by CPLEX, and the other is codified as a Constraint Satisfaction Problem (CSP) and solved by a metaheuristic using OptaPlanner, which is able to deal with large-scale problems. The solutions generated by these two methods are compared. Finally, we built a prototype of a digital factory for coordination in HHC that gathers our findings, which is called *BLAutom*. The conception of this prototype follows the principles of Model-Driven Architecture (MDA) and its development has been achieved through the use of the meta-modeling tool ADOxx. This prototype facilitates the coordinators' work by providing them with the ability to progressively modify their requirements and preferences in terms of performance. In practice, this tool emphasizes the search for the stakeholders' satisfaction (care beneficiary and caregiver) in the field of HHC.

Keywords: Model driven engineering, Domain specific modelling language, Optimization, Satisfaction of stakeholders, Home health care, Coordination

REMERCIEMENTS

Au cours de mes trois années de travail et de vie au sein du Centre Génie Industriel de l'IMT Mines Albi et de la Direction de la Recherche et de l'Innovation Technologique de Berger-Levrault, j'ai eu la chance de bénéficier de l'aide d'un grand nombre de personnes. Aussi, l'été 2021 était pour moi attendu avec une grande impatience tant il m'était important de m'acquitter de cette dette et de montrer ma reconnaissance à toutes les personnes dont l'intervention, tout au long de ce magnifique projet, a favorisé son aboutissement.

En premier lieu, je tiens tout particulièrement à remercier mes deux directeurs de thèse, **M. Hervé PINGAUD** et **M. Franck FONTANILI**, pour la confiance qu'ils m'ont accordé en acceptant d'encadrer ce travail, pour tout le temps qu'ils m'ont consacré, pour leurs précieuses directives ainsi que pour la qualité de leur suivi durant toute la période de ma thèse. Je me souviens encore du stress que j'ai pu ressentir lors de mon entretien avec vous il y a trois ans et de mon manque de confiance quant à ma performance. Mais aujourd'hui, grâce à votre encadrement attentif, j'ai réussi à obtenir mon doctorat et je poursuivrai ma collaboration avec vous à l'avenir dans la continuité de ces travaux de recherche.

Je tiens également à remercier mon encadrant **M. Elyes LAMINE** pour sa patience, sa confiance, mais également pour les connaissances qu'il a su me transmettre sur le système de santé en France. Ta disponibilité et tes conseils ont été d'une grande importance pour le bon déroulement de ma thèse. Travailler en ta compagnie a été un réel plaisir car, outre ton appui scientifique, tu as toujours été là pour me soutenir et me conseiller au cours de ma thèse.

Un remerciement également pour mon encadrant côté industriel, **M. Christophe BORTOLASO** et **M. Mustapha DERRAS** dont l'aide m'aura été très précieuse au cours de cette immersion dans ce monde nouveau de l'entreprise. **Christophe**, je tiens particulièrement à te remercier pour la disponibilité dont tu as fait preuve pour répondre à mes questions et pour toute l'attention professionnelle que tu as eu à mon égard avec tes conseils très utiles pour l'avancée de mes travaux.

Je remercie très sincèrement mes deux rapporteurs, **Mme Virginie GOEPP** et **M. Christophe MERLO**, et mes trois examinateurs, **Mme Sondes CHAABANE**, **Mme Maria DI MASCOLO**, **M. Jean Pierre BOUREY**, pour avoir accepté de juger ce travail et pour leur participation au jury.

Je souhaite également remercier l'équipe du Centre Génie Industriel d'IMT Mines Albi, qui m'a accueilli tout au long de la thèse. **Xavier, Jacques, Mattieu, Frédéric, Claude, Sina, Maliheh, Nafe, Sanna, Mustapha, Aurélie, Eva, Manon, Guillaume, Raphaël, Robin, Abdallah, Julien, Sébastien, Nicolas, Cléa, Jiayao...**, merci à vous tous pour les très bons moments passés ensembles et les conversations intéressantes. Et enfin merci à **Florent, Stéphanie, Katarzyna, Kévin, Florian, Lina, Camille, Ikram, Mokhtar, Hamza, Mehdi, Elise...**, mes collègues de la Direction de la Recherche et de l'Innovation Technologique de Berger-Levrault pour leur amitié. Ça a été un réel plaisir de vous fréquenter pendant toute cette période.

Remerciements

Pour finir, je tiens à remercier ma famille et mes proches, et en particulier **Heather, Thomas** et, bien sûr, ma femme **Mengzhu** pour son accompagnement en toutes circonstances tout au long de cette thèse, son écoute attentive et son soutien moral indispensable.

LISTE D'ABREVIATIONS (EN FRANÇAIS ET ANGLAIS)

A

ABC	Artificial Bee Colony algorithm
ACO	Ant Colony Optimization
ALL	ADOxx Library Language
ADL	ADOxx Development Language
ATL	ATLAS Transformation Language
AGGIR	Autonomie-Gérontologie-Groupe Iso Ressources
ANS	Agence du Numérique en Santé
ASIP	Agence des Systèmes d'Information Partagés de santé
ANAP	Agence Nationale d'Appui à la Performance

B

BPM	Business Process Management
BPMN	Business Process Model and Notation
BPRIM	Business Process-Risk management – Integrated Method
BPC	Branch-Price-and-Cut
BL	Berger-Levrault

C

CSP	Constraint Satisfaction problem
CRH	Caregivers Routing Heuristic
CIM	Computation Independent Model
CIMOSA	Computer Integrated Manufacturing Open System Architecture
COM	Component Object Model
CHU	Centre Hospitalier Universitaire
CNSA	Caisse Nationale de Solidarité pour l'Autonomie
CLS	Contrats Locaux de Santé

D

DSML	Domain Specific Modeling Language
DRL	Drools Rule Language
DGOS	Direction générale de l'Offre de Soins

E

EMF	Eclipse Modeling Framework
EHPAD	Etablissement d'Hébergement pour Personnes Agées Dépendantes

F

FFA	FireFly Algorithm
FO	Fonction Objectif
FNEHAD	Fédération Nationale des Etablissements d'Hospitalisation A Domicile

G

GA	Genetic Algorithm
GTSP	Generalized Travelling Salesman Problem

Liste d'abréviations (en français et anglais)

GRASP	Greedy Randomized Adaptive Search Procedure
GMF	Graphical Modeling Framework
GRAI/GIM	Graphe de Résultats et Activités Interreliées/ GRAI Integrated Methodology
GERAM	Generalised Enterprise Reference Architecture and Methodology
GReAT	Graph Rewriting And Transformation
GIR	Groupe Iso-Ressource

H

HHCRSP	Home Health Care Routing and Scheduling Problem
HSA	Harmony Search Algorithm
HPST	Hôpital, Patients, Santé & Territoires
HAD	Hospitalisation A Domicile

I

ISO	International Standards Organization
IDEF	Integration DEFinition
INSEE	Institut National de la Statistique et des Etudes Economiques
ISEO	Improved self-adaptive Social Engineering Optimizer
ISIS	Institute for Software Integrated Systems
IHM	Interface Homme-Machine
IDM	Ingénierie dirigée par les modèles

L

LB	Label Algorithm
----	-----------------

M

MTL	Model Transformation Language
MDA	Model Driven Architecture
MTSP	Multiple Traveling Salesman Problem
MOF	Meta Object Facility
M2M	Model to Model transformation
M2C	Model to Code transformation
MEMO	Multi-perspective Enterprise MOdeling
MIC	Model Integrated Computing
MM	Méta-Modèle
M ² M	Méta-Méta-Modèle
M-C/MM-C	Modèle Conceptuel/ Méta-Modèle Conceptuel
M-G/MM-G	Modèle Graphe/ Méta-Modèle Graphe
M-S/MM-S	Modèle orienté Solution/ Méta-Modèle orienté Solution
MAD	Maintien A Domicile

O

OMG	Object Management Group
OMILAB	Open Models Initiative Laboratory
OPL	Optimization Programming Language
OSS	Open Shop Scheduling

P

PoC	Proof of Concept
-----	------------------

PIM	Platform Independent Model
PSM	Platform Specific Model
PLM	Product Life-cycle Management
PERA	Purdue Enterprise Reference Architecture
PLNE	Programmation Linéaire en Nombres Entiers
PLNM	Programmation Linéaire en Nombres Mixtes
PAD	Prise en charge A Domicile
PPCS	Plan Personnalisé de Coordination en Santé
PRS	Projets Régionaux de Santé
<u>Q</u>	
QVT	Query / View / Transformation
<u>R</u>	
RO	Recherche Opérationnelle
<u>S</u>	
SA	Simulated Annealing
SEO	Social Engineering Optimizer
SADT	Structured Analysis and Design Technique
SSADM	Structured System Analysis Design Method
SSIAD	Service de Soins Infirmiers A Domicile
SAAD	Service d'Aide et d'Accompagnement à Domicile
SPASAD	Services Polyvalents d'Aide et de Soins A Domicile
SNAC	Service Numérique d'Appui à la Coordination
<u>T</u>	
TSP	Travelling Salesman Problem
TSPM	Travelling Salesman Problem with Multiple visits
TS	Tabu Search
<u>U</u>	
UML	Unified Modelling Language
<u>V</u>	
VRP	Vehicle Routing Problem
VRPTW	Vehicle Routing Problem with Time Windows
VNS	Variable Neighbourhood Search
VND	Variable Neighbourhood Descent
VMTL	Visual Model Transformation Language

SOMMAIRE

RÉSUMÉ	I
ABSTRACT	III
REMERCIEMENTS	V
LISTE D'ABRÉVIATIONS (EN FRANÇAIS ET ANGLAIS)	VII
SOMMAIRE	XI
CHAPITRE I. INTRODUCTION	1
1. UN UNIVERS DE SERVICES RYTHME PAR L'OFFRE ET LA DEMANDE DE SOINS.....	2
2. NOTION DE PARCOURS DE L'USAGER.....	3
3. SANITAIRE ET MEDICO-SOCIAL : EPISODES DE PRISE EN CHARGE A DOMICILE.....	5
4. PILOTAGE DE LA PRISE EN CHARGE A DOMICILE AU SEIN D'UN PARCOURS DE SOINS : LA FONCTION DE COORDINATION	7
5. EXISTANT / DEFIS ET PROBLEMATIQUES METIERS ASSOCIEES.....	12
5.1. Existant : contexte industriel	12
5.2. Défis et les problématiques métiers associées à la coordination opérationnelle	14
6. ORGANISATION DU MANUSCRIT	16
CHAPITRE II. FONDEMENTS SCIENTIFIQUES ET ÉTAT DE L'ART	19
1. INTRODUCTION	20
2. ÉTAT DES CONNAISSANCES SUR LA COORDINATION DE LA PRISE EN CHARGE A DOMICILE	20
2.1. Analyse des caractéristiques de l'offre et de la demande de services en PAD.....	20
2.2. Éléments entrant dans la prise de décision pour appairer offre et demande en PAD	26
2.3. Bilan des travaux antérieurs sur l'aide à la décision pour la coordination de PAD	37
3. CONCLUSIONS.....	47
CHAPITRE III. POSITIONNEMENT, QUESTIONS DE RECHERCHE ET JUSTIFICATION DU SUJET D'ÉTUDE	49
1. POSITIONNEMENT DE TRAVAUX DE RECHERCHE.....	50
1.1. Notion de modèle et de méta-modèle	51
1.2. Notion du langage de modélisation spécifique à un domaine (DSML)	53
1.3. Principe de l'Ingénierie Dirigée par les Modèles (IDM).....	57
2. CONCLUSION ET APERÇU DES QUESTIONS DE RECHERCHE.....	64
CHAPITRE IV. MODÉLISATION ORIENTÉE MÉTIER BASÉE SUR UN LANGAGE DE MODÉLISATION SPÉCIFIQUE AU DOMAINE (DSML)	67
1. INTRODUCTION	68
2. MODÉLISATION CONCEPTUELLE ET LANGAGE SPÉCIFIQUE.....	70
2.1. Quelques expériences utilisant un DSML	72
3. CONCEPTION DE DEUX LANGAGES SPÉCIFIQUES PARENTS.....	73
4. CONCEPTION INCREMENTALE DU DSML-HHCRSP	75
4.1. Conception préliminaire du DSML-TSP.....	75
4.2. Transition du DSML-TSP vers le DSML-HHCRSP	81
5. DÉPLOIEMENT DU DSML-HHCRSP	94
5.1. ADOxx : environnement de méta modélisation supportant la définition du DSML	95
5.2. Méthode de modélisation.....	96
5.3. Phase de validation.....	101
6. CONCLUSION.....	105

CHAPITRE V. AIDE À LA DÉCISION POUR LA COORDINATION CENTRALISÉE ET OPÉRATIONNELLE DE PAD : FORMULATION ET RÉOLUTION DU PROBLÈME D’OPTIMISATION.....	106
1. INTRODUCTION.....	107
2. DESCRIPTION DU PROBLEME D’OPTIMISATION	108
3. METHODE EXACTE BASEE SUR LA PROGRAMMATION LINEAIRE EN NOMBRES MIXTES (PLNM)	110
3.1. PIM : modèle en Programmation Linéaire en Nombres Mixtes (PLNM).....	110
3.2. Description du modèle en Programmation Linéaire en Nombres Mixtes (PLNM)	111
3.3. PSM : modèle codé en OPL pour l’appel à CPLEX.....	117
4. METHODE APPROCHEE BASEE SUR LA SATISFACTION DE CONTRAINTES (CSP).....	119
4.1. PIM : modèle en satisfaction de contrainte (CSP)	119
4.2. PSM : modèle en satisfaction de contrainte orienté OptaPlanner.....	120
5. EXPERIMENTATION : METHODE EXACTE VS METHODE APPROCHEE.....	134
5.1. Génération de jeux de test.....	134
5.2. Analyse comparative de la performance globale de la méthode exacte et de la méthode approchée ...	136
5.3. Analyse comparative sur le cas ML_s55_p28_mp14_k4	138
6. CONCLUSION	143
CHAPITRE VI. GÉNÉRATION D’UN OUTIL DE COORDINATION OPÉRATIONNELLE PAR IDM.....	145
1. INTRODUCTION.....	146
2. LA TRANSFORMATION DE MODELE A MODELE.....	148
2.1. Quelques expériences passées.....	148
2.2. Le langage de transformation des modèles.....	149
2.3. Conception générale du prototype.....	151
3. CONCEPTION DETAILLEE DE LA FABRIQUE DE LA COORDINATION.....	152
3.1. Transformation PIM-PSM	153
3.2. Transformation CIM-PIM.....	170
3.3. Résolution du modèle par CPLEX	173
4. DEVELOPPEMENT D’UN OUTIL DE TRANSFORMATION EN ADOXX : BL.AUTOM.....	177
5. CONCLUSION	180
CHAPITRE VII. CONCLUSION GÉNÉRALE	181
1. SYNTHÈSE DES CONTRIBUTIONS.....	182
2. PERSPECTIVES.....	184
2.1. Pistes d’amélioration du DSML-HHCRSP	184
2.2. Benchmarking des méthodes de résolution.....	185
2.3. Prise en compte des incertitudes par coexistence entre l’optimisation et la simulation	187
2.4. Renforcement de notre chaîne de transformation	188
REFERENCES.....	189
LISTE DES FIGURES	207
LISTE DES TABLEAUX	211
LISTE DES PUBLICATIONS	213

CHAPITRE I. INTRODUCTION

CHAPITRE I. INTRODUCTION	1
1. UN UNIVERS DE SERVICES RYTHME PAR L'OFFRE ET LA DEMANDE DE SOINS.....	2
2. NOTION DE PARCOURS DE L'USAGER.....	3
3. SANITAIRE ET MEDICO-SOCIAL : EPISODES DE PRISE EN CHARGE A DOMICILE.....	5
4. PILOTAGE DE LA PRISE EN CHARGE A DOMICILE AU SEIN D'UN PARCOURS DE SOINS : LA FONCTION DE COORDINATION	7
5. EXISTANT / DEFIS ET PROBLEMATIQUES METIERS ASSOCIEES.....	12
5.1. Existant : contexte industriel	12
5.2. Défis et les problématiques métiers associées à la coordination opérationnelle	14
6. ORGANISATION DU MANUSCRIT.....	16

1. Un univers de services rythmé par l'offre et la demande de soins

Le vieillissement de la population et l'augmentation de l'espérance de vie, telles qu'on peut les constater aujourd'hui en France et dans les autres pays développés, conduisent à un accroissement du nombre de personnes en situation de perte d'autonomie et dans des situations de fragilité (cf. la Figure 1). Quelques chiffres illustrent la gravité de ces situations : selon (Beumel and Papon, 2020), notre espérance de vie en France a désormais atteint l'âge de 85,6 ans pour les femmes et 79,7 pour les hommes. Cette augmentation de l'espérance de vie accélère le vieillissement de la population en France, nous pouvons affirmer maintenant que demain, en 2050 approximativement, 1/3 de la population française sera âgée de plus de 60 ans selon les estimations de (Tournadre, 2002). La majorité de ces personnes âgées aimerait vieillir dans de bonnes conditions d'autonomie. Elles souffrent néanmoins fréquemment de maladies chroniques avec un besoin d'assistance sur le long terme.

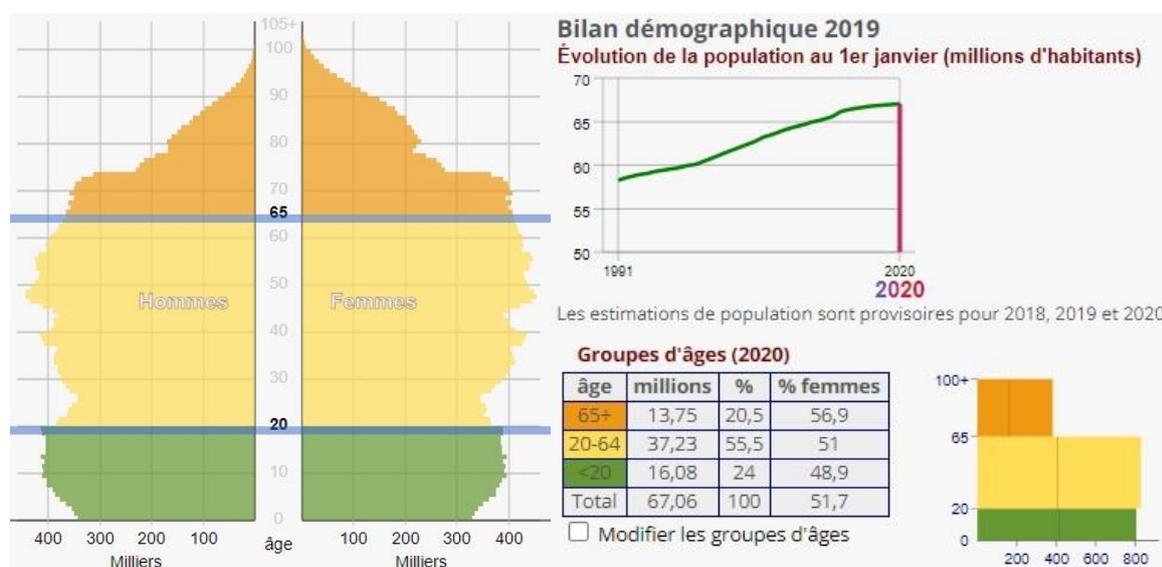


Figure 1 : Pyramide des âges au 01/01/2020 de la France métropolitaine dont la source provient de l'INSEE¹

Afin de faire face à la forte demande de soins médicaux de longue durée due au vieillissement et à l'augmentation de l'espérance de vie, le système de santé publique en France est en mutation depuis plus d'une décennie. La préservation du modèle d'assurance maladie qui prévaut en France depuis la seconde guerre mondiale est un catalyseur de cette mutation. Les déficits budgétaires chroniques deviennent difficiles à porter du fait du poids de la dette publique provenant de la prise en charge des patients à l'hôpital. Le changement politique s'est concrétisé par trois réformes de la Santé en moins de douze ans : la loi HPST (Hôpital, Patients, Santé & Territoires) en 2009², puis la loi de santé de 2016³ et aujourd'hui la stratégie « Ma santé 2022 »⁴, et ceci pour répondre à ces besoins de prévention, ou de soins, dont le but est de « soigner

¹ Pyramide des âges 2020 - France métropolitaine : <https://www.insee.fr/fr/statistiques/3312958>

² Loi n° 2009-879 du 21 juillet 2009 portant réforme de l'hôpital et relative aux patients, à la santé et aux territoires

³ Loi n° 2016-41 du 26 janvier 2016 de modernisation du système de santé

⁴ Loi n° 2019-774 du 24 juillet 2019 relative à l'organisation et à la transformation du système de santé

mieux, soigner plus, soigner ensemble ». Ce sont les maître-mots retenus par l'État, garant de la gestion rigoureuse et coopérative du système de santé français (Lamine, 2019).

En mettant le patient au cœur du projet médical, elles ont induit de nouvelles formes d'organisation des soins tendant à plus de continuité entre les épisodes de soins dans le cadre du parcours du patient. Dans (Lamine, 2019), un terme plus générique est donné au parcours du patient : « le parcours de l'utilisateur ».

2. Notion de parcours de l'utilisateur

Débutons par une analyse du concept de parcours. Selon (Lamine, 2019), ce concept signifie « une prise en charge globale et continue du patient et de l'utilisateur, dans un territoire donné au plus près de son lieu de vie, avec une meilleure attention portée à l'individu et à ses choix, nécessitant l'action coordonnée de plusieurs professionnels de santé et d'autres professionnels des champs sociaux et médico-sociaux ». Cette notion est attestée par l'importance, au cours des dix dernières années, des études dont elle a fait l'objet et des publications qui lui sont consacrées (Calvez, 2019) (SGMCAS, 2016) (Hall, 2013).

(Lamine, 2019) rappelle la définition réglementaire des trois classes de parcours qui ensemble forment un « parcours de l'utilisateur », à savoir :

- **Le parcours de soins** : Cela concerne la prise en charge d'un patient, les intervenants étant des acteurs du système sanitaire hospitalier et ambulatoire (SGMCAS, 2016). Il consiste à confier à un médecin-traitant les différentes interventions de professionnels de santé pour un même assuré, dans un objectif de rationalisation des soins. Ces interventions sont consignées dans un plan de soins. Le respect de celui-ci conditionne la prise en charge nominale des dépenses de santé.
- **Le parcours de santé** : Cela concerne la délivrance coordonnée de prestations sanitaires et sociales pour répondre aux besoins de prévention et de soins des personnes (SGMCAS, 2016). Le code de la santé publique précise que les parcours de santé visent la coordination des acteurs sanitaires, sociaux et médico-sociaux en lien avec les usagers et les collectivités territoriales, en tenant compte des spécificités géographiques, démographiques et saisonnières de chaque territoire afin de concourir à l'équité du système. Ainsi, le parcours de santé inclut le parcours de soins auquel s'ajoutent les politiques de prévention et l'éducation thérapeutique. Les parcours de santé font partie intégrante des politiques de santé et les grandes orientations sont fixées dans les Projets Régionaux de Santé (PRS). Ces orientations font ensuite place à des déclinaisons territoriales par négociation avec les élus de départements et de villes, ce sont les Contrats Locaux de Santé (CLS) qui tiennent compte des particularités des bassins de vie.
- **Le parcours de vie** : Cela concerne la réponse aux besoins de la personne dans son environnement, intégrant des interactions avec la sphère sociétale, notamment l'éducation et l'emploi. Par essence, chaque individu a un seul parcours de vie qui court de sa naissance à son décès. Un parcours de vie intègre donc les deux autres classes de parcours.

Comme illustré par la Figure 2, ces trois parcours sont traités dans un rapport récent de (ANAP, 2020). Cette agence de l'Etat y caractérise un parcours comme « une succession d'épisodes de prise en charge correspondant à une situation particulière de l'utilisateur à une période donnée de sa vie ». Deux dimensions, une temporelle et une spatiale, qualifie la mise en œuvre du parcours de l'utilisateur (ANAP, 2020) dans une optique de qualité de service :

- **Notion temporelle** : l'organisation d'une prise en charge coordonnée et organisée sur la période de temps nécessaire pour un patient ou usager.
- **Notion spatiale** : l'organisation d'une prise en charge sur un territoire, dans la proximité du domicile du patient ou usager.

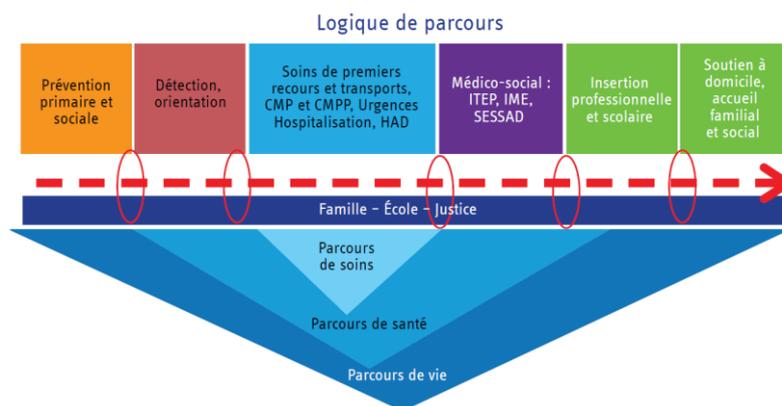


Figure 2 : Logique de parcours de l'utilisateur (ANAP, 2020) dont la source provient de l'ARS Languedoc-Roussillon (2012)¹

En ce sens, nous adoptons ce concept du parcours de l'utilisateur dans un but délibéré de placer les patients au cœur de la prise de décision pour favoriser de manière optimale leur prise en charge, tout en respectant les caractéristiques temporelles et spatiales des parcours actifs.

Les patients préfèrent très majoritairement rester chez eux, plutôt que de vivre de longs séjours à l'hôpital ou en établissement spécialisé (EHPAD² en France par exemple). Par ailleurs, l'évolution de la prise en charge des patients à l'hôpital privilégie les soins ambulatoires pour un certain nombre d'actes chirurgicaux. Une réduction du temps de séjour à une seule journée est ciblée³. Une organisation spécifique évite de mobiliser une chambre en service spécialisé induisant une diminution de charge de travail dans l'établissement et une réduction des dépenses de soins. Le patient a alors une admission en matinée et une sortie en soirée, si son état de santé à la suite de l'acte de soins le permet.

Ce sont les raisons qui expliquent en grande partie l'essor rapide de structures de prise en charge à domicile (que nous désignerons sous le sigle PAD) par des réseaux d'intervenants

¹ Les parcours de santé de l'ARS <https://www.bourgogne-franche-comte.ars.sante.fr/les-parcours-de-sante-de-lars>

² Etablissement d'Hébergement pour Personnes Agées Dépendantes

³ Chirurgie ambulatoire : une alternative à l'hospitalisation conventionnelle <https://solidarites-sante.gouv.fr/soins-et-maladies/prises-en-charge-specialisees/article/la-chirurgie-ambulatoire>

professionnels du sanitaire ou du médico-social en France. Ces structures viennent compléter celles déjà actives dans la relation à l'usager à domicile, et qui forment une force d'intervention indispensable sur le champ du médico-social.

3. Sanitaire et médico-social : épisodes de prise en charge à domicile

Selon l'Institut National de la Statistique et des Etudes Economiques (INSEE) (Blanpain and Buisson, 2016), « Ces dernières années, le maintien à domicile est devenu beaucoup plus systématique jusqu'à l'âge de 85 ans. À partir de 90 ans, au contraire, le placement en institution s'est fortement développé ». La réponse aux besoins de soutien à l'autonomie des personnes âgées est donc une volonté nationale qui inclut une organisation pour le soutien à domicile, pas nécessairement en prodiguant des soins, mais en délivrant les services qui contribuent à la prévention des maladies et au bien-être des personnes.

Des études sur le fonctionnement des hôpitaux montrent qu'ils doivent agir sous une pression de plus en plus forte (Belorgey, 2016). Le budget étant gelé, l'engorgement des lits et des salles d'opération augmente. Ceci est amplifié encore une fois par les besoins de prise en charge de certains patients requérant des soins simples, mais nombreux ou réguliers. Le rapport de l'INSEE (Kubiak, 2012) met en lumière ce besoin. Ainsi, la tendance est de prendre en charge les patients directement à leur domicile. L'évolution est notamment visible via les chiffres-clé de la Fédération Nationale des Etablissements d'Hospitalisation A Domicile (FNEHAD) (cf. la Figure 3). Ce n'est plus au patient de venir à l'hôpital, mais à des intervenants spécialisés de se déplacer chez les bénéficiaires. Dans les faits, ce service est souvent proposé par des structures de prise en charge à domicile.



Figure 3 : Chiffres clé de l'HAD 2019 dont la source provient du FNEHAD¹

En France, cinq catégories de structure existent pour assurer la continuité, la qualité et la sécurité de la prise en charge des patients à leur domicile² :

(1) Hospitalisation A Domicile (HAD)

L'hospitalisation à domicile (HAD) représente le niveau d'intervention médical le plus haut (Rejeb, 2013). Selon le Décret n°92-1101 du 2 octobre 1992³ : « les structures d'hospitalisation à domicile permettent d'assurer au domicile du malade, pour une période limitée, mais révisable en fonction de l'évolution de son état de santé, des soins médicaux et paramédicaux continus et

¹ Chiffres clé de l'HAD 2019 : <https://www.fnehad.fr/chiffres-cles/>

² Site du ministère des solidarités et de la santé : <https://solidarites-sante.gouv.fr/soins-et-maladies/prises-en-charge-specialisees/had-10951/article/intervention-conjointe-d-had-et-d-un-ssiad>

³ Décret n°92-1101 du 2 octobre 1992 relatif aux structures de soins alternatives à l'hospitalisation mentionnées à l'article L. 712-2 du code de la santé publique

nécessairement coordonnés. Les soins en HAD se différencient de ceux habituellement dispensés à domicile par la complexité et la fréquence des actes ».

(2) Service de Soins Infirmiers A Domicile (SSIAD)

Selon le Décret n°81-448 du 8 mai 1981¹, « les services de soins à domicile assurent, sur prescription médicale, aux personnes âgées ou dépendantes, les soins infirmiers ou d'hygiène générale, les concours nécessaires à l'accomplissement des actes essentiels de la vie, ainsi qu'éventuellement d'autres soins relevant d'auxiliaires médicaux ».

La forme de soins proposée par le SSIAD est moins lourde que dans celui d'une HAD. Ces soins visent généralement à éviter l'hospitalisation des personnes âgées et à retarder un placement en maison de retraite.

(3) Maintien A Domicile (MAD)

Selon (Rejeb, 2013), le maintien à domicile « dispense en général des services sociaux et des soins de faible intensité : des soins de nursing habituellement réalisés par des aides-soignants avec une légère intervention médicale. La personne qui bénéficie du MAD peut choisir ses propres prestataires et le coût des soins est individualisé par acte et par produit. De nombreux prestataires proposent des services de MAD ». Aucune définition juridique n'existe, d'après nos recherches.

(4) Service d'Aide et d'Accompagnement à Domicile (SAAD)

Selon la loi n°2015-1776 du 28 décembre 2015², les services d'aide et d'accompagnement à domicile sont « les organismes privés ou publics qui interviennent en qualité de prestataires, pour des actions liées aux actes essentiels de la vie, au soutien à domicile, à la préservation ou la restauration de l'autonomie dans l'exercice des activités de la vie quotidienne et au maintien et au développement des activités sociales et des liens avec l'entourage. Ils accompagnent notamment les personnes âgées dépendantes, les personnes handicapées ainsi que les familles fragilisées ». Par définition, nous pouvons dire que le MAD est un sous-groupe du SAAD.

(5) Services Polyvalents d'Aide et de Soins A Domicile (SPASAD)

Selon l'Arrêté du 30 décembre 2015³, les services polyvalents d'aide et de soins à domicile constituent une nouvelle forme d'organisation des services à domicile. Ces services « apportent à la fois un accompagnement dans les actes de la vie quotidienne et des soins aux personnes prises en charge. Ils regroupent donc des services qui assurent les missions d'un SSIAD et les missions d'un SAAD ». Les services fournis par les SPASAD permettent aux bénéficiaires de ne plus être obligés de faire appel à deux services. L'accompagnement est fluide et sécurisant. Les bénéficiaires disposent d'un interlocuteur unique pour la mise en place de l'intervention et son suivi, qui sont assurés par une même équipe (CNSA, 2020).

¹ Décret n°81-448 du 8 mai 1981 relatif aux conditions d'autorisation et de prise en charge des services de soins à domicile pour personnes âgées

² Loi n°2015-1776 du 28 décembre 2015 relative à l'adaptation de la société au vieillissement.

³ Arrêté du 30 décembre 2015 fixant le cahier des charges des expérimentations relatives aux services polyvalents d'aide et de soins à domicile prévues à l'article 49 de la loi n° 2015-1776 du 28 décembre 2015 relative à l'adaptation de la société au vieillissement

Ce rapide panorama des structures de PAD en France dessine une architecture peuplée d'organisations locales et adaptée à cette forme de prise en charge. Cela rend tangible la possibilité d'inscrire l'usager de services de santé dans un parcours. En l'occurrence, ce parcours d'usager est principalement décliné comme un parcours de soins coordonné, dans un système de santé tourné principalement vers une logique de médecine curative. Un besoin de pilotage est ici explicité par cet adjectif « coordonné ». Il date de la réforme de l'assurance maladie de 2004 qui a imposé le médecin traitant dans un rôle de pilote de parcours. Depuis, lentement, l'organisation du système de santé vise une continuité et une meilleure synergie entre intervenants impliqués dans un parcours.

C'est un constat, sur le terrain, la réalité opérationnelle du parcours de soins coordonné tarde à être en ligne avec ces directives institutionnelles. En conséquence, pour répondre au mieux aux demandes de PAD, ces structures compétentes sont directement et très concrètement confrontées à des difficultés de management en termes de définition et de pilotage d'activités, susceptibles de limiter la qualité du service rendu à la personne. C'est le fondement de la coordination qui est obligatoire dans ce type d'organisation non hiérarchique.

C'est dans ce contexte que se situe notre recherche. Notre sujet aborde l'aide à la décision pour l'organisation du maintien et/ou des soins à domicile de **bénéficiaires**. Nous étudions plus particulièrement le thème de la coordination des **intervenants** (professionnels de santé, professionnels du social, entourage) impliqués dans ce type de dispositif.

4. Pilotage de la prise en charge à domicile au sein d'un parcours de soins : la fonction de coordination

L'inscription de la coordination dans l'article 1 de la loi de santé de 2016¹, atteste de son importance pour ce secteur d'activités. Ainsi, nous restituerons d'abord une synthèse des recherches en rapport avec ce concept de coordination au sens large. Dans la théorie de la coordination (Malone and Crowston, 1994), la coordination consiste à gérer les dépendances entre des activités dans une finalité explicite. Les auteurs avancent cinq grandes catégories de dépendance :

- Liées aux partages de ressources.
- Liées aux affectations des tâches.
- Liées à des relations entre producteur/consommateur
- Liées à des contraintes de simultanéité.
- Liées à la décomposition d'une tâche en sous-tâches ou d'objectifs en sous-objectifs.

Le Tableau 1 tente une projection de ces dépendances dans la PAD.

¹ Loi n° 2016-41 du 26 janvier 2016 de modernisation du système de santé

Tableau 1 : Exemples de coordination dans la PAD selon les cinq catégories de dépendance définies dans la théorie de la coordination (Malone and Crowston, 1994)

Catégories de dépendances	Exemples de coordination dans la PAD
Partage de ressources	Interventions exigeant la mobilisation d'un même intervenant chez plusieurs bénéficiaires
Affectation des tâches	Affectation des ressources adaptées aux interventions
Relation entre producteur et consommateur	Préférence et incompatibilité entre les professionnels de santé et les bénéficiaires
Contrainte de simultanéité	Interventions exigeant la mobilisation de plusieurs intervenants chez un même bénéficiaire
Décomposition	Décomposition du plan de soins personnalisé du bénéficiaire en interventions

Dans cette acception, la coordination rejoint un exercice de conception de processus métier où il faut réunir les activités par leurs flux d'entrée et de sortie, puis fixer les conditions liées à leur exécution dans le temps (conditions de début d'activité et conditions de fin), c'est-à-dire à la structure de contrôle de l'exécution des activités. Dans le secteur de la santé, elle est identifiée comme un levier essentiel et stratégique pour l'évolution du système de santé et l'optimisation de son fonctionnement. (McDonald et al., 2007) ont délivré une définition de la coordination spécifique à ce domaine : « la coordination est l'organisation délibérée des activités de soins à un patient par plusieurs acteurs du système de santé. Elle implique la mobilisation de personnels et d'autres ressources. Elle est souvent opérée par des échanges d'informations entre les parties prenantes des soins ». Plus récemment, (Lamine, 2019) définit la coordination comme un processus décisionnel d'agencement des acteurs et des activités dans le temps et dans l'espace reposant sur un partage, un échange de connaissance et une délibération, dans le but de satisfaire le bénéficiaire ultime (cf. la Figure 4). Effectivement, la coordination dans le système de la prise en charge à domicile consiste à satisfaire la demande de services de soins et d'accompagnement en envoyant les bonnes informations aux bons intervenants à un bon moment et au bon endroit. A cette fin, cet auteur évoque trois formes de coordination :

- Par ajustement mutuel. C'est une forme de coordination lâche où le parcours se construit au fil de l'eau essentiellement par l'effet des échanges entre intervenants, et par des séquences de microdécisions prises dans le flux des activités, sans réelle vision du futur autre qu'immédiat. Cette forme est probablement celle qui est la plus utilisée sur le terrain par les réseaux médico-sociaux. Elle absorbe naturellement les effets des incertitudes.
- Par standardisation. Elle suppose l'établissement d'un protocole d'interventions défini de manière générique et pensé pour couvrir tous les cas. Cette conception du protocole fait l'objet d'un travail de groupe. Cette forme de coordination obéit aux canons des démarches qualité où l'on prescrit ce qu'il faut faire bien avant la mise en opération d'un parcours. Tout écart au standard défini est expliqué et tracé. C'est souvent cette démarche qui est suivie par les établissements hospitaliers sur leur territoire d'exercice.
- Par centralisation. La coordination sous cette forme est celle qui passe par l'établissement de plans pour dérouler les parcours en respect des plans de soins. C'est une forme qui hérite de la standardisation en ce qu'elle considère un processus métier, mais il n'est plus standardisé. Elle hérite aussi de l'ajustement mutuel dans la mesure où

un parcours d'intervenant est équilibré pour satisfaire les exigences d'affectation aux parcours des usagers. C'est cette forme de coordination que nous traitons dans ce mémoire.

La Figure 4 montre ce à quoi aboutit l'exercice de coordination. Il s'agit de deux plannings qui sont maintenus en cohérence. Celui des interventions (en haut) décrit la réponse à la demande. Celui du bas décrit la mobilisation de l'offre de services en réponse à cette demande. Chaque partie est en lien avec la satisfaction d'une partie prenante, respectivement le patient usager pour le plan supérieur, puis l'intervenant pour l'inférieur.

Chaque forme de coordination va imposer sa propre méthode d'élaboration de ce résultat. L'ajustement mutuel consiste à ajouter une petite tranche verticale qui est le futur immédiat dans un montage chronologique de ces plans. La standardisation conduit à placer des patrons d'activités sur la partie supérieure et à appliquer des règles prédéfinies pour la partie inférieure. Le plan se construit comme un puzzle avec les pièces connues. Enfin, la centralisation donne à un coordinateur le rôle de planificateur, c'est-à-dire qu'il doit calculer le plan prévisionnel en fonction de la demande et de l'offre sur un horizon conséquent.

Le processus décisionnel ne mobilisera pas les activités constitutives (partage, échange, délibération) de la même manière, ni dans le même ordre, selon la forme de coordination.

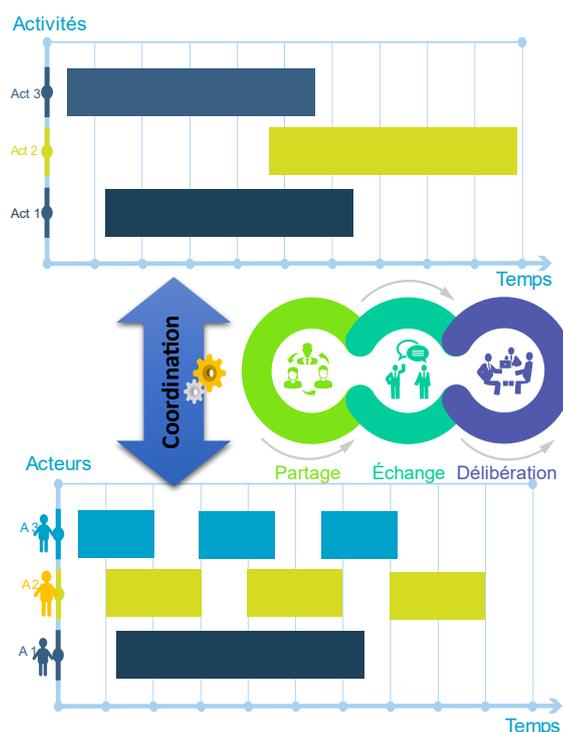


Figure 4 : Représentation synthétique de l'exercice de coordination (Lamine, 2019)

Il s'agit donc de prendre un ensemble des décisions pour définir les conditions d'interventions d'un pool d'acteurs mobilisés dans une structure non hiérarchique, dont il faut harmoniser les activités pour produire un bouquet de services de proximité au patient. La cohérence de ces décisions se juge principalement par la conformité à la spécification des soins que le patient doit recevoir et par l'engagement accepté des intervenants professionnels dont il faut satisfaire les conditions de travail (charge de travail, délais entre chaque acte, etc.).

Sur cette base, la théorie du pilotage hiérarchisé des systèmes de production s'applique aisément à de telles structures de santé. (Gourgand, 2008) a rappelé le bien-fondé de ces trois niveaux de prise de décision avec des horizons de temps de portée différente pour fixer un cadre décisionnel à l'organisation d'une PAD : opérationnel, tactique et stratégique. Au titre de la coordination dans la santé, (Lamine, 2019) définit trois niveaux de décision selon le cadre et l'horizon considéré :

- Coordination stratégique : le niveau institutionnel (niveau macro) pour lequel la coordination consiste en l'établissement de liens de partenariat entre les différentes parties-prenantes du système de santé, que ce soit à l'échelle locale (ville, département ou région) ou à l'échelle nationale. C'est le cadrage géographique de la mission sur un temps long.
- Coordination tactique : le niveau de la structure ou du dispositif de prise en charge (niveau méso) pour lequel la coordination réside en la mobilisation et à l'organisation des moyens humains et matériels pour assurer les missions de la structure. Il s'agit de constituer les forces en place pour assurer la mission sur un horizon de moyen terme.
- Coordination opérationnelle : le niveau du patient ou de la personne (niveau micro) pour lequel la coordination consiste en l'organisation des actes de soin et d'accompagnement de chaque personne. Il s'agit de mobiliser ces forces compétentes pour répondre à la demande au quotidien.

Nous pensons que le concept de coordination a un centre de gravité situé sur **le niveau opérationnel** des décisions (Lanzarone and Matta, 2014) (Matta et al., 2014) (Rest and Hirsch, 2016) (Shao et al., 2012), dans une période de temps pour laquelle les équipes sont clairement dimensionnées et le portefeuille de patients apte à délivrer une commande claire de services de soins. C'est le poids des incertitudes sur l'ensemble des données caractéristiques de l'offre et de la demande qui pousse ce centre de gravité au plus proche du terrain. Dans un système distribué et dans le cadre d'une économie de la santé où organisations publique et privée sont engagées, c'est aussi la strate de coordination la plus stable.

Nous avons donc choisi de limiter notre périmètre d'études à la coordination centralisée au niveau opérationnel. Le coordinateur (ou le décideur d'une structure de PAD) a en charge la production de cette coordination centralisée pour piloter les ressources productives.

Selon (ANAP, 2020), le coordinateur d'une structure de PAD a pour missions d'établir, d'organiser et de gérer des liens permettant l'optimisation de la prise en charge sanitaire, sociale et médico-sociale d'une personne dans un parcours de soins, ainsi que de coordonner la réalisation d'une démarche pluridisciplinaire selon le projet de soins établi¹.

¹ Référentiel des fiches-métiers de la santé : <https://solidarites-sante.gouv.fr/metiers-et-concours/les-metiers-de-la-sante/les-fiches-metiers/#c>

Il existe plusieurs catégories de problèmes relativement classiques qui s'entremêlent dans cette spécification de ce qu'est la prise de décision pour la coordination au niveau opérationnel d'une structure PAD. En s'inspirant du vocabulaire de la gestion de production, il apparaît deux types de problèmes :

- L'affectation des ressources
- La planification de tournées.

Selon (Malapert, 2006), ils se définissent comme suit :

- **Affectation des ressources** qui consiste à assigner une activité (intervention) à une ressource (intervenant) de manière univoque en fonction d'un objectif.
- **Planification de tournées** qui est un mélange de besoins d'ordonnancement et de routage. Un problème d'ordonnancement consiste à organiser dans le temps la réalisation des activités, compte tenu de restrictions temporelles et de la disponibilité des ressources requises. Un problème de routage a pour but de trouver un ensemble de routes (tournées) pour une flotte d'intervenants actifs disponibles à partir d'une ou plusieurs bases de départ et de retour (structure de PAD), afin de satisfaire les demandes d'un ensemble de bénéficiaires.

Dans ce premier périmètre définissant notre approche de la coordination, nous avons émis trois **hypothèses**.

- (1) Nous étudierons la coordination des interventions à domicile des multiples intervenants impliqués dans la prise en charge des patients ou des usagers (désignés ci-après par le terme de bénéficiaires) gérés par des structures de type HAD, SSIAD, MAD, SAAD et SPASAD. Il est à noter que les autres cas de coordination, essentiellement associés à des directives de personnes isolées (tels qu'une infirmière libérale, un médecin traitant, un membre de la famille) ne sont pas couverts par ces travaux de thèse.
- (2) Pour ces structures de prise en charge à domicile, la taille maximale du portefeuille des usagers est connue, ainsi que le territoire sur lequel la PAD va s'exercer. C'est en général des choix stratégiques qui sont pris pour une longue durée et qui président à la recevabilité des demandes de prise en charge par la structure.
- (3) La composition de l'équipe de professionnels intervenant est aussi supposée connue. C'est le dimensionnement de l'offre de services à la personne qui découle de la connaissance du volume global de la demande. Elle est déterminée sur des choix tactiques. Sur un horizon de court terme, cette composition entre dans la formulation du problème. Il faut faire se rencontrer l'offre (dont la capacité maximale est alors connue) et la demande (dont le volume de charge est alors contraint), c'est à dire faire les affectations des intervenants sur les plans de soins de chaque bénéficiaire et calculer la tournée quotidienne de chaque intervenant. Retenons qu'une tournée est une suite ordonnée d'interventions à domicile, réalisée par un type donné d'intervenant.

5. Existant / défis et problématiques métiers associées

5.1. Existant : contexte industriel

Experte des métiers des services publics, la société Berger-Levrault est spécialisée dans la réalisation de logiciels pour des administrations publiques locales dans les secteurs de l'éducation, de la santé, du sanitaire, du social et de la gestion des territoires. Avec 51 000 clients et 1 800 collaborateurs en France, en Espagne, au Canada et au Maroc, Berger-Levrault est historiquement une société d'impression de documents administratifs, fondée entre 1676 et 1681 à Strasbourg, et qui existe donc depuis plus de 6 siècles (de 1463 à aujourd'hui)¹. Ces logiciels participent à la modernisation de l'action publique et à la simplification des processus administratifs par l'offre de solutions numériques aux agents de l'État et aux usagers des services publics.

Dans le domaine de santé, Berger-Levrault a l'ambition d'être durablement le premier éditeur de logiciels proposant une offre globale et intégrée à tous les acteurs publics et privés de la santé. Les solutions de Berger-Levrault s'adressent aux établissements hospitaliers (CH, CHU et cliniques) aux établissements de soins médico-sociaux (handicap, social, EHPAD).

L'offre de produits de Berger-Levrault a une large couverture fonctionnelle pour le secteur de la santé et du médico-social. Elle s'étend sur deux grands ensembles :

- (1) L'ensemble des fonctions de gestion de l'établissement (e.g. actes administratifs, finance, RH, etc.) constitue une première suite de solutions. Ce premier ensemble inclut la gestion des résidents/patients/usagers.
- (2) Le dossier de soins, le dossier médical et l'action sociale forment un second ensemble de solutions pour des utilisateurs qui sont des praticiens du monde de la santé et du social.

Pour préserver cette position dominante et anticiper les besoins nouveaux du marché sur la base de ce capital accumulé par une offre continuellement adaptée au secteur de la santé et du social, Berger-Levrault veut prendre l'initiative d'étoffer son catalogue avec des produits adaptés aux besoins nouveaux soulevés par la mise en place de la prise en charge à domicile, et plus généralement la coordination au niveau opérationnel entre toutes les parties prenantes autour du concept de parcours de soins, de santé et de vie. C'est un axe stratégique de son développement.

Actuellement, au sein de ce catalogue, Berger-Levrault diffuse un logiciel qui porte le nom de « BL.domicile² », proposant des solutions expertes pour une couverture métier complète et évolutive des besoins de SAAD, SSIAD et SPASAD. BL.domicile cible les services de soins et d'accompagnement à domicile en fournissant une gestion complète des plannings d'activités et un dossier médico-social du patient, dit dossier de liaison partagé (cf. la Figure 5). L'objectif du développement futur de BL.domicile est de renforcer la qualité des interventions, et corrélativement d'accélérer le processus administratif de facturation vis-à-vis de financeurs

¹ Berger-Levrault : notre histoire de 1463 à aujourd'hui : <https://www.berger-levrault.com/fr/groupe-berger-levrault/notre-histoire-de-1463-a-aujourd-hui/>

² BL.domicile: <https://www.berger-levrault.com/fr/produit/bl-domicile/>

multiples. La mission de R&D de BL se concentre sur le premier objectif et doit contribuer à son atteinte. Il se décline en des développements de BL.domicile afin de renforcer la qualité des prestations de services des structures de PAD. Dans BL.domicile, un pari est que l'exhaustivité du dossier de liaison partagé associé à une application mobile dédiée aux intervenants facilitera la coordination et assurera aux coordinateurs une capacité de suivi en temps réel des prestations d'aide et de soins à domicile.

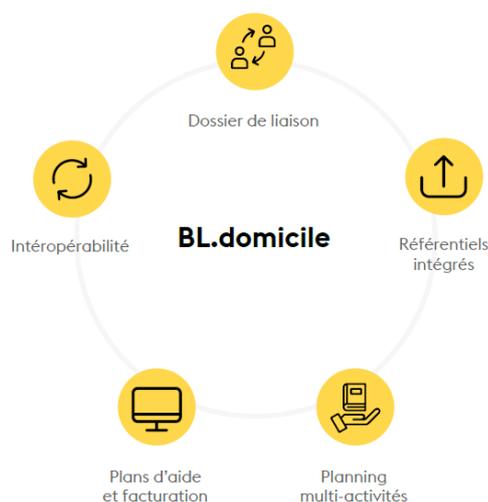


Figure 5 : Couverture fonctionnelle de « BL.domicile »

Berger-Levrault soutient depuis plusieurs années une dynamique de développement de son activité de Recherche et Innovation pour fournir à ses clients des produits innovants. L'entreprise s'est emparée de nombreux sujets pour relever les défis de la gestion de plannings et de la gestion de flux, grâce notamment à l'intelligence artificielle et aux algorithmes d'optimisation. A cette fin, Berger-Levrault déploie une logique de partenariat avec des laboratoires de recherche publics compétents pour promouvoir une recherche finalisée en lien avec sa vision stratégique. Les perspectives sont un renforcement de la position de l'éditeur auprès de tous les acteurs impliqués dans ces organisations de santé marquées par ce trait d'une dynamique de fonctionnement en réseau.

Ainsi, dès 2015, de nombreuses initiatives ont été amorcées, dont les travaux de cette thèse font partie. Comme nous le soulignons dans la section 1 de ce chapitre, le sujet a des implications sociétales bien plus larges que les simples intérêts industriels, telles que la volonté de permettre aux personnes âgées de vieillir dans de bonnes conditions d'autonomie, par exemple. Mais aussi de fournir des moyens d'exercer une responsabilité sociale en faisant en sorte de satisfaire des professionnels de santé qui sont en tension permanente. Ces enjeux sociétaux sont centraux pour l'éditeur et relèvent d'une réponse globale, par la science et les technologies du numérique, à un risque très important d'engorgement de notre système de santé. En ce sens le contexte industriel de cette thèse offre un « terrain » qui n'est pas prospectif, il est question de collaboration sur un sujet particulièrement contextuel et actuel avec des retombées attendues à court terme de nos travaux de recherche dans les futures versions de BL.domicile.

5.2. Défis et les problématiques métiers associées à la coordination opérationnelle

Sans la coordination des intervenants de telles structures de PAD, « on perd de la qualité, on perd des chances, cela crée des dysfonctionnements, des événements indésirables, parce que les informations n'ont pas été échangées ou partagées ou qu'elles l'ont été, mais pas au bon moment », explique Philippe Burnel, délégué à la Stratégie des Systèmes d'Information de Santé au sein du Ministère de la Santé et des Affaires Sociales (PSAD, 2015), intégré depuis à l'Agence du Numérique en Santé (ANS) par fusion avec l'Agence des Systèmes d'Information Partagés de santé (ASIP) .

A l'ère du numérique, la réponse à des besoins de coordination dans le cadre d'un fonctionnement en réseau d'acteurs professionnels, mobilise les technologies de l'information et de la communication afin de mieux faire face à la complexité des situations et pour trouver des réponses efficacement. Cependant, peu d'éditeurs se sont engagés aujourd'hui dans une offre à la hauteur des enjeux et avec une couverture fonctionnelle suffisante pour tenir compte de la diversité des situations et des besoins, même pour le produit BL.domicile. En regard des contextes métiers évoqués dans les sections précédentes, une série de défis s'imposent au pilotage des structures de PAD :

- (1) **Représentation de la PAD dans un univers de services à la personne.** Cela factorise le spectre des 5 catégories de structures de PAD présentées dans la section 3 de ce chapitre.
- (2) **Diversité organisationnelle et fonctionnelle des structures de PAD.** Cette diversité se reflète dans le fait que chaque structure a ses propres règlements, modes de gestion et processus opérationnels.
- (3) **Coordination centralisée de professionnels de soins aux diverses compétences.** Le coordinateur reste au cœur dans ce système distribué (la prise en charge à domicile) pour piloter les ressources productives des soins aux diverses compétences.
- (4) **Satisfaire au mieux les usagers et les prestataires des services.** Les intervenants et les bénéficiaires ont des préférences et des personnalités particulières. C'est une prise en compte du facteur humain à sa juste proportion.
- (5) **Prendre en compte plusieurs sources d'incertitude pesant sur les activités.** La prise en compte de l'incertitude dans la recherche d'une solution de coordination opérationnelle a pour but ultime d'avoir une capacité de réaction face à tout forme d'évènement incertain.

Dans la suite, nous proposons trois classes de problèmes associées à ces défis :

- **La qualification des intervenants.** Cette dimension de problèmes est associée au défi (3). Elle porte sur :
 - La connaissance des ressources disponibles et du rôle de chacun.
 - L'implication des différents intervenants dans la prise en charge d'un patient.
 - Le suivi des exécutions des interventions lorsqu'il y a interdépendance entre les interventions.
- **Les lieux de délivrance des services aux usagers.** Cette dimension de problèmes est associée aux défis (3) et (5). Elle porte sur :

- La diversité du moyen de transport entraînant la différence des coûts de transport et des temps de déplacement.
- L'organisation des tournées renvoyant vers les règles de gestion en vigueur, par exemple, l'organisation en nombre de tournées fixes ou variables, les ressources actives limitées ou illimitées, l'optimisation de la qualité ou du coût de la tournée.
- La prise de décision parmi des itinéraires multiples entre les lieux de délivrance des services aux usagers.
- Les perturbations imprévues durant les déplacements des intervenants, par exemple, une panne de véhicule, un embouteillage.
- **La qualité permanente de l'offre de services répondant à la demande.** Cette dimension de problème est associée aux défis (4) et (5). Elle porte sur :
 - La personnalisation des plans de soins
 - La personnalisation et la satisfaction du bénéficiaire
 - Le changement des disponibilités d'intervenants
 - L'annulation des services à des bénéficiaires

Dans la Figure 6, nous alignons trois des cinq défis identifiés aux trois classes de problèmes de la coordination opérationnelle dans la PAD. Cette répartition conduit une **transformation de connaissances métiers vers un environnement numérique de coordination** afin de soutenir les coordinateurs (principalement les coordinateurs) de structures variées (associées aux défis (1)(2)) à agir par une juste prise en compte des trois catégories de décision qui leur incombent : l'affectation des ressources, la planification de tournées (et donc le routage et l'ordonnancement).

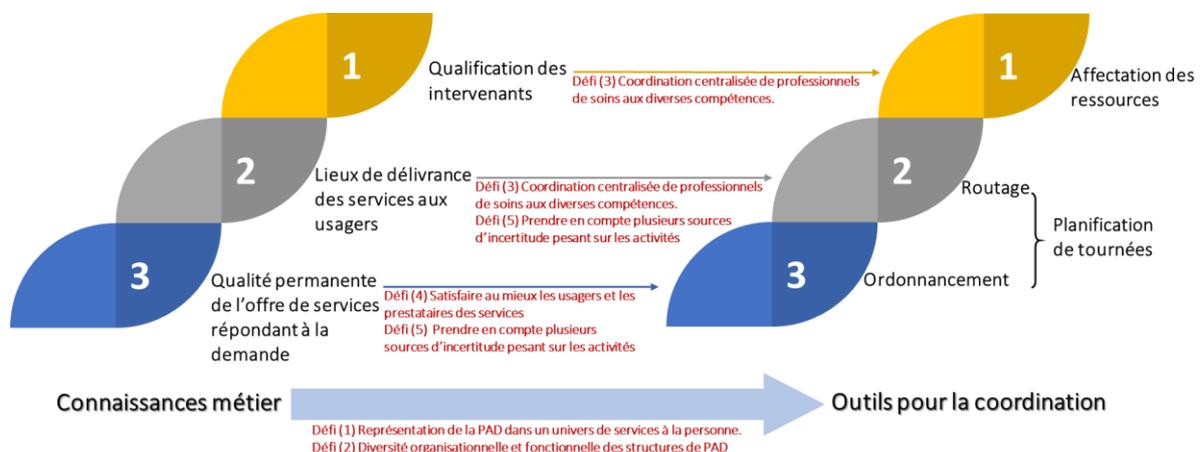


Figure 6 : Alignement des défis sur les classes de problèmes de coordination opérationnelle

Actuellement, il n'y a que peu de systèmes d'information utilisés dans les structures de PAD qui possèdent des fonctions aptes à soutenir ces besoins de coordination. En effet, la coordination formule des problèmes qui demandent de nombreuses connaissances sur le système opérant et les moyens associés, d'une part, et sur la disponibilité des informations et des ressources, d'autre part. Aujourd'hui, ces facteurs limitants tendent à se résorber et une maturité des organisations est constatée vis-à-vis de ce sujet. Ainsi, nous soulignons l'absence de services de planification pour développer de l'intelligence dans l'organisation.

Cette dernière affirmation trouve sa source dans l'initiative de création des Services Numériques d'Appui à la Coordination, lancée par l'Etat dans le cadre du virage numérique de « Ma Santé 2022 ». Argumentons cela à l'aide de la feuille de route du programme national « e-Parcours ». Une instruction de la DGOS (Direction générale de l'Offre de soins) datant de mai 2019 décrit le niveau de dépendance du programme e-Parcours à d'autres chantiers ouverts par l'Etat. Il s'agirait de rassembler dans un bouquet de services, dits Service Numérique d'Appui à la Coordination (SNAC) : (1) un dossier de coordination, (2) un annuaire propre aux intervenants, (2) un plan personnalisé de coordination en santé, (3) des mécanismes de repérage et d'alerte de situations critiques, (4) un carnet de liaison, (5) un réseau social et une messagerie sécurisée réservés aux professionnels, (6) un partage d'agenda, (7) un soutien logistique de la trajectoire de patient mobilisant des connaissances officielles et actualisées des ressources.

Aucune de ces 7 composantes n'entrent réellement dans la mécanique d'intégration que nous préconisons avec un outil numérique au service d'une coordination centralisée. Les SNAC sont avant tout pensés comme une base forte pour soutenir une coordination par ajustement mutuel. C'est déjà une ambition forte. Mais la coordination ne peut se réduire, nous l'avons expliqué, à du partage et de l'échange, notre ambition est de soutenir la délibération avec des outils numériques puissants mettant à disposition des fonctions de planification.

Nous montrerons par la suite pourquoi c'est un problème complexe, et ce qui justifie de parler d'ambition.

6. Organisation du manuscrit

Ce mémoire se compose de six chapitres. Après la description du contexte, du périmètre des études ainsi que l'exposé des problématiques métiers associés aux défis identifiés dans le **chapitre I**, le reste du document est organisé comme suit.

Le **chapitre II** est dédié aux fondements scientifiques, et à l'existant au commencement de nos travaux. Il débute avec un rappel de la théorie de la coordination et se prolonge avec sa définition appliquée au cadre de la PAD. Puis, nous effectuons une analyse approfondie de concepts relatifs à l'offre et la demande de services de PAD, et plus particulièrement sur le pilotage de la prestation de services. Dans ce pilotage de système distribué, la coordination pour l'aide à la décision dans la PAD est essentielle. Ces problèmes de coordination sont connus sous l'appellation HHCRSP (*Home HealthCare Routing and Scheduling Problem*). Nous en faisons une analyse bibliographique et inscrivons notre étude dans le cadre d'une coordination opérationnelle dont le soutien par le numérique est encore perfectible sur le terrain.

Dans le **chapitre III**, nous affichons l'ambition de nos travaux de recherches en posant les trois questions de recherche que nous traiterons. Nous situons ces questions de recherche sur des verrous scientifiques à lever. Ainsi, c'est dans le cadre de l'architecture dirigée par les modèles (MDA, avec ses trois niveaux CIM-PIM-PSM) que nous inscrivons les pistes étudiées pour répondre à ces questions. Cela définit le cadre de la seconde partie de ce manuscrit de thèse.

Le **chapitre IV** est consacré à la description du besoin de coordination opérationnelle, qui s'inscrit au niveau CIM du cadre précité. Nous concevons un langage de modélisation spécifique au domaine (DSML) pour capturer les connaissances qui caractérisent le bon fonctionnement de la coordination dans la PAD, dans toute sa diversité. Ayant conscience de la complexité du

problème de coordination opérationnelle de PAD, la conception du DSML s'est faite selon une démarche progressive exploitant un voisinage avec le problème de tournée du voyageur de commerce (TSP), considéré comme noyau de base de la modélisation. C'est l'émancipation de ce noyau dit DSML-TSP qui permet de tendre vers le DSML-HHCRSP. Nous expliquons l'implémentation de ce langage dans l'environnement de méta modélisation ADOxx et exposons nos résultats en termes de capacité d'expression du besoin par un coordinateur de PAD.

Le **chapitre V** décrit deux méthodes de planification HHCRSP :

- (1) Une méthode exacte en Programmation Linéaire en Nombre Mixte (PLNM) résolue par le solveur CPLEX,
- (2) Une méthode approchée en satisfaction de contraintes couplée à une méta-heuristique, apte à traiter des problèmes de grande taille et résolue par OptaPlanner.

La formulation mathématique du problème d'optimisation intègre les connaissances véhiculées par l'expression du besoin du chapitre précédent. Une originalité de la formulation des problèmes étudiés est la prise en compte de plusieurs formes de satisfaction de parties-prenantes de la PAD. Pour analyser la qualité des solutions, nous avons effectué une expérimentation approfondie basée sur 13 cas d'utilisation extraits de cas réels fournis par Berger-Levrault, avec une analyse critique et comparative des solutions générées par les deux méthodes. Là encore, puisque les méthodes se révèlent avoir de réelles qualités, une nouvelle option s'ajoute à la diversité de conception d'un environnement numérique de coordination de la PAD : ces deux méthodes sont candidates à occuper les niveaux PIM et PSM du MDA.

Une preuve de concept en Ingénierie Dirigée par les Modèles (IDM), pour en valoriser la pratique sur le sujet de la coordination en PAD, est présentée au **chapitre VI**. Nous montrerons la place laissée à l'architecture dirigée par les modèles pour faire face à cette somme de diversité du besoin et de sa satisfaction. C'est dans une chaîne des transformations de modèle appelée **la fabrique de la coordination**, allant du niveau CIM (DSML-TSP) au niveau PSM (codes OPL du solveur CPLEX), que l'IDM se déploie. Dans un but avant tout pédagogique, nous avons testé cette preuve de concept autour du problème du voyageur de commerce (TSP). La mise en application sur ADOxx et le prototype qui en a résulté *BL-Autom* sont présentés en fin de chapitre.

Le **chapitre VII** conclut ce mémoire de thèse avec une synthèse des contributions et une discussion sur les pistes de recherche pour la suite de nos travaux.

CHAPITRE II. FONDEMENTS SCIENTIFIQUES ET ETAT DE L'ART

CHAPITRE II. FONDEMENTS SCIENTIFIQUES ET ETAT DE L'ART	19
1. INTRODUCTION	20
2. ETAT DES CONNAISSANCES SUR LA COORDINATION DE LA PRISE EN CHARGE A DOMICILE	20
2.1. Analyse des caractéristiques de l'offre et de la demande de services en PAD	20
2.1.1. Caractériser l'intervenant dans toute sa variété	21
2.1.2. Caractériser l'offre de la structure de PAD	23
2.1.3. Caractériser la demande dans l'intérêt du bénéficiaire	24
2.2. Eléments entrant dans la prise de décision pour apparier offre et demande en PAD	26
2.2.1. Exigence de métiers	26
2.2.1.1. Faisabilité et qualification d'une intervention.....	27
2.2.1.2. Règles organisationnelles habituelles	28
2.2.1.3. Exigences temporelles.....	29
2.2.1.4. Exigences géographiques.....	32
2.2.1.5. Synthèse	33
2.2.2. Recherche de performance de la PAD par la structure en charge de la coordination.....	34
2.2.2.1. Contrôle des temps.....	34
2.2.2.2. Contrôle des coûts	35
2.2.2.3. Equilibrage de pénibilité des tournées	35
2.2.2.4. Synthèse	35
2.3. Bilan des travaux antérieurs sur l'aide à la décision pour la coordination de PAD	37
2.3.1. Le problème de la planification des interventions dans la prise en charge à domicile (HHCRSP)	37
2.3.1.1. Problèmes adjacents au HHCRSP.....	37
2.3.2. Incertitudes liées au HHCRSP opérationnel.....	38
2.3.3. Méthodes de résolution.....	39
2.3.3.1. Méthodes exactes	39
2.3.3.2. Méthodes approchées	41
2.3.3.3. Approches hybrides.....	44
2.3.3.4. Approches liées aux incertitudes opérationnelles	44
2.3.4. Synthèse.....	46
3. CONCLUSIONS	47

1. Introduction

Dans ce travail de recherche, un point focal est une amélioration de la coordination de prises en charge de patients à domicile (PAD). Ce sujet est abordé dans le cadre large d'une fonction de pilotage que nous déclinons sous forme de gestion des plannings des interventions et des intervenants, de leur création à leur maintien. Une partie de notre travail consiste en la définition des données qui permettent de caractériser au maximum toutes les configurations raisonnables que pourraient rencontrer un coordinateur de PAD. Ce n'est qu'à travers la caractérisation des connaissances fondamentales de la coordination opérationnelle dont il a la charge que nous serons en mesure d'identifier des pistes de progrès pour la planification, pour assister une génération de plannings en première intention.

Ce chapitre est ainsi dédié à un état des connaissances sur la coordination de la prise en charge à domicile sous la forme d'une étude bibliographique de ce sujet (la section 2 de ce chapitre). Partant de (1) nos acquis du chapitre précédent sur la théorie de la coordination (Malone and Crowston, 1994), (2) les éléments de théorie sur la coordination d'une prise en charge à domicile de (Lamine, 2019), et (3) notre souhait d'adopter une démarche systémique (système de décision en production de services), notre analyse prend le parti d'aborder le contenu par une partition entre offre et demande de services.

Cela est réalisé en considérant trois aspects complémentaires : (1) les caractéristiques de l'offre et de la demande, (2) les éléments entrant dans la prise de décision pour appairer offre et demande en PAD, (3) Le bilan des travaux antérieurs sur l'aide à la décision pour la coordination de PAD. Ces trois aspects sont abordés dans cet ordre dans la suite de ce chapitre. Enfin, dans la dernière section, nous concluons sur les enseignements acquis par ce travail sur les connaissances acquises.

2. Etat des connaissances sur la coordination de la prise en charge à domicile

La coordination en santé est une activité de pilotage dont il faut être conscient qu'elle s'applique à des systèmes sociotechniques complexes. Afin d'aller plus avant dans la description de la coordination dans les structures de prise en charge à domicile, nous avons employé une démarche classique d'analyse de systèmes. Nous avons sélectionné trois dimensions caractéristiques d'une telle démarche pour d'une part, classer les connaissances issues de la littérature et expliciter les facteurs causant cette complexité et, d'autre part, identifier les points pouvant faire l'objet d'études plus poussées car la connaissance y reste incomplète. Détaillons ces trois dimensions :

- (1) Caractéristiques de l'« offre » et de la « demande ».
- (2) Eléments entrant dans la prise de décision pour appairer offre et demande en PAD.
- (3) Bilan des travaux antérieurs sur l'aide à la décision pour la coordination de PAD.

2.1. Analyse des caractéristiques de l'offre et de la demande de services en PAD

Nous disposons d'un recul historique et de traces expérimentales (cf. la section 5.1 du chapitre I) permettant de faire un retour d'expériences de la réalité vécue de très nombreuses prises en

charges à domiciles. Les besoins des patients sont spécialisés, de sorte que les prestataires de soins de santé sont confrontés à un grand nombre de profils différents.

D'après (En-nahli et al., 2015) la modélisation des relations « offre-demande » est compliquée par la conjugaison des facteurs tels que les qualifications des intervenants, les exigences organisationnelles et l'équité de la charge de travail chez les intervenants. De même, l'absentéisme imprévisible et les temps de repos journaliers pour les intervenants, ainsi que les restrictions sur la charge de travail des aidants naturels, la variation de la demande des patients, sont autant de facteurs qui peuvent influencer la prise de décision. A cela s'ajoute la compatibilité ou l'affinité des compétences entre les intervenants et les patients. A ce stade, il est opportun de rappeler que la prestation de soins est perçue, dans nos travaux, comme une prestation de services à la personne assurée par des organisations compétentes et qualifiées.

Ainsi, une forme de répartition entre offre et demande est un point d'entrée de notre analyse selon deux angles : (1) le rapport à la théorie de la coordination (Malone and Crowston, 1994), de quelles dépendances relèvent l'offre et la demande ? (2) Le rapport à la coordination centralisée, et quelle valeur ajoutée de cette partition offre-demande dans l'aide à la décision ? Ces considérations donnent une première impulsion pour un recours à la modélisation de systèmes. Et cela selon deux volets : un pour les intervenants inscrits dans une structure de PAD et qui représentent l'espace de l'offre ; l'autre qualifiant les attentes des patients auxquels l'espace de l'offre devrait répondre, c'est l'espace de la demande.

Si on se confine aux seuls acteurs de la PAD, nous avons donc 3 entités qui entrent nécessairement en jeu dans un modèle de coordination : (1) le praticien (**intervenant**), (2) le bénéficiaire de prescription de soins (**bénéficiaire**), et (3) l'organisme (**structure de PAD**) permettant d'inscrire tous les bénéficiaires à prendre en charge par les intervenants qui sont autorisés à intervenir sur un territoire géographique déterminé par l'Agence Régionale de Santé (Laffont, 2010). Chacune de ces trois entités porte son lot de connaissances, c'est une première structuration de notre problème.

Chemin faisant, c'est dans l'interaction entre ces connaissances que se construira la dynamique de fonctionnement du système cible de coordination centralisée, et qu'il faut inscrire son pilotage. Sachant qu'on ne peut ignorer les éléments de contexte venant de l'environnement dudit système. La littérature scientifique mobilise évidemment ces connaissances de manière intensive et elle propose souvent d'en bâtir des classifications avec un point de vue préférentiel, une vue **intervenant** associée à une vue **structure de PAD** côté offre, puis une vue **bénéficiaire** côté demande. Notre proposition est donc légèrement différente.

2.1.1. Caractériser l'intervenant dans toute sa variété

En France, l'infirmier¹ et l'aide-soignant sont les deux métiers principaux qui prodiguent des prestations paramédicales (infirmières et/ou aides-soignantes). Généralement, les soins d'hygiène, désignés généralement sous l'appellation anglophone « soins de nursing », l'aide à la toilette, l'aide à l'habillage et la surveillance des constantes par exemple, peuvent être dispensés par des aides-soignants sous la responsabilité d'un infirmier, ou par des infirmiers, alors que les

¹ Un diplôme d'Etat est exigé pour devenir un IDE : « Infirmier Diplômé d'Etat »

actes médico-techniques nécessitant une prescription médicale, l'injection par exemple, sont uniquement réalisés par les infirmiers. En Allemagne comme en France, on distingue principalement deux types de métiers pour les soins à domicile, à savoir l'infirmier professionnel et l'infirmier auxiliaire (Yuan and Fügenschuh, 2015). Ces deux types d'infirmiers se différencient par leur qualification et par leur coût salarial. Les infirmiers professionnels sont capables d'effectuer tous les types de soins allant de soins de nursing aux soins médico-techniques (injections, les bandages, par exemple). L'infirmier auxiliaire reçoit généralement une formation pratique de courte durée. Il peut effectuer des soins simples comme le nettoyage ou le changement de pansements. Toutefois, ils ne sont pas autorisés à effectuer des actes tels que des prescriptions ou des injections (Yuan and Fügenschuh, 2015).

La gamme de qualifications est très large et selon le contexte, comprend, par exemple des aptitudes (aussi dit « compétences » des intervenants) en communication comme plusieurs langues parlées, ou des aptitudes en délégation de logistique comme une permission de faire des courses ou une autorisation pour administrer des médicaments. Dans (Shao et al., 2012), une liste de professions envisagées vis-à-vis des interventions de rééducation sont présentées : les physiothérapeutes, les assistants en physiothérapeute, les ergothérapeutes, les assistants certifiés en ergothérapie, les pathologistes / orthophonistes. Selon (Gayraud et al., 2013), la durée d'intervention serait liée au niveau de qualification.

La nature du contrat de travail d'un intervenant est généralement reliée à la capacité de travail et aux horaires de la prise en charge des interventions. Selon (Cheng and Rich, 1998) (Wirnitzer et al., 2016), deux types de contrats de travail sont habituellement proposés aux intervenants conventionnés avec une structure PAD : « salarié à plein temps » et « non salarié à temps partiel ». Les salariés sont payés quotidiennement pour un poste à temps plein, indépendamment du fait qu'ils soient programmés de manière intensive ou non. Les non-salariés à temps partiel sont souvent payés par heure en fonction des interventions réalisées. Ces contrats génèrent une grande variété d'amplitudes des horaires de travail, qui peuvent être étendues ou réduites en fonction des besoins de la structure et de ses bénéficiaires, ainsi qu'en accord avec les intervenants (Méndez-Fernández et al., 2020). Les horaires de travail des intervenants doivent être conformes non seulement aux contrats de travail, mais aussi au Code du travail et aux textes réglementaires relatifs à la PAD. Toute transgression des règlements et des lois pourrait être un problème sérieux pour une structure de PAD (Lin et al., 2018).

Le confort de travail des intervenants est en lien avec leur niveau de rémunération et avec le respect des préférences personnelles. Le salaire des infirmières auxiliaires (aides-soignantes) est inférieur à celui des infirmières professionnelles, il est estimé à 70 % de ce que reçoivent les infirmières professionnelles (Yuan and Fügenschuh, 2015). Dans (Shao et al., 2012), le salaire du thérapeute est calculé par heure. Concernant le respect des préférences personnelles, la liste de préférences spécifiées par les intervenants doit être respectée afin d'assurer la satisfaction des intervenants lors la réalisation de soins. Par exemple, certains intervenants peuvent présenter des incompatibilités avec des bénéficiaires possédant des animaux de compagnie, notamment à cause d'allergies respiratoires.

En plus des interventions, les intervenants sont également rémunérés pour le temps administratif nécessaire à la préparation et à la documentation des interventions effectuées (Shao et al., 2012), et aux réunions hebdomadaires (Hiermann et al., 2015). Dans (Shao et al., 2012),

les missions administratives correspondent à 25 % du temps d'intervention pour les thérapeutes et 20 % pour les thérapeutes assistants. Selon (Thompson and Goodale, 2006), ce temps administratif est un paramètre important qui est pris en considération dans le calcul du taux de rendement synthétique d'un intervenant et qui est généralement inscrit dans le contrat. Ce taux est alors défini comme suit :

$$\frac{\text{Temps (intervention)}}{\text{Temps(intervention)+Temps(administratif)}}$$

Les taux de rémunération sont différents selon l'intervention, le trajet entre les bénéficiaires et les tâches administratives à faire (Shao et al., 2012).

Le moyen de transport est une donnée nécessaire pour soutenir les intervenants qui se rendent aux domiciles des bénéficiaires en vue d'effectuer l'intervention planifiée (Hiermann et al., 2015). Les intervenants doivent parfois organiser leur tournée en fonction des temps des trajets entre les bénéficiaires, qui dépendent du mode de transport choisi comme la marche à pied, la voiture (le permis B est exigé en France) ou les transports publics (Rest and Hirsch, 2016), la voiture électrique (Erdem and Koç, 2019), le vélo électrique (Szander et al., 2019). De même, ces temps des trajets peuvent varier en fonction de la nature du transport adopté, à savoir s'il s'agit d'un transport collectif ou un transport individuel. Normalement, l'utilisation d'un transport individuel (la voiture privée) est plus rapide que le transport collectif, mais plus coûteux (Goodarzian et al., 2020). Dans (Quintanilla et al., 2020), les plans de taxi sont plus flexibles, à savoir que les taxis ne sont pas pré-affectés à certains intervenants, et ceci leur permet de commencer les tournées à tout moment et en tout lieu.

En résumé, les intervenants sont au cœur de l'offre et sont souvent inscrits dans un collectif au sein des structures de PAD. Ils y ont pour mission première de répondre avec qualité aux besoins des bénéficiaires. Ces intervenants sont qualifiés pour exercer les interventions et disposent d'un contrat avec la structure pour garantir leur revenu, leurs heures de travail et leur confort au travail, même s'il s'agit d'une disponibilité avec un temps partiel. Ils ont également une potentielle préférence en ce qui concerne les bénéficiaires à soigner. Enfin, les intervenants doivent utiliser différents moyens de transport afin de se rendre sur le lieu de résidence des bénéficiaires pour répondre à leurs demandes.

2.1.2. Caractériser l'offre de la structure de PAD

L'essor rapide de structures de prise en charge à domicile permet de répondre aux besoins de bénéficiaires à un niveau collectif. L'une des raisons de cette croissance rapide est la démonstration de l'efficacité et de la performance des services de prise en charge à domicile face à la diversité des besoins de bénéficiaires (Bahadori-Chinibelagh et al., 2019). Les structures offrent une large gamme de services aux personnes afin de soulager les familles, les hôpitaux et les services de soins. Pour atteindre ce but, ces structures s'attachent les services d'acteurs qualifiés : infirmières, thérapeutes, travailleurs sociaux, médecins (Mankowska et al., 2014). Elles font aussi appel à des prestataires pour ce qui relève des services à la personne : portage des repas, blanchisserie, etc. Dans (Goodarzian et al., 2020), les traitements d'anticoagulation sanguine, les soins chroniques et les soins palliatifs sont considérés comme des services spéciaux fournis aux bénéficiaires. Et en plus de soins infirmiers et de physiothérapie, le soin d'hygiène tel que la toilette, l'habillage (Fathollahi-Fard et al., 2020) et la préparation des repas (Bazirha et

al., 2019) sont également disponibles dans la liste des services proposés par la structure. De même, l'étude de (Shao et al., 2012) présente un service de rééducation au sein d'une structure. Le service peut concerner l'une des trois catégories : physique, professionnelle et parole.

La réalité du périmètre territorial couvert par la structure lui procure un attribut de localisation géographique de l'offre.

Il est à noter que la culture, la réglementation et les pratiques en vigueur sur un territoire peuvent influencer les mécanismes de PAD. (Duque et al., 2015) présentent une organisation décentralisée pour la planification des intervenants. La zone couverte par cette organisation est divisée en 44 régions indépendantes. Chaque région est gérée par un manager régional qui est responsable de la planification des activités des soignants. Par conséquent, une stratégie a été appliquée de manière indépendante à chaque région.

A partir d'un exemple de taille importante, (Du et al., 2017) mettent en évidence les critères de coordination et le poids des conventions de soins en PAD en Chine. L'organisation repose principalement sur des cabinets de médecins familiaux installés dans les structures (une variante de notre médecin traitant). Un cabinet de médecin de famille regroupe deux ou trois médecins généralistes qui fournissent des soins médicaux sur rendez-vous au domicile des bénéficiaires. En moyenne, un médecin peut visiter 10 à 15 familles qui ont signé des contrats de prise en charge à domicile avec son cabinet. Selon les « *Guiding Opinions on Promoting the Contractual Healthcare Services from Family Doctors* » (Shan-shan et al., 2018) émis par le Conseil d'Etat de Chine, les structures de PAD sont réglementées en fonction de ces contrats. Les résidents et les familles ont le libre choix d'une équipe de médecins et signent avec eux un engagement pour valider la prise en charge à domicile. Dans cet engagement, le contenu, la nature et les délais des services contractuels des structures de PAD, ainsi que les responsabilités, les droits, les obligations des deux parties, et d'autres questions pertinentes, ont été clairement définis. En principe, l'accord contractuel est valable pour un délai d'un an. Lorsque les contrats originaux expirent, les résidents et les familles peuvent choisir d'étendre les contrats ou de changer de fournisseurs de PAD. Après l'achèvement des soins médicaux pour tous les bénéficiaires, les intervenants peuvent retourner à leur cabinet. Le poids des conventions de soins que signent les acteurs de la santé pèsent sur le jugement de la qualité de prise en charge. Même si elle est délicate à appréhender dans ce secteur où la dimension humaine est au tout premier plan, la performance de l'organisation ne peut être ignorée et doit être évaluée avec un esprit de progrès que nul ne saurait contester si elle améliore nos conditions de vie, incluant la fin de vie.

2.1.3. Caractériser la demande dans l'intérêt du bénéficiaire

Chaque bénéficiaire est un cas d'espèce. Sa demande est personnalisée. Les bénéficiaires suivis par une structure présentent donc différentes caractéristiques initiales en termes de pathologie, d'âge, de contexte domestique et d'autres conditions cliniques, psychologiques et sociales. Dans (Fathollahi-Fard et al., 2020), l'ensemble des données relatives aux bénéficiaires enregistrés doit être collecté par les structures de PAD, à savoir le nom, le sexe, l'âge, l'adresse, les types de maladies, l'historique de leur maladie, les symptômes, la durée prévue de la prise en charge et d'autres données nécessaires aux bénéficiaires. Les bénéficiaires reçoivent des soins différents et, même s'ils sont dans un même état pathologique, ils ont des caractéristiques personnelles qui font que la prise en charge de chaque cas est potentiellement spécifique (Lanzarone et al., 2010)

(Zefouni, 2012) (Rejeb, 2013). L'état de santé de bénéficiaires peut être mesuré à l'aide d'une échelle de mesure adaptée à sa situation thérapeutique. Par exemple, en France, l'indicateur GIR (Groupe Iso-Ressource) a été conçu pour mesurer le niveau de perte d'autonomie des personnes âgées (Petite, 2006). Il est calculé à partir de la grille AGGIR (Autonomie-Gérontologie-Groupe Iso Ressources) et prend en compte certains critères comme la cohérence (converser et se comporter de manière responsable) pour se déplacer en autonomie ou pour utiliser des moyens de communication (Gayraud et al., 2013). Cette mesure est destinée à identifier la nature des besoins d'une personne âgée dépendante, et ainsi à adapter les interventions en fonction de sa situation (Petite, 2006).

Certains bénéficiaires possèdent des habitudes et une liste d'intervenants préférés (En-nahli et al., 2015) (Braekers et al., 2016a) du fait de l'établissement de relations de confiance entre ces intervenants et eux (ou leurs aidants) lors des précédentes interventions. Dans certain cas, il faut considérer que certains intervenants pourraient être incompatibles à une affectation potentielle pour laquelle ils sont compétents, mais qu'ils ne veulent pas assurer en raison d'une habitude spécifiée par le bénéficiaire, comme par exemple les bénéficiaires qui sont fumeurs (Trautsamwieser and Hirsch, 2014).

Un bénéficiaire (le patient ou la personne en situation de handicap, par exemple) doit disposer de son propre PPCS (Plan Personnalisé de Coordination en Santé)¹. Ce PPCS a été imaginé et disséminé à l'initiative de la HAS. Il est le consensus présidant à la PAD d'une personne donnée, fixe les interventions nécessaires des professionnels de santé, des personnels médico-sociaux et sociaux, en réponse aux besoins de soins des usagers du système de santé (ANAP, 2020). Au niveau opérationnel, les bénéficiaires nécessitent parfois un certain nombre d'interventions à domicile à une fréquence journalière ou hebdomadaire. Par conséquent, il s'agit en quelque sorte d'une consigne à respecter qui spécifie les conditions à remplir par les intervenants (à un moment approprié en un lieu approprié) (Yuan and Fügenschuh, 2015). Ces interventions incluent par exemple : la toilette, le nettoyage du lieu de vie, le changement de pansements, la prescription de médicaments ou des injections (Carello and Lanzarone, 2014). La durée de chaque intervention est fixée et donnée à l'avance, mais l'heure de début peut être modifiée dans une fenêtre temporelle spécifiée par le bénéficiaire (Yuan and Fügenschuh, 2015). Ces fenêtres de temps désignent les disponibilités des bénéficiaires, qui limitent les heures d'arrivée au plus tôt et au plus tard des intervenants requis au lieu de vie de bénéficiaire (Du et al., 2017). Les bénéficiaires souhaitent souvent la ponctualité et une bonne réactivité des intervenants quant à leur demande. Pour exécuter certaines interventions particulières, il est nécessaire que deux intervenants soient présents simultanément ou que des équipements spécifiques soient apportés au domicile du bénéficiaire (Carello and Lanzarone, 2014).

La Figure 7 propose une récapitulation des caractéristiques contribuant à une connaissance approfondie et structurée des PAD dans cette partition offre-demande. Ces paramètres abondent dans le sens d'une réelle complexité dont il faut tenir compte si l'on souhaite rester pragmatique et faire face à une diversité de systèmes.

¹ Plan Personnalisé de Coordination en Santé (PPCS) : https://www.has-sante.fr/jcms/p_3083915/fr/plan-personnalise-de-coordination-en-sante

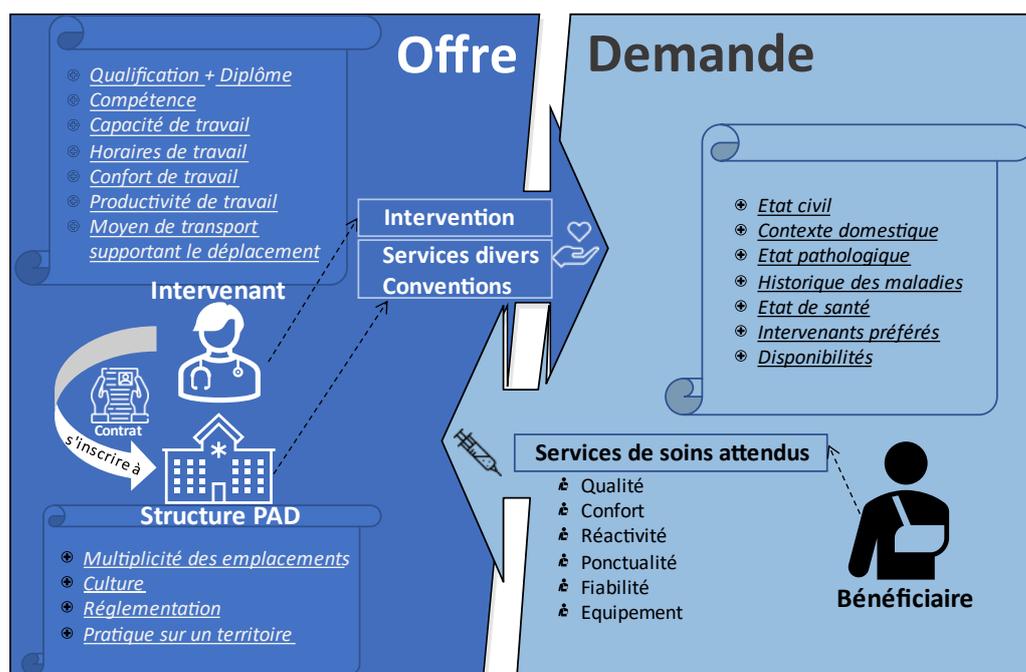


Figure 7 : Offre et demande du système de PAD : une structuration de connaissances

2.2. Eléments entrant dans la prise de décision pour apparier offre et demande en PAD

Sous l'hypothèse que nous saurions caractériser une PAD, quelle que soit celle-ci, nous devons maintenant aborder ce qu'est la nécessité de coordonner les intervenants. Si le bon fonctionnement de notre système repose sur la bonne coordination des activités (l'affectation des intervenants aux interventions, la planification de tournées), dans la vraie vie, il est difficile de l'assurer en raison du nombre de règles métiers existantes et de la diversité des exigences exprimées par les différentes structures PAD (cf. la section 3 du chapitre I). Ce point est traité à la section 2.2.1 de ce chapitre.

La coordination est un acte de pilotage et, en cela, elle obéit à une logique de performance. Les critères de performance qui vont guider les décisions prises par le coordinateur dans un champ des possibles forment une seconde vague de connaissances que nous allons également étudier dans ce paragraphe (la section 2.2.2 de ce chapitre).

Dans la même logique qu'au paragraphe précédent, nous abordons ces connaissances par des sources qui sont essentiellement des articles de l'abondante littérature scientifique sur le sujet.

2.2.1. Exigence de métiers

Les exigences liées à l'affectation des ressources aux interventions à domicile et à la planification de tournées des intervenants dépendent de conditions de validité des opérations. Certaines de ces conditions doivent être nécessairement satisfaites pour aboutir à une coordination valide, notamment le respect de la réglementation en vigueur, alors que d'autres sont plutôt relatives à la qualité de l'organisation et peuvent donc ne pas être entièrement satisfaites. Ainsi, ces exigences sont classées en exigences dures et exigences souples. Le qualificatif « dure » signifie une obligation à satisfaire, alors que « souple » est attaché à une préférence et ne conduit pas à une invalidation en cas de non-respect (Hiermann et al., 2015).

2.2.1.1. Faisabilité et qualification d'une intervention

- Faisabilité d'une intervention

La relation entre offre et demande est souvent définie au premier abord par une relation binaire (Cheng and Rich, 1998). Une paire « intervenant-bénéficiaire » est ainsi créée par une simple mise relation ayant pour but de traduire le degré de faisabilité (c'est-à-dire, « faisable » ou « non faisable »). L'absence de mise en relation induit une impossibilité d'affectation de l'intervenant aux interventions de ce bénéficiaire pour lesquelles il serait qualifié.

Dans (Mankowska et al., 2014), la faisabilité d'une intervention est exprimée par la prise en compte globale des besoins individuels des bénéficiaires et des qualifications individuelles des intervenants. La granularité dans la définition des besoins peut se situer à l'échelle de l'individu, du groupe d'individus, de l'établissement, puis du territoire de soins. Ce sont des mécaniques de groupage/dégroupage de ressources qui sont visées.

- Qualification d'une intervention

(Castillo-Salazar et al., 2012) mettent en évidence ces rapports d'adéquation entre la demande des bénéficiaires et la qualification des intervenants. Ils classent les demandes des bénéficiaires en plusieurs types selon les compétences requises. D'une manière équivalente, les intervenants sont divisés sur la base de leurs compétences propres et organisés dans une structure hiérarchique. Un intervenant hautement qualifié peut servir des bénéficiaires qui émettent des demandes particulières moins qualifiées. Au contraire, un intervenant peu qualifié ne peut pas servir les bénéficiaires dont les demandes nécessitent des compétences plus élevées. Ainsi, il devient nécessaire de bien faire correspondre les compétences des intervenants avec les compétences requises, car certaines compétences sont très coûteuses et l'obtention de leur certificat d'infirmier est complexe et longue (Khodabandeh et al., 2020).

Si des intervenants hautement qualifiés sont autorisés à fournir des services à un niveau de qualification exigé peu élevé, un coût de rétrogradation va en découler (Khodabandeh et al., 2020). La rétrogradation (et en corollaire l'insatisfaction des intervenants) est causée par une sous-utilisation d'une partie des compétences des intervenants par le coordinateur. Aussi, cette affectation d'un intervenant hautement qualifié doit être limitée aux bénéficiaires qui en ont besoin (Castillo-Salazar et al., 2012). Le meilleur résultat est de réussir à affecter des intervenants en mobilisant toutes leurs compétences. Mais on comprend que c'est un exercice subtil.

Pour exprimer cette dimension dans les relations d'affectation, (Fikar and Hirsch, 2015), proposent que chaque intervenant soit associé à un niveau de qualification et chaque bénéficiaire nécessitant une intervention avec une certaine exigence de niveau de qualification. Ainsi, les interventions ne peuvent être effectuées que par un intervenant ayant au moins le même niveau de qualification, avec une limitation de la surqualification d'intervenants affectés. Dans (Lieder et al., 2015), la qualification est hiérarchisée en 3 niveaux :

- (1) Le niveau 1 (le plus bas) correspond à la distribution de nourritures et de boissons, le nettoyage ;
- (2) Le niveau 2 permet le lever et le coucher, la prise des repas, la toilette, la préparation des lits et le ménage ;

- (3) Le niveau 3 (le plus haut) couvre la distribution de médicaments et la surveillance médicale.

2.2.1.2. Règles organisationnelles habituelles

- Notions attachées à la ressource humaine

Comme longuement discuté à la section 2.1.3 de ce chapitre, certains bénéficiaires possèdent une liste d'intervenants préférés (En-nahli et al., 2015), ou un intervenant particulier préféré (Braekers et al., 2016a). Ces contraintes préférentielles sont conçues pour satisfaire ces relations « bénéficiaires-intervenants » spécifiées par les acteurs et créent un espace de confort pour les acteurs. Dans (Méndez-Fernández et al., 2020), les auteurs soulignent l'importance de considérer l'opinion des bénéficiaires lors de l'affectation des intervenants. Ceci est dû au fait que ces bénéficiaires sont satisfaits de ces intervenants qui les prennent en charge. Toutefois, en cas de conflit, il est souhaitable de changer l'intervenant affecté à un bénéficiaire. Les auteurs utilisent une liste de niveaux d'affinité (de 0 à 5 où 5 signifie la satisfaction d'affectation maximum des bénéficiaires) qui précisent dans quelle mesure il est approprié d'affecter un intervenant à chaque bénéficiaire. Ce concept d'« affinité » est utilisé par un client de Berger-Levrault. Le niveau d'affinité des bénéficiaires est évalué lors des appels de suivi téléphoniques entre la structure et le bénéficiaire. Certains bénéficiaires sont plus exigeants que d'autres et demandent à avoir expressément les mêmes intervenants pour effectuer ses interventions demandées.

Il existe dans certains cas des incompatibilités entre les bénéficiaires et les intervenants dues à des allergies (Chaieb et al., 2020) ou à la nature de l'intervention qui devrait normalement être effectuée, selon le profil du bénéficiaire, par deux intervenants, qui doivent avoir dans ce cas les mêmes qualifications. À titre d'exemple, nous pouvons citer les interventions d'aide au lever ou au coucher ou celles des soins d'hygiène pour les bénéficiaires à forte corpulence (Mascolo et al., 2018).

- Règles de priorité

Certains bénéficiaires disposent d'une priorité majeure pour être soigné. La priorité de ces bénéficiaires signifie que l'intervenant affecté doit se rendre à l'endroit où l'intervention est requise avec une priorité plus élevée pour l'effectuer. Cette priorité des interventions a un impact sur l'affectation des bénéficiaires. Les exemples de telles caractéristiques peuvent provenir de spécificités liées aux besoins de soins des bénéficiaires. Par exemple, la visite d'un bénéficiaire nécessitant une analyse sanguine serait tôt le matin, bien qu'il puisse être optimal de lui rendre visite à la fin de la journée si seules les coordonnées géographiques sont prises en compte. Cette priorité risque d'obliger les intervenants à effectuer des trajets plus longs, ce qui augmente leur charge de travail (Yalçındağ et al., 2016). Dans (Du et al., 2017), les niveaux de priorité sont identifiés et déterminés par la gravité de l'état des bénéficiaires. Souvent, les bénéficiaires ayant un état de santé fragile (mesuré par l'indicateur GIR, par exemple) bénéficient de cette notion de priorité de soins.

- Règles de régularité

La régularité de l'intervenant signifie que le même intervenant devrait être affecté au même bénéficiaire pour le même type d'intervention. Le premier intervenant affecté est considéré comme l'intervenant de référence. La relation d'affectation est conservée pendant une longue

période. La régularité de l'intervenant préserve la qualité de service perçue par les bénéficiaires, puisque les bénéficiaires reçoivent des soins du même intervenant et n'ont pas à développer continuellement de nouvelles relations avec de nouveaux intervenants. La perte potentielle d'informations entre les intervenants est ainsi évitée. Ces deux aspects sont extrêmement importants dans le cadre de la prise en charge à domicile, où le bénéficiaire est assisté pendant une période assez longue. En effet, ils contribuent à la continuité des soins qui est considérée comme une des préoccupations majeures des organisations de PAD et dans la logique de parcours d'usagers. Elle a pour finalité d'éviter toute rupture dans le suivi du malade (Lamine et al., 2010) et désigne la façon selon laquelle un bénéficiaire perçoit la série de services qu'il reçoit, comme une succession d'événements connexes, cohérents et compatibles avec ses besoins et sa situation personnelle.

Parmi les travaux qui se sont intéressés à cette règle de régularité, nous pouvons citer ceux de (Martinez et al., 2018) qui assurent la régularité des soins en minimisant le nombre d'intervenants différents affectés à un même bénéficiaire, ainsi que ceux de (Méndez-Fernández et al., 2020) qui changent l'intervenant affecté au bénéficiaire qu'en cas de non-respect spécifié de la règle de régularité. De même, dans (Carello and Lanzarone, 2014), les bénéficiaires doivent être classés en fonction de leurs besoins et la régularité des intervenants doit être garantie pour un sous-ensemble d'entre eux. (Lanzarone and Matta, 2014) traite le problème de l'affectation des bénéficiaires admis à leur intervenant de référence en utilisant une approche analytique, qui tient compte de la probabilité que la charge de travail d'un intervenant soit supérieure à sa capacité maximale du travail journalier.

La contrainte de régularité peut aussi considérer une régularité temporelle. Les bénéficiaires apprécient d'avoir les mêmes créneaux horaires pour les soins. Cette contrainte est prise en compte pour une durée de plusieurs jours (Misir et al., 2010) ou de plusieurs semaines (Ducque et al., 2015) (Grenouilleau et al., 2020), selon les consignes affichées dans le plan de soins.

2.2.1.3. Exigences temporelles

Le respect des règles temporelles est considéré comme un critère majeur dans la mesure où notre système de PAD a une dynamique. Le prisme offre-demande éclaire le rapport au temps sous le jour d'une mesure de satisfaction des bénéficiaires et des intervenants. Ces placements des interventions sur une échelle de temps se concrétisent par des synchronisations d'agendas individuels des deux catégories d'acteur. Le décideur doit donc honorer, autant que possible, les possibilités et les préférences des uns et des autres. Nous constatons le succès d'un service numérique comme Doctolib dans une mécanique de prise de rendez-vous. Ici, le problème est plus difficile. (1) Les interventions du plan de soins se substituent à un seul rendez-vous. (2) Les rendez-vous se font au domicile. (3) Ce sont plusieurs professionnels qui sont sollicités dans un certain ordre.

- Exigences temporelles usuelles

Les interventions demandées doivent être réalisées sans perte de qualité du service, avec la garantie d'une durée d'intervention suffisante, gage de respect de la charge afférente à la prestation de service au bénéficiaire (Yuan and Fügenschuh, 2015). Une fenêtre de temps désigne un intervalle sur l'échelle de temps utilisée comme spécification de dates possibles pour le début d'une intervention. Elle peut prendre la forme d'horaires préférentiels ou de

disponibilités du bénéficiaire à son domicile (Feng and Wang, 2018) (Hiermann et al., 2015). Par exemple, un bénéficiaire demandera une intervention dans une journée donnée et disposera de plusieurs périodes de disponibilité (Bazirha et al., 2019). Dans (Manavizadeh et al., 2020) chaque service est divisé en service vital et service normal. Pour les services vitaux, compte tenu de la sensibilité et de l'importance de l'état de santé de bénéficiaires, ils doivent être assignés à l'intérieur de la fenêtre de temps (Khodabandeh et al., 2020) et aucun écart n'est autorisé.

Pour les services normaux, une tolérance est possible. Ainsi, deux types de fenêtres de temps sont retenus dans la littérature (Bazirha et al., 2019) (Dekhici et al., 2019): la fenêtre de temps dure (ou fixe) ou la fenêtre de temps souple (ou flexible).

Si l'intervenant arrive au domicile d'un bénéficiaire avant l'heure du rendez-vous (avant la fenêtre de temps de début d'une intervention plus tôt), il doit attendre l'heure du rendez-vous pour commencer le service. Cette durée d'attente de l'intervenant correspond au « temps d'inactivité d'intervenant ». Si l'intervenant arrive après l'heure de rendez-vous (après la fenêtre de temps de début d'une intervention au plus tard), le bénéficiaire doit alors attendre le service avec des risques d'angoisses et de plaintes (Braekers et al., 2016a). Cette durée d'attente des bénéficiaires est désignée sous la dénomination du « temps d'attente de bénéficiaire » (cf. la Figure 8). Nous l'avons pris en considération dans nos travaux d'optimisation des tournées des intervenants (Zhang et al., 2019a). Ils seront présentés en détail dans le chapitre V.

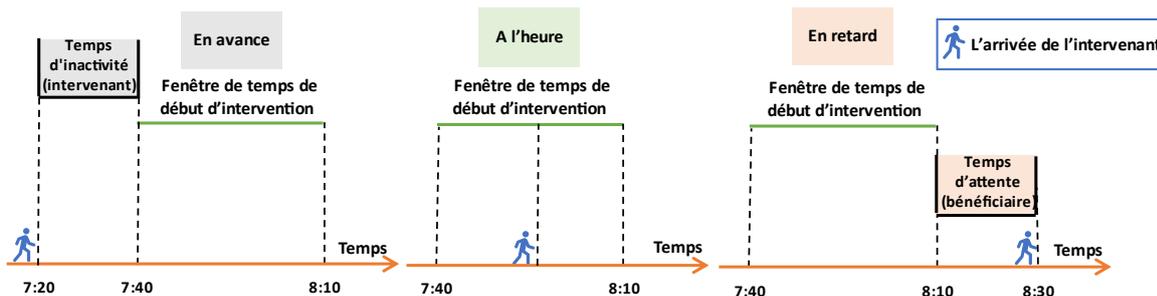


Figure 8 : « Temps d'inactivité d'intervenant » et « temps d'attente de bénéficiaire » en cas de non-respect des fenêtres de temps de début d'intervention

Dans (Bazirha et al., 2019), une avance et un retard maximum des interventions sont définis. Ils correspondent respectivement au « temps d'inactivité d'intervenants » maximum et au « temps d'attente de bénéficiaires » maximum et ils sont fixés par le décideur. Dans (Nasir and Dang, 2018), une pénalité est ajoutée au non-respect des fenêtres de temps demandées. Certains bénéficiaires sont très exigeants, ils préfèrent un rendez-vous ponctuel et précis à une plage horaire (Martinez et al., 2019).

Par ailleurs, dans (Rest and Hirsch, 2016) et (Shao et al., 2012), les auteurs se sont intéressés aux horaires de travail des intervenants que les structures de PAD doivent obligatoirement respecter. Sur un horizon hebdomadaire, par exemple, si un intervenant a un contrat de travail de 25 heures par semaine, alors le décideur doit essayer de lui proposer un planning d'intervention le mobilisant sur 25 heures. Mais une durée de travail maximale journalière peut aussi peser sur la définition de son planning. Un intervalle temporel pourrait être appliqué pour indiquer le temps de début au plus tôt et de fin au plus tard de la tournée de l'intervenant (Di Gaspero and Urli, 2014). En cas de non-respect de la capacité maximale du travail journalier, les intervenants pourront être autorisés à effectuer un certain nombre d'heures supplémentaires au cours de la

période de planification (Wirnitzer et al., 2016). Ce volume d'heures supplémentaires dépend de l'écart entre la capacité et les heures de travail effectif de l'intervenant.

D'autres règles temporelles, telles que celles liées à la disponibilité d'un équipement médical, ou de l'aidant sont également à prendre en considération lors de la génération des plannings d'interventions (Yalçındağ et al., 2016).

- Règles de dépendance entre les activités

Dans la vie réelle, les bénéficiaires peuvent demander plusieurs interventions dans une même journée (Marcon et al., 2017) (Heching and Hooker, 2016). A chaque intervention est alors attribuée au moins une fenêtre de temps possible. Dans ce cas, une séquence indique la chronologie des interventions dans un ordre à respecter et le chevauchement dans le temps entre les interventions est alors impossible (Lin et al., 2016).

Selon (Mankowska et al., 2014), l'interdépendance des interventions peut inclure des situations de prérequis. Si les médicaments doivent être administrés un certain temps avant de fournir un repas, par exemple. Ou encore, si un test sanguin doit être réalisé avant les activités de chimiothérapie. Ces dépendances temporelles sont généralement imposées par le projet thérapeutique (Redjem and Marcon, 2016). Dans la littérature, ces situations sont souvent considérées comme un jeu de « contraintes de précédence » (Parragh and Doerner, 2018) (Manavizadeh et al., 2020).

D'autres interventions peuvent exiger la mobilisation de deux intervenants, ou plus, chez un même bénéficiaire (Decerle et al., 2019) (Decerle et al., 2018), comme lors d'une intervention de soins d'hygiène chez un bénéficiaire lourdement handicapé. Ces contraintes sont souvent désignées comme des « contraintes de synchronisation » (Bredström and Rönnqvist, 2008) (Frifita et al., 2017).

- Exigences d'inter-intervention

Cette règle définit un délai minimal entre deux interventions successives, qui doit être respecté pour obtenir un service de qualité souhaitée.

- (1) Cet écart peut correspondre à la durée minimale imposée entre deux interventions successives assurées par un même intervenant afin de lui permettre d'effectuer le déplacement entre les adresses des deux bénéficiaires (Redjem and Marcon, 2016), (Hiermann et al., 2015).
- (2) Cette règle d'interopération peut exprimer une durée minimale, imposée par le bénéficiaire, entre deux visites d'un même type d'intervenant ou pas, sur une même journée. Cette durée traduit ainsi le besoin du bénéficiaire de se reposer pendant un certain temps après une intervention avant de passer à la suivante. Dans (Haddadene et al., 2016), une durée maximum est imposée pour limiter ce délai.

- Exigences de coupure légale

Au fil de la journée, la fatigue s'accumule, sans prendre de pause, chaque intervention devient plus difficile à accomplir et nécessite de plus en plus d'efforts. La pause du midi, ou tout simplement une coupure lors de la journée, limite la fatigue et augmente l'efficacité du travail. De plus, une pause réduit le risque d'erreur.

Selon le site officiel de l'administration française¹, « la réglementation ne prévoit aucune obligation spécifique concernant la pause méridienne. Toutefois, dès que le temps de travail quotidien atteint 6 heures, le salarié doit bénéficier d'un temps de pause d'au moins 20 minutes consécutives ». Dans (Trautsmawieser and Hirsch, 2014), les intervenants ne sont autorisés à faire qu'une seule pause par tournée. Cette pause peut être prise à n'importe quel moment de la journée à condition que la durée maximale de travail consécutive ne dépasse pas la capacité de travail de l'intervenant pour une journée. Etudiant deux exemples de clients de Berger-Levrault, nous avons rencontré des positions différentes. Pour l'un, l'organisation, qui propose principalement un service de portage de repas à domicile, autorise à ses salariés (correspondants ici aux livreurs) une pause repas de 30 minutes pendant la journée de service. Pour l'autre, en aide à domicile, les intervenants ont le droit d'avoir une pause de midi, de 45 minutes entre 11h et 14h.

Dans (Rest and Hirsch, 2016) et (Shao et al., 2012), les coupures dans une journée peuvent également être divisées. La durée de coupure est liée au coût salarial des intervenants. Dans (Méndez-Fernández et al., 2020), lors du calcul du coût journalier de chaque intervenant, la structure déduit éventuellement les heures associées à la pause la plus longue. Si la durée de cette pause est égale ou supérieure à 2 heures, elle ne sera pas rémunérée.

2.2.1.4. Exigences géographiques

L'organisation des interventions se situe dans un système spatialement distribué car chaque intervention est associée à un lieu précis avec ses coordonnées géographiques (le domicile du bénéficiaire). Ce sont des données clés pour construire des tournées d'intervenants.

- Sectorisation

Ce problème de construction des tournées peut être formulé comme un problème de zonage, c'est-à-dire d'agencement des services en plusieurs unités territoriales distinctes, mais couvrant le territoire marchand de la structure. Cette division peut permettre non seulement de répondre plus rapidement à la demande d'un bénéficiaire, mais aussi d'augmenter la qualité du service fourni du fait de la diminution du temps de déplacement pour se rendre chez les bénéficiaires (Benzarti et al., 2013). Dans (Carello and Lanzarone, 2014), les intervenants répartis en districts répondent à des demandes ainsi sectorisées et le fait de travailler dans des zones de taille restreinte au sein d'une équipe plus petite, renforce la motivation des intervenants tout en établissant une proximité plus grande avec le coordinateur (Benzarti et al., 2013).

- Multimodalité des transports

Un autre aspect de cette distribution spatiale est relatif aux modalités des intervenants. En effet, pour assumer une responsabilité sociétale (bien-être des salariés et développement durable), certaines structures de PAD cherchent à combiner plusieurs modes de transports, en remplacement de la voiture. En fonction des lieux à desservir, il peut s'agir de transport collectif (bus, tram, train et métro) ou individuel (vélo, à pied) (Hiermann et al., 2015). Selon (Rest and Hirsch, 2016), la plupart des soignants de l'ARC (*Austrian Red Cross*) à Vienne (Autriche) utilisent souvent plusieurs modes de transport pour se rendre chez les bénéficiaires. Dans (Hiermann et

¹ Pause déjeuner du salarié : Quelles sont les règles ? <https://www.service-public.fr/particuliers/vosdroits/F34555>

al., 2015), les intervenants réalisent leurs tournées en se déplaçant entre les maisons des bénéficiaires et le bureau central de la structure généralement en voiture ou parfois même en vélo. Pour illustration, la combinaison entre le taxi et la marche à pied est une alternative étudiée par (Quintanilla et al., 2020).

2.2.1.5. Synthèse

Le Tableau 2 réunit les connaissances issues de ces lectures scientifiques et dont nous avons fait état dans ce paragraphe. Nous avons ainsi une première vue de la complexité à laquelle un éditeur comme Berger-Levrault doit faire face pour supporter numériquement une fonction de coordination de PAD.

Nous constatons qu'il n'y pas de référence répondant exhaustivement à ces sept colonnes, amenant à un ensemble de 19 exigences

Tableau 2 : Synthèse d'exigences de PAD

Référence	Type		Point de vue		Echelle		Faisabilité et qualification	Règles organisationnelles habituelles			Exigences temporelles						Exigences géo.		
	Affectation	Planification de tournées	Bénéficiaire	Intervenant	Jour	Jour et/ou semaine		Ressources humaines	Règles de priorité	Règles de régularité	Temporelles - fenêtre de temps dure	Temporelles - fenêtre de temps souple	Temporelles - heure de travail maximale	Dépendances entre les activités - précédence	Dépendances entre les activités - synchronisation	Exigences d' inter-intervention	Exigences de coupure légale	Sectorisation	Multimodal
(Shao et al., 2012)	X	X	X	X	X	X	X				X		X				X		
(Trautswieser and Hirsch, 2014)	X		X	X		X	X				X		X				X		
(Carello and Lanzarone, 2014)	X	X	X	X		X	X				X		X				X		
(Mankowska et al., 2014)	X	X	X		X		X				X		X	X	X				
(B. Yuan et al., 2015)	X	X	X		X		X				X								
(En-nahli et al., 2015)	X	X	X	X	X		X	X			X						X		
(Z. Yuan and Fügenschuh, 2015)		X	X	X		X					X		X						
(Hiermann et al., 2015)	X	X	X	X	X		X				X	X							X
(Duque et al., 2015)	X		X	X		X	X				X		X						
(Rest & Hirsch, 2016)	X	X	X	X	X		X				X		X				X		X
(Redjem and Marcon, 2016)		X	X		X						X		X						
(Yalcındağ et al., 2016)	X	X		X	X		X					X							
(Du et al., 2017)	X	X	X		X		X	X			X								
(Nasir & Dang, 2018)	X	X	X	X	X		X				X		X				X		
(Lin et al., 2018)	X	X	X	X		X	X				X		X						
(Martinez et al., 2019)	X	X	X	X		X	X		X	X		X							
(Bazirha et al., 2019)	X	X	X		X		X				X								
(Çolak et al., 2019)		X		X	X							X							
(Chabane et al., 2019)	X	X	X	X		X	X	X				X	X	X	X	X	X		

(Fathollahi-Fard et al., 2020)		X	X		X					X							
(Goodarzian et al., 2020)	X	X	X	X		X	X			X		X					
(Quintanilla et al., 2020)		X		X	X		X						X				X
(Méndez-Fernández et al., 2020)	X		X	X		X		X		X		X				X	
(Khodabandeh et al., 2020)	X	X	X	X	X		X			X			X				
(Manavizadeh et al., 2020)	X	X	X	X	X		X				X		X	X			
(Zhan et al., 2021)	X	X	X		X						X						

Cette analyse met en évidence le très peu de travaux (8% des travaux, 2/26) prenant en compte les exigences d’inter-intervention entre les activités et les préférences des acteurs (la colonne « Ressources humaines », 12% des travaux, 3/26). Nous trouvons là des pistes intéressantes d’exploitation de tolérances possibles pour formaliser des notions de satisfaction.

Si les attentes de chaque catégorie d’acteur de la PAD sont consignées dans ce Tableau 2, il semble logique qu’une satisfaction totale de bénéficiaire s’oppose à une faculté à rendre le service attendu. Il en va de même pour un intervenant. Il s’en dégage une forme d’équilibre à trouver pour un coordinateur entre l’offre et la demande. Enfin, les horizons au niveau opérationnel sont souvent les mailles de temps à privilégier pour caractériser cette recherche d’équilibre.

2.2.2. Recherche de performance de la PAD par la structure en charge de la coordination

Puisque les structures de PAD sont soumises à la recherche d’équilibre dans le champ des possibilités en jouant sur les exigences négociables, fixer un cap pour obtenir une bonne coordination conduit à spécifier ce qui détermine cette qualité voulue. Les enjeux traditionnels de rentabilité, de satisfaction client ou encore de développement de la patientèle existent aussi dans ce type de structure. Mais la recherche de valeur ajoutée par comparaison avec l’offre de structures concurrentes n’est pas sans fondement dans une économie de santé où les structures libérales ont le droit d’exercer. L’innovation par la prestation de services est alors aussi une clé de performance. A titre d’exemple, évoquons le niveau de satisfaction des bénéficiaires amélioré par un respect des affinités entre bénéficiaires et intervenants. Quant à la satisfaction des intervenants, elle pourrait augmenter avec le maintien permanent d’une charge de travail équilibrée et raisonnable entre eux, voilà une autre voie qui peut aussi se révéler être innovante.

Afin de donner du sens à leur pilotage, les coordinateurs doivent donner des orientations et les concrétiser dans leur décision. Ces dernières décrivent généralement ce qui doit être fourni ou réalisé pour produire de la valeur. Elles peuvent se décliner sous forme d’une palette d’objectifs à atteindre ou de risques à éviter. Nous traitons cette question selon trois points de vue dans ce qui suit.

2.2.2.1. Contrôle des temps

Dans l’abondante littérature consacrée au pilotage des processus métiers de PAD, nous observons de nombreux critères de performance. Le critère le plus usité est le temps de déplacement, ce qui revient à utiliser les ressources sur du temps de service aux bénéficiaires. Dans (Méndez-Fernández et al., 2020), la planification du travail doit minimiser le temps perdu entre deux services consécutifs, incluant le temps de déplacement. La minimisation de la durée totale des déplacements a aussi une implication sur les coûts déplacement qu’il faut contrôler (Yalçındağ et al., 2016). Dans certaines publications, les auteurs proposent des approches multicritères d’aide à la décision (Liu et al., 2018) (Fathollahi-Fard et al., 2019).

2.2.2.2. Contrôle des coûts

Le coût du transport est donc l'un des principaux leviers économiques pour les structures de PAD (Fathollahi-Fard et al., 2020). Ainsi, ces structures cherchent à le minimiser (En-nahli et al., 2015).

(Hiermann et al., 2015) présentent un cadre général pour résoudre un problème d'ordonnancement de PAD dans un environnement multimodal. L'objectif de ce cadre est d'affecter les intervenants aux bénéficiaires et de déterminer des tournées multimodales d'une façon efficace, afin de réduire le coût de transport, tout en considérant la maximisation de la satisfaction personnelle au niveau du respect des horaires préférentiels spécifiés. Dans (Mankowska et al., 2014), la distance parcourue par les intervenants est minimisée quotidiennement, ce qui correspond à une minimisation du coût total des déplacements (Quintanilla et al., 2020).

Pour couvrir toutes les demandes, le coût d'exploitation est un autre facteur économique. Il s'agit de mobiliser le juste effectif d'intervenants pour couvrir la demande de services (Manavizadeh et al., 2020). Cela doit être réalisé sans perte de qualité de l'intervention, avec la garantie d'une durée d'intervention suffisante et le respect des disponibilités spécifiées par les bénéficiaires (Yuan and Fügenschuh, 2015). Dans (Khodabandeh et al., 2020), un compromis est envisagé entre le coût total du déplacement des infirmières et les coûts de rétrogradation. Puisque les coûts de rétrogradation (discuté dans la section 2.2.1.1 de ce chapitre) sont porteurs de forte dégradation de performance économique.

Dans les travaux de (Yalçındağ et al., 2016), si les capacités de travail sont dépassées, les intervenants doivent absorber une surcharge. Des heures supplémentaires entraînent un coût significatif puisque les taux horaires sont différents. Dans (Shao et al., 2012), si les heures de travail effectuées pendant une semaine pour chaque intervenant sont au-dessus de 40 heures, les heures supplémentaires doivent être payées en plus. Dans (Carello and Lanzarone, 2014), le but est de réduire les coûts liés aux heures supplémentaires des intervenants jour par jour.

2.2.2.3. Equilibrage de pénibilité des tournées

Dans le contexte de la prise en charge à domicile, la pénibilité des tournées peut se caractériser soit par un rythme de travail tendu généré par un nombre de visites important, soit par un environnement physique agressif lié à la complexité de l'intervention. Ainsi, la répartition égale des interventions, dans leur degré de difficulté comme dans leur charge, entre les intervenants est un levier possible pour ne pas générer d'inégalités. Dans (Yalçındağ et al., 2016) les taux d'utilisation de chaque intervenant sont appliqués pour réaliser l'équilibrage de la charge personnelle. Ces taux sont définis comme étant le ratio entre la charge de travail réelle de l'intervenant et sa capacité de travail théorique. Dans (Quintanilla et al., 2020), les auteurs cherchent à équilibrer le nombre de visites de chaque intervenant.

Par ailleurs, l'équilibrage peut se faire sur la base d'une échelle de gravité médicale (mesurée par l'indicateur GIR en France, par exemple) influant sur les affectations entre bénéficiaire et intervenant. Ainsi, dans (Gayraud et al., 2013), l'objectif est de répartir les bénéficiaires (les patients âgés) fortement dépendants (peu autonomes) parmi les différentes tournées.

2.2.2.4. Synthèse

Le Tableau 3 récapitule les orientations qu'une structure de PAD peut sélectionner pour définir son pilotage et peser sur les processus de l'organisation. Ils sont issus d'une collecte effectuée dans plusieurs références bibliographiques.

Tableau 3 : Synthèse des orientations recherchées par des structures de PAD

Référence	Contrôle des temps			Contrôle des coûts				Equilibrage de pénibilité des tournées			
	Temps de déplacement	Charge totales	Temps inactivité	Trajet	Exploitation	Heures supplémentaires	Rétrogradation	Equilibrage de charge	Equilibrage de nombre de visites	Equilibrage de routage véhicule	Equilibrage de difficulté des tournées
(Shao et al., 2012)				X	X	X					
(Gayraud et al., 2013)	X										X
(Trautsamwieser and Hirsch, 2014)		X									
(Carello and Lanzarone, 2014)						X					
(Mankowska et al., 2014)	X			X							
(B. Yuan et al., 2015)				X	X						
(En-nahli et al., 2015)	X		X					X			
(Z. Yuan and Fügenschuh, 2015)		X			X						
(Hiermann et al., 2015)		X									
(Duque et al., 2015)		X									
(Rest & Hirsch, 2016)	X		X								
(Redjem and Marcon, 2016)		X	X								
(Yalçındağ et al., 2016)		X						X			
(Du et al., 2017)				X	X	X					
(Nasir & Dang, 2018)	X		X					X			
(Lin et al., 2018)				X		X					
(T. Zhang et al., 2018)		X	X	X							
(Bazirha et al., 2019)			X								
(Çolak et al., 2019)	X										
(Chabane et al., 2019)				X	X						
(Fathollahi-Fard et al., 2020)	X			X							
(Goodarzian et al., 2020)	X	X		X						X	
(Quintanilla et al., 2020)	X			X				X			
(Méndez-Fernández et al., 2020)	X	X									
(Khodabandeh et al., 2020)	X						X				
(Manavizadeh et al., 2020)	X			X	X						
(Zhan et al., 2021)			X	X							

Ce tableau permet de mettre en évidence que la majorité des travaux s'intéressent à la minimisation des temps (78%, 21/27) et que la moitié d'entre eux s'intéressent aux coûts (56%, 15/27). Nous notons également que ces travaux tiennent davantage compte de la rentabilité de la structure et marginalisent la satisfaction des intervenants et des bénéficiaires.

Enfin, peu de travaux (22%, 6/27) abordent l'équilibrage de pénibilité des tournées dans leur mode de coordination. Côté intervenant, l'équilibrage de la charge de travail (11% des travaux,

3/27) évite une diminution de leur satisfaction due à des dépassements excessifs des prévisions. L'équilibrage de difficulté (4% des travaux, 1/27) des tournées empêche le surmenage et la dégradation de la qualité des interventions, ce qui constitue un risque lorsqu'un intervenant effectue trop de soins difficiles. Côté bénéficiaire, la qualité des soins sera améliorée grâce à l'équilibrage des pénibilités. Côté structure de PAD, l'équilibrage du nombre de visites (1/27) et de routage véhicules (1/27) est un moyen de garantir une utilisation efficace du temps de travail des intervenants, à savoir la maximisation de taux d'occupation des intervenants ($\text{occupation\%} = \frac{\text{Temps}_{\text{opération total}} + \text{Temps}_{\text{déplacement total}}}{\text{Charge totale}}$).

2.3. Bilan des travaux antérieurs sur l'aide à la décision pour la coordination de PAD

La recherche opérationnelle s'intéresse aux problèmes de coordination. Eu égard à la structuration des problèmes dans cette spécialité, la coordination de la PAD a des interfaces évidentes avec la gestion des compétences sur les ressources humaines (affectation des compétences aux activités, par exemple), et aussi avec les problèmes de tournées (voyageur de commerce, par exemple). Cette optimisation de la coordination est souvent abordée sous la forme d'un problème d'optimisation des plannings des intervenants, un sujet très bien étudié par cette communauté scientifique. Cette classe de problèmes est décrite dans la littérature sous l'intitulé « *Home Health Care Routing and Scheduling Problem - HHCRSP* ».

La présente section relate une revue de la littérature dans ce domaine, permettant d'appréhender les méthodes les plus usitées pour la résolution des problèmes HHCRSP et les concepts clés afférents.

2.3.1. Le problème de la planification des interventions dans la prise en charge à domicile (HHCRSP)

Le problème de la planification des interventions de PAD consiste à décider quelles ressources humaines doivent être affectées, et à quel moment, pour assurer les interventions pendant l'horizon de planification afin de satisfaire les demandes de chaque bénéficiaire (Fard et al., 2018) (Demirbilek et al., 2019). La littérature scientifique est conséquente et montre toute l'étendue de ce problème complexe. Le problème a progressivement évolué d'une vision centrée sur la définition de tournées à domicile des personnels infirmiers vers une vision pluriprofessionnelle avec prise en compte des compétences nécessaires pour une prise en charge de qualité du bénéficiaire tenant compte de l'environnement social du demandeur (Lanzarone and Matta, 2014).

De manière générale, les travaux qui s'intéressent à ce problème, œuvrent dans un cadre de réutilisation d'algorithmes bien connus pour résoudre d'autres problèmes présentant des similitudes avec une formulation adaptée du problème HHCRSP. Parmi ces problèmes, nous pouvons citer le problème du voyageur de commerce, ainsi que celui de tournées de véhicules. (Di Mascolo et al., 2021) (Cissé et al., 2017) (Fikar and Hirsch, 2017) (Marcon and Chaabane et al., 2016) mettent en évidence des modèles de recherche opérationnelle développés récemment pour résoudre un problème HHCRSP.

2.3.1.1. Problèmes adjacents au HHCRSP

- Problème du voyageur de commerce

(Du et al., 2017) ont identifié le problème du routage des intervenants à domicile en tant qu'une variante du problème du voyageur de commerce.

Le problème du voyageur de commerce (*Travelling Salesman Problem*, TSP) a été initialement adressé par Sir William Rowan Hamilton dès le XIXe siècle (Martinez, 2020). Il formule le problème suivant : « un voyageur est tenu de visiter les villes données une seule et unique fois, en partant de n'importe quelle ville et en retournant au lieu de départ initial. Quel itinéraire choisir afin de minimiser la distance totale parcourue ? » (Lin, 1965). La formulation a été originellement proposée par (Dantzig et al., 1954). Ce problème peut être ainsi vu comme le besoin de définir une « simple-tournée ».

D'après (Gutin and Punnen, 2006), de nombreuses variantes du TSP existent. Parmi elles, le « multi-TSP » est un énoncé par lequel chaque voyageur visite un sous-ensemble des villes exactement une fois et retourne à son point de départ. L'état de l'art de (Bektas, 2006) fait une étude systématique sur les applications pratiques du « multi-TSP », et sur la manière de réussir à étendre ce « multi-TSP » à une grande variété de problèmes de tournées, tissant un lien fort entre ces deux classes.

Plus récemment, (Bahadori-Chinibelagh et al., 2019), (Erdem and Koç, 2019) et (Fatemi Ghomi et al., 2018) considèrent le HHCRSP comme le problème du voyageur de commerce avec des fenêtres de temps qui s'ajoutent aux exigences.

- Problème de tournées des véhicules

Le problème de tournées de véhicules (VRP), initialement introduit par (Dantzig and Ramser, 1959), consiste à collecter ou distribuer des objets auprès de chaque client, à l'aide d'une flotte de véhicules, sachant que chaque véhicule peut desservir plusieurs clients avec une capacité limitée. Ce problème est une extension classique du problème du voyageur de commerce, appartenant à la classe des problèmes NP-difficiles (Lenstra and Kan, 1981). La formulation VRP couvre de nombreux domaines d'application différents, tels que la distribution de marchandises, de journaux, de courrier et de besoins médicaux, ainsi que la collecte de déchets chimiques, qui ne peuvent être résolus facilement avec les méthodes traditionnelles (Çolak et al., 2019).

Il existe de nombreuses variantes du VRP selon (Eksioglu et al., 2009) (Braekers et al., 2016b). Parmi ces variantes, le problème de tournées des véhicules avec des fenêtres de temps (VRPTW) est le plus étudié, en lien avec le HHCRSP (Torres-Ramos et al., 2014), le concept « fenêtre de temps » (discuté dans la section 2.2.1.3) est appliqué pour représenter la flexibilité à la fois de la demande (intervalle des services demandés) et de l'offre (intervalle de temps de travail). Dans plusieurs travaux sur le HHCRSP, le problème est considéré comme une extension du problème de tournées des véhicules (Fernandez et al., 2018) (Goodarzian et al., 2020).

2.3.2. Incertitudes liées au HHCRSP opérationnel

Le HHCRSP vise à produire le planning des interventions à domicile en vue de l'assurance d'une bonne coordination. Cependant, la production d'un planning est souvent réalisée d'une manière ad-hoc à mesure que les problèmes se posent. C'est un véritable challenge organisationnel de faire face aux événements incertains. Ces événements imprévisibles, par exemple un intervenant

changeant sa disponibilité, entraînent une perturbation du planning initial pour les intervenants et les bénéficiaires.

Dans la coordination opérationnelle, cinq types d'évènements incertains sont susceptibles de se produire dans le temps :

- Changement des disponibilités des intervenants.(Lin et al., 2018) (Yuan and Jiang, 2017)
- Changement des disponibilités des bénéficiaires. (Lin et al., 2018) (Yuan and Jiang, 2017)
- Variabilité de la durée des interventions. (T. Zhang et al., 2018)
- Annulation des interventions des bénéficiaires durant les tournées des intervenants.(Lin et al., 2018) (Yuan and Jiang, 2017)
- Problème inopiné durant la tournée de l'intervenant (panne de véhicule, embouteillages par exemple).(Lin et al., 2018)

En ce sens, la prise en compte de l'incertitude dans la recherche d'une solution de coordination opérationnelle a pour but ultime d'avoir une capacité de réaction face à ces événements incertains. La robustesse d'une solution est alors recherchée par la structure en charge de la coordination. Elle est définie comme la capacité d'un système à maintenir un niveau de performance satisfaisant en présence de variations défavorables des conditions de fonctionnement (Greco et al., 2012). A défaut, si le pilotage effectif d'une PAD n'arrive pas à respecter le planning initial et que des services commencent à être impossibles à satisfaire, il faudra développer de la résilience, qui désigne la capacité d'un système à absorber l'impact d'une perturbation majeure et à poursuivre les opérations ou les services critiques (Burnard and Bhamra, 2011).

2.3.3. Méthodes de résolution

Selon la nature mathématique des variables caractérisant les décisions à prendre par un coordinateur, la préférence donnée à un ou plusieurs critères de performance, les types de contraintes pesant sur l'offre et la demande qui sont admises dans le champ de formulation mathématique du problème, les solutions varient. Conscient de ces caractéristiques, une méthode de résolution du HHCRSP sert à trouver une ou plusieurs solutions. Souvent la méthode de résolution choisie se trouve très adaptée à un type particulier de formulation. Dès lors, il est flagrant de constater que la taille des problèmes de terrain induit des difficultés en termes de temps de calcul et relativise la portée opérationnelle de beaucoup de solutions séduisantes.

Deux approches sont classiquement répertoriées en recherche opérationnelle : les méthodes exactes et les méthodes approchées. Les méthodes hybrides mélangent ces deux types.

2.3.3.1. Méthodes exactes

Les méthodes exactes consistent à explorer l'intégralité des combinaisons possibles pour construire une solution. L'espace de recherche peut être structuré en arbre ou treillis, ce qui permet d'utiliser des techniques d'élagages (*branch and bound*, par exemple) pour réduire l'espace de recherche et orienter l'exploration de l'espace de recherche. La complexité de ces algorithmes est en général exponentielle (Solnon, 2013). Les techniques d'élagage et les heuristiques ne sont pas assez efficaces pour éviter l'explosion combinatoire dans le cas général.

Par l'utilisation des méthodes exactes, nous sommes généralement capables de trouver une solution optimale par une Programmation Linéaire en Nombres Entiers (PLNE) ou par une Programmation Linéaire en Nombres Mixtes (PLNM). En pratique, les méthodes exactes ne permettent pas de résoudre les cas de taille réelle (plus de 100 interventions à planifier pour un jour, par exemple) car les temps de calcul sont très sensibles à la taille du problème. Elles sont souvent opérées à l'aide de solveurs de programmation mathématique tels que CPLEX¹, Gurobi², et FICO Xpress³ ou par un logiciel de modélisation mathématique GAMS⁴.

Citons les travaux de (Yalçındağ et al., 2016) où une approche simultanée a été proposée, qui permet de prendre à la fois les décisions d'affectation et de routage. L'approche proposée est formulée par un modèle en PLNM et résolue par le solveur CPLEX sur un petit cas d'étude comportant 25 bénéficiaires et 8 intervenants. Dans (Quintanilla et al., 2020), GAMS a été utilisé pour résoudre un problème d'équilibrage de nombre de visites réalisées par les intervenants. Dans (Méndez-Fernández et al., 2020), les auteurs ont recours au solveur Gurobi pour déterminer la solution d'un problème de planification des interventions hebdomadaires, formulé en PLNE et sur un cas de taille réduite (14 bénéficiaires et 3 intervenants). Dans (Rest and Hirsch, 2016), la formulation du problème de routage est basée sur un modèle PLNM et résolue par le solveur FICO Xpress pour un cas d'étude de 6 intervenants et 30 interventions à planifier.

Au regard de la pluralité des objectifs de performance, l'approche ϵ -contrainte est l'une des méthodes les plus populaires de l'optimisation multi-objectifs (Haimes, 1971). Elle tente d'optimiser l'objectif le plus important en considérant des bornes supérieures ou inférieures pour les autres objectifs. En fait, dans cette méthode, l'objectif principal est considéré comme la fonction objectif, et les autres sont ajoutés comme des contraintes du modèle. Un front de Pareto peut être obtenu par variation systématique des bornes des contraintes (Khodabandeh et al., 2020). Pour cet auteur, une optimisation bi-objectif basée sur ϵ -contrainte est proposée pour trouver un compromis entre la réduction du temps de déplacement des intervenants et la réduction des coûts de rétrogradation des intervenants sur un cas expérimental (25 bénéficiaires, 5 intervenants).

Pour repousser les limites d'une durée de calcul exponentielle, l'utilisation d'une méthode exacte adossée à une technique de relaxation, telle que la relaxation lagrangienne (Fisher, 1981), peut réduire significativement le temps de calcul en garantissant une bonne qualité de solutions (Fathollahi-Fard et al., 2020). Pour ce faire, la formulation de PLNE/PLNM devrait être relaxée en éliminant une série de contraintes du modèle et en les ajoutant à la fonction objectif (Mestria, 2018) avec des coefficients dits multiplicateurs de Lagrange qui seront des variables d'optimisation (Golshahi-Roudbaneh et al., 2017), (Astorino et al., 2018). Notons que cette approche est couramment employée quand des relations sont des fonctions non linéaires des variables.

¹ CPLEX Optimiseur : <https://www.ibm.com/fr-fr/analytics/cplex-optimizer>

² Gurobi : <https://www.gurobi.com/>

³ FICO Xpress Optimisation : <https://www.fico.com/en/products/fico-xpress-optimization>

⁴ GAMS (General Algebraic Modeling System): <https://www.gams.com/>

2.3.3.2. Méthodes approchées

Les méthodes approchées ne permettent pas de garantir l'obtention de « la meilleure » solution, mais elles se limitent à trouver « une bonne » solution.

Elles s'appuient sur des algorithmes à base de métaheuristiques qui sont des alternatives sérieuses aux méthodes exactes grâce à leur capacité à explorer efficacement de très grands espaces de recherche avec un temps de calcul raisonnable. La communauté de recherche opérationnelle est très prolifique ces dernières années sur ces algorithmes de recherche locale ou globale, par voisinages successifs permettant de s'attaquer à des problèmes très complexes. Notons que le voisinage d'une solution est un sous-ensemble de solutions qu'il est possible d'atteindre par une série de transformations astucieuses (Marmion, 2011).

La qualité d'une méthode approchée est souvent évaluée par une étude de l'écart de résultat avec une approche exacte sur un problème de petite taille. Cette comparaison peut aussi se faire relativement à un planning construit au fil du temps par un coordinateur expérimenté, selon une approche empirique.

Les méthodes approchées sont souvent des approches itératives, elles essayent d'explorer intelligemment l'espace de recherche par (1) l'exploitation d'informations collectées à chaque itération pour guider vers les zones prometteuses, (2) la garde d'une part d'exploration aléatoire de l'espace de recherche pour ne pas s'enfermer dans une zone restreinte et ainsi laisser ouverte une porte de sortie. Un avantage de ces approches est de fournir au moins une solution, même si l'algorithme est interrompu avant la condition de fin (après au moins une itération). La complexité de ces approches est en général polynomiale (Solnon, 2013).

Une stratégie de recherche locale (Aarts et al., 2003) a souvent été appliquée pour résoudre le HHCSP. En raison de leur faiblesse, plusieurs métaheuristiques étendant les stratégies de recherche locale ont été proposées pour éviter d'être piégé dans un optimum local, telles que celles évoquées ci-dessous.

- La recherche de voisinage variable – VNS

La recherche de voisinage variable a été proposée par (Mladenović and Hansen, 1997). Elle est basée sur l'idée de changements systématiques de la structure des voisinages en vue de trouver une meilleure solution. VNS procède par une méthode de descente en explorant une série de voisinages prédéfinis pour trouver un minimum local. Chaque fois, un point de départ est généré de manière aléatoire en utilisant le voisinage actuel dans une phase dite de secourage pour exécuter une descente locale. Le nouveau minimum local trouvé est comparé à l'actuel, la recherche bondit vers la nouvelle solution si, et seulement si, elle est meilleure. Par conséquent, la VNS n'est pas une méthode de parcours de trajectoire et ne spécifie pas de mouvements interdits (Nasir and Dang, 2018).

Dans (Bazirha et al., 2019), la VNS est proposée pour traiter les cas volumineux en recherchant une bonne solution en un temps de calcul très court vis-à-vis d'un modèle de planification à l'horizon journalier avec les multiples disponibilités des bénéficiaires. Dans (Nasir and Dang, 2018), un modèle de PLNM flexible est résolu par une méthode VNS, qui considère l'arrivée et le départ dynamiques des bénéficiaires, ainsi que l'embauche de nouveaux intervenants avant la mise en service.

- La recherche tabou - TS

La TS a d'abord été proposée par (Glover and Laguna, 1998). TS est caractérisée par l'utilisation d'une « liste tabou » qui consiste à bloquer toute zone voisine où la recherche a déjà été effectuée. Pour éviter de tomber dans la solution optimale locale, cette liste de tabous est adoptée dans TS pour enregistrer et sélectionner la solution de voisinage générée, cela étant destiné à guider la direction de recherche suivante. Le choix des paramètres algorithmiques de TS, à savoir la taille de la liste tabou, le nombre maximal d'itération et le nombre de solutions voisines, est étudié par (Sarmady, 2012).

Le TS est massivement utilisé pour résoudre le HHCRSP. Par exemple, pour la résolution d'un problème de planification de tournées dans la PAD avec la prise en compte de moyens de transport (Szander et al., 2019) ou bien pour la résolution d'un problème d'équilibrage de la charge de travail des intervenants (Hertz and Lahrichi, 2009). Il y a également les travaux (Rest and Hirsch, 2016) sur un problème de planification quotidienne en combinant des modes de transport public lors des tournées du personnel de l'ARC (*Austrian Red Cross*) de Vienne. La taille du plus grand problème résolu est 46 intervenants et 202 interventions. Les solutions sont comparées empiriquement avec un planning réaliste fourni par l'ARC. Dans (Lamine et al., 2019), les auteurs ont mis en place une plateforme de coordination des interventions à domicile dans laquelle ils ont eu recours au TS pour générer des plannings de tournées tenant compte des préférences temporelles des bénéficiaires. Nous pouvons aussi citer les travaux de (Liu et al., 2014) (Triki et al., 2014) portant sur la mise en place d'une logistique périodique des soins à domicile, ou ceux de (Yuan and Jiang, 2017) concernant un problème de planification en temps réel avec gestion des perturbations.

- Le recuit simulé – SA

Le recuit simulé est à l'origine abordé par (Kirkpatrick et al., 1983), c'est une métaheuristique basée sur une analogie avec la simulation du comportement d'un matériau métallique lors d'un traitement thermique (le recuit). Le choix des paramètres de SA à savoir la température initiale, la température du système, la loi de décroissance de la température, les critères d'arrêt, est étudié par (Ben-Ameur, 2004).

SA est largement utilisé pour la génération de solutions approximatives de HHCRSP. Dans (Manavizadeh et al., 2020), le recuit simulé est utilisé pour traiter un problème comportant des interventions multiples (un bénéficiaire demande plusieurs interventions par jour) et interdépendantes avec des fenêtres de temps souples. De surcroît, un problème de distribution et de ramassage (*pick-up*) simultanés (Shi et al., 2018), un problème de planification de ressources humaines (He and Shi, 2019), sont tous résolus par SA.

- Autres algorithmes métaheuristiques

Dans (Goodarzian et al., 2020), un algorithme de l'ISEO (*Improved self-adaptive Social Engineering Optimizer*) pour les cas de grande taille (à un maximum de 133 bénéficiaires, 56 intervenants par jour) a fourni une solution très proche de l'optimum. Le problème abordé est un problème bi-objectif de routage et d'ordonnancement de services de soins à domicile avec des fenêtres de temps. Trois métaheuristiques inspirées de la nature, basées sur l'intelligence des essaims, ont fait l'objet d'une étude comparative avec l'ISEO, à savoir les algorithmes FFA (*FireFly Algorithm*)

(Yang, 2010) , ABC (*Artificial Bee Colony algorithm*) (Karaboga, 2005), SEO (*Social Engineering Optimizer*) (Fathollahi-Fard et al., 2018).

Dans (Shao et al., 2012), le problème de la planification des horaires des intervenants est formulé comme un modèle de PLNM. Mais ce modèle comporte plusieurs composants limitant sa formulation sous cette forme, à l'instar des différentes classes de bénéficiaires, c'est à dire des modèles de traitement hebdomadaires facultatifs ou d'une structure de paiement complexe. Pour ce faire, l'approche glouton GRASP (*Greedy Randomized Adaptive Search Procedure*) développée par (Feo et al., 1991), a été utilisée. Elle a aussi été utilisée dans (Quintanilla et al., 2020) en vue de la résolution du problème de routage multimodal.

Dans (Chabane et al., 2019), le problème de planification de la tournée d'intervenants est formulé comme étant un problème de satisfaction de contraintes (*Constraint Satisfaction problem - CSP*), dans la mesure où il faut trouver une affectation d'intervenants, avec des créneaux horaires bien définis, respectant des critères d'affectation. Ce problème ainsi formulé a été ensuite résolu par des algorithmes de colonies de fourmis (*Ant Colony Optimization - ACO*). De même, dans (T. Zhang et al., 2018), et pour optimiser la planification des interventions à domicile, un ACO modifié a été proposé pour tenir compte de la répartition dense des bénéficiaires dans les communautés chinoises, qui sont différentes de celles d'Europe.

Dans (Çolak et al., 2019), l'algorithme des gains de Clarke et Wright (Clarke and Wright, 1964) a été utilisé pour établir l'ensemble de tournées des intervenants en combinant les points à visiter, par une assurance du plus grand gain au niveau de coût de trajet (temps de déplacement minimum).

(Du et al., 2017) proposent une approche à base d'un Algorithme Génétique (*Genetic Algorithm - GA*) couplé à une recherche locale pour résoudre le modèle de PLNE proposé en ordonnancement des services de soins à domicile. Idem, dans (Yalçındağ et al., 2016), un GA a été utilisé pour résoudre l'affectation des intervenants et leur routage d'une manière séquentielle. La taille du problème résolu est réaliste, il est composé de 150 bénéficiaires et de 15 intervenants par jour.

(Redjem and Marcon, 2016) présentent un algorithme heuristique, nommé CRH (*Caregivers Routing Heuristic*), qui se déroule en deux étapes pour résoudre le problème de routage de taille réelle. La première étape consiste à mettre en place les tournées optimales avec les temps de déplacement les plus courts sans la prise en compte de contraintes liées aux bénéficiaires. La deuxième étape intègre les contraintes relaxées pour construire la solution finale. Cet algorithme CRH a permis d'obtenir des meilleurs résultats que l'ancienne approche proposée par ces mêmes auteurs (Redjem et al., 2011) où, avec le même temps de calcul, on peut passer de « 10 intervenants, 10 bénéficiaires » à « 30 intervenants, 60 bénéficiaires ».

(Hiermann et al., 2015) choisissent aussi une approche basée sur deux étapes en formulation CSP. Dans la première étape, ils génèrent des solutions initiales par une procédure aléatoire. Au cours de la deuxième étape, les solutions initiales générées sont itérativement améliorées en appliquant l'une des quatre approches méta-heuristiques suivantes : VNS, l'algorithme mémétique, la recherche de dispersion ou le recuit simulé.

Dans (Duque et al., 2015), une approche de planification en deux étapes a été proposée : l'optimisation du niveau de service et l'optimisation de la distance totale parcourue.

L'application de l'algorithme de recherche locale dans la résolution des deux étapes a conduit à une excellente performance en termes de satisfaction de préférence personnelle (intervenants préférés), de satisfaction des exigences temporelles (fenêtres de temps exigées) des bénéficiaires et de la distance du trajet d'acheminement des intervenants. La qualité des solutions a été examinée en rapport avec planning réaliste fourni par *Landelijke Thuiszorg*, une organisation offrant des services de soins à domicile dans plusieurs régions belges.

2.3.3.3. Approches hybrides

Les approches hybrides se basent généralement sur la combinaison d'une méthode exacte et d'une méthode approchée.

(Yuan et al., 2015) adoptent une approche hybride basée sur la décomposition de Dantzig–Wolfe (Dantzig and Wolfe, 1960) et ceci pour transformer le problème de tournée des intervenants formulé sous un modèle PLNM en un problème maître et un sous-problème de prix (*pricing problem*). Le problème maître est d'abord résolu par l'algorithme Branch and Bound embarqué dans le solveur CPLEX. Les algorithmes gloutons, VND (*Variable Neighbourhood Descent*) et LB (*Label Algorithm*) sont ensuite utilisés pour résoudre le sous-problème. De même, dans (Trautsamwieser and Hirsch, 2014), la formulation du problème a suivi la décomposition de Dantzig–Wolfe et ensuite une approche BPC (*Branch-Price-and-Cut*). Pour fournir à l'algorithme BPC les bonnes bornes supérieures avec un temps de calcul réduit, un algorithme VNS a été développé.

L'efficacité du calcul d'une méthode exacte peut être considérablement améliorée grâce à une initialisation via une méthode approchée. (Yuan and Fügenschuh, 2015) développent un algorithme glouton et de recherche locale pour construire une solution initiale faisable. Puis le problème formulé en PLNE optimise cette solution initiale à travers le solveur CPLEX. L'instance de test est un cas réel, fourni par un prestataire local de services de soins à domicile. Il contient 99 bénéficiaires, 460 interventions et 9 intervenants pour une semaine. Dans (Martinez et al., 2018), un modèle en PLNE a été développé pour planifier les interventions quotidiennes sur l'échelle hebdomadaire. La solution initiale est obtenue par un algorithme glouton, avant d'être optimisée par CPLEX.

Dans (Fathollahi-Fard et al., 2020), l'algorithme hybride VNS-SA (*Variable Neighbour Search - Simulated Annealing*) est proposé afin d'optimiser une solution initiale par une affectation aléatoire. Concernant le processus d'optimisation, le VNS fonctionne comme la boucle principale de l'algorithme et le SA constitue une sous-boucle. Le problème résolu est pour 200 bénéficiaires et 20 intervenants.

2.3.3.4. Approches liées aux incertitudes opérationnelles

Le problème d'organisation des interventions de PAD est rendu difficile en partie à cause de l'incertitude liée à l'horizon de planification choisi (Shi et al., 2018) (Du et al., 2019) (Shi et al., 2019). La difficulté réside dans le fait que les temps d'interventions chez les bénéficiaires ne sont généralement pas déterministes. D'ailleurs à ce titre, dans la plupart des travaux de recherche, à l'instar de (Yuan et al., 2015), (Lanzarone and Matta, 2014) et de (Carello and Lanzarone, 2014), les durées des interventions sont considérées comme étant stochastiques. Compte tenu de ce

caractère stochastique, l'heure d'arrivée de l'intervenant au domicile de chaque bénéficiaire peut être avancée ou retardée par rapport à l'heure de rendez-vous (Yuan et al., 2015).

(Lanzarone et al., 2010) proposent une approche pour développer un tel modèle stochastique afin d'étudier les changements dans les conditions cliniques, fonctionnelles et sociales des bénéficiaires recevant un service de PAD. Ces changements peuvent influencer la durée de chaque visite. Le modèle proposé se compose d'une chaîne de Markov, qui a bien fonctionné dans l'analyse d'un cas réel et qui peut être appliquée aux intervenants pour contrôler la variabilité de la durée de chaque visite.

Une perturbation exogène lors des tournées des intervenants, telle que les accidents et les embouteillages, est un autre facteur d'incertitude (discuté dans la section 2.3.2 de ce chapitre). En effet, ces perturbations entraînent de l'incertitude dans le calcul du temps de déplacement des intervenants vers leurs points de rendez-vous. Dans (Carello and Lanzarone, 2014), une solution robuste qui prend en compte les incertitudes de temps pour le routage des véhicules des intervenants d'une structure de PAD est présentée. Le modèle développé est testé sur des instances réelles liées à des intervenants opérant en Italie. L'estimation du temps nécessaire à la visite d'un ensemble de bénéficiaires, qui sont situés dans des lieux géographiques spécifiques, peut être réalisée par l'observation des données historiques lors des périodes précédentes. L'estimateur Kernel (Wand and Jones, 1994) est introduit par (Yalçındağ et al., 2016) pour prédire le temps de déplacement opérationnel. Ce mécanisme est utilisé dans une approche en deux étapes qui résout séquentiellement les problèmes d'affectation et de routage. (Rest and Hirsch, 2016) proposent une approche pour calculer les temps de déplacement dépendant des horaires des transports collectifs, qui sont toujours incertains.

La durée d'intervention et le temps de déplacement sont considérés comme des variables stochastiques dans (Zhan et al., 2021). Le modèle de PLNM avec une méthode *L-shaped* (Van Slyke and Wets, 1969) a été établi pour déterminer la tournée des intervenants et les heures de rendez-vous des bénéficiaires de sorte à minimiser les coûts totaux de déplacement et de temps libre des intervenants, et de temps d'attente des bénéficiaires. Ceci est supporté par une programmation stochastique avec un mécanisme de linéarisation afin de prendre en compte des attributs stochastiques.

Grâce aux progrès technologiques, de nombreux établissements sanitaires sont équipés de dispositifs de communications mobiles sans fil et d'objets connectés (Milburn and Spicer, 2013). Avec ces équipements, les structures de PAD peuvent garder le contrôle de l'état de leurs bénéficiaires. Lorsqu'un bénéficiaire subit un incident soudain à son domicile, par exemple, une chute ou une hypoglycémie, la structure est capable d'obtenir des informations à distance via les équipements médicaux installés chez bénéficiaires, puis d'envoyer des intervenants pour fournir des interventions immédiates. Une telle intervention soudaine chez un bénéficiaire entraîne la nécessité d'une nouvelle affectation au moment opportun. Dans ce contexte, (Lin et al., 2018) ont utilisé l'algorithme HSA (*Harmony search algorithm*) pour investiguer ce problème qui envisage conjointement du routage, de l'affectation et réaffectation des ressources, et inclut deux modèles. Le premier modèle détermine simultanément l'affectation des intervenants et l'acheminement des véhicules au domicile des bénéficiaires, en vue de minimiser le coût total des heures supplémentaires des intervenants et du routage des véhicules. Le second modèle prend en compte la réaffectation des ressources due à la survenue d'incidents soudains chez les

bénéficiaires. Par conséquent, ce modèle de réaffectation détermine une nouvelle solution qui minimise non seulement l'objectif du premier modèle, mais aussi la différence entre la solution originale et la nouvelle solution, car les intervenants préfèrent une nouvelle solution qui ne diffère pas beaucoup de la solution originale.

2.3.4. Synthèse

Le Tableau 4 propose une qualification des approches de résolution extraites de ces travaux passés selon une série de critères qualificatifs de la méthode utilisée pour résoudre le problème de coordination. Nous avons ajouté le critère « type de formulation du problème » qui permet de renseigner la forme mathématique utilisée pour modéliser le problème étudié. Le champ « Type » indique la catégorie du problème de coordination abordé à l'aune des dimensions structurantes du problème de coordination évoquées dans la partie spécification.

Tableau 4 : Synthèse sur les approches d'aide à la décision pour la coordination

Référence	Type		Type de formulation du problème	Méthode exacte		Méthode approchée		Méthode hybride	Approche liée aux incertitudes		
	Affectation	Planification de tournées		Si utilisé ?	Méthode	Solveur	Si utilisé ?		Algorithme	Rotation des véhicules	Durée des interventions
(Lanzarone et al., 2010)	X									X	MC
(Shao et al., 2012)	X	X	PLNM	X		CPLEX	X	GRASP			
(Trautsmwieser and Hirsch, 2014)	X		PLNM	X	DWD+BPC	Xpress MP	X	VNS	X		
(Carollo and Lanzarone, 2014)	X	X	PLNM	X		CPLEX				X	CCA
(Mankowska et al., 2014)	X	X	PLNM	X		CPLEX	X	VNS			
(B. Yuan et al., 2015)	X	X	PLNM	X	DWD+ Branch and Bound	CPLEX	X	Glouton, VND, LA	X	X	NDSP
(En-nahli et al., 2015)	X	X	PLNM	X		CPLEX					
(Z. Yuan and Fügenschuh, 2015)		X	PLNE	X		CPLEX	X	Glouton, LS	X		
(Hiermann et al., 2015)	X	X	CSP				X	(1) RP (2) VNS, MA, SS, SA			
(Duque et al., 2015)	X		PLNM				X	LS (2 étapes)			
(Rest & Hirsch, 2016)	X	X	PLNM	X		FICO Xpress	X	TS			
(Redjem and Marcon, 2016)		X					X	CRH (2 étapes)			
(Yalçındağ et al., 2016)	X	X	PLNE	X		CPLEX	X	GA		X	KR
(Du et al., 2017)	X	X	PLNM				X	GA, LS	X		
(Nasir & Dang, 2018)	X	X	PLNM	X		CPLEX	X	VNS			
(Lin et al., 2018)	X	X					X	HSA			
(T. Zhang et al., 2018)	X	X	PLNM				X	ACO		X	NDSP
(Bazirha et al., 2019)	X	X	PLNM	X		CPLEX	X	VNS			
(Çolak et al., 2019)	X	X					X	Saving algorithm			
(Chabane et al., 2019)	X	X	CSP				X	ACO			
(Lamine et al., 2019)	X	X					X	TS			
(Fathollahi-Fard et al., 2020)		X	PLNM	X	Relaxation lagrangienne	GAMS	X	VNS-SA	X		
(Goodarzian et al., 2020)	X	X	PLNM	X	ϵ -contrainte	GAMS	X	ISEO, FFA, ABC, SEO			
(Quintanilla et al., 2020)		X	PLNM	X		GAMS	X	GRASP			
(Méndez-Fernández et al., 2020)	X		PLNE	X		Gurobi	X	SA			
(Khodabandeh et al., 2020)	X	X	PLNM	X	ϵ -contrainte	CPLEX					

(Manavizadeh et al., 2020)	X	X	PLNM	X		GAMS	X	SA				
(Zhan et al., 2021)	X	X	PLNM	X	L-shaped	CPELX		Look-one-step-back		X	X	FSSP
PLNM : Programmation Linéaire en Nombre Mixtes PLNE : Programmation Linéaire en Nombres Entiers BPC: Branch and Price Cut DWD: Dantzig Wolfe Decomposition CSP: Constraint Satisfaction Problem VNS: Variable Neighbourhood Search VND: Variable Neighbourhood Descent LS: Local Search LA: Label Algorithm CRH: Caregivers Routing Heuristic RP: Random Procedure MA: Memetic Algorithm SS: Scatter Search						SA: Simulated Annealing GA: Genetic Algorithm ISEO: Improved self-adaptive Social Engineering Optimizer FFA: FireFly Algorithm ABC: Artificial Bee Colony algorithm SEO: Social Engineering Optimizer HAS: Harmony search algorithm ACO: Ant Colony optimization CAA: Cardinality-Constrained Approach MC: Markov chain KR: Kernel Regression NDSP: Normal Distribution Stochastic Programming FSSP: Finite Support Stochastic Programming						

A la lecture de ce Tableau 4, nous pouvons constater que 70% des travaux étudiés proposent un modèle pour résoudre les deux problèmes de coordination opérationnelle que nous avons identifiées dans la section 4 du chapitre I. Concernant la formulation du problème, deux types sont utilisés :

- 75% (21/28) de ces travaux présentent une formulation du problème en PLNM ou PLNE.
- Peu de travaux formulent le problème en CSP, 7% (2/28).

Les problèmes formulés en PLNM ou PLNE sont souvent résolus par un solveur commercial (86%, 18/21), les solutions sont optimales mathématiquement, mais pour les problèmes de petite taille. Le solveur CPLEX est utilisé le plus fréquemment (61%), suivi ensuite du logiciel GAMS (36%). Trois articles (Trautsamwieser & Hirsch, 2014) (B. Yuan et al., 2015) (Fathollahi-Fard et al., 2020) utilisent une approche de décomposition ou une approche de relaxation des contraintes pour augmenter l'efficacité des calculs par des approches exactes face aux problèmes de grande taille. Deux articles (Goodarzian et al., 2020) (Khodabandeh et al., 2020) utilisent ϵ -contrainte visant à résoudre les problèmes multi-objectifs.

Du fait de la limitation du temps de calcul de la méthode exacte, 82% (23/28) des travaux appliquent une méthode approchée lorsqu'ils tentent de résoudre des problèmes de grande taille issus de cas réels. Cinq articles optent pour la combinaison des deux approches afin de diminuer le temps de calcul des modèles et d'augmenter l'exactitude des résultats.

Enfin, parmi les références étudiées, il y a six articles qui s'intéressent aux incertitudes dans leurs modèles. En fonction de notre classification des incertitudes dans la coordination opérationnelle, les travaux de (Carello & Lanzarone, 2014) et de (Yalçındağ et al., 2016) s'intéressent aux incertitudes du temps de routage des véhicules. Quant aux travaux de (B. Yuan et al., 2015) et de (Lanzarone et al., 2010), ce sont les incertitudes sur la durée des interventions qui sont visées. Pour (Zhan et al., 2021), les deux incertitudes sont prises en compte simultanément. Par ailleurs, il est important de souligner qu'il n'existe aucun article qui utilise l'approche de simulation à événements discrets pour étudier un problème de coordination opérationnelle sous incertitude en PAD, en complément d'une approche par optimisation.

3. Conclusions

Le capital des connaissances dressé par cet état de l'art montre que la prise en compte de manière exhaustive des multiples dimensions du problème de coordination dans le cadre de la PAD reste un sujet à fort potentiel en recherche.

Dans un premier temps, nous énumérons les caractéristiques de l'« offre » et de la « demande » de la coordination au niveau opérationnel dans la PAD. L'objectif est d'établir une base de connaissance du système à coordonner dans une partition « offre-demande » et ceci à un niveau très générique. Parmi les facteurs de diversité entrant dans cet exercice de structuration des connaissances, nous citerons la nature de la structure, les personnels engagés dans la réalisation des missions (administratifs, soignants, aidants, et intervenants divers), les usagers bénéficiaires, et les moyens de transport.

Ensuite, nous nous concentrons sur l'identification des caractéristiques des exigences de coordination opérationnelle de la PAD. Il s'agit d'identifier et de structurer les exigences opérationnelles des multiples problèmes posés par la définition d'une trajectoire organisationnelle pour le pilotage de la PAD, et ceci afin de cerner l'espace des différentes prises de décision qui présideront à la coordination. En effet, beaucoup d'auteurs se sont intéressés à des problèmes couvrant partiellement le besoin de coordination. Le routage des intervenants, l'affectation des intervenants aux demandes de PAD des bénéficiaires, sur des échelles de temps opérationnelles variables (jour, semaine), sont autant d'illustrations de ce à quoi peuvent être confrontés des acteurs qui ont la volonté de se coordonner. Notons que le bon fonctionnement d'une PAD est basé sur la bonne coordination de ses activités, qui dans la vraie vie, reste difficile à assurer en raison de nombre de contraintes métiers existantes et souvent propres à chaque structure de PAD.

Enfin, nous effectuons un bilan des travaux antérieurs sur l'aide à la décision pour la coordination de PAD. Dans cette partie, nous nous intéressons à l'organisation du pilotage et à la mécanique de la prise de décisions. Dans la littérature, ce sont souvent des méthodes d'optimisation qui sont appliquées pour résoudre les problèmes formulés. La coordination se concrétise par des plans qui sont les consignes pour le pilotage. Appréhendé de manière globale, le jugement de valeur porté sur l'organisation du pilotage passe par une critique de la qualité des décisions prises et par une estimation des propriétés du système en fonctionnement. La prise en compte des incertitudes, par exemple, est un facteur important dans la PAD, du fait de la nature même des activités de soins qui sont à prodiguer.

CHAPITRE III. POSITIONNEMENT, QUESTIONS DE RECHERCHE ET JUSTIFICATION DU SUJET D'ETUDE

CHAPITRE III. POSITIONNEMENT, QUESTIONS DE RECHERCHE ET JUSTIFICATION DU SUJET D'ETUDE	49
1. POSITIONNEMENT DE TRAVAUX DE RECHERCHE.....	50
1.1. Notion de modèle et de méta-modèle	51
1.1.1. Modèle	51
1.1.2. Méta-modèle.....	52
1.2. Notion du langage de modélisation spécifique à un domaine (DSML)	53
1.2.1. Méthode de modélisation	55
1.3. Principe de l'Ingénierie Dirigée par les Modèles (IDM)	57
1.3.1. Architecture dirigée par les modèles.....	58
1.3.2. Chainage de modèles dans l'architecture dirigée par les modèles (CIM-PIM-PSM)	59
1.3.3. Transformation semi-automatisée pour le chainage de modèles	60
1.3.3.1. Catégories de la transformation des modèles	61
1.3.4. Aperçu de quelques environnements de méta-modélisation	62
1.3.4.1. Plateforme de méta modélisation sous Eclipse (EMF, GMF, GEF)	62
1.3.4.2. GME (Generic Modeling Environment)	63
1.3.4.3. ADOxx	63
2. CONCLUSION ET APERÇU DES QUESTIONS DE RECHERCHE	64

1. Positionnement de travaux de recherche

La formulation des problèmes de coordination de PAD fait partie des attendus de cette thèse. Elle repose d'abord sur le capital actuel des connaissances sur le sujet, tel que traité dans les sections 2.1 et 2.2 du chapitre II. Nous avons dressé le constat d'un système riche composé de multiples caractéristiques humaines, individuelles et collectives, territoriales et temporelles au chapitre I. L'éditeur qui soutient cette recherche veut pouvoir couvrir le plus grand nombre de structures. En conséquence, nous cherchons à capturer, de manière exhaustive et aussi formelle que possible, toutes ces connaissances utiles pour traiter ce problème de coordination opérationnelle de PAD. Mais nous voulons aussi l'utiliser de manière souple et toujours avec cohérence.

Cette position fonde notre première question de recherche :

Q1 : Comment mobiliser les connaissances organisationnelles et managériales pour planifier et coordonner les interventions à domicile, et évaluer la qualité de la prise en charge en prenant en compte la diversité des structures PAD ?

Par notre culture scientifique, mais aussi pour des raisons d'efficacité face à des formes de complexité, nous privilégions le recours à la modélisation. La modélisation est une clé pour aborder ces diversités, permettant d'établir une base à un niveau très générique composée de différents artefacts forgeant la description d'une PAD, et de s'armer pour affronter la complexité des problèmes de coordination sous-jacents par l'usage de modèles de processus métier.

Rappelons au passage que la modélisation est définie par (Le Moigne, 1990) comme « l'action d'élaboration et de construction intentionnelle, par composition de symboles, de modèles susceptibles de rendre intelligible un phénomène perçu complexe, et d'amplifier le raisonnement de l'acteur projetant une intervention délibérée au sein du phénomène ». Nous souhaitons modéliser la structure et le fonctionnement d'un système distribué qui est une organisation. Sur cette base et selon (Vernadat, 1999), la modélisation d'entreprise (ou en entreprise) désigne la construction de modèles d'une partie déterminée d'une entreprise pour en expliquer la structure et le fonctionnement ou pour en analyser le comportement. Selon (Bernus et al., 1994), elle consiste à décrire l'organisation et les processus opérationnels d'une entreprise, soit dans le but de simuler ces processus pour comparer divers scénarios, soit dans le but de les analyser et de les restructurer pour améliorer les performances de l'entreprise. Dans un cas comme dans l'autre, l'objectif reste d'obtenir une meilleure compréhension des comportements de l'entreprise, de communiquer de manière plus complète et plus pertinente entre les acteurs, d'améliorer les performances (qualitatives et quantitatives), de devenir plus flexible et plus réactif, d'obtenir une meilleure décision (Lamine, 2001). La représentation par les processus métier (aussi dénommée BPM) est incluse dans cette théorie, les processus métier constituant une clé efficace pour conceptualiser l'organisation du travail.

Une autre définition bien en phase avec notre problème de diversité des connaissances véhiculées par des processus de coordination de PAD, est donnée par (Karagiannis, 2015). Il considère la modélisation d'entreprise comme un cadre d'ingénierie qui fournit des moyens de représentation des connaissances relatives à des objectifs déterminés par les parties prenantes. Cette vision est largement partagée par la communauté scientifique en ingénierie des systèmes qui considère la modélisation comme une discipline scientifique ayant pour but de développer

et de promouvoir des méthodes, des techniques, des modèles et des outils permettant de maîtriser le comportement d'un système (y compris socio-technique) dans le temps. Elle permet donc de capturer différentes facettes, par exemple, les processus, les produits, les ressources et le contexte métier (Karagiannis et al., 2016), à travers des modèles faisant sens pour leur lecteur.

Mais qu'est-ce qu'un tel modèle ? Cette couche abstraite qui permet de mieux comprendre un système complexe par l'agencement de ses composantes ? Nous parcourons la littérature pour appréhender les notions de « modèle » et de « méta-modèle ».

1.1. Notion de modèle et de méta-modèle

1.1.1. Modèle

Le modèle est une représentation abstraite de la réalité, fonction des objectifs visés, et exprimé à l'aide d'un langage de modélisation. Il est l'artefact (le résultat) de la modélisation. Selon (Calvez, 1990), il s'agit d'une interprétation explicite par son utilisateur de la compréhension d'une situation ou, plus simplement, d'une idée qu'il se fait sur la situation. Il permet d'aborder le monde de manière simplifiée, en minimisant les effets de la complexité, le danger et l'irréversibilité de la réalité (Rothenberg et al., 1989).

Dans (Thabet, 2020), plusieurs définitions ont été retenues pour renforcer la description du modèle :

- Un ensemble de déclarations sur le système à l'étude (Seidewitz, 2003).
- Une abstraction d'un système permettant de faire des prédictions ou des inférences (Kühne, 2006),
- Une simplification d'un système construit avec un objectif prévu à l'esprit de sorte qu'un modèle devrait être capable de répondre à des questions à la place du système d'origine (Bézivin and Gerbé, 2001).

Toutefois, la réalité est clairement plus complexe que n'importe quel modèle, indépendamment de son degré de sophistication. Le modélisateur doit déterminer quels aspects sont pertinents et lesquels ne le sont pas, en fonction de l'objectif à atteindre. En cela, modéliser est un art. La production d'un modèle est la projection d'une perception de la réalité par le modélisateur. Il est fréquent que lors du développement d'un modèle, le modélisateur prend conscience de nouvelles informations auxquelles il n'avait jamais porté attention et prend des décisions sur ce qui doit être représenté ou pas (Schon, 1984).

Ainsi en modélisation d'entreprise, le modèle est une voie pour extérioriser, construire et partager des connaissances sur l'entreprise (Fischer, 2005). Il constitue un moyen de partage et d'échange si le langage de modélisation est connu des parties. La communication et le partage des connaissances entre les différents acteurs inscrits dans un système avec un contexte métier est donc facilitée (Bortolaso, 2012). Nous trouvons là un écho à notre définition de la coordination en PAD. Il va se pratiquer par l'analyse d'un ensemble de modèles décrivant divers aspects d'une entreprise, on parle de l'examen des points de vue (tâches à réaliser, informations à échanger, processus métier, organisation, disposition des machines, etc.) qui permettent de bâtir une vision holistique de son fonctionnement quand on les relie.

1.1.2. Méta-modèle

Un méta-modèle est en quelque sorte le moule du modèle. C'est une autre abstraction, plus élevée, qui met en évidence les propriétés du modèle lui-même. Si on considérait le modèle comme un objet, le méta-modèle serait son moule. Le méta-modèle précise les concepts qui deviendront les primitives du langage de modélisation (Fill and Karagiannis, 2013). Selon (Karagiannis, 2015), le méta-modèle est comme une boîte terminologique résultant d'un effort itératif d'acquisition de connaissances. Selon (Benaben, 2012), le métamodèle représente le langage utilisé pour exprimer un modèle, une partie de l'univers du discours. Cette interprétation reste en phase avec celle donnée par (Chapron, 2006) et par laquelle il considère un méta-modèle comme : « une description de tous les concepts d'un langage ». Il découle de cette perception qu'un méta-modèle permet de (James et al., 2013) (Karagiannis et al., 2016):

- Définir explicitement les concepts constituant un langage de modélisation spécifique, et donc son pouvoir d'expression.
- Exploiter l'extensibilité du langage.
- Fournir les moyens de créer des palettes de modélisation graphique
- Valider un modèle.
- Valoriser des usages pour lesquels un modèle est prédestiné.
- Améliorer l'efficacité de la gestion de modèles.
- Mettre à disposition un format d'échange entre systèmes.

Les méta-modèles sont ainsi des moyens de production des modèles. Grâce aux meilleures pratiques et à l'expérience acquise lors de la conceptualisation au moyen, pour nous, de la modélisation d'entreprise, les méta-modèles soutiennent les experts du domaine pour comprendre un contexte métier donné par des capacités syntaxiques et sémantiques (Favre, 2005).

La Figure 9 illustre la relation entre modèle et méta-modèle (Thabet, 2020). Un modèle est conforme à un méta-modèle, ce qui signifie que les modèles doivent respecter les règles définies au niveau du métamodèle (Paige et al., 2014). Plus classiquement, le métamodèle est un modèle de modèles, définissant un langage, spécifique à un domaine applicatif ou plus général, pour créer des modèles conformes au métamodèle (OMG, 2015).

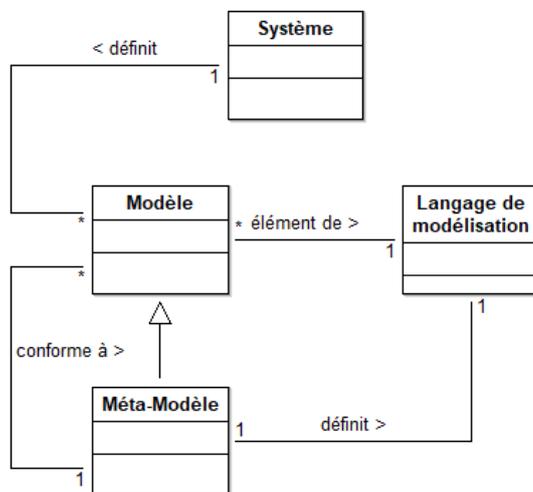


Figure 9 : Relations entre la réalité, le modèle, le méta-modèle et le langage de modélisation (Thabet, 2020)

La capitalisation de connaissances sur le domaine HHCRSP peut être réalisée par un langage de spécifique au domaine (DSL), ou plus précisément, par un langage de modélisation spécifique au domaine (DSML). Toute variante du problème HHCRSP connue ou à venir devrait pouvoir être exprimée par une instance du méta-modèle définissant ce DSML. La syntaxe de ce langage donnera naissance à une structure de données permettant d'exporter ou d'importer des données sur une PAD, puis de permettre d'utiliser ces données par un coordinateur afin de réaliser ses objectifs de représentation, et ensuite ses volontés de prise de décision. Cette faculté offre aussi des ouvertures pour assurer l'interopération entre les modèles (import/export) et faciliter la réutilisation.

Ainsi, nous posons une question de recherche qui affine la question Q1 :

Q1-1 : Comment définir un langage de modélisation spécifique au domaine (DSML) permettant de représenter des connaissances qui peuvent peser sur le bon fonctionnement de la coordination dans la PAD, et dans toute sa variété ?

Nous illustrons par la suite la notion de DSML à travers une revue de la littérature.

1.2. Notion du langage de modélisation spécifique à un domaine (DSML)

Dans la modélisation d'entreprise, l'utilisation de DSML a tendance à croître. C'est une voie différente de la voie classique mobilisant des langages de modélisation d'entreprise. (Doumeingts, 1990) définit le langage de modélisation comme « une représentation de la connaissance à communiquer sans ambiguïté. Il permet de construire des modèles selon des composants associés ». Au-delà de cette définition classique, l'idée d'utiliser des langages de modélisation conçus pour des objectifs plus spécifiques, appelés langages de modélisation spécifiques au domaine (DSML), a remporté une popularité croissante (Frank, 2013). Pour clarification, le DSML est destiné à représenter les concepts clés du domaine spécifique et seulement de lui. Il formalise la structure, le comportement et les exigences de la demande dans ces domaines particuliers. Les familiers connaisseurs de ce domaine sont dans leur univers de discours et dans leurs schémas cognitifs (Rivera, 2010). Les utilisateurs potentiels d'un DSML incluent donc les experts du domaine, les analystes de systèmes et les développeurs de logiciels (Frank, 2013). Dans notre contexte, il correspond à l'expert métier de PAD avec, au tout premier rang, le coordinateur d'une structure de PAD.

La conception d'un DSML aborde trois formes de contenu : les concepts du domaine et les règles afférentes (syntaxe abstraite); la notation utilisée pour représenter ces concepts, qu'elle soit textuelle ou graphique (syntaxe concrète); et la sémantique du langage (Thabet, 2020) (cf. la Figure 11):

- Syntaxe abstraite : elle est généralement spécifiée par un métamodèle, qui est lui-même un modèle et décrit les concepts du langage. Les relations entre concepts (vocabulaire) et les règles de structuration (grammaire) contraignent les éléments du modèle, ainsi que leurs combinaisons afin de respecter ces règles. La pertinence des concepts dépend fortement du niveau d'abstraction (Greenfield and Short, 2003).
- Syntaxe concrète : elle fournit une réalisation de la syntaxe abstraite sous forme de concordance entre un concept du métamodèle et sa représentation textuelle ou graphique. Pour des langages visuels, il est nécessaire d'établir une correspondance sans

équivoque entre concepts et symboles visuels permettant de les représenter (cf. la Figure 10, le triangle du sens (Sienou, 2009)). De même que pour les langages textuels, des liens sont nécessaires entre les éléments du métamodèle (mots, ponctuation) et les structures syntaxiques du DSML textuel (phrase) (Rivera, 2010).

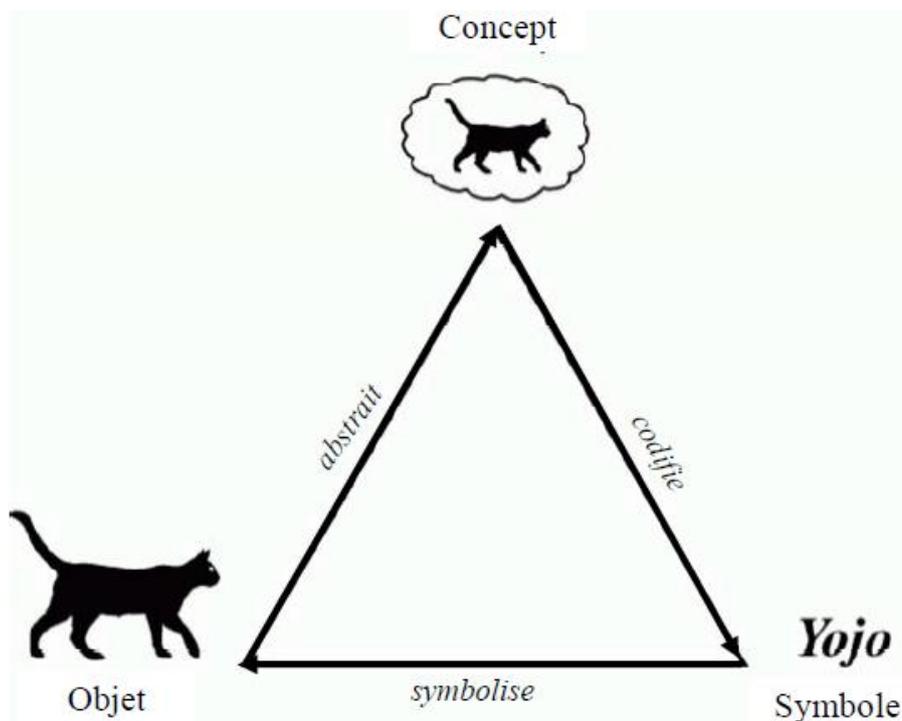


Figure 10 : Triangle du sens (Sienou, 2009)

- Sémantique : la sémantique d'un DSML est généralement donnée avec le langage naturel, elle se compose des mots clés ou des symboles spéciaux dont la signification est analogue à une norme acceptée par des humains. Néanmoins, même si les humains peuvent naturellement comprendre la signification de la plupart des termes d'un DSML dans leur langage naturel, une machine ne peut pas agir sur de telles hypothèses. Par ailleurs, dans la modélisation d'entreprise où les modèles sont des artefacts clés à partir desquels l'ensemble du système peut être dérivé et implémenté, une base précise et formelle nous aidera considérablement à les définir et à ne pas produire de résultats inattendus, et encore à être en mesure d'effectuer une analyse formelle sur les modèles. Par conséquent, il existe également des approches formelles pour définir la sémantique du DSML. Un exemple est l'approche de la sémantique translationnelle, qui implique la définition de la sémantique d'un DSML par la définition d'un mapping entre le DSML lui-même et un autre langage disposant d'une sémantique clairement définie, comme les Machines d'Etat Abstraites (Börger, 2010), les Réseaux de Petri (Peterson, 1977), par exemple (Rivera, 2010).

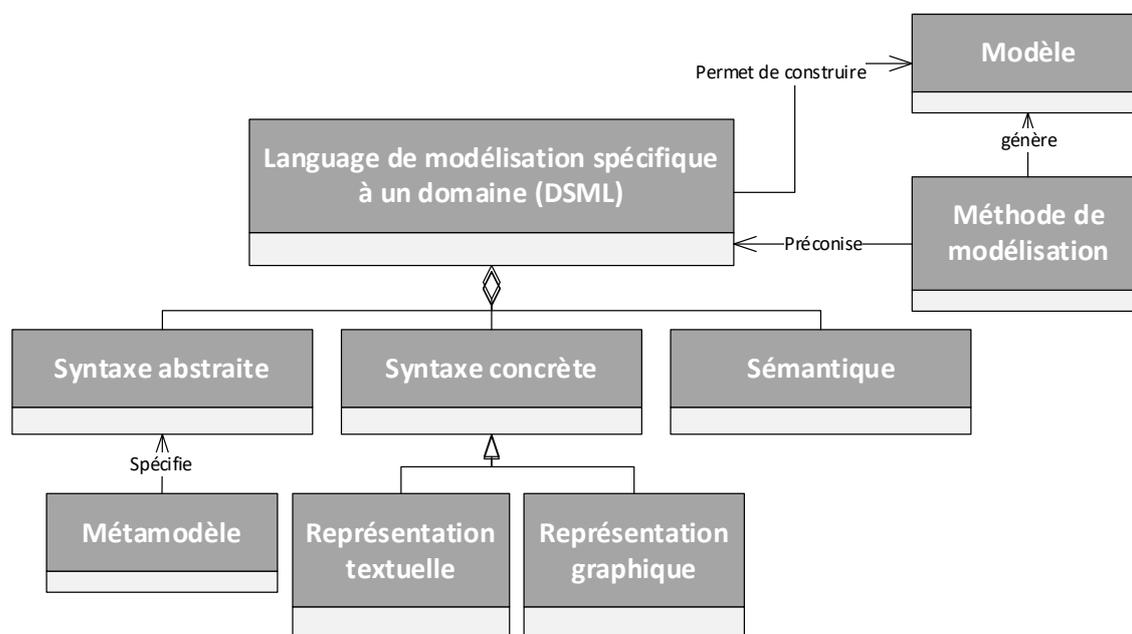


Figure 11 : Définition d'un DSML adapté de (Thabet, 2020) et relation entre la modélisation, le DSML et la méthode de modélisation

1.2.1. Méthode de modélisation

Une méthode de modélisation est une technique de construction d'un modèle dirigée par un ensemble d'étapes bien définies qui permettent de guider, voire de simplifier, le travail du modélisateur, et met à sa disposition un ou plusieurs langages de modélisation. Par exemple, en système d'information, le langage UML 2.0 met à disposition 14 diagrammes différents.

Dans (Amokrane, 2016), une série de méthodes de modélisation est évoquée suivant un ordre chronologique. Nous citons, parmi les méthodes les plus utilisées, les méthodes de modélisation fonctionnelle qui permettent une clarification et une décomposition analytique de la complexité d'un système : SADT (*Structured Analysis and Design Technique*) (Marca and McGowan, 1987), SSADM (*Structured System Analysis Design Method*) (Ashworth, 1988). Les méthodes de modélisation informationnelle qui permettent de définir une structure de données ou d'informations pouvant ainsi être mises en œuvre dans une base de données, comme Merise.

Au fil du temps, les méthodes de modélisation à fonction unique n'ont plus répondu aux besoins de la modélisation. D'où la naissance de cadres de modélisation, souvent accompagnés d'un langage de modélisation et d'une méthodologie, qui couvrent un large spectre d'utilisation allant de la modélisation fonctionnelle et informationnelle, jusqu'à la modélisation décisionnelle en passant par la modélisation comportementale et organisationnelle, et ceci pour la structuration de connaissances d'un domaine spécifique. Il est mis de l'ordre dans cette pluralité de modélisation grâce à une architecture d'entreprise. Une telle architecture est « une structure de modèles à utilité partielle reposant sur des dimensions complémentaires, au sein de laquelle on chemine afin de comprendre le tout et dans l'espoir d'être en capacité de répondre à des questions » (Pingaud, 2005). Parmi les architectures les plus fréquemment utilisées, nous pouvons citer, sans être exhaustif : PERA (*Purdue Enterprise Reference Architecture*) (Williams, 1994), la famille IDEF (*Integration DEFinition*) (Presley and Liles, 1995), CIMOSA (*Computer Integrated Manufacturing Open System Architecture*) (Zelm et al., 1995), GRAI/GIM (*Graphe de Résultats et Activités Interreliées/ GRAI Integrated Methodology*) (Chen et al., 1997),

GERAM(*Generalised Enterprise Reference Architecture and Methodology*) (Bernus and Nemes, 1997), ACNOS (Mhamedi and Vernadat, 1999), ARIS (Scheer, 2000), TOGAF (Desfray and Raymond, 2019) (TOGAF, 2021).

D'une manière générale et comme illustrée dans la Figure 12 proposée par (Karagiannis and Kühn, 2002), l'architecture pose un cadre de modélisation qui doit posséder trois parties : (1) un « langage de modélisation » qui définit l'ensemble des construits du langage. (2) La « procédure de modélisation » qui permet de définir les étapes à suivre par les acteurs pour atteindre leur objectif. (3) Les « mécanismes et algorithmes » qui couvrent les fonctionnalités destinées à manipuler le contenu des modèles face à des objectifs multiples tels que la simulation, la visualisation, la transformation et l'évaluation des modèles (Karagiannis et al., 2016).

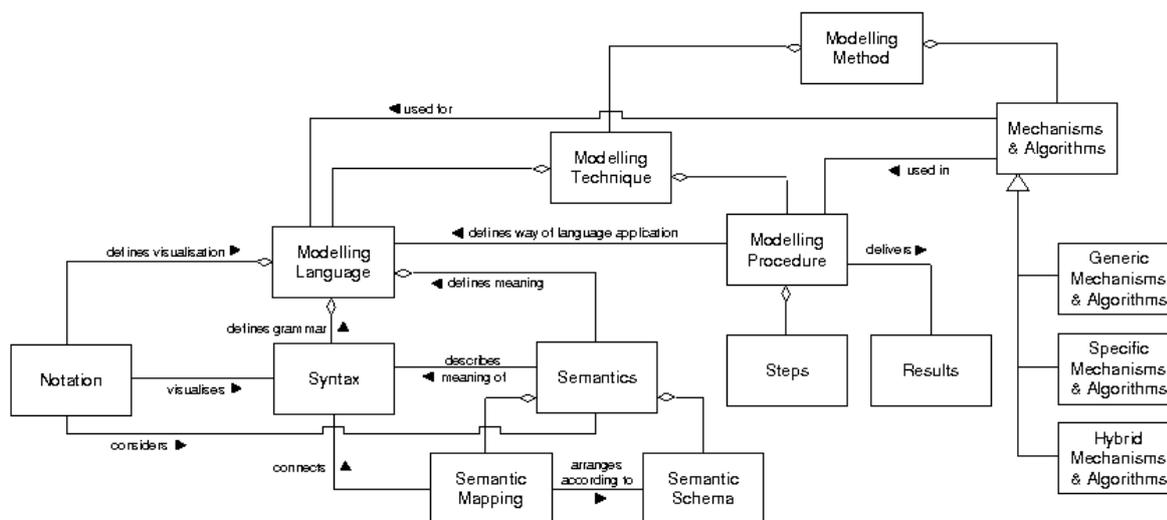


Figure 12 : Composants de méthode de modélisation d'OMILAB (Karagiannis and Kühn, 2002)

Notre objectif reste d'aider au pilotage des interventions à domicile de professionnels sociaux, médicaux, médico-sociaux et paramédicaux autour du patient bénéficiaire, pour organiser collectivement leur quotidien. En cela, la fonction de coordination opérationnelle doit permettre de remplir des objectifs de performance que nous avons déjà discutés. Fidèle à notre logique systémique, nous voulons développer l'unification des concepts caractérisant la coordination de PAD par un DSML, dans un univers de discours où le vocabulaire est emprunté au monde professionnel concerné. Ce langage doit permettre au coordinateur de formuler son besoin. Il s'agira ensuite de traduire cette expression du besoin en un modèle mathématique de prise de décisions pour bâtir des plannings prévisionnels, puis le cas échéant, les adapter lors du pilotage de la structure. Ainsi, les fondements de cette thèse reposent sur la recherche d'une solution d'un problème de coordination justement formulé à partir d'une connaissance structurée par la rencontre de l'offre et de la demande.

Nous en déduisons notre deuxième question de recherche :

Q2 : Comment formuler mathématiquement des problèmes de coordination à partir des connaissances métiers ?

En référence à la section 2.3 du chapitre II, nous savons qu'il existe de nombreuses méthodes de résolution du problème formulé. Afin de pouvoir concilier la fidélité de la solution de coordination (la qualité des plannings en termes de performance) et les impératifs des calculs

(temps de calcul et convergence des méthodes), nous dissocions la question Q2 en deux questions :

Q2-1 : Comment enrichir l'espace de formulation de problèmes de coordination avec des formulations mathématiques variées et dans des langages adaptés ?

Q2-2 : Comment orienter ces formulations pour être compatible avec les outils de calcul de solution, puis publier ces solutions et les agrémenter d'indicateurs permettant de juger leurs qualités ?

Ces deux questions constitutives de Q2 nous conduisent à une conception générale de notre outil numérique. Elles nous placent en quelque sorte face à un système à définir qui posséderait des conditions limites : la bonne traduction des besoins en condition initiale à l'aide d'un langage puissant, la génération efficace des plannings en condition finale avec un solveur idoine. C'est notre responsabilité de chercheur de relier le début à la fin. En effet, nous cherchons à capturer, de manière exhaustive et aussi formelle que possible, toutes les connaissances utiles pour traiter ce problème de coordination opérationnelle de PAD dans toute ses diversités. Puis via une variété de formulation mathématique avec leurs méthodes de résolution adaptées, nous espérons trouver des solutions de coordination au niveau opérationnel. Ces deux phases seront vues comme des étapes d'une « chaîne de production » de modèles dont la finalité est d'aider les coordinateurs des structures de PAD dans leurs prises de décision, et ceci afin d'assurer un bon équilibre dans la rencontre entre l'offre et la demande de PAD.

Mais comment se réalise une telle production ? Est-ce réaliste de prendre cette posture ? Comment conjuguer une relation subtile entre les connaissances métiers et la capacité à décider ? Décider quoi et quand, en plus du comment ? C'est en ayant recours aux théories de l'Ingénierie Dirigée par les Modèles (IDM) que nous empruntons cette piste de recherche. L'IDM a développé un corpus disciplinaire qui permet de concevoir cette chaîne en travaillant une logique d'interactions entre des modèles à finalités différentes (depuis le modèle du besoin jusqu'au modèle de calcul).

Dans le but d'imaginer une telle démarche de recherche de solution en coordination de PAD, nous posons notre dernière question de recherche Q3 :

Q3 : Comment définir une série de règles de transformation de modèles à partir d'un langage de modélisation spécifique au domaine (DSML) permettant de supporter un processus dématérialisé allant des connaissances orientées métiers vers les connaissances mathématiques requises pour traiter les problèmes de coordination, et les formuler de manière systématique ?

Nous souhaitons par la suite fournir une base théorique et méthodique solide sur les concepts clés, et la terminologie associée, qui fondent l'IDM.

1.3. Principe de l'Ingénierie Dirigée par les Modèles (IDM)

L'Ingénierie Dirigée par les Modèles (IDM) est « une pratique d'ingénierie des systèmes utilisant les capacités des technologies informatiques pour décrire à travers des modèles, concepts, et langages, à la fois le problème posé (besoin) et sa solution » (Schmidt, 2006).

Cette pratique fournit une approche en vue de dériver la valeur des modèles et de l'architecture, à l'appui du cycle de vie complet des systèmes physiques, organisationnels et informatiques incluant les conceptions à tous les niveaux, comme une entreprise entière, un processus, des structures d'information (OMG, 2014). Le focus est mis sur la création et l'exploitation de modèles de domaines, qui sont des modèles conceptuels. Ainsi, la pratique met en évidence et utilise des représentations abstraites de connaissances et des activités qui régissent un domaine d'application particulier (Rossi et al., 2016). Citons une application sur l'alignement de l'organisation dans la conception de systèmes d'information (Avila and Goepp et al., 2018).

Au cœur des processus d'IDM, il y a des modèles qui subissent des transformations (Khecharem, 2016). Le modèle n'est pas que descriptif ou normatif, il est fait place à d'autres utilités. Il prend un rôle encore plus prescriptif en matière de performance et même opérant (il devient exécutable) ce qui est une vraie innovation disruptive dans la communauté informatique. Les modèles y sont considérés comme des artefacts techniques peuplant le cycle de vie de l'ingénierie logicielle. Ce sont ces modèles qui prennent un caractère opérationnel quand ils permettent d'élaborer et de contrôler les codes informatiques produits automatiquement, ainsi que les appels de services numériques réalisés intelligemment.

Le bénéfice escompté dans l'utilisation de l'IDM au sein du cycle de vie de développement logiciel vise « à accroître la productivité et à réduire le temps de mise sur le marché en permettant un développement à un niveau d'abstraction plus élevé et en utilisant des concepts plus proches des problèmes à résoudre, plutôt que ceux offerts par les langages de programmation » (Sendall and Kozaczynski, 2003). À l'aide des modèles au sein de l'IDM, nous sommes en mesure de mieux gérer la complexité des systèmes de grande taille, ainsi que l'interaction et la collaboration entre les organisations, les personnes, les matériels et les logiciels (OMG, 2014).

1.3.1. Architecture dirigée par les modèles

Le consortium américain OMG (*Object Management Group*) a popularisé la pratique de l'IDM. Les membres de l'OMG ont publié leur vision de l'architecture dirigée par les modèles, en anglais Model Driven Architecture (MDA) comme un standard, à la fin de l'année 2001 (Poole, 2001). Cette architecture clarifie la filiation entre modèle, métamodèle et méta-métamodèle. Le MDA est illustré par la fameuse pyramide schématisée par la Figure 13.

La couche M0 représente les éléments qui existent réellement dans le contexte d'un environnement informatique ou dans le monde réel relatif au sujet à traiter. La couche au-dessus de M0 est la première abstraction de la réalité par les modèles (M1), c'est une simplification du sujet dont le but est de répondre aux questions posées en rapport avec ce sujet (cf. nos questions de recherche Q1 à Q3). Dans la littérature, il y a une foison d'exemples, d'un modèle UML (*Unified Modeling Language*) (Muller and Gaertner, 2000), d'une partie d'une application (un modèle de système logiciel inclut un diagramme de classe UML, des diagrammes de séquence UML et des images de l'interface homme-machine), d'un processus dématérialisé en BPMN (*Business process model and notation*) (Allweyer, 2016) ou d'un modèle de simulation. Les modèles du M1 sont conformes à un méta-modèle qui se situe dans la couche M2. Ce méta-modèle fixe ce qui peut être exprimé dans des modèles valides du M1 (cf. la Figure 9). De même, il précise la syntaxe abstraite et la sémantique d'un langage de modélisation. Au sommet de cette hiérarchie

se trouve la couche de méta-modélisation (M3) qui est principalement chargée de fournir un langage pour spécifier les méta-modèles (Thabet, 2020).

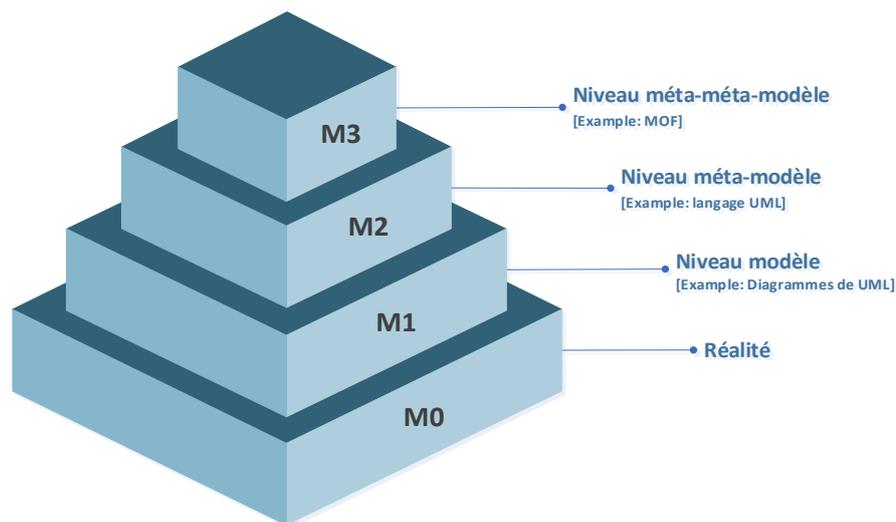


Figure 13 : Pyramide à quatre niveaux du MDA selon l'OMG (OMG, 2014)

1.3.2. Chainage de modèles dans l'architecture dirigée par les modèles (CIM-PIM-PSM)

Sous l'angle d'un cycle de vie, l'architecture dirigée par les modèles spécifie trois phases, par défaut : (1) une phase où le point de vue est indépendant de tout développement technique en numérique (traduire l'expression des besoins dite *Computation Independent*, CI), (2) une phase où le développement numérique a débuté, mais qui est indépendante de toute plateforme de développement (proposer une spécification de solution dite *Platform Independent*, PI) et (3) une phase où le point de vue sera spécifique de la plateforme de développement (coder le service numérique, *Platform specific*, PS) (Truyen, 2006) :

- Le point de vue CI se focalise sur le contexte et les exigences du système sans tenir compte de sa structure et des traitements qu'il réalise.
- Le point de vue PI se concentre sur les capacités opérationnelles d'un système en dehors du contexte d'une plate-forme¹ spécifique en montrant uniquement les parties d'une spécification complète qui peuvent être extraites de cette plate-forme.
- Le point de vue PS enrichit le point de vue PI avec des détails relatifs à l'utilisation d'une plateforme spécifique.

En relation directe avec cette première structuration, trois modèles d'un système sont positionnés, un par phase (Truyen, 2006) (Thabet, 2020) :

- Le CIM (*Computation Independent Model*) est fréquemment appelé modèle d'entreprise ou de domaine puisqu'il utilise des termes familiers aux experts du domaine métier. Ce modèle présente ce que le système a prévu de faire, mais masque toutes les spécifications liées aux technologies informatiques afin de rester indépendant de la manière dont ce

¹ Un ensemble de sous-systèmes et de technologies assurant un ensemble de fonctionnalités cohérent à travers des interfaces et des patrons d'utilisation

système sera mis en œuvre. Le CIM contribue de manière importante à combler le fossé existant entre les experts métiers et les informaticiens en charge de la mise en œuvre du système.

- Le PIM (*Platform Independent Model*) présente un degré d'indépendance suffisant pour permettre sa mise en correspondance avec une ou plusieurs plateformes. Pour y parvenir, la définition d'un ensemble de services et de structures se fait généralement de manière abstraite, sans tenir compte des détails techniques.
- Le PSM (*Platform Specific Model*) combine les spécifications du PIM avec les détails nécessaires pour stipuler de quelle manière un système utilise un type de plateforme spécifique (supposant parfois qu'on dispose aussi d'une modélisation de la plate-forme, PDM). Il sert à la génération du code exécutable.

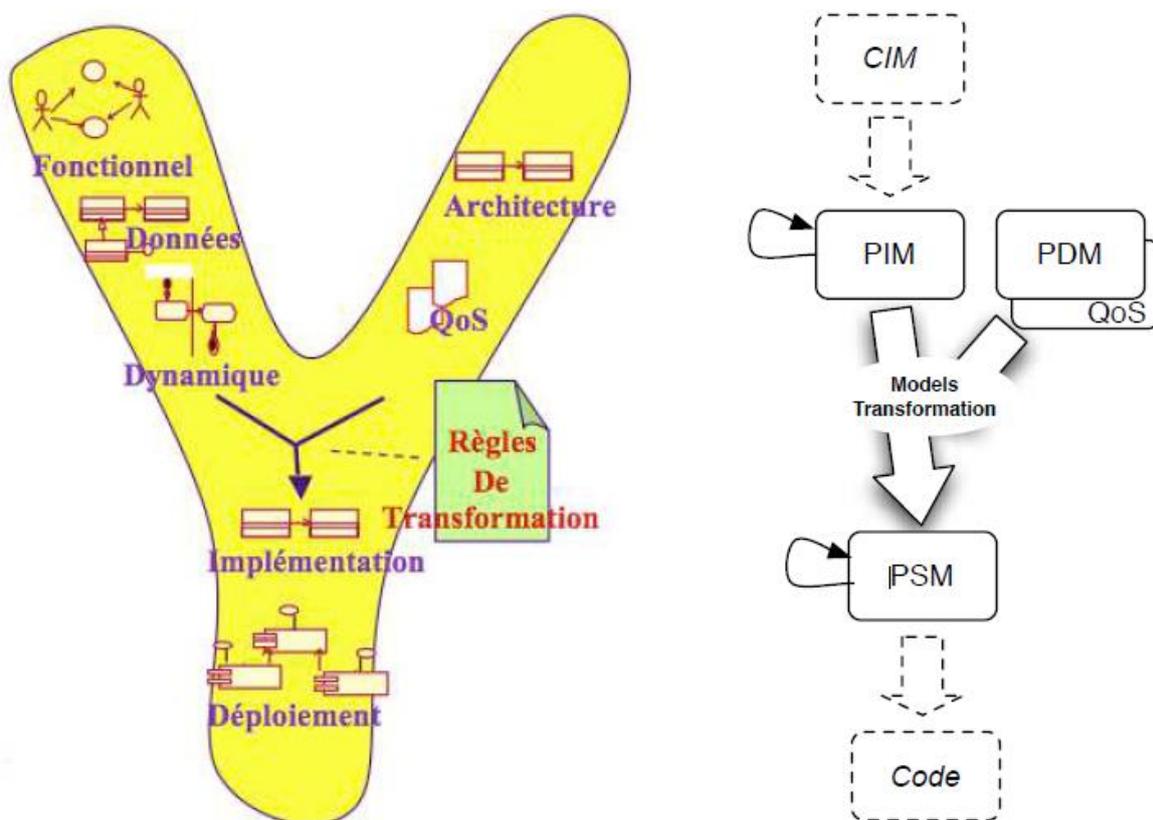


Figure 14 : Architecture dirigée par les modèles, la logique de transformation sous forme d'un Y (Combemale, 2008)

La Figure 14 illustre une manière d'enchaîner les phases (partie droite) sous forme d'un « Y ». Les modèles sont transformés en allant du haut vers le bas de ce « Y ». La partie gauche est un exemple des connaissances véhiculées, illustrées par des pictogrammes résumant des diagrammes UML 2.0.

La transformation de modèles qui est ici décrite par des règles positionnées entre PIM et PSM, va être élargie dans notre étude à toutes les phases.

1.3.3. Transformation semi-automatisée pour le chaînage de modèles

La transformation des modèles est un aspect clé au sein de l'IDM. L'idée est donc de partir d'un modèle métier et indépendant de la technique (CIM), pour arriver à générer un autre modèle

indépendant d'une plateforme de développement (PIM), puis de générer à la suite un modèle spécifique à une plateforme (PSM). Nous ne remettons pas en cause le processus défini par (Kleppe et al., 2003) qui vise à « la génération automatique d'un modèle cible à partir d'un modèle source, selon une description des transformations par des règles ». Enfin, le modèle PSM est souvent traduit directement dans un langage informatique exécutable par la plateforme spécifique, comme par exemple Java, .NET, SQL.

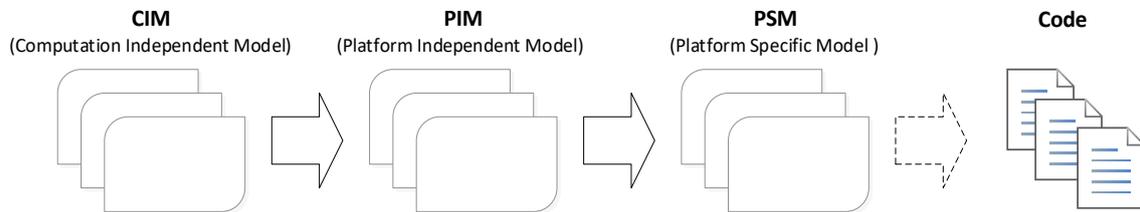


Figure 15 : Architecture dirigée par les modèles : CIM-PIM-PSM-Code

1.3.3.1. Catégories de la transformation des modèles

La transformation entre les modèles sert toujours des intérêts particuliers. A cet égard, une classification est proposée dans (Kahani and Cordy, 2015) qui distingue deux catégories en fonction du résultat obtenu : (1) la transformation de modèle à modèle, (2) la transformation de modèle à code.

(1) Transformation de modèle à modèle (*Model-to-Model*, M2M)

La transformation M2M consiste à transformer un modèle source en un modèle cible. Si le modèle source et le modèle cible sont conformes au même méta-modèle, il s'agit d'une « **transformation endogène** » et sinon il s'agit d'une « **transformation exogène** » (Nastov, 2016). Conformément à la définition de (Czarnecki and Helsen, 2003), il existe cinq approches de transformations M2M :

- **Manipulations directes** qui offrent une représentation des modèles internes, plus une API (*Application Programming Interface*) pour la manipuler. En général, elles sont mises en œuvre dans un cadre logiciel (Framework) orienté objet, ce dernier fournit une infrastructure minimale pour organiser les transformations.
- **Transformations relationnelles** qui regroupent les approches déclaratives par des relations mathématiques pour décrire les règles de mise en correspondance (*Mapping rules*) entre les métamodèles. L'idée de base est de déclarer le type d'élément source et cible de la relation, la spécification de ces relations est exprimée à l'aide de contraintes.
- **Transformations basées sur les graphes** qui sont similaires aux approches relationnelles, elles sont capables d'exprimer les règles de mise en correspondance de manière déclarative, mais ici fondées sur des graphes typés, attribués et étiquetés.
- **Transformations dirigées par la structure** qui comportent deux phases distinctes : la première phase concerne la création de la structure hiérarchique du modèle cible, tandis que la seconde phase fixe les attributs et les références dans la cible. Le cadre logiciel global détermine la stratégie de programmation et d'application, les usagers se préoccupent uniquement de fournir les règles de transformation.

- **Approches hybrides** qui combinent différentes techniques des catégories précédentes. Ces approches ont donné naissance à des langages de transformation, par exemple ATL (*ATLAS Transformation Language*) (Jouault et al., 2006).

(2) Transformation de modèle à code (*Model-to-code*, M2C)

La transformation M2C est considérée comme un cas particulier de la transformation M2M, connue sous le terme de « la génération de code ou de texte » (Nastov, 2016). Cette transformation se fait entre les modèles sources conformes à leur métamodèle, d'une part, et le méta modèle cible qui est souvent celui du langage de programmation cible, d'autre part. Il existe deux types de transformations M2C (Czarnecki and Helsen, 2003) :

- **Transformation basée sur les visiteurs** qui consiste à fournir des mécanismes de visiteurs pour parcourir la représentation interne d'un modèle et écrire les codes dans un flux de texte.
- **Transformation basée sur des patterns**, les patterns se composent du texte cible contenant des fragments de méta code, ce qui permet d'accéder aux informations de la source et d'effectuer une sélection de code et une expansion itérative.

La Figure 16 résume la transformation des modèles comme une forme de structure de contrôle du processus transformationnel au sein de l'architecture dirigée par les modèles.

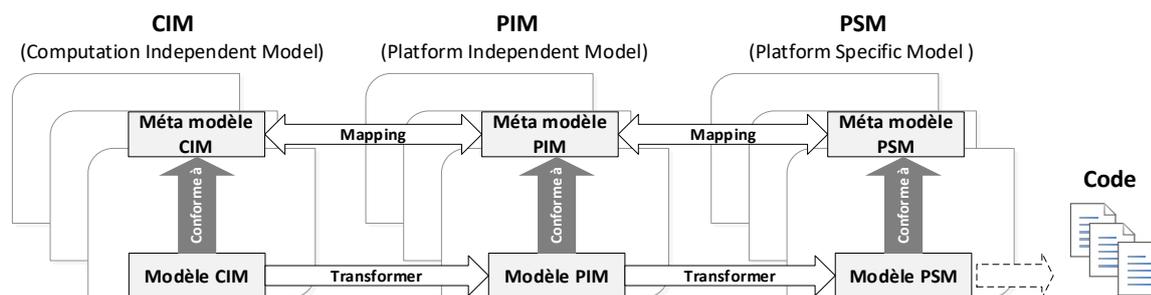


Figure 16 : Transformation des modèles basée sur l'architecture dirigée par les modèles

1.3.4. Aperçu de quelques environnements de méta-modélisation

L'environnement de méta-modélisation consiste à mettre en œuvre des méthodes de modélisation. Il repose non seulement sur la conception des langages de modélisation et la génération des outils pour instancier ces langages (Lamine et al., 2020), mais aussi sur la réalisation des fonctionnalités importantes, par exemple les transformations M2M sous forme de mécanismes et d'algorithmes (Fill and Karagiannis, 2013).

1.3.4.1. Plateforme de méta modélisation sous Eclipse (EMF, GMF, GEF)

Trois cadres logiciels : EMF (*Eclipse Modeling Framework*) (Steinberg et al., 2008), GMF (*Graphical Editing Framework*), GEF (*Graphical Modeling Framework*), forment une plateforme de méta modélisation au sein de l'environnement Eclipse. Cette plateforme fournit une infrastructure solide permettant à l'utilisateur de :

- Concevoir structurellement les métamodèles avec les syntaxes abstraites et les sémantiques quant au domaine étudié, basé sur le langage de méta modélisation Ecore embarqué dans EMF.

- Programmer les comportements graphiques et les notations graphiques souhaités (syntaxes concrètes) pour les concepts définis dans les méta-modèles, basés sur GMF et GEF (Kolovos et al., 2010).
- Décrire les règles de correspondance supportant les transformations M2M entre le métamodèle source et cible, basé sur le langage standard QVT (*Query/ View/ Transformation*) intégré dans l'environnement EMF.

Toutefois, l'utilisation de cette plateforme n'est pas facile. L'utilisateur doit avoir une connaissance approfondie du langage de programmation Java pour spécifier un langage de modélisation et les règles de transformation, ensuite mettre en œuvre un éditeur de modèle visuel utilisant GMF et GEF. Même si GMF soutient la mise en place d'éditeurs de modèles visuels au-dessus d'EMF (Aniszczyk, 2006), (Kolovos et al., 2009) ont remarqué que « la mise en œuvre d'un éditeur visuel utilisant les installations intégrées de GMF est une tâche particulièrement complexe et risquée et nécessite une courbe d'apprentissage abrupte ».

1.3.4.2. GME (*Generic Modeling Environment*)

GME (*Generic Modeling Environment*) est une plate-forme de méta modélisation configurable qui permet de créer facilement des environnements de modélisation spécifiques à un domaine, développées à l'ISIS (*Institute for Software Integrated Systems*) de l'université Vanderbilt. Cette plate-forme est aussi le composant principal de la dernière génération de technologies MIC (*Model Integrated Computing*) et a une architecture extensible reposant sur COM (*Component Object Model*) de Microsoft (Davis, 2003). Ainsi, l'environnement GME est entièrement basé sur des composants avec des interfaces publiques. En ce qui concerne l'approche MIC de GME, un point crucial est la génération, où les modèles conceptuels sont transformés en modèles exécutables et en modèles d'analyse. Les transformations sont basées sur la structure de graphe. Le système de transformation des graphes doit soutenir ces transformations dans des domaines hétérogènes (Agrawal et al., 2006).

GME est bien documenté et peut être considéré comme un outil à la fois générique et spécifique à l'utilisation. La généricité provient de l'architecture de GME qui est entièrement basée sur des composants. Les langages de programmation principaux pour la manipulation de l'environnement sont C++ et Visual Basic. La spécificité est due au fait que les fonctionnalités concernant les transformations M2M exigent une bonne compréhension de la notion de graphe pour adapter la représentation des concepts dans le domaine et la traduction des règles de transformations en graphe.

1.3.4.3. ADOxx

ADOxx est une plateforme de méta-modélisation dédiée au développement et à la configuration de méta-modèles conçus pour un domaine spécifique ou générique. La plateforme permet de mettre en œuvre des méthodes de modélisation complètes, consistant non seulement en la définition d'un langage de modélisation spécifique à un domaine, mais aussi en une procédure de modélisation, et les fonctionnalités adéquates sous forme de mécanismes et d'algorithmes. Cette plateforme est largement utilisée pour construire des preuves de concept (PoC) dans les activités de recherche en ingénierie dirigée par les modèles. Elle a créé une communauté

OMILAB (Fill and Karagiannis, 2013) qui partage des expériences et des applications ainsi développées dans une optique de réutilisabilité.

ADOxx comprend deux modules : *ADOxx Development Toolkit* et *ADOxx Modeling Toolkit*. *ADOxx Development Toolkit* est la base de l'implémentation avec :

- Une approche de configuration permettant de développer le langage de modélisation personnalisé et spécifique à un domaine, à travers la mise en œuvre de la syntaxe, de la sémantique et de la notation graphique de modélisation.
- Une approche de script permettant d'utiliser de nombreuses fonctionnalités prédéveloppées (la gestion des utilisateurs, l'interrogation et la simulation par exemple) pour enrichir le langage de modélisation avec des algorithmes et des mécanismes disponibles, ou en écriture libre par AdoScript pour les fonctionnalités particulières comme les transformations M2M.

ADOxx Modeling Toolkit autorise les utilisateurs bénéficiant de privilèges de connexion à utiliser et vérifier les langages de modélisation, ainsi qu'à appliquer les fonctionnalités développées par *ADOxx Development Toolkit*.

L'utilisation de la plate-forme ADOxx ne requiert aucune connaissance d'un langage de programmation. Les méta modèles peuvent être spécifiés visuellement au sein de *ADOxx Development Toolkit*, et les éditeurs de modèles (*ADOxx Modeling Toolkit*) sont automatiquement créés pour une utilisation sur le bureau ou sur le web sans aucune compilation. Par ailleurs, ADOxx fournit un environnement multi-utilisateur et un répertoire lié à une base de données relationnelle destinée aux méta-modèles et modèles développés. Au cours des vingt dernières années, plus de 60 langages de modélisation spécifiques à un domaine ont été réalisés avec ADOxx (Thabet, 2020).

Les environnements de méta-modélisation se multiplient aujourd'hui. Ces outils donnent la possibilité de mettre au point des palettes graphiques conformes à un méta-modèle du domaine pour pouvoir produire des diagrammes illustrant des points de vue de la modélisation. Ce sont des moyens qui ont été promus par le développement de l'IDM, et que nous souhaitons mobiliser de manière opportune dans le cadre de ces travaux. Dans (Thabet et al., 2018) (Lamine et al., 2020) (Thabet, 2020), le degré de spécificité et d'utilisabilité de six outils de méta-modélisation (EMF/GMF/GEF, DiaMeta, VMSDK, GME, MetaEdit+, ADOxx) a été évalué. Nous invitons le lecteur intéressé à se référer à ces travaux de notre laboratoire. Leur synthèse conclut qu'ADOxx est l'outil qui offre le meilleur rapport en analyse de la valeur, c'est à dire selon un ratio qualité/coût d'ingénierie.

2. Conclusion et aperçu des questions de recherche

Conformément à l'état des connaissances sur la coordination de PAD présenté aux chapitres I et II, et en adoptant le principe d'une architecture dirigé par les modèles (CIM-PIM-PSM), notre travail de recherche doit s'attacher à capturer de manière exhaustive toutes les dimensions du problème de coordination opérationnelle dans la PAD afin de répondre à des exigences d'acteurs multiples et variés. La suite de ce mémoire retrace notre recherche en trois parties présentées comme une chaîne linéaire (cf. la Figure 17).

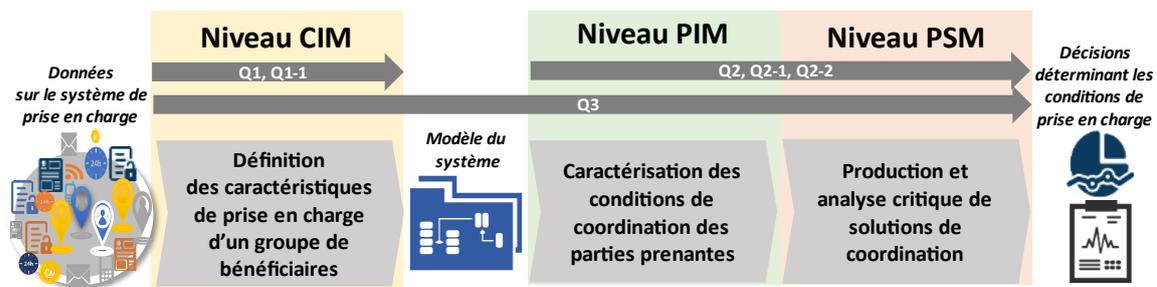


Figure 17 : Architecture dirigée par les modèles pour l'aide à la décision de coordination dans la prise en charge à domicile avec le positionnement des questions de recherche

La première partie, le CIM, est un module de définition de prise en charge à domicile. Il permettra de soutenir des exigences de personnalisation de masse du processus de prise en charge d'un bénéficiaire. C'est un point dont il faut tenir compte dans la mise au point d'un outil de modélisation dédié aux spécifications de problème de coordination opérationnelle, et donc appelant un langage adapté à la description de la coordination opérationnelle de structure de PAD. Les questions de recherche posées (Q1, Q1-1) recevront une réponse par la conception d'un langage de modélisation spécifique au domaine (DSML) :

Q1 : Comment mobiliser les connaissances organisationnelles et managériales pour planifier et coordonner les interventions à domicile, et évaluer la qualité de la prise en charge en prenant en compte la diversité des structures PAD ?

- **Q1-1 : Comment définir un langage de modélisation spécifique au domaine (DSML) permettant de représenter des connaissances qui peuvent peser sur le bon fonctionnement de la coordination dans la prise en charge à domicile dans toute sa variété ?**

Ces Q1 et Q1-1 seront donc traitées sous cet angle dans le chapitre IV. C'est l'étude de la condition initiale. Le langage spécifique sera élaboré à partir d'un méta-modèle proche des éléments caractéristiques du domaine et fidèle à notre analyse de la littérature. Les concepts caractérisant les problèmes de coordination opérationnelle seront introduits avec un soin apporté à la sémantique des éléments du méta-modèle qui les rassemble.

La deuxième partie couvre les phases relatives aux PIM et PSM. C'est un outil de gestion de connaissances permettant de formuler, puis de résoudre des problèmes de coordination à partir des modèles fournis par le premier module dont il est alors question. Ainsi, les questions de recherches posées (Q2, Q2-1, Q2-2) pour envisager la formulation et la résolution des problèmes de coordination seront concernées par nos propositions.

Q2 : Comment formuler mathématiquement des problèmes de coordination à partir des connaissances métiers ?

- **Q2-1 : Comment enrichir l'espace de formulation de problèmes de coordination avec des formulations mathématiques variées et dans des langages adaptés ?**

- **Q2-2 : Comment orienter ces formulations pour être compatible avec les outils de calcul de solution, puis publier ces solutions et les agréments d'indicateurs permettant de juger leurs qualités ?**

Ces questions Q2, Q2-1 et Q2-2 feront l'objet du chapitre V. C'est l'étude de la condition finale. La gestion de connaissances doit permettre de spécifier des fonctions traduisant des objectifs à satisfaire, de décrire des contraintes et de déclarer les variables à calculer (domaines de définition des nombres entiers et réels), à partir de la connaissance fournie par le modèle métier traduite en un modèle mathématique d'optimisation.

Enfin la troisième partie prend du recul et revient sur une interaction entre CIM, PIM et PSM. C'est l'étude de la trajectoire d'ingénierie qui relie la condition initiale à la condition finale. Nous nous concentrons sur la transformation semi-automatisée à partir des acquis des réponses à Q1 et Q2. Cette transformation de modèles doit être menée judicieusement afin de pouvoir générer le PSM après avoir spécifié les problèmes propres à une structure par l'utilisateur au niveau CIM. La question de recherche posée (Q3) est alors l'objet de nos préoccupations :

Q3 : Comment définir une série de règles de transformation de modèles à partir d'un langage spécifique au domaine (DSL) permettant de supporter un processus dématérialisé allant des connaissances orientées métiers vers les connaissances mathématiques requises pour traiter les problèmes de coordination, et les formuler de manière systématique ?

Ce sera le sujet abordé dans le chapitre VI. L'étude est surtout motivée par une proposition pour entreprendre les transformations entre les modèles (*Model to Model transformation – M2M*), mais aussi vis-à-vis du code (*Model to Code transformation – M2C*).

L'ensemble des trois parties forme notre ambition, celle d'une suite cohérente permettant une flexibilité dans l'exercice du métier de coordinateur, en allégeant au maximum les tâches de définition de problèmes, celles de gestion des données utiles pour l'aide à la décision et celles de recherche de solutions faisables et concourant à l'obtention d'une performance globale voulue.

CHAPITRE IV. MODELISATION ORIENTEE METIER BASEE SUR UN LANGAGE DE MODELISATION SPECIFIQUE AU DOMAINE (DSML)

CHAPITRE IV. MODELISATION ORIENTEE METIER BASEE SUR UN LANGAGE DE MODELISATION SPECIFIQUE AU DOMAINE (DSML)	67
1. INTRODUCTION	68
2. MODELISATION CONCEPTUELLE ET LANGAGE SPECIFIQUE.....	70
2.1. Quelques expériences utilisant un DSML	72
3. CONCEPTION DE DEUX LANGAGES SPECIFIQUES PARENTS	73
4. CONCEPTION INCREMENTALE DU DSML-HHCRSP	75
4.1. Conception préliminaire du DSML-TSP.....	75
4.1.1. Les variantes du TSP	75
4.1.2. Spécification du Méta modèle du DSML-TSP : syntaxe abstraite et sémantique.....	77
4.1.3. Spécification de la syntaxe concrète du DSML-TSP.....	79
4.2. Transition du DSML-TSP vers le DSML-HHCRSP	81
4.2.1. Spécification du méta modèle-HHCRSP : syntaxe abstraite et sémantique	81
4.2.1.1. Vue de l'offre	82
4.2.1.2. Vue de la demande	85
4.2.1.3. Vue décisionnelle sur la coordination opérationnelle des services de PAD	86
4.2.2. Spécification de syntaxe concrète du DSML-HHCRSP.....	90
5. DEPLOIEMENT DU DSML-HHCRSP	94
5.1. ADOxx : environnement de méta modélisation supportant la définition du DSML	95
5.2. Méthode de modélisation.....	96
5.2.1. Atelier 1 : modélisation de l'offre.....	98
5.2.1.1. P1.1 : Modélisation des intervenants.....	98
5.2.1.2. P1.2 : Modélisation de la structure de PAD	98
5.2.2. Atelier 2 : modélisation de la demande	98
5.2.2.1. P2.1 : Modélisation des bénéficiaires et leur demande d'interventions.....	98
5.2.3. Atelier 3 : modélisation de l'interaction entre l'offre et la demande	99
5.2.3.1. P3.1 : modélisation des localisations des acteurs autour de la structure	99
5.2.3.2. P3.2 : modélisation des exigences.....	100
5.2.3.3. P3.3 : modélisation des orientations de la structure.....	100
5.3. Phase de validation.....	101
5.3.1. Modélisation d'un cas de TSP.....	101
5.3.2. Modélisation d'un cas de HHCRSP	102
6. CONCLUSION.....	105

1. Introduction

Dans ce chapitre, notre objectif est de répondre aux questions de recherche suivantes :

Q1 : Comment mobiliser les connaissances organisationnelles et managériales pour planifier et coordonner les interventions à domicile, et évaluer la qualité de la prise en charge en prenant en compte la diversité des structures PAD ?

Q1-1 : Comment définir un langage de modélisation spécifique au domaine (DSML) permettant de représenter des connaissances qui peuvent peser sur le bon fonctionnement de la coordination dans la prise en charge à domicile dans toute sa variété ?

Le bilan des connaissances du Chapitre II dresse le constat d'un besoin de mieux modéliser les problèmes de coordination opérationnelle du fait, entre autres, qu'ils reposent sur de multiples données individuelles, collectives, territoriales, économiques et réglementaires. Ainsi, puisque les demandes sont variées et les capacités d'interventions fournies par chaque structure de PAD ne le sont pas moins, il existe une forte pression qui s'exerce sur ces structures pour régulièrement piloter leurs flux de services. Poursuivre les bons objectifs est toujours le cap, mais ces objectifs sont nombreux, parfois antagonistes et ces caractéristiques dressent quelquefois des barrières en termes de qualité de décision. Dans ce contexte, l'amélioration continue qui irrigue nos démarches qualité n'est pas seulement confinée dans les processus opérationnels et supports d'une telle organisation, mais elle impacte les processus de pilotage. Elle interroge sur les modes de pensée, les activités de prises de décision, les pratiques et les outils d'aide à la décision.

Nous savons de longue date que les avantages concurrentiels, tout comme la quête de valeur ajoutée, ne peuvent prendre une réalité tangible que par un savoir-faire en aide à la décision. Le génie industriel a joué un rôle important dans cette prise de conscience. Rappelons si besoin est, l'importance donnée à la tension des flux, au juste à temps, ou encore à la satisfaction des usagers ou des agents de production dans les travaux de recherche en gestion des opérations. C'est la raison pour laquelle cette planification est de plus en plus abordée avec une formulation « maison », tenant compte de spécificités locales dans une organisation donnée. La tendance est à introduire toujours plus de connaissances métier dans la prise de décision.

Le contexte actuel est devenu celui d'une société de l'exigence immédiate. Les circuits de prestation des services, comme d'autres types de circuits, ne doivent pas seulement poursuivre des critères de performance économique et sociale, ils doivent accepter des changements de la demande et savoir y répondre. L'agilité est devenue un maître-mot. Le poids de ces incertitudes dans les espaces de prise de décision est aussi une connaissance. Elles conduisent inexorablement à vouloir plus de flexibilité dans les usages de différents outils de calcul et invitent à développer un nouvel esprit de continuité des activités de prise de décision, prolongeant ainsi logiquement le concept de continuité des activités opérationnelles. Les voies à explorer sont alors celles d'une progression dans la formulation du pilotage en explicitant les prises de décision avec plus d'exactitude vis-à-vis de la réalité du terrain, d'une part, et en pensant à quantifier les degrés de liberté dont on peut faire usage pour réagir en cours de pilotage, d'autre part.

Pour toutes ces raisons, nous avons investigué la mise au point d'un système d'aide à la décision réaliste et adaptable, leur donnant avant tout la possibilité de manipuler des combinaisons

subtiles d'orientations et d'exigences métier avec lesquelles ils doivent composer. En outre, du fait de cette nature intrinsèquement instable des demandes de coordination dans la prise en charge à domicile, le coordinateur aura régulièrement besoin de reformuler des questions et à leur trouver d'autres réponses. C'est un métier central dans le cadre d'une coordination centralisée. Pour assumer ce rôle, le coordinateur n'est pas toujours outillé pour réagir et nous supposons ici qu'un recours à un modèle pourrait lui faciliter la vie. Seul, il n'a pas forcément l'expertise nécessaire pour considérer l'impact du changement sur sa planification initiale. En revanche, son expertise métier lui donne la capacité à travailler des diagnostics organisationnels et à imaginer des modifications de ses données de planification pour agir et toujours coordonner avec justesse. Ainsi, nous devrions maîtriser une translation de leurs connaissances et de leur expertise métier vers une capacité à mieux préparer la prise de décision. Il manque un système numérique d'aide à la décision qui permettrait de mieux saisir ces connaissances. Dessiner ce système est au cœur de notre travail de recherche, il débute par une étude de la modélisation à un niveau générique qui serait appropriée pour permettre à ce coordinateur de s'exprimer et de gérer le changement à moindre effort et à un niveau particulier.

Résumons-nous. Nous émettons l'hypothèse qu'avec le temps, le verrou scientifique se déplace d'un savoir-faire factuel en aide à la décision vers un besoin nouveau de prise de décision rapide. Les outils doivent être conçus pour faire face à la variété des situations à traiter et pouvoir prendre des décisions circonstanciées en contexte de pression temporelle (ce qui revient à considérer que le nombre d'interlocuteurs dans la chaîne décisionnelle est réduit). L'objectif poursuivi est donc de chercher comment donner au coordinateur la faculté de faire une spécification en ligne et à moindre effort de ses problèmes.

Une telle ligne de pensée invite à revisiter la place de la décision dans la modélisation conceptuelle. Car cette ambition ne peut pas se libérer de la quantité et de la qualité des connaissances métier représentatives de la réalité du fonctionnement de l'organisation. Si l'expression du besoin de coordination est réalisée à partir de modèles métier, il faut être le plus proche possible de son univers. D'où la proposition de concevoir un langage de modélisation spécifique au domaine (DSML) et de le faire avec un outil de méta-modélisation pour maîtriser cet acte de conception.

Dans ce chapitre, nous commençons par un exposé des caractéristiques et des éléments clés dans la conception d'un DSML. Ensuite dans la section 3 de ce chapitre, nous présentons la conception de deux DSML. Nous le ferons avec une démarche progressive exploitant un voisinage entre deux classes de modèles connus en recherche opérationnelle, à savoir le TSP et le HHCRSP. Puis dans la section 4 de ce chapitre, nous compléterons la définition de ce langage en vue d'une implémentation dans notre environnement de développement. Il s'agira de transformer la syntaxe abstraite et la sémantique du domaine en une syntaxe concrète de notre DSML. Cela se concrétise par des palettes d'objets graphiques qui seront décrites. Dans la section 5 de ce chapitre, nous appliquons une méthode de modélisation faisant usage de ce DSML dans notre cas d'étude, dans une vision « fabrique de la coordination de la PAD ». Ce faisant, nous montrerons les pratiques de l'environnement de méta-modélisation ADOxx pour la saisie des données par le coordinateur. Cette méthode est employée par la suite, avec quatre déclinaisons de modèles : deux TSP et deux HHCRSP afin de démontrer la faculté de faire face à de la diversité.

2. Modélisation conceptuelle et langage spécifique

Disposer d'un modèle de système facilite le renseignement des données utiles, que le modèle a structuré, pour l'étude d'un problème. Une Interface Homme-Machine (IHM) capturera alors de manière efficace et complète les connaissances nécessaires à la mise en œuvre de besoins de décision. C'est ce que nous voulons étudier pour la coordination opérationnelle des structures de PAD. L'analyse et la conception de cette IHM va requérir de communiquer avec des coordinateurs disposant d'une expertise métier, alors même qu'ils n'ont pas de langage unifié, et parlent même des langues naturelles différentes dans certaines organisations internationales.

La modélisation conceptuelle a été poussée comme une solution pour faire face à ces problèmes. Aujourd'hui, si l'ingénierie système la met en avant, pour les organisations de production de biens et de services, la modélisation d'entreprise le faisait auparavant. Les normes ISO 19439¹ et 19440² qui en fixent les bases théoriques ont vingt ans. Un modèle conceptuel y est défini comme une abstraction d'un système représentant sa structure et son fonctionnement. Cela se fait à l'aide de concepts qui sont supposés être connus. Initialement, il y a eu un effort conséquent pour mettre à disposition des langages de modélisation génériques, cherchant des concepts assez universels dont les instanciations prenaient un caractère précis par le jeu de la sémantique associée aux objets graphiques. Plus récemment, la modélisation conceptuelle a trouvé un nouvel élan avec le développement de langages pour décrire un système dans un domaine cible et spécifique (Bock and Frank, 2016). Ces concepts spécifiques au domaine sont conventionnellement construits avec une syntaxe abstraite qui instancie la sémantique utilisée par les modèles issus de langage générique. Par la même, le modélisateur est plus proche de l'univers de discours du domaine concerné. Le vocabulaire du langage est intelligible et sa grammaire est précise. Les travaux de (Fill, 2017) expliquent ce que les annotations sémantiques de modèle conceptuel à base de langage générique apportent en termes la modélisation conceptuelle orientée métier. Cet auteur pointe des buts qui sont les motivations profondes dans l'usage de DSML auxquels nous sommes attachés :

- Gestion de la complexité du système.
- Mise en œuvre facilitée de l'interaction entre les personnes et les composantes de systèmes.
- Préfiguration plus fine des usages de modèles pour l'analyse ou la simulation.

Cette idée de concevoir des modèles par usage de DSML remporte une popularité croissante. De fait, ils sont pratiques, très intuitifs, soutiennent la productivité de la modélisation et contribuent à améliorer la qualité et la compréhensibilité des modèles (Michael and Mayr, 2015). Si les langages de modélisation d'entreprise sont souvent promus par des logiciels BPM qui relèvent d'un marché de l'édition logicielle, le développement de solutions numériques pour modéliser avec des DSML passe par une voie différente. Ce sont les éditeurs qui développent des environnements d'ingénierie dirigée par les modèles qui procurent les facilités de conception et de développement.

¹ ISO 19349 Entreprise intégrée - Cadre de modélisation d'entreprise : www.iso.org/fr/standard/33833.html

² ISO 19440 Modélisation et architecture d'entreprise - Construits pour la modélisation d'entreprise : www.iso.org/fr/standard/74491.html

Afin d'éviter toute surcharge en termes de syntaxe de langage, mais aussi avec le souci d'avoir tout degré de liberté sur le plan technique pour écrire, utiliser et réutiliser les connaissances portées par un tel langage, le cadre retenu est celui de la méta-modélisation. Un méta-modèle est ici une spécification formelle de DSML. Les processus du développement d'un DSML ont été étudiés en détail par (Frank, 2010) (Michael and Mayr, 2015). Pour ce faire (Frank, 2013), il faut le développer en trois temps :

- Un modèle de niveau M2 est imaginé (syntaxe abstraite d'un langage)
- Une description sémantique textuelle lui est ajoutée pour mieux définir les relations entre les concepts
- Une déclinaison des concepts du M2 sous forme graphique permet de créer la palette graphique de modélisation (syntaxe concrète d'un langage)

La Figure 18 schématise comment un DSML s'inscrit dans la fameuse pyramide en quatre niveaux popularisés par l'OMG (OMG, 2014) (même si, ici, nous ne dessinons pas la pyramide). Nous reprenons la convention de nommage de ces niveaux, de M0 à M3. Au troisième étage de cette figure, le niveau du M2, nous définissons un DSML en utilisant un langage de méta-modélisation fourni par le niveau M3. Le méta-modèle du DSML doit donc être conforme à ce méta méta-modèle. Au niveau M1, nous utilisons le DSML pour créer des modèles concrets (M1) qui représentent une réalité du niveau M0 (Michael and Mayr, 2015). Parce que l'IDM s'est répandue plus rapidement dans le secteur du génie logiciel que dans d'autres secteurs, la lecture faite de l'architecture MDA de l'OMG par beaucoup d'auteurs de ce domaine proposent que le niveau M0 soit le code d'une application.

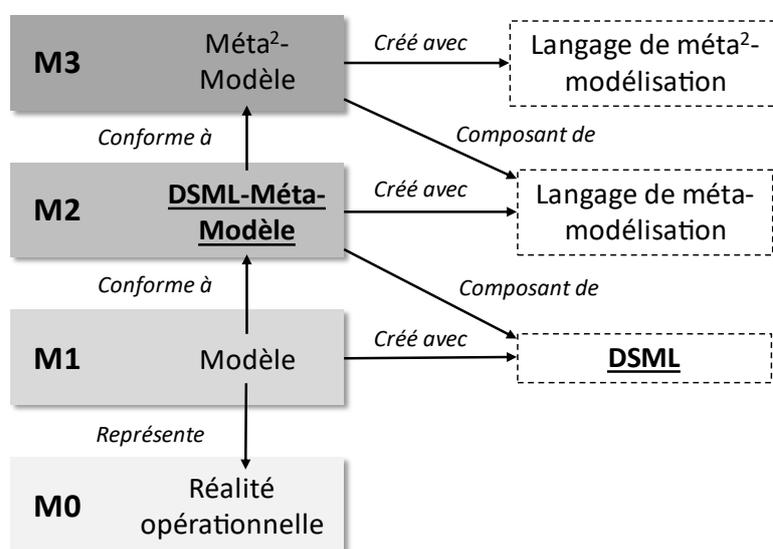


Figure 18 : Positionnement du DSML dans la pyramide en quatre niveaux d'OMG (Michael and Mayr, 2015).

Les modélisateurs sont donc les utilisateurs du DSML. Ils agissent au niveau M1. Dans notre domaine d'étude, ce sont les coordinateurs de la structure de PAD que nous imaginons candidats pour jouer ce rôle. Le DSML apporte de la valeur. Il les libère de la nécessité de créer des concepts spécifiques au domaine pour traduire leur vision en langage naturel. Il leur suggère de réutiliser ce qui est mis à leur disposition. Le DSML permet alors de favoriser la qualité des modèles, puisque les concepts d'un DSML ont été soigneusement développés par les chercheurs

et les ingénieurs pour être expressifs dans le domaine d'application. Enfin, les spécifications du DSML peuvent inclure des restrictions spécifiques à un domaine qui empêchent dans une certaine mesure, la construction de modèles inappropriés (Bock and Frank, 2016). C'est ce qu'on nomme la « vérification » dans les outils qui ont développé ce type de fonction.

Il faut souligner ici que notre usage du modèle est connu. Un modèle d'une structure de PAD a vocation à décrire *in fine* un problème relevant de la recherche opérationnelle. Cette reconnaissance préliminaire de la destination des usages de modèle influence la conception de notre DSML. Il est adapté à la description du besoin de prendre un certain type de décisions. Ici, l'enjeu est celui d'un problème de coordination opérationnelle de la PAD, donc **d'affectation des ressources et de planification de tournées**. Elles devront pouvoir être modélisées. Le DSML doit proposer des éléments de syntaxe concrète pour aider à représenter ces fonctions de gestion. De plus, toute variante du problème de coordination connue pourra être modélisée en utilisant les primitives natives de notre langage. Donc, tout en restant spécifique au domaine, la variété de celui-ci sous-tend un pouvoir certain d'expression et de couverture du langage que nous devons intégrer dans le registre des besoins.

Ainsi, la syntaxe de notre DSML donnera donc naissance à une capacité à renseigner les données entrant dans la formulation du problème à poser. Ce qui autorisera aussi d'échanger avec lui pour exporter ou importer tout ou partie de ces données.

2.1. Quelques expériences utilisant un DSML

De nombreux DSML sont disponibles et ils sont logiquement appliqués dans différents contextes métiers. Le DSML lorsqu'il est inscrit dans une démarche d'ingénierie dirigée par les modèles, tend à simplifier la complexité technique rencontrée dans un acte de conception et développement. C'est peut-être ce qui explique leur essor dans les logiciels de type PLM (*Product Life-cycle Management*) en particulier. Ils offriraient l'avantage de réduire les barrières qui freinent l'adoption de savoirs expérientiels dans les projets, c'est-à-dire de connaissances en provenance des utilisateurs de livrables de ces projets.

Dans (Tang et al., 2008), le processus de définition d'un DSML vise la modélisation de jeux sérieux pour aider à leur conception. Une approche évolutive est adoptée lors de la définition du DSML, telle que proposée par (Tolvanen, 2006). Un cadre d'ingénierie pour la conception de jeux sérieux est ensuite proposé, tenant compte des exigences propres à cet exercice de création d'un jeu sérieux et des résultats de l'analyse par des joueurs experts pendant la conception.

Dans (Bock and Frank, 2016), les auteurs décrivent une méthode de modélisation d'entreprise multi-perspective (MEMO) et une implémentation de celle-ci dans ADOxx. MEMO comprend un ensemble de plusieurs DSML, intégrés dans la perspective d'accompagner une vision architecturale du système d'information dans une organisation (couches métier/fonctionnelle/logique). C'est au sein du même environnement de méta-modélisation qu'a été développé le DSML d'AdoBPRIM, appliqué à la prise en charge médicamenteuse en environnement hospitalier (Lamine et al., 2020).

Les entreprises doivent non seulement fournir de bons produits ou services, mais sont aussi désormais juridiquement responsables sur le plan environnemental et social de leurs actes et des

conséquences internes et externes. Pour (Campos and Grangel, 2018), la fameuse responsabilité sociale et environnementale (RSE) est devenue un sujet stratégique qui oblige à repenser de nombreux processus d'entreprise. Pour y parvenir, un DSML est défini pour représenter la RSE à l'aide d'un profil UML.

Le système de santé est le cadre de nos travaux de recherche, il existe des initiatives dans ce domaine. (Strobel, 2021) propose un DSML pour décrire l'usage des objets connectés dans les systèmes de soins de santé. Cet usage, selon ces auteurs, permettrait de repousser les seuils de saturation de ressources médicales. Le développement de ce DSML est basé sur la démarche en trois étapes proposées par (Frank, 2013).

Dans (Heß et al., 2015), pour matérialiser les avantages attendus des parcours cliniques dans les hôpitaux, une approche par modélisation tenant compte des particularités des systèmes hospitaliers est introduite. Un DSML pour les parcours cliniques est ainsi présenté. Des DSML naissent de spécialisation de langages de modélisation génériques, c'est ce que propose (Braun et al., 2014) en suggérant de modéliser des parcours cliniques sous forme d'une extension de BPMN : BPMN4CP.

Concernant notre domaine d'application, aucune application du DSML n'a été trouvée en vue de la modéliser des problèmes de coordination opérationnelle de structure de PAD. Et cela est aussi vrai pour la spécification de problèmes dits classiques de la Recherche Opérationnelle où les DSML se réduisent à la préparation de problèmes mathématiques bien formulés propres à des solveurs.

En résumé, les DSML sont développés et mobilisés pour faire face à des domaines où les connaissances sont très variées et insuffisamment structurées. Néanmoins, la plupart d'entre eux n'ont pas de spécifications élaborées sur une même base. Ils sont nativement étrangers les uns aux autres et avec très peu de filiation.

Par conséquent, nous avons d'abord développé un langage spécifique pour décrire un problème de voyageur de commerce (TSP) et dans sa plus stricte simplicité. Puis, nous l'étendons par une mécanique de dérivation et spécialisation du DSML initial. La disponibilité de ce premier noyau de DSML formera donc les fondations d'une démarche de conception du DSML répondant au besoin de coordination opérationnelle de la structure de PAD. Il sera utilisé pour modéliser un problème HHCRSP sous-jacent en recherche opérationnelle.

Nous exploitons donc une série de dépendances entre ce que nous nommerons par la suite le DSML-TSP et le DSML-HHCRSP.

3. Conception de deux langages spécifiques parents

La démarche de conception des DSML reflète ces considérations générales. La définition du DSML débute avec le noyau (au niveau M2), c'est-à-dire le langage « DSML-TSP » que l'on mettra à l'épreuve pour estimer les difficultés qui se dressent quand on veut le rendre apte à traiter différentes variantes du voyageur de commerce. Ces variantes forment autant de problèmes différents sur lesquels nous pourrions raisonner. Cette émancipation du DSML-TSP nous conduira logiquement vers le DSML-HHCRSP.

Pourquoi définir ce noyau de base ? Puisqu'ayant conscience de la complexité du problème de coordination opérationnelle de PAD (cf. la section 2.1 du chapitre II), nous avons choisi de

monter en compétences à partir d'un problème plus simple et très accessible, tout en étant dans la même famille. Nous supposons que la classe de problèmes HHCRSP n'est qu'une variété spéciale du problème de routage de flotte de véhicules VRP (*Vehicle Routing Problem*) (Toth and Vigo, 2002). Plus encore, c'est au sein de cette classe VRP que nous situons le problème du voyageur de commerce (*Travelling Salesman Problem*, TSP). C'est (Dantzig and Ramser, 1959) qui développent cette idée que la famille des problèmes TSP peut être considérée comme une sous-classe de celle des VRP. Le VRP est identifié par (Bektas, 2006) comme un « multi-TSP ». Par conséquent, la compréhension du TSP devient le candidat idéal pour une démarche progressive par étapes, allant vers des problèmes plus difficiles. Car le DSML du TSP emportera moins de concepts au niveau métier que le VRP, puis que le HHCRSP. La Figure 19 schématise ces relations de filiation entre TSP, m-TSP, VRP et HHCRSP.



Figure 19 : Relation entre TSP, m-TSP, VRP et HHCRSP

Pratiquement, notre pari est donc que nous pouvons atteindre la définition des concepts du HHCRSP, incluant leurs attributs et des syntaxes concrètes, en héritant des concepts du TSP. Par exemple, le concept « voyageur » dans le DSML-TSP sera la source de celui d'« intervenant » dans le DSML-HHCRSP. Par conséquent, nous traitons de mécanismes d'héritage.

La Figure 20 est une vision détaillée du contenu de la Figure 18 expliquant les principes que nous venons d'exposer.

Nous avons choisi une base conceptuelle au niveau M3 qui est mise à disposition par ADOxx. Comme ce Méta-Méta-Modèle (M²M) est intégré dans un environnement de méta modélisation, cette plate-forme de développement « *low-code* » nous permet de (1) développer rapidement les DSMLs via des interfaces graphiques par interaction homme-machine, (2) mobiliser dans une panoplie de fonctions, celles qui permettent aux utilisateurs finaux d'utiliser, de tester et d'évaluer les résultats de la conception des DSML que ce soit en cours de réalisation ou pour la solution ultime. Concernant notre démarche de recherche, cet ensemble a été bénéfique et adapté à l'évolution de notre perception de ce que ces langages devaient savoir modéliser. Le lecteur peut se référer à (Fill and Karagiannis, 2013) pour en savoir plus sur la description de ce M²M ADOxx. Le haut niveau d'abstraction de ce dernier est visible dans le modèle conceptuel de la Figure 21.

Les métamodèles, ainsi que les modèles créés avec ces DSMLs, sont compatibles parce qu'ils ont la même « origine », ils sont formés à partir de méta-concepts communs dans ADOxx. En d'autres termes, les différents DSMLs sont liés parce que chacun possède des concepts qui par des (méta)relations pointent d'autres concepts des autres langages. Ces relations intégratives sont de qualité parce que tous les langages sont définis à partir du même M²M.

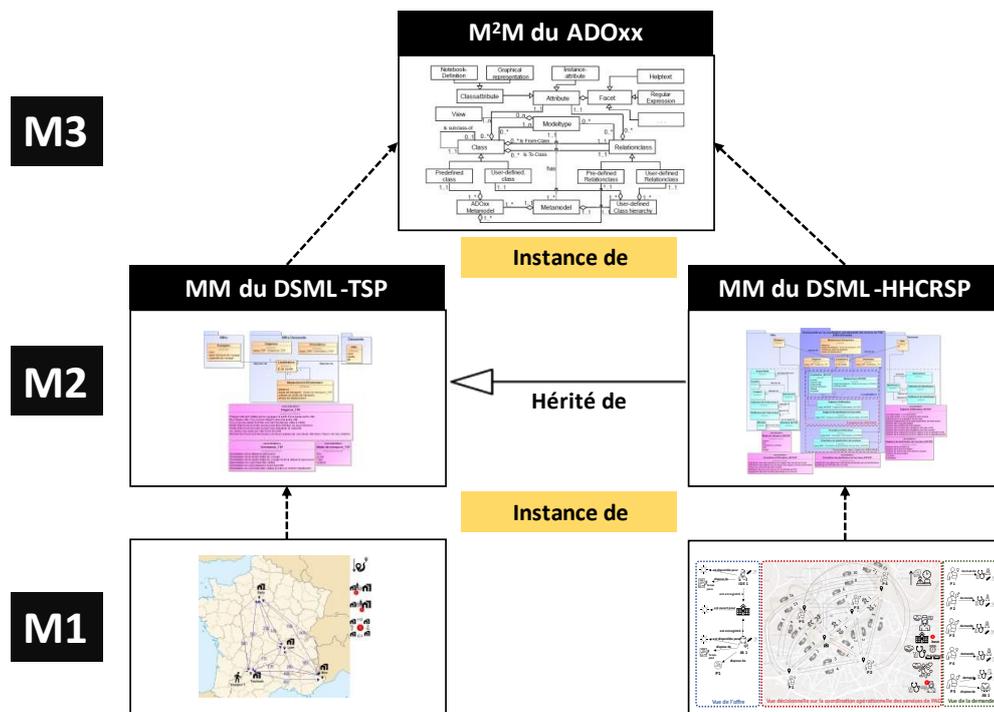


Figure 20 : Architecture et positionnement à l'OMG du DSML-HHCRSP

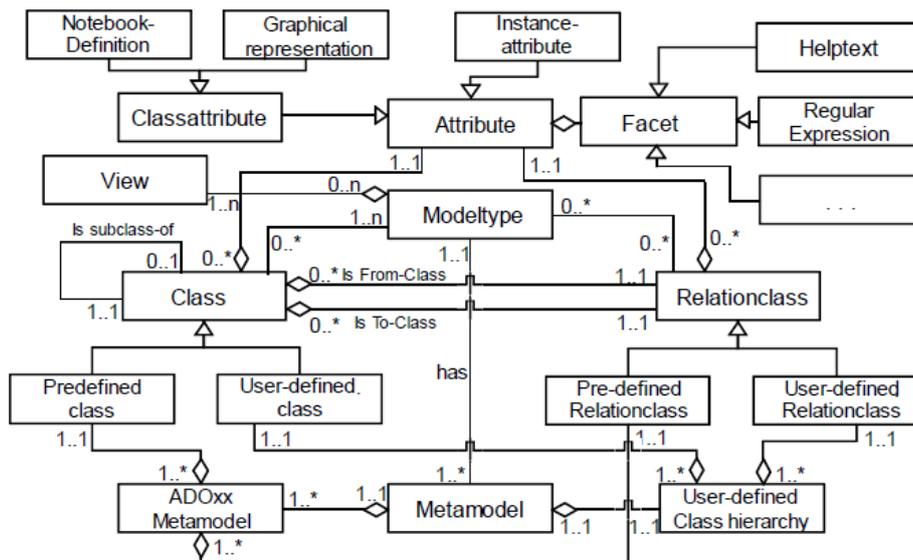


Figure 21 : Méta-Méta-Modèle (M²M) du ADOxx (Fill and Karagiannis, 2013)

4. Conception incrémentale du DSML-HHCRSP

4.1. Conception préliminaire du DSML-TSP

4.1.1. Les variantes du TSP

Comme mentionné dans la section 2.3.1.1 du chapitre II., le problème du voyageur de commerce (abréviation de sa traduction anglaise : TSP) est une classe connue dans l'ensemble de ces problèmes. Rappelons simplement son énoncé le plus simple et le plus répandu. Un voyageur doit visiter une et une seule fois un ensemble donné de villes. Sa tournée est conçue pour le satisfaire, au sens où le TSP est souvent présenté comme un problème de minimisation

de la distance parcourue par le voyageur. Cette forme du plus court chemin est un problème NP complet. Il existe de nombreuses variantes du TSP selon (Gutin and Punnen, 2006). Ces auteurs décrivent un ensemble d'au moins dix variantes de cette formulation basique. Evoquons de manière synthétique les changements apportés, en trois parties :

- La fonction traduisant l'objectif à optimiser peut varier pour traduire des moyens de transport en compétition avec des temps de transport différents, à savoir la maximisation du gain total des visites dans le *Max TSP* (Barvinok et al., 2007), la minimisation du coût maximum d'une tournée dans le *Bottleneck TSP* (Kabadi and Punnen, 2007) et la minimisation du coût total des visites suivant un chemin Hamiltonien dans le *Messenger problem* (Jünger et al., 1995).
- Les contraintes peuvent varier. Ainsi, dans le *Prize-Collecting TSP* (Balas, 1989), chaque ville est associée à un poids non négatif, une tournée du voyageur est considérée comme faisable si et seulement si le poids total des villes visitées n'est pas inférieur à un seuil donné. Dès lors, dans une extension, le *Capacitated Prize-Collecting TSP* (Tang and Wang, 2008) propose que chaque voyageur possède une capacité de voyage au point de départ, le poids total d'une tournée (apparentée à une charge) ne peut pas dépasser cette capacité. Dans le *TSPM (TSP with multiple visits)* (Gutin and Punnen, 2006), le fonctionnement peut relaxer la notion de visite unique en spécifiant au moins une visite par ville.
- La nature du problème peut changer quand on touche aux hypothèses fondamentales. Dans le *Clustered TSP* (Guttmann-Beck et al., 2000), plusieurs sous problèmes coexistent avec une recherche de coût minimal dans un ensemble des villes partitionné en plusieurs classes indépendantes. Ce même problème peut se décliner dans un autre but au sein du *GTSP (Generalized TSP)* (Noon and Bean, 1993), cherchant une seule ville dans chacune de ces classes. Puis, le *Covering Tour Problem* (Gendreau et al., 1997) est une variante de ce GTSP, où les classes de villes ne sont pas nécessairement disjointes. Enfin, dans le *MTSP (Multiple Traveling Salesman Problem)* (Song et al., 2003), la couverture des villes peut être imaginée en mobilisant plusieurs voyageurs partant de la même ville.

Le Tableau 5 est une synthèse de ces variantes avec une classification où sont incluses les propriétés qui doivent être considérées en surcharge de la formulation du TSP initial. Cet ajout est à la source de besoins d'extension du DSML-TSP quand nous voudrions couvrir ces variantes en modélisation avec notre DSML-TSP.

Tableau 5 : Synthèse des variantes du TSP

Variante du TSP	Type de variante			Propriétés supplémentaires par rapport au TSP classique		
	Objectif	Contrainte	Nature	Attributs	Type d'objectif	Type de contrainte
Max TSP	X				La maximisation du coût total des visites	
Bottleneck TSP	X				La minimisation du coût maximum d'une tournée	
Messenger problem	X				La minimisation du coût totale des visites suivant un chemin Hamiltonien	
Prize-Collecting TSP		X		-Poids non négatif associés aux villes -Seuil minimum de voyage associé aux voyageurs		Le poids total d'une tournée ne peut pas être inférieur au seuil minimum

Capacitated Prize-Collecting		X		-Poids non négatif associés aux villes -Capacité de voyage associée aux voyageurs		Le poids total d'une tournée ne peut pas dépasser la capacité
TSPM		X				Au moins une visite par ville d'une tournée
Clustered TSP			X	Id de clusters associés à la localisation de villes		
Covering Tour Problem			X	Id de clusters associés à la localisation de villes		La recherche d'une tournée la plus courte qui passe par une seule ville dans chacun de ces clusters
MTSP			X			

4.1.2. Spécification du Méta modèle du DSML-TSP : syntaxe abstraite et sémantique

Nous devons définir ce métamodèle du DSML-TSP à partir des éléments présentés dans le paragraphe précédent qui s'apparentent à des connaissances « métier ». Nous maintenons une démarche de conception progressive, partant d'un modèle de la formulation simple pour aller vers un modèle qualifié pour les variantes du Tableau 5.

Les principaux concepts que nous identifions dans une première approche de ce TSP sont les suivants :

- Le « **Voyageur** » est la personne qui effectuera une tournée dans un ensemble des villes. Ses attributs sont : son identifiant (ex : le nom) pour une configuration de base, auquel s'ajoutent son seuil minimum de voyage pour le *Prize-Collecting TSP* et sa capacité pour le *Capacitated Prize-Collecting TSP*.
- La « **Ville** » est un lieu visité par le voyageur et connu de lui, caractérisée aussi par un identifiant (« **ID** »), s'y ajouteraient le poids de visite pour le *Prize-Collecting TSP* et le *Capacitated Prize-Collecting TSP*.

Conceptuellement, le TSP n'est pas défini dans un cadre de services distinguant offre et demande. Mais pour préparer une progression vers le DSML-HHCRSP qui le sera, nous pouvons supposer que le « voyageur » est un acteur du monde de l'offre, un vendeur travaillant pour une agence qui assure des prestations pour des clients. Toutes les « villes » sont les lieux où s'expriment des demandes. Ce sont là que résident des clients qui forment individuellement leur besoin à satisfaire.

Sur le plan des décisions, le TSP est un problème de planification des tournées. L'offre est mobile, elle se déplace pour satisfaire la demande au plus près, là où elle est exprimée, dans une relation de proximité. Ce n'est pas qu'une rencontre marchande, elle est aussi physique. Présenté ainsi, c'est un problème assimilable à celui d'une prise de décision pour que la rencontre entre l'offre et la demande soit couronnée d'un certain succès ouvrant la porte à une notion de performance escomptée. Dès lors, nous définissons deux concepts offre et demande contenant respectivement des éléments physiques : le voyageur et la ville. Concernant la satisfaction de la demande (entité statique) par l'offre (entité dynamique), c'est le concept de « **Localisation** » qui s'impose comme le point de rencontre physique. Etant données les connaissances géographiques sur le point de départ du voyageur et la localisation de chaque ville, le concept de « déplacement » embarque cette réalité du mouvement du voyageur, suivant une logique de parcours des villes depuis la structure de départ jusqu'au retour à la fin. Le déplacement n'est en réalité qu'une somme de déplacements élémentaires (« **Déplacement élémentaire** ») qui se situent entre des paires de localisation. Si nous appelons tournée ce « déplacement » total, il naît un besoin de décider de l'itinéraire (ordre des déplacements élémentaires) le meilleur parmi

l'ensemble des itinéraires possibles. Pour être réaliste, chacun de ces déplacements élémentaires peut se définir de plusieurs façons selon que l'on considère :

- La distance exprimée en longueur dans une unité de kilomètre.
- Le choix d'un mode de transport parmi plusieurs options. Par exemple avec trois modalités : avion, train ou voiture. La vitesse approximative de ce mode de transport (en km/h) conjuguée avec la distance parcourue par le moyen choisi donnera un temps de voyage (en minutes) entre deux villes.

C'est l'introduction des exigences métier qui cadrera la rencontre entre l'offre et la demande. Afin d'imaginer les primitives du DSML-TSP exprimant cela, il faut introduire les éléments pesant sur la prise de décision.

Passons pragmatiquement par une analyse sur le cas du TSP. Dans la formulation basique, une première volonté est que le voyageur ne visite chaque ville qu'une seule fois et termine la tournée dans sa ville de départ (le voyage est un circuit unique). Elle doit être garantie par les décisions prises. Il s'agit donc ici d'expliquer cette exigence qui se transformera en un jugement de cohérence qui sera porté sur le choix de la tournée. Mais quand on quitte cette formulation pour aller vers des variantes, d'autres exigences complètent ou se substituent à cette première volonté. Il y a alors deux difficultés qui apparaissent :

- (1) Le modélisateur doit donc personnaliser son ensemble d'exigences avec le DSML-TSP. Techniquement, une liste d'énumérations incluse dans le méta-modèle va permettre d'instancier ces exigences de manière flexible. On peut assimiler ces opérations à ce qu'un programmeur ferait quand il souhaite mobiliser une fonction appropriée à ses exigences dans une bibliothèque. Par exemple, un attribut « *type_TSP* » peut pointer vers cette liste d'énumération considérée comme un type de variable dans la spécification du DSML par le méta-modèle.
- (2) La conjonction des exigences choisies amplifie la problématique de maintien de cohérence. Il s'agit de préserver un espace de décisions faisables en gérant des conflits potentiels entre exigences. Par exemple, dans le TSP basique, l'exigence exprimée supra peut se désagréger en un ensemble de trois exigences qui coexistent et ensemble, traduisent une même réalité :
 - Chaque ville est visitée par le voyageur à partir d'une seule autre ville.
 - De chaque ville, il n'y a qu'un départ vers une autre ville.
 - Il n'y a qu'une seule tournée couvrant toutes les villes à visiter.

Enfin, toujours sur la rencontre entre l'offre et la demande, un autre besoin est de pouvoir décrire ce qu'est la notion de préférence dans l'espace des décisions faisables pour affiner le résultat et le réduire à une seule solution. Par exemple, la minimisation de la distance totale de la tournée peut être le phare qui guide cette préférence pour un calcul de TSP. La Recherche Opérationnelle appelle cela un objectif. Nous proposons d'adopter le terme d'**orientation** pour exprimer ces préférences. Evidemment, nous tisserons des relations entre orientations et objectifs dans la suite de ce mémoire. Pour ce faire, cela offre des moyens sémantiques pour qualifier les solutions dans l'optique de les discriminer. La problématique pour notre DSML-TSP reste la même que pour les exigences. Il peut y avoir de multiples formulations des orientations. On peut ainsi préférer des distances ou des temps les plus petits, ou bien viser une

préférence économique (coût ou marge). Notons que la recherche de décision peut aussi poursuivre plusieurs lièvres à la fois, c'est-à-dire chercher des formes de compromis entre plusieurs préférences antinomiques. L'approche en Recherche Opérationnelle sera alors multicritère.

Pour couvrir des variantes du TSP, nous devons recenser les types d'exigences et d'orientations que ces variantes utilisent. Une proposition en ce sens est accessible dans le Tableau 5 où la colonne « objectif » doit être lue comme une **orientation**, et celle des « contraintes » comme une **exigence**. Ce renommage est pour le but d'augmenter la compréhensivité de notre DSML vis-à-vis les utilisateurs finaux (les experts métiers), par un emprunt des vocabulaires orientées professionnelle selon nos constats dans la section 2.2 du chapitre II. Enfin, celle de la « nature » implique les deux colonnes précédentes et simultanément.

La Figure 22 illustre le métamodèle du DSML-TSP que nous avons conçu pour couvrir ce périmètre de variantes.

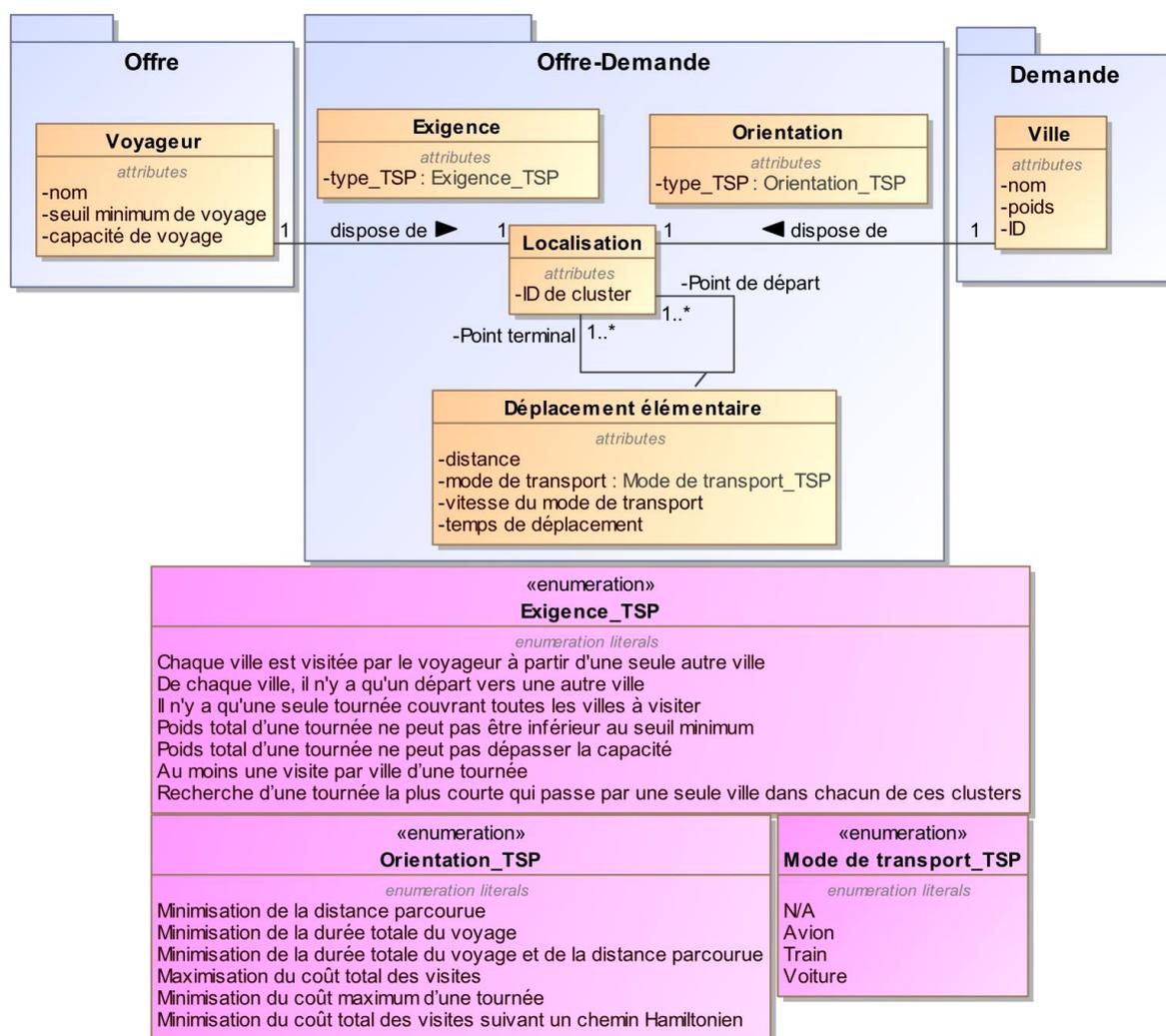


Figure 22 : Illustration de métamodèle du DSML-TSP

4.1.3. Spécification de la syntaxe concrète du DSML-TSP

Dans cette section, une représentation graphique est choisie pour instancier les éléments du langage conforme au méta-modèle du DSML-TSP. Elle peut s'apparenter aux syntaxes concrètes du DSML-TSP. Elle est décrite dans la colonne de droite du Tableau 6.

Le modèle travaille sur un fond cartographique représentant la partie du territoire incluant toutes les villes du modèle. Cette représentation fournit de la lisibilité à l'utilisateur dans la mesure où une tournée est fondamentalement associée à un ensemble de points sur cette carte géographique qu'il faut relier pour définir une tournée.

Pour le concept ayant l'attribut de type « énumération », ce type est un type de données spécial qui permet à un attribut de se présenter sous la forme d'un ensemble de constantes prédéfinies. La valeur sélectionnée de l'attribut doit être égale à l'une des constantes qui lui ont été attribuées.

Afin de traiter les primitives associées aux concepts relevant d'un type « énumération », nous décidons de définir une représentation graphique pour chacun des éléments d'une liste pour que la modélisation exprime les choix du modélisateur qui peuplent l'espace relatif à la rencontre de l'offre et de la demande. Nous avons essayé de donner une expression significative de chaque concept de ce tableau, pensant ainsi favoriser l'expressivité du langage et, par conséquent, une facilité d'utilisation (caractère intuitif de l'icône).

Tableau 6 : Représentation graphique (syntaxe concrète) des concepts inclus dans le DSML-TSP

Nom de concept	Élément de l'énumération de l'attribut « type_TSP »	Syntaxe concrète
Voyageur	-	
Ville	-	
Localisation	-	
Déplacement élémentaire	-	
Exigence	Chaque ville est visitée par le voyageur à partir d'une seule autre ville.	
	De chaque ville, il n'y a qu'un départ vers une autre ville.	
	Il n'y a qu'une seule tournée couvrant toutes les villes à visiter.	
	Poids total d'une tournée ne peut pas être inférieur au seuil minimum. (Prize-Collecting TSP)	
	Poids total d'une tournée ne peut pas dépasser la capacité. (Capacitated Prize-Collecting)	
	Au moins une visite par ville d'une tournée. (TSPM)	
	Recherche d'une tournée la plus courte qui passe par une seule ville dans chacun de ces clusters. (Covering Tour Problem)	
Orientation	Minimisation de la distance parcourue.	
	Minimisation de la durée totale du voyage.	

Minimisation de la durée totale du voyage et de la distance parcourue.	
Maximisation du coût total des visites. (Max TSP)	
Minimisation du coût maximum d'une tournée. (Bottleneck TSP)	
Minimisation du coût total des visites suivant un chemin Hamiltonien. (Messenger problem)	

4.2. Transition du DSML-TSP vers le DSML-HHCRSP

Pour cette nouvelle étape consacrée à la définition du DSML-HHCRSP, une partie de ce que nous venons de présenter est *de facto* réutilisable. Cette section décrit ainsi notre logique d'émancipation du DSML-TSP pour tendre vers la conception finale du DSML-HHCRSP. Nous avons choisi de mener cette nouvelle étape de conception en cultivant l'idée d'une variante de TSP qui serait un TSP clustérisé multi-voyageurs :

- (1) Il y a de nouveau une grande diversité d'organisations, constat qui influence grandement la définition d'un langage. Il faut que la connaissance (objets métiers et les relations qui les lient) véhiculée par chaque organisation soit traduite dans le modèle.
- (2) Dans une démarche de méta-modélisation, c'est la qualité du méta-modèle qui fixe le pouvoir d'expression du DSML. Cela a des conséquences sur la structuration des données pour laquelle on conservera un patron identique au DSML-TSP : celui de la prestation de service, fruit d'un rapport entre offre et demande. On pourra donc adapter l'interface d'un coordinateur pour saisir, de manière efficace et de façon exhaustive, le juste volume de données qu'il doit renseigner pour mettre en œuvre la fonction cible de coordination.
- (3) Toujours dans le même esprit, le DSML-HHCRSP est un langage adapté à la description d'une PAD fondant un tremplin pour des prises de décision. Cet ensemble de décisions ne fait sens que dans la rencontre entre offre et demande avec son lot d'exigences et d'orientations.

Techniquement, la définition du DSML-HHCRSP mobilise des notions d'héritage du méta-modèle du DSML-TSP. À titre d'exemple, nous devons définir un concept d'« **Intervenant** » dans le DSML-HHCRSP en héritant du concept de « **Voyageur** » dans DSML-TSP parce qu'il concrétise l'offre. La nature de ces deux concepts est filiale. Tous les deux sont des « humains » qui peuvent répondre à une demande de services en se déplaçant, et chose nouvelle, qui auront aussi des préférences et des tolérances que nous chercherons à modéliser. Par conséquent, un « intervenant » possède plus d'attributs propres que le « voyageur », notamment parce que la demande est une prise en charge de patients, bénéficiaires des services eux aussi avec leurs spécificités.

4.2.1. Spécification du méta modèle-HHCRSP : syntaxe abstraite et sémantique

Étant donnée l'étendue du domaine ciblé en termes de caractéristiques (cf. la section 2 du chapitre II), la présentation du métamodèle est divisée en trois parties correspondant à 3 vues déjà employées dans le DSML-TSP, par anticipation de cette extension par héritage :

- L'offre qui n'est plus cantonnée à l'intervenant seul, mais intègre une vision plus collective en incluant une description de la structure de PAD qui les coordonnera.
- La demande qui est plus fournie en ce qu'elle distingue un « bénéficiaire » et une « intervention »
- L'interaction entre l'offre et la demande dont la caractérisation devient plus fournie également, conséquence immédiate des deux points précédents.

4.2.1.1. Vue de l'offre

- **Partie « Intervenant »**

Décrivons dans cette section les concepts sur l'identification de chacun des intervenants, avec ses attributs. Le concept « **Intervenant** » est hérité de celui de « **Voyageur** » dans le DSML-TSP. Les attributs définis dans le « Voyageur » sont partiellement hérités, par exemple celui de « nom ». Comme il y a plusieurs instances de ce concept qui seront utilisées en modélisation, une structure abritant rarement un seul intervenant, nous avons besoin de l'identifier.

- (1) Il est appelé pour une intervention, un acte de soins qui tombe souvent sous le coup d'un droit d'exercer. Les notions de métier et de compétences deviennent nécessaires pour un modélisateur. Nous pouvons déclarer un premier triplet de concepts qui encadre qui il est :
 - Numéro d'identification, attributs d'état civil (prénom et nom de famille).
 - Qualification associée au métier.
 - Aptitudes possédées pour des demandes correspondant à une capacité de prise en charge bénéficiaires ayant des profils particuliers. Par exemple, dans le Tableau 7, nous fournissons une liste de telles aptitudes.

Tableau 7 : Exemple des aptitudes des intervenants

Abréviation	Désignation
Malvoyant	Capacité à prendre en charge un malvoyant.
Malentendant	Capacité à prendre en charge un malentendant.
Alzheimer	Capacité à prendre en charge des troubles du comportement perturbateurs.
Fin de vie	Capacité à prendre en charge une personne en fin de vie.
Déficiência physique	Capacité à prendre en charge une personne en situation de handicap physique.
Déficiência mentale	Capacité à prendre en charge une personne en situation de handicap mental.

- (2) Nous pouvons compléter cette description par un second volet qui décrit ce que sont les conditions de travail d'un intervenant. Car chaque intervenant a sa propre disponibilité qui dépend de son contrat avec la structure. Ce concept « **Disponibilité** » est caractérisé par :
 - La plage calendaire de disponibilité par journée avec un découpage en matinée, après-midi, soirée ou toute la journée. Chaque intervenant respecte des obligations de son contrat travail. La nature de contrat dépend du statut social de l'intervenant (employé ou libéral sous-traitant).
 - Ce concept « **Contrat** » est un condensé de toutes les règles qui touchent au temps de travail :
 - Le type de contrat, à temps plein ou à temps partiel.
 - La capacité de travail par jour exprimée en heure.

- Le taux de rémunération horaire, en euros par heure.
 - Le taux de charge consacré aux missions administratives, en pourcentage du temps de travail. Ce taux permet de déduire le nombre d'heures consacrées exclusivement aux interventions.
 - o Chaque intervenant peut bénéficier d'une coupure longue ou plusieurs coupures courtes en plusieurs tranches, ce concept « **Coupure** » est donc caractérisé par la durée de coupure.
- (3) Chaque intervenant a son profil professionnel et son rapport particulier au travail de PAD. Dès lors, le concept « **Tolérance d'intervenant** » est défini et il porte sur deux éléments :
- o La tolérance pour réaliser des heures supplémentaires de travail, en minutes. C'est une durée de dépassement du temps de travail du par l'agent à sa structure de rattachement. Chaque intervenant donne donc un seuil au-delà duquel l'intervenant n'est plus satisfait de la tournée qu'on lui propose.
 - o Le taux de satisfaction vient éclairer la notion de seuils (respectivement de temps de travail et de temps supplémentaire de travail) avec une vision psychologique (en %). Il fait référence à la tolérance d'un individu pour une variation du temps de travail en heures supplémentaires autour du seuil de tolérance. Par exemple, un intervenant travaille au maximum 8 heures par jour et tolère 10 minutes d'heures supplémentaires (sa tolérance). Au moment où il travaille 8 heures et 10 minutes, sa satisfaction est de 50 % (son taux de satisfaction). Si certains intervenants ne peuvent absolument pas accepter de travailler au-delà de la tolérance imposée, ce taux de satisfaction sera fixé à 0%. Il va de soi que l'introduction de ce type de paramètres a pour but de relaxer des limites quand un problème de coordination est sans solution, du fait d'un déficit d'offre vis-à-vis de la demande.
- (4) Certains intervenants ont des préférences relatives aux bénéficiaires placés dans leur portefeuille de tournée. Parmi le concept de « **Préférence d'intervenant** », on trouve :
- o La motivation de cette préférence. La préférence peut être due à une incompatibilité entre bénéficiaire et intervenant telle que l'allergie et une habitude de fumer.
 - o La liste des identifiants de ces bénéficiaires classés par raison d'incompatibilité.

- **Partie « Structure de PAD »**

Les structures de PAD offrent la possibilité aux bénéficiaires de se maintenir à domicile tout en bénéficiant d'une prise en charge pluridisciplinaire. Plusieurs types d'intervenants (professionnels médicaux ou paramédicaux, par exemple) œuvrent dans ces structures afin de répondre aux besoins spécifiques des bénéficiaires. Le concept de « **Structure de PAD** » apparaît comme un centre de contrôle du système de PAD.

Ce concept est caractérisé par :

- o La référence à une qualification officielle de PAD autorisant les types de services offerts, par exemple SSIAD, HAD, SPASAD et MAD.
- o Le fonctionnement obéit à des règles d'organisation. Parmi celles-ci, un attribut est destiné à modéliser les cas où une structure possède plusieurs options de définition de tournée. Si la valeur de l'attribut « *fonction* » d'une structure est égal à "Base", il s'agit

d'un cas où un lieu de référence (en général, le siège) constitue un point de ralliement. Ce sont le point de départ et le point de retour des intervenants. Lorsque ce même attribut est égal à "Labo", il s'agit d'une structure qui impose un passage par un lieu spécifique à la fin de la tournée. Par exemple, un laboratoire d'analyses où les échantillons biologiques prélevés pendant les interventions de la tournée doivent être déposés. Ce genre de fonctionnement doit se faire dans le temps de travail.

Le concept de « **Période** » concerne l'horizon de temps de planification qui est représentatif du temps d'ouverture calendaire d'une structure de PAD. Ce temps d'ouverture est découpé en périodes. Chaque période est supposée quotidienne. Comme pour l'intervenant, cette maille de temps est ensuite divisée en quatre périodes : matin, après-midi, soir et toute la journée. Chaque période est un horizon de planification indépendant pour un jour, caractérisé par un jalon événement de début et un de fin. La structure peut contrôler les périodes d'ouverture en paramétrant son choix parmi ces possibilités. L'échelle de temps la plus fine est la minute, c'est cette unité de temps qui permet de qualifier les périodes.

Ce concept peut faciliter la représentation des temps de coupure d'activité des intervenants (temps de pause ou toute particularité liée à l'organisation du travail) pendant leur service. Les coupures d'un intervenant sont très difficiles à modéliser en raison de la variabilité de leurs comportements.

La Figure 23 schématise cette vue « Offre » du métamodèle du DSML-HHCRSP.

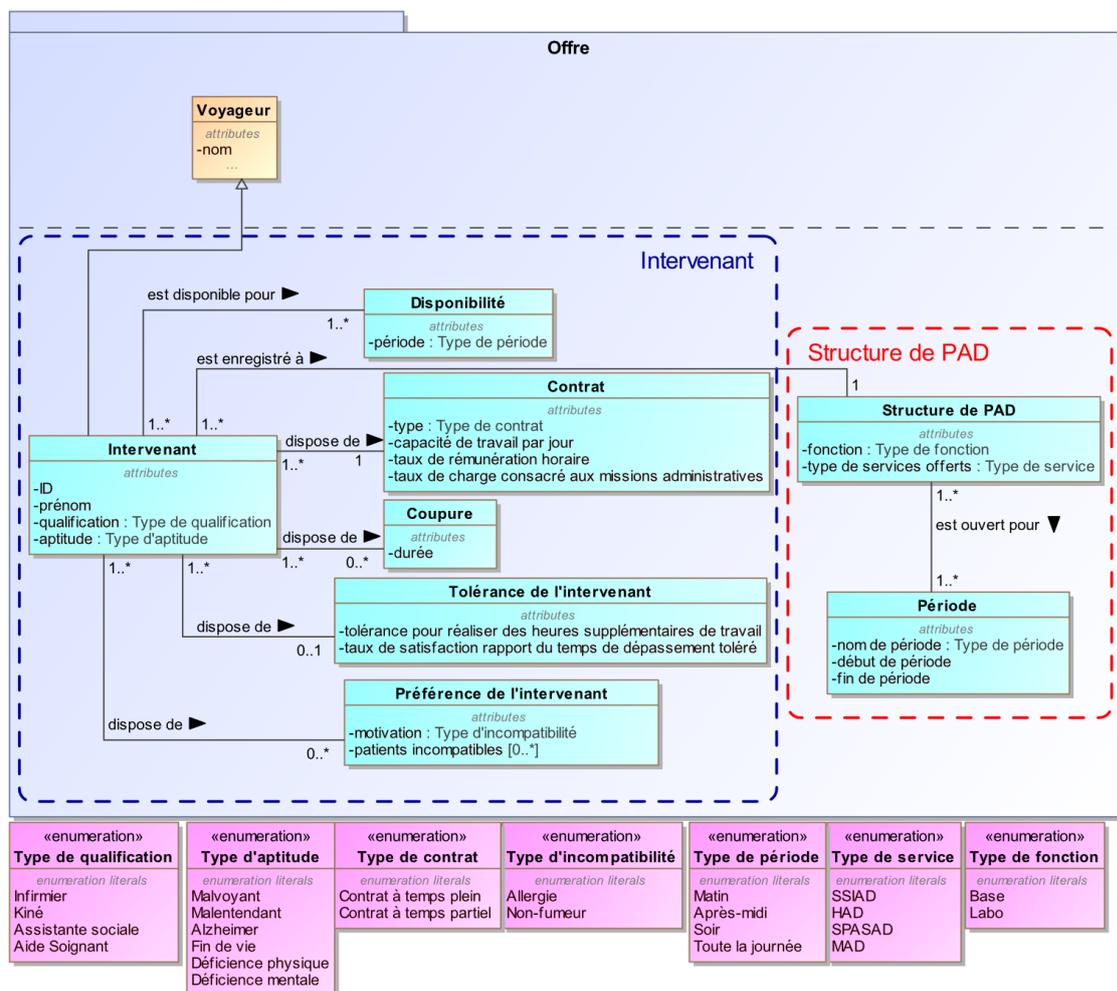


Figure 23 : Illustration du métamodèle du DSML-HHCRSP vue « offre »

4.2.1.2. Vue de la demande

La demande provient des patients qui sont des bénéficiaires (des consommateurs de soins de santé) des services au sein d'un système de PAD. Ce sont souvent des patients fragiles ou fragilisés, requérant des interventions régulières dans le cadre d'un plan personnalisé de soins, dans leur lieu de vie. Dès lors, le concept de « **Bénéficiaire** » est naturellement hérité du concept de « **Ville** » dans le DSML-TSP. Relevant tous deux de la demande. Les attributs du concept « **Bénéficiaire** » sont toutefois plus nombreux, et les attributs « nom » et « ID » sont définis à l'aide de la relation d'héritage. On peut distinguer trois classes qui le complètent. La première définit les besoins de services et prend pour nom « **Intervention** ». La seconde est symétrique d'une classe dans la partie « Offre », il s'agit des préférences (« **Préférence de bénéficiaire** ») du bénéficiaire vis-à-vis de ses intervenants. La troisième va également faire écho à une classe de l'offre puisqu'il s'agit des tolérances (« **Tolérance de bénéficiaire** »), posant un cadre pour une partie de la qualité de services.

Dressons une liste de ces concepts et de leurs attributs :

- (1) Le « **Bénéficiaire** » a une identification par son état civil auquel s'ajoutent des données personnelles des bénéficiaires (sexe, date de naissance, âge, taille, poids).
- (2) Chaque bénéficiaire demande une ou plusieurs interventions, ce concept d'« **Intervention** » est caractérisé par :
 - Le numéro d'identification (ID).
 - L'intitulé d'acte de soins et la durée d'exécution.
 - Le nombre d'intervenants requis, car certaines interventions exigent deux ou plusieurs intervenants.
 - Un indicateur de gravité : nous inspirant de l'échelle GIR graduant un niveau de perte d'autonomie d'une personne âgée, cet indicateur donne le niveau de difficulté de chaque prise en charge.
 - La qualification professionnelle requise (infirmier, aide-soignant, par exemple) et les aptitudes requises (cf. le Tableau 7).
 - La priorité d'être soigné en 3 niveaux : fort, faible ou sans priorité selon l'état de santé de bénéficiaires évalué par les coordinateurs.
 - L'inscription sur une échelle de temps. Deux modalités horaires sont possibles :
 - Par une fenêtre de temps : par exemple entre 8h-12h du matin, le bénéficiaire attend l'intervenant pour l'acte de soins prescrit. La fenêtre de temps d'une intervention est alors caractérisée par deux attributs : temps de début le plus tôt, temps de début le plus tard.
 - Par un horaire de rendez-vous, un événement temporel : pour certains bénéficiaires très exigeants sur les temps de début d'une intervention, ils peuvent demander une intervention à une heure fixée, par exemple, à 8h du matin.
- (3) Chaque bénéficiaire a sa propre tolérance en matière d'incertitude pesant sur la prestation de service, le concept de « **Tolérance du bénéficiaire** » est caractérisé par :
 - La tolérance pour les interventions commencées en dehors des plages horaires de rendez-vous demandées par le bénéficiaire, exprimée dans une unité de temps.

- Le taux de satisfaction (en %) pour un temps supplémentaire toléré (en unités de temps). Il quantifie une tolérance en cas de retard comme une sensibilité. Ce taux reflète le niveau de satisfaction du bénéficiaire.
 - Le délai inter-intervention qui est un temps minimal d'attente demandé par le bénéficiaire entre deux interventions successives. Ce délai est un facteur de confort des bénéficiaires, afin de respecter une plage de « repos » entre deux interventions successives.
 - La tolérance pour le non-respect de délai inter- intervention, seuil de temps critique de non-respect du délai inter- intervention.
 - Le taux de satisfaction (en %) pour non-respect du délai inter- intervention minimal.
- (4) Certains bénéficiaires spécifient explicitement leur préférence pour des intervenants en raison de la confiance accumulée avec le temps et/ou les habitudes de soins. Ce concept de « **Préférence du bénéficiaire** » est caractérisé par une motivation et une liste d'intervenants préférés en relation avec la raison concernée.

La Figure 24 schématise la vue « Demande » du métamodèle du DSML-HHCRSP.

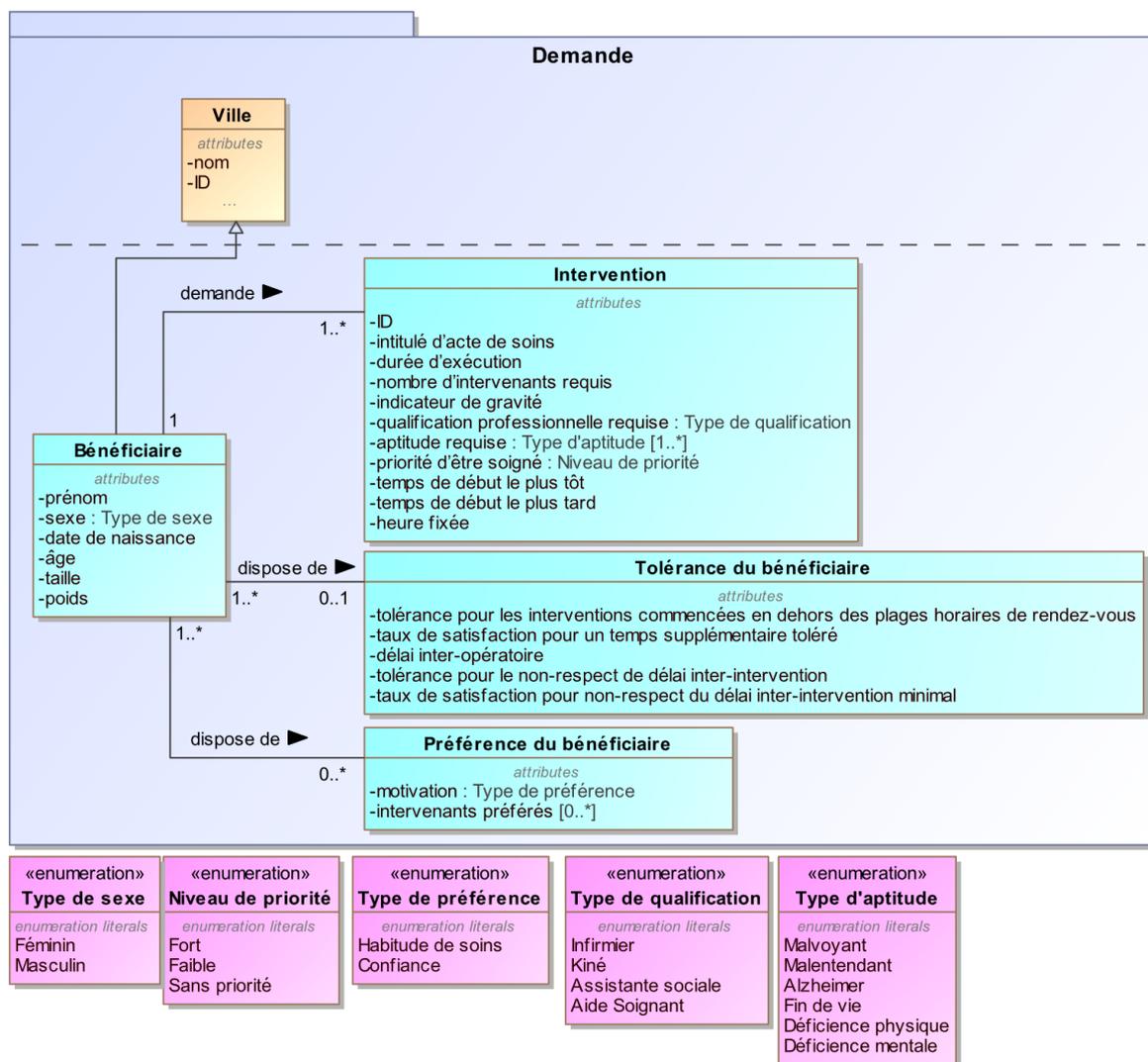


Figure 24 : Illustration du métamodèle du DSML-HHCRSP vue « demande »

4.2.1.3. Vue décisionnelle sur la coordination opérationnelle des services de PAD

Comme pour le DSML-TSP, cette vue collige les connaissances nécessaires autour de l'espace des décisions à prendre par la structure de PAD afin de remplir ses missions de coordination opérationnelle. Elle constitue un espace de médiation entre l'offre et la demande. Elle sera donc au milieu des deux autres vues. Elle réutilise ce qui a été présenté aux paragraphes qui précèdent. Nous allons faire cette présentation en trois parties. La première traite de la rencontre physique entre intervenant et bénéficiaire, c'est la localisation du service et donc le déplacement de l'intervenant. La description de toutes les exigences à considérer pour couvrir les besoins de modéliser des affectations d'intervenants et des planifications de leur tournée correspond à la partie « Exigence du HHCRSP ». Enfin, pour produire une solution dans un espace non déterministe, il faut ajouter les préférences dans la partie « Orientation des choix ».

- **Partie « Localisation »**

Les structures, les intervenants et les bénéficiaires possèdent une identité géographique qui repère leur localisation. Pour l'intervenant, ces données sont dynamiques puisqu'il se déplace. Le concept « **Localisation*** » dans le DSML-HHCRSP, hérite du concept « **Localisation** » dans le méta-modèle du DSML-TSP. Il a pour but de pourvoir cadrer des déplacements entre le lieu de la structure et des lieux de réalisation des services dus aux bénéficiaires.

Pour envisager une variante du HHCRSP pour des structures ayant une couverture territoriale large, avec un découpage en zones d'intervention, c'est-à-dire des cas où les demandes seraient clustérisées, l'attribut « *ID de cluster* » hérite du même attribut venant du concept « **Localisation** » du méta-modèle du DSML-TSP.

Concernant le concept « **Déplacement** », il est de nature relationnelle parce qu'il n'a de sens qu'en relatif à un minimum de deux localisations. Nous retrouvons les attributs basiques définis dans le concept « **Déplacement élémentaire** » du méta-modèle du DSML-TSP. Nous y définissons davantage les modes de transport pour des déplacements opérés dans un petit territoire au moyen de la voiture, du taxi, de transport en commun (métro ou bus), de vélo ou même à pied. Le coût de l'unité de distance parcourue est propre à chaque mode de transport (en euro au kilomètre, nul pour le mode pédestre) permettant aux coordinateurs de bâtir des modèles incluant des calculs de coût.

- **Partie « Exigence du HHCRSP »**

Ces exigences métier seront divisées en deux catégories distinguant d'emblée les primitives pour l'affectation des ressources (les intervenants aux interventions), d'une part, et la planification des tournées, d'autre part. Elles sont donc présentées respectivement sous forme du concept d'« **Exigence d'affectation** » et d'« **Exigence de planification de tournée** » au sein du méta-modèle du DSML-HHCRSP. A ce stade de la conception, l'héritage est également dual avec le méta-modèle du DSML-TSP pour la classe parente « **Exigence** » et celle d'« **Orientation** ». Pour ces deux concepts, la variété est traitée à l'aide du même mécanisme de listes d'énumération au sein du type d'attribut intitulé « *type_HHCRSP* ».

Les exigences prises en compte dans les capacités d'expression des modèles écrits avec ce DSML-HHCRSP sont les suivantes :

Exigences métier pour la modélisation de l'affectation des ressources

- (1) Les règles de mobilisation de l'intervenant de chaque intervention :

- Toutes les interventions médicales et sociales sont réalisables avec un intervenant qui sera, par définition de la PAD, présent sur place (une intervention ⇒ un intervenant).
 - Certaines interventions exigent plusieurs intervenants (une intervention ⇒ un ou plusieurs intervenants).
- (2) La préférence sur le prestataire de services : un bénéficiaire requérant un intervenant précis ne doit être traité que par cet intervenant dans la mesure du possible. À l'opposé, un intervenant qui ne veut pas d'un bénéficiaire dans son portefeuille d'interventions doit être entendu dans la mesure du possible. Il ne faut pas générer de situation de blocage dans le croisement bilatéral des préférences. Par exemple, un intervenant sollicité par un bénéficiaire, alors que celui-ci l'a désigné comme incompatible ou indésirable. Tenir compte des informations sur les relations humaines, par nature moins rationnelles, est un exercice difficile qui apporte certes une plus-value en termes de satisfaction et de bien-être des acteurs de la PAD, mais qui peut aussi engendrer des instabilités en matière de pilotage et de coordination sous le double constat de la versatilité des avis et de la sensibilité de ces informations qui sont parfois matinales d'implicite.
 - (3) Le respect réglementaire de la juste qualification de l'intervenant : un intervenant ne doit effectuer que des interventions pour lesquelles il est qualifié.
 - (4) Le respect de coupures légales dans le temps de travail.
 - (5) Le respect des priorités des interventions (cf. l'attribut « *priorité d'être soigné* » du concept « **Intervention** » - p.85).
 - (6) Le respect des horaires des prestations prévues et l'absence de retard en qualité de tolérance et de satisfaction. Si cette exigence est forte, les bénéficiaires ne tolèrent aucun retard par rapport aux engagements.
 - (7) Le respect de la charge de travail en rapport avec la capacité, et donc de l'appétence pour des heures supplémentaires en qualité de tolérance et de satisfaction. Si cette exigence est forte, les intervenants ne tolèrent aucune durée de dépassement de leurs capacités.

Exigences métier pour la modélisation de la planification de tournées

- (1) L'ordre cohérent de réalisation des services au regard des disponibilités des intervenants :
 - La tournée d'un intervenant est effectivement une séquence ordonnée d'interventions. Un même intervenant ne peut servir deux bénéficiaires différents en même temps (respect de la précédence des interventions).
 - Pour les interventions exigeant plusieurs intervenants, il faut synchroniser les ressources affectées à une même intervention (respect de la synchronisation des interventions).
- (2) Le respect du rythme des interventions, c'est à dire des temps de pause (le délai inter-opérateur) entre deux interventions successives : en qualité de tolérance et de satisfaction.
- (3) Le respect de sectorisation (ou zonage) quand elle existe : Si le territoire est divisé en district, les interventions relevant du chaque district sont effectuées par les intervenants éligibles dans ce district.
- (4) Le respect de la règle d'une tournée-type parmi deux options :

- Commencer le travail de chaque intervenant à partir du lieu de la structure et le terminer à cet endroit (une base).
- Commercer le travail de chaque intervenant à partir de son lieu de vie ou du lieu de la structure de PAD, et le terminer à son lieu de vie ou à autre emplacement de cette structure afin d'accomplir les tâches particulières, par exemple, la délivrance des échantillons biologiques à la fin de la tournée (multi bases).

Le Tableau 8 récapitule ces exigences d'affectation et de planification de tournée incluant les énumérations.

Tableau 8 : Liste d'énumération des exigences basée sur l'état de l'art

Affectation des ressources	Planification de tournées
(1) Règles de mobilisation de l'intervenant de chaque intervention <ul style="list-style-type: none"> ○ Une intervention ⇒ un intervenant ○ Une intervention ⇒ un ou plusieurs intervenants (2) Préférence sur le prestataire de services (3) Respect réglementaire de la juste qualification de l'intervenant (4) Respect des coupures légales (5) Respect des priorités des interventions (6) Respect des horaires des prestations prévues <ul style="list-style-type: none"> ○ Exigence forte ○ Exigence souple en qualité de tolérance et de satisfaction (7) Respect de la charge de travail en rapport avec la capacité <ul style="list-style-type: none"> ○ Exigence forte ○ Exigence souple en qualité de tolérance et de satisfaction 	(1) Ordre cohérent de réalisation des services au regard des disponibilités des intervenants <ul style="list-style-type: none"> ○ Respect de la précédence ○ Respect de la synchronisation (2) Respect du rythme des interventions <ul style="list-style-type: none"> ○ Exigence souple en qualité de tolérance et de satisfaction (3) Respect de sectorisation (4) Respect de la règle d'une tournée-type <ul style="list-style-type: none"> ○ Une base ○ Multi bases

• Partie « Orientation des choix du HHCRSP »

Comme nous venons de le spécifier, la coordination opérationnelle repose sur plusieurs exigences souvent très complémentaires, mais aussi parfois antagonistes. Une aide à la décision est donc par essence multicritère. Des préférences doivent être énoncées et modélisées pour orienter vers une solution du HHCRSP qui respecte des équilibres multiples et variés entre offre et demande.

Si au bout du bout, les buts poursuivis sont souvent économiques, d'autres critères entrent en ligne de compte. Plusieurs d'entre eux sont étudiés dans la littérature ont fait l'objet d'une présentation dans le chapitre II. Classiquement, nous citerons le temps du trajet et le coût du trajet comme des indicateurs susceptibles d'orienter vers un choix judicieux. C'est donc un héritage de l'« **Orientation** » du DSML-TSP qui donne naissance aux concepts d'« **Orientation d'affectation** » et d'« **Orientation de planification de tournées** » dans le méta-modèle du DSML-HHCRSP

Bâtissons ici une liste de ces préférences possibles qui seront autant d'indicateurs pouvant parachever la construction d'un modèle :

Orientation pour modéliser l'affectation des ressources

- (1) La satisfaction des intervenants sur le respect des heures de travail.
- (2) La satisfaction des bénéficiaires sur le respect des plages horaires demandées.

- (3) Le temps non productif des intervenants (temps administratif ou pauses).
- (4) La réalisation des services dus sur le portefeuille de PAD.
- (5) L'engagement des intervenants disponibles sur le terrain.

Orientation pour modéliser la planification des tournées

- (1) La satisfaction des délais inter-interventions demandés par les bénéficiaires.
- (2) L'équilibrage des portefeuilles de bénéficiaires entre les différents intervenants dans une optique d'équité en pénibilité du travail (l'équilibrage de difficulté des tournées).

Sur la Figure 25, nous présentons le méta modèle du DSML-HHCRSP.

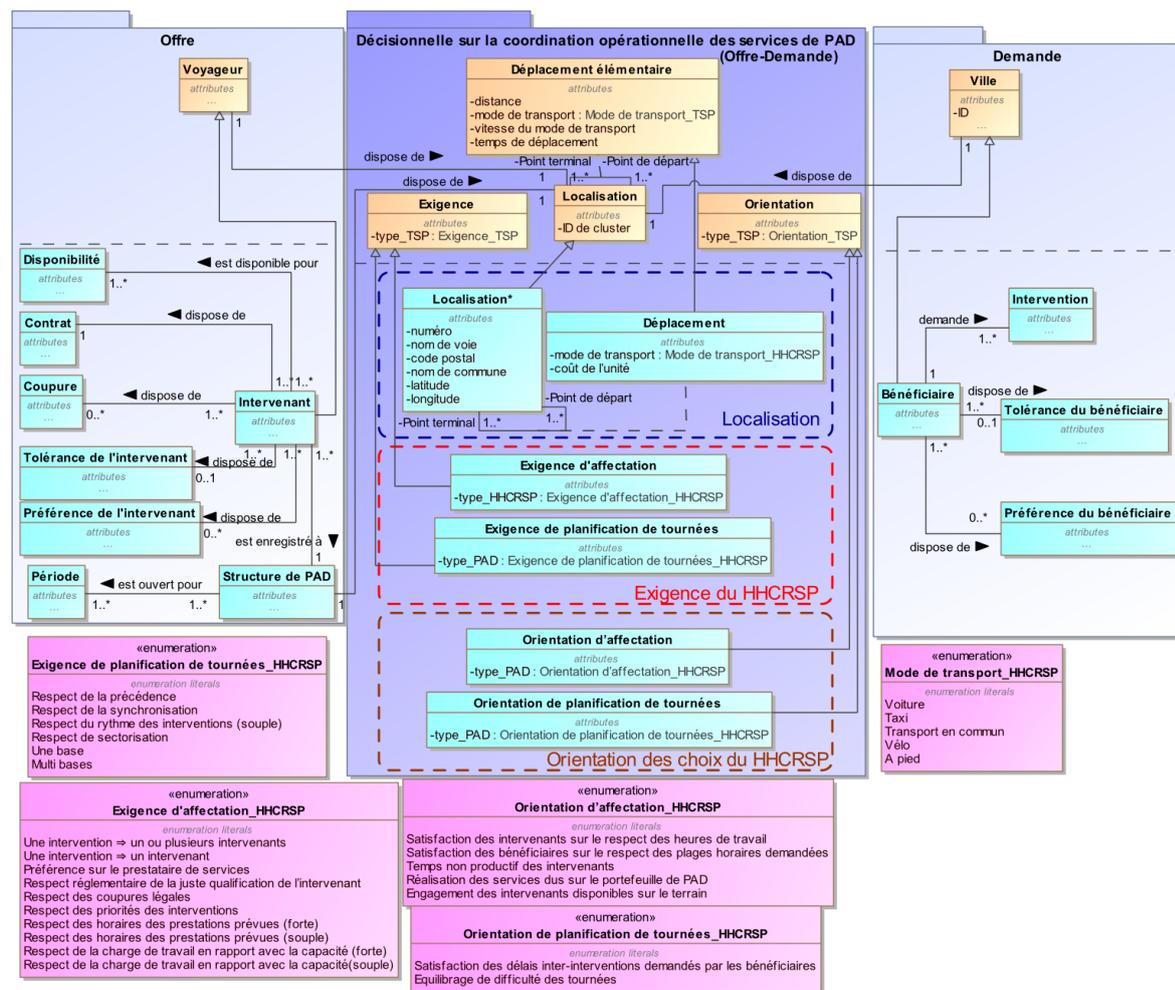


Figure 25 : Illustration du métamodèle du DSML-HHCRSP

4.2.2. Spécification de syntaxe concrète du DSML-HHCRSP

Dans cette section, une représentation graphique est construite sur la base du méta-modèle du DSML-HHCRSP. Elle est décrite dans le Tableau 9. Cette représentation synthétise les symboles choisis pour distinguer visuellement les concepts définis dans le méta-modèle du DSML-HHCRSP répartis dans des vues différentes. Notons que certaines notations déjà définies dans le DSML-TSP, par exemple celle de « Localisation », sont proches de celles du DSML-HHCRSP. Aussi, cette représentation fournit de la lisibilité et de l'expressivité du langage pour l'utilisateur, permettant d'établir un modèle qui constitue, en soit, un capital de connaissances. L'intérêt pour les coordinateurs est de pouvoir paramétrer à moindre effort leurs problèmes de coordination opérationnelle (les données nécessaires caractérisant chaque

concept, les exigences et les orientations spécifiques) et ainsi personnaliser leur besoin de pilotage.

Tout comme pour le DSML-TSP, il y a des primitives dans le DSML-HHCRSP qui sont des attributs du type énumération. Ce n'est pas l'attribut qui a une syntaxe concrète, mais chaque élément de la liste d'énumération. Mais à la différence du DSML-TSP, des concepts de ce DSML-HHCRSP peuvent avoir plusieurs attributs d'énumération. Nous introduisons deux règles pour déterminer l'emploi du langage dans ce cas de figure :

- Règle de l'attribut principal : comme dans DSML-TSP, la représentation graphique du concept est déterminée par l'élément sélectionné de la liste d'énumération. Pour une seule exception, le concept « **Intervention** », nous avons gardé une notation unique relative à ce concept.

Généralement, il n'y a qu'un autre attribut à ajouter, dit secondaire. Seul le concept « **Intervention** » fait exception à cela. En effet, il possède trois attributs de type énumération (qualification requise, aptitudes requises, niveau de priorité). Dès lors, nous avons gardé une notation unique relative à ce concept avec le choix d'un attribut principal et de deux des trois attributs d'énumération qui sont déclarés comme des attributs secondaires.

- Règle de l'attribut secondaire : L'élément choisi dans ces attributs secondaires complète la représentation visuelle issue de l'application de la règle principale. Techniquement, le symbole principal est annoté par un ou plusieurs autres petits « symboles », en exposant. Cela spécialise l'objet. Généralement, il n'y a qu'un attribut secondaire d'un concept à l'exception de l'« **Intervention** », il possède trois attributs secondaires de type énumération (qualification requise, aptitudes requises, niveau de priorité). Dès lors, les trois symboles relatifs aux attributs secondaires sont annotés dans l'ordre à côté de la notation principale et unique. Prenons un exemple de concept « **Intervenant** » qui fait exception. Un intervenant est qualifié par l'énoncé de sa profession, ici un personnel Infirmier, qui possède deux aptitudes. La représentation graphique est alors celle de la Figure 26 où les deux exposants sont respectivement les deux aptitudes : malvoyant et fin de vie.

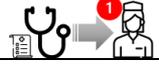
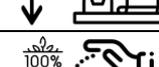


Figure 26 : Représentation graphique d'un intervenant avec la qualification « infirmier » et les aptitudes « Malvoyant » et « Fin de vie »

Tableau 9 : Représentation graphique (syntaxe concrète) des concepts inclus dans le DSML-HHCRSP

Nom de concept	Enumération				Vue			Syntaxe concrète
	Nom d'attribut	Description de l'élément	Principal	Secondaire	Offre	Demande	Offre-Demande	
Intervenant	Qualification	Infirmier	X		X			

		Kiné					
		Aide-Soignant					
		Assistante sociale					
	Aptitude	Malvoyant					
		Malentendant					
		Alzheimer			X		
		Fin de vie					
		Déficiência physique					
Déficiência mentale							
Disponibilité	Période	Matin					
		Après midi	X		X		
		Soir					
		Toute la journée					
Contrat	Type	Contrat à temps plein	X		X		
		Contrat à temps partiel					
Coupure	-	-	-	-	X		
Tolérance de l'intervenant	-	-	-	-	X		
Préférence de l'intervenant	Raison	Allergie	X		X		
		Non-fumeur					
Structure de PAD	Fonction	Base	X				
		Labo (pour la structure de PAD)			X		
	Type de services offerts	SSIAD, HAD, SPASAD, MAD		X			Annoté avec du texte à côté de la notation principale
Bénéficiaire	Genre	Féminin	X			X	
		Masculin					
Intervention	Qualification requise	Identique à <i>qualification</i> d'« Intervenant »		X			La notation du « niveau de priorité » est annotée avec du texte à côté de la notation principale :
	Aptitudes requises	Identique à <i>aptitude</i> d'« Intervenant »		X		X	
	Niveau de priorité	Fort, Faible, Sans priorité.		X			
Tolérance du bénéficiaire	-	-	-	-		X	

Préférence du bénéficiaire	Raison	Habitude de soins	X			X									
		Confiance													
Exigence d'affectation	Type_ HHCRSP	Une intervention ⇒ un intervenant	X			X									
		Une intervention ⇒ un ou plusieurs intervenants.													
		Préférence sur le prestataire de services													
		Respect réglementaire de la juste qualification de l'intervenant													
		Respect des coupures légales													
		Respect des priorités des interventions													
		Respect des horaires des prestations prévues (forte)													
		Respect des horaires des prestations prévues (souple)													
		Respect de la charge de travail en rapport avec la capacité (forte)													
		Respect de la charge de travail en rapport avec la capacité (souple)													
		Exigence de planification de tournées						Type_ HHCRSP	Respect de la précédence	X			X		
									Respect de la synchronisation						
Respect du rythme des interventions (souple)															
Respect de sectorisation															
Une base															
Multi bases															
Orientation d'affectation	Type_ HHCRSP	Satisfaction des intervenants sur le respect des heures de travail	X			X									
		Satisfaction des bénéficiaires sur le respect des plages horaires													
		Temps non productif des intervenants													
		Réalisation des services dus sur le portefeuille de PAD													
		Engagement des intervenants disponibles sur le terrain													

Orientation de planification de tournées	Type_HHCRSP	Satisfaction des délais inter-opérateurs demandés par les bénéficiaires	X				X	
		Equilibrage de difficulté des tournées						

Les relations simples que sont les caractéristiques des bénéficiaires et des intervenants sont matérialisées par des flèches simples annotées avec une sémantique appropriée. Le Tableau 10 illustre cette représentation graphique des concepts relationnels du DSML-HHCRSP. Concernant la représentation graphique du concept relationnel, nous renforçons la représentation du « **Déplacement élémentaire** » du DSML-TSP (intitulé « **Déplacement** » dans le méta-modèle du DSML-HHCRSP) en ajoutant un symbole sur la flèche du « *mode de transport* », combiné avec la distance à parcourir, en kilomètres. Cet attribut est classé attribut principal.

Tableau 10 : Représentation graphique (syntaxe concrète) des relations du DSML-HHCRSP

Nom de concept relationnel	Concepts reliés		Enumération		Syntaxe concrète
	Concept au point de départ	Concept au point d'arrivée	Nom d'attribut	Description de l'élément	
Relation classique	Intervenant	Disponibilité	-	-	→ Sémantique décrit en M2
	Intervenant	Contrat	-	-	
	Intervenant	Coupure	-	-	
	Intervenant	Tolérance d'intervenant	-	-	
	Intervenant	Préférence d'intervenant	-	-	
	Intervenant	Période	-	-	
	Bénéficiaire	Intervention	-	-	
	Bénéficiaire	Tolérance de bénéficiaire	-	-	
	Bénéficiaire	Préférence de bénéficiaire	-	-	
Déplacement	Localisation*	Localisation*	Mode de transport	Voiture	50 →
				Taxi	50 →
				Transport en commun	50 →
				Vélo	50 →
				A pied	50 →

5. Déploiement du DSML-HHCRSP

Dans cette section, nous décrivons d'abord la méthode de modélisation avec le DSML-TSP, puis avec le DSML-HHCRSP, tous deux implémentés au sein de l'environnement de méta-modélisation ADOxx. Nous expliquons les principales fonctionnalités d'ADOxx utilisées pour développer ces DSML. Ensuite, nous présentons une application pour chacun des deux DSML, produisant un modèle de son cas d'utilisation. Enfin, nous illustrons comment un changement du cas d'utilisation HHCRSP se traite grâce au DSML-HHCRSP.

5.1. ADOxx : environnement de méta modélisation supportant la définition du DSML

ADOxx est une plate-forme multi-utilisateur qui se compose de deux modules : *Development Toolkit* et *Modelling Toolkit* (cf. la section 1.3.4.3 du chapitre III). L'utilisation de ces modules ne requiert aucun développement informatique avec un codage dans un langage de programmation (Java ou C par exemple). Un métamodèle de DSML peut être spécifié via une interface élaborée du composant « *Development Toolkit* ». Puis, la conception de l'outil de modélisation mettant en œuvre ce DSML est confiée au deuxième composant *Modelling Toolkit*. Il permet d'éditer des modèles qui sont conformes au méta-modèle. L'utilisabilité d'ADOxx est conviviale, son apprentissage est relativement simple comme le souligne (Thabet, 2020) :

- ADOxx¹ est bien documenté permettant une compréhension aisée du fonctionnement de l'outil.
- ADOxx est flexible permettant à l'utilisateur de parvenir à ses objectifs de modélisation quels que soient les spécificités de son langage.

ADOxx fournit une base native avec une collection attrayante de DSML réutilisables qui peuvent servir pour le développement d'autres DSML. Les méta-modèles sont basés sur des classes du M²M ADOxx et décrits à l'aide d'ALL (*ADOxx Library Language*). Ce méta langage fournit donc des moyens de décrire les méta-modèles (cf. la Figure 27). Dans notre cadre, nous avons pu développer le DSML-TSP avec ALL, puis le DSML-HHCRSP a été généré comme une extension de celui-ci. La Figure 27 est une capture d'écran illustrant aussi la relation d'héritage dans *ADOxx Development Toolkit*, à savoir que la notation « `__` » + « **Voyageur** » + « `__` », « **Intervenant** » fixe la classe parent et sa classe héritière.

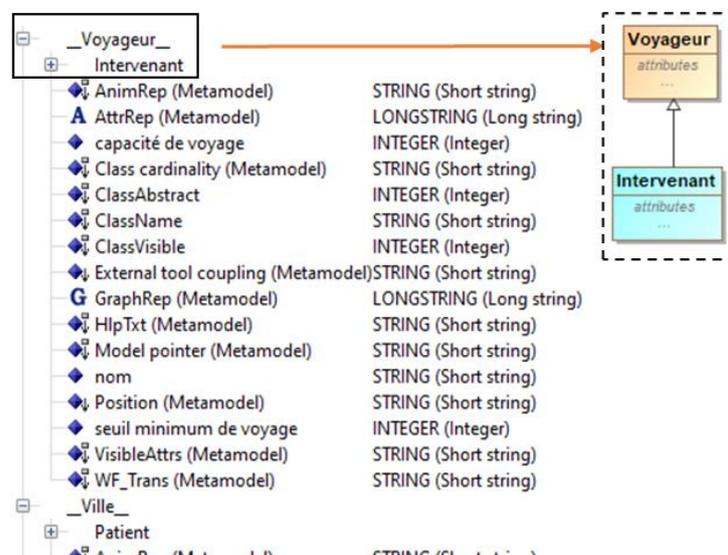


Figure 27 : Illustration d'ALL et de la relation d'« héritage » dans ADOxx « Development Toolkit »

Les modèles conçus avec un DSML sous ADOxx peuvent être exportés selon deux formats : ADL (*ADOxx Development Language*, format natif) ou en XML (format standard). Les fichiers au format ALL et ADL représentent respectivement le méta-modèle et le modèle du DSML. Cet

¹ La documentation et le forum sont disponibles sur <https://www.adoxx.org/live/home>

artifice ouvre la possibilité d'échanger des DSMLs entre utilisateurs pour les utiliser et les évaluer. C'est extrêmement pratique dans un cadre de co-conception.

La représentation graphique des éléments d'un DSML a recours à un langage nommé *GraphRep*¹. Ce dernier sert à définir les symboles associés à des classes et des classes relationnelles du méta-modèle. Dans *GraphRep*, nous pouvons aussi modifier la visualisation d'une classe en définissant des algorithmes pour les éléments dans une liste d'énumération (cf. la Figure 28). Concernant la définition des attributs dans une classe, le langage *AttrRep*² sert à la spécification de la visibilité d'attribut pour le modélisateur.

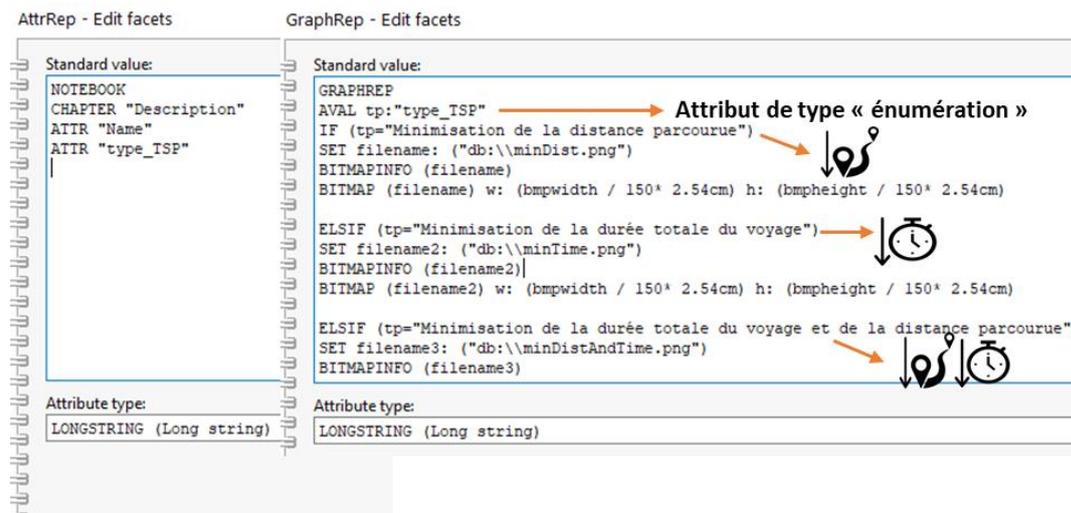


Figure 28 : Exemple du AttrRep et GraphRep dans ADOxx « Development Toolkit »

Le méta-modèle du DSML-HHCRSP, tel qu'illustré dans la Figure 25, a d'abord été décrit à l'aide d'ALL. Puis les notations graphiques de ces concepts ont obéi aux spécifications du Tableau 6 (pour le DSML-TSP), du Tableau 9 et du Tableau 10 (pour le DSML-HHCRSP). Elles ont été implémentées avec le langage *GraphRep*.

5.2. Méthode de modélisation

Nous inspirant de la terminologie en vigueur en gestion de production, nous présentons notre démarche de modélisation par une analogie à une chaîne de production industrielle. Ce choix peut apparaître un peu lourd ici, mais il sera plus compréhensible quand nous aborderons le traitement complet de l'IDM dans le chapitre VI.

Un module proposé dans notre architecture logicielle se compose de plusieurs ateliers qui assurent la production des modèles. Ce qui serait un flux de produits devient donc un flux de connaissances. Les modèles sont des artefacts d'ingénierie qui évoluent dans leur contenu au fil de la chaîne de production jusqu'à permettre de délivrer un résultat répondant à notre besoin. Dans chaque étape du processus de modélisation, la production des modèles est faite avec le DSML-HHCRSP.

¹ Présentation de GraphRep : <https://www.adoxx.org/live/graphrep>

² Présentation de AttrRep : <https://www.adoxx.org/live/attrrep>

La spécification du besoin de coordination de la PAD par un modèle dument renseigné est le résultat que nous visons. S'agissant d'une modélisation métier, nous la qualifierions de CIM dans une démarche d'IDM. Le processus d'ensemble de la modélisation (cf. la Figure 29) est composé d'activités de modélisation réparties dans trois ateliers. Chaque atelier embarque une partie de notre métamodèle du DSML-HHCRSP. L'objectif est donc de donner au décideur, expert métier (coordinateur de la structure de PAD, par exemple), la faculté de faire une spécification en ligne et à moindre effort de ses problèmes avec les bonnes données à travers de l'outil de modélisation. Son travail doit être dirigé pour que les données composites soient, autant que faire se peut, renseignées à partir des données élémentaires dans le bon ordre. Rien d'étonnant donc à ce que le processus de modélisation débute avec, en parallèle, une modélisation de l'offre et une modélisation de la demande. Sur cette base, le problème de coordination opérationnelle de la structure de PAD peut-être, à son tour, modélisée (L. Zhang et al., 2018). C'est un total de six processus que nous avons ventilés dans les ateliers, auxquels nous avons adjoint une nomenclature à deux niveaux, comme il est précisé sur la Figure 29.

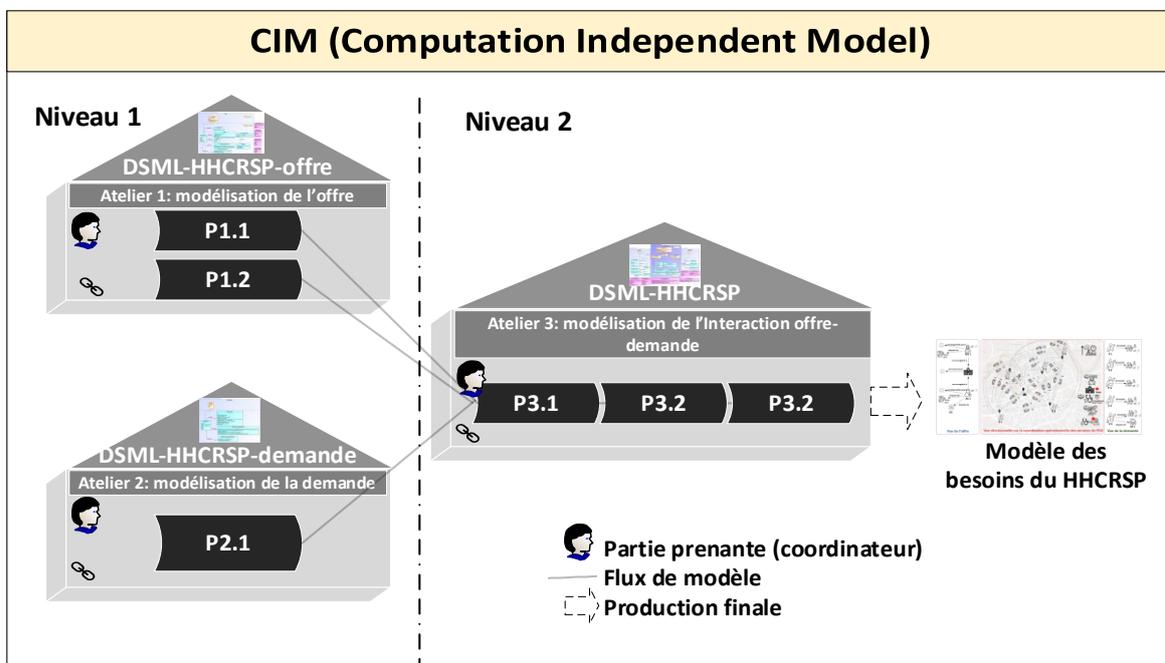


Figure 29 : Fonctionnement et composants de la méthode de modélisation

Nous proposons conceptuellement une représentation graphique de ces processus qui mobiliseront des IHM de saisie de données, sous la forme de diagramme de flux (apparentés au formalisme SADT (Marca and McGowan, 1987)) où les modèles sont générés dans une logique en correspondance totale avec les étapes de la présentation du méta-modèle du DSML-HHCRSP de ce chapitre :

- Vue de l'offre (cf. la section 4.2.1.1 de ce chapitre) ⇒ Atelier 1 : modélisation de l'offre
- Vue de la demande (cf. la section 4.2.1.2 de ce chapitre) ⇒ Atelier 2 : modélisation de la demande
- Vue décisionnelle sur la coordination opérationnelle des services de PAD (cf. la section 4.2.1.3 de ce chapitre) ⇒ Atelier 3 : modélisation de l'interaction entre offre et demande

5.2.1. Atelier 1 : modélisation de l'offre

Cet atelier traite tous les intervenants et la structure qui les regroupent. Il capture les données de tous ces intervenants et de la structure en entrée, le modèle de l'offre est à disposition du modélisateur en sortie. Deux processus parallèles sont activés, notés P1.1 et P1.2.

5.2.1.1. P1.1 : Modélisation des intervenants

Ce processus permet de décrire les données des intervenants inscrits dans la structure. Au sein de la vue conceptuelle de l'offre, il s'agit d'instancier des éléments relatifs au concept de même nom dans le méta-modèle du DSML-HHCRSP (cf. la Figure 30).

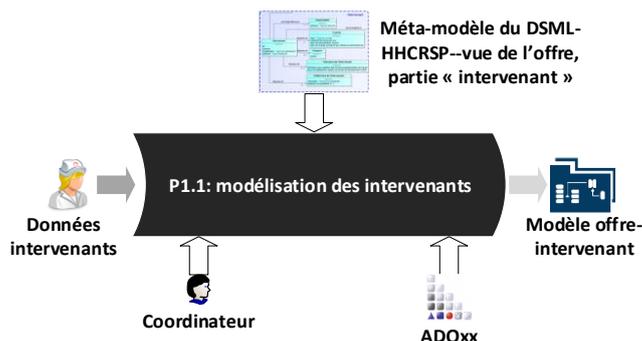


Figure 30 : Analyse SADT pour l'atelier 1 P1.1 : Modélisation des intervenants

5.2.1.2. P1.2 : Modélisation de la structure de PAD

En collectant les caractéristiques de sa réalité opérationnelle, ce processus donne naissance à l'entité de gestion dans le modèle. Le résultat est un modèle de la structure qui complète la qualification de son offre (cf. la Figure 31).

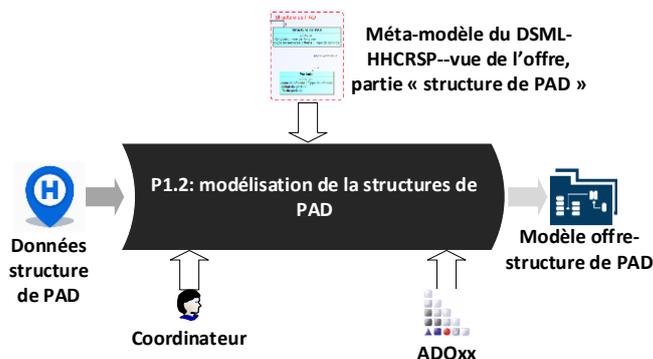


Figure 31 : Analyse SADT pour l'atelier 1 P1.2 : Modélisation de la structure de PAD

5.2.2. Atelier 2 : modélisation de la demande

Un seul processus est défini dans cet atelier.

5.2.2.1. P2.1 : Modélisation des bénéficiaires et leur demande d'interventions

Ce processus capture les données statiques de tous les bénéficiaires et les données plus dynamiques relatives à leur demande d'interventions. Cette part de la modélisation concerne certaines données qui sont dépendantes du temps et de l'espace. Un modèle de la demande est le résultat obtenu en sortie (cf. la Figure 32).

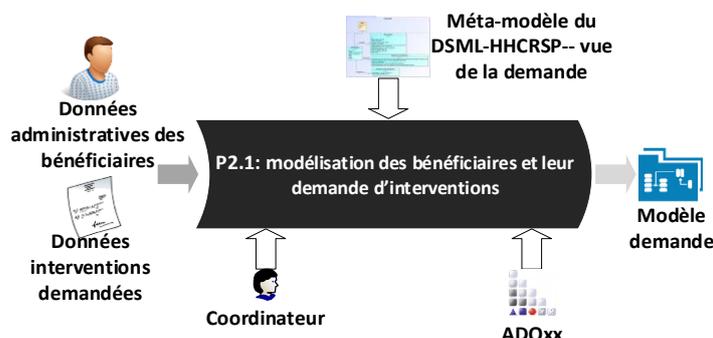


Figure 32 : Analyse SADT pour l'atelier 2 P2.1 : Modélisation des bénéficiaires et leur demande d'interventions

5.2.3. Atelier 3 : modélisation de l'interaction entre l'offre et la demande

Cet atelier aborde la modélisation du pilotage et la coordination opérationnelle par la structure de PAD à travers trois processus consécutifs, à savoir :

- Les connaissances territoriales permettent de localiser les entités de l'offre et de la demande à coordonner (P.3.1).
- Des exigences sur le fonctionnement traduisent des règles d'organisation et de gestion (P3.2).
- Des orientations souhaitées par l'organisation (les exigences fixent des enjeux sur le niveau de performance souhaité, par exemple) sont déclarées dans le modèle qui en résulte (P3.3).

Les trois processus de modélisation définis en amont de cet atelier (l'atelier 1 : P1.1, P1.2 et l'atelier 2 : P3.3) ont fourni les éléments dont cet atelier a besoin. Le modèle dit de l'offre et de la demande décrit l'espace de prise de décision. Le méta-modèle du DSML-HHCRSP en entier sert de base pour illustrer ce champ des possibles.

5.2.3.1. P3.1 : modélisation des localisations des acteurs autour de la structure

Ce processus de modélisation permet de représenter le déplacement des intervenants délivrant les services aux bénéficiaires. Ces données sont conciliées en modélisant toutes les informations géographiques des intervenants et des bénéficiaires. Il s'agit donc de géolocaliser les deux principales catégories d'acteurs impliqués de la prestation de services (bénéficiaire et intervenant) (cf. la Figure 33).

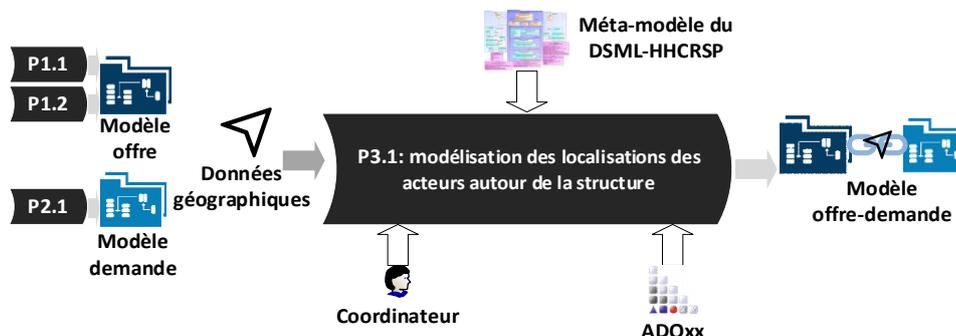


Figure 33 : Analyse SADT pour l'atelier 3 P3.1 : modélisation des localisations des acteurs autour de la structure

5.2.3.2. P3.2 : modélisation des exigences

Il s'agit ici de poser un ensemble de limites sur cet espace constitué par les modèles créés précédemment pour assurer la faisabilité de la coordination. C'est à ce stade que la notion de tournée prend corps. Le bon fonctionnement de la coordination opérationnelle est en effet limité par les nombreuses règles déjà discutées dans ce chapitre et qu'il faut respecter. Ce processus de modélisation permet aussi d'aborder une forme de personnalisation du pilotage, de la coordination sur-mesure, pour coller au fonctionnement de la structure donnée (cf. la Figure 34).

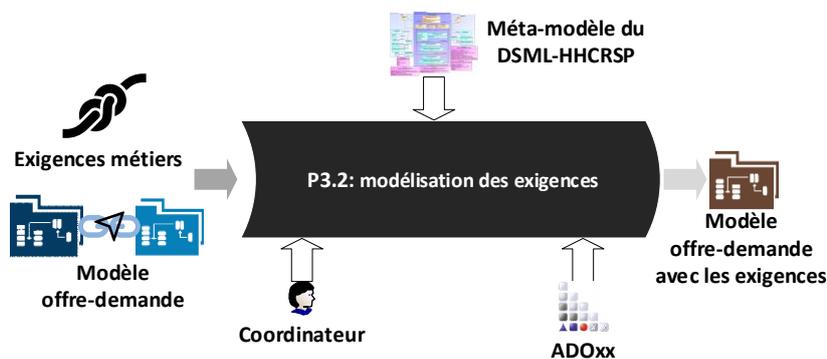


Figure 34 : Analyse SADT pour l'atelier 3 P3.2: modélisation des exigences

5.2.3.3. P3.3 : modélisation des orientations de la structure

Ce processus traduit un point essentiel du pilotage qui va fixer des attendus d'une bonne coordination. A l'entrée, le coordinateur spécifie les orientations qu'il souhaite voir appliquer lors de la définition des plannings de tournée. Par exemple, il faut modéliser les préférences en termes de satisfaction des acteurs et/ou de maîtrise du modèle économique sous-jacent. Ce modèle final est l'artéfact sortant de P3.3 et constitue le CIM de la structure de PAD en vue d'une coordination opérationnelle sur des mailles de temps et d'espace données. Il a normalement mis en scène toutes les connaissances compilées dans le méta-modèle du DSML-HHCRSP et a donc fait un usage intensif du langage spécifique (cf. la Figure 35).

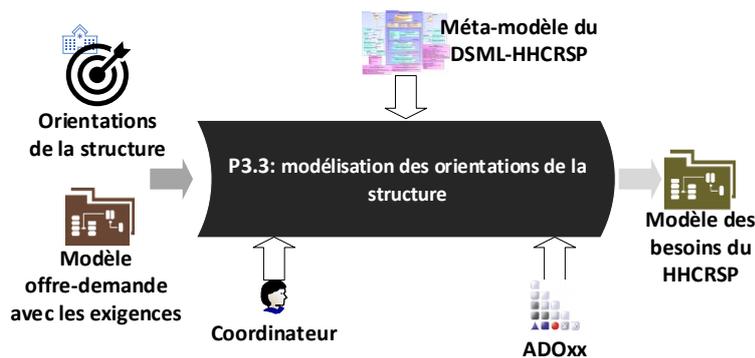


Figure 35 : Analyse SADT pour l'atelier 3 P3.3 : modélisation des orientations de la structure et l'assemblage

Cette dernière étape de la « traversée » virtuelle de l'atelier opère un assemblage de tous les modèles conçus. En ce sens, c'est un vrai exercice d'intégration.

5.3. Phase de validation

5.3.1. Modélisation d'un cas de TSP

Sur la Figure 36, nous présentons le résultat d'un modèle défini en utilisant le DSML-TSP. La toile de fond est une carte géographique du territoire français (implicitement le périmètre de chalandise du voyageur de commerce), où on peut positionner de manière réaliste les villes qui prennent la syntaxe concrète d'un symbole de type « maison » avec une sémantique claire de nom de ville (Paris, Lyon, par exemple). Ces symboles fixent la localisation des points de rendez-vous du voyageur. Les flèches qui relient les villes sont des associations de type connexion routière, et portent l'attribut de la distance. Ce sont tous les trajets a priori possibles.

Le voyageur est symbolisé par une icône de « forme humaine », avec le nom de ce voyageur qui est localisé à son point de départ de la tournée. Dans une interface animée comme on en trouve dans les simulateurs à événements discrets, cette icône serait supposée se déplacer de ville en ville en suivant les flèches pour exécuter sa tournée, et ainsi concrétiser les décisions à prendre à partir du modèle du TSP pour satisfaire les exigences en suivant les orientations. La finalité du modèle est bien de préparer cet espace de décision (c'est un centre de décision) pour déterminer cet ordre dans lequel l'icône humaine va se déplacer dans les quatre villes. Et par conséquent, les routes qui seront effectivement empruntées.

Des éléments graphiques sont disposés à droite de la Figure 36 dans un volet dont la partie supérieure traite des orientations, alors que la partie inférieure traite des exigences. L'objet en tête de liste précise donc ici qu'il faut trouver une route entre chaque paire de villes avec une distance minimale à parcourir. Les deux suivants expliquent qu'il s'agit d'un flux traversant une et une seule fois toute ville. Le dernier explique qu'il ne faut tolérer que les solutions avec une et une seule tournée. Ces objets forment le corps explicite du centre de décision (exigences et orientations).

Ce modèle créé avec *ADOxx Modelling Toolkit* fournit aussi une interface avec des fenêtres dynamiques (visible sur la figure 36 et la Figure 38 sous forme de *Notebook definition*, sous l'aspect d'un formulaire) qui permettent de renseigner les données que sont les valeurs des attributs définis au sein de chaque concept.

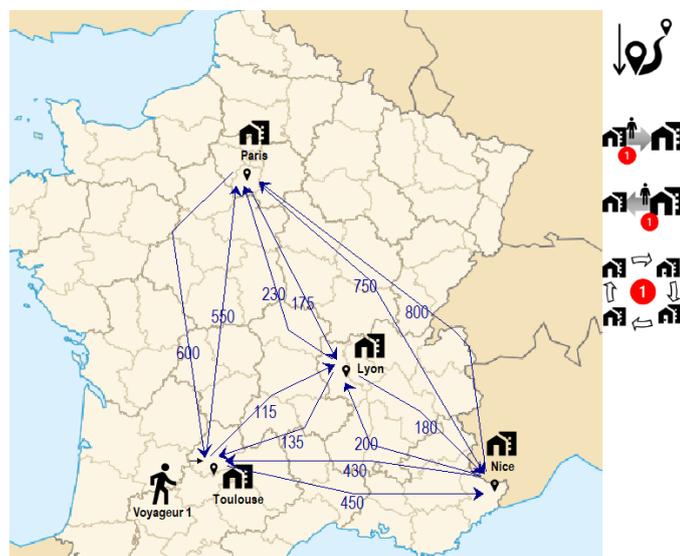


Figure 36 : Modèle de besoin du TSP : 4 villes à visiter avec 3 exigences et 1 orientation

Pour illustrer sur cet exemple de modèle ce qui va changer si le problème TSP est changé, nous proposons de modifier le mode de transport employé pour chaque trajet. Nous avons mixé les modes de transport en choisissant l'avion pour les chemins entre les villes les plus éloignées, c'est-à-dire entre Paris et les villes du sud, et le train pour les autres. Sur la Figure 37, l'icône de l'orientation a changé. De plus, l'exigence de la tournée unique a été remplacé par « au moins une visite par ville d'une tournée ». Ainsi, le TSP classique illustré dans la Figure 36 se transforme en un TSPM (Gutin and Punnen, 2006).

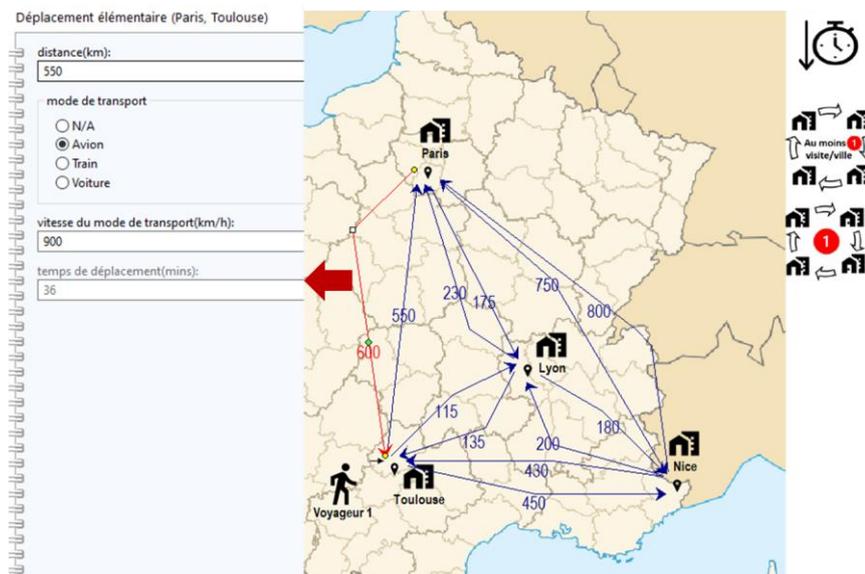


Figure 37 : Modèle de besoin du TSPM (TSP with multiple visits)

5.3.2. Modélisation d'un cas de HHCRSP

Afin de mettre à l'épreuve notre prototype de DSML-HHCRSP, nous proposons un second cas d'études. Notons que les informations explicites caractérisant chaque concept sont saisies par les coordinateurs au moyen d'une interface *Notebook definition*.

- Vue de l'offre

Sur la Figure 38, le cas d'usage est ciblé dans un territoire urbain, la ville d'Albi. Une structure de PAD a son siège au centre-ville, elle est considérée comme la base. Tout intervenant doit débiter sa tournée et revenir à la structure à la fin de sa tournée. Cette règle est une exigence confirmée par l'icône disposée dans la partie droite basse du modèle. Cet intervenant est disponible toute la journée sur la période d'ouverture de la structure, ce qui est signifié par un symbole « soleil transparent ». Aussi, cet intervenant qui a un contrat à temps plein avec la structure a une capacité de travail maximum par jour de 10h, son salaire par heure est de 11€/heure brut, et le taux de charge de ses missions administratives est nul. Il est qualifié infirmier () avec les aptitudes « malvoyant » () et « malentendant » () .

- Vue de la demande

Les quatre bénéficiaires sont deux femmes et deux hommes, et demandent chacun une seule intervention. Les informations explicites caractérisant chaque intervention sont saisies par les coordinateurs. Les notations graphiques dites secondaires de chaque intervention représentent ici la qualification () et les aptitudes « malvoyant » () et « malentendant » () exigées en vue de les réaliser avec succès.

- Vue décisionnelle sur la coordination opérationnelle des services de PAD

D'abord, les lieux de vie de chaque bénéficiaire sont présentés par un symbole « localisation » similaire au cas du TSP. Les flèches qui relient les localisations sont des associations de type connexion routière et portent l'attribut de la distance, complété par le moyen de transport symbolisé par une voiture ou un homme qui se déplace à pied.

Les symboles relatifs au centre de décision sont disposés dans la partie droite. Les deux premiers sont destinés aux orientations : la satisfaction des intervenants sur le respect des charges de travail maximum exigés et la satisfaction des bénéficiaires sur le respect des plages horaires de leurs demandes. Les trois suivants sont les exigences métier à respecter. Au niveau de l'affectation, la qualification et les aptitudes de l'intervenant doivent répondre aux qualifications et aptitudes requises pour chacune des quatre interventions. Au niveau de la planification de tournée, il faut respecter la règle du point de départ et d'arrivée vis-à-vis de la base, la structure. Par ailleurs, les quatre interventions devront suivre une séquence stricte d'exécution, le chevauchement entre interventions lors la tournée est donc strictement interdit.

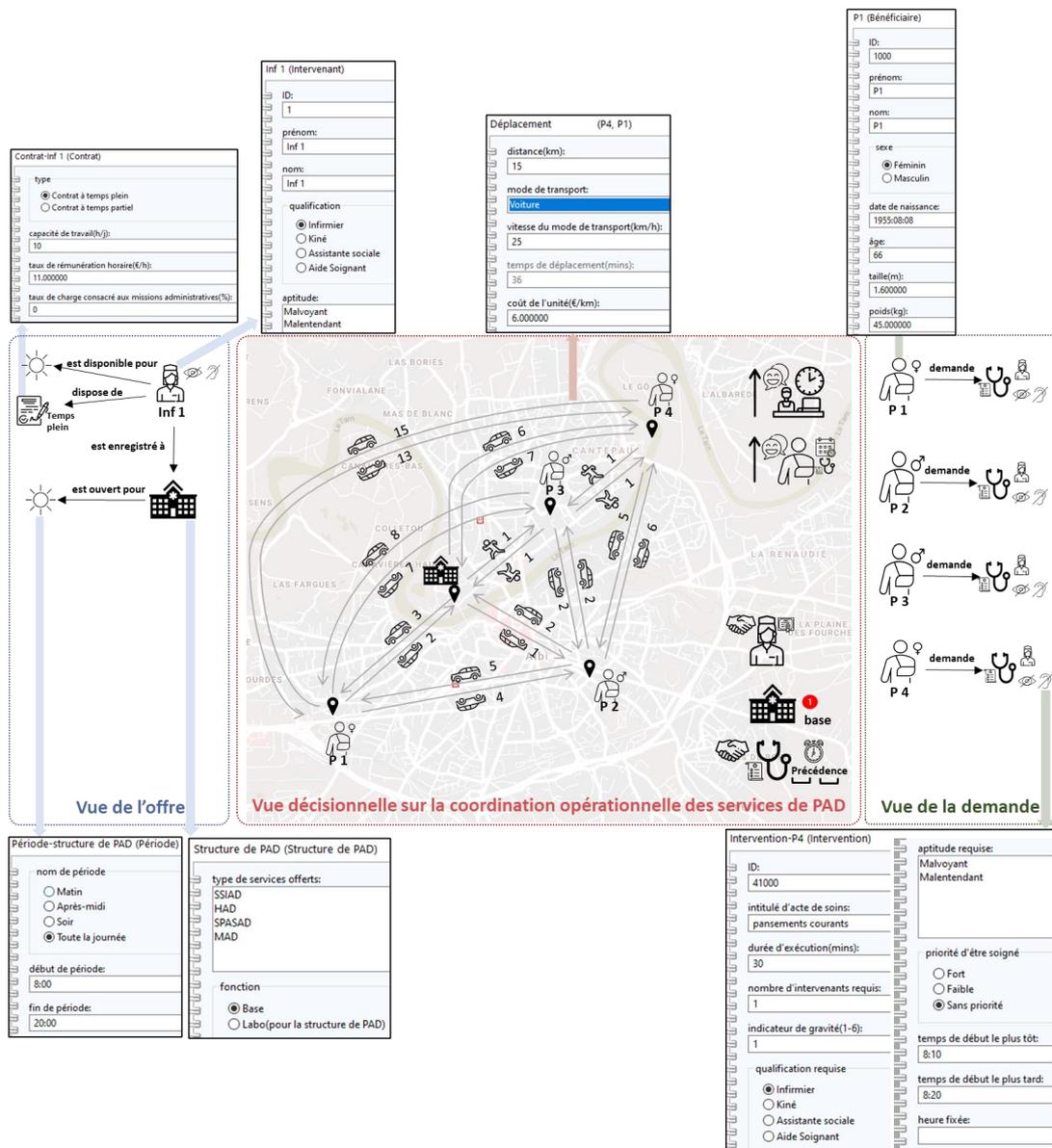


Figure 38 : Modèle de besoin du HHCRSP avec l'illustration des interfaces « Notebook définition » dans ADOxx : 4 interventions à réaliser par 1 intervenant, avec 3 exigences et 2 orientations

Revenons sur l'idée d'introduire, comme pour le TSP, une variante dans ce cas pour l'offre, la demande et la prise de décision.

- Vue de l'offre

Rendons le problème différent en incluant un second intervenant. Sur la Figure 39, cet autre intervenant est enregistré avec la qualification d'« aide-soignant ». Il possède les mêmes aptitudes et disponibilité que son collègue infirmier. Il déteste le fait que le bénéficiaire ait l'habitude de fumer en spécifiant sa préférence du type non-fumeur (symbolisé par l'icône ) avec le nom de bénéficiaire incriminé (ici "P1").

- Vue de la demande

Un nouveau bénéficiaire masculin est enregistré qui demande une intervention avec des exigences identiques aux autres (les informations basiques de prescription de soins, la qualification et les aptitudes). Mais il spécifie de plus une préférence personnelle de confiance, préférant être soigné par l'infirmier (symbolisé par l'icône ) et complété par le nom de l'intervenant préféré en dessous : "AS 1").

- Vue décisionnelle sur la coordination opérationnelle des services de PAD

La satisfaction sur le respect des plages horaires des bénéficiaires a été supprimée. Pour envisager les préférences personnelles des deux nouveaux acteurs, deux exigences d'affectation sont ajoutées permettant d'assurer la concordance au sein de la relation préférentielle intervenant-bénéficiaire (symbolisé par ) , et chaque intervention est exécutée par un seul intervenant (symbolisé par   ).

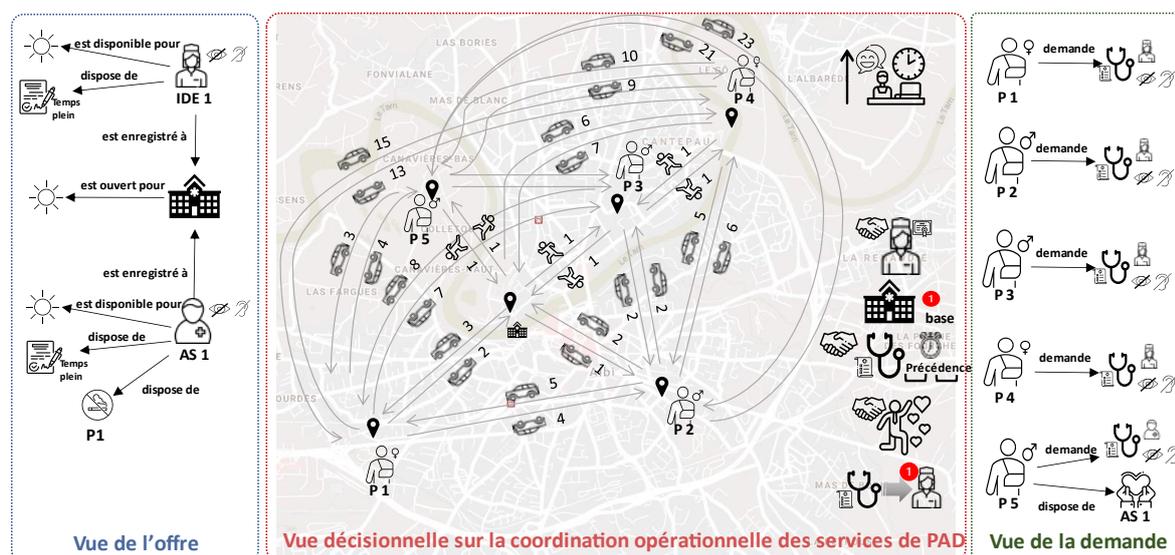


Figure 39 : Modèle de besoin du HHCRSP : 5 interventions à répondre par 2 intervenant, avec 5 exigences et 1 orientation

Tirons quelques enseignements de ces études de cas :

- (1) Ces modèles constituent une base solide de connaissances pour caractériser et améliorer le fonctionnement de la coordination dans la prise en charge à domicile, dont

l'acquisition est canalisée par un processus qui est adapté. La collecte des données est mise en perspective dans le but de développement d'application logicielle déroulant de manière structurée les processus qui donneront naissance aux modèles.

- (2) Le développement de ce modèle HHCRSP constitue un bon moyen de fournir de bonnes pratiques sur les problèmes de coordination opérationnelle pour des structures de PAD. En effet, comprendre un langage est un saut cognitif qui fait croître la capacité d'imagination. L'usage d'un logiciel est aussi une façon d'apprendre. Souvent, ces structures ont des caractéristiques spécifiques en termes de diversité des profils d'intervenants enregistrés, de portefeuilles de bénéficiaires, de variété des contraintes métier à respecter, de préférence de performance à atteindre. Pour tout cela, le DSML a un vrai pouvoir d'expression. Le DSML est par ailleurs extensible dans ce cadre de méta modélisation et dans l'environnement ADOxx.
- (3) La définition du DSML-HHCRSP a été basée sur le capital des connaissances en coordination opérationnelle de PAD issu la phase de l'état de l'art (cf. le chapitre II) et aussi sur le constat des pratiques des clients de Berger-Levrault, et par ce croisement, nous espérons avoir abordé une majorité des besoins de structures de PAD.
- (4) La conception de ces modèles n'est qu'un point d'entrée puisque leur finalité est d'aider à la décision. L'atelier présenté ci-dessus n'est qu'une première étape de l'IDM, le traitement du niveau CIM. C'est une phase initiale importante qui définit maintenant plus concrètement notre contexte d'étude.

6. Conclusion

La description de problèmes de coordination opérationnelle est traitée par la conception d'un langage de modélisation (DSML-HHCRSP) très structuré. Nous avons posé ses bases selon une analogie avec une chaîne de transformation de connaissances. Tirant bénéfice des travaux sur l'état de l'art du Chapitre II, nous avons étudié la théorie en montant en compétences sur le besoin de données pour formuler des représentations de la coordination opérationnelle de structure de PAD, qui étant consignées dans des modèles, seront à la source des calculs de plannings de tournée. Les connaissances requises pour une description exhaustive de ces structures et des besoins de l'aide à la décision ont été recensées et structurées selon une démarche de méta modélisation avec la production et la mise à l'épreuve d'un DSML.

Face à la complexité du sujet, et avec l'ambition de l'affronter pour couvrir le plus grand nombre de structures de PAD, nous avons donc imaginé une « usine » dont trois ateliers auraient en charge la préparation de la matière première pour l'aide à la décision. Sa présentation est symbolique d'une description fonctionnelle du début d'une chaîne d'ingénierie dirigée par les modèles. Ces ateliers de niveau CIM étaient l'objet de ce chapitre. Le passage par un méta-modèle où les choix d'une vision par l'offre et la demande sont des choix cruciaux, s'est révélé très solide à l'usage. Il ouvre de nouvelles portes que nous allons emprunter dans ce qui suit.

La prochaine phase est une phase complémentaire de conception des ateliers en aval du CIM, ceux de PIM et du PSM. Il nous faut donc embrasser un ensemble de formulation de problèmes de coordination à partir des modèles. Pour ce faire, nous devons revenir vers la Recherche Opérationnelle. Nous le ferons en gardant à l'esprit le but de pouvoir faire des variations de problèmes à moindre effort, mais aussi d'être couvrant, c'est-à-dire d'embarquer un haut niveau de détail des besoins dans notre traitement du HHCRSP.

CHAPITRE V. AIDE A LA DECISION POUR LA COORDINATION CENTRALISEE ET OPERATIONNELLE DE PAD : FORMULATION ET RESOLUTION DU PROBLEME D'OPTIMISATION

CHAPITRE V. AIDE A LA DECISION POUR LA COORDINATION CENTRALISEE ET OPERATIONNELLE DE PAD : FORMULATION ET RESOLUTION DU PROBLEME D'OPTIMISATION.....	106
1. INTRODUCTION.....	107
2. DESCRIPTION DU PROBLEME D'OPTIMISATION	108
3. METHODE EXACTE BASEE SUR LA PROGRAMMATION LINEAIRE EN NOMBRES MIXTES (PLNM)	110
3.1. PIM : modèle en Programmation Linéaire en Nombres Mixtes (PLNM).....	110
3.1.1. Modélisation orientée graphe	110
3.2. Description du modèle en Programmation Linéaire en Nombres Mixtes (PLNM)	111
3.2.1. Jeu des paramètres.....	111
3.2.2. Jeu des variables de décision.....	113
3.2.3. Ecriture de la fonction objectif.....	113
3.2.4. Contraintes	114
3.3. PSM : modèle codé en OPL pour l'appel à CPLEX.....	117
4. METHODE APPROCHEE BASEE SUR LA SATISFACTION DE CONTRAINTES (CSP).....	119
4.1. PIM : modèle en satisfaction de contrainte (CSP)	119
4.2. PSM : modèle en satisfaction de contrainte orienté OptaPlanner.....	120
4.2.1. Description d'OptaPlanner.....	120
4.2.1.1. Principes de modélisation d'un CSP avec OptaPlanner	121
4.2.1.2. Spécification des contraintes du problème	124
4.2.1.3. Résolution du problème.....	125
4.2.2. Modélisation du problème	126
4.2.2.1. Classe d'information du problème (Problem Fact)	126
4.2.2.2. Classes d'entité de planification (Planning Entity, @PlanningEntity).....	127
4.2.3. Spécification des contraintes	128
4.2.4. Résolution du problème.....	132
5. EXPERIMENTATION : METHODE EXACTE VS METHODE APPROCHEE.....	134
5.1. Génération de jeux de test.....	134
5.2. Analyse comparative de la performance globale de la méthode exacte et de la méthode approchée ...	136
5.3. Analyse comparative sur le cas ML_s55_p28_mp14_k4	138
5.3.1. Satisfaction de la planification sur la fenêtre de temps	139
5.3.2. Respect du délai entre interventions	141
5.3.3. Respect de la capacité disponible	142
5.3.4. Equilibrage de pénibilité des tournées	143
6. CONCLUSION	143

1. Introduction

Dans ce chapitre, notre objectif est de répondre aux questions de recherche suivantes :

Q2 : Comment formuler mathématiquement des problèmes de coordination à partir des connaissances métiers ?

Q2-1 : Comment enrichir l'espace de formulation de problèmes de coordination avec des formulations mathématiques variées et dans des langages adaptés ?

Q2-2 : Comment orienter ces formulations pour être compatible avec les outils de calcul de solution, puis publier ces solutions et les agrémenter d'indicateurs permettant de juger leurs qualités ?

L'utilisation du langage spécifique, le DSML-HHCRSP, faite dans le chapitre IV permet au coordinateur d'une structure de PAD de générer des modèles à partir d'une expression codée de son besoin, y compris lorsqu'il s'agit de modifier le besoin dans un contexte très dynamique. **Dans ce chapitre, nous nous concentrons sur les étapes qui valorisent cette connaissance du besoin : la formulation et la résolution du problème formulé à partir de ce modèle.** Ces deux étapes sont celles qui mobilisent les savoirs et savoir-faire d'experts et chercheurs en aide à la décision (cf. chapitre II). Il faut proposer des services numériques encapsulant des techniques de recherche opérationnelle, appelés optimiseurs ou solveurs. L'objectif est donc de traiter de tels problèmes d'aide à la décision

Notre étude comporte deux volets. Dans le premier volet, nous utilisons une méthode exacte, basée sur une formulation en Programmation Linéaire en Nombres Mixtes (PLNM). Cette formulation est transcrite en langage OPL (PSM) afin d'être soumise à un solveur commercial « CPLEX » (logiciel conçu et diffusé par la société IBM). Dans un second volet, nous modifions la formulation du problème pour le traiter selon une méthode de satisfaction de contrainte (CSP). OptaPlanner est le solveur sur lequel nous avons déployé la formulation afin de tester plusieurs méthodes à base de méta-heuristiques, dites méthodes approchées. L'implémentation d'une méthode exacte et de méthodes approchées permet de mener des expériences avec notre environnement de test afin de pouvoir réaliser des comparaisons. Ceci est en particulier motivé par le besoin de juger la performance en termes de temps de calcul et de taille de cas d'études issus pour certains de données fournis par des clients de Berger-Levrault.

Dans la Figure 40, nous positionnons ces deux méthodes de résolution sur la partie droite de l'architecture dirigée par les modèles.

Le chapitre est organisé de la manière suivante. Avec la section 2 de ce chapitre, nous débutons la description générale du problème sous forme littérale en énumérant les hypothèses retenues et les contraintes traduisant les exigences du coordinateur. Dans la section 3 de ce chapitre, nous présentons d'abord la méthode exacte basée sur une formulation PLNE qui sera au solveur CPLEX pour déterminer la solution exacte des cas d'études. Dans la section 4 de ce chapitre, nous décrivons la génération de solutions approchées avec une formulation adaptée au solveur OptaPlanner. Nous décrivons ainsi de manière assez détaillée la démarche de formulation en CSP et les pratiques liées à la résolution que nous préconisons de suivre avec OptaPlanner. Dans la section 5 de ce chapitre, nous avons effectué une expérimentation basée sur 13 cas d'utilisation, selon une gradation en taille et difficulté. Les cas les plus difficiles sont ces cas réels venant du terrain et proposés par Berger-Levrault. A ce stade, nous avons mené une analyse

critique des solutions générées par la méthode exacte et par la méthode approchée jusqu'à une certaine limite de taille. Cette expérimentation a pour but non seulement de trouver cette limite de taille du problème où la méthode exacte va devenir inefficace, mais aussi d'évaluer l'écart entre la solution exacte et une solution approchée. Dans la section 6 de ce chapitre, nous concluons les travaux de recherche présentés dans ce chapitre.

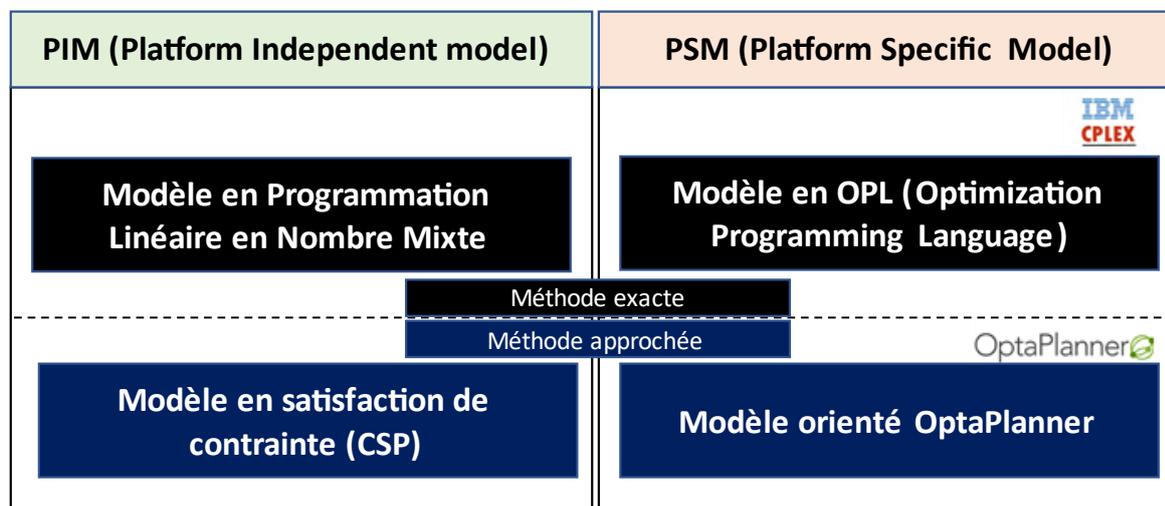


Figure 40 : Positionnement de deux méthodes de résolution dans l'architecture dirigée par les modèles

2. Description du problème d'optimisation

La cible consiste à générer une solution de plannings quotidiens de la PAD rendant les services attendus aux bénéficiaires, en décidant les tournées que doivent réaliser les intervenants actifs. L'objectif poursuivi est globalement de maximiser la satisfaction de tous les acteurs de la PAD : bénéficiaires, intervenants et structure. Une intervention définit un ensemble conjoint de services rendus à un bénéficiaire par un intervenant lors d'une seule et même visite au domicile. Le problème est conforme au DSML-HHCRSP présenté dans le chapitre IV, avec ici, un premier jeu d'**hypothèses** portant sur les données :

- Toute tournée part de, et finit à, la structure. Pour chaque tournée, nous ne tenons pas compte de la pause, ni des imprévus dus à la maladie, à un accident de trajet, aux vacances, à toute forme d'indisponibilité.
- Si l'intervenant arrive en avance chez un bénéficiaire, il doit attendre jusqu'à ce qu'à le bénéficiaire (fenêtre de temps de début d'une intervention au plus tôt) soit disponible.
- Il n'est pas possible d'ajouter un nouvel intervenant à la liste des intervenants disponibles et une intervention à la liste des demandes en cours de journée.
- Certaines données des cas d'utilisation du terrain ne sont pas accessibles en raison des obligations dues au RGPD (Règlement Européen sur la Protection des Données)¹. Cela concerne la pénibilité des interventions, la tolérance des acteurs, les différents taux de satisfaction. Pour pallier ce déficit, elles seront générées aléatoirement.

¹ RGPD : <https://eur-lex.europa.eu/legal-content/FR/TXT/?uri=CELEX:32016R0679>

- Pour les informations géographiques, les lieux de résidence des bénéficiaires sont qualifiés à partir du site « open data France ¹», nous prenons la ville d'Albi comme territoire d'expérimentations.
- Ces informations géographiques (adresse ou coordonnées GPS) des bénéficiaires constituent les données d'entrée d'une matrice d'itinéraire, dont le but est de calculer le temps de trajet nécessaire d'un lieu de résidence à un autre en distance euclidienne (à vol d'oiseau), sans tenir en compte du trafic routier, et en prenant une vitesse moyenne de déplacement de 20 km/h.
- Dans la modélisation mathématique du problème, tous les paramètres liés au temps (le temps de déplacement, la durée d'intervention, par exemple) sont chiffrés sous forme de nombres entiers et exprimés en minutes, par exemple 300 (minutes) = 6 heures du matin.
- Afin d'expliquer la relation entre l'offre et la demande compte tenu de la qualification et de l'incompatibilité, une matrice de faisabilité (bénéficiaire-intervenant) en nombres binaires est construite, agrégeant ces deux critères (cf. la section 2.2.1.1 du chapitre II).
- Chaque jour ouvrable, les bénéficiaires et leurs demandes doivent être enregistrés dans la base de données de la structure de la PAD. Chaque bénéficiaire peut solliciter plus d'une intervention par jour. Si c'est le cas, les disponibilités d'un bénéficiaire sont caractérisées par autant de fenêtres de temps pour chaque demande d'intervention.
- Chaque bénéficiaire est caractérisé par son identité, l'adresse de son lieu de vie, le délai d'inter-intervention souhaité, deux tolérances et deux taux de satisfaction respectifs, au cas où le temps d'attente ou le délai inter-intervention ne sont pas respectés. Cette tolérance temporelle personnalisée peut être également spécifiée dans les mêmes termes par les intervenants en fonction du plafond d'heures contractuelles et aussi des heures supplémentaires. Selon le constat dressé dans la section 2.2.1.2 du chapitre II (cf. le Tableau 2), aucune étude ne prendrait en compte une telle tolérance avec un taux de satisfaction personnalisé en fonction des préférences des intervenants et des bénéficiaires.
- Chaque intervention est caractérisée par une identification, une durée de traitement, un ensemble de compétences spécifiques exigeantes et un indicateur reflétant le degré de difficulté par rapport à l'opération effectuée par l'intervenant (1 représente le plus facile et 6 le plus difficile). Le début de chaque intervention est dans une fenêtre de temps d'une étendue d'une heure.

Nous complétons la définition du problème de coordination opérationnelle de PAD par une description des orientations et des exigences qui deviennent des objectifs à poursuivre et des contraintes du problème d'optimisation.

- Aucune intervention ne peut être abandonnée.
- Toutes les interventions sont réalisables avec *a minima* un intervenant présent sur place.
- Un même intervenant ne peut pas intervenir sur deux bénéficiaires différents en même temps.

¹ open data France : <https://www.data.gouv.fr/fr/>

- Un intervenant ne doit effectuer que des interventions faisables (les interventions qualifiées, compatibles entre le bénéficiaire et l'intervenant affecté).
- Les intervenants (infirmières, kinésithérapeutes, aides-soignants, etc.) sont disponibles pour prendre en charge les bénéficiaires en respectant leur charge de travail maximale.
- Chaque intervenant compatible possède les compétences nécessaires pour réaliser les interventions demandées.
- Le respect au mieux des fenêtres de temps demandées par les parties-prenantes.
- Le respect au mieux des délais inter-intervention demandés par les bénéficiaires.
- La durée d'une tournée doit respecter au mieux la tolérance de l'intervenant à ses plages de travail.
- L'ensemble des tournées devrait être équilibré en termes de difficulté.

Les règles de satisfaction des acteurs expriment une préférence qui est négociable, les exigences aussi, et elles deviennent donc des « contraintes souples » (Hiermann et al., 2015). Leur non-satisfaction ne remet pas en cause la faisabilité d'une PAD quotidienne. Mais notre objectif général est bien de maximiser la satisfaction, autant que possible. Les exigences qui doivent être respectées pour des raisons impératives de cohérence et de faisabilité d'une coordination sont évidemment considérées comme des « contraintes dures » (Hiermann et al., 2015). En cas de violation de telles contraintes, le résultat de la planification sera considéré comme non admissible, parce qu'erroné et/ou infaisable.

3. Méthode exacte basée sur la Programmation Linéaire en Nombres Mixtes (PLNM)

3.1. PIM : modèle en Programmation Linéaire en Nombres Mixtes (PLNM)

3.1.1. Modélisation orientée graphe

En recherche opérationnelle, qu'il s'agisse de problèmes de planification de tournées ou d'affectation de ressources dans un système de production de services ou de biens, la discipline fait souvent recours à une spécification du problème par un graphe simple. Soit $G = (V, E)$, un tel graphe où V est l'ensemble des nœuds et E est l'ensemble des arcs (Bondy and Murty, 1976). Nous nous inspirons des travaux de (Enríquez et al., 2015) dans lesquels les auteurs mobilisent un métamodèle décrivant un tel graphe sur le HHCRSP (cf. la Figure 41 et le Tableau 11). Ce métamodèle a vocation à préparer la formulation du problème de planification de PAD. Nous appliquons cette démarche dans la formulation de notre problème.

Tableau 11 : Concepts d'un graphe

Graph (G)	Méta-classe d'un graphe qui est composé de sommets et d'arcs.
Vertex (V)	Ensemble de sommets (nœuds) qui composent un graphe. Le sommet possède une association avec la méta-classe "Graph" où un graphe peut contenir aucun, un ou plusieurs sommets.
Edge (E)	Ensemble des arcs qui composent un graphe. Il a deux associations : l'une avec la méta-classe "Graph" car il peut contenir de zéro à plusieurs arcs. Une autre avec la méta-classe "Vertex", parce que chaque arc est constitué de deux sommets qu'ils relient. Un sommet peut être connecté à zéro, un ou plusieurs arcs.

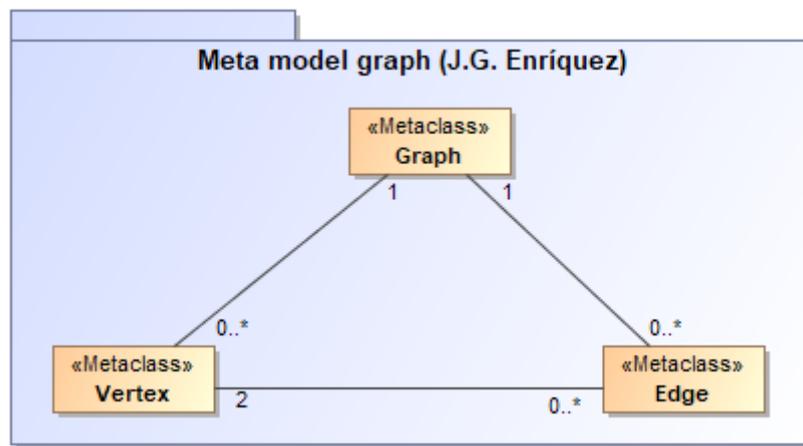


Figure 41 : Métamodèle graphe (Enríquez et al., 2015)

Selon (Fortz, 2012), quatre ensembles doivent être considérés dans une telle optique :

- Les scénarios alternatifs : variables de décision et inconnues du problème.
- Les paramètres : données connues qui précisent le problème à traiter
- Les contraintes : toute forme d'exigences considérées comme contraignantes pour qualifier une décision d'acceptable et qui, logiquement, sont des relations de contrainte dans la formulation du problème.
- L'objectif poursuivi : la raison d'être de la résolution qui cherche le scénario présentant la meilleure solution vis à vis d'une fonction traduisant cet objectif (à minimiser ou à maximiser).

Chaque élément du graphe est donc associé symboliquement à une caractéristique du problème à formuler. La volonté de traiter une prise de décision fixe une partition dans ces caractéristiques entre la connaissance d'attributs de concepts qui sont des inconnus et celles qui sont connus. C'est-à-dire que certaines données portées par le graphe sont considérées comme des variables de décision à calculer par la résolution d'un problème qui est à construire sous forme d'optimum. Les autres données sont appelées paramètres de ce problème d'optimisation. Nous avons donc un nouveau vocabulaire à maîtriser avec ces notions de variables et de paramètres. De même, nous allons considérer que la formulation du problème fondant la prise de décision est aussi un modèle. Le graphe est interprété pour exprimer des contraintes, pas forcément toutes. Dans le cas d'une affectation ou d'une tournée, les arcs jouent ce rôle de relation entre entités en associant deux sommets (intervention-intervenant, ou domicile du bénéficiaire-intervenant).

C'est donc à partir d'une modélisation orientée graphe qu'il est utile de raisonner pour dériver une forme mathématique de notre modèle d'optimisation.

3.2. Description du modèle en Programmation Linéaire en Nombres Mixtes (PLNM)

3.2.1. Jeu des paramètres

Soient :

- i, j et v désignent une intervention.
- i^*, j^* et v^* désignent un nœud.

- k désigne un intervenant.
- p désigne un bénéficiaire.
- P : l'ensemble des bénéficiaires, avec pour index $p \in [0, m]$ où m désigne le nombre de bénéficiaires.
- Y : l'ensemble des bénéficiaires qui demandent plus de 2 interventions, avec pour indice $q \in [0, l]$ où l désigne le nombre de tels bénéficiaires. Notons que $Y \subseteq P$.
- K : l'ensemble des intervenants, avec pour indice $k \in [1, n]$ où n désigne le nombre d'intervenants inscrits dans la structure de PAD pour l'horizon de coordination découpé par jour. Ils sont disponibles pour réaliser les interventions quotidiennes chez des bénéficiaires des services.
- C : l'ensemble des interventions demandées par P , avec pour indices $i, j, v \in [0, r]$ où r désigne le nombre de services de soins demandés par les m bénéficiaires.
- N : l'ensemble de nœuds du graphe de tournée, $N = C \cup \{0\}$ avec pour indice $i^*, j^*, v^* \in [0, r^*]$ où r^* désigne le nombre de nœuds. Par convention, pour chaque intervenant, $\{0\}$ indique la structure du HHC. Du point de vue du bénéficiaire, $\{0\}$ indique le point temporel où toutes les interventions (ou une seule) demandées par un bénéficiaire commencent et se terminent.
- M : une constante positive très élevée appliquée à la fin de la période couverte. Les lecteurs peuvent se référer à (Cococcioni and Fiaschi, 2020) pour en savoir plus sur la définition de cette constante et les valeurs qui peuvent lui être attribuées (à ce qui est désigné dans la littérature comme le « *big M* »).
- $[e_i, l_i]$: la fenêtre de temps demandée par le bénéficiaire pour le début de l'intervention i . La définition de la fenêtre de temps est sur la section 2.2.1.3 du chapitre II.
- dif_i : l'indicateur de gravité représentant la difficulté pour opérer une intervention i , $dif_i \in [1, 6]$.
- $duree_{i^*}$: le temps de prestation de service au nœud i^* , la valeur vaut 0 pour le nœud 0.
- $temp_{i^*j^*}$: le temps de déplacement entre des nœuds i^* et j^* . C'est un poids de l'arc entre deux nœuds.
- δ_p : le délai minimal entre 2 interventions demandées par le même bénéficiaire p .
- α_p : la durée maximale de tolérance du bénéficiaire p une fois que la borne maximale de la fenêtre d'intervention est dépassée.
- α_p^* : le taux de satisfaction en pourcentage du bénéficiaire p par rapport à α_p , $\alpha_p^* \in [0, 100]$.
- β_p : la durée maximale de tolérance sur le délai δ_p entre 2 interventions du bénéficiaire p si ce délai n'est pas respecté.
- β_p^* : le taux de satisfaction en pourcentage du bénéficiaire p par rapport à β_p , $\beta_p^* \in [0, 100]$.
- ξ_k : la charge de travail maximale de l'intervenant k par jour.
- γ_k : la tolérance de l'intervenant k lorsque la charge de travail maximale est dépassée.
- γ_k^* : le taux de satisfaction en pourcentage de l'intervenant k par rapport à γ_k , $\gamma_k^* \in [0, 100]$.
- d_{pi} : la matrice d'expression de la demande du bénéficiaire p pour une intervention i , un élément est égal à 1 s'il désigne demande effective, 0 sinon.
- $faisa_{ki}$: la matrice d'expression de la faisabilité entre l'intervenant k et l'intervention i demandée par un bénéficiaire (agrégeant compétence et compatibilités réciproques), un élément est égal à 1 s'il indique une telle faisabilité, 0 sinon.

3.2.2. Jeu des variables de décision

Soient les variables binaires suivantes :

- $X_{i^*j^*k} = \begin{cases} 1 & \text{Si l'intervenant } k \text{ se déplace de } i^* \text{ à } j^* \text{ pour effectuer une intervention,} \\ 0 & \text{sinon} \end{cases}$
- $Y_{ik} = \begin{cases} 1 & \text{Si l'intervenant } k \text{ effectue l'intervention } i, \\ 0 & \text{sinon} \end{cases}$
- $Z_{i^*j^*p} = \begin{cases} 1 & \text{Si bénéficiaire } p \text{ demande l'intervention } i^* \text{ avant l'intervention } j^*, \\ 0 & \text{sinon} \end{cases}$

Ainsi que les variables entières ou réelles qui complètent le jeu :

- $debut_i$: l'heure de début de l'intervention i , en minutes.
- W_i : le temps d'attente en cas de dépassement de l'heure de début de l'intervention i la plus tardive, en minutes.
- R_{ijp} : le délai inter-opérateur entre des interventions i et j demandées par le bénéficiaire p , en minutes.
- O_k : la charge de travail réelle de l'intervenant k , en minutes.
- $depass_k$: le dépassement de la charge de travail de l'intervenant k , en minutes.
- G_k : la somme des difficultés des interventions effectuées par l'intervenant k , $G_k \in [0, \infty]$.
- W_p : le taux de satisfaction quant à l'ensemble des temps d'attente des interventions demandées par le bénéficiaire p , $W_p \in [0,1]$
- R_q : le taux de satisfaction quant à l'ensemble des délais inter-interventions demandées par le bénéficiaire q , $R_q \in [0,1]$.
- O_k : le taux de satisfaction quant au dépassement de la charge de travail de l'intervenant k , $O_k \in [0,1]$.
- G_k : le coefficient de difficulté de la tournée de l'intervenant k , $G_k \in [0,1]$.

3.2.3. Ecriture de la fonction objectif

Notre fonction objectif est une expression scalaire à valeur réelle qui se compose de deux parties : la partie « faisabilité » et la partie « qualité ». Notre approche est donc multicritère.

La partie faisabilité correspond à la partie entière de cette valeur scalaire réelle (à gauche de la somme). Elle désigne la réussite des opérations d'affectation des interventions aux intervenants. C'est un niveau de services rendus. Un succès se juge donc par cette valeur qui doit être égale au nombre des interventions à effectuer dans le jour concerné.

La qualité correspond à la partie décimale de ce même scalaire (à droite de cette somme). Elle exprime les satisfactions que chaque partie prenante de l'offre et de la demande peut tirer de la solution. Il s'agit d'une agrégation de W_p , R_q avec $p \in P$, $q \in Y$ du point de vue du bénéficiaire, et de O_k, G_k avec $k \in K$ du point de vue de l'intervenant.

Afin de normer cette fonction, nous calculons le ratio entre cette valeur décimale de la qualité et sa borne supérieure, pour garantir que le résultat se situe dans la plage $[0,1]$.

$$\text{maximiser } \sum_{k \in K} \sum_{i \in C} Y_{ik} + \frac{\sum_{p \in P} W_p + \sum_{q \in Y} R_q + \sum_{k \in K} O_k + \sum_{k \in K} G_k}{n \cdot 2 + m + l} \quad (1)$$

3.2.4. Contraintes

La contrainte (2) assure que chaque intervention est effectuée par l'intervenant k exactement une fois et que chaque intervention est effectuée par un seul intervenant.

$$\sum_{k \in K} Y_{ik} = 1 \quad \forall i \in C \quad (2)$$

La contrainte (3) assure l'affectation correcte entre les intervenants et les interventions.

$$Y_{ik} \leq \text{faisa}_{ki} \quad \forall i \in C, \forall k \in K \quad (3)$$

Du point de vue de l'intervenant, les contraintes (4) et (5) assurent que chaque intervention attribuée à l'intervenant k a un prédécesseur et un successeur. Ces contraintes s'apparentent au contrôle des entrées des villes dans le TSP, mais ici il faut opérer en multi-tournées.

$$\sum_{j^* \in N: j^* \neq i} X_{ij^*k} = Y_{ik} \quad \forall i \in C, \forall k \in K \quad (4)$$

$$\sum_{i^* \in N: i^* \neq j} X_{i^*jk} = Y_{jk} \quad \forall j \in C, \forall k \in K \quad (5)$$

Du point de vue du bénéficiaire, les contraintes (6) et (7) assurent que les interventions demandées seront exécutées suivant la séquence fixée en respect de la faisabilité.

$$\sum_{j^* \in N: j^* \neq i} Z_{ij^*p} = d_{pi} \quad \forall i \in C, \forall p \in P \quad (6)$$

$$\sum_{i^* \in N: i^* \neq j} Z_{i^*jp} = Y_{pj} \quad \forall j \in C, \forall p \in P \quad (7)$$

Les contraintes (8) et (9) assurent que tous les intervenants partent et terminent leurs tournées à la structure de PAD.

$$\sum_{i^* \in N} X_{0i^*k} = 1 \quad \forall k \in K \quad (8)$$

$$\sum_{i^* \in N} X_{i^*0k} = 1 \quad \forall k \in K \quad (9)$$

Les contraintes (10) et (11) assurent que l'ensemble des interventions demandées par le même bénéficiaire p commencent ont une structure d'ordre temporelle (le point de début et de fin est décrit dans la définition de N). Il s'agit de respecter les consignes du plan de soins.

$$\sum_{i^* \in N} Z_{0i^*p} = 1 \quad \forall p \in P \quad (10)$$

$$\sum_{i^* \in N} Z_{i^*0p} = 1 \quad \forall p \in P \quad (11)$$

La contrainte (12) garantit que chaque intervention doit commencer après l'heure de début au plus tôt demandée par le bénéficiaire.

$$debut_i \geq e_i \quad \forall i \in C \quad (12)$$

La contrainte (13) assure que deux interventions effectuées par le même intervenant se déroulent l'une après l'autre.

$$debut_i + duree_i + temp_{ij} \leq debut_j + M \cdot \left(1 - \sum_{k \in K} X_{ijk}\right) \quad \forall i, j \in C: i \neq j \quad (13)$$

La contrainte (14) assure que deux interventions demandées par un même bénéficiaire respectent le délai inter-intervention fixé (contrainte souple).

$$debut_i + duree_i + \sum_{q \in Y} (\delta_p - R_{ijq}) \leq debut_j + M \cdot \left(1 - \sum_{q \in Y} Z_{ijq}\right) \quad \forall i, j \in C: i \neq j \quad (14)$$

La contrainte (15) détermine la valeur du temps d'attente de l'intervention i .

$$W_i = \max(debut_i - l_i, 0) \quad \forall i \in C \quad (15)$$

La contrainte (16) détermine la charge de l'intervenant k .

$$O_k = \sum_{i^* \in N} \sum_{j^* \in N: i^* \neq j^*} (temp_{i^*j^*} + duree_{j^*}) \cdot X_{i^*j^*k} \quad (16)$$

La contrainte (17) détermine la surcharge de travail de l'intervenant k .

$$depass_k = \max(O_k - \xi_k, 0) \quad \forall k \in C \quad (17)$$

La contrainte (18) détermine la difficulté des interventions effectuées par l'intervenant k .

$$G_k = \sum_{i \in C} Y_{ik} \cdot dif_i \quad \forall k \in K \quad (18)$$

Comme mentionné dans la section 2, notre modèle est destiné à maximiser la satisfaction totale de tous les acteurs de la PAD. Nous tolérons la violation des contraintes temporelles telles que le temps d'attente et l'absence de délai inter-opérateur du côté des bénéficiaires, ainsi que le dépassement de la charge maximum du côté des intervenants. En fonction des valeurs de la tolérance et du taux de satisfaction de chaque partie prenante, une fonction de régression linéaire est appliquée pour modéliser la satisfaction, qualifiée par une réponse scalaire, en regard de la valeur de dépassement temporel reflétant le non-respect d'une contrainte. La Figure 42 schématise cette fonction de régression linéaire pour générer le taux de satisfaction de ces deux catégories d'acteurs dans la PAD. Prenant par exemple la contrainte (14) (rythme des interventions pour un même bénéficiaire), le taux de satisfaction sera de 100% (1 en nombre

entier) lorsque le délai inter-intervention demandé est respecté. Si la contrainte n'est pas satisfaite, la fonction linéaire (19) exprime le taux de satisfaction.

$$TAUX\% = \frac{\frac{Tol\%}{100} - 1}{Tol} \cdot Violation + 1 \quad (19)$$

La fonction (19) est appliquée aux contraintes (14) (15) (16), où Tol est la tolérance d'un bénéficiaire ou d'un intervenant (se référer au paramètre $\alpha_p, \beta_p, \gamma_k$), $Tol\%$ est le taux de satisfaction d'un bénéficiaire ou d'un intervenant (se référer au paramètre $\alpha_p^*, \beta_p^*, \gamma_k^*$) par rapport à sa tolérance Tol . Le résultat de $TAUX\%$ (axe des ordonnées) est compris dans l'intervalle $[0,1]$. Elle se construit à partir de deux points (comme toute droite) : (1) la limite de la plage temporelle où le taux est de 100%, (2) la donnée du taux de satisfaction de l'individu pour la valeur de la durée maximale de tolérance. La droite de la partie en rouge de la Figure 42 n'est jamais atteinte, par contrôle du domaine de définition du taux de satisfaction.

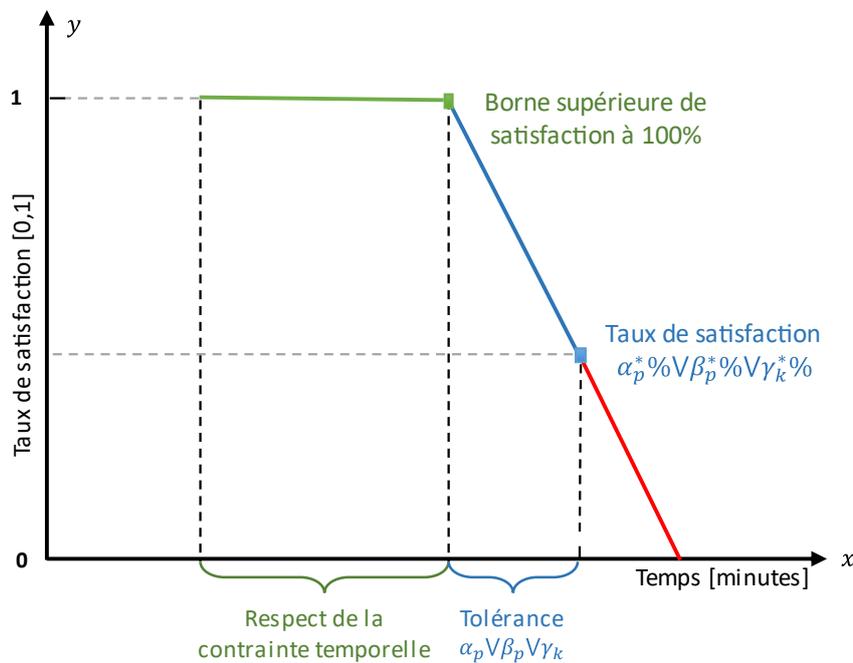


Figure 42 : Illustration de la satisfaction des acteurs dans la PAD basée sur une fonction de régression linéaire

A partir de cette hypothèse commune à toutes les formes de satisfaction (ou d'insatisfaction) par une fonction de régression linéaire, les contraintes souples (20) à (22) calculent respectivement les deux taux de satisfaction du bénéficiaire $p : \mathbb{W}_p$ et \mathbb{R}_p , et le taux de satisfaction de l'intervenant $k : \mathbb{O}_k$.

$$\mathbb{W}_p = \frac{\sum_{i \in \mathcal{C}} \left(\frac{\alpha_p^*}{100} - 1 \right) \cdot W_i + 1}{\sum_{i \in \mathcal{C}} d_{pi}} \quad \forall p \in P \quad (20)$$

$$\mathbb{R}_q = \frac{\sum_{i \in C} \sum_{j \in C: i \neq j} \left(\frac{\beta_p^*}{100} - 1 \cdot R_{ijq} + 1 \cdot Z_{ijq} \right) \cdot d_{qi} \cdot d_{qj}}{\sum_{i \in C} d_{qi}} \quad \forall q \in Y \quad (21)$$

$$\mathbb{O}_k = \frac{\gamma_k^*}{100} - 1 \cdot \text{depass}_k + 1 \quad \forall k \in K \quad (22)$$

La contrainte (23) détermine le coefficient de difficulté \mathbb{G}_k .

$$\mathbb{G}_k = \min \left(1, \frac{G_k}{\frac{\sum_{i \in C} \text{dif}_i}{n}} \right) \quad \forall k \in K \quad (23)$$

Les contraintes (24) à (29) portent sur des domaines de définitions et garantissent que $\mathbb{W}_p, \mathbb{R}_p, \mathbb{O}_k, \mathbb{G}_k$ sont des variables réelles continues dans $[0,1]$ et des variables non négatives.

$$\mathbb{W}_p \in [0,1] \quad \forall p \in P \quad (24)$$

$$\mathbb{R}_q \in [0,1] \quad \forall q \in Y \quad (25)$$

$$\mathbb{O}_k \in [0,1] \quad \forall k \in K \quad (26)$$

$$\mathbb{G}_k \in [0,1] \quad \forall k \in K \quad (27)$$

$$\text{debut}_i \geq 0 \quad \forall i \in C \quad (28)$$

$$R_{ijp} \geq 0 \quad \forall i, j \in C: i \neq j, \forall p \in P \quad (29)$$

3.3. PSM : modèle codé en OPL pour l'appel à CPLEX

En rapport au constat dressé dans la section 2.3 du chapitre II (cf. le Tableau 4), les problèmes formulés en PLNM sont souvent résolus par un solveur commercial (86% de tous les articles que nous avons étudiés). Le solveur CPLEX est utilisé le plus fréquemment (61% des articles utilisant des solveurs pour la résolution de modèles). C'est pourquoi nous avons choisi a priori CPLEX pour résoudre notre modèle de PLNM.

CPLEX est accessible via un environnement de développement intégré (*Integrated Development Environment - IDE*) nommé *Cplex Studio IDE* (sous Windows) ou *oplide* (sous Linux) (CPLEX, 2011). Le langage utilisé dans Cplex Studio IDE est OPL (*Optimization Programming Language*) (Van Hentenryck, 1999). C'est un langage qui permet d'écrire facilement des modèles de PLNE ou PLNM grâce à une syntaxe proche de la formulation mathématique traditionnelle d'un problème d'optimisation. Selon (Flatberg, 2009), la motivation expliquant la décision de recourir à des langages de modélisation tels que OPL tient principalement en deux raisons :

- Il fournit une syntaxe proche de la formulation mathématique, ce qui facilite le passage de la formulation mathématique à une forme pouvant être résolue par l'ordinateur.
- Il permet une séparation nette entre le modèle et les données associées. Le même modèle peut alors être résolu avec des données différentes avec peu d'effort supplémentaire.

La construction de notre modèle de PLNM à l'aide d'OPL est réalisée par « projets ». Pour ce faire, nous devons créer un projet OPL dans *Cplex Studio IDE* qui doit contenir au minimum un fichier « modèle » et un fichier de « configuration d'exécution ». En effet, chaque projet est constitué de plusieurs types de fichiers (Toussaint, 2016):

- Un fichier modèle (.mod) qui contient le modèle à résoudre.
- Un fichier de données (.dat) qui contient les données pour un modèle.
- Un fichier de paramètres (.ops), facultatif, pour paramétrer la méthode de calcul dans CPLEX.
- Un fichier de configuration d'exécution (.oplproject) qui indique à l'IDE ce qu'il doit faire quand l'utilisateur demande l'exécution du projet, c'est-à-dire quel est le modèle à résoudre et quels sont les paramètres et les données.

Afin d'illustrer ce modèle de PLNM en OPL, un exemple du code pour un cas du TSP classique à 4 villes est fourni sur la Figure 43. On y distingue une partie déclarative en tête de code où données (mots clés : *int*, *range*, *tuple*, *setof*, *int*, lignes 1 à 10) et variables (mot-clé *dvar*, lignes 11 et 12) sont distinguées de manière abstraite. Puis, il y a une partie exécutable où les relations explicitant la fonction à optimiser (*minimize*, lignes 15) et, ensuite, les contraintes qui sont définies après le mot clé qui explique que la solution est assujettie à des contraintes (*subject to*, lignes 16 à 23). Dans le fichier de données (*TSPsolutionModel.dat*) dont le contenu est l'encadré du bas de la figure, nous spécifions les valeurs du cas d'utilisation que nous voulons résoudre. L'écriture en OPL est vraiment très proche des formulations mathématiques traditionnelles recommandées par l'*American Mathematical Society*¹, et que la recherche opérationnelle pratique de manière assez universelle.

```

1 int totalNumberOfCityNode=...;
2 int beginNodeIndex=...;
3 range cityNodeSet=beginNodeIndex..totalNumberOfCityNode;
4 tuple routingEdge{
5     int startNodeID;
6     int endNodeID;
7 }
8 setof(routingEdge) totalEdgeTuple=...;
9 int distanceMatrix[totalEdgeTuple]=...;
10 int travellingTimeMatrix[totalEdgeTuple]=...;
11 dvar boolean dvarAssignment[totalEdgeTuple];
12 dvar int+ dvarNoSubTour[cityNodeSet];
13
14
15 minimize sum ( <i,j> in totalEdgeTuple ) distanceMatrix[<i,j>] * dvarAssignment[<i,j>];
16 subject to {
17 forall( i in cityNodeSet, <i,j> in totalEdgeTuple )
18     sum( j in cityNodeSet: j!=i ) dvarAssignment[<i,j>]==1;
19 forall( j in cityNodeSet, <i,j> in totalEdgeTuple )
20     sum( i in cityNodeSet: i!=j ) dvarAssignment[<i,j>]==1;
21 forall( i in cityNodeSet: i>1, j in cityNodeSet: j>1 && j!=i )
22     dvarNoSubTour[i] - dvarNoSubTour[j] + (totalNumberOfCityNode - 1) * dvarAssignment[<i,j>] <= totalNumberOfCityNode - 2;
23 }
24
25
26 main {
27     thisOplModel.generate();
28     cplex.solve();
29     var ofile = new IloOplOutputFile("D:\\Users\\liwen.zhang\\Desktop\\tspOFResult.txt");
30     ofile.write(cplex.getBestObjValue()+" ");
31     ofile.write(cplex.getSolvedTime())
32     ofile.close();
33     var ofCSV = new IloOplOutputFile("D:\\Users\\liwen.zhang\\Desktop\\tspAssResult.csv");
34     for(var i in thisOplModel.dvarAssignment){
35         ofCSV.writeln(i.startNodeID+", "+i.endNodeID+", "+thisOplModel.dvarAssignment[i]);
36     }
37     ofCSV.close();
38 }

```

TSPsolutionModel.mod

```

1 beginNodeIndex=1;
2 totalNumberOfCityNode=4;
3 totalEdgeTuple={<1,2><1,4><1,3><2,4><2,1><2,3><4,2><4,1><4,3><3,1><3,2><3,4>};
4 distanceMatrix=[550,175,750,115,600,450,135,230,200,800,430,180,];
5 travellingTimeMatrix=[0,0,0,0,0,0,0,0,0,0,0,];

```

TSPsolutionModel.dat

Figure 43 : Modèle de PLNM en langage OPL pour le TSP classique

¹ American Mathematical Society: <https://www.ams.org/home/page>

4. Méthode approchée basée sur la satisfaction de contraintes (CSP)

Selon l'analyse bibliographique de la section 2.3 du chapitre II (cf. le Tableau 4), pour beaucoup de problèmes de taille réaliste (supérieure à 80 interventions par jour, par exemple), une méthode exacte peut nécessiter un temps de calcul très long pour trouver une solution optimale, pouvant atteindre des durées incompatibles avec le besoin (plusieurs jours ou mois). Il est de grande complexité au sens de l'analyse combinatoire. Pour tenter de réduire ce temps de calcul, nous proposons une résolution de problèmes HHCRSP formulés sur la base d'un CSP dont la particularité réside dans l'utilisation active des contraintes du problème pour réduire la taille de l'espace des solutions à parcourir.

4.1. PIM : modèle en satisfaction de contrainte (CSP)

Un CSP (*Constraint Satisfaction Problem*) (Floudas and Pardalos, 1990) est une méthode de résolution très utilisée pour des problèmes réels avec des contraintes. La programmation par contraintes est un paradigme puissant pour résoudre des problèmes d'optimisation combinatoire en faisant appel à un large éventail de techniques, relevant de l'intelligence artificielle (Rossi et al., 2006). L'idée de la programmation par contraintes est de résoudre des problèmes en énonçant des contraintes sur le domaine du problème et, par conséquent, de trouver une solution les satisfaisant toutes (Barták, 1999). Selon (Russell and Norvig, 2002), un CSP est défini par un triplet (X, D, C) tel que :

- $X = \{X_1, X_2, \dots, X_N\}$ est l'ensemble des variables du problème.
- D est une fonction qui associe à chaque variable X_i son domaine $D(X_i)$ (valeurs possibles de X_i).
- $C = \{C_1, C_2, \dots, C_M\}$ est l'ensemble des contraintes du problème. Chaque contrainte est définie par un couple (t, R) tel que : t est l'ensemble des variables liées à la contrainte, $t \in X$, R définit une relation entre ces variables.

Obéissant à cette définition, nous construisons dans la Figure 44 un méta-modèle représentant un CSP.

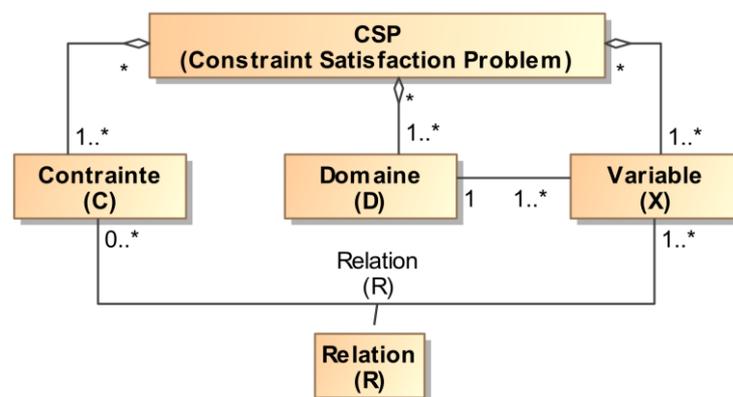


Figure 44 : Méta-modèle du CSP

Les CSP sont multiples, nous pouvons citer à titre d'exemples, le problème de la coloration de graphe (Jensen and Toft, 2011) et le problème des N-Reines (Gutiérrez Naranjo et al., 2009). La formulation en CSP est très peu utilisée pour la résolution du problème HHCRSP (7% des publications selon notre analyse bibliographique, cf. la section 2.3.4 du chapitre II et le Tableau

4). La complexité d'un CSP est NP-difficile, en général (Rossi et al., 2006). En bref, cela signifie que la solution optimale ne peut être déterminée dans un délai raisonnable, tout comme la formulation basée sur la PLNM. Avec des cas du monde réel, les objectifs de Berger-Levrault sont de résoudre un HHCRSP ayant une taille conséquente (dépassant la centaine d'interventions par jour), avec les contraintes que nous avons définies, en utilisant le moins de temps de calcul possible, et en investissant la juste proportion de ressources financières (l'achat de licence de solveur commercial, CPLEX, par exemple) et humaines dans le processus de développement de cet outil numérique de pilotage de la PAD. Afin de rentrer dans le cadre de ces exigences industrielles, nous avons opté pour une alternative, le solveur OptaPlanner (OptaPlanner, 2021).

OptaPlanner¹ est un solveur à base de satisfaction de contraintes. Nous avons trouvé des références dans la littérature où des travaux où OptaPlanner sont cités. Dans (Kosecka-Žurek, 2019), une application préexistante d'OptaPlanner a été utilisée pour résoudre le problème du VRP. OptaPlanner a également été utilisé pour résoudre une partie de VRP impliquant le mécanisme dit de collecte des déchets (Lozano Murciego et al., 2015). Dans un autre travail, il est fait référence à un problème d'attribution des tâches (Macik, 2016). Dans (Rios de Souza and Martins, 2020), le travail vise à explorer différentes solutions en utilisant OptaPlanner pour traiter un problème de conditionnement des poubelles, où il est question de comparer les résultats obtenus par des métaheuristiques différentes, embarquées dans OptaPlanner.

4.2. PSM : modèle en satisfaction de contrainte orienté OptaPlanner

4.2.1. Description d'OptaPlanner

OptaPlanner, publié sous licence Apache Software est un moteur de planification léger et ouvert. Il permet aux programmeurs Java de traiter des modèles spécifiques à un domaine en réutilisant un modèle existant. Les modèles mis à disposition dans OptaPlanner sont nombreux. Nous pouvons citer en particulier le VRP et le TSP (cf. la section 2.3.1.1 du chapitre II) qui sont voisins du sujet que nous traitons. Une des forces de cet outil numérique est de pouvoir combiner (1) des heuristiques visant à la génération d'une solution initiale du problème à résoudre, et (2) des métaheuristiques d'optimisation (la recherche tabu, par exemple) pour parcourir l'espace de recherche, avec un système de calcul de score très performant. Comme expliqué dans la section 2.3 du chapitre II, les métaheuristiques sont applicables à un très large éventail de HHCRSP. En fait, le principal avantage d'une métaheuristique est qu'elle ne nécessite pas beaucoup de réglages pour chaque cas d'utilisation. Il faut uniquement choisir un petit nombre de paramètres intrinsèques à l'algorithme (paramètres de température pour le recuit simulé, par exemple). Beaucoup de ces algorithmes démontrent qu'en leur donnant suffisamment de temps, ils convergent plus ou moins rapidement vers une solution acceptable (Blum and Roli, 2003). Pour des cas d'utilisation réalistes, une solution quasi-optimale peut souvent être considérée comme acceptable. Il appartient à l'utilisateur de trouver le bon compromis entre le temps de calcul et l'atteinte de l'optimalité du résultat (Macik, 2016).

La construction d'un modèle d'optimisation avec OptaPlanner se réalise en 3 étapes.

¹ <https://www.optaplanner.org/>

4.2.1.1. Principes de modélisation d'un CSP avec OptaPlanner

OptaPlanner est conçu selon une approche orientée objet. Cela impacte la manière de modéliser le problème spécifié. Il s'agit de définir des formules mathématiques par sélection et assemblage de classes existantes au moyen d'une interface. La configuration emploie un mécanisme d'annotations.

Nous distinguons trois catégories de classes définies ci-après.

(1) Classe d'information du problème (*Problem Fact*)

Ce sont les classes ne pouvant être changées par le solveur lors de la recherche de solution (les données restant identiques lors de la recherche de solutions), par exemple : « **Bénéficiaire** » et « **Structure de PAD** ». Toutes les propriétés d'une classe d'information d'un problème sont des propriétés du problème. Ces classes sont sans annotation. Ainsi, nous pouvons affirmer que parmi toutes les classes définies dans le modèle, si la classe n'est pas annotée, cela signifie qu'il s'agit d'une classe « *Problem Fact* ».

(2) Classe d'entité de planification (*Planning Entity*).

Ces classes peuvent subir des changements pendant le processus de recherche de la solution. Les propriétés qui changent au cours de la planification sont des variables de décision dites principales (*Planning Variable*). Ces classes sont annotées par le mot clé `@PlanningEntity`, leurs propriétés sont annotées par `@PlanningVariable`, et les autres sont des propriétés sans annotation.

Dans certains cas d'utilisation que nous avons testés, comme le TSP, le VRP ou le HHCRSP, il est nécessaire d'enchaîner les instances du *Planning Entity* dans le modèle. Ceci signifie que les *Planning Entity* forment une chaîne en pointant les uns vers les autres qui conduit, *in fine*, à prendre en charge une structure d'ordre entre les entités à planifier. Chaque *Planning Entity* fait donc partie d'une chaîne ouverte (*open-ended chain*) qui part d'une ancre et finit à un autre (cf. la Figure 45). On peut noter une série d'exigences pour la construction d'une chaîne dans un tel modèle (OptaPlanner, 2021) :

- Une chaîne n'est jamais une boucle. Le maillon final est toujours ouvert.
- Chaque chaîne possède toujours exactement une « ancre » (*anchor*). Cette « ancre » n'est jamais une instance de la classe *Planning Entity*.
- Une chaîne n'est jamais un arbre, elle est toujours linéaire. Chaque « ancre » ou *Planning Entity* a donc un et un seul *Planning Entity* en bout de chaîne.
- Chaque *Planning Entity* fait partie d'une chaîne.
- Une ancre sans *Planning Entity* pointant vers elle-même est également considérée comme une chaîne.

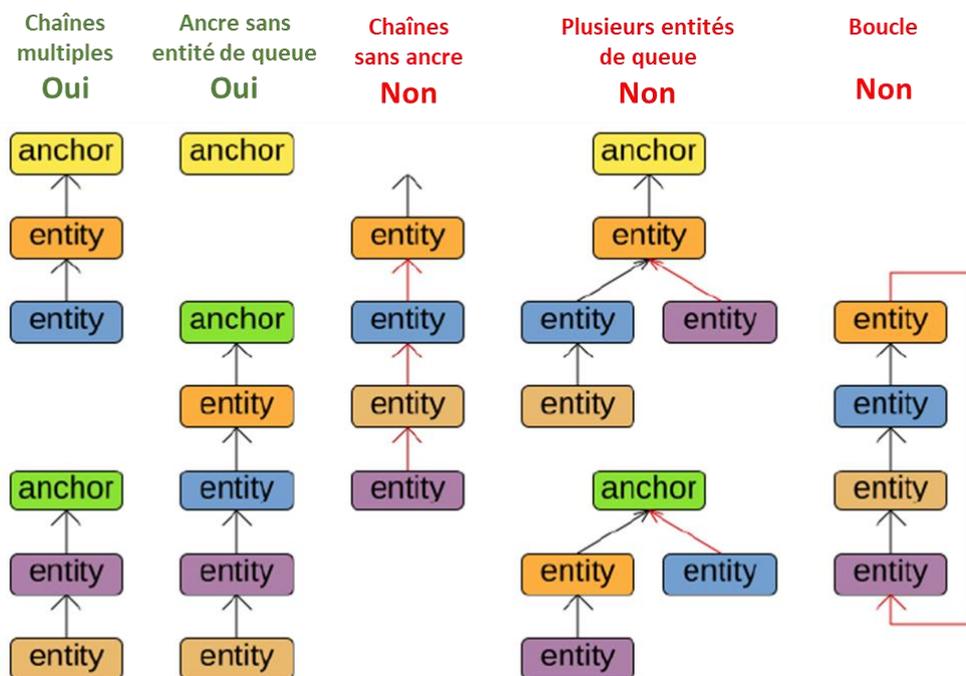


Figure 45 : Exemples de chaînes valides et invalides (OptaPlanner, 2021)

Lorsque les *Planning Entity* sont chaînés, les variables de décision (`@PlanningVariable`) définis dans ces entités sont de fait également chaînés :

- La variable de décision (annoté par `@PlanningVariable`) définie dans chaque objet du *Planning Entity* est considérée comme un pointeur vers un élément précédent. Elle est appelée variable principale.

Il existe dans OptaPlanner un concept de **variable secondaire** intitulée « variable d'ombre » qui est dépendante de la **variable principale**. Cette dépendance est de nature mathématique, c'est une relation de causalité pour le calcul du secondaire à partir du principal. Une « variable d'ombre » est une variable de planification dont la valeur correcte peut être déduite de l'état des variables de décision principales (`@PlanningVariable`) (OptaPlanner, 2021). Par exemple dans le TWVRP (*Time Windowed Vehicle Routing Problem*), le temps d'arrivée d'un véhicule (variable secondaire) chez un client peut être calculé en fonction des clients précédemment visités par ce véhicule (variable principale). Lorsque les clients pour un véhicule changent, l'heure d'arrivée prévue pour chaque client est automatiquement ajustée. Du point de vue de l'optimisation, OptaPlanner n'optimise effectivement que les variables de décision principales (`@PlanningVariable`) : il s'assure simplement que si une variable principale est modifiée, toutes les variables d'ombre dépendantes sont adaptées en conséquence. Ainsi, chaque variable d'ombre doit indiquer sa source par une annotation complémentaire comme par exemple : `@CustomShadowVariable(sourceVariableName="nom de variable principale")`.

Trois catégories de variables secondaires sont disponibles dans OptaPlanner comme suit :

- Une variable d'ombre customisée (annotée par `@CustomShadowVariable`) dépend de la variable de décision principale. Un *écouteur* doit être défini pour recalculer la valeur de la variable d'ombre customisée.

- Une variable d'ombre inversée (annotée par `@InverseRelationShadowVariable`) définie dans chaque objet du *Planning Entity* est considérée comme un pointeur vers l'élément suivant. Ce dernier agit sur le prochain objet du *Planning Entity* ou est nul si cet élément est le dernier. Cette variable coexiste en dualité avec la variable d'ombre customisée (`@CustomShadowVariable`)
- Un *Planning Entity* a une variable d'ancre (annotée par `@AnchorShadowVariable`), de sorte qu'il n'est pas nécessaire de traverser toute la chaîne pour obtenir son ancre.

La Figure 46 illustre par un exemple le codage en OptaPlanner des relations entre 4 types de variables dans le cas d'un modèle HHCRSP. Notons que l'« *arrêt précédent* » (`@PlanningVariable`) dans l'« *Intervention* » et la « *prochaine intervention* » (`@InverseRelationShadowVariable`) dans l'« *Arrêt* » visent à construire une chaîne entre les objets de l'« *intervention* ». Puis, l'« *intervenant affecté* » (`@AnchorShadowVariable`) à l'« *Intervention* » signifie que les objets d'« *Intervention* » chaînés doivent avoir un intervenant (l'objet de l'« *Intervenant* ») affecté. Enfin, le « *temps de début planifié* » (`@CustomShadowVariable`) dépend de l'« *arrêt précédent* » (`@PlanningVariable`), sa valeur doit être recalculée lorsque le chainage des interventions est modifié.

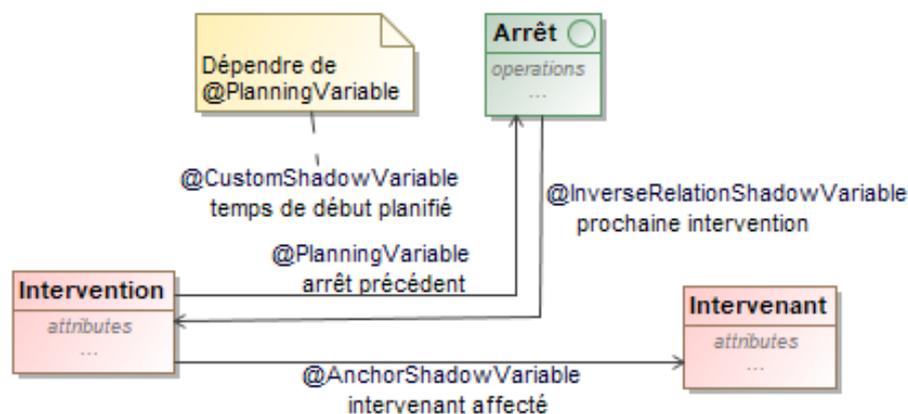


Figure 46 : Illustration de la relation entre quatre types de variable dans OptaPlanner

(3) Classe de solution de planification (*Planning Solution*).

Cette classe représente la solution du problème (annotée par `@PlanningSolution`), elle contient l'ensemble des objets (annotés par `@ProblemFactCollectionProperty`) instanciés de classes « *Problem Fact* » et « *Planning Entity* ».

OptaPlanner possède un système de score sur chaque contrainte pour juger la qualité d'une solution. Par défaut, ce système de score est composé de 3 niveaux (dur, moyen et souple) ou de 2 niveaux (dur et souple). Un score dur est significatif du respect de « *contrainte dure* ». A contrario, si le score dur d'une solution est inférieur à 0 (la valeur de pénalité quant à la violation de contrainte est souvent une valeur négative), cette solution est donc considérée comme une solution « *infaisable* ». Un score moyen ou souple est lié au respect de « *contrainte souple* ». Le score moyen et souple le plus élevé conduit à une meilleure solution.

Ce système de score est utilisé pour hiérarchiser un groupe de contraintes selon leur types. Par exemple, dans notre HHCRSP avec un score à 2 niveaux, un intervenant ne peut pas exécuter deux interventions en même temps et cela constitue une contrainte dure. Le respect de cette

contrainte est associé à un score dur. Le respect de la préférence personnelle d'un intervenant va relever d'une contrainte souple. Il est associé à un score souple. Les niveaux de deux scores différents sont comparés lexicographiquement, c'est-à-dire que la solution avec le score (0 dur / -200 souple) est meilleure que la solution avec le score (-1 dur / 0 souple). Car le score dur prime sur le score souple.

Le Tableau 12 est une synthèse du mécanisme d'annotations dans OptaPlanner.

Tableau 12 : Mécanisme d'annotations dans OptaPlanner

Nom	Annotation	Description
Classe d'information du problème	-	Ce sont les classes ne pouvant être changées par le solveur lors de la recherche de solution
Classe d'entité de planification	@PlanningEntity	Ces classes peuvent subir des changements pendant le processus de recherche de la solution
Classe de solution de planification	@PlanningSolution	Cette classe représente la solution du problème
	@ProblemFactCollectionProperty	L'ensemble des objets instanciés de classes « <i>Problem Fact</i> » et « <i>Planning Entity</i> »
Variable de décision principale	@PlanningVariable	Les propriétés qui changent au cours de la planification, appartenant à la classe d'entité de planification
Variable d'ombre customisée (Variable de décision secondaire)	@CustomShadowVariable	La valeur de cette variable est déduite de l'état des variables de décision principales
Variable d'ombre inversée (Variable de décision secondaire)	@InverseRelationShadowVariable	Cette variable coexiste en paire avec la variable d'ombre customisée
Variable d'ancre (Variable de décision secondaire)	@AnchorShadowVariable	Les entités de planification chaînées possèdent une « ancre », qui indique le début d'une chaîne. Par exemple un intervenant est une « ancre » quant à sa tournée.

4.2.1.2. Spécification des contraintes du problème

Cette phase consiste à écrire les règles représentant les contraintes du problème à respecter. OptaPlanner propose 4 méthodes à cette fin :

- Codage en Java : Implémentation de toutes les contraintes dans une méthode en Java. Cette manière de déclarer n'est pas évolutive.
- Définition par des flux de contraintes (*Constraint streams*) : Les *ConstraintStreams* sont une forme de programmation fonctionnelle pour le calcul incrémental de score en Java simple, qui est facile à lire, à écrire et à déboguer. Cette façon de déclarer les contraintes est évolutive.
- Codage en Java incrémental : Implémentation de plusieurs méthodes de bas niveau en Java pour déclarer les contraintes. Cette manière de déclarer est évolutive, mais très difficile à mettre en œuvre et à maintenir.
- Usage d'un moteur de règle, en l'occurrence, il s'agit du moteur Drools¹: Implémentation de chaque contrainte comme une règle de score distincte en utilisant ce moteur de règles. Cette manière de déclarer est évolutive.

¹ Moteur de règle Drools : <https://www.drools.org/>

4.2.1.3. Résolution du problème

OptaPlanner opère généralement en deux étapes (cf. la Figure 47) : (1) la résolution du problème par le solveur et (2) l'évaluation de la solution par le calculateur de score.

(1) Pour la résolution du problème, une solution initiale est d'abord générée à l'aide d'heuristiques de construction, dans un temps limité et avec la meilleure qualité possible. Il existe 5 heuristiques de construction qu'OptaPlanner prend en charge : *First fit* avec sa variante *Weakest fit* et *Strongest fit* (Bays, 1977), *Allocate entity from queue* (Semeria, 2001), *Cheapest insertion* (Hassin and Keinan, 2008) et *Regret insertion* (Diana and Dessouky, 2004), *Allocate from pool* (Ke and Fang, 2004), *Scaling construction heuristics* (Katayama et al., 2009).

La deuxième phase consiste à rechercher continuellement par itération une meilleure solution que la solution courante grâce à des algorithmes de type métaheuristiques. Dans OptaPlanner, la configuration permet de choisir ces algorithmes d'optimisation et de définir leurs paramètres intrinsèques. L'optimisation se termine lorsque des critères d'arrêt sont atteints. Un critère d'arrêt est souvent défini comme l'atteinte, soit d'une limite maximum de temps de calcul, soit d'un certain nombre d'itérations. Tout comme l'heuristique de construction, OptaPlanner propose de nombreux types de métaheuristiques pour la recherche locale :

- La recherche locale simple : *Hill climbing* (Goldfeld et al., 1966) et *Late acceptance* (Burke and Bykov, 2012).
- Les métaheuristiques : TS (*Tabu Search*) (Glover and Laguna, 1998), SA (*Simulated Annealing*) (Van Laarhoven and Aarts, 1987), *Great deluge* (Dueck, 1993), VND (*Variable Neighborhood Descent*) (Gao et al., 2008).

(2) L'évaluation d'une solution est faite par le calculateur de score (*ScoreDirector*). Le score d'une solution caractérise la qualité d'une solution, et permet donc de savoir si une solution est meilleure qu'une autre. Plus le score est élevé, meilleure est la solution.

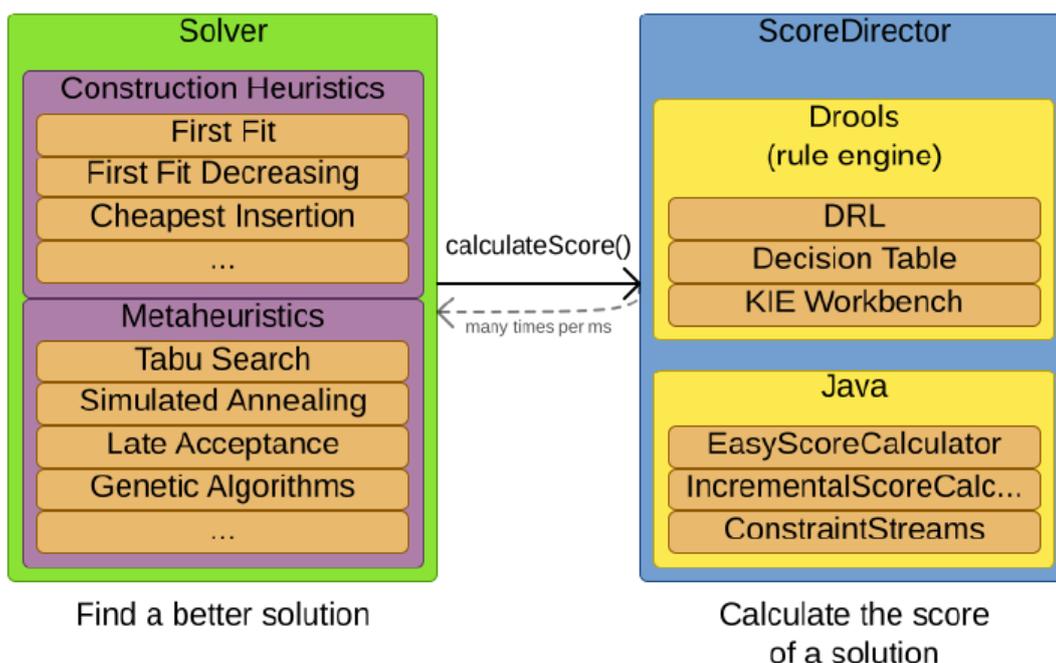


Figure 47 : Aperçu synthétique de mécanismes de résolution du problème par OptaPlanner (OptaPlanner, 2021)

4.2.2. Modélisation du problème

Les sous-sections suivantes utilisent les étapes évoquées précédemment. Nous illustrons ici notre développement de la méthode CSP avec OptaPlanner pour construire le modèle de prise de décision d'un HHCRSP.

Nous spécifions d'abord les classes utiles pour notre de domaine d'étude. Pour chaque classe et ses attributs, nous utilisons une notation afin de faciliter la description des contraintes faite à la section 4.2.3 de ce chapitre. La Figure 48 illustre ce choix de structuration.

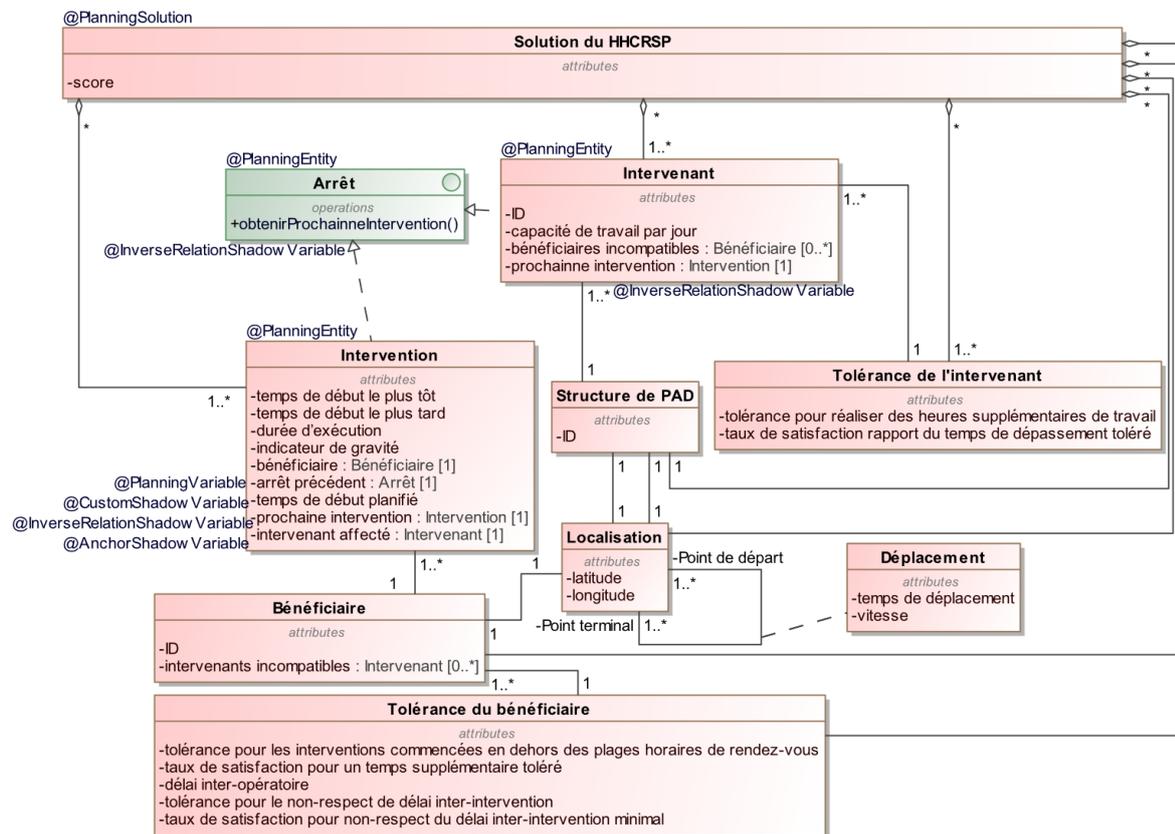


Figure 48 : Modèle du HHCRSP orienté OptaPlanner

4.2.2.1. Classe d'information du problème (*Problem Fact*)

Dans cette catégorie, les classes ne disposent pas de variables de décisions. Nous énumérons les paramètres avec leurs types en Java en les associant aux classes d'information dans le Tableau 13.

Tableau 13 : Classes d'information du modèle du HHCRSP

Nom de classe	Attribut			
	Nom	Type	Notation	Description
Structure de PAD (<i>PAD</i>)	ID	String	id_{pad}	-
Bénéficiaire (<i>P</i>)	ID	String	id_p	-
	Intervenants incompatibles	List<>	$imcomp_p$	Ces intervenants ne peuvent pas exécuter les interventions demandées par les objets de <i>P</i> .
	Délai inter-intervention	Long	δ_p	Comme δ_p dans la méthode exacte

Tolérance de bénéficiaire (<i>TolP</i>)	Tolérance pour les heures supplémentaires d'intervention	Long	α_p	Comme α_p dans la méthode exacte
	Taux de satisfaction pour le temps supplémentaire toléré	Long	α_p^*	Comme α_p^* dans la méthode exacte
	Tolérance pour non-respect des délais	Long	β_p	Comme β_p dans la méthode exacte
	Taux de satisfaction pour non-respect des délais toléré	Long	β_p^*	Comme β_p^* dans la méthode exacte
Tolérance d'intervenant (<i>TolK</i>)	Tolérance pour les heures supplémentaires de travail	Long	γ_k	Comme γ_k dans la méthode exacte
	Taux de satisfaction rapport du temps de dépassement toléré	Long	γ_k^*	Comme γ_k^* dans la méthode exacte
Localisation (<i>LOC</i>)	Altitude	Double	alt_{i^*}, alt_{j^*}	Informations géographiques d'une localisation.
	Longitude	Double	lon_{i^*}, lon_{j^*}	
Déplacement (<i>DEP</i>)	Temps de déplacement	Long	$temp_{i^*j^*}$	Classe relationnelle. Comme $temp_{i^*j^*}$ dans la méthode exacte, il s'agit du temps de déplacement entre deux localisations i^*, j^* .
	Vitesse	Long	vit	20 km/h comme vitesse moyenne estimée en centre-ville pour calculer $temp_{i^*j^*}$

Notons que $temp_{i^*j^*}$ est calculé à vol d'oiseau en fonction des 2 coordonnées terrestres de la classe « Localisation ». Pour déduire $temp_{i^*j^*}$, nous appliquons la distance euclidienne entre deux points i^* et j^* (Danielsson, 1980) comme le montre la relation (30).

$$temp_{i^*j^*} = \frac{\sqrt{(alt_{i^*} - alt_{j^*})^2 + (lon_{i^*} - lon_{j^*})^2}}{vit} \quad (30)$$

4.2.2.2. Classes d'entité de planification (*Planning Entity*, @PlanningEntity)

Il y a trois classes de ce type dans la Figure 48 : « Arrêt », « Intervenant » et « Intervention ». Nous décrivons les classes d'entité de planification dans le Tableau 14. Concernant la relation de dépendance entre les variables de décision principales et les variables de décisions secondaires, nous avons « *arret_i* » qui est à la source de toutes les autres variables secondaires : « *debut_i* », « *prochaC_k* », « *prochaC_i* », « *afpec_i* ». Aussi, nous décrivons les domaines associés aux variables de décision dans le Tableau 15 .

Tableau 14 : Classes d'entités de planification du modèle HHCRSP

Nom de classe	Attribut			
	Nom	Type	Notation	Description
Arrêt	-	-	-	Interface avec la méthode <i>obtenirProchaineIntervention()</i> . Cette méthode est annotée par <i>@InverseRelationShadowVariable</i> . Les classes « Intervention » et « Intervenant » implémentent cette interface arrêt dans un but de synchronisation. La spécification de la méthode permet de faire connaître l'attribut « prochaine intervention » dans ces deux classes. Cette interface est la base de la création d'une tournée pour chaque intervenant, puisque chaque tournée est constituée d'un intervenant et des interventions qu'il réalise.
Intervenant (<i>K</i>)	ID	String	id_k	-
	Charge maximale de travail /jour	Long	ξ_k	Paramètre. Comme ξ_k dans la méthode exacte
	Bénéficiaires incompatibles	List<>	$imcomp_k$	Paramètre. Élimination de l'affectation des objets de <i>K</i> aux interventions demandées par les bénéficiaires incompatibles.

	Prochaine intervention	Objet <Intervention>	$prochaC_k$	Variable d'ombre inversée basée sur la spécification de la méthode $obtenirProchaineIntervention()$.
Intervention (C)	Temps de début au plus tôt	Long	e_i	Paramètre. Comme $[e_i, l_i]$ dans la méthode exacte
	Temps de début au plus tard	Long	l_i	Paramètre. Comme $[e_i, l_i]$ dans la méthode exacte
	Durée d'exécution	Long	$duree_{i^*}$	Paramètre. Comme $duree_{i^*}$ dans la méthode exacte
	Indicateur de gravité	Long	dif_i	Paramètre. Comme dif_i dans la méthode exacte
	Bénéficiaire	Objet <Bénéficiaire>	$benef_i$	Paramètre. Bénéficiaire d'une intervention, une intervention est demandée par un bénéficiaire.
	Arrêt précédent	Objet <Arrêt>	$arret_i$	Variable de décision principale.
	Temps de début planifié	Long	$debut_i$	Variable d'ombre représentant le temps de début de chaque intervention.
	Prochaine intervention	Objet <Intervention>	$prochaC_i$	Variable d'ombre inversée basée sur la spécification de la méthode $obtenirProchaineIntervention()$.
	Intervenant affecté	Objet <Intervenant>	$affec_i$	Variable d'ancre représentant l'intervenant affecté.

Tableau 15 : Domaines des variables de décision du modèle HHCRSP

Nom de classe	Nom de variable	Type de variable	Domaine
Intervenant (K)	Prochaine intervention	Variable d'ombre inversée	$prochaC_k = \{C_1, C_2, \dots, C_r\} \forall k \in K$
Intervention (C)	Arrêt précédent	Variable de décision principale	$arret_i = \{C_1, C_2, \dots, C_r\} \cup \{K_1, K_2, \dots, K_n\} \forall i \in C$
	Temps de début planifié	Variable d'ombre	$debut_i = [0, 1440] \forall i \in C$
	Prochaine intervention	Variable d'ombre inversée	$prochaC_i = \{C_1, C_2, \dots, C_r\} \forall i \in C$
	Intervenant affecté	Variable d'ancre	$affec_i = \{K_1, K_2, \dots, K_n\} \forall i \in C$

4.2.3. Spécification des contraintes

Nous avons choisi le langage DRL (*Drools Rule Language*) de Drools pour coder les contraintes. Ce choix permet de déclarer les contraintes sous forme de règles qui sont séparées du modèle, ce qui permet de facilement ajouter, modifier ou même supprimer des contraintes. Drools fournit aussi le support du calcul incrémental des scores caractérisant l'optimalité d'une solution, sans code supplémentaire. La déclaration de contrainte en Drools se compose de deux parties en logique de premier ordre : la condition « **when** » et la conséquence « **then** ».

Dans la partie « **when** », nous spécifions la condition déclenchant la contrainte et déclarons les paramètres en utilisant la syntaxe « \$ + nom ».

Dans la partie « **then** », nous déclarons les éléments nécessaires en vue du calcul de score. Par exemple, pour calculer le score de la contrainte [7], nous avons besoin des paramètres concernant la tolérance de l'intervenant quant au dépassement de sa capacité maximale de travail journalier : γ_k et γ_k^* . Pour ce faire, il nous suffit d'appeler les méthodes $getTolerance()$ et $getToleranceRate()$ définies dans la classe « Intervenant » pour obtenir γ_k et γ_k^* de chaque intervenant.

Pour notre HHCRSP, nous développons 7 contraintes, 3 contraintes dures et 4 contraintes souples. Étant donné qu'OptaPlanner maximise le score résultant de toutes les contraintes énumérées dans le problème à résoudre, chaque contrainte non satisfaite est affectée d'une valeur négative avec la syntaxe « *scoreHolder.addHardConstraintMatch(kcontext, -1)* » pour les contraintes dures (la valeur est souvent égale à -1), « *scoreHolder.addSoftConstraintMatch (kcontext, -score value)* » pour les contraintes souples.

La contrainte [1] est une contrainte dure qui permet de garantir que si les intervenants arrivent en avance au domicile d'un bénéficiaire, ils doivent attendre l'heure de début au plus tôt de la fenêtre de temps de début d'intervention demandée par le bénéficiaire.

Contrainte 1 : le début d'une intervention avant l'heure de début au plus tôt de la fenêtre de temps bénéficiaire est interdit

```

1  When
2      C (
3      |   $e_i > debut_i$  )
4      )
5  Then
6      scoreHolder.addHardConstraintMatch(kcontext, -1);

```

La contrainte [2] est une contrainte dure pour empêcher le chevauchement temporel des interventions multiples demandées par un bénéficiaire.

Contrainte 2 : le chevauchement entre les interventions multiples demandées par un bénéficiaire est interdit

```

1  When
2      $Intervention_1 : C (
3      |   $debut_i \neq null$ ,
4      |  $Début_1 :  $debut_i$  ,
5      |  $Durée_1 :  $duree_i$ *
6      )
7      $Intervention_2 : C (
8      |   $debut_i \neq null$ ,
9      |  $Début_2 :  $debut_i$ ,
10     |  $Durée_2 :  $duree_i$ *
11     )
12 Then
13     If ($Intervention_1 != $Intervention_2) && $Début_2 < $Début_1 + $Durée_1
14     |  scoreHolder.addHardConstraintMatch(kcontext, -1);
15 End

```

La contrainte [3] est une contrainte dure pour s'assurer qu'il n'existe pas de relation d'incompatibilité entre le bénéficiaire p et l'intervenant k ,

Contrainte 3 : l'affectation d'un intervenant en cas d'incompatibilité avec un bénéficiaire est interdite

```

1  When
2      $Intervenant : K (
3      |  $bénéficiaires_incompatibles :  $imcomp_k$ 
4      )
5      $Bénéficiaire : P (
6      |  $Intervenant_incompatibles :  $imcomp_p$ 
7      )
8      $Intervention : C (
9      |   $afpec_i == \$Intervenant$ ,
10     |   $benef_i \text{ memberOf } imcomp_k$ ,
11     |   $afpec_i \text{ memberOf } imcomp_p$ 
12     )
13 Then

```

La contrainte [4] est une contrainte souple dont l'objectif est d'équilibrer les difficultés de tournées des intervenants. Pour ce faire, nous appliquons une stratégie de score consistant à maximiser l'écart négatif entre la difficulté totale de la tournée de chaque intervenant ($\$Difficulté_tournée$) et la difficulté moyenne de chaque intervenant ($\$Difficulté_totale / \$Nombre_Intervenant$).

Contrainte 4 : équilibrage au mieux des difficultés de tournées des intervenants

```

1  When
2      $Intervenant : K (
3      | $id : idk )
4      )
5      accumulate (
6      | $Intervention : C (
7      | | affecti == $Intervenant,
8      | | $Difficulté : difi
9      | ),
10     | $Difficulté_tournée : sum($difficulté) // somme des indicateurs de difficulté par tournée
11     )
12     $Nombre_Intervenant : Number() from accumulate (
13     | $Intervenant : K ( ),
14     | Count ($Intervenant) // calculer le nombre d'intervenants
15     )
16     accumulate (
17     | $Intervention : C (
18     | | $Difficulté : difi
19     | ),
20     | $Difficulté_totale : sum($Difficulté) // somme des indicateurs de difficulté totaux
21     )
22  Then
23     scoreHolder.addSoftConstraintMatch (kcontext,
24     - Math.abs($Difficulté_tournée - $Difficulté_totale / Nombre_Intervenant )

```

À l'identique des contraintes (20) à (22) dans la méthode exacte, les contraintes [5] à [7] sont ici des contraintes souples représentant la satisfaction des contraintes temporelles avec les tolérances personnalisées selon les profils individuels des acteurs. La fonction de régression linéaire est encore appliquée, mais ici pour modéliser la relation entre le score et le temps de dépassement (cf. la Figure 49). Notons que si le temps de dépassement de ces 3 contraintes est inférieur à la tolérance concernée ($\alpha_p, \beta_p, \gamma_k$), la pente de la fonction linéaire étant égale à $(100 - \alpha_p^* \vee \beta_p^* \vee \gamma_k^*)$, plus l'acteur est sensible au dépassement du temps souhaité (plus la valeur de $\alpha_p^* \vee \beta_p^* \vee \gamma_k^*$ est faible), plus la pente de la fonction est élevée. Pour tenir compte de l'insatisfaction des acteurs quant aux résultats de la planification sur les 3 contraintes temporelles en cas de dépassement de leurs tolérances, la pente de la deuxième partie de cette fonction linéaire est 10 fois plus grande que la première partie. Le fait de contrôler la pente renforce également la fonction linéaire appliquée avec la méthode exacte (cf. la Figure 42), ce qui permet d'éviter d'attribuer des valeurs aux variables de décision relatives à ces 3 contraintes au-delà de la tolérance de l'utilisateur lors de la recherche de la solution optimale globale.

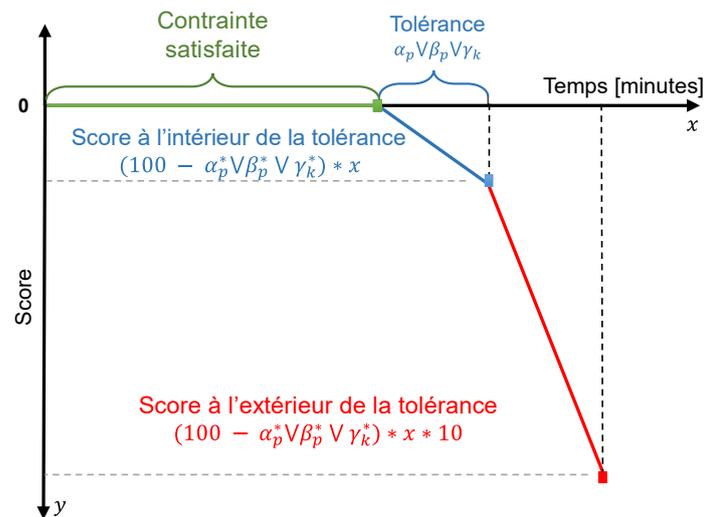


Figure 49 : Fonction de régression linéaire pour modéliser la relation entre le score et le temps de dépassement des 3 contraintes temporelles

Les contraintes [5] à [7] calculent donc respectivement le score par rapport au respect de la fenêtre de début d'une intervention demandée (contrainte 5), au respect du délai inter-intervention demandé par un bénéficiaire (contrainte 6) et au respect de la charge de travail maximale d'un intervenant (contrainte 7).

Contrainte 5 : Respect au mieux des fenêtres de temps demandées par les bénéficiaires

```

1  When
2      $Intervention : C (
3      |    $l_i < debut_i$ ,
4      |   $Fenêtre_temps_tard :  $l_i$ ,
5      |   $Début :  $debut_i$ 
6      )
7  Then
8      long Temps_dépassé = $Début - $Fenêtre_temps_tard
9      long Score = 100 -  $\alpha_p^*$  //  $\alpha_p^*$  est obtenu par appeler la méthode getOvertimeToleranceRate()
10     If (Temps_dépassé  $\leq \alpha_p$ ) //  $\alpha_p$  est obtenu par appeler la méthode getOvertimeTolerance()
11     |   scoreHolder.addSoftConstraintMatch (kcontext, - Score*Temps_dépassé)
12     Else
13     |   scoreHolder.addSoftConstraintMatch (kcontext, - Score*Temps_dépassé*10)
14     End

```

Contrainte 6 : Respect au mieux des délais inter-intervention demandés par les bénéficiaires

```

1  When
2      $Intervention_1 : C (
3      |    $debut_i \neq null$ ,
4      |   $bénéficiaire_1 :  $benef_i$ ,
5      |   $Début_1 :  $debut_i$ ,
6      |   $Durée_1 :  $duree_i^*$ 
7      )
8      $Intervention_2 : C (
9      |    $debut_i \neq null$ ,
10     |    $debut_i > \$Début_1$  // assurer $Intervention_2 est demandée après $Intervention_1
11     |   $bénéficiaire_2 :  $benef_i$ ,
12     |   $Début_2 :  $debut_i$ 
13     )
14  Then

```

```

15     long Début_2_désiré = $Début_1 + $Durée_1 +  $\delta_p$  ;
16     // Début de $Intervention_2 au plus tôt.  $\delta_p$  est obtenu par appeler la méthode getInteroperation()
17     If ($Début_2 < Début_2_désiré) // surcharge d'un intervenant
18     |   long Non_respect_Interop = Début_2_désiré - $Début_2 ;
19     |   long Score = 100 -  $\beta_p^*$  //  $\beta_p^*$  est obtenu par appeler la méthode getInteropToleranceRate()
20     |   If (Non_respect_Interop  $\leq$   $\beta_p$ ) //  $\gamma_k$  est obtenu par appeler la méthode getInteropTolerance()
21     |   |   scoreHolder.addSoftConstraintMatch (kcontext, - Score* Non_respect_Interop)
22     |   |   Else
23     |   |   |   scoreHolder.addSoftConstraintMatch (kcontext, - Score* Non_respect_Interop *10)
24     |   |   End
25     End

```

Contrainte 7 : Respect au mieux de la charge de travail maximale d'un intervenant effectuant une tournée

```

1  When
2      $Intervenant :K (
3      |   $Charge_max :  $\xi_k$ 
4      )
5      accumulate (
6      |   $Intervention_O : C (
7      |   |   afeci == $Intervenant,
8      |   |   $Durée : dureei*
9      |   |   ),
10     |   $Durée_totale : sum($Durée) // temps d'opération totaux par intervenant
11     )
12     accumulate (
13     |   $Intervention_T : C (
14     |   |   afeci == $Intervenant,
15     |   |   arreti != null,
16     |   |   i*  $\leftarrow$  arreti.getBénéficiaire().getLocalisation(),
17     |   |   j*  $\leftarrow$  this.getBénéficiaire().getLocalisation(),
18     |   |   $Temps_trajet : tempi*j*
19     |   |   )
20     |   $Temps_trajet_total : sum($Temps_trajet) // temps de trajet (sauf le retour à la structure)
21     )
22     $Intervention_dernière : C (
23     |   afeci == $Intervenant,
24     |   attreti != null && prochaCi == null,
25     |   i*  $\leftarrow$  attreti.getIntervenant().getStructureDePAD().getLocalisation(),
26     |   j*  $\leftarrow$  this.getBénéficiaire().getLocalisation(),
27     |   $Temps_retour : tempi*j* // temps de trajet de retour à la structure
28     )
29  Then
30      long Temps_travail = $Durée_totale + $Temps_trajet_total + $Temps_retour ;
31      If (Temps_travail > $Charge_max) // surcharge d'un intervenant
32      |   long Temps_dépassé = Temps_travail - $Charge_max ;
33      |   long Score = 100 -  $\gamma_k^*$  //  $\gamma_k^*$  est obtenu par appeler la méthode getOvertimeWorkloadToleranceRate()
34      |   If (Temps_dépassé  $\leq$   $\gamma_k$ ) //  $\gamma_k$  est obtenu par appeler la méthode getOvertimeWorkloadTolerance()
35      |   |   scoreHolder.addSoftConstraintMatch (kcontext, - Score*Temps_dépassé)
36      |   |   Else
37      |   |   |   scoreHolder.addSoftConstraintMatch (kcontext, - Score*Temps_dépassé*10)
38      |   |   End
39      End

```

4.2.4. Résolution du problème

OptaPlanner permet de définir la condition d'arrêt des calculs, de choisir l'heuristique de construction de la solution initiale et l'algorithme d'optimisation (à base de métaheuristiques). Et cela, afin de trouver la meilleure configuration algorithmique (combinaison entre les

heuristiques de construction et les métaheuristiques) pour un problème donné. L'outil propose une configuration appelée « benchmark » permettant de tester un problème résolu par des algorithmes différents. Ce benchmark est très utile surtout pour comparer les heuristiques ou les métaheuristiques qui doivent être utilisées pour le problème à résoudre. Le processus de configuration du solveur est longuement discuté dans (Macik, 2016). Pour cet auteur qui teste le problème d'affectation des tâches, le résultat met en évidence que la meilleure heuristique de construction est le *First Fit*. Dans (Rios de Souza and Martins, 2020), le *First Fit* est aussi utilisé pour construire la solution initiale. Nous avons donc choisi ce *First Fit* comme algorithme de construction de la solution initiale durant le process de benchmark.

Pour la configuration du benchmark, nous respectons la configuration par défaut d'OptaPlanner : 5 métaheuristiques sont utilisées : *Hill Climbing*, *Late Acceptance*, *Great deluge*, TS, et VND. Aucune personnalisation n'a été apportée aux paramètres par défaut de chacun de ces algorithmes, ceci dans le but de comparer les résultats obtenus avec les mêmes conditions de départ. De plus, deux critères ont été utilisés comme condition d'arrêt : soit lorsque le score global atteint la valeur zéro (le score maximal), soit lorsque le temps de calcul dépasse 8 minutes sans gain obtenu sur la meilleure solution en cours.

Suite à l'exécution de la solution pour le cas d'utilisation « L_s130_p90_mp24_k19 » (présenté dans la section 5), le cas le plus grand et le plus complexe (cf. le Tableau 17), nous avons constaté que l'algorithme TS présente les meilleurs résultats, mais comme illustré dans la Figure 50, tous les algorithmes permettent d'obtenir une solution faisable (le score de la partie « contrainte dure » = 0). Pour la partie « contrainte souple », le score de la solution générée par un TS est le plus élevé (-370) parmi tous les autres résultats (cf. la Figure 51). Ainsi, dans la section 5, nous adoptons le TS en tant qu'algorithme de résolution pour effectuer une analyse comparative avec la solution obtenue avec la méthode exacte.

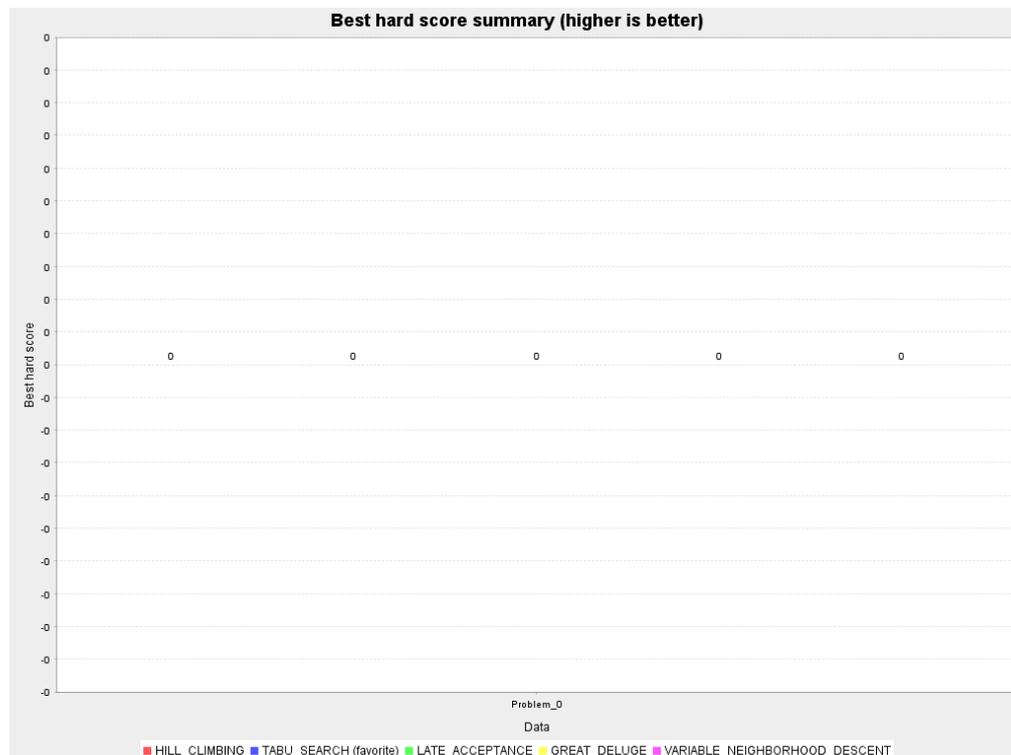


Figure 50 : Résultat du « benchmark » pour le cas d'utilisation « L_s130_p90_mp24_k19 » : le score de la partie « contrainte dure »

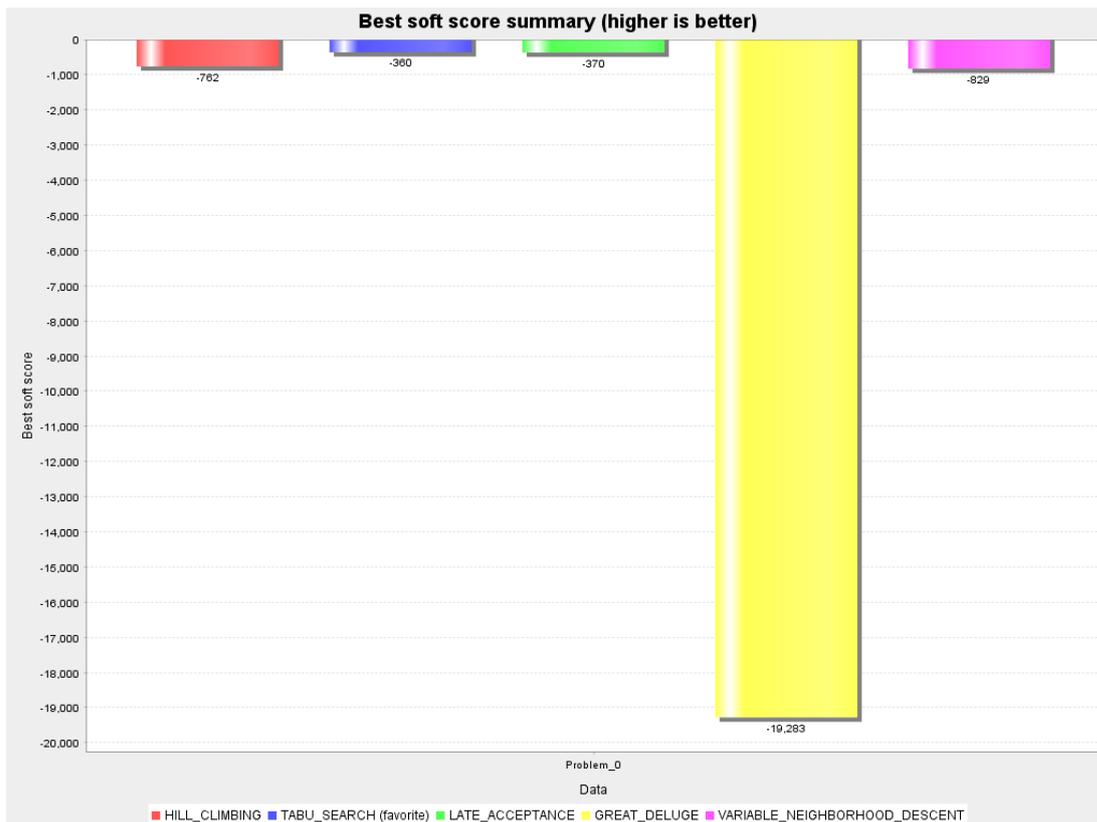


Figure 51 : Résultat du « benchmark » pour le cas d'utilisation « L_s130_p90_mp24_k19 » : le score de la partie « contrainte souple »

5. Expérimentation : méthode exacte vs méthode approchée

5.1. Génération de jeux de test

Le jeu de données a été extrait de la base de données d'un produit opérationnel venant d'un client de Berger-Levrault. Nous avons généré 13 instances et les avons soumises à CPLEX et à OptaPlanner pour réaliser l'expérimentation finale. Afin de permettre une comparaison de qualité entre deux solutions générées à partir de deux solveurs différents, nous traduisons la représentation du résultat de la méthode approchée (format « score » : 0hard/156soft, par exemple), de sorte que la solution de la méthode approchée soit représentée par la valeur scalaire de la fonction objectif de la méthode exacte (format « fonction objectif » illustré dans l'équation (1) avec le sens donné à partie entière et partie décimale : 43,9966 pour le cas ayant 43 interventions à planifier, par exemple). Ce processus de traduction est mis en œuvre par une fonctionnalité de notre environnement de test (Zhang et al., 2020b) (cf. la Figure 52).

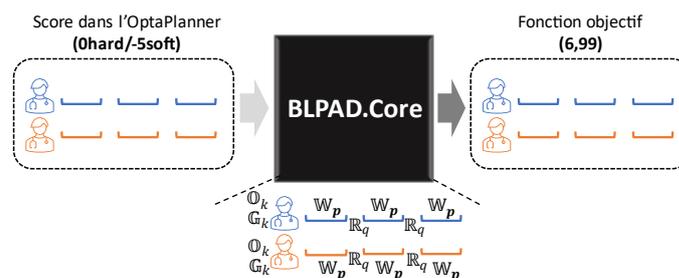


Figure 52 : Processus de traduction du score dans la méthode approchée vers la fonction objectif dans la méthode exacte

Nous testons le temps de calcul et la qualité de la solution de coordination, incluant les 4 composantes de la fonction objectif reflétant la satisfaction des acteurs. Le test est effectué sur un ordinateur avec un processeur Intel(R) Core(TM) i7-7500U à 2,90 GHz et 16,0 Go de RAM sous système d'exploitation Windows 10.

Pour tous les cas d'utilisation, nous générons aléatoirement dans les limites d'une amplitude donnée les paramètres de satisfaction des acteurs et la difficulté des interventions. Le Tableau 16 illustre ce choix des paramètres « satisfaction » et « difficulté ».

Tableau 16 : Paramètre de « satisfaction » des acteurs et de « difficulté » des interventions

Point de vue	Paramètre	Explication	Valeur minimum	Valeur maximum	Commentaire
Intervenant	γ_k	Tolérance de l'intervenant k lorsque la charge de travail maximale est dépassée.	50	100	En minutes
	γ_k^*	Taux de satisfaction de l'intervenant k par rapport à γ_k	30	50	En pourcentage
	ξ_k	Charge de travail maximale de l'intervenant k	600	600	En minutes
Bénéficiaire	δ_p	Délai inter-opérateur entre 2 interventions demandées par le bénéficiaire p	20	60	En minutes
	α_p	Tolérance du bénéficiaire p par rapport au début des interventions demandées.	30	60	En minutes
	α_p^*	Taux de satisfaction du bénéficiaire p par rapport à α_p	30	50	En pourcentage
	β_p	Tolérance du bénéficiaire p une fois que le δ_p n'est pas respecté	1	20	En minutes
	β_p^*	Taux de satisfaction du bénéficiaire p par rapport à β_p .	30	50	En pourcentage
	df_i	Difficulté pour opérer une intervention	1	6	

Ensuite, nous catégorisons les instances de cas en 4 groupes : S (taille petite, 10-30 interventions), M (taille moyenne, 35-50 interventions), ML (taille moyenne-grande, 55-70 interventions) et L (taille grande, 110 et 130 interventions). Réutilisant nos règles de nommage, l'indice s indique le nombre d'interventions demandées, l'indice p indique le nombre de bénéficiaires, l'indice mp indique le nombre de bénéficiaires ayant des demandes multiples et l'indice k représente le nombre d'intervenants enregistrés. Par exemple, « S_s13_p6_mp6_k3 » correspond à un cas de petite taille (S), avec 13 interventions demandées (s13) par 6 bénéficiaires (p6) qui ont tous des demandes multiples (mp6), ces interventions étant effectuées par 3 intervenants (k3). De plus, nous avons extrait de CPLEX le nombre total de variables (les variables binaires, les variables entières et réelles) et le nombre total de contraintes pour chaque problème à résoudre afin d'avoir une vision globale sur la difficulté potentielle de chaque problème (cf. Tableau 17). En général, plus le nombre de variables et de contraintes est élevé, plus le problème est difficile à résoudre par le solveur.

Tableau 17 : Nombre de variables et de contraintes pour chaque cas d'utilisation

Nom de cas	Nombre de variables binaires	Nombre de variables entières et réelles	Nombre de contraintes
S_s13_p6_mp6_k3	350	82	275
M_s37_p21_mp11_k6	6268	189	1670
M_s40_p25_mp10_k6	7376	201	1898
M_s43_p23_mp12_k6	8338	224	2145
M_s47_p26_mp13_k7	11575	244	2568
M_s50_p26_mp13_k7	13212	267	2872
ML_s55_p28_mp14_k4	9128	294	3035

ML_s55_p28_mp14_k8	19311	310	3513
ML_s60_p31_mp18_k9	23922	315	4090
ML_s61_p32_mp17_k8	22002	315	4062
L_s81_p56_mp13_k11	50218	433	6804
L_s110_p74_mp22_k15	120428	573	12050
L_s130_p90_mp24_k16	175744	662	16107

5.2. Analyse comparative de la performance globale de la méthode exacte et de la méthode approchée

Précisons d'emblée que l'implantation du même modèle sur Optaplanner et CPLEX nous a conduit à faire des adaptations. L'introduction d'un système de pénalité qui dicte la progression d'une méthode de résolution métaheuristique sous OptaPlanner fait qu'il existe une limite dans cette volonté de comparer.

Les résultats de nos expériences sont présentés dans le Tableau 18. Nous y reportons la performance globale en termes de temps de calcul en secondes (CPU), l'écart entre la partie « qualité » des Fonctions Objectifs (FO) des solutions générées par les deux méthodes (Ecart-QFO, cf. l'équation (31)), et l'écart pour la partie « qualité » entre la borne supérieure (UB) et FO de la méthode approchée (Ecart-UB, cf. l'équation (32)). UB est la borne supérieure donnée par le meilleur nœud qui est extrait de CPLEX lors du calcul de la solution optimale pour chaque cas d'utilisation par la méthode exacte

$$\text{Ecart - QFO} = \frac{(\text{FO}(\text{Exacte}) - \text{FO}(\text{Approchée})) - r}{\text{FO}(\text{Exacte}) - r} * 100\% \quad (31)$$

$$\text{Ecart - UB} = \frac{(\text{UB} - \text{FO}(\text{Approchée})) - r}{\text{UB} - r} * 100\% \quad (32)$$

Concernant le critère d'arrêt de calcul, pour la méthode exacte, nous fixons la limite supérieure du temps de calcul dans CPLEX à 36000 secondes. Concernant la conservation du résultat optimal en respectant le paramètre de configuration de base dans CPLEX, nous laissons CPLEX arrêter l'optimisation lorsque la différence en termes de valeur OF entre la solution courante et UP est inférieure à 0,6%. Pour la méthode approchée, la condition d'arrêt du calcul dans OptaPlanner correspond à une absence d'amélioration durant 8 minutes d'optimisation. En déduisant ces 8 minutes de la totalité du temps de calcul, nous pouvons calculer le temps nécessaire pour obtenir le meilleur résultat.

Tableau 18 : Résultat général de l'expérimentation

Nom de cas	Méthode exacte (CPLEX)			Méthode approchée (La recherche tabou -OptaPlanner)			Ecart -QFO	Ecart -UB
	UB	FO	CPU(s)	FO	Score	CPU(s)		
S_s13_p6_mp6_k3	13.9694	13.9684	1.2	13.9193	0hard/-4261soft	73.5	4.91%	5.01%
M_s37_p21_mp11_k6	37.9991	37.9954	10.6	37.9974	0hard/-5soft	11.6	-0.20%	0.17%
M_s40_p25_mp10_k6	40.9992	40.9984	20.4	40.9984	0hard/-5soft	10.8	0.00%	0.08%
M_s43_p23_mp12_k6	43.9989	43.9959	20.8	43.9978	0hard/-4soft	13.9	-0.19%	0.11%
M_s47_p26_mp13_k7	47.9988	47.9966	11.1	47.9976	0hard/-5soft	12.8	-0.11%	0.12%
M_s50_p26_mp13_k7	50.9993	50.9949	134.1	50.9990	0hard/-2soft	12.3	-0.41%	0.03%
ML_s55_p28_mp14_k4	56.0000	55.9948	7799.4	56.0000	0hard/0soft	22.7	-0.53%	0.00%
ML_s55_p28_mp14_k8	56.0000	55.9952	280.0	56.0000	0hard/0soft	33.2	-0.49%	0.00%
ML_s60_p31_mp18_k9	60.9994	60.9934	149.2	60.9971	0hard/-8soft	11.8	-0.38%	0.23%
ML_s61_p32_mp17_k8	62.0000	61.9954	236.797	62.0000	0hard/0soft	13.6	-0.23%	0.00%
L_s81_p56_mp13_k11	81.9989	-	36000	81.9907	0hard/-4491soft	677.7	-	0.82%

L_s110_p74_mp22_k15	110.9986	-	36000	110.9964	0hard/-2254soft	627.7	-	0.22%
L_s130_p90_mp24_k16	130.9999	-	36000	130.9990	0hard/-360soft	676.3	-	0.09%

Pour la méthode exacte, le résultat démontre la performance de notre modèle pour obtenir la solution optimale (écart pour la partie « qualité » entre FO de la méthode exacte et UP ≈ 0 , les résultats détaillés sont présentés dans (Zhang et al., 2021a)). Le temps de calcul est satisfaisant même avec une dimension du problème allant jusqu'à la taille moyenne-grande (ML_s61_p32_mp17_k8). Cependant, pour les cas de plus grande taille, la méthode exacte ne peut pas trouver de solution au bout du temps de calcul maximal et CPLEX s'arrête sur un échec.

Pour la méthode approchée, une solution est toujours trouvée pour chaque cas d'utilisation, même pour les cas de grande taille. Concernant la qualité de chaque solution, nous constatons que la méthode approchée est presque compétitive avec la méthode exacte par analyse des écarts (Ecart (FO)) quand ils sont disponibles (10 premiers cas). Pour le temps de calcul de la méthode approchée, celui des cas de taille S et ML sont courts et stables (21,62 secondes en moyenne), tandis que pour les trois grandes tailles, le temps de calcul est plus long (563,9 secondes en moyenne) mais la qualité des résultats est satisfaisante, voir L_s81_p56_mp13_k11 : 81.9907 obtenant une la borne supérieure (UB) de 81.9989; L_s110_p74_mp22_k15 : 110.9964 (UB=110.9986) ; L_s130_p90_mp24_k16 (UB=130.9999).

Le Tableau 19 présente les résultats détaillés des 4 composantes de la satisfaction pour chaque cas : la satisfaction de la planification sur la fenêtre de temps, le respect du délai inter-intervention, le respect de capacité disponible et enfin l'équilibrage de pénibilité des tournées. Nous indiquons d'abord la valeur maximale que chaque composant peut atteindre dans la colonne « borne sup. ». Pour chaque méthode de résolution, nous présentons les valeurs obtenues pour chaque composant dans la colonne « coeff » et puis nous calculons le taux de satisfaction (la colonne « Satis. % ») pour chaque composant à l'aide de l'équation (33). De nouveau, afin de comparer la qualité des solutions obtenues par les 2 méthodes pour chaque composant, nous calculons l'écart de taux de satisfaction entre les résultats des 2 méthodes (dans la colonne « Ecart % ») à l'aide de l'équation (34).

$$\text{Satis. \%} = \frac{\text{coeff}}{\text{Borne sup.}} * 100\% \quad (33)$$

$$\text{Ecart \%} = \frac{\text{Satis. \%}(\text{Exacte}) - \text{Satis. \%}(\text{Approchée})}{\text{Satis. \%}(\text{Exacte})} * 100\% \quad (34)$$

Tableau 19 : Résultat expérimental : 4 composantes de la fonction objectif représentant la satisfaction des acteurs

Nom de cas	Satisfaction de la planification sur la fenêtre de temps						Respect de l'inter-intervention					
	Borne sup.	Exacte		Approchée		Ecart %	Borne sup.	Exacte		Approchée		Ecart %
		Coeff	Satis. %	coeff	Satis. %			Coeff	Satis. %	coeff	Satis. %	
S_s13_p6_mp6_k3	6	5.4727	91.2%	5.7690	96.2%	-5.4%	6	6	100%	5.1250	85.4%	14.6%
M_s37_p21_mp11_k6	21	20.9646	99.8%	21	100%	-0.2%	11	11	100%	11	100%	0.0%
M_s40_p25_mp10_k6	25	25	100%	25	100%	0.0%	10	10	100%	10	100%	0.0%
M_s43_p23_mp12_k6	23	23	100%	23	100%	0.0%	12	12	100%	12	100%	0.0%
M_s47_p26_mp13_k7	26	26	100%	26	100%	0.0%	13	13	100%	13	100%	0.0%
M_s50_p26_mp13_k7	26	25.9368	100%	26	100%	-0.2%	13	13	100%	13	100%	0.0%
ML_s55_p28_mp14_k4	28	27.7387	99.1%	28	100%	-0.9%	14	14	100%	14	100%	0.0%
ML_s55_p28_mp14_k8	28	28	100%	28	100%	0.0%	14	14	100%	14	100%	0.0%
ML_s60_p31_mp18_k9	31	30.8425	99.5%	31	100%	-0.5%	18	18	100%	18	100%	0.0%
ML_s61_p32_mp17_k8	32	32	100%	32	100%	0.0%	17	17	100%	17	100%	0.0%
L_s81_p56_mp13_k11	56	-	-	55.3992	98.9%	-	13	-	-	12.9067	99.3%	-
L_s110_p74_mp22_k15	74	-	-	73.8813	100%	-	22	-	-	21.9447	100%	-
L_s130_p90_mp24_k16	90	-	-	90	100%	-	24	-	-	23.8500	99.4%	-
Nom de cas	Respect de capacité disponible						Equilibrage de pénibilité des tournées					
	Borne sup.	Exacte		Approchée		Ecart %	Borne sup.	Exacte		Approchée		Ecart %
		Coeff	Satis. %	coeff	Satis. %			Coeff	Satis. %	coeff	Satis. %	
S_s13_p12_mp1_k3	3	3	100%	3	100%	0%	3	2.9592	98.6%	2.6531	88.4%	10.3%
M_s37_p21_mp11_k6	6	6	100%	6	100%	0%	6	5.8321	97.2%	5.8855	98.1%	-0.9%
M_s40_p25_mp10_k6	6	6	100%	6	100%	0%	6	5.9270	98.8%	5.9270	98.8%	0.0%
M_s43_p23_mp12_k6	6	6	100%	6	100%	0%	6	5.8052	96.8%	5.8961	98.3%	-1.6%
M_s47_p26_mp13_k7	7	7	100%	7	100%	0%	7	6.8176	97.4%	6.8742	98.2%	-0.8%
M_s50_p26_mp13_k7	7	7	100%	7	100%	0%	7	6.7935	97.0%	6.9457	99.2%	-2.2%
ML_s55_p28_mp14_k4	4	4	100%	4	100%	0%	4	4	100%	4	100%	0.0%
ML_s55_p28_mp14_k8	8	8	100%	8	100%	0%	8	7.7200	96.5%	8	100%	-3.6%
ML_s60_p31_mp18_k9	9	9	100%	9	100%	0%	9	8.7136	96.8%	8.8058	97.8%	-1.1%
ML_s61_p32_mp17_k8	8	8	100%	8	100%	0%	8	7.7037	96.3%	8	100%	-3.8%
L_s81_p56_mp13_k11	11	-	-	11	100%	-	11	-	-	10.8511	98.6%	-
L_s110_p74_mp22_k15	15	-	-	15	100%	-	15	-	-	14.7209	98.1%	-
L_s130_p90_mp24_k16	16	-	-	16	100%	-	16	-	-	16.0000	100%	-

Cette démonstration met en évidence que, sauf dans le cas d'utilisation S_s13_p12_mp1_k3 où la méthode exacte est plus performante pour le respect du délai inter-intervention (Ecart % = 14,6%) et pour l'équilibrage de la pénibilité des tournées (Ecart % = 10,3%), la méthode approchée à base de recherche Tabou d'OptaPlanner présente pratiquement la même optimalité que la méthode exacte à base de CPLEX (Ecart % < 1%). Pour les trois cas d'utilisation où la méthode exacte ne parvient pas à trouver de résultats dans le délai imparti, la méthode approchée présente également une excellente performance dans chaque composante (Satis. % ∈ [98%, 100%]).

5.3. Analyse comparative sur le cas ML_s55_p28_mp14_k4

La performance globale nous permet de donner un aperçu de la qualité des résultats obtenus par les deux méthodes. Bien que nous ayons pu examiner numériquement les résultats correspondant à chaque cas d'utilisation, il est difficile pour les utilisateurs finaux, notamment les professionnels non experts en mathématiques tels que les coordinateurs d'une structure de PAD, de comprendre les résultats numériques présentés dans le Tableau 18 et le Tableau 19. Là encore, cette analyse vise à vérifier le respect de 4 contraintes liées à la satisfaction des acteurs. La Figure 53 illustre la localisation des lieux de vie des bénéficiaires (à Albi). Dans ce cas d'utilisation, 55 interventions sont demandées par 28 bénéficiaires dont 14 bénéficiaires

demandent plus de 2 interventions, et 4 intervenants sont disponibles pour réaliser l'ensemble de demandes.

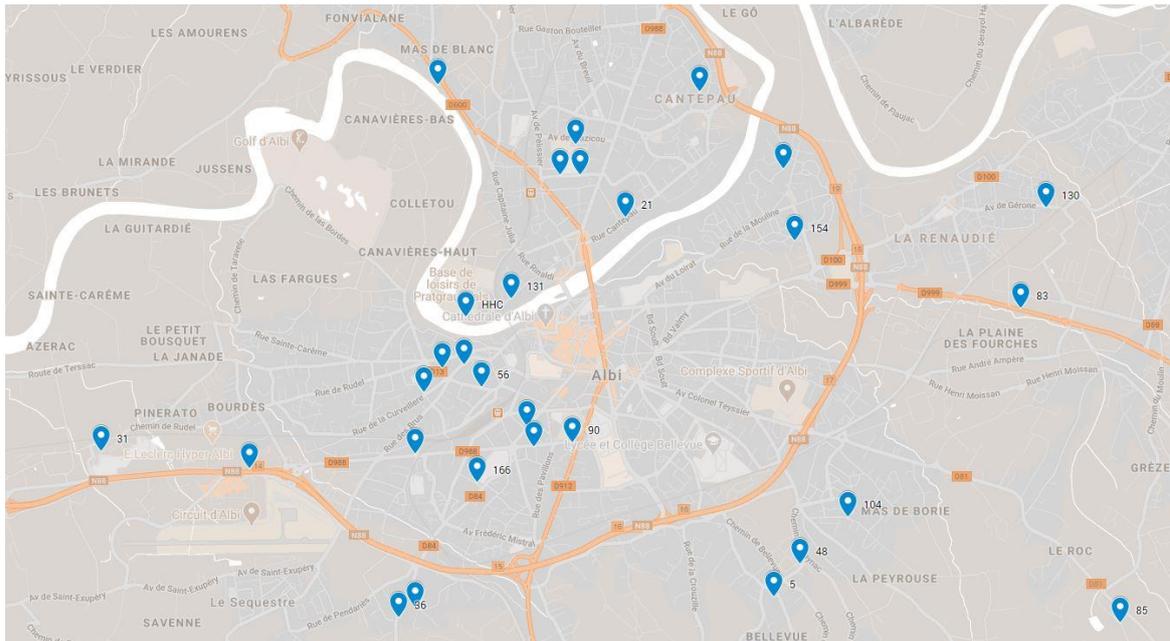


Figure 53 : Localisation des lieux de vie des bénéficiaires du cas ML_s55_p28_mp14_k4

5.3.1. Satisfaction de la planification sur la fenêtre de temps

Pour la meilleure solution trouvée, la Figure 54 et la Figure 55 représentent, sous la forme d'un diagramme de Gantt, la succession des interventions de chaque tournée avec les deux méthodes. Cela permet de valider la satisfaction de la planification sur la fenêtre de temps demandée. Notons que chaque ligne correspond à une intervention caractérisée par une heure de début (variable $debut_i$) et une heure de fin ($debut_i + duree_i$) planifiée. Chaque couleur correspondant à une tournée réalisée par un intervenant. Les 3 points autour de chaque intervention représentent la fenêtre de temps demandée par le bénéficiaire (paramètres e_i et l_i) et le niveau de tolérance concernant le retard de l'intervenant (la durée entre le point 2 et 3, le paramètre α_p). Si chaque intervention commence entre le point 1 et 3, le bénéficiaire est satisfait du résultat, même s'il y a un peu de retard, car celui-ci reste dans la tolérance du bénéficiaire (la satisfaction du bénéficiaire diminue un peu et cela dépend du taux de satisfaction personnalisé α_p^* du bénéficiaire au point 3). Si chaque intervention commence en dehors du temps correspondant aux 3 points, cela entraînera le mécontentement du bénéficiaire dû à un retard de l'intervenant.

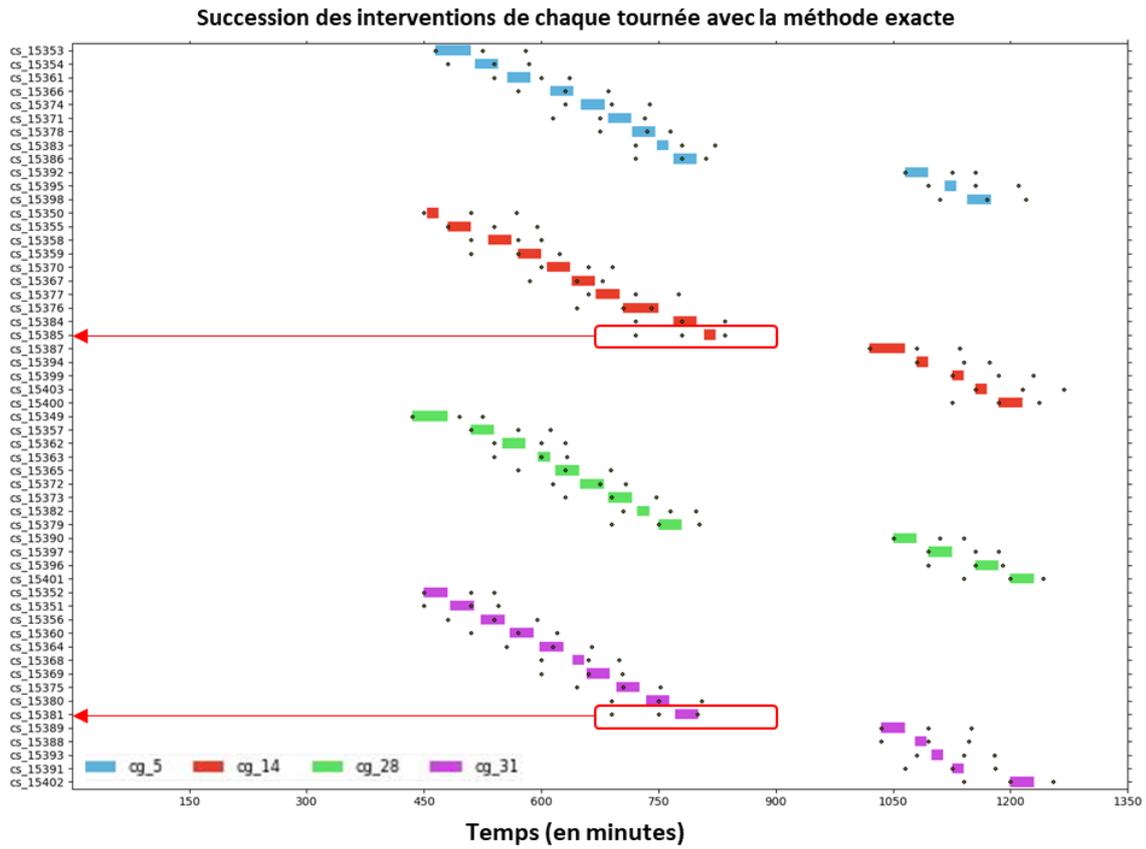


Figure 54 : Succession des interventions de chaque tournée avec la méthode exacte

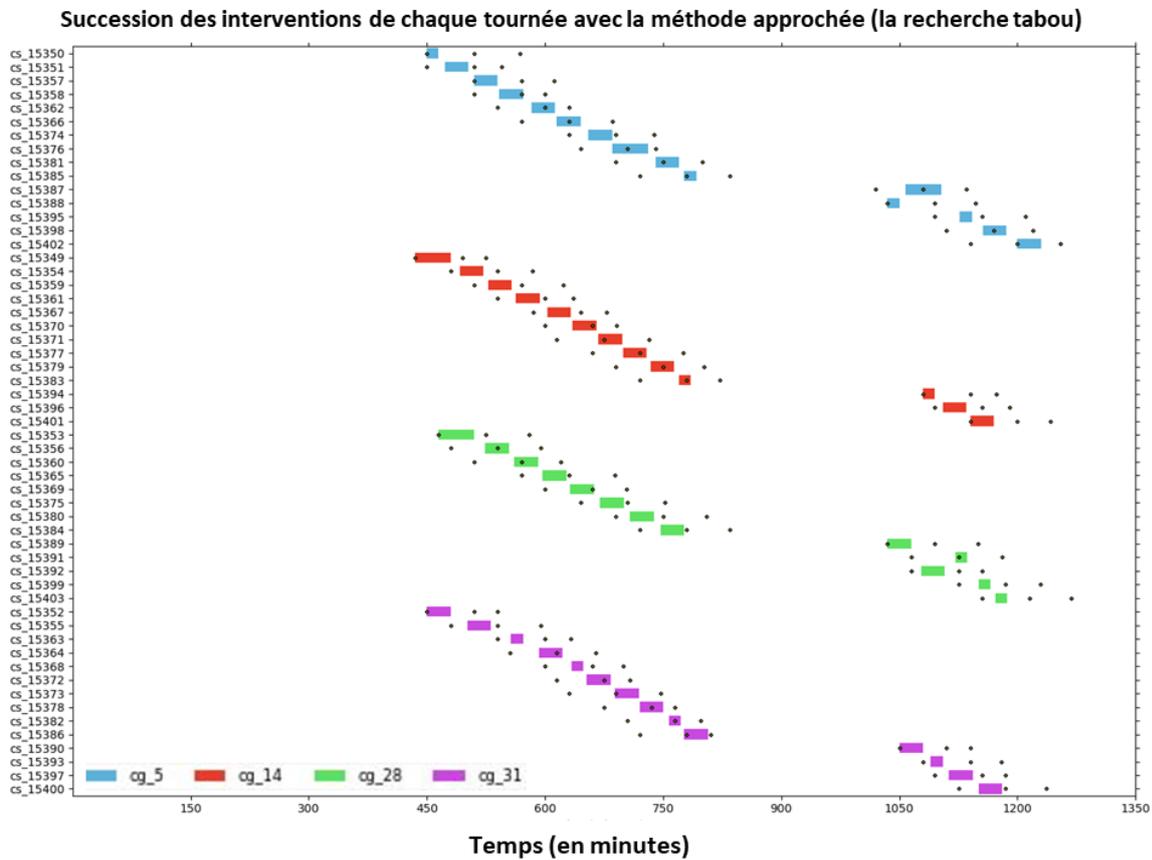


Figure 55 : Succession des interventions de chaque tournée avec la méthode approchée

Les 2 solutions respectent les fenêtres de temps requises par les bénéficiaires puisque chaque intervention commence à l'intérieur du temps correspondant aux 2 premiers points, à l'exception de « les interventions 15385 et 15381 » planifiée par la méthode exacte (les interventions en cercle rouge dans la Figure 54) qui commence entre la tolérance de retard du bénéficiaire.

5.3.2. Respect du délai entre interventions

La contrainte de délai entre interventions est activée uniquement lorsque le bénéficiaire demande plusieurs interventions. La Figure 56 et la Figure 57 montrent les interventions demandées par les « bénéficiaires à interventions multiples ». Chaque ligne est le résultat de la programmation d'une intervention, et sa couleur représente celle du bénéficiaire qui a demandé cette intervention. Les 3 points qui suivent chaque intervention représentent le délai inter-intervention attendu du bénéficiaire et la tolérance pour le non-respect du délai inter-intervention. La différence de temps entre les points 1 et 3 est le délai inter-intervention souhaité (paramètre δ_p), et celle entre les points 2 et 3 est la tolérance (paramètre β_p).

Si l'intervention commence après le point 3 associé à l'intervention précédente parmi les interventions multiples demandées par le même bénéficiaire, ce bénéficiaire est satisfait à 100% du résultat. Si l'intervention commence entre le point 2 et 3 liés à l'intervention précédente, le bénéficiaire est moins satisfait (la satisfaction dépend du taux de satisfaction personnalisé β_p^* au point 2) du résultat, mais cela reste à l'intérieur de son niveau de tolérance. Dans le cas contraire, si l'intervention commence entre le point 1 et 2 correspondant à l'intervention précédente, le bénéficiaire n'est pas satisfait du résultat avec un taux de satisfaction très faible.

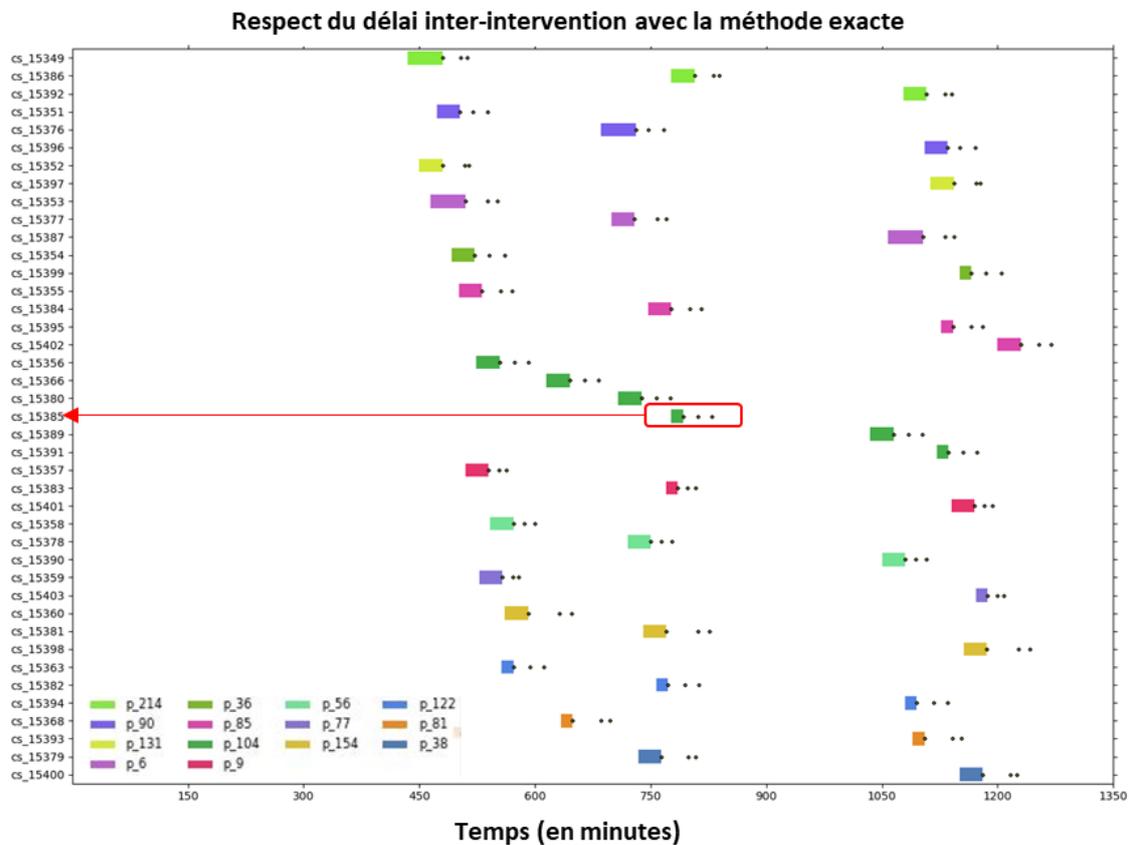


Figure 56 : Diagramme de Gantt sur le respect du délai inter-intervention avec la méthode exacte

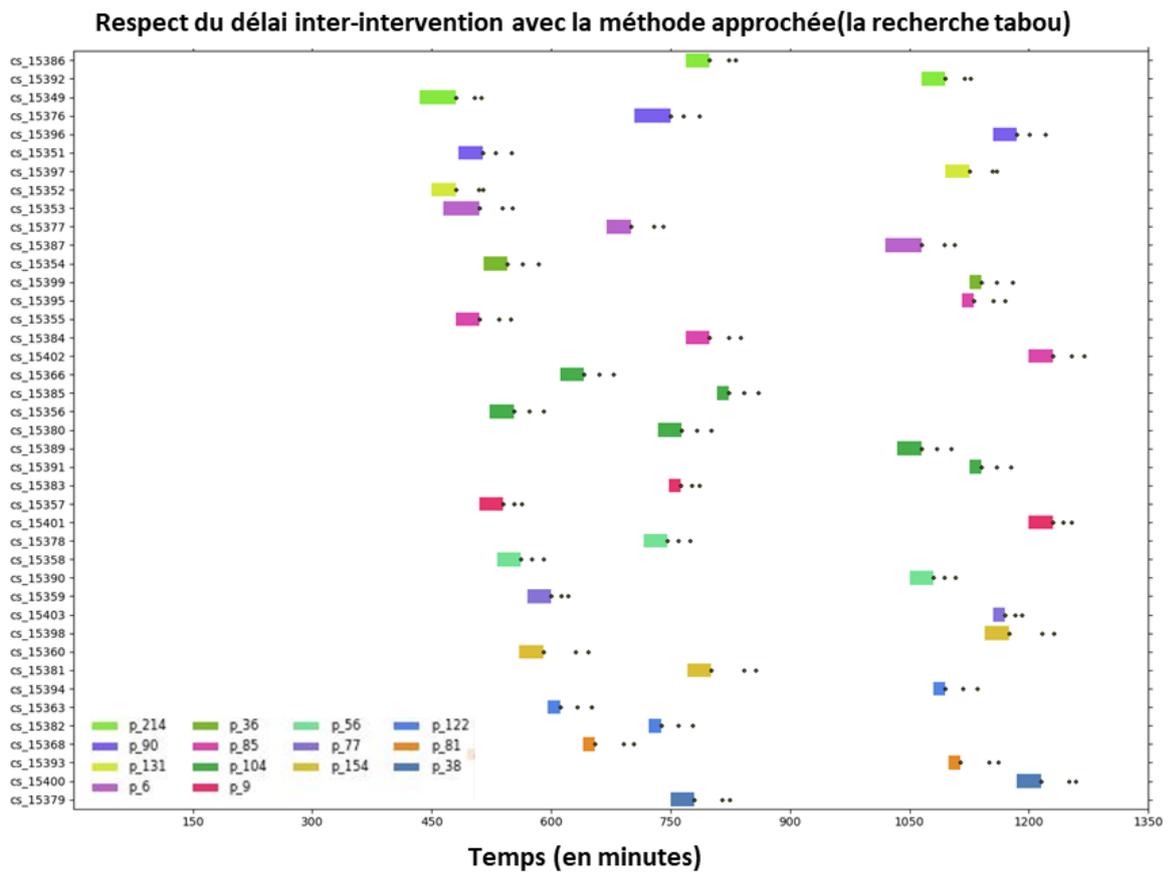


Figure 57 : Diagramme de Gantt sur le respect du délai inter-intervention avec la méthode approchée

Nous pouvons voir sur la Figure 56 et la Figure 57 que toutes les interventions pour des bénéficiaires à demandes multiples respectent toutes les délais inter-interventions demandées, à l'exception de « l'intervention 15385 » planifiée par la méthode exacte (l'intervention en cercle rouge dans la Figure 56) qui démarre entre les points 2 et 3 relatifs à sa demande précédente, mais en restant dans la tolérance de ce bénéficiaire.

5.3.3. Respect de la capacité disponible

Sur la Figure 58, nous pouvons constater que, pour chaque méthode, la charge de travail de tout intervenant est inférieure à sa capacité maximale (paramètre ξ_k). Afin de permettre aux coordinateurs de la PAD de mieux comprendre la répartition du temps de travail de chaque intervenant, nous avons divisé ce temps de travail quotidien en deux parties : le temps consacré aux traitements des interventions, et le temps dédié aux déplacements (au total, 484 minutes à gauche, et 505 minutes à droite, alors que le temps opératoire est le même des deux côtés, à savoir 1515 minutes).

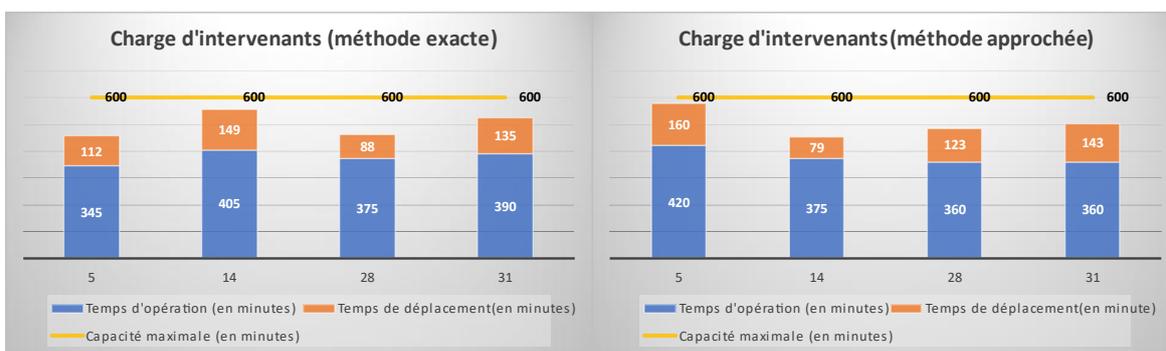


Figure 58 : Diagrammes en vue de la qualification de deux solutions sur le critère du respect de capacité disponible

5.3.4. Equilibrage de pénibilité des tournées

D'après la Figure 59, la pénibilité totale des interventions traitées par chaque intervenant durant sa tournée quotidienne s'équilibre parfaitement. Notons que la difficulté du traitement d'une intervention est caractérisée par l'indicateur dif_i .

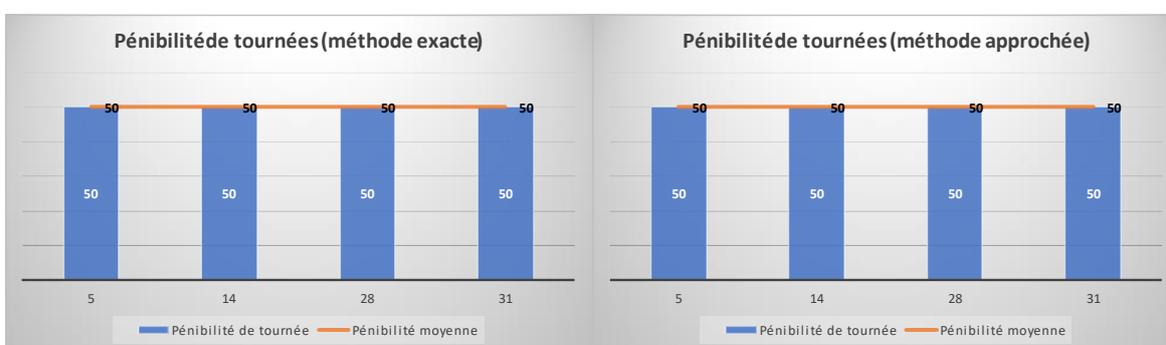


Figure 59 : Diagrammes en vue de la qualification de deux solutions sur le critère d'équilibrage de pénibilité des tournées

6. Conclusion

Après la construction d'une base de connaissances des besoins de coordination de PAD à partir de la définition du DSML-HHCRSP dans le chapitre IV, nous avons traité dans ce chapitre cœur du système de prise de décision en conformité avec les règles de coordination à prendre en compte lors de la prise en charge des bénéficiaires à domicile. Cela a consisté à traduire les expressions du besoin identifiées en un modèle mathématique de prise de décision.

Tout d'abord, nous avons posé les termes de notre étude par la formulation de jeux d'hypothèses dont certaines sont originales puisqu'elles traitent de la satisfaction des parties prenantes.

L'originalité de notre travail réside dans les tolérances de non-satisfaction des 4 dernières contraintes, dites souples. Ces tolérances sont personnalisées en fonction des profils des intervenants et des bénéficiaires. Une fonction de régression linéaire est aussi introduite pour expliquer la relation entre le temps dépassé toléré et non-toléré, relativement à un taux de satisfaction de chaque individu. Cette personnalisation peut être traduite comme une prise en compte de la diversité des profils des acteurs liés aux offres et demandes de services.

Ensuite, deux méthodes sont introduites pour réaliser la construction et la résolution du modèle de prise de décision au niveau PIM et PSM. La première est une méthode exacte basée sur la

Programmation Linéaire en Nombres Mixtes (PLNM). Le modèle s'inscrit donc au niveau PIM en IDM. A l'aide du langage OPL embarqué dans CPLEX, ce PIM est ensuite traduit en un PSM orienté solveur commercial CPLEX, puis résolu par ce même solveur pour obtenir la solution optimale.

La deuxième méthode est une méthode approchée à base d'un CSP (Problème de satisfaction de contraintes). Nous avons d'abord construit directement notre modèle PSM orienté OptaPlanner, puis à l'aide d'un module de Benchmark, nous avons identifié que l'algorithme métaheuristique de type recherche tabou est le plus performant.

Avec ces deux méthodes, nous avons donc enrichi notre environnement de test afin de pouvoir réaliser des comparaisons. Une méthode exacte est supposée nous fournir la meilleure solution, mais présente des limites dès que la taille du problème est importante. Avec une méthode approchée, nous n'avons pas la garantie d'obtenir la solution optimale, mais uniquement une « bonne » solution, et ceci y compris sur des problèmes de très grande taille. Pour examiner la qualité des solutions, nous avons effectué une expérimentation approfondie basée sur 13 cas d'utilisation extraits de cas réels, avec une analyse critique des solutions générées entre méthode exacte et méthode approchée. Nous avons réussi à trouver la limite de la méthode exacte au niveau de temps d'exécution. L'écart entre la solution exacte et la solution approchée a pu alimenter des analyses sur la qualité des solutions générées, sous des configurations différentes.

A l'issue de l'expérimentation pour les 13 cas d'utilisation, nous nous sommes concentrés sur un cas d'utilisation intitulé « ML_s55_p28_mp14_k4 », afin de mieux visualiser et examiner la qualité des solutions générées par les deux méthodes pour certains critères, notamment pour la satisfaction des 4 contraintes souples impliquées dans notre modèle. Après cette étape, nous avons acquis de la confiance sur le fait que la qualité du résultat de la méthode approchée est très voisine de la solution générée par la méthode exacte. Côté coordinateur de la structure de PAD, les solutions sont publiées visuellement et agrémentées d'indicateurs permettant de juger de leur qualité.

La prochaine phase est une phase de mise en synergie de ces résultats des chapitres IV et V. L'imagination d'une fabrique de la coordination, menée en IDM et rythmée par le triplet CIM→PIM→PSM est à l'origine de la construction d'un prototype permettant d'affronter de nombreuses situations différentes. Le focus est mis sur l'enchaînement entre les 3 modules et sur une preuve de concept.

CHAPITRE VI. GENERATION D'UN OUTIL DE COORDINATION OPERATIONNELLE PAR IDM

CHAPITRE VI. GENERATION D'UN OUTIL DE COORDINATION OPERATIONNELLE PAR IDM	145
1. INTRODUCTION	146
2. LA TRANSFORMATION DE MODELE A MODELE	148
2.1. Quelques expériences passées	148
2.2. Le langage de transformation des modèles	149
2.3. Conception générale du prototype	151
3. CONCEPTION DETAILLEE DE LA FABRIQUE DE LA COORDINATION	152
3.1. Transformation PIM-PSM.....	153
3.1.1. Spécification du MM-G	155
3.1.2. Spécification du Méta-Modèle orienté Solution (MM-S) au PSM.....	158
3.1.2.1. Vue « déclaration des données »	158
3.1.2.2. Vue « construction des formulations »	158
3.1.3. Processus de transformation.....	161
3.1.3.1. Règles de mapping entre MM-G et MM-S.....	161
3.1.3.2. Règles de navigation dans le MM-S	167
3.1.3.3. Règle de parsing du M-G et du M-S.....	168
3.2. Transformation CIM-PIM.....	170
3.2.1. Processus de transformation.....	171
3.2.1.1. Règles de mapping entre MM-C et MM-G	171
3.2.1.2. Règles de navigation dans le MM-G.....	172
3.3. Résolution du modèle par CPLEX.....	173
3.3.1. Cas d'étude : changement du problème.....	174
3.3.1.1. Cas d'utilisation 1	174
3.3.1.2. Cas d'utilisation 2	175
4. DEVELOPPEMENT D'UN OUTIL DE TRANSFORMATION EN ADOXX : BLAUTOM	177
5. CONCLUSION.....	180

1. Introduction

Ce chapitre est consacré à une preuve de concept en IDM pour en valoriser la pratique sur le sujet de la coordination en PAD. Nous finirons ainsi de répondre à nos interrogations en abordant la dernière question **Q3 : Comment définir une série de règles de transformation de modèles à partir d'un langage spécifique au domaine (DSL) permettant de supporter un processus dématérialisé allant des connaissances orientées métiers vers les connaissances mathématiques requises pour traiter les problèmes de coordination, et les formuler de manière systématique ?** Mais nous irons aussi plus loin en nous nourrissant des résultats des chapitres IV et V pour aller jusqu'aux calculs de planning pour produire une preuve concrète.

Ainsi, le chapitre V a montré que pour résoudre un problème de prise de décisions, un raisonnement de recherche opérationnelle passe par une formulation mathématique en optimisation sous contraintes. Il s'agit de poser les termes du problème à soumettre à des algorithmes de recherche de solutions, algorithmes qui sont adaptés à cette formulation. Sur le cas du HHCRSP, cette écriture repose sur l'exploitation des connaissances compilées au chapitre IV dans un modèle de besoin.

Dit autrement, ce raisonnement permet donc (1) de passer de la perception du besoin par un ensemble de multigraphes conformes au DSML qui ont une sémantique métier à (2) des graphes orientés qui ont une sémantique à vocation plus mathématique, et ensuite (3) à une implémentation dans des fichiers requis pour appeler un optimiseur.

Prenons un exemple simple, celui d'une tournée, un point symbolisera un lieu à rejoindre, et un arc représentera le trajet entre deux villes. Une variable binaire est alors associée à un arc, elle possède deux indices qui désigne le sens du trajet (le lieu du départ et le lieu d'arrivée). Ces indices sont gérés par un ensemble ordonné de tous les lieux. La solution se traduira par des variables nulles ($=0$) pour les arcs non retenus au final, les autres forment la tournée choisie et auront une valeur strictement positive ($=1$). Sur cette base, la formulation de la fonction traduisant l'objectif qualifie les performances attendues, et les relations entre les variables traduisent les règles de fonctionnement. Nous retrouvons là les ingrédients de notre rencontre entre offre et demande, d'une part, et un besoin de projeter des contenus du modèle de besoin vers des éléments de la formulation mathématique pour aider à la prise de décision, d'autre part. On peut les résumer en :

- Une sélection des variables à calculer qui formeront ensemble le planning de tournées,
- Une transition des orientations du modèle de besoin vers l'écriture de la fonction à optimiser,
- Une transition des exigences du modèle de besoin vers une écriture de contraintes
- Un assemblage des résultats des trois alinéas ci-dessus pour écrire un code soumis à un solveur avec tout à la fois les données, les variables, la fonction à optimiser et les contraintes.

Ce faisant, et ayant posé une première hypothèse sur la formulation mathématique du problème à traiter, nous devons intégrer les conséquences de celle-ci sur la recherche de solutions. Nous supposons de plus qu'un solveur en programmation linéaire en nombres mixtes est disponible au sein de l'environnement de travail. L'appel de ce solveur se fait par un langage qui lui est

spécifique et qui permet de lui transmettre les données pour faire ses calculs. L'espace des solutions du problème mathématique est parcouru par ce solveur qui cherche la solution faisable avec un objectif poursuivi optimisé. L'appel du solveur est donc aussi associé à un modèle que nous appelons modèle logique de calcul des solutions.

Nous décrivons donc le mécanisme de l'aide à la décision au moyen d'une chaîne de connaissances qui évolue depuis un stade préliminaire d'analyse des besoins métier jusqu'à une proposition de décisions satisfaisantes au regard de ces besoins. Trois formes de modèles jalonnent donc séquentiellement cette chaîne. Cette logique d'enchaînement correspond à un processus cognitif que différents experts contribuent à faire progresser dans le contexte actuel. Nous le résumons sur la Figure 60. Il est composé de quatre étapes : définir le contexte, exprimer les besoins de prise de décision (expert métier), identifier et formuler le problème d'aide à la décision, préparer le calcul de la solution (expert en RO). A l'issue de chaque étape, le raisonnement a produit un modèle qui constitue l'entrée de l'étape suivante.

C'est la façon dont ce processus linéaire s'exécute durant tout l'acte de prise de décision.

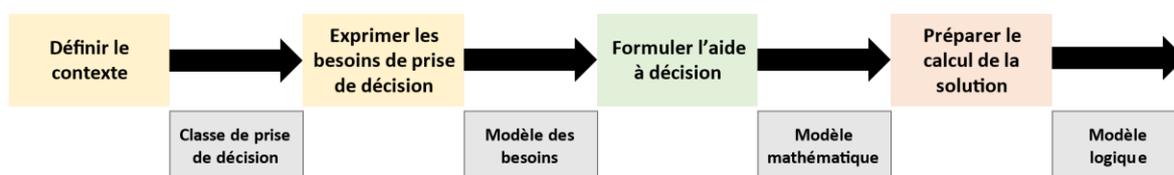


Figure 60 : Processus cognitif d'une prise de décision

Notre choix de développement d'un outil de génération de planning est calqué sur ce processus cognitif. En cela, il cherche à l'explicitier et à le soutenir. L'idée n'est ni de tout automatiser, ni d'ubériser les métiers de la recherche opérationnelle. Rappelons que nous cherchons à générer de multiples formes de problème de coordination, tout en gardant la capacité à calculer leurs plannings.

Pour faire référence à une vue d'ensemble, nous le décrivons comme une **fabrique de la coordination**. Cette fabrique a des ateliers de transformation de connaissances en analogie avec une chaîne de production industrielle. Une chaîne de production est « l'ensemble des moyens qualifiés pour opérer une fabrication de produits manufacturés dans un volume et un temps maîtrisé. Elle s'étend des stocks de matières premières jusqu'à ceux de produits finis ». En plaquant cette définition sur notre sujet, nous identifions des stades d'avancement de la fabrication qui va aller de la définition du système (et de ses besoins en prise de décision) à la production des plannings de bénéficiaires et d'intervenants obtenue par un optimiseur, appelé en toute connaissance de cause. Nous faisons reposer l'architecture fonctionnelle de notre fabrique de coordination sur une approche modulaire. Une série de modules assimilés à des ateliers de transformation réalisent les opérations nécessaires sur la connaissance. Nous avons déjà utilisé cette analogie au chapitre IV. Ce sera notre façon de vous présenter la manière d'implanter le processus d'IDM dans un environnement de méta-modélisation. Chaque module correspond à une étape du processus cognitif de la Figure 60 (avec des conventions de couleur : jaune, vert et rose). Sauf qu'ici, ce n'est pas un bien matériel qui est concerné, mais quelque chose de plus symbolique : une connaissance contenue dans un modèle. La transformation de la connaissance sera effectivement assimilée à une transformation de modèles. Chaque atelier a

son propre objectif de transformation. Les flux entre ateliers sont finalement des flux de modèles.

Nous montrerons la place laissée à l'architecture dirigée par les modèles dans la chaîne des transformations. En amont et en aval de chaque atelier, nous pourrions faire référence à des modèles de type CIM, PIM et PSM. Ainsi, les modèles y sont considérés comme des artefacts techniques peuplant le cycle de vie de l'ingénierie logicielle symbolisé par le schéma en Y introduit au chapitre III (cf. la Figure 14). Ce sont ces modèles qui prennent un caractère opérationnel quand ils permettent in fine d'élaborer le code informatique d'appel à l'optimiseur. Pour ce faire, ce sont des transformations de modèle à modèle (M2M) et de modèle à code (M2C) qui doivent contrôler les pontages entre les connaissances lors du passage dans les ateliers, de CIM à PIM, puis de PIM à PSM.

Dans ce chapitre, nous commencerons par évoquer des travaux qui sont dans la même veine en ce qui concerne l'IDM, il s'agit de modèles de systèmes de production de services ou de biens, de niveau CIM ou PIM. Toutefois, n'ayant pas trouvé de référence à des travaux s'approchant d'une chaîne de traitement comme la nôtre, c'est-à-dire visant à réaliser des calculs avec un optimiseur, nous justifierons le recours à une preuve de concept. Dans un but avant tout pédagogique, mais avec aussi la préservation de l'information vis-à-vis des concurrents de l'entreprise qui finance ces travaux, nous développons cette preuve de concept sur le problème du voyageur de commerce (TSP). Nous avons déjà évoqué la parenté entre TSP et HHCRSP à la section 2.3.1.1 du chapitre II et la section 3 du chapitre IV (cf. la Figure 19). Cette position ne nous écarte donc pas de notre cible. Le développement du prototype impose le choix d'un langage de transformation de modèles (MTL). Celui-ci sera présenté succinctement dans la section 2 de ce chapitre. Dans la section 3 de ce chapitre, nous exposons l'approche globale avec une spécification des trois modèles et une explication de deux transformations appliquées, nous y montrons également comment ce prototype permet de rendre assez systématique le changement de besoin en proposant un nouveau passage dans la fabrique de la coordination avec une nouvelle spécification métier, allant ainsi d'un problème de TSP à un autre problème de TSP. Enfin, nous résumerons dans la section 4 comment l'IDM a été mise en application sur ce TSP et dans l'environnement ADOxx. « BL.Autom » est le nom choisi pour désigner notre fabrique de la coordination

2. La transformation de modèle à modèle

2.1. Quelques expériences passées

Ces 13 dernières années, les méthodes de transformation de modèles ont fait l'objet de plusieurs études dans notre communauté scientifique. Certains travaux font référence à l'architecture dirigée par les modèles (MDA), d'autres pas.

En santé, un prototype basé sur l'IDM est en charge de générer une interface homme-machine sur des technologies mobiles (modèle cible), à partir de la modélisation d'un plan de soins (modèle source) (Khambati et al., 2008). (Cuadrado et al., 2014) transforment un modèle source qui est un modèle de traitement dématérialisé de l'information (un *workflow* spécifié avec plusieurs options de langages possibles côté source (diagrammes d'activité UML, YAWL et BPMN) en un Réseau de Petri, côté cible. (Rhazali et al., 2016) utilisent la notation BPMN pour

modéliser les processus métier d'un niveau du CIM à une cible PIM qui comprend des vues statique, dynamique et fonctionnelle, représentées par 3 types de diagrammes UML adaptés à cela. (Gómez et al., 2018) expliquent comment appliquer du M2M pour transformer un modèle CIM intitulé « *International Electrotechnical Commission common information model* » à un modèle cible de « *Modelica* ». Ce modèle cible permet d'effectuer une simulation dynamique d'un système physique complexe. (Kang et al., 2018) présentent aussi un modèle de simulation d'un système de systèmes (SoS), composé d'un modèle de combat et d'un modèle de réseau. La transformation agit sur le modèle SoS pour produire un modèle de simulation intégré.

En termes d'application dans l'industrie, les problèmes d'interopérabilité entre partenaires impliqués dans différentes configurations des réseaux de collaboration ont été largement traités. Citons les travaux de (Lemrabet and Bourey et al., 2010) sur l'approche MDI (*Model Driven Interoperability*), dans le cadre d'une architecture orientée services. Ces travaux ont trouvé une déclinaison en PLM (Merlo et al., 2014). Nous citons également les travaux de notre équipe sur l'ingénierie de système d'information de médiation MISE (Mu et al., 2017). Le contexte est une collaboration d'entreprises, il s'agit de constituer une base permettant de rassembler des connaissances sur cette collaboration au niveau CIM, au moyen d'un modèle organisationnel. Le niveau PIM représente les connaissances techniques issues de cette base pour en extraire un modèle de processus collaboratif en langage BPMN. Enfin le PSM effectue une extraction de la représentation cartographique de ces processus collaboratifs en BPMN pour orchestrer des services numériques dans un bus de services d'entreprise (Wang et al., 2017).

Ces deux derniers exemples sont proches de notre problématique. Notre cible n'est pas de concevoir un système d'information collaboratif, mais dans un domaine d'application tel que le système de santé, de réussir à piloter un système distribué. L'emploi de l'architecture MDA est largement inspiré de celles du MDI et de MISE.

Mais, à notre connaissance, nous n'avons pas trouvé de références traitant d'application de l'IDM pour la transformation de modèles métier vers un modèle mathématique et alimentant une optimisation combinatoire pour résoudre des problèmes de la coordination au niveau opérationnel, mêlant affectation des ressources et planification de tournées.

2.2. Le langage de transformation des modèles

Nous parlons de fabrique de la coordination et d'ateliers de transformation, en prolongeant cette analogie, l'outil utilisé pour opérer ces transformations dans chaque atelier est le langage de transformation des modèles (MTL) (Bézivin et al., 2006). Un MTL est un langage qui joue un rôle très spécifique dans l'écosystème de l'IDM. Dans son usage, il établit un pont, soit entre modèles à différents niveaux d'abstraction (M2M), soit entre modèle et artefacts secondaires tels que le code et la documentation (M2C) (Sendall and Kozaczynski, 2003).

Faisons un rapide bilan de MTL connus. L'OMG a proposé QVT (*Query / View / Transformation*) comme norme et ligne directrice des futurs langages pour la transformation des modèles (Kurtev, 2007). VMTL (*Visual Model Transformation Language*) (Acretoiaie et al., 2018) est un MTL qui supporte les transformations endogènes. L'implémentation de VMTL se fait sur EMF (*Eclipse Modeling Framework*) (Steinberg et al., 2008) avec le moteur de transformation *Henshin* (Arendt et al., 2010). (Taentzer et al., 2005) préconisent d'effectuer la transformation d'un modèle d'un graphe vers un autre graphe. Ces auteurs pointent quatre outils supportant ce type

de transformation : AGG (Taentzer, 2003), AToM (De Lara and Vangheluwe, 2002), VIATRA2 (Varró and Pataricza, 2004) and VMFS (Levendovszky et al., 2005). Un autre MTL : le langage GReAT (*Graph Rewriting And Transformation*) est introduit par (Agrawal et al., 2006), il permet des transformations d'un domaine à un autre en utilisant des métamodèles hétérogènes. Il est soutenu par GME (*Generic Modeling Environment*) (Davis, 2003) qui permet du prototypage rapide et la réalisation d'outils de transformation.

ATL (*Atlas Transformation Language*), largement cité dans la littérature, est un MTL de type QVT (Jouault et al., 2006) développé et maintenu par OBEO¹ et AtlanMod². ATL fournit les constructions déclaratives et impératives, il est perçu comme un langage hybride de transformation des modèles supportant à la fois les transformations M2M et M2C. Concernant son environnement d'implémentation, les programmes de transformation sont principalement écrits avec ATL, et une machine virtuelle est en charge de la compilation des programmes ATL (Wang et al., 2017), puis le code est exécuté dans EMF (Steinberg et al., 2008).

AdoScript (*ADOxx specific Scripting language*) est un langage de scripts possédant les caractéristiques d'un MTL, qui est intégré dans la plateforme de méta modélisation ADOxx développé par l'OMILAB. Ce langage permet d'accéder et de traiter les informations stockées dans les modèles, puis de mobiliser des fonctionnalités multiples, comme par exemple une transformation M2M ou M2C, la visualisation et la simulation. Concernant la fonctionnalité de transformation M2M, AdoScript peut mettre en œuvre des algorithmes et des mécanismes de transformation à l'aide de techniques d'analyse et de parcours de modèles. En outre, il dispose d'une interface native pour créer des fichiers au format XML, et ainsi, peut être relié à des programmes externes pour enrichir le champ des possibles (Fill and Karagiannis, 2013).

Le Tableau 20 illustre la comparaison de 5 MTLs parmi ceux présentés ci-dessus. Pour en savoir plus, l'étude de (Kahani et al., 2019) établit un catalogue complet comprenant plus de 60 outils de transformation basés sur des MTLs.

Tableau 20 : Comparaison entre cinq langages de transformation (MTL)

Nom de MTL	Référence	Type de M2M		Approche de transformation	Plateforme d'implémentation		
		Endogène	Exogènes		Oui	Non	Si oui, laquelle ?
QVT	(Kurtev, 2007)	X	X	Hybride	X		EMF
VMTL	(Acrejoaie et al., 2018)	X		Manipulation directe	X		EMF
GReAT	(Agrawal et al., 2006)	X	X	Basée sur les graphes	X		GME
ATL	(Jouault et al., 2008)	X	X	Hybride	X		EMF
AdoScript	(Fill and Karagiannis, 2013)	X	X	Hybride	X		ADOxx

Nous avons retenu AdoScript dans l'environnement ADOxx, d'abord parce que c'est l'environnement dans lequel nous avons implanté notre DSML. Mais cet environnement est

¹ Le groupe de modélisation des logiciels de Nantes, un groupe conjoint de l'Institut National de Recherche en Informatique et en Automatique (Inria).

² Editeur logiciel : <https://www.obeo.fr/fr/>

« une plate-forme de méta modélisation » (Karagiannis and Kühn, 2002), qui assure un large ensemble de fonctions permettant la conception de langages de modélisation, et l'emploi de moyens de transformation. Il permet de répondre à une large palette d'objectifs, qu'ils s'inscrivent dans un contexte générique, ou uniquement à l'intérieur d'un domaine spécifique (Thabet, 2020). De plus, cette plateforme est largement utilisée pour construire des preuves de concept (PoC) en recherche. Elle est entourée d'une communauté OMILAB (Fill and Karagiannis, 2013) qui partage des expériences et met en commun les applications développées dans une optique de réutilisabilité. Là encore, ayant rejoint cette communauté, notre équipe de recherche a pu bénéficier de soutien de la communauté, et a pu dresser le constat d'une absence d'application ayant un voisinage avec notre travail dans ce cercle averti.

2.3. Conception générale du prototype

Retraçons la Figure 60 dans l'optique de pouvoir bâtir notre fabrique de la coordination. Dans un premier temps, nous situons nos modèles dans le cadre de l'architecture dirigée par les modèles :

- Le module CIM véhicule les connaissances métier pour une spécification de haut niveau d'une prise de décision. C'est le sens que nous donnons à notre **modèle de besoin** de coordination centralisée en PAD.
- Le module PIM est une étape intermédiaire allant vers l'exécution du besoin décrit par le CIM et mobilisant maintenant des connaissances de recherche opérationnelle. Mais à ce stade, aucun environnement de développement n'a été choisi. Toutes les options restent ouvertes. C'est le sens donné à notre **modèle mathématique** pour l'aide à la décision qui ne fait que de la formulation de problèmes.
- Le niveau PSM dicte l'implémentation du PIM dans un environnement d'exécution connu. Il s'agit d'instancier les concepts du PIM dans un langage spécifique adapté à un moteur de résolution pour produire une solution de prise de décision. C'est le sens donné à notre **modèle logique** visant à faire opérer les calculs par l'optimiseur. Par exemple, en faisant le choix du solveur CPLEX comme moteur de résolution, nousinstancions le modèle mathématique de programmation linéaire en langage OPL. Le PSM correspond aux deux fichiers du modèle logique écrits en OPL dont CPLEX a besoin pour lancer les calculs (un fichier de données en .dat et un fichier de problème en .mod).

A chaque étape dans la logique de fonctionnement de notre prototype, c'est-à-dire dans chaque atelier de BL.Autom, notre fabrique de la coordination, une transformation de modèles va donc être opérée (passage CIM-PIM, puis PIM-PSM). Pour ce faire, le MTL applique des règles de transformation en s'appuyant sur les méta-modèles de la source et de la cible. Il nous faut d'abord définir les méta-modèles pour nos trois modèles (c'est chose faite pour le niveau métier au chapitre IV). Et, à partir de cette base, imaginer qu'une série de règles de mapping entre les méta-modèles ouvre la voie pour générer le modèle cible à partir du modèle source (cf. la Figure 61).

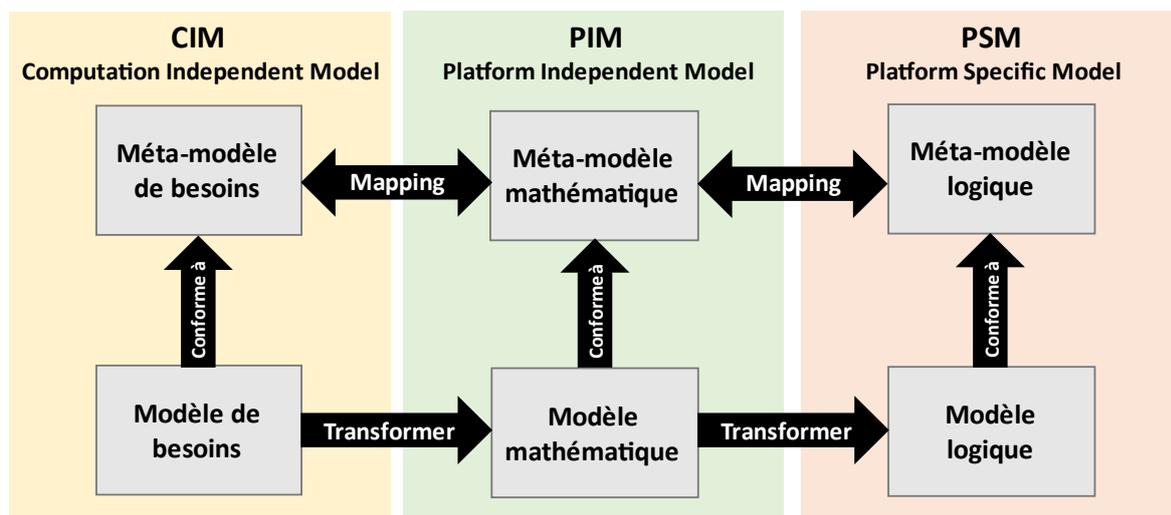


Figure 61 : Principe de transformations des modèles du niveau CIM à celui du PSM

Dans un discours énoncé différemment, nous dirons que la première transformation, du modèle des besoins au modèle mathématique, aura pour effet de faire muter les orientations et les exigences particulières d'une organisation en objectif d'optimisation et en relations mathématiques décrivant des contraintes métier que la recherche d'optimum doit respecter.

La deuxième transformation modifie le problème mathématique de coordination en un formalisme adapté à la méthode de résolution. C'est le modèle logique qui est le modèle cible, à savoir un modèle logique dans un langage adapté à l'optimiseur (OPL, par exemple). Ce modèle sera une entrée du solveur d'optimisation (CPLEX pour OPL, par exemple) qui produira en sortie une solution de coordination optimisée et déterministe, s'il a convergé.

3. Conception détaillée de la fabrique de la coordination

Nous avons donc choisi de monter en compétences sur ce travail de développement de notre outil d'IDM en construisant le prototype sur un problème simple et accessible, celui du TSP avec un énoncé complet dans un but d'illustration permanente. Dans notre travail, nous le redécouvrons sous l'angle de l'IDM. Les éléments essentiels de notre fabrique de la coordination, telle que nous la concevons par transformation successive de modèles, vont être mis en pratique avec un souci d'être didactique, autant que possible. Notre but est de faire la découverte de la capacité de traitement automatisé que nous pouvons atteindre selon ce principe. Nous allons donc rendre « vivants » les différents modèles.

Pour mieux décrire ce fonctionnement du prototype, nous resterons d'abord au niveau de son architecture fonctionnelle, et nous suivrons l'ordre inverse des flux de modèles. Nous partirons d'un modèle logique, d'un code en OPL.

Nous avons donné un nom simplifié à chaque modèle dans une déclinaison plus technique :

- Le modèle conceptuel orienté métier s'appelle **M-C**
- Le modèle mathématique qui est un graphe s'appelle **M-G**
- Le modèle logique qui est le code OPL s'appelle **M-S**

Pour chacune des transformations, l'atelier sera considéré en revanche dans le sens du flux, celui de la transformation, c'est-à-dire M-G vers M-S, puis M-C vers M-G. La logique de chaque

paragraphe de notre plan est la même : nous présenterons d'abord les modèles cible et source, puis nous ferons référence à leurs méta-modèles respectifs. Ayant ainsi planté le décor, nous expliciterons la transformation par ses règles. La Figure 62 évoque trois types de règle : (1) les règles de mise en relation (*mapping*) spécifient la naissance d'un (ou plusieurs) élément(s) du modèle cible à partir d'un (ou plusieurs) éléments du modèle source ; (2) les règles de navigation (ou butinage en référence au terme *browsing*) consistent à traiter localement les éléments du modèle cible qui sont issus d'une transformation pour assurer de la cohérence entre eux et garantir la conformité au méta-modèle référent ; (3) les règles d'analyse de données structurées par annotation (*parsing*) sont des traitements locaux qui facilitent la production d'un code informatique obéissant aux règles syntaxiques d'un langage (OPL, par exemple) pour le qualifier d'exécutable.

L'ensemble de la chaîne de transformation est résumé sur la Figure 62.

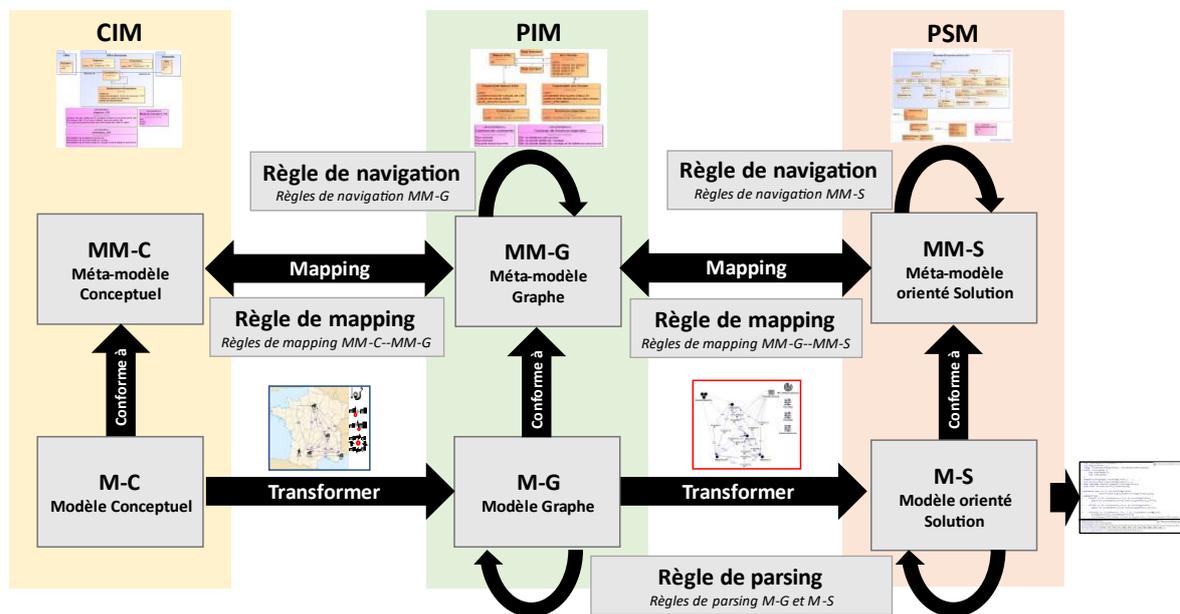


Figure 62 : Chaîne de transformations des modèles du niveau CIM à celui du PSM - application au TSP

3.1. Transformation PIM-PSM

La transformation de PIM à PSM concerne la génération du modèle M-S à partir du modèle M-G. La plateforme spécifique est CPLEX, le modèle M-S est écrit dans un code en *Optimization Programming Language* (OPL). Il a une syntaxe et une sémantique proches des canons de la formulation mathématique pratiquée par la communauté de l'optimisation. OPL fournit des primitives (mots clés et opérateurs) interprétables par CPLEX. Le problème d'optimisation combinatoire (ici notre TSP) est traduit dans ce code en spécifiant d'abord une partie déclarative de données pertinentes (jeu de paramètres et jeu de variables de décision), puis une partie de relations formées par le jeu des contraintes et la fonction objectif. CPLEX est très puissant et a déployé une panoplie de règles de calcul autour d'une approche par séparation et évaluation progressive pour les variables entières (*branch and bound* (Lawler and Wood, 1966)) et du simplexe pour les variables réelles, en programmation linéaire. Il reconnaît la nature d'un problème qui lui est soumis et traite en autonomie les réglages rendant la résolution efficace. Nous avons éprouvé ces capacités au chapitre V.

La Figure 63 et la Figure 64 montrent respectivement un exemple de M-G et de M-S pour un TSP à quatre villes sur un territoire métropolitain avec une fonction objectif de minimisation des distances parcourues.

Le M-G porte des annotations qui permettent de mémoriser la structure du graphe au niveau des arcs (Arc Route<2,1>, par exemple) et des nœud numérotés (Nœud Ville 1, par exemple). Les concepts concernant les ensembles homogènes sont aussi présentés (« **Ensemble Nœud Ville** », « **Ensemble Arc Route** ») et sont reliés à leurs éléments par des lignes pointillées. Enfin, les objets qui sont positionnés sur la droite correspondent à des relations mathématiques dont l'évaluation mobilise des éléments du graphe. Nous avons la fonction à minimiser (min. la distance parcourue), ainsi que deux contraintes portant sur les entrées et sorties de ville (« Flux entrant », « Flux sortant ») qui serviront à limiter le nombre de visites autorisées, et une contrainte qui restreint la solution à une seule et unique tournée (« Aucune sous tournée »). Le lecteur peut trouver la légende utile à une parfaite compréhension du M-G dans le Tableau 23.

Le M-S est un modèle en OPL qui se compose de deux fichiers. Dans le fichier « *TSP.mod* », une partie déclarative des données manipulées avec leur type (*int*, *range*, *tuple*, *setof*, *int*, lignes 1 à 10), puis ce sont les variables (*dvar*, lignes 11 et 12) qui sont distinguées de manière abstraite. Enfin, il possède une partie d'instructions exécutables composée des lignes de code relatives aux relations explicitant la fonction à optimiser (*minimize*, lignes 14 à 15) et les contraintes sont définies à l'aide de mots clés (*subject to*, lignes 16 à 25). Dans un second fichier « *TSP.dat* », les valeurs des données déclarées peuvent être renseignées avant de lancer un calcul. Il est facile d'y repérer les caractéristiques de notre cas d'études avec les 4 villes (« *nombre_total_de_noeuds_de_ville*») et la spécification des distances entre les villes (« *matrice_de_distance*») est faite par une liste en tuple des arcs (« *ensemble_des_tuples_debut_fin*»). L'indice du premier nœud (« *indice_de_noeud_initial*») est donné, permettant de fournir la ville de début de tournée.

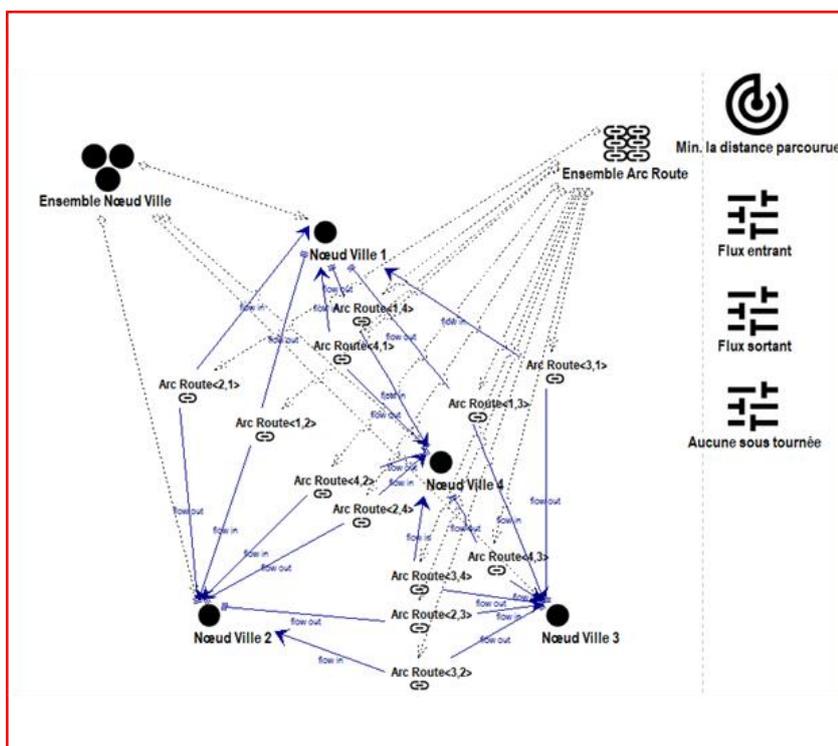


Figure 63 : Modèle Graphe (M-G) sur un cas d'étude

```

1 int nombre_total_de_noeuds_de_ville=...;
2 int indice_de_noeud_initial=...;
3 range Ensemble_Noeud_Ville=
4   indice_de_noeud_initial..nombre_total_de_noeuds_de_ville;
5 ⊕ tuple Arc_Route {
6   int ID_du_noeud_de_depart ;
7   int ID_du_noeud_de_fin ;
8 }
9 setof(Arc_Route) ensemble_des_tuples_debut_fin =...;
10 int matrice_de_distance[ensemble_des_tuples_debut_fin]=...;
11 dvar boolean dvar_affectation[ensemble_des_tuples_debut_fin];
12 dvar int+ dvar_aucune_sous_tournee[Ensemble_Noeud_Ville];
13
14 ⊕ minimize sum (<i,j> in ensemble_des_tuples_debut_fin)
15   matrice_de_distance[<i,j>]*dvar_affectation[<i,j>];
16 ⊕ subject to{
17 ⊕ forall( i in Ensemble_Noeud_Ville, <i,j> in ensemble_des_tuples_debut_fin )
18   sum( j in Ensemble_Noeud_Ville:j!=i ) dvar_affectation[<i,j>]==1;
19 ⊕ forall( j in Ensemble_Noeud_Ville, <i,j> in ensemble_des_tuples_debut_fin )
20   sum( i in Ensemble_Noeud_Ville:i!=j ) dvar_affectation[<i,j>]==1;
21 ⊕ forall(i in Ensemble_Noeud_Ville: i>1, j in Ensemble_Noeud_Ville:j>1&&j!=i)
22   dvar_aucune_sous_tournee[i]-dvar_aucune_sous_tournee[j]+
23   (nombre_total_de_noeuds_de_ville-1)*dvar_affectation[<i,j>]
24   <=nombre_total_de_noeuds_de_ville-2;
25 }

```

 **TSP.mod**

```

1 indice_de_noeud_initial=1;
2 nombre_total_de_noeuds_de_ville=4;
3 ensemble_des_tuples_debut_fin={<1,2><1,4><1,3><2,4><2,1><2,3>
4 <4,2><4,1><4,3><3,1><3,2><3,4>};
5 matrice_de_distance=[550,175,750,115,600,450,135,230,200,800,
6 430,180,];

```

 **TSP.dat**

Figure 64 : Modèle orienté Solution (M-S) transféré du Modèle Graphe (M-G) sur un cas d'étude

Le M-S doit être conforme à un méta-modèle appelé MM-S. De même, le M-G sera conforme au méta-modèle appelé MM-G. Nous allons les définir et les manipuler.

3.1.1. Spécification du MM-G

En recherche opérationnelle, la TSP est considéré comme un problème d'optimisation combinatoire NP-difficile. Cela signifie que le temps de résolution ne peut être borné par une formule polynomiale en fonction de la taille du problème soumis. Ce problème est très majoritairement abordé sous forme d'un graphe simple orienté, de la forme $G = (V, E)$ où V est l'ensemble des nœuds et E l'ensemble des arcs (Bondy and Murty, 1976). Partant de la conceptualisation d'un tel graphe telle que proposée par (Enríquez et al., 2015), nous l'avons étendu pour notre atelier. Les concepts peuplant notre MM-G sont répertoriés sur le Tableau 21.

Tableau 21 : Synthèse des concepts du MM-G

1	Nœud Ville	Le nœud est apparenté à une ville. Un graphe peut contenir zéro ou plusieurs nœuds de ville, et un nœud de ville fait partie d'un graphe. Il est caractérisé par un nom et un identifiant (« <i>ID</i> »).
2	Arc Route	L'arc de routage est apparenté au chemin reliant des villes. Il hérite de la méta-classe « Arc » d'un graphe. Un graphe peut contenir zéro ou plusieurs arcs de routage et un arc de routage fait partie d'un graphe. Il existe deux interprétations d'un même arc : l'arc est entrant dans une ville ou l'arc est sortant d'une autre ville, en raison de la structure intrinsèque du graphe orienté. Un « Arc Route » est caractérisé par son « <i>nom</i> », l'« <i>ID du nœud de départ</i> » (ex. 3), l'« <i>ID du nœud de fin</i> » (ex. 1), par commodité ces deux attributs sont regroupés dans une

		structure de type « <i>tuple début-fin</i> » (ex. <3,1>) et, enfin, la « <i>longueur-arc</i> » est associée à une distance ou à une durée en unité donnée (ex : 8 kms)
3	Ensemble Nœud Ville	L'ensemble des nœuds de ville d'un graphe. C'est un concept clé pour l'optimisation combinatoire, car le jeu des variables de décision sera formellement bâti à partir de ce concept. L'ensemble des nœuds est une classe dont certains attributs deviendront par la suite des paramètres du modèle mathématique (ex : le nombre de villes). L'ensemble des nœuds est caractérisé par son « <i>nom</i> », puis par le « <i>nombre total de nœuds de ville</i> » et l'« <i>indice de nœud initial</i> ».
4	Ensemble Arc Route	L'ensemble des arcs de routage du graphe. Des règles similaires à celles du « Ensemble Nœud Ville » sont appliquées à ce concept, notamment la définition de variables de décision et de paramètres. L'« Ensemble Arc Route » est caractérisé par son « <i>nom</i> », l'« <i>ensemble des tuples début-fin</i> » et la « <i>matrice de distance ou de temps</i> » en unité donnée.
5	Contrainte	Le descripteur de contraintes caractérisé par l'attribut « <i>nom</i> »
6	Fonction objectif	Le descripteur de fonctions objectifs caractérisé par l'attribut « <i>nom</i> »
7	Contenu de contrainte	Liste énumérative d'expressions mathématiques candidates
8	Contenu de fonction objectif	Liste énumérative d'expressions mathématiques candidates

Les contenus évoqués dans les deux dernières lignes du tableau 2 exigent de préciser les jeux de variables de décision qu'il faut leur appairier. Par exemple, une contrainte nommée « Aucune sous tournée » sera liée à une variable de décision binaire nommée « *dvar_ aucune sous tournée* » qui sera nommée dans son expression mathématique en faisant usage d'éléments de l'« Ensemble Nœud Ville ». Le Tableau 22 donne un aperçu de ces mécanismes d'appariement pour l'exemple du TSP.

Tableau 22 : Synthèse d'une association entre contenu de contraintes et variables de décision associée

Contenu de contrainte	Type de variable de décision	Nommage de la variable de décision
Flux entrant	Booléen	dvar_affectation
Flux sortant		
Aucune sous tournée	Entier positif	dvar_ aucune sous tournée

La Figure 65 représente le MM-G sous forme d'un diagramme de classes UML, c'est-à-dire une connaissance du graphe décrite selon 3 aspects :

- Visualisation de la structure de ce graphe.
- Génération des données d'entrée dont l'optimiseur CPLEX aura besoin.
- Appariement des variables de décision aux contraintes et à la fonction objectif.

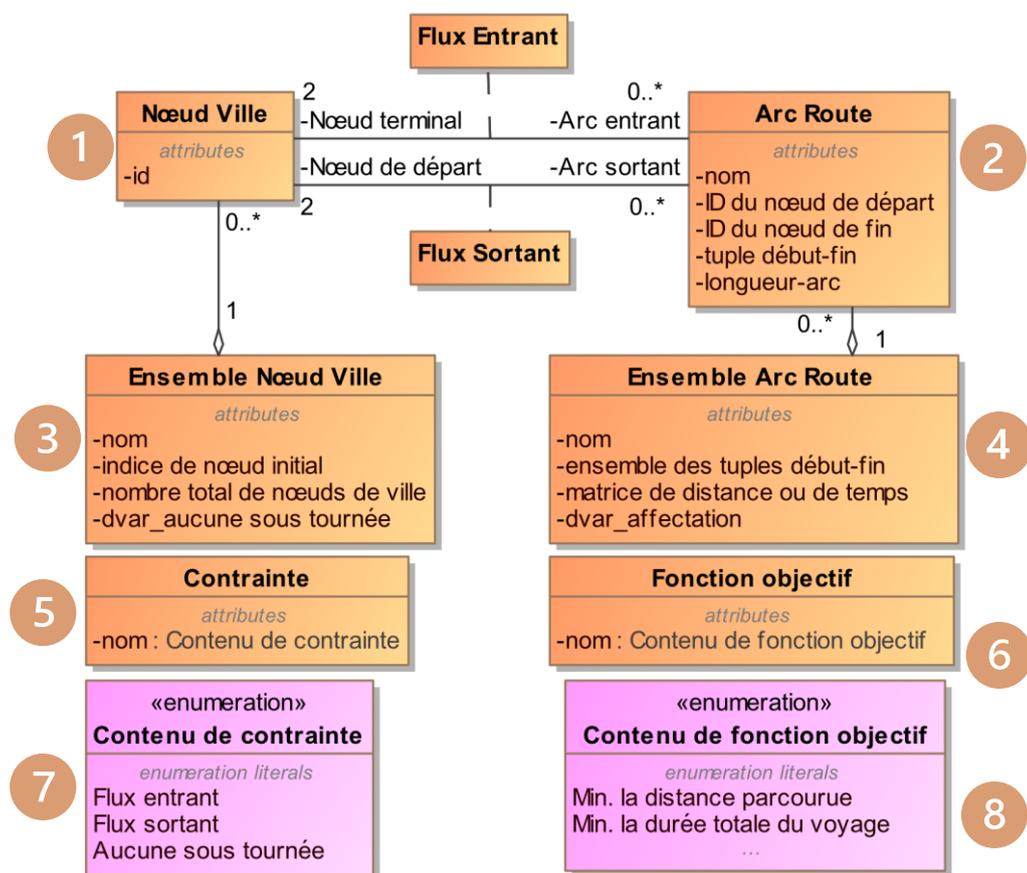


Figure 65 : Méta-Modèle Graphe (MM-G) au niveau PIM

Le Tableau 23 est une représentation graphique (ou syntaxe concrète) choisie pour l'instanciation des entités pour dessiner un modèle (M-G) conforme au MM-G.

Tableau 23 : Représentation graphique des concepts inclus dans le MM-G

Nom de concept du MM-G	Représentation graphique
Nœud Ville	
Arc Route	
Ensemble Nœud Ville	
Ensemble Arc Route	
Contrainte	
Fonction objectif	
Flux Entrant	
Flux Sortant	

Cette syntaxe concrète utilise des icônes qui sont assimilables à ceux que nous avons choisis dans la section 4.1.3 du chapitre IV (cf. le Tableau 6) pour le DSML-TSP, mais nous précisons ici que ce ne sont pas les mêmes et surtout qu'ils n'ont absolument pas la même sémantique, ni la même destination (niveau PIM, et non CIM). Cela sera traité ultérieurement dans ce chapitre.

3.1.2. Spécification du Méta-Modèle orienté Solution (MM-S) au PSM

Le MM-S est inspiré des travaux de (Assouroko and Denno, 2016). En effet, notre recherche bibliographique nous a conduit vers ces chercheurs nord-américains, travaillant au département du commerce du NIST qui ont pris l'initiative, il y a plus de 5 ans, de réaliser un modèle conceptuel du langage OPL. Nous nous sommes inspirés de leurs travaux dans ce qui suit.

La structure de ce méta-modèle est la suivante :

- Une vue « déclaration des données ».
- Une vue « construction des formulations ».

3.1.2.1. Vue « déclaration des données »

Nous énumérons l'ensemble des concepts dans le Tableau 24 qui contient les types de données dans la syntaxe du langage OPL, l'utilité est située dans la traduction des données du M-G.

Tableau 24 : Synthèse des concepts du MM-S (vue « déclaration des données »)

1	Déclaration Données	<p>C'est la liaison établie entre les données et leurs types en OPL. Les attributs sont</p> <ul style="list-style-type: none"> • « <i>nom</i> » : le nom retenu pour référencer une donnée à déclarer. • « <i>type</i> » : le type de données en OPL. • « <i>est_Dvar</i> » : un booléen indiquant si une donnée peut être modifiée en cours d'exécution, c'est-à-dire qu'on désigne une variable de décision, ainsi déclarée dans l'espace de décision. • « <i>indice</i> » : cet attribut est de type « énumération » comprenant trois formes de déclaration d'indices en scalaire, couple ou triplet: « i or j or k », « <i,j> » et « <i,j,k > ».
2	AbstractType	<p>Un descripteur abstrait et générique de type de données (scalaire, vectoriel ou de structure hybride) en OPL dont des sous-classes peuvent être dérivées avec une implémentation par un mot-clé. Héritant de ce concept générique, les autres concepts fils sont extraits de (Assouroko and Denno, 2016).</p>

3.1.2.2. Vue « construction des formulations »

Il s'agit d'un pool de patrons de relations mathématiques dans lesquelles les syntaxes associées aux formulations de contraintes et à la fonction objectif sont embarquées. Notre transformation est donc destinée à identifier le bon patron (*pattern*) de formulation avec les composants qui lui sont associés provenant du M-G. Le Tableau 25 est une synthèse des concepts de la vue « construction des formulations » du MM-S.

Tableau 25 : Synthèse des concepts du MM-S (vue « construction des formulations »)

3	Construction Formulation	<p>Un concept abstrait qui est un conteneur de patrons de relations écrites en OPL. Il est donc caractérisé par :</p> <ul style="list-style-type: none"> • L'identifiant de formulation (<i>nom</i>) : identique à l'identifiant de concept du M-G (les noms des contraintes ou des fonctions objectifs). • Le « <i>contenu du patron</i> » : doit recenser les caractéristiques d'appel de patrons OPL avant l'affectation des données requises par le calcul (paramètres ou variables de décision).
4	Construction Contrainte	<p>Ce concept hérite du concept « Construction Formulation ». Il a un attribut « équation dans CPLEX » visant à afficher la formulation complète d'une contrainte après avoir effectué l'assemblage des composants du patron concerné.</p>
5	Construction FO	<p>Ce concept hérite du concept « Construction Formulation ». Il remplit une fonction similaire à « Construction Contrainte », mais pour la fonction objectif.</p>
6	Composants Formulation	<p>Ce concept est une forme d'accessor en langage orienté objet. Il regroupe les attributs pour renseigner les composants de formulation pour le patron concerné, en suivant les renseignements consignés dans les recommandations d'usage de la bibliothèque qui stocke ces patrons. Il est caractérisé par le nom du patron concerné(<i>nom</i>), et des paramètres et variables de décision avec un effectif adapté, <i>paramètre 1[P1], paramètre 2[P2], paramètre 3[P3], variable de décision 1[D1], variable de décision 2 [D2]</i>.</p>

La Figure 66 est une représentation de ce MM-S sous forme d'un diagramme de classes UML. Notons que les descripteurs de structures de données (le concept « **AbstractType** » avec l'ensemble de ses sous-concepts) sont en anglais reprenant directement les travaux de notre source d'inspiration (Assouroko and Denno, 2016)

3.1.3. Processus de transformation

Le processus de transformation se compose de trois parties.

- (1) La première partie est une transformation relevant d'une transformation « M2M ». Elle est supportée par les règles de mapping entre MM-G et MM-S.
- (2) La seconde partie réalise un butinage (*browsing*) interne au M-S, dans le niveau PSM. Il s'agit d'un traitement automatique visant à être syntaxiquement correct dans les appels des deux « constructeurs » de formulation.
- (3) La troisième partie est une phase d'analyse de données du code OPL, un *parsing*. Il s'agit plutôt d'une transformation « M2C ». Un parseur développé en Java est défini pour réaliser un assemblage avec l'affectation des mots clés en OPL.

A l'issue de l'exécution de ce jeu de règles, nous récoltons le code du modèle final qui est exécutable par CPLEX et peut techniquement lui être soumis pour trouver la solution du TSP.

3.1.3.1. Règles de mapping entre MM-G et MM-S

- Déclaration des données

La traduction des données dans le bon type est centrée sur deux ensembles : l'« **Ensemble Nœud Ville** » et l'« **Ensemble Arc Route** ». Cette traduction s'opère vers la partie déclarative du code OPL avec une distinction faite entre paramètres du modèle et variables de décision. Chacun se voit ensuite attribué son bon mot clé concernant l'indication du type de données requis par OPL. Cette traduction suit 2 étapes d'une transformation « M2C » :

- (1) La déclaration de l'« **Ensemble Nœud Ville** » commence par un choix de type : « **RangeType** », puis des types primitifs entiers sont choisis pour ses deux composants : borne supérieure (« *nombre total de nœuds de ville* ») et borne inférieure (« *indice de nœud initial* »). Enfin, la déclaration de la variable de décision « *dvar_aucune sous tournée* ».

A titre d'un exemple, l'algorithme 1 est un exemple de règles de mapping appliquées pour déclarer « *dvar_aucune sous tournée* » en AdoScript. Notons que l'expression de cette variable en OPL est comme suit :

```
dvar int+ dvar_aucune_sous_tournee[Ensemble_Noed_Ville].
```

Nous devons d'abord indiquer les caractéristiques basiques de cette variable par la création d'un objet de la classe « **Déclaration Données** » dans le **M-S** (la ligne 2 et les lignes 6-19 de l'algorithme 1). Le type de « *dvar_aucune sous tournée* » est un tableau vectoriel (type « **Array** ») à une dimension (la ligne 4 et les lignes 20-30 de l'algorithme 1), commençant par une déclaration de son type primitif (**int+**, la ligne 5 avec les lignes 24-26 et les lignes 31-34 de l'algorithme 1) et se poursuivant par une indication de la taille du tableau (« **Ensemble Nœud Ville** », la ligne 3 avec les lignes 27-29 de l'algorithme 1).

Algorithme 1 : la déclaration du « dvar_aucune sous tournée »

```

1   $\omega$  ← L'attribut « dvar_aucune sous tournée » de l'objet du « Ensemble Nœud Ville » dans le M-G
2   $\mu$ : Pour  $\omega$ , créer l'objet du « Déclaration données » dans le M-S
3   $\vartheta$ : Pour  $\omega$ , créer l'objet du « Ensemble Nœud Ville » qui est l'indice de la taille de  $\omega$  dans le M-S
4   $t$ : Pour  $\omega$ , créer l'objet du « ArrayType » dans le M-S // type du  $\omega$ 
5   $\kappa$ : Pour  $\omega$ , créer l'objet du « PositiveIntegerType » dans le M-S // type primitif du  $\omega$ 
6  Pour    $i$  dans tous les attributs du  $\mu$  // Affectation des attributs du  $\mu$ 
7  |      Si   ( $i = \text{"nom"}$ )
8  |      |      $i \leftarrow \text{"dvar_aucune sous tournée"}$ 
9  |      Fin
10 |     Si   ( $i = \text{"type"}$ )
11 |     |      $i \leftarrow t$ 
12 |     Fin
13 |     Si   ( $i = \text{"est_Dvar"}$ )
14 |     |      $i \leftarrow \text{"True"}$ 
15 |     Fin
16 |     Si   ( $i = \text{"indice"}$ )
17 |     |      $i \leftarrow \text{"i ou j ou k"}$ 
18 |     Fin
19 Fin
20 Pour    $j$  dans tous les attributs du  $t$  // Affectation des attributs du  $t$ 
21 |     Si   ( $j = \text{"rep"}$ )
22 |     |      $j \leftarrow \text{"["}$ 
23 |     Fin
24 |     Si   ( $j = \text{"baseType"}$ ) // type primitif du  $\omega$ 
25 |     |      $j \leftarrow \kappa$ 
26 |     Fin
27 |     Si   ( $j = \text{"subscripts"}$ ) // l'indice de la taille de  $\omega$ 
28 |     |      $j \leftarrow \vartheta$ 
29 |     Fin
30 Fin
31 Pour    $k$  dans tous les attributs du  $\kappa$  // Affectation des attributs du  $\kappa$ 
32 |     Si   ( $k = \text{"rep"}$ )
33 |     |      $k \leftarrow \text{"int+"}$ 
34 |     Fin
35 Fin

```

- (2) Vient ensuite une série de traductions autour du concept de « **Ensemble Arc Route** ». La Figure 67 montre un exemple de mapping pour déclarer l'« *ensemble des tuples début-fin* » et l'« **Arc Route** ». A la lecture de cette figure, la déclaration de l'« *ensemble des tuples début-fin* » est réalisée en deux étapes pour aboutir à son expression OPL (cf. le Tableau 26), formulée dans notre cas comme suit :

`setof(Arc_Route) ensemble_des_tuples_debut_fin.`

Tableau 26 : Etapes à suivre pour la déclaration de l'« ensemble des tuples début-fin »

1	La déclaration de son type comme un ensemble (« SetType »).
2	L'indication de l'attribut « <i>subscript</i> » du « SetType » comme étant un « Arc Route »

Puis la déclaration de l'« **Arc Route** » est faite en trois étapes pour donner en OPL, l'expression suivante:

```

tuple Arc_Route {
int ID_du_noeud_de_depart ;
int ID_du_noeud_de_fin ; }

```

Tableau 27 : Etapes à suivre pour la déclaration de l'« Arc Route »

1	La déclaration de son type « tuple ». C'est le type « Record » en mettant la valeur de l'attribut « isTuple » à "true". Comme le « Record » possède un attribut « Record Fields », nous avons deux champs à déclarer. Celui de « ID du nœud de départ » et celui de « ID du nœud de fin ».
2	La déclaration primitive de premier champ « ID du nœud de départ » en entier (« IntegerType »).
3	La déclaration primitive de deuxième champ « ID du nœud de fin » en entier (« IntegerType »).

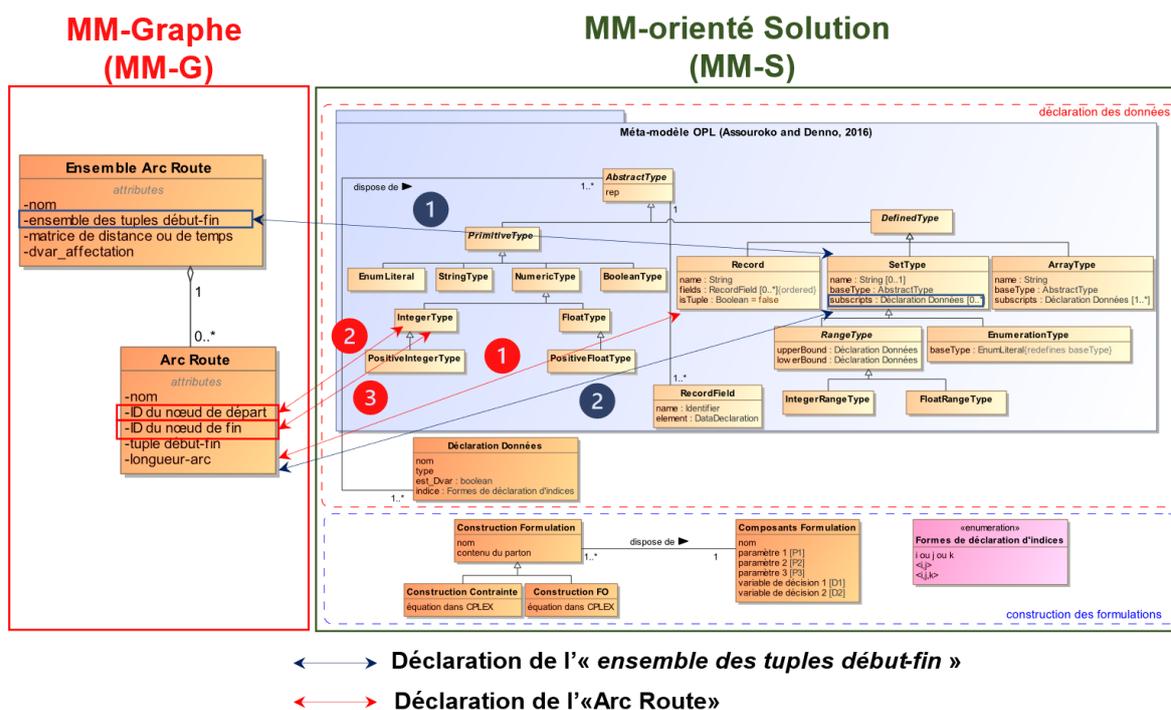


Figure 67 : Exemple de mapping pour l'« Arc Route »

- Construction des patrons et des composants de formulation

Cette phase de transformation utilise des pointeurs pour construire les lignes de codes relatives aux relations mathématiques de la partie des instructions exécutables du code OPL, à partir des patrons de formulation OPL extraits d'une bibliothèque (cf. Figure 68). C'est le premier temps. Cette bibliothèque de relations capitalise des écritures classiques des contraintes et de fonctions traduisant un objectif à moindre effort sans la prise en compte des composants qui consistent à construire le corps d'une équation, tel que les mots clés, les opérateurs et l'ordre dans lequel ils doivent être placés les uns par rapport aux autres. Le pointeur est donc le moyen de désigner la relation qui sera sélectionnée dans la bibliothèque.

Ensuite, dans un second temps, une règle permet de générer des composants en complétant les patrons avec les paramètres attendus par la relation. Cette phase fait appel à un catalogue de spécification des patrons de la bibliothèque (cf. le Tableau 28).

La bibliothèque est indexée pour faire la correspondance avec les concepts « **Contrainte** » et « **Fonction objectif** » définis dans le MM-G. Elle est écrite sous la forme de fonctions (« *Function* ») en AdoScript (cf. les quatre exemples de la Figure 68).

```

1 # Patron de contrainte "Flux entrant"
2 FUNCTION fluxEntrantConstConstruction p1:string p2:string d1:string
3   return: ("forall( i in " + p1 + ", <i,j> in " + p2 + " )
4     sum( j in "+ p1 +":j!=i ) "+ d1 +"[<i,j>]==1;")
5 # Patron de contrainte "Flux sortant"
6 FUNCTION fluxSortantConstConstruction p1: string p2:string d1:string
7   return: ("forall( j in " + p1 + ", <i,j> in "+ p2 + " )
8     sum( i in " + p1 +":i!=j ) "+ d1 +"[<i,j>]==1;")
9 # Patron de contrainte "Aucune sous tournée"
10 FUNCTION aucuneSousTournéeConstConstruction p1:string p2:string d1:string d2:string
11   return: ("forall(i in "+ p1 +": i>1, j in "+ p1 +":j>1 && j!=i )
12     "+ d1 +"[i] - "+ d1 +"[j] + (" + p2 + " - 1) * "+ d2 +"[<i,j>] <= "+ p2 + " - 2;")
13 # Patron de fonction objectif
14 FUNCTION foConstruction p1:string p2:string d1:string
15   return: ("minimize sum ( <i,j> in " + p1 + " ) "
16     + p2 + "[<i,j>] * " + d1 + "[<i,j>];")

```

Figure 68 : Extrait de la bibliothèque de patrons de formulation OPL

Le catalogue aide à identifier une distribution des composants en paramètre ou en variable de décision. Dans le Tableau 28, un symbole « P » désigne un paramètre et un symbole en « D » désigne une variable de décision.

Tableau 28 : Extrait du catalogue des composants de formulation en OPL

Contenu de contraintes et fonctions objectives		Contrainte			FO	
		Flux entrant	Flux sortant	Aucune sous tournée	Min. la distance parcourue	Min. la durée totale du voyage
Concept	Ensemble Nœud Ville	P1	P1	P1		
	<i>indice de nœud initial</i>					
	<i>nombre total de nœuds de ville</i>			P2		
Attributs	<i>dvar_aucune sous tournée</i>			D1		
	Ensemble Arc Route					
	<i>ensemble des tuples début-fin</i>	P2	P2		P1	P1
Attributs	<i>matrice des distances ou des temps</i>				P2	P2
	<i>dvar_affectation</i>	D1	D1	D2	D1	D1

Ces deux éléments de connaissance, bibliothèque et catalogue, sont donc des apports requis pour exécuter les règles de mapping entre MM-G et MM-S avec deux buts :

- D'identifier les concepts « **Construction Formulation** » dans le MM-S associés aux concepts « **Contrainte** » et « **Fonction objectif** » dans le MM-G,
- D'identifier les concepts « **Composants Formulation** » dans le MM-S associés aux concepts « **Contrainte** » et « **Fonction objectif** » dans le MM-G, puis d'affecter les composants de données aux attributs de « **Composants Formulation** » dans le MM-S.

La Figure 69 schématise ce mapping entre **MM-G** et **MM-S**. Ces deux catégories de règles de mapping sont supportées par deux procédures écrites en AdoScript :

- L’algorithme 2 décrit comment faire la construction des patrons. Commençons par la création des objets du « **Construction Formation** » dans le M-S, qui est relatives à ceux de la « **Contrainte** » ou de la « **Fonction objectif** » dans le M-G (les lignes 1-4). Puis à l’aide de la bibliothèque de patron (cf. la Figure 68), nous associons le patron à la description de la formulation (les lignes 5-21).
- L’algorithme 3 montre comment construire des composants. Cette construction commence par la création des objets du « **Composants Formulation** » dans le M-S liés aux contraintes et fonction objectif spécifiées dans le M-G (les ligne 1-4). Puis à l’aide du catalogue des composants (cf. le Tableau 28), nous établissons le lien entre les composants et la description de la formulation (les lignes 5-28).

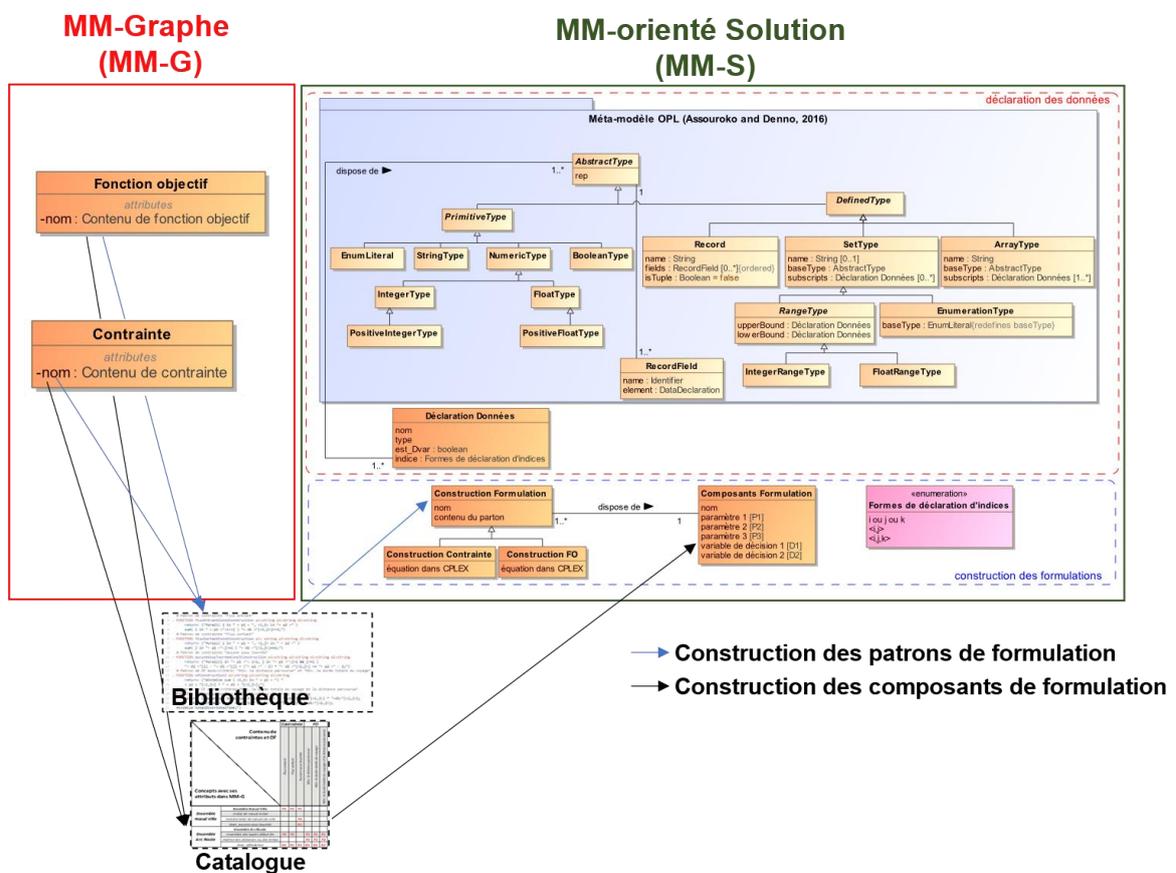


Figure 69 : Construction des patrons et des composants de formulations à l’aide de la bibliothèque et du catalogue

Algorithme 2 : construction des patrons de formulations

```

1  biblio ← La bibliothèque de patrons de formulation OPL
2   $\alpha$  ← L'objet du « Contrainte » dans le M-G
3   $\beta$  ← L'objet du « Fonction objectif » dans le M-G
4   $\alpha^*$ : Pour  $\alpha$ , créer l'objet du « Construction Contrainte » dans le M-S
5   $\beta^*$ : Pour  $\beta$ , créer l'objet du « Construction FO » dans le M-S
6  Si ( $\alpha.nom = \text{"Flux entrant"}$ )
7  |    $\alpha^*.nom \leftarrow \alpha.nom$ 
8  |    $\alpha^*.contenu\ du\ parton \leftarrow biblio.fluxEntrantConstConstruction()$ 
9  Fin
10 Si ( $\alpha.nom = \text{"Flux sortant"}$ )
11 |    $\alpha^*.nom \leftarrow \alpha.nom$ 
12 |    $\alpha^*.contenu\ du\ parton \leftarrow biblio.fluxSortantConstConstruction()$ 
13 Fin
14 Si ( $\alpha.nom = \text{"Aucune sous tournée"}$ )
15 |    $\alpha^*.nom \leftarrow \alpha.nom$ 
16 |    $\alpha^*.contenu\ du\ parton \leftarrow biblio.aucuneSousTourneeConstConstruction()$ 
17 Fin
18 Si ( $\beta.nom = \text{"Min. la distance parcourue"}$  Ou  $\text{"Min. la durée totale du voyage"}$ )
19 |    $\beta^*.nom \leftarrow \beta.nom$ 
20 |    $\beta^*.contenu\ du\ parton \leftarrow biblio.foConstruction()$ 
21 Fin

```

Algorithme 3 : construction des composants de formulations

```

1  catalog ← Le catalogue des composants de formulation en OPL
2   $\alpha$  ← L'objet du « Contrainte » dans le M-G
3   $\beta$  ← L'objet du « Fonction objectif » dans le M-G
4   $\alpha^*$ : Pour  $\alpha$ , créer l'objet du « Composants Formulation » dans le M-S
5   $\beta^*$ : Pour  $\beta$ , créer l'objet du « Composants Formulation » dans le M-S
6  Si ( $\alpha.nom = \text{"Flux entrant"}$  Ou  $\text{"Flux sortant"}$ )
7  |    $\alpha^*.nom \leftarrow \alpha.nom$ 
8  |    $v \leftarrow catalog.prendre\ les\ composants\ de\ \text{"Flux entrant"}$  Ou  $\text{"Flux sortant"}$  {P1, P2, D1}
9  |    $\alpha^*.param\ etre\ 1\ [P1] \leftarrow v.P1(\text{"Ensemble Nœud Ville"})$ 
10 |    $\alpha^*.param\ etre\ 2\ [P2] \leftarrow v.P2(\text{"ensemble des tuples début-fin"})$ 
11 |    $\alpha^*.variable\ de\ décision\ 1\ [D1] \leftarrow v.D1(\text{"dvar_affectation"})$ 
12 Fin
13 Si ( $\alpha.nom = \text{"Aucune sous tournée"}$ )
14 |    $\alpha^*.nom \leftarrow \alpha.nom$ 
15 |    $v \leftarrow catalog.prendre\ les\ composants\ de\ \text{"Aucune sous tournée"}$  {P1, P2, D1, D2}
16 |    $\alpha^*.param\ etre\ 1\ [P1] \leftarrow v.P1(\text{"Ensemble Nœud Ville"})$ 
17 |    $\alpha^*.param\ etre\ 2\ [P2] \leftarrow v.P2(\text{"nombre total de nœuds de ville"})$ 
18 |    $\alpha^*.variable\ de\ décision\ 1\ [D1] \leftarrow v.D1(\text{"dvar_aucune sous tournée"})$ 
19 |    $\alpha^*.variable\ de\ décision\ 2\ [D2] \leftarrow v.D2(\text{"dvar_affectation"})$ 
20 Fin
21 Si ( $\beta.nom = \text{"Min. la distance parcourue"}$  Ou  $\text{"Min. la durée totale du voyage"}$ )
22 |    $\beta^*.nom \leftarrow \beta.nom$ 
23 |    $\beta^*.contenu\ du\ parton \leftarrow biblio.foConstruction()$ 
24 |    $v \leftarrow catalog.prendre\ les\ composants\ de\ deux\ fonctions\ objectifs\ \{P1, P2, D1\}$ 
25 |    $\beta^*.param\ etre\ 1\ [P1] \leftarrow v.P1(\text{"ensemble des tuples début-fin"})$ 
26 |    $\beta^*.param\ etre\ 2\ [P2] \leftarrow v.P2(\text{"matrice des distances ou des temps"})$ 
27 |    $\beta^*.variable\ de\ décision\ 1\ [D1] \leftarrow v.D1(\text{"dvar_affectation"})$ 
28 Fin

```

Suite à l'exécution des processus de transformation de modèles et l'application de ces règles, les données du graphe sont déclarées, les syntaxes de chaque contrainte (« **Contrainte** ») et de chaque fonctions objectif (« **Fonction objectif** ») sont décrites à partir de patrons

judicieusement choisis, et des composants de formulation dont ils ont besoin. La Figure 70 est l'état partiel du code OPL à la suite de l'application des règles de mapping entre MM-G et MM-S.

```

int nombre_total_de_noeuds_de_ville

int indice_de_noeud_initial

range Ensemble_Noeud_Ville
    indice_de_noeud_initial
    nombre_total_de_noeuds_de_ville

tuple Arc_Route
    int ID_du_noeud_de_depart
    int ID_du_noeud_de_fin

setof(Arc_Route) ensemble_des_tuples_debut_fin
int matrice_de_distance[ensemble_des_tuples_debut_fin]
boolean dvar_affectation[ensemble_des_tuples_debut_fin]
int+ dvar_aucune_sous_tournee[Ensemble_Noeud_Ville]

minimize sum (<i,j> in P1)P2[<i,j>]*D1[<i,j>];
P1 : ensemble_des_tuples_debut_fin
P2 : matrice_de_distance
D1 : dvar_affectation

forall( i in P1, <i,j> in P2)sum( j in P1:j!=i ) D1[<i,j>]==1;
P1 : Ensemble_Noeud_Ville
P2 : ensemble_des_tuples_debut_fin
D1 : dvar_affectation

forall( j in P1, <i,j> in P2)sum( i in P1:i!=j ) D1[<i,j>]==1;
P1 : Ensemble_Noeud_Ville
P2 : ensemble_des_tuples_debut_fin
D1 : dvar_affectation

forall(i in P1: i>1, j in P1:j>1&&j!=i) D1[i]- D1[j]+(P2-1)* D2[<i,j>]<= P2-2;
P1 : Ensemble_Noeud_Ville
P2 : nombre_total_de_noeuds_de_ville
D1 : dvar_aucune_sous_tournee
D2 : dvar_affectation
    
```

Figure 70 : Code OPL suite à l'application des règles de mapping entre MM-G et MM-S

3.1.3.2. Règles de navigation dans le MM-S

Ces navigations locales visent à assembler tous les résultats hérités de « **Construction Formulation** » avec ceux de « **Composants Formulation** ». A titre d'exemple, les règles de navigation sur la formulation de contrainte « Flux entrant » sont illustrées dans l'algorithme 4, en suivant deux étapes indiquées dans le Tableau 29 (cf. la Figure 71).

Tableau 29 : Etapes à suivre pour la construction de la formulation de contrainte « Flux Entrant »

1	L'assemblage composants patrons pour construire les formulations (les lignes 1-11 de l'algorithme 3).
2	Les résultats de l'assemblage final sont affectés à l'attribut « <i>équation dans CPLEX</i> » du concept « Construction Contrainte » ou « Construction FO » (les lignes 12 de l'algorithme 3).

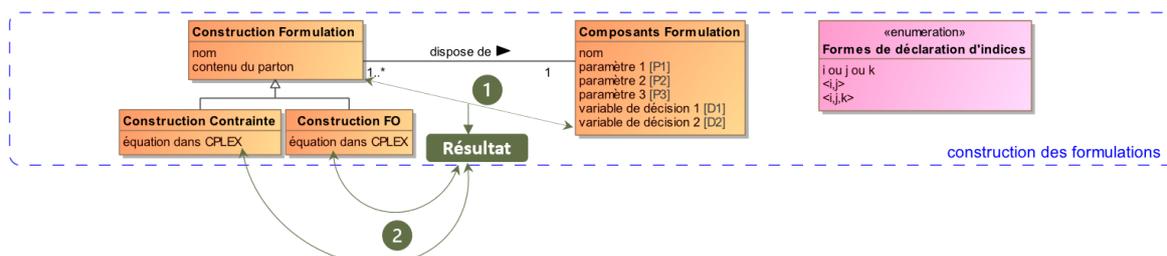


Figure 71 : Illustration de navigation pour la formulation de contrainte « Flux Entrant »

Algorithme 4 : l'assemblage composants-patrons pour la formulation de contrainte « Flux entrant »

```

1  Pour  x dans tous les objets du « Composants Formulation » dans le M-S // parcourir
2  |      Si      (x.nom = "Flux entrant")
3  |      |      nom ← x.nom
4  |      |      P1 ← x.paramètre 1 [P1]
5  |      |      P2 ← x.paramètre 2 [P2]
6  |      |      D1 ← x.variable de décision 1 [D1]
7  |      Fin
8  |      Pour  κ dans tous les objets de la « Construction Contrainte » dans le M-S // parcourir
9  |      |      Si      (κ.nom = nom)
10 |      |      |      σ ← κ.contenu du patron // prendre le contenu du patron du « Flux entrant »
11 |      |      |      η ← Affecter {P1, P2, D1}aux P1, P2, D1 concernés dans le σ // assemblage
12 |      |      |      κ.équation dans CPLEX ← η // affichage des résultat d'assemblage
13 |      |      Fin
14 |      Fin
15 Fin

```

La Figure 72 est l'état partiel du code OPL suite à l'application des règles navigation dans le MM-S.

```

int nombre_total_de_noeuds_de_ville

int indice_de_noeud_initial

range Ensemble_Noeud_Ville
    indice_de_noeud_initial
    nombre_total_de_noeuds_de_ville;

tuple Arc_Route
    int ID_du_noeud_de_depart
    int ID_du_noeud_de_fin

setof(Arc_Route) ensemble_des_tuples_debut_fin
int matrice_de_distance[ensemble_des_tuples_debut_fin]
boolean dvar_affectation[ensemble_des_tuples_debut_fin]
int+ dvar_aucune_sous_tournee[Ensemble_Noeud_Ville]

minimize sum (<i,j> in ensemble_des_tuples_debut_fin)

    matrice_de_distance[<i,j>]*dvar_affectation[<i,j>];
forall( i in Ensemble_Noeud_Ville, <i,j> in ensemble_des_tuples_debut_fin )
    sum( j in Ensemble_Noeud_Ville:j!=i ) dvar_affectation[<i,j>]=1;

forall( j in Ensemble_Noeud_Ville, <i,j> in ensemble_des_tuples_debut_fin )
    sum( i in Ensemble_Noeud_Ville:i!=j ) dvar_affectation[<i,j>]=1;

forall(i in Ensemble_Noeud_Ville: i>1, j in Ensemble_Noeud_Ville:j>1&&j!=i)
    dvar_aucune_sous_tournee[i]-dvar_aucune_sous_tournee[j]+
    (nombre_total_de_noeuds_de_ville-1)*dvar_affectation[<i,j>]
    <=nombre_total_de_noeuds_de_ville-2;

```

Figure 72 : Codes OPL suite à l'application des règles de navigation dans le MM-S

3.1.3.3. Règle de *parsing* du M-G et du M-S

Ce processus joue un rôle de parseur du modèle M-G et M-S. Ces deux modèles sont disponibles au format XML, car les modèles créés dans ADOxx peuvent être ainsi exportés. Rappelons qu'un modèle OPL dans CPLEX se compose de deux parties : un fichier de données (*.dat*) et un fichier de modèle (*.mod*). Le parseur (analyseur syntaxique) opère pour extraire du M-S le fichier cible « *.mod* », d'une part. Il effectue aussi des extractions et exportations du M-G vers le fichier cible « *.dat* », qui renseigne les valeurs des paramètres concernés définis dans le « *.mod* »

(« *indice_de_noeud_initial* », « *nombre_total_de_noeuds_de_ville* », « *ensemble_des_tuples_debut_fin* », « *matrice_de_distance* », par exemple), d'autre part.

Le parseur est développé en Java dans une plateforme JAXB¹ (Java Architecture for XML Binding), qui fournit un mécanisme rapide et pratique pour lier des schémas XML et des représentations Java. Ceci nous permet d'intégrer facilement des données XML et des fonctions de traitement dans les applications Java. De plus, une bibliothèque Java pour structurer des données par des algorithmes de traitement de graphes (JGraphT²) a également été mise en application. Cette librairie permet de réaliser l'analyse de la syntaxe (parser) à partir de graphe les informations dans le modèle M-G et M-S.

Pour terminer la génération du modèle OPL, une série de règles développées en Java affecte les mots clés de premier niveau (cf. Figure 73) :

- Les variables de décision « *dvar_aucune_sous_tournée* » et « *dvar_affectation* » doivent avoir un mot clé « *dvar* » en préfixe.
- L'ensemble de contraintes doit être entourée par « *subject to {...}* ».

Cette phase permet la mise en forme de deux fichiers de sortie en ajoutant par exemple :

- Le « =... ; » qui suit les paramètres déclarés.
- Le « { } » pour les données du type « *tuple* ».
- Le connecteur « ... » pour les données du type « *range* ».
- Le séparateur « ; » ou l'espace (signifié par « \n » en expression régulière dans Java) pour isoler les données et les formulations spécifiées.

```
//Pour .dat du modèle OPL
public static String data(String dataName,String dataValue) {
    return dataName+"="+dataValue+";";
}
//Pour .mod du modèle OPL
public static String simpleParameter(String type, String paramName) {
    return type+" "+paramName+"="+"...";
}
//Parser les paramètres avec le type primitif et affecter les mots clés de l'OPL
public static String rangeParameter(String type,String paramName, String compoName1,String compoName2) {
    return type+" "+paramName+"="+compoName1+".."+compoName2+";";
}
//Parser les paramètres avec le type "range" et affecter les mots clés de l'OPL
public static String setOfParameter(String type,String paramName, String subScript) {
    return type+"("+subScript+")"+" "+paramName+"="+"...";
}
//Parser les paramètres avec le type "setOf" et affecter les mots clés de l'OPL
public static String tupleParameter(String type,String paramName,String compoType1,String compoName1,String compoName2) {
    return type+" "+paramName+"{"+"\\n\\t"+compoType1+" "+compoName1+";"+\\"n\\t"+compoType2+" "+compoName2+";"+\\"n"+"}";
}
//Parser les paramètres avec le type "tuple" et affecter les mots clés de l'OPL
public static String arrayParameter(String typeArr, String typeBase,String paramName, String subScript) {
    return typeBase+" "+ paramName +typeArr.substring(0,1)+subScript+typeArr.substring(1,2)+"="+"...";
}
//Parser les paramètres avec le type "array" et affecter les mots clés de l'OPL
public static String dvar2Dim(String typeArr, String typeBase,String paramName, String subScript) {
    return "dvar"+" "+typeBase+" "+paramName+typeArr.substring(0,1)+subScript+typeArr.substring(1,2)+";";
}
//Parser les variables de décision et affecter les mots clés de l'OPL
public static String constraintTSP(List<String> constList) {
    String constContent = "";
    for (String constC : constList) {
        constContent=constContent+constC+"\\n";
    }
    return "subject to {"+"\\n"+constContent+"}";
}
//Affecter les mots clés pour l'ensemble de contraintes du modèle
```

Figure 73 : Règles de *parsing* basées sur une fonction Java pour l'affectation des mots clés et la mise en forme de fichiers CPLEX

¹ Développons en Java – JAXB : <https://www.jmdoudoux.fr/java/dej/chap-jaxb.htm>

² JGraphT: <https://jgraph.org/>

3.2. Transformation CIM-PIM

Cet atelier de transformation traduit le modèle de besoin de coordination M-C (niveau CIM) en ce modèle M-G de niveau PIM.

La Figure 74 illustre par un exemple une telle traduction pour notre exemple d'un TSP à quatre villes. Il est frappant de constater une grande équivalence entre les représentations graphiques des deux modèles sur cette figure. Mais les raffinements mathématiques fruits de cette transformation ne se traduisent que partiellement sous une forme visuelle. Toutefois, la carte géographique qui sert de fond au M-C a disparu dans le M-G. Les icônes d'orientation et d'exigences n'ont pas du tout la même signification que la fonction objectif et les contraintes. On voit apparaître les ensembles qui vont être à la base de la formulation en recherche opérationnelle dans le M-G.

Le lecteur du chapitre IV de ce mémoire de thèse a déjà dévoilé la signification des syntaxes concrètes et la sémantique du M-C. Nous avons donc ici à faire un simple rappel du méta-modèle MM-C. Comme le modèle M-G et son méta-modèle MM-G ont fait l'objet d'une large présentation à la section 3.1.1 de ce chapitre, le contenu original de ce qui suit sera la présentation des règles de transformation M2M.

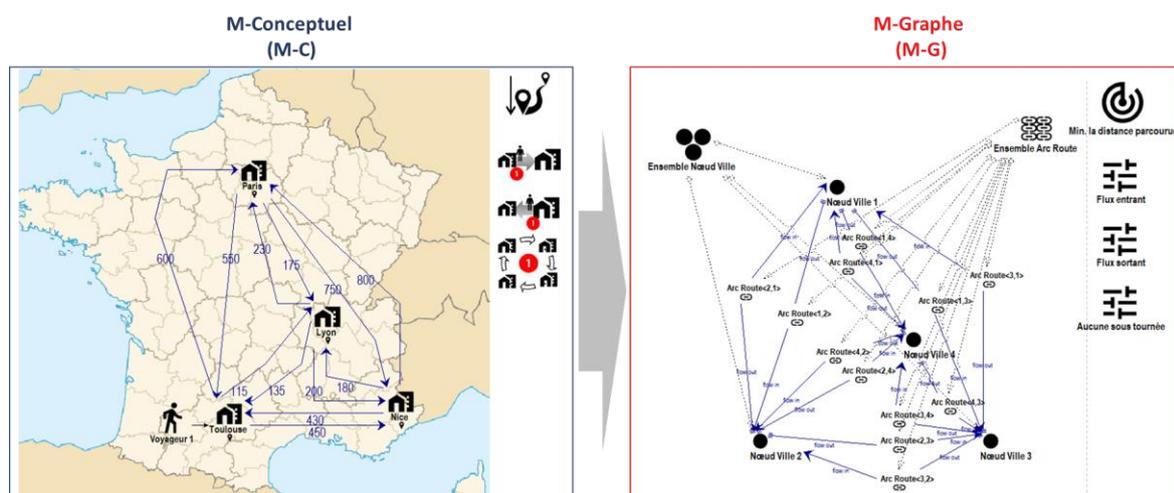


Figure 74 : Traduction de M-C en M-G pour un cas d'étude

Nous rappelons ce premier méta-modèle à la source de la transformation. Il est consigné sur la Figure 75. Ce MM-C n'est qu'une partie du méta-modèle du DSML-TSP, celle qui est impliquée dans les règles. La partie qui est éludée a servi à définir l'interface homme-machine pour saisir les données d'un problème de coordination opérationnelle de PAD en fin de chapitre IV.

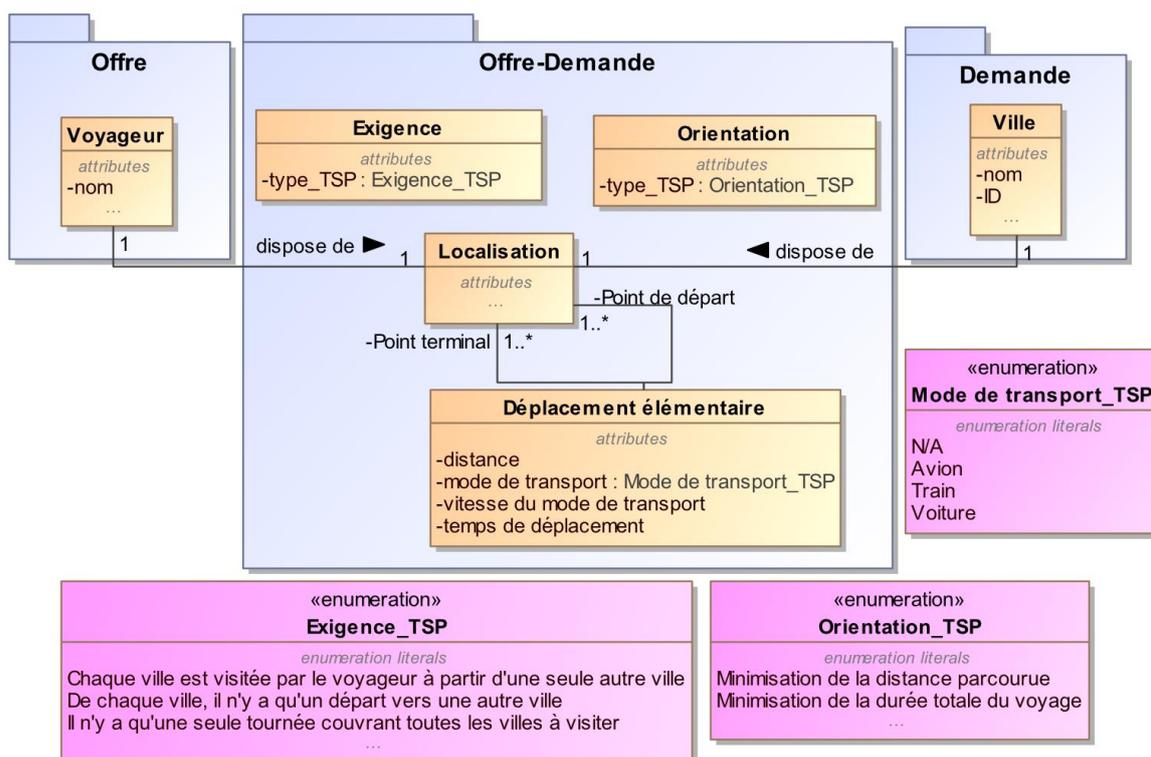


Figure 75 : Modèle Conceptuel (MM-C) de niveau CIM, sous ensemble du méta-modèle du DSML-TSP

Le résultat de la transformation est notre modèle cible M-G, qui doit être conforme au méta-modèle MM-G (cf. la Figure 65).

3.2.1. Processus de transformation

Les processus de transformation se compose de deux parties.

- (1) La première partie est une transformation qui obéit à une logique « M2M » fondée sur des règles de mapping entre MM-C et MM-G.
- (2) La seconde partie prône une navigation interne au PIM. Cela pour générer les données du graphe créé à partir des entités résultant de la génération du M-G par le M2M, ainsi qu'une spécification de contenu des contraintes et de la fonction traduisant l'objectif.

3.2.1.1. Règles de mapping entre MM-C et MM-G

Le processus de mapping s'opère progressivement, en 3 étapes comme indiqué dans le Tableau 30:

Tableau 30 : Etapes à suivre pour l'application des règles de mapping entre MM-C et MM-G

1	Dans un premier temps, nous tissons le mapping par une relation bijective d'une « Ville » à un « Nœud Ville ». La valeur de l'attribut d'identification dans la cible est celle de l'identifiant de la source « Ville » ainsi copiée dans le « Nœud Ville ».
2	La deuxième étape concerne la construction des chemins entre les villes. Le « Déplacement élémentaire » (une flèche représente la relation en M-C) spécifie la distance ou le temps de déplacement entre chaque paire de villes. Dans le M-G, la relation sera définie dans un format de triplet [« Flux Entrant » (relation) – « Arc Route » (classe) – « Flux Sortant » (relation)]. Ceci permet de prémâcher le travail à venir pour la contrainte d'une entrée unique et d'une sortie unique dans chaque ville (nommées " Flux entrant " et " Flux sortant ") si la tournée

	voulue est un circuit hamiltonien. Quant aux attributs, nous nous concentrons non seulement sur la distance ou le temps, mais aussi sur l'identification des arcs autour de ces « Nœud Ville » avec une cohérence entre nœuds et arcs.
3	Enfin, l'« Exigence » et l'« Orientation » dans le MM-C sont traitées. Ces concepts sont transférés respectivement aux composants « Contrainte » et « Fonction objectif » dans le MM-G. Par cette opération, nous fournissons un pointeur pour indiquer le patron des équations relatives dans le modèle mathématique présenté par la suite

La Figure 76 illustre l'état partiel du M-G, résultant de la seule application de ces règles de mapping entre MM-C et MM-G

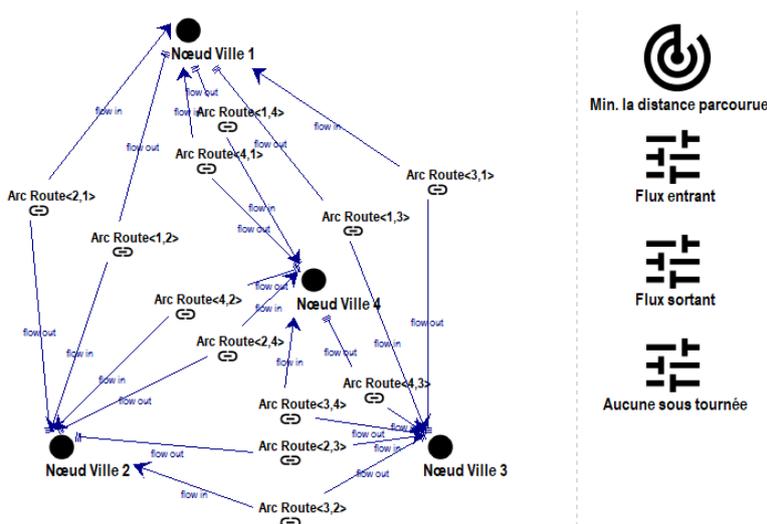


Figure 76 : Traduction de M-C en M-G pour un cas d'étude : l'état partiel du M-G résultant de la seule application des règles mapping entre MM-C et MM-G

3.2.1.2. Règles de navigation dans le MM-G

Cette navigation dans le MM-G structure les éléments venant de la traduction en des ensembles pertinents pour une bonne formulation mathématique. Ce sont les concepts « **Ensemble Nœud Ville** » et « **Ensemble Arc Route** » définis dans le MM-G qui sont à la source de cette transformation interne au MM-G. L'objectif est de regrouper la valeur des attributs de même type dans des classes homogènes, comme l'ensemble des distances définies dans une matrice de routage, par exemple. Ceci est réalisé par des règles dites de navigation en deux étapes comme illustré dans le Tableau 31.

Tableau 31 : Etapes à suivre pour l'application des règles de navigation dans le MM-G

1	La vocation de ces règles de navigation est d'abord de collecter une valeur singulière (ex : une distance) en entrée et de l'insérer dans la bonne collection (le bon ensemble) en sortie (ex : une matrice de distance), à sa juste place.
2	Ensuite, il s'agit de l'identification du nombre total du nœud ville ainsi que l'« <i>indice de nœud initial</i> ».

La Figure 77 schématise les règles de mapping appliquées pour soutenir ce processus de transformation des modèles d'une part, et les règles de navigation débouchant sur la génération des « ensembles » écrite en AdoScript, d'autre part.

Les lecteur peut se référer à (Zhang et al., 2020a) pour accéder au pseudo code détaillé concernant ces deux catégories de règles. Le M-G est généré à la fin de l'exécution de ces règles quand toutes les sources de connaissance ont été traitées.

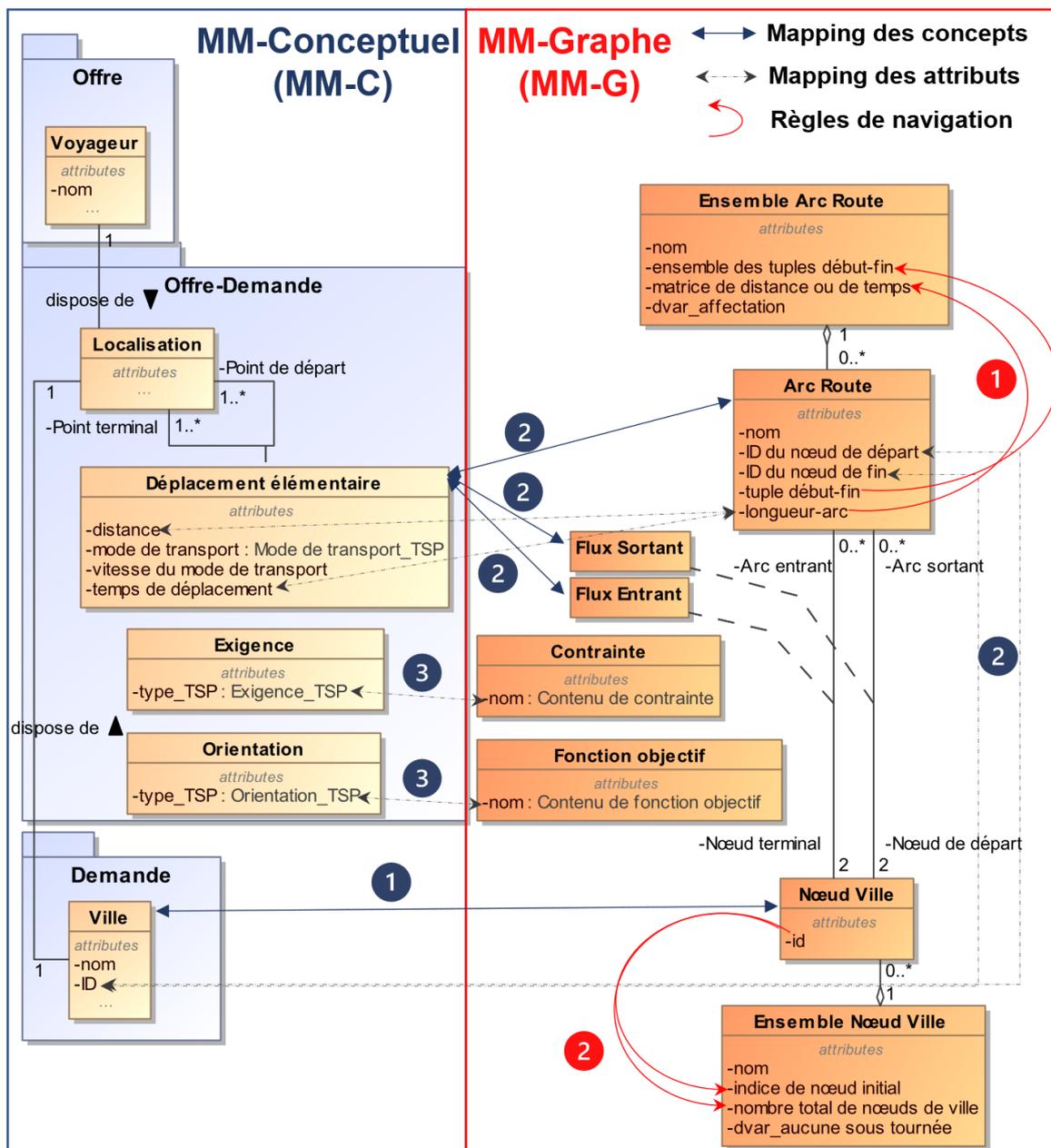


Figure 77 : Règles de mapping entre MM-C et MM-G et les règles de navigation dans MM-G

3.3. Résolution du modèle par CPLEX

Nous savons comment procède la fabrique de la coordination BLAutom, c'est-à-dire les opérations réalisées dans nos deux ateliers de transformation pour obtenir le modèle M-S en OPL à partir du modèle M-C en DSML-TSP. Le modèle M-S fournit des codes sources permettant de lancer le calcul sur CPLEX. Le solveur exécute les calculs d'une méthode exacte d'optimisation combinatoire afin de générer une proposition d'optimum, les meilleures décisions définissant la tournée.

Toujours sur notre même exemple de la Figure 74, nous l'avons mis en application sur BLAutom avec le jeu de données choisi qui est visible sur la Figure 78, la fabrique de la

coordination a fonctionné. Nous avons un premier résultat positif : la tournée retenue pour le voyageur « voyageur 1 » est la suivante : Toulouse→Paris→Lyon→Nice→Toulouse). Elle est optimale et le code soumis à CPLEX est celui que nous aurions créé si nous avions fait le codage en « manuel ». La solution est donc validée.

Nous avons donc pu faire un retour vers la syntaxe concrète du modèle M-C pour visualiser ces résultats de manière graphique en changeant simplement la couleur des chemins retenus entre la série de villes formant la tournée. L'objectif atteint est de 1405 kms qualifiant une distance totale minimale et un temps de calcul très faible sur ce problème de très petite taille (cf. la Figure 78). Les trois exigences sont bien satisfaites comme l'indique l'affirmation portée sous forme d'annotation à côté de la question « satisfaite ? » de chaque exigence.

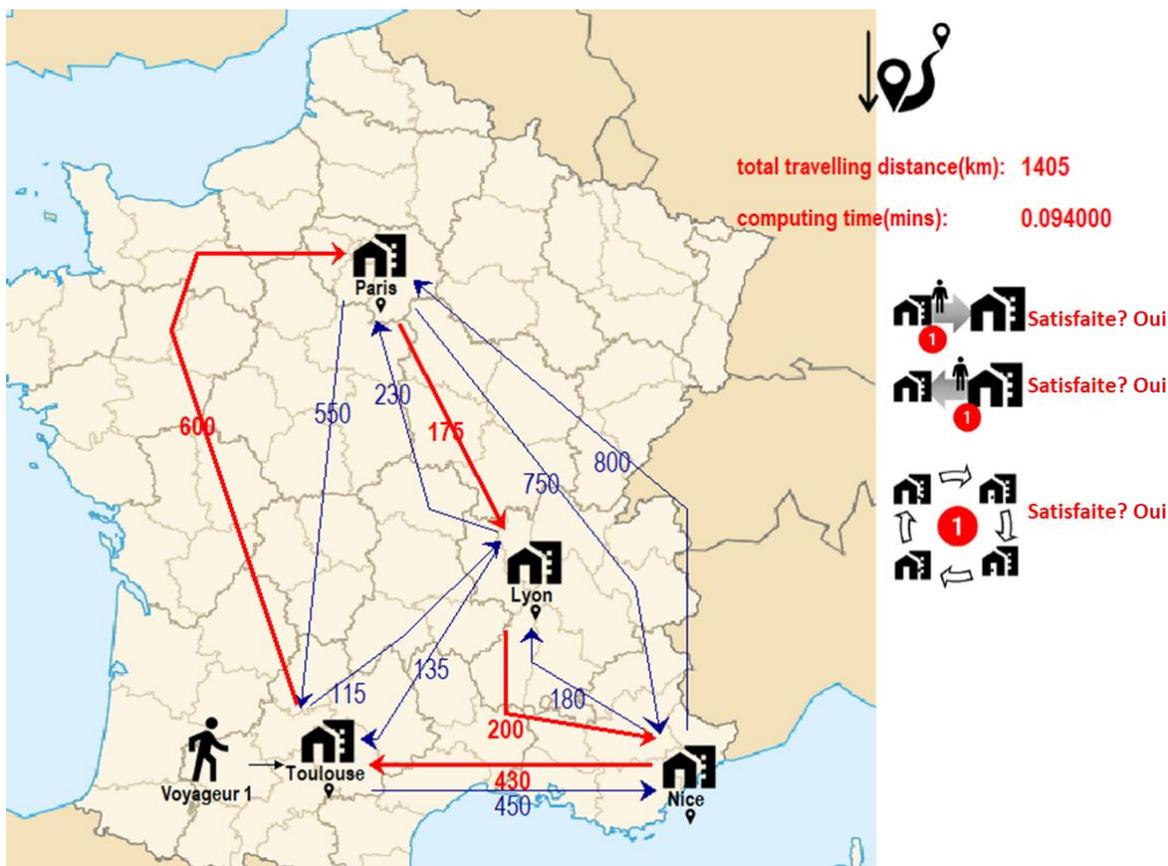


Figure 78 : Illustration de la visualisation de résultat par le cas d'études

3.3.1. Cas d'étude : changement du problème

Afin de d'éprouver notre prototype de BL.Autom, outre cette première expérience pour montrer sa réalité opérationnelle sur l'exemple simple, nous avons aussi souhaiter faire une preuve de concept plus étendue en montrant sa capacité à faire face à des changements de problème. Nous avons donc inclus deux cas d'utilisation supplémentaires de BL.Autom, correspondant à des problèmes TSP différents, dans notre plan d'expériences. L'idée est de travailler par réutilisation du premier modèle M-C de la Figure 74 pour ces deux cas d'utilisation :

- (1) Le premier cas consiste à modifier les contraintes prise en compte dans le TSP
- (2) Le second cas comporte une modification de l'objectif et de la dimension du problème.

3.3.1.1. Cas d'utilisation 1

Dans ce cas, nous avons toujours quatre villes à visiter par le voyageur « Voyageur 1 » qui sont : Toulouse, Paris, Lyon et Nice, mais si la fonction à optimiser est toujours une distance parcourue minimale, la contrainte de la tournée unique a été supprimée. Seules deux exigences sont suggérées :

- Chaque ville est visitée par le voyageur à partir d'une seule autre ville
- De chaque ville, il n'y a qu'un départ vers une autre ville

Les trois modèles correspondant aux 3 niveaux de l'architecture dirigée par les modèles sont présentés sur la Figure 79. Le contrôle ex post du code OPL généré par BL.Autom a été faite et nous l'avons validé. Nous avons donc été en mesure de lancer CPLEX et d'analyser les résultats.

Nous avons maintenant une solution formée par deux tournées distinctes. Une se fait sur l'aller-retour entre Paris et Toulouse, l'autre se fait sur l'aller-retour entre Lyon et Nice. On quitte bien l'unique circuit hamiltonien. La distance parcourue est plus faible, de 1285 kilomètres et CPLEX a mis 0,015 minutes (0,9 seconds) pour trouver ce résultat.

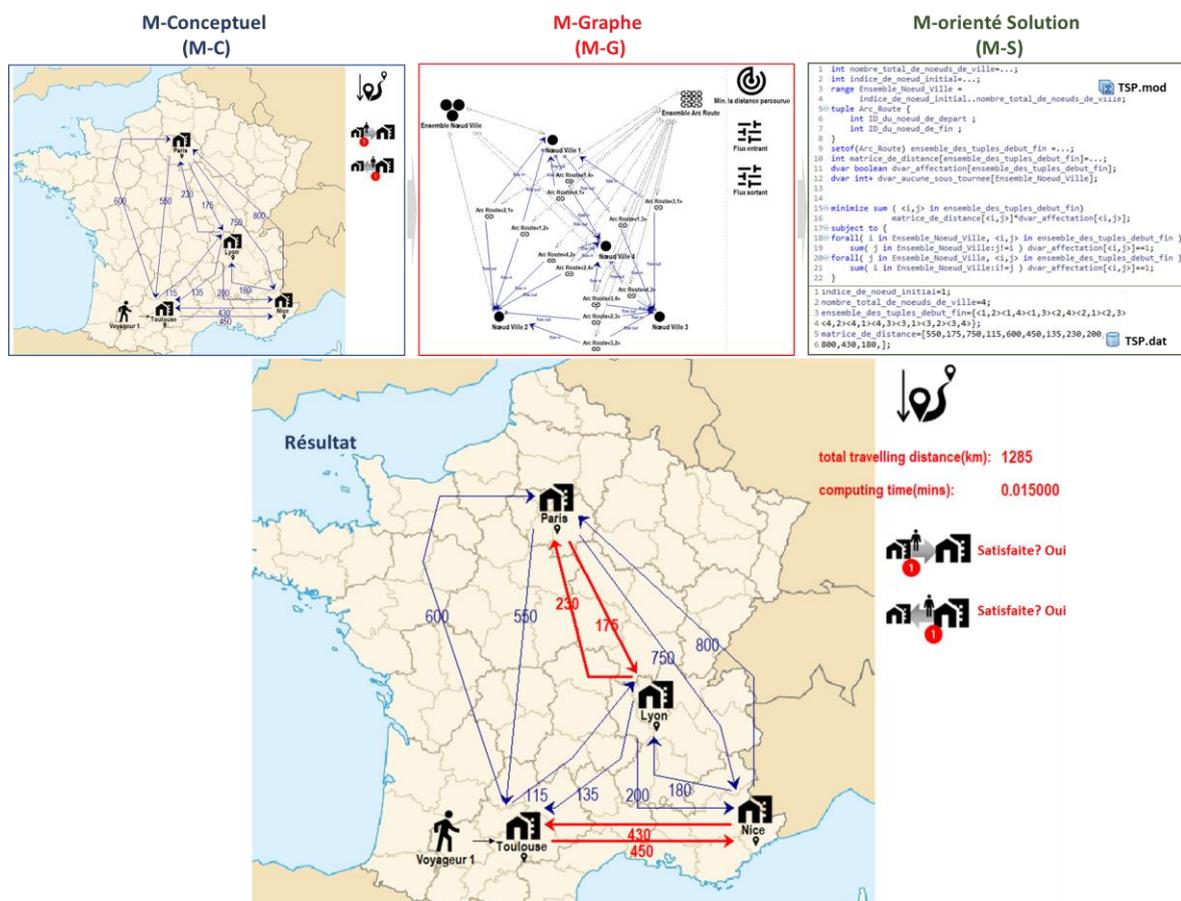


Figure 79 : Spécification du cas d'utilisation 1 du TSP différent : les trois modèles et le résultat obtenu

Pour réaliser cette expérience, l'effort a été minimal puisqu'il a suffi d'éliminer une exigence dans le M-C (l'icône de tournée unique) et de relancer la fabrique de la coordination.

3.3.1.2. Cas d'utilisation 2

Dans ce second cas, nous avons ajouté une nouvelle ville à visiter, Brest. Il s'agit maintenant d'un besoin de TSP avec cinq villes à visiter par le voyageur « Voyageur 1 » avec la prise en compte des mêmes trois exigences du TSP classique.

Cependant, nous modifions la formulation de la fonction à optimiser puisqu'un temps minimal de voyage est souhaité comme nouvelle orientation. Le passage à une caractérisation de chaque trajet entre deux villes par un temps de voyage plutôt qu'une distance est assisté par une interaction homme-machine dans la conception du M-C. Cela permet de spécifier le mode de transport employé pour chaque trajet. Ainsi, nous avons mixé les modes de transport en choisissant l'avion (la vitesse est estimée à 900km/h) pour les chemins de plus longue distance, (entre Toulouse et Brest, Toulouse et Paris, Nice et Paris, Lyon et Brest, Brest et Nice). Les liaisons pour les autres villes se font par le train (la vitesse est estimée à 320 km/h).

Le résultat obtenu par BL.Autom, et les trois modèles « fabriqués » sont présentés sur la Figure 80. De nouveau, le code OPL est validé par contrôle *a posteriori*.

Nous obtenons une tournée partant de Toulouse, passant par Paris, Lyon, Nice, Brest, et revenant finalement à Toulouse. Dans cette tournée, de Paris à Lyon (trajet 2) et de Lyon à Nice (trajet 3), le voyage se fait en train. Mais de Toulouse à Paris (trajet 1) et de Nice à Brest (trajet 4), c'est l'avion qui est choisi par l'optimiseur. Finalement le voyageur retourne de Brest à son point de départ, Toulouse, (trajet 5) en avion. La somme des temps de voyage est de 3h et 26 mins (206 minutes). Le temps de calcul utilisé par CPLEX pour exécuter M-S était de 0,094 minutes (5,64 seconds).

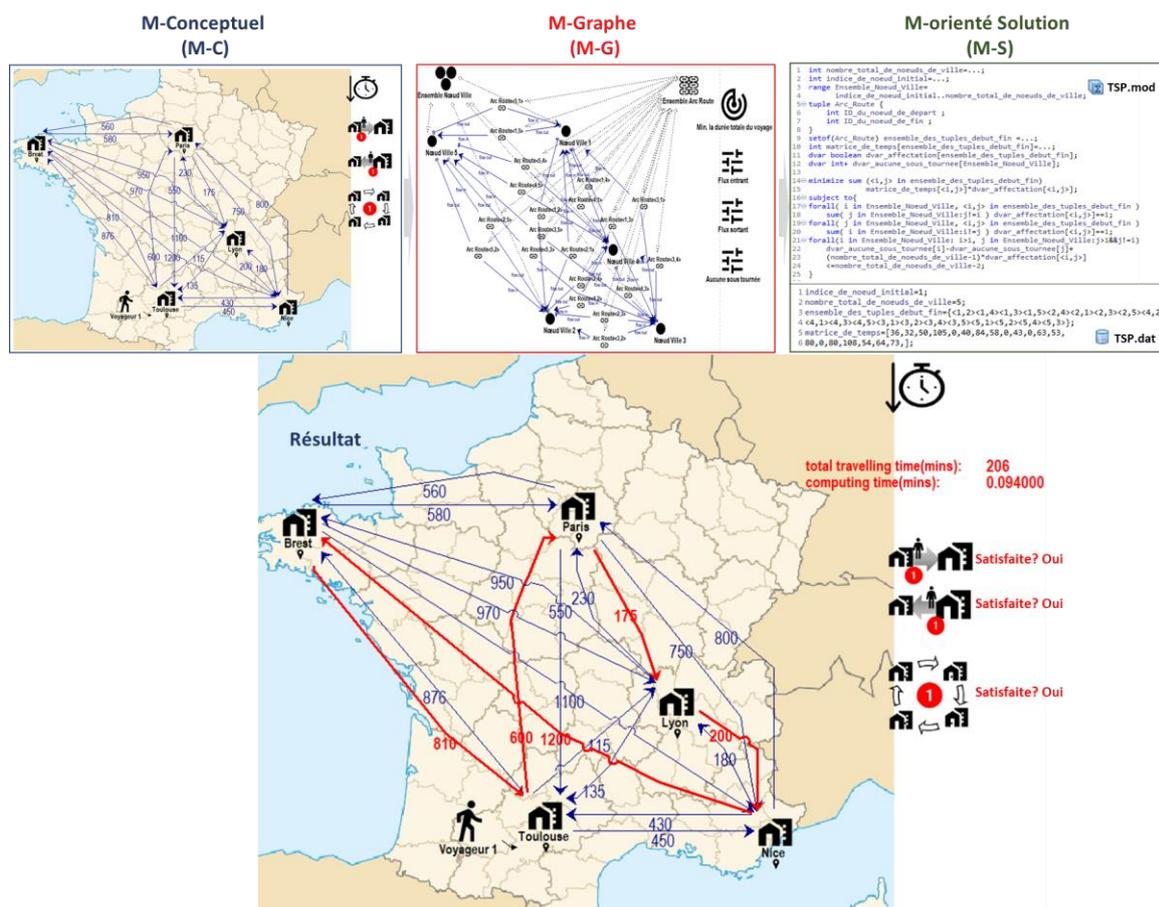


Figure 80 : Spécification du cas d'utilisation 2 du TSP différent : les trois modèles et le résultat obtenu

Pour ce cas d'utilisation, il a fallu ajouter la nouvelle ville et les trajets qui la concernent sur le modèle M-C, puis modifier l'icône de l'orientation des calculs (ici une montre symbolisant un temps de voyage).

4. Développement d'un outil de transformation en ADOxx : BL.Autom

Dans le cadre d'une vision architecturale de BL.Autom et en allant au-delà de l'analyse fonctionnelle déjà présentée ci-dessus de notre fabrique de la coordination, il faut donner un aperçu de sa couche logique. Nous le faisons dans cette section en fournissant quelques éléments techniques du développement de BL.Autom sur la plate-forme de méta-modélisation ADOxx.

En premier lieu, les trois méta-modèles MM-C, MM-G, MM-S sont écrits à l'aide des primitives d'ADOxx *Library Language* au sein du module intitulé *Développement Toolkit*. Les syntaxes concrètes des concepts (notations graphiques) du MM-C et du MM-G ont été implémentées avec un éditeur graphique *GraphRep* (cf. la section 5.1 du chapitre IV). Ensuite, les deux catégories de règles de transformation que sont les règles de mapping et les règles de navigation ont systématiquement été rédigées à l'aide du langage de transformation de modèles AdoScript. Ces règles sont intégrées au moteur de la transformation par une interface dynamique intitulée « *External coupling* » qui permet d'ajouter des fonctionnalités supplémentaires à la plate-forme. La Figure 81 présente quelques captures d'écrans de ces choix d'implémentation de BL.Autom dans ADOxx.

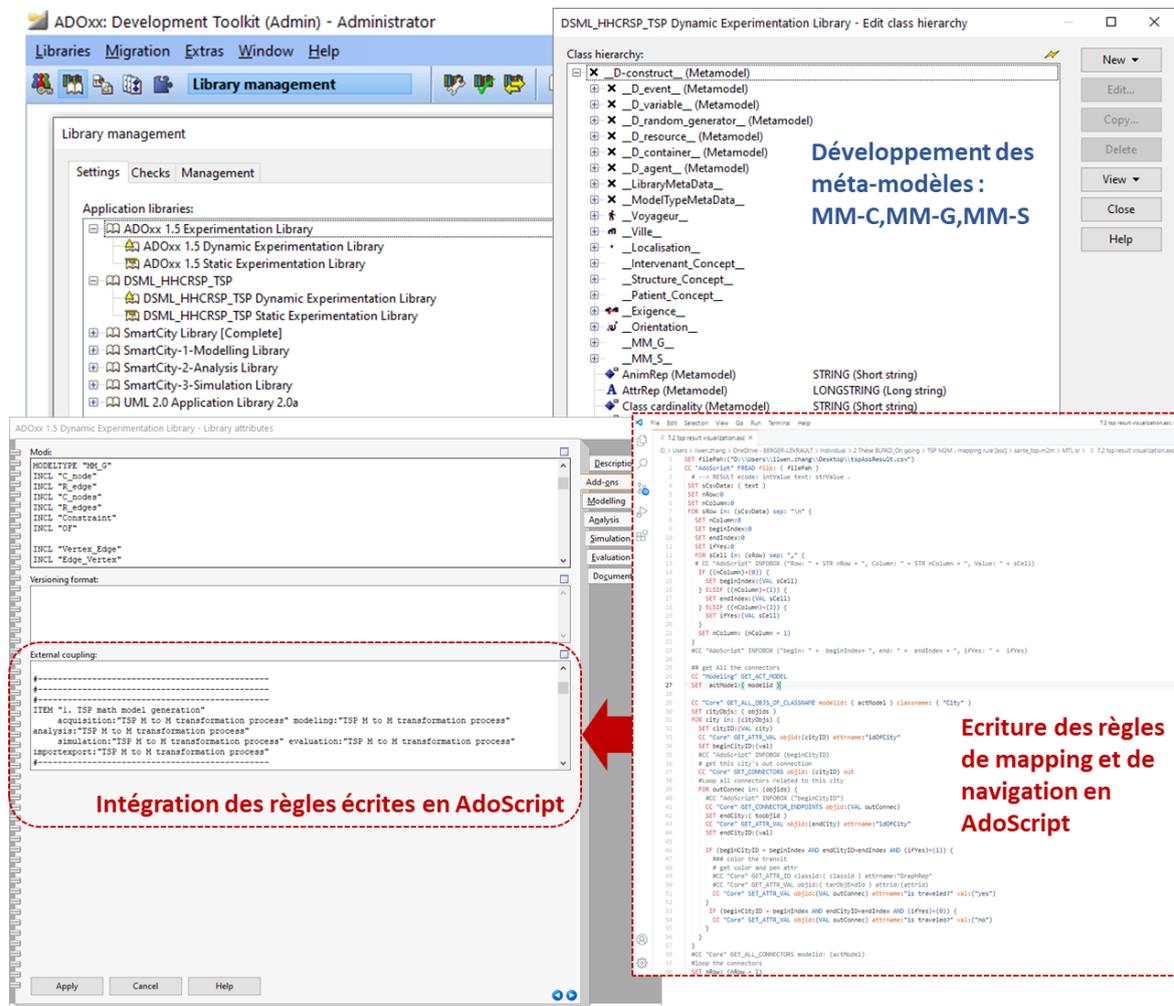


Figure 81 : Captures d'écran de l'implémentation de l'outil de transformation BL.Autom

Concernant la mise en exploitation du prototype de BL.Autom, la Figure 82 présente une capture d'écran. C'est l'interface qui récupère le modèle M-C. La fonctionnalité « *TSP M to M transformation process* » est accessible par un menu de la barre supérieure et son activation par

sélection est l'événement qui lance la transformation des modèles par l'option « *TSP math model generation* ». Quand l'autre option est sélectionnée, « *TSP result visualization* », c'est que l'utilisateur veut publier les résultats des calculs par CPLEX, son appel en ligne ayant été intégré aux capacités de BL.Autom. C'est par ce chemin sur l'interface que le M-C est récupéré et que les résultats de l'optimisation sont superposés sur la représentation initiale.

Quant à la première option « *TSP math model generation* », à la fin des transformations, le modèle M-G est publié automatiquement sous une forme graphique (illustrée dans la Figure 82) d'une part, et le modèle M-S est exporté sous forme d'un fichier xml, d'autre part. Ce fichier est ensuite soumis à notre parseur (cf. la section 3.1.3.3 de ce chapitre) pour exporter à nouveau le contenu vers deux fichiers OPL (en *.mod* et en *.dat*).

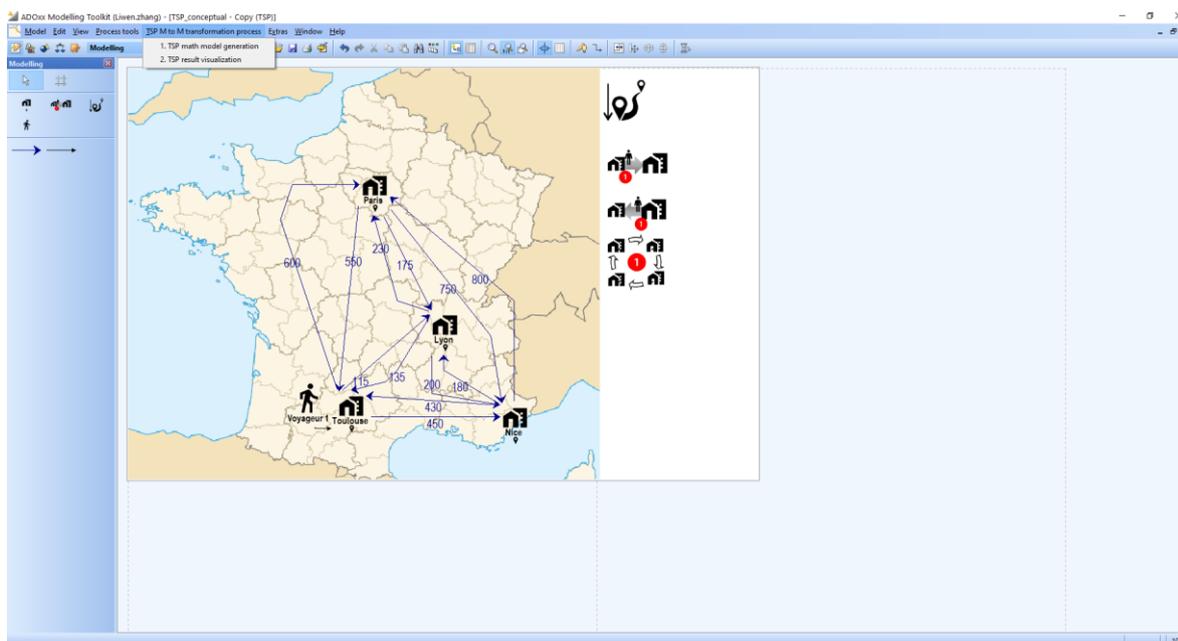


Figure 82 : Ecran de BL.Autom: le M-C avant appel de « *TSP M to M transformation process* »

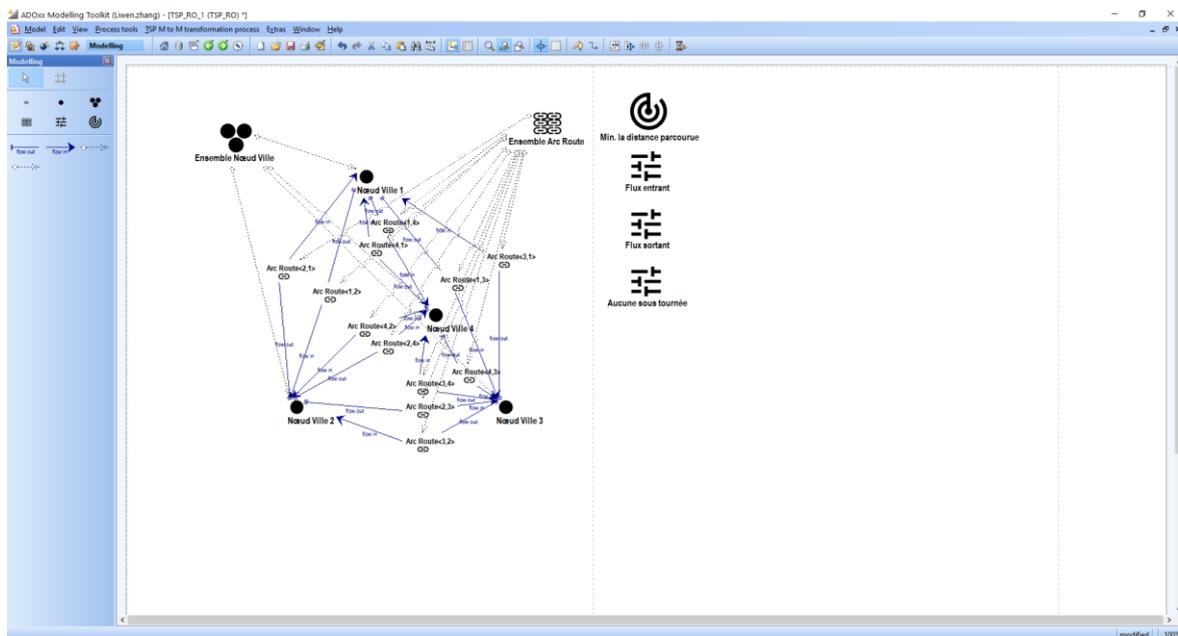


Figure 83 : Ecran de BL.Autom : M-G publié après transformation du M-C

Pour la deuxième option « *TSP result visualization* », il s'agit d'afficher le résultat de tournée généré par CPLEX. Le résultat d'un TSP est transmis à l'aide de deux fichiers de sortie (un au format texte, l'autre au format feuille de calcul), comme illustré dans la Figure 84 :

- Le premier fichier « *tspOFResult.txt* » contient la valeur de fonction objectif et le temps d'exécution requis pour l'obtenir.
- L'autre fichier « *tspAssResult.csv* » présente le triplet de trois valeurs pour chacune de ses lignes qui correspondent à l'ensemble des chemins entre les villes déclarés dans le M-C. Pour chaque ligne en colonne 1, le chiffre identifie une ville de départ. En colonne 2, il s'agit de l'identifiant de la ville d'arrivée, la dernière colonne est la valeur de la variable de décision binaire précisant si ce chemin est retenu ou pas (1 affecté, 0 sinon).

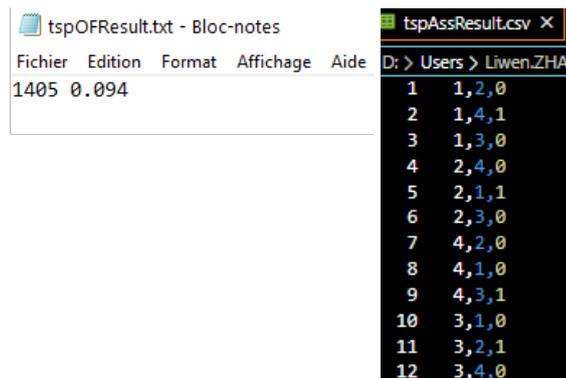


Figure 84 : Contenus des fichiers de sortie de BL.Autom après appel de CPLEX

Ainsi, à partir de ces contenus de la Figure 84, la solution du TSP est trouvée à partir des lignes 2,5,9 et 11. Les chemins retenus sont donc (1,4), (4,3), (3,2), (2,1). Il en découle une tournée (1,4,3,2,1) (cf. la Figure 78).

Un décideur n'ayant pas des connaissances mathématiques et ayant contribué au **M-C** aura naturellement envie de retrouver des propositions dans le même univers de discours. Un décideur appréciera certainement de disposer d'une IHM afin d'obtenir de manière efficace et compréhensible des connaissances correctes et compréhensibles sur le TSP obtenu. Les directives d'AdoScript que nous avons développées dans cette optique aboutissent à une visualisation de ces résultats en 2 étapes :

- Analyser les informations du fichier « *tspOFResult.txt* », puis les afficher en conformité avec le contenu du concept « **Orientation** » dans l'interface du M-C, sous les intitulés de « *total travelling time(mins)* » (ou « *total travelling distance(km)* ») et de « *computing time(mins)* » respectivement.
- Analyser les informations dans « *tspAssResult.csv* », et les afficher pour matérialiser la tournée du voyageur. Au niveau de l'affichage, une coloration en rouge a simplement été mise en application lorsque le chemin entre deux villes est retenu dans la solution optimale.

La Figure 78 est une illustration de ce résultat obtenu par l'option « *TSP result visualization* ».

5. Conclusion

Dans ce chapitre, nous décrivons donc le mécanisme d'aide à la décision au moyen d'une fabrique de la coordination, suivant les trois niveaux (CIM-PIM-PSM) de l'architecture dirigée par les modèles (MDA). La fabrique opère des évolutions de connaissances depuis un stade préliminaire d'analyse des besoins métier jusqu'à une proposition de décisions satisfaisantes au regard de ces besoins. Trois modèles complémentaires jalonnent séquentiellement cette chaîne, un par niveau MDA :

- Le modèle CIM : le modèle de besoins de prise de décision.
- Le modèle PIM : le modèle mathématique pour l'aide à la décision.
- Le modèle PSM : le modèle logique écrit en OPL (.dat et .mod) qui est une hypothèse posée en termes de calcul pour l'optimisation par le solveur CPLEX.

Nous avons mis l'accent sur la transformation des modèles entre ces trois modèles en adoptant un langage de transformation approprié (MTL) appliqué dans 2 modules (CIM→PIM, PIM→PSM).

Pour expliquer notre démarche d'IDM, nous avons décidé de décrire l'ensemble de la chaîne de transformation qu'est cette fabrique de la coordination à l'aide d'un exemple simple : le problème du voyageur de commerce (TSP). Ce sujet n'est pas éloigné de notre problématique de la coordination de la PAD puisqu'il traite de tournées. Pour cette phase, nous avons spécifié les connaissances à manipuler au cours des étapes principales du processus de transformation souhaités :

- Les trois méta-modèles relatifs à chaque niveau MDA : le Méta-Modèle des besoins (MM-C) pour le CIM, le Méta-Modèle du Graphe de RO (MM-G) pour le PIM et le Méta-Modèle orienté Solution (MM-S)
- Les règles explicites : les règles de mapping entre les méta-modèles, les règles de navigation et les règles de *parsing* entre et dans les méta-modèles.

L'implémentation de l'outil de transformation BL.Autom s'est faite sur la plate-forme de méta-modélisation ADOxx. Ce développement expliqué ici pour la résolution par méthode exacte a visé la production automatique d'un code en langage OPL (le M-S) qui est ensuite soumis au solveur CPLEX. L'implémentation de notre prototype de cette fabrique, nommé BL.Autom, a permis de mener des expériences sur plusieurs variantes de TSP. Le retour d'expérience prouve la faisabilité et montre la valeur de notre proposition, en pratique.

CHAPITRE VII. CONCLUSION GENERALE

CHAPITRE VII. CONCLUSION GENERALE	181
1. SYNTHESE DES CONTRIBUTIONS	182
2. PERSPECTIVES	184
2.1. Pistes d'amélioration du DSML-HHCRSP	184
2.2. Benchmarking des méthodes de résolution	185
2.3. Prise en compte des incertitudes par coexistence entre l'optimisation et la simulation.....	187
2.4. Renforcement de notre chaine de transformation.....	188

1. Synthèse des contributions

Les travaux de cette thèse s'inscrivent dans le contexte de la coordination holistique de prises en charge à domicile. Nous avons choisi de limiter notre périmètre d'études à la coordination opérationnelle. Le problème de la coordination opérationnelle en PAD inclut une problématique d'affectation de ressources et de planification de tournées. Ces problèmes sont décrits dans la littérature sous l'intitulé « *Home Health Care Routing and Scheduling Problem – HHCRSP* ». Notre objectif est de proposer un système d'aide au pilotage des interventions à domicile de professionnels sociaux, médicaux, médico-sociaux et paramédicaux autour du patient bénéficiaire pour organiser collectivement leur quotidien. En cela, la fonction de coordination opérationnelle doit être supportée par un environnement numérique pour remplir des objectifs que nous allons définir. Nous avons répondu à cette demande de soutien par le développement d'un prototype et une preuve de concept. Le début du pilotage débute par la production de plans par un coordinateur.

A travers la capitalisation des connaissances bibliographiques, nous capturons de manière exhaustive et formelle les connaissances utiles pour traiter de ce problème de coordination opérationnelle de PAD. Nous utilisons une structure basée sur les rapports entre l'offre et la demande de services de soins à domicile. Cette classification donne une répartition des artefacts de modélisation en deux volets, un caractérisant les acteurs délivrant les services, l'autre qualifiant les attentes des bénéficiaires de mêmes services. Fort de cette classification pertinente, nous analysons de manière approfondie les concepts fondamentaux liés à l'offre et à la demande en vue de réaliser l'organisation du pilotage dans la prise en charge à domicile sur la base de différentes mécaniques d'optimisation.

Nous proposons une architecture modulaire d'aide à la décision de prise en charge à domicile. Cette architecture se compose des trois parties de l'architecture dirigée par les modèles : CIM-PIM-PSM.

Le premier module CIM a pour fonction la définition de la prise en charge à domicile. Il fournit les primitives de modélisation dans un langage spécifique au domaine (DSML) et fidèles à la vision opérationnelle du terrain. Il permettra de soutenir des exigences de personnalisation de masse du processus de prise en charge d'un bénéficiaire. **Il s'agit de la première contribution, le développement d'un DSML intitulé « DSML-HHCRSP » basé sur un DSML noyau intitulé « DSML-TSP » pour capturer le besoin de coordination centralisée au niveau opérationnel.** Le TSP étant un problème simplifié du HHCRSP, les deux problèmes font partie de la même famille. Le DSML-HHCRSP a été élaboré à partir d'un méta-modèle proche des éléments caractéristiques du domaine et fidèle au plus grand nombre de cas connus dans le système de santé national. Un soin particulier sera apporté à la sémantique des éléments de ce méta-modèle, en incluant un vocabulaire emprunté au monde professionnel concerné. L'implémentation du DSML-TSP et du DSML-HHCRSP sont réalisés à l'aide de l'environnement de méta-modélisation ADOxx.

Avec l'ambition de couvrir le plus grand nombre possible de structures de PAD, nous avons imaginé une « fabrique de la coordination » constituée de deux ateliers, relatifs aux étapes du processus d'IDM. Au cœur de cette fabrique, il faut déclarer les variables à calculer, de spécifier de fonction objectif à satisfaire, de décrire des contraintes qui découlent de l'expression du besoin par le DSML. La connaissance fournie par le modèle métier est traduite en un modèle

mathématique d'optimisation combinatoire. Ce modèle mathématique est ensuite traduit en un code informatique pour appeler un optimiseur en ligne, prenant en charge la résolution de problèmes par optimisation selon une méthode choisie. Ils produisent donc une solution de coordination opérationnelle pour les bénéficiaires et pour les intervenants en garantissant au maximum leur satisfaction. **Il s'agit de la deuxième contribution, la formulation et la résolution basée sur deux méthodes de recherche opérationnelle pour l'aide à la décision du problème de la coordination opérationnelle dans la PAD.**

Nous avons investi le champ de la formulation mathématique en mobilisant des règles métier peu adressées jusqu'alors : l'équilibrage en termes de difficulté des tournées, le respect au mieux des fenêtres de temps des interventions et des interopérations demandées par les bénéficiaires, en prenant en considération la tolérance personnalisable en cas de non-respect des limites. De surcroît, une fonction de régression linéaire est introduite pour expliquer la relation entre le temps dépassé toléré ou non-toléré, et le taux de satisfaction de chaque individu. Concernant les méthodes de résolution, deux méthodes sont utilisées :

- La méthode exacte basée sur la Programmation Linéaire en Nombres Mixtes (PLNM) pour générer la solution optimale en situation déterministe.
- La méthode approchée basée sur la Problème de satisfaction de contraintes (CSP) afin de résoudre les problèmes de grande taille avec un temps de calcul raisonnable.

Pour vérifier la faisabilité et la qualité de la solution, nous avons présenté les résultats d'une expérimentation basée sur 13 cas d'utilisation, avec une analyse critique des solutions générées par comparaison entre méthode exacte et méthode approchée. Les résultats montrent que (1) la méthode exacte est capable de trouver une solution optimale pour les cas de taille moyenne, (2) la méthode approchée arrive à trouver un résultat approximatif pour toutes les tailles de problèmes et (3) la qualité des résultats obtenus par la méthode approchée est élevée pour les problèmes qui peuvent être résolus avec les deux méthodes (l'écart entre les deux fonctions objectives est $\leq 5,4\%$).

Enfin, nous avons établi un pontage entre CIM, PIM et PSM. **C'est la dernière contribution de cette thèse ; il s'agit de réunir les résultats acquis dans les parties précédentes pour générer un outil de coordination opérationnelle basée sur l'architecture dirigée par les modèles.** Commençons par la présentation de deux niveaux d'architecture de notre environnement d'aide à la prise de décision pour des problèmes de coordination opérationnelle. La couche fonctionnelle est inspirée du processus cognitif de l'expert en recherche opérationnelle, décrit en quatre phases. La couche logique implémente ces fonctions attendues et a permis de décrire une suite des modules de traitement de trois modèles dans le respect des principes du MDA (CIM, PIM et PSM). Les processus de la transformation entre les modèles sont décrits comme une fabrique de la coordination. Le moyen de transformation repose sur des règles de transformation de connaissances (mapping et algorithmes de transformation) entre les méta-modèles, qui spécifient ce à quoi l'entrée et la sortie de l'atelier doivent être conformes. De plus, le moyen de fabrication repose sur des règles de navigation et de parsing dans chaque atelier, qui visent à assurer la qualité de la production des connaissances intermédiaires (PIM) ou des connaissances finies (PSM). Ce cadre nous a permis d'éclairer toutes les étapes pour un passage d'une connaissance métier à une solution de plannings par l'intermédiaire de connaissances en RO.

Pour prouver la faisabilité d'une telle chaîne IDM assimilable à une « fabrique de coordination », nous avons en pratique implémenté un prototype de transformation, BLAutom, sur ADOxx. Le développement pour la résolution par méthode exacte a visé la production automatique d'un code en langage OPL afin de soumettre la recherche de solutions au solveur CPLEX. La preuve de concept est obtenue à partir du problème du voyageur de commerce (TSP).

2. Perspectives

2.1. Pistes d'amélioration du DSML-HHCRSP

Dans le Chapitre II, nous observons qu'il y a une très grande diversité de structures et d'organisations qui remplissent des missions de PAD. Il est donc évident qu'il y a une grande diversité dans la représentation de ces organisations, représentation qui influe grandement sur le cadre de toute planification. Le DSML-HHCRSP développé dans le Chapitre IV est élaboré à partir d'un méta-modèle proche des éléments caractéristiques du domaine et fidèle au plus grand nombre de cas connus dans le système de santé national. Une des dimensions futures de ce module est la mise en place d'une analyse des sources d'incertitudes qui peuvent peser sur le bon fonctionnement de la coordination, et l'intégration dans le modèle des éléments de langage pour les représenter.

De plus, la conception du DSML n'a pas abordé l'aspect de la cohérence des choix faits sur les exigences et les orientations. Une piste de recherche à court terme est l'étude des interdépendances potentielles entre les exigences et les orientations pour une telle structure de PAD, par exemple pour une structure donnée, deux exigences telles que « le respect des coupures légales » et « le respect des priorités des interventions » peuvent avoir à coexister. Cette étude peut aider les coordinateurs, lors de l'énumération des exigences et des orientations d'une structure de PAD dans le modèle via DSML-HHCRSP, à sélectionner directement les éléments qui y sont associés et à éliminer ceux qui entrent en conflit.

Au niveau de l'usage du DSML-HHCRSP, nous avons détecté une possible limite dans l'IHM (cf. la Figure 39). Même si nous avons adopté une modélisation multi-vues, le modèle de besoin reste encore très surchargé graphiquement et peut réduire la capacité du coordinateur à exprimer et valider son besoin. Pour l'acquisition des connaissances à travers le modèle, ce qui leur a été proposé est une visualisation du modèle en 2D. Supposons que chaque intervenant ait son propre itinéraire préféré, un intervenant A préférant par exemple aller du bénéficiaire B1 au bénéficiaire B2 et ne souhaitant pas revenir de B2 à B1. Si de nombreux intervenants sont enregistrés auprès d'une structure de PAD, l'itinéraire personnalisé de chaque intervenant est difficile à représenter clairement à travers notre modèle 2D. Si nous insistons pour distinguer ces concepts personnalisés par des couleurs ou des notations particulières, certaines parties prenantes (les utilisateurs du langage de modélisation) risquent de ne pas comprendre pleinement tous les aspects liés à des notations de modélisation qui deviennent très sophistiquées. Ainsi, il est intéressant d'étudier de nouvelles pistes d'IHM qui peuvent aider à atteindre une compréhension commune et à augmenter la qualité pragmatique des modèles.

Récemment, dans (Gordeeva et al., 2021), une représentation 3D est proposée comme solution possible pour améliorer la perception du modèle et de son contenu. Afin d'augmenter l'implication de l'utilisateur pendant qu'il travaille sur un modèle, la solution proposée utilise la

réalité virtuelle comme forme de représentation 3D (cf. la Figure 85). Selon cet auteur, l'utilisateur peut naviguer dans les modèles et entre les modèles.

Cette nouvelle direction d'une modélisation en 3D à l'aide de la réalité virtuelle permet d'améliorer la compréhension de la syntaxe concrète d'un DSML et aussi d'améliorer la capacité des utilisateurs à accéder à la qualité sémantique de leur modèle. Dans ce mode de visualisation 3D, la zone d'affichage des informations est dotée de dimensions supplémentaires. Les coordinateurs pourraient visualiser les informations dans la dimension appropriée en fonction de leurs besoins, en lieu et place d'une kyrielle de fenêtres pop-up contenant les informations contextuelles des objets en mode 2D.

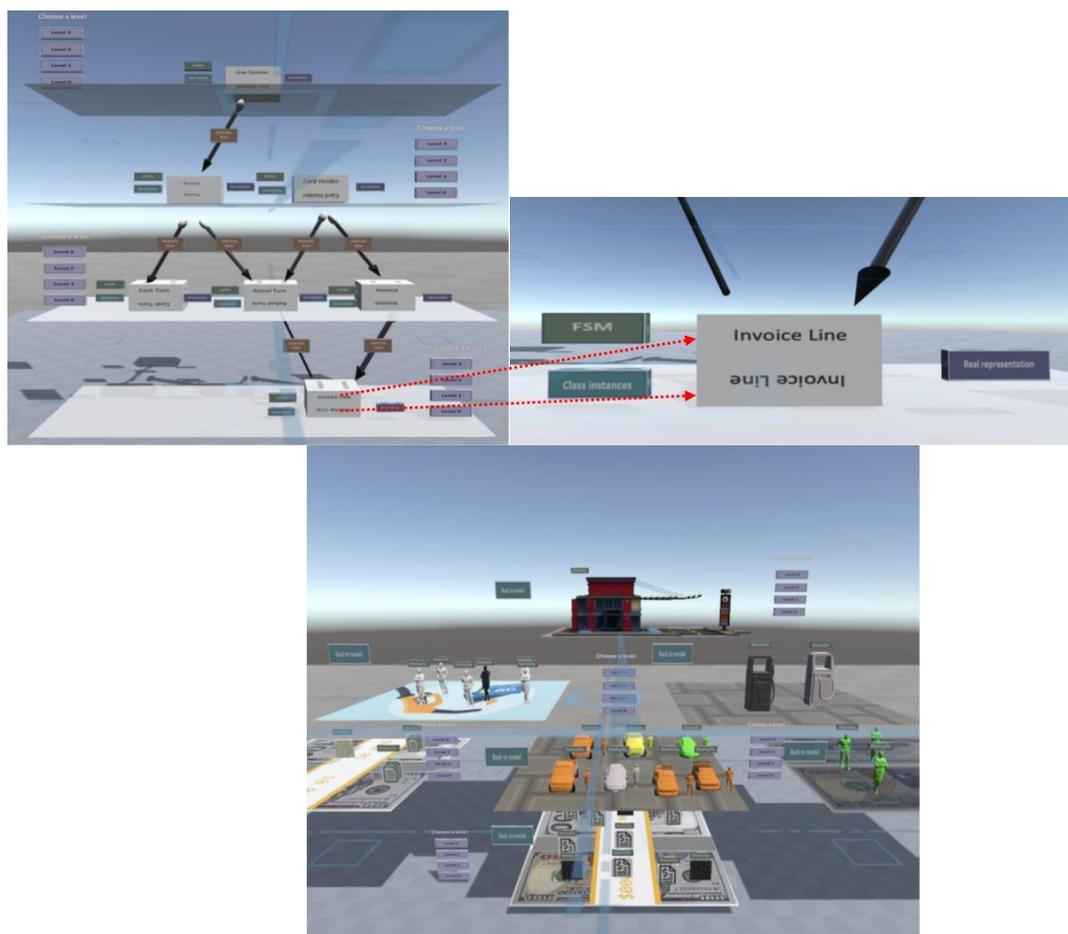


Figure 85 : Modélisation multi-acteurs avec la 3D et la réalité virtuelle (Gordeeva et al., 2021)

2.2. Benchmarking des méthodes de résolution

La multitude de formulations de problèmes de coordination opérationnelle de PAD est corrélée à une très grande palette de moyens de résolution numérique du problème formulé (cf. le Tableau 4). La formulation du problème est impactée par ce choix de résolution. Nous avons proposé dans le chapitre V deux approches différentes pour formuler les problèmes : les moyens de recherche de solutions sont variés. A court terme, nous envisageons d'abord de compléter ce travail par une étude de benchmark sur les méta-heuristiques, il s'agit de comparer les méthodes de résolution proposées sous OptaPlanner dans le chapitre V avec un service d'optimisation de planification des activités, développé et commercialisée par Berger-Levrault. Le but est de se forger une conviction sur les forces et faiblesses de cette palette de méthodes. La solution développée à ce jour par Berger-Levrault s'appuie sur un algorithme de « colonies de fourmis »

(ACO) et une modélisation sous la forme d'un problème de satisfaction de contrainte (CSP) (Chabane et al., 2019) que nous n'avons pas sélectionné dans cette thèse.

La revue de la littérature nous permettra de choisir des benchmarks de tailles différentes pour tester les solutions générées par les méthodes à comparer, nous pouvons citer quelques benchmarks accessibles (Zhang et al., 2020b) :

- Le benchmark publié par (Mankowska et al., 2014), incluant 7 groupes avec 70 instances au total (chaque groupe a 10 instances, générées aléatoirement), caractérisés par le nombre de bénéficiaires, le nombre de bénéficiaires qui demandent un service simple et un service double, le nombre de soignants.
- Le benchmark présenté par (Braekers et al., 2016a) incluant 90 instances : 30 petites (10-25 services) et 60 de tailles réalistes (50-300 services). Elles sont caractérisées par le nombre d'intervenants à temps plein (8-10 heures/jour), le nombre d'intervenants à temps partiel (4-6 heures/jour), le mode de transport (en voiture, en transport en commun ou mixte), le nombre d'interventions et le nombre de bénéficiaires.

Ensuite, dans l'idée de gagner en fiabilité, ces méthodes de résolution seront testées sur des données réelles de clients, et non uniquement sur les 13 cas d'utilisation présentés au chapitre V. Nous avons besoin de données provenant d'un plus grand nombre de clients de Berger-Levrault pour nous assurer de la faisabilité (le respect des contraintes formulées) et de la rigueur (la valeur de fonctions objectives) des plannings générés. Cette expérimentation devra être accompagnée d'une co-validation avec les clients de Berger-Levrault, afin de vérifier que les plannings générés soient, d'une part, cohérents et respectent toutes les contraintes de coordination opérationnelle et qu'ils soient, d'autre part, pertinents et efficaces par rapport aux plannings réalistes en pratique dans une structure de PAD. Aussi, cette phase de co-validation avec les clients nous permettra de recueillir de nouvelles exigences et orientations des différentes structures de PAD afin d'améliorer nos formulations, ce qui permet à notre solveur de couvrir à l'avenir plus de besoins de planification provenant de structures de PAD diverses. Nous pouvons citer par exemple quelques exigences moins adressées dans la littérature :

- La priorité de la demande : il serait plus opportun de donner un degré de priorité au type d'actes à réaliser, par exemple la distribution de médicaments obligatoires. De plus, si le bénéficiaire peut être entouré par sa famille, il est préférable que l'intervenant programme l'intervention en présence de la famille afin de s'assurer que ses instructions soient prises en compte.
- Le multimodal (Hiermann et al., 2015) : cette option donne la possibilité aux coordinateurs de sélectionner un modèle de transport mixte, si nécessaire, par exemple « transports en commun + vélo », afin de répondre aux facteurs économiques et écologiques ou à la flexibilité relative aux modalités de déplacement (« taxi + à pied » par exemple)

A long terme, notre ambition est d'avoir une architecture qui permette de choisir une option préférentielle de méthode de résolution adaptée à partir d'un conseil prodigué au coordinateur pour le problème de planification présenté. C'est une autre vision d'un paramétrage pour le client. Nous insistons ici sur la flexibilité donnée par BL.Autom à toute évolution de l'organisation et à tout changement de planification de la part du client.

2.3. Prise en compte des incertitudes par coexistence entre l'optimisation et la simulation

Notre approche de résolution présentée dans le chapitre V est déterministe, sans prise en compte des incertitudes, et basée sur les théories de l'optimisation combinatoire. Certaines approches stochastiques évoquées dans le chapitre II abordent les incertitudes opérationnelles, par exemple l'approche reposant sur les chaînes de Markov vise à traiter la durée des interventions stochastiques (Lanzarone et al., 2010) et celle utilisant la NDSP (*Normal Distribution Stochastic Programming*) envisage simultanément la stochastique du temps de routage et de la durée des interventions. En effet, si notre approche permet de produire une solution de planning optimum (la méthode exacte) ou quasi-optimum (la méthode approchée), la lourdeur des solutions basées sur des formulations stochastiques des problèmes d'optimisation nous a semblé rédhibitoire. A ce stade-là, nous envisageons de faire coexister notre approche proposée dans le chapitre V avec des algorithmes de calcul basés sur les théories de la simulation à événements discrets. Cette nouvelle direction permet d'analyser la sensibilité a posteriori par l'usage de la simulation autour de solutions produites par l'optimisation. Nous pouvons intégrer différentes variables entachées d'incertitudes dans cette phase d'analyse, en sus de celles que nous avons déjà repérées dans nos observations de terrain, telle l'incertitude liée à la durée du trajet ou encore à la disponibilité du bénéficiaire. Selon (Barton and Meckesheimer, 2006), les modèles de simulation permettent à l'utilisateur de comprendre les performances du système et d'aider à la prédiction du comportement, afin de soutenir les diagnostics et la conception du système de nature stochastique. Nous avons déjà produit quelques tests pour juger de la pertinence de cette idée.

En effet, la simulation d'événements discrets peut être utilisée en ligne ou hors ligne pour évaluer les solutions générées au sein de l'environnement déterministe. De plus, ce mécanisme pourrait améliorer la validité des résultats de l'optimisation en s'attaquant à la source de la variabilité des résultats du modèle (Parnianifard et al., 2019).

Nous avons pu identifier quelques travaux utilisant cette technique de simulation sur des problèmes similaires, notamment pour la planification en temps réel des opérations de fabrication discrète dynamique (*dynamic discrete manufacturing*). Ainsi, dans (Sivakumar, 2003), les performances des différentes approches de planification sont comparées par des expériences de simulation hors ligne, au sein d'un environnement simulé de fabrication dynamiques soumis à des perturbations telles que les pannes de machines. Ces résultats expérimentaux sont utilisés comme indices de référence pour le mécanisme de planification en temps réel afin de sélectionner les meilleures approches pour une évaluation plus approfondie en fonction des conditions de fabrication réelles. Enfin, la simulation à événements discrets est utilisée en ligne pour évaluer les approches sélectionnées et les plannings correspondants dans le but de déterminer la meilleure solution. Pour faire face aux temps stochastiques par rapport à une activité, dans (Nasiri et al., 2017), une planification en temps réel basée sur la simulation est développée pour optimiser le problème OSS (*Open Shop Scheduling*). Enfin, une approche de multi-réponses optimisées basée sur la simulation à événements discrets pour la planification d'un atelier open-shop non préventif avec des temps de disponibilités stochastiques est présentée afin de minimiser le temps d'attente moyen des tâches.

A cette fin, il apparaît que la simulation à événements discrets est un outil pour la validation du planning déterministe en planification opérationnelle en situation prévisionnelle ou réelle au sein d'un système hautement dynamique et imprévisible (Sivakumar, 2003), comme l'est notre système de PAD.

2.4. Renforcement de notre chaîne de transformation

Dans le chapitre VI, l'implémentation de notre « usine » de la connaissance sur le problème du voyageur de commerce (TSP) prouve la faisabilité de la chaîne d'IDM en pratique. Rappelons que notre but était d'explorer cette voie et de défricher un terrain vierge ; celui de l'emploi de l'IDM dans les problèmes de prise de décision. Nous avons fait de nombreuses hypothèses dans cette prospection que nous avons soulignées dans cette présentation (MTL, bibliothèque de formules, solveur CPLEX). Nous nous attacherons à les lever progressivement pour donner plus de capacité à cet environnement de prise de décision.

Sur la base de notre prototype BL.Autom, nous pouvons envisager d'étendre cette approche. Nous avons un travail de réflexion à mener pour tester la valeur ajoutée tangible de cette fabrique pour un décideur, tout comme pour l'éditeur de logiciel dans sa relation-client. Quels sont les avantages de cette démarche que n'ont pas d'autres formes de développement logiciel ? En partant de ce prototype, la question est donc de démontrer s'il serait possible de traiter une grande variété de problèmes à moindre effort pour un grand nombre de décideurs de structures différentes. Mais d'autres avantages liés à l'interopérabilité avec d'autres services numériques (géolocalisation, édition de cartes, données des moyens de transport etc.) sont aussi acquis dans une telle chaîne pour assurer la maintenance évolutive de solutions.

A termes, dans un premier temps, nous pouvons choisir à renforcer la démarche d'IDM par l'application de la méthode approchée basée sur OptaPlanner. Ensuite, étant donné que nous avons évoqué précédemment une piste de recherche vis-à-vis de la coexistence entre l'optimisation et l'approche de la simulation, cette coexistence des moyens de calcul est un facteur de plus en termes de capacités d'adaptation des connaissances acquises par les deux approches car il faut élaborer et renseigner des fichiers à destination d'un optimiseur ou d'un simulateur de flux qui soient reliés au même problème et qui partagent les mêmes décisions. Ainsi, notre fabrique de la coordination devra alors être étendue pour générer des fichiers de simulation à événements discrets à partir des sorties de l'atelier CIM, ou l'ajout d'une nouvelle couche pour le modèle de simulation juste après le modèle orienté solution (M-S) au niveau PSM, avec une série de nouvelles règles de transformation supportant la transformation M2M du M-S au modèle de simulation.

A l'avenir, cette approche par couplage modélisation/optimisation/simulation basée sur une mise en relation de l'offre et de la demande de services et implantée dans un environnement d'IDM peut présenter un potentiel pour bien d'autres secteurs d'activités, et pas seulement dans le domaine du médico-social. Globalement, tous les secteurs nécessitant une activité de planification de services à la personne se prêteraient à l'exercice, par exemple la planification des cours pour les établissements d'enseignement supérieur serait un cas d'étude pertinent qui pourrait impliquer des retombées indirecte de nos travaux de recherche pour d'autres domaines d'affaires de l'entreprise.

REFERENCES

-
- Aarts, E., Aarts, E.H., Lenstra, J.K., 2003. Local search in combinatorial optimization. Princeton University Press.
- Acretoiaie, V., Störrle, H., Strüber, D., 2018. VMTL: a language for end-user model transformation. *Softw. Syst. Model.* 17, 1139–1167. <https://doi.org/10.1007/s10270-016-0546-9>
- Agrawal, A., Karsai, G., Neema, S., Shi, F., Vizhanyo, A., 2006. The design of a language for model transformations. *Softw. Syst. Model.* 5, 261–288. <https://doi.org/10.1007/s10270-006-0027-7>
- Allweyer, T., 2016. BPMN 2.0: introduction to the standard for business process modeling. BoD–Books on Demand.
- Amokrane, N., 2016. De l'Ingénierie des Besoins à l'Ingénierie des Exigences : Vers une Démarche Méthodologique d'Ingénierie de Systèmes Complexes, de Vérification et de Validation Appliquée à l'Informatisation des PME (phdthesis). Université Montpellier.
- ANAP, 2020. Le numérique en santé - Parcours, coordination et outils numériques - Notions de base illustrées par quelques cas d'usages [WWW Document]. URL <https://ressources.anap.fr/numerique/publication/2792> (accessed 3.26.21).
- Aniszczyk, C., 2006. Learn Eclipse GMF in 15 minutes [WWW Document]. URL <http://www.ibm.com/developerworks/library/os-ecl-gmf/index.html> (accessed 1.15.21).
- Arendt, T., Biermann, E., Jurack, S., Krause, C., Taentzer, G., 2010. Henshin: advanced concepts and tools for in-place EMF model transformations, in: International Conference on Model Driven Engineering Languages and Systems. Springer, pp. 121–135.
- Ashworth, C.M., 1988. Structured systems analysis and design method (SSADM). *Inf. Softw. Technol., The Software Life Cycle* 30, 153–163. [https://doi.org/10.1016/0950-5849\(88\)90062-6](https://doi.org/10.1016/0950-5849(88)90062-6)
- Assouroko, I., Denno, P.O., 2016. A metamodel for optimization problems (No. NIST IR 8096). National Institute of Standards and Technology. <https://doi.org/10.6028/NIST.IR.8096>
- Astorino, A., Gaudioso, M., Miglionico, G., 2018. Lagrangian relaxation for the directional sensor coverage problem with continuous orientation. *Omega* 75, 77–86.
- Avila, O., Goepf, V., Kiefer, F., 2018. Addressing alignment concerns into the design of domain-specific information systems. *J. Manuf. Technol. Manag.*
- Bahadori-Chinibelagh, S., Fathollahi-Fard, A.M., Hajiaghaci-Keshteli, M., 2019. Two Constructive Algorithms to Address a Multi-Depot Home Healthcare Routing Problem. *IETE J. Res.* 0, 1–7. <https://doi.org/10.1080/03772063.2019.1642802>
- Balas, E., 1989. The prize collecting traveling salesman problem. *Networks* 19, 621–636.
- Barták, R., 1999. Constraint programming: In pursuit of the holy grail, in: Proceedings of the Week of Doctoral Students (WDS99). MatFyzPress Prague, pp. 555–564.

- Barton, R.R., Meckesheimer, M., 2006. Metamodel-based simulation optimization. *Handb. Oper. Res. Manag. Sci.* 13, 535–574.
- Barvinok, A., Gimadi, E.K., Serdyukov, A.I., 2007. The maximum TSP, in: *The Traveling Salesman Problem and Its Variations*. Springer, pp. 585–607.
- Bays, C., 1977. A comparison of next-fit, first-fit, and best-fit. *Commun. ACM* 20, 191–192.
- Bazirha, M., Kadrani, A., Benmansour, R., 2019. Daily scheduling and routing of home health care with multiple availability periods of patients, in: *International Conference on Variable Neighborhood Search*. Springer, pp. 178–193.
- Beaumel, C., Papon, S., 2020. Bilan démographique 2019. La fécondité se stabilise en France. Insee Prem.
- Bektas, T., 2006. The multiple traveling salesman problem: an overview of formulations and solution procedures. *omega* 34, 209–219.
- Belorgey, N., 2016. L'hôpital sous pression: enquête sur le "nouveau management public". La découverte.
- Benaben, F., 2012. Conception de Système d'Information de Médiation pour la prise en charge de l'Interopérabilité dans les Collaborations d'Organisations (thesis). Institut National Polytechnique de Toulouse.
- Ben-Ameur, W., 2004. Computing the Initial Temperature of Simulated Annealing. *Comp Opt Appl* 29, 369–385. <https://doi.org/10.1023/B:COAP.0000044187.23143.bd>
- Benzarti, E., Sahin, E., Dallery, Y., 2013. Operations management applied to home care services: Analysis of the districting problem. *Decis. Support Syst.*, 1. Analytics and Modeling for Better HealthCare 2. Decision Making in Healthcare 55, 587–598. <https://doi.org/10.1016/j.dss.2012.10.015>
- Bernus, P., Nemes, L., 1997. The contribution of the generalised enterprise reference architecture to consensus in the area of enterprise integration, in: *Enterprise Engineering and Integration*. Springer, pp. 175–189.
- Bernus, P., Nemes, L., Morris, R., 1994. Possibilities and limitations of reusing enterprise models (new requirements for enterprise engineering tools). *IFAC Proc.* Vol. 27, 11–16.
- Bézivin, J., Büttner, F., Gogolla, M., Jouault, F., Kurtev, I., Lindow, A., 2006. Model Transformations? Transformation Models!, in: Nierstrasz, O., Whittle, J., Harel, D., Reggio, G. (Eds.), *Model Driven Engineering Languages and Systems*. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 440–453. https://doi.org/10.1007/11880240_31
- Bézivin, J., Gerbé, O., 2001. Towards a precise definition of the OMG/MDA framework, in: *Proceedings 16th Annual International Conference on Automated Software Engineering (ASE 2001)*. IEEE, pp. 273–280.
- Blanpain, N., Buisson, G., 2016. Projections de population à l'horizon 2070. Deux fois plus de personnes de 75 ans ou plus qu'en 2013. INSEE.
- Blum, C., Roli, A., 2003. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Comput. Surv.* CSUR 35, 268–308.
- Bock, A., Frank, U., 2016. Multi-perspective enterprise modeling—conceptual foundation and implementation with ADOxx, in: *Domain-Specific Conceptual Modeling*. Springer, pp. 241–267.
- Bondy, J.A., Murty, U.S.R., 1976. *Graph theory with applications* (Vol.290). London: Macmillan
- Börger, E., 2010. The abstract state machines method for high-level system design and analysis, in: *Formal Methods: State of the Art and New Directions*. Springer, pp. 79–116.

- Bortolaso, C., 2012. Méthode de conception assistée par les modèles pour les systèmes interactifs mixtes (phdthesis). Université Paul Sabatier-Toulouse III.
- Braekers, K., Hartl, R.F., Parragh, S.N., Tricoire, F., 2016a. A bi-objective home care scheduling problem: Analyzing the trade-off between costs and client inconvenience. *Eur. J. Oper. Res.* 248, 428–443. <https://doi.org/10.1016/j.ejor.2015.07.028>
- Braekers, K., Ramaekers, K., Van Nieuwenhuysse, I., 2016b. The vehicle routing problem: State of the art classification and review. *Comput. Ind. Eng.* 99, 300–313.
- Braun, R., Schlieter, H., Burwitz, M., Esswein, W., 2014. BPMN4CP: Design and implementation of a BPMN extension for clinical pathways, in: 2014 IEEE International Conference on Bioinformatics and Biomedicine (BIBM). IEEE, pp. 9–16.
- Bredström, D., Rönnqvist, M., 2008. Combined vehicle routing and scheduling with temporal precedence and synchronization constraints. *Eur. J. Oper. Res.* 191, 19–31. <https://doi.org/10.1016/j.ejor.2007.07.033>
- Burke, E.K., Bykov, Y., 2012. The late acceptance hill-climbing heuristic. Univ. Stirling Tech Rep.
- Burnard, K., Bhamra, R., 2011. Organisational resilience: development of a conceptual framework for organisational responses. *Int. J. Prod. Res.* 49, 5581–5599. <https://doi.org/10.1080/00207543.2011.563827>
- Calvez, C., 2019. Parcours (de soins, de santé, de vie) et Coordination. Dossier Documentaire, Technical report, École des hautes études en santé
- Calvez, J.-P., 1990. Spécification et conception des systèmes: une méthodologie. masson.
- Campos, C., Grangel, R., 2018. A domain-specific modelling language for Corporate Social Responsibility (CSR). *Comput. Ind.* 97, 97–110.
- Carello, G., Lanzarone, E., 2014. A cardinality-constrained robust model for the assignment problem in Home Care services. *Eur. J. Oper. Res.* 236, 748–762. <https://doi.org/10.1016/j.ejor.2014.01.009>
- Castillo-Salazar, A., Landa-Silva, D., Qu, R., 2012. A survey of workforce scheduling and routing. Presented at the Proceedings of the 9th International Conference on the Practice and Theory of Automated Timetabling (PATAT 2012), Oslo, Norway, pp. 283–302.
- Chabane, Y., Bortolaso, C., Derras, M., 2019. When Ants Take Care of Humans: ACO for Home-Care Services Planning Optimization., in: *MedInfo*. pp. 546–550.
- Chaieb, M., Jemai, J., Mellouli, K., 2020. A decomposition - construction approach for solving the home health care scheduling problem. *Health Care Manag. Sci.* 23, 264–286. <https://doi.org/10.1007/s10729-019-09479-z>
- Chapron, J., 2006. L'urbanisme organisationnel : méthode et aides à la décision pour piloter l'évolution du système d'information de l'entreprise (phdthesis). Ecole Nationale Supérieure des Mines de Saint-Etienne.
- Chen, D., Vallespir, B., Doumeingts, G., 1997. GRAI integrated methodology and its mapping onto generic enterprise reference architecture and methodology. *Comput. Ind.* 33, 387–394.
- Cheng, E., Rich, J.L., 1998. A home health care routing and scheduling problem.
- Cissé, M., Yalçındağ, S., Kergosien, Y., Şahin, E., Lenté, C., Matta, A., 2017. OR problems related to Home Health Care: A review of relevant routing and scheduling problems. *Oper. Res. Health Care* 13–14, 1–22. <https://doi.org/10.1016/j.orhc.2017.06.001>

- Clarke, G., Wright, J.W., 1964. Scheduling of vehicles from a central depot to a number of delivery points. *Oper. Res.* 12, 568–581.
- CNSA, 2020. Les SPASAD (services polyvalents d'aide et de soins à domicile) [WWW Document]. URL <https://www.pour-les-personnes-agees.gouv.fr/vivre-a-domicile/beneficier-de-soins-a-domicile/les-spasad-services-polyvalents-daide-et-de-soins-domicile> (accessed 3.31.21).
- Cococcioni, M., Fiaschi, L., 2020. The Big-M method with the numerical infinite M. *Optim. Lett.* <https://doi.org/10.1007/s11590-020-01644-6>
- Çolak, R., Işık, A.H., Yiğit, T., 2019. A New Method for Routing in Home Health Care Services. *BRAIN Broad Res. Artif. Intell. Neurosci.* 10, 14–27.
- Combemale, B., 2008. Approche de métamodélisation pour la simulation et la vérification de modèle--Application à l'ingénierie des procédés. Institut National Polytechnique de Toulouse-INPT.
- CPLEX, I.I., 2011. Cplex users manual. Version.
- Cuadrado, J.S., Guerra, E., de Lara, J., 2014. A Component Model for Model Transformations. *IEEE Trans. Softw. Eng.* 40, 1042–1060. <https://doi.org/10.1109/TSE.2014.2339852>
- Czarnecki, K., Helsen, S., 2003. Classification of model transformation approaches, in: *Proceedings of the 2nd OOPSLA Workshop on Generative Techniques in the Context of the Model Driven Architecture*. USA, pp. 1–17.
- Danielsson, P.-E., 1980. Euclidean distance mapping. *Comput. Graph. Image Process.* 14, 227–248.
- Dantzig, G., Fulkerson, R., Johnson, S., 1954. Solution of a large-scale traveling-salesman problem. *J. Oper. Res. Soc. Am.* 2, 393–410.
- Dantzig, G.B., Ramser, J.H., 1959. The Truck Dispatching Problem. *Manag. Sci.* 6, 80–91.
- Dantzig, G.B., Wolfe, P., 1960. Decomposition principle for linear programs. *Oper. Res.* 8, 101–111.
- Davis, J., 2003. GME: the generic modeling environment, in: *Companion of the 18th Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications*. pp. 82–83.
- De Lara, J., Vangheluwe, H., 2002. AToM 3: A Tool for Multi-formalism and Meta-modelling, in: *International Conference on Fundamental Approaches to Software Engineering*. Springer, pp. 174–188.
- Decerle, J., Grunder, O., Hajjam El Hassani, A., Barakat, O., 2019. A hybrid memetic-ant colony optimization algorithm for the home health care problem with time window, synchronization and working time balancing. *Swarm Evol. Comput.* 46, 171–183. <https://doi.org/10.1016/j.swevo.2019.02.009>
- Decerle, J., Grunder, O., Hajjam El Hassani, A., Barakat, O., 2018. A memetic algorithm for a home health care routing and scheduling problem. *Oper. Res. Health Care* 16, 59–71. <https://doi.org/10.1016/j.orhc.2018.01.004>
- Dekhici, L., Redjem, R., Belkadi, K., El Mhamedi, A., 2019. Discretization of the firefly algorithm for home care. *Can. J. Electr. Comput. Eng.* 42, 20–26.
- Demirbilek, M., Branke, J., Strauss, A., 2019. Dynamically accepting and scheduling patients for home healthcare. *Health Care Manag. Sci.* 22, 140–155. <https://doi.org/10.1007/s10729-017-9428-0>
- Desfray, P., Raymond, G., 2019. TOGAF, Archimate, UML et BPMN-3e éd. Dunod.

- Di Gaspero, L., Urli, T., 2014. A CP/LNS Approach for Multi-day Homecare Scheduling Problems, in: Blesa, M.J., Blum, C., Voß, S. (Eds.), *Hybrid Metaheuristics*, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 1–15. https://doi.org/10.1007/978-3-319-07644-7_1
- Di Mascolo, M., Martinez, C., Espinouse, M.-L., 2021. Routing and scheduling in Home Health Care: A Literature Survey and Bibliometric Analysis. *Comput. Ind. Eng.* 107255.
- Diana, M., Dessouky, M.M., 2004. A new regret insertion heuristic for solving large-scale dial-a-ride problems with time windows. *Transp. Res. Part B Methodol.* 38, 539–557.
- Doumeingts, G., 1990. Méthodes pour concevoir et spécifier les systèmes de production, in: CIM. pp. 89–103.
- Du, G., Liang, X., Sun, C., 2017. Scheduling Optimization of Home Health Care Service Considering Patients' Priorities and Time Windows. *Sustainability* 9, 253. <https://doi.org/10.3390/su9020253>
- Du, G., Zheng, L., Ouyang, X., 2019. Real-time scheduling optimization considering the unexpected events in home health care. *J. Comb. Optim.* 37, 196–220. <https://doi.org/10.1007/s10878-017-0220-3>
- Dueck, G., 1993. New optimization heuristics: The great deluge algorithm and the record-to-record travel. *J. Comput. Phys.* 104, 86–92.
- Duque, P.M., Castro, M., Sörensen, K., Goos, P., 2015. Home care service planning. The case of Landelijke Thuiszorg. *Eur. J. Oper. Res.* 243, 292–301.
- Eksioglu, B., Vural, A.V., Reisman, A., 2009. The vehicle routing problem: A taxonomic review. *Comput. Ind. Eng.* 57, 1472–1483.
- En-nahli, L., Allaoui, H., Nouaouri, I., 2015. A Multi-objective Modelling to Human Resource Assignment and Routing Problem for Home Health Care Services. *IFAC-Pap., 15th IFAC Symposium on Information Control Problems in Manufacturing* 48, 698–703. <https://doi.org/10.1016/j.ifacol.2015.06.164>
- Enríquez, J., Domínguez-Mayo, F., Escalona, M., García, J.G., Lee, V., Goto, M., 2015. Entity Identity Reconciliation based Big Data Federation-A MDE approach. *Int. Conf. Inf. Syst. Dev. ISD*.
- Erdem, M., Koç, Ç., 2019. Analysis of electric vehicles in home health care routing problem. *J. Clean. Prod.* 234, 1471–1483. <https://doi.org/10.1016/j.jclepro.2019.06.236>
- Fard, A.M.F., Hajiaghayi-Keshteli, M., Paydar, M.M., 2018. A Location-Allocation-Routing Model for Home Health Care Supply Chain Problem. *World Acad. Sci. Eng. Technol. Int. J. Ind. Manuf. Eng.* 5.
- Fatemi Ghomi, S.M.T., Arabzadeh, E., Karimi, B., 2018. A Multi Period Routing and Scheduling optimization Model for Home health Care Problem. *Ind. Manag. Stud.* 16, 1–30.
- Fathollahi-Fard, A.M., Govindan, K., Hajiaghayi-Keshteli, M., Ahmadi, A., 2019. A green home health care supply chain: New modified simulated annealing algorithms. *J. Clean. Prod.* 240, 118200. <https://doi.org/10.1016/j.jclepro.2019.118200>
- Fathollahi-Fard, A.M., Hajiaghayi-Keshteli, M., Mirjalili, S., 2020. A set of efficient heuristics for a home healthcare problem. *Neural Comput. Appl.* 32, 6185–6205. <https://doi.org/10.1007/s00521-019-04126-8>
- Fathollahi-Fard, A.M., Hajiaghayi-Keshteli, M., Tavakkoli-Moghaddam, R., 2018. The social engineering optimizer (SEO). *Eng. Appl. Artif. Intell.* 72, 267–293.

- Favre, J.-M., 2005. Foundations of model (Driven)(Reverse) engineering: models--Episode I: stories of the fidus papyrus and of the solarus, in: Dagstuhl Seminar Proceedings. Schloss Dagstuhl-Leibniz-Zentrum für Informatik.
- Feng, Q., Wang, J., 2018. Vehicle Routing Problem with Time Windows in Aged Service: A Delivery Service Model for the Elderly, in: 2018 15th International Conference on Service Systems and Service Management (ICSSSM). IEEE, pp. 1–4.
- Feo, T.A., Venkatraman, K., Bard, J.F., 1991. A GRASPTM for a difficult single machine scheduling problem. *Comput. Oper. Res.* 18, 635–643.
- Fernandez, I.M., Jurado, I.G., Freire, S.L., Rodríguez, L.C., Bouzas, J.C., 2018. A Task Planning Problem in a Home Care Business. *Proceedings* 2, 1186. <https://doi.org/10.3390/proceedings2181186>
- Fikar, C., Hirsch, P., 2017. Home health care routing and scheduling: A review. *Comput. Oper. Res.* 77, 86–95. <https://doi.org/10.1016/j.cor.2016.07.019>
- Fikar, C., Hirsch, P., 2015. A matheuristic for routing real-world home service transport systems facilitating walking. *J. Clean. Prod., Decision-support models and tools for helping to make real progress to more sustainable societies* 105, 300–310. <https://doi.org/10.1016/j.jclepro.2014.07.013>
- Fill, H.-G., 2017. SeMFIS: A flexible engineering platform for semantic annotations of conceptual models. *Semantic Web* 8, 747–763. <https://doi.org/10.3233/SW-160235>
- Fill, H.-G., Karagiannis, D., 2013. On the Conceptualisation of Modelling Methods Using the ADOxx Meta Modelling Platform. *Enterp. Model. Inf. Syst. Archit. EMISAJ* 8, 4–25. <https://doi.org/10.18417/emisa.8.1.1>
- Fischer, K., 2005. Advanced technologies for interoperability of heterogeneous enterprise networks and their applications. *Eur. Coord. Action Agent-Based Comput.* 34.
- Fisher, M.L., 1981. The Lagrangian relaxation method for solving integer programming problems. *Manag. Sci.* 27, 1–18.
- Flatberg, T., 2009. A short OPL tutorial. January.
- Floudas, C.A., Pardalos, P.M., 1990. A Collection of Test Problems for Constrained Global Optimization Problems, *Lecture Notes in Computer Science*. Springer.
- Fortz, B., 2012. Recherche opérationnelle et applications. 53. <http://homepages.ulb.ac.be/~bfortz/ro.pdf>
- Frank, U., 2013. Domain-Specific Modeling Languages: Requirements Analysis and Design Guidelines, in: Reinhartz-Berger, I., Sturm, A., Clark, T., Cohen, S., Bettin, J. (Eds.), *Domain Engineering*. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 133–157. https://doi.org/10.1007/978-3-642-36654-3_6
- Frank, U., 2010. Outline of a method for designing domain-specific modelling languages (Research Report No. 42). ICB-Research Report.
- Frifita, S., Masmoudi, M., Euch, J., 2017. General variable neighborhood search for home healthcare routing and scheduling problem with time windows and synchronized visits. *Electron. Notes Discrete Math.* 58, 63–70. <https://doi.org/10.1016/j.endm.2017.03.009>
- Gao, J., Sun, L., Gen, M., 2008. A hybrid genetic and variable neighborhood descent algorithm for flexible job shop scheduling problems. *Comput. Oper. Res.* 35, 2892–2907.
- Gayraud, F., Deroussi, L., Grangeon, N., Norre, S., 2013. A new mathematical formulation for the home health care problem. *Procedia Technol.* 9, 1041–1047.

- Gendreau, M., Laporte, G., Semet, F., 1997. The covering tour problem. *Oper. Res.* 45, 568–576.
- Glover, F., Laguna, M., 1998. Tabu search, in: *Handbook of Combinatorial Optimization*. Springer, pp. 2093–2229.
- Goldfeld, S.M., Quandt, R.E., Trotter, H.F., 1966. Maximization by quadratic hill-climbing. *Econom. J. Econom. Soc.* 541–551.
- Golshahi-Roudbaneh, A., Hajiaghahi-Keshteli, M., Paydar, M.M., 2017. Developing a lower bound and strong heuristics for a truck scheduling problem in a cross-docking center. *Knowl.-Based Syst.* 129, 17–38.
- Gómez, F.J., Vanfretti, L., Olsen, S.H., 2018. CIM-Compliant Power System Dynamic Model-to-Model Transformation and Modelica Simulation. *IEEE Trans. Ind. Inform.* 14, 3989–3996. <https://doi.org/10.1109/TII.2017.2785439>
- Goodarzian, F., Abraham, A., Fathollahi-Fard, A.M., 2020. A biobjective home health care logistics considering the working time and route balancing: a self-adaptive social engineering optimizer. *J. Comput. Des. Eng.*
- Gordeeva, T., Tiukhova, E., Snoeck, M., 2021. Multi-stakeholder modeling with 3D and VR. Press.
- Gourgand, M., 2008. La modélisation, la simulation et l'optimisation des flux dans les systèmes hospitaliers. *Bull. Société Fr. Rech. Opérationnelle D'Aide À Décision* 21, 8–12.
- Greco, R., Di Nardo, A., Santonastaso, G., 2012. Resilience and entropy as indices of robustness of water distribution networks. *J. Hydroinformatics* 14, 761–771.
- Greenfield, J., Short, K., 2003. Software factories: assembling applications with patterns, models, frameworks and tools, in: *Companion of the 18th Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications*. pp. 16–27.
- Grenouilleau, F., Lahrichi, N., Rousseau, L.-M., 2020. New decomposition methods for home care scheduling with predefined visits. *Comput. Oper. Res.* 115, 104855.
- Gutiérrez Naranjo, M.Á., Martínez del Amor, M.Á., Pérez Hurtado de Mendoza, I., Pérez Jiménez, M. de J., 2009. Solving the N-queens puzzle with P systems. *Proc. Seventh Brainstorming Week Membr. Comput. Vol 199-210 Sevilla ETS Ing. Informática 2-6 Febrero 2009*.
- Gutin, G., Punnen, A.P., 2006. *The traveling salesman problem and its variations*. Springer Science & Business Media.
- Guttmann-Beck, N., Hassin, R., Khuller, S., Raghavachari, B., 2000. Approximation algorithms with bounded performance guarantees for the clustered traveling salesman problem. *Algorithmica* 28, 422–437.
- Haddadene, S.A., Labadie, N., Prodhon, C., 2016. NSGAII enhanced with a local search for the vehicle routing problem with time windows and synchronization constraints. *IFAC-Pap.* 49, 1198–1203.
- Haimes, Y., 1971. On a bicriterion formulation of the problems of integrated system identification and system optimization. *IEEE Trans. Syst. Man Cybern.* 1, 296–297.
- Hall, R., 2013. Patient flow. *AMC* 10, 12.
- Hassin, R., Keinan, A., 2008. Greedy heuristics with regret, with application to the cheapest insertion algorithm for the TSP. *Oper. Res. Lett.* 36, 243–246.

- He, C., Shi, Y., 2019. Workforce Scheduling and Routing for Home Health Care Services, in: Proceedings of the 3rd International Conference on Computer Science and Application Engineering. pp. 1–5.
- Heching, A., Hooker, J.N., 2016. Scheduling home hospice care with logic-based Benders decomposition, in: International Conference on AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems. Springer, pp. 187–197.
- Hertz, A., Lahrichi, N., 2009. A patient assignment algorithm for home care services. *J. Oper. Res. Soc.* 60, 481–495.
- Heß, M., Kaczmarek, M., Frank, U., Podleska, L., Täger, G., 2015. A domain-specific modelling language for clinical pathways in the realm of multi-perspective hospital modelling.
- Hiermann, G., Prandtstetter, M., Rendl, A., Puchinger, J., Raidl, G.R., 2015. Metaheuristics for solving a multimodal home-healthcare scheduling problem. *Cent. Eur. J. Oper. Res.* 23, 89–113. <https://doi.org/10.1007/s10100-013-0305-8>
- James, W., Athansios, Z., Nicholas, M., Louis, R., Dimitios, K., Richard, P., Fiona, P., 2013. What do metamodels really look like. *Front. Comput. Sci.*
- Jensen, T.R., Toft, B., 2011. Graph coloring problems. John Wiley & Sons.
- Jouault, F., Allilaire, F., Bézivin, J., Kurtev, I., 2008. ATL: A model transformation tool. *Sci. Comput. Program.* 72, 31–39. <https://doi.org/10.1016/j.scico.2007.08.002>
- Jouault, F., Allilaire, F., Bézivin, J., Kurtev, I., Valduriez, P., 2006. ATL: a QVT-like transformation language, in: Companion to the 21st ACM SIGPLAN Symposium on Object-Oriented Programming Systems, Languages, and Applications. pp. 719–720.
- Jünger, M., Reinelt, G., Rinaldi, G., 1995. The traveling salesman problem. *Handb. Oper. Res. Manag. Sci.* 7, 225–330.
- Kabadi, S.N., Punnen, A.P., 2007. The bottleneck TSP, in: The Traveling Salesman Problem and Its Variations. Springer, pp. 697–735.
- Kahani, N., Bagherzadeh, M., Cordy, J.R., Dingel, J., Varró, D., 2019. Survey and classification of model transformation tools. *Softw. Syst. Model.* 18, 2361–2397.
- Kahani, N., Cordy, J.R., 2015. Comparison and evaluation of model transformation tools. Queens Univ. Kingston Tech Rep.
- Kang, B.G., Choi, S.H., Kwon, S.J., Lee, J.H., Kim, T.G., 2018. Simulation-Based Optimization on the System-of-Systems Model via Model Transformation and Genetic Algorithm: A Case Study of Network-Centric Warfare [WWW Document]. Complexity. <https://doi.org/10.1155/2018/4521672>
- Karaboga, D., 2005. An idea based on honey bee swarm for numerical optimization. Technical report-tr06, Erciyes university, engineering faculty, computer
- Karagiannis, D., 2015. Agile modeling method engineering, in: Proceedings of the 19th Panhellenic Conference on Informatics - PCI '15. Presented at the the 19th Panhellenic Conference, ACM Press, Athens, Greece, pp. 5–10. <https://doi.org/10.1145/2801948.2802040>
- Karagiannis, D., Buchmann, R.A., Burzynski, P., Reimer, U., Walch, M., 2016. Fundamental conceptual modeling languages in OMiLAB, in: Domain-Specific Conceptual Modeling. Springer, pp. 3–30.
- Karagiannis, D., Kühn, H., 2002. Metamodelling platforms, in: EC-Web. p. 182.
- Katayama, N., Chen, M., Kubo, M., 2009. A capacity scaling heuristic for the multicommodity capacitated network design problem. *J. Comput. Appl. Math.* 232, 90–101.

- Ke, R., Fang, H., 2004. Rotating Bidding: A Mechanism to Allocate Common Pool Resource, Comparing with Bargaining Solution. CHINA Econ. Q.-BEIJING- 3, 331–356.
- Khambati, A., Grundy, J., Warren, J., Hosking, J., 2008. Model-Driven Development of Mobile Personal Health Care Applications, in: Proceedings of the 2008 23rd IEEE/ACM International Conference on Automated Software Engineering, ASE '08. IEEE Computer Society, Washington, DC, USA, pp. 467–470. <https://doi.org/10.1109/ASE.2008.75>
- Khecharem, A., 2016. MuVArch: une approche de méta-modélisation pour la représentation multi-vues des architectures hétérogènes embarqués.
- Khodabandeh, P., Kayvanfar, V., Rafiee, M., Werner, F., 2020. A bi-objective home health care routing and scheduling model with considering nurse downgrading costs.
- Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P., 1983. Optimization by Simulated Annealing. Science 220, 671–680. <https://doi.org/10.1126/science.220.4598.671>
- Kleppe, A.G., Warmer, J., Warmer, J.B., Bast, W., 2003. MDA explained: the model driven architecture: practice and promise. Addison-Wesley Professional.
- Kolovos, D.S., Rose, L.M., Abid, S.B., Paige, R.F., Polack, F.A., Botterweck, G., 2010. Taming EMF and GMF using model transformation, in: International Conference on Model Driven Engineering Languages and Systems. Springer, pp. 211–225.
- Kolovos, D.S., Rose, L.M., Paige, R.F., Polack, F.A., 2009. Raising the level of abstraction in the development of GMF-based graphical model editors, in: 2009 ICSE Workshop on Modeling in Software Engineering. IEEE, pp. 13–19.
- Kosecka-Zurek, S., 2019. Application of it tools in optimization of logistics problems. Tech. Trans. 12, 177–186.
- Kubiak, Y., 2012. Personnes âgées dépendantes: le maintien à domicile, solution privilégiée mais exigeante.
- Kühne, T., 2006. Matters of (meta-) modeling. Softw. Syst. Model. 5, 369–385.
- Kurtev, I., 2007. State of the art of QVT: A model transformation language standard, in: International Symposium on Applications of Graph Transformations with Industrial Relevance. Springer, pp. 377–393.
- Laffont, J.-Y., 2010. De l'agence régionale d'hospitalisation à l'agence régionale de santé. Gerontol. Soc. 33 / n° 132, 223–228.
- Lamine, E., 2019. Ingénierie dirigée par les modèles de la transformation numérique du système de santé Application aux parcours d'utilisateurs (Habilitation à Diriger les Recherches).
- Lamine, E., 2001. Définition d'un modèle de propriété et proposition d'un langage de spécification associé: LUSP. Montpellier 2.
- Lamine, E., Bastide, R., Bouet, M., Gaborit, P., Gourc, D., Marmier, F., Pingaud, H., Schneider, M., Toumani, F., 2019. Plas'O'Soins: An Interactive ICT Platform to Support Care Planning and Coordination within Home-Based Care. IRBM 40, 25–37. <https://doi.org/10.1016/j.irbm.2018.10.015>
- Lamine, E., Thabet, R., Sienu, A., Bork, D., Fontanili, F., Pingaud, H., 2020. BPRIM: An integrated framework for business process management and risk management. Comput. Ind. 117, 103199.
- Lamine, E., Zefouni, S., Bastide, R., Pingaud, H., 2010. A system architecture supporting the agile coordination of homecare services, in: Working Conference on Virtual Enterprises. Springer, pp. 227–234.

- Lanzarone, E., Matta, A., 2014. Robust nurse-to-patient assignment in home care services to minimize overtimes under continuity of care. *Oper. Res. Health Care, Special Issue of the 2012 conference of the EURO working group Operational Research Applied To Health Services (ORAHs)* 3, 48–58. <https://doi.org/10.1016/j.orhc.2014.01.003>
- Lanzarone, E., Matta, A., Scaccabarozzi, G., 2010. A patient stochastic model to support human resource planning in home care. *Prod. Plan. Control* 21, 3–25. <https://doi.org/10.1080/09537280903232362>
- Lawler, E.L., Wood, D.E., 1966. Branch-And-Bound Methods: A Survey. *Oper. Res.* 14, 699–719.
- Le Moigne, J.L., 1990. *La modélisation des systèmes complexes*, 1990. Paris Dunod.
- Lemrabet, Y., Bigand, M., Clin, D., Benkeltoum, N., Bourey, J.-P., 2010. Model Driven Interoperability in practice: preliminary evidences and issues from an industrial project, in: *Proceedings of the First International Workshop on Model-Driven Interoperability*. pp. 3–9.
- Lenstra, J.K., Kan, A.R., 1981. Complexity of vehicle routing and scheduling problems. *Networks* 11, 221–227.
- Levendovszky, T., Lengyel, L., Mezei, G., Charaf, H., 2005. A systematic approach to metamodeling environments and model transformation systems in VMTS. *Electron. Notes Theor. Comput. Sci.* 127, 65–75.
- Lieder, A., Moeke, D., Koole, G., Stolletz, R., 2015. Task scheduling in long-term care facilities: A client-centered approach. *Oper. Res. Health Care* 6, 11–17. <https://doi.org/10.1016/j.orhc.2015.06.001>
- Lin, C.-C., Hung, L.-P., Liu, W.-Y., Tsai, M.-C., 2018. Jointly rostering, routing, and rostering for home health care services: A harmony search approach with genetic, saturation, inheritance, and immigrant schemes. *Comput. Ind. Eng.* 115, 151–166. <https://doi.org/10.1016/j.cie.2017.11.004>
- Lin, M., Chin, K.S., Wang, X., Tsui, K.L., 2016. The therapist assignment problem in home healthcare structures. *Expert Syst. Appl.* 62, 44–62.
- Lin, S., 1965. Computer solutions of the traveling salesman problem. *Bell Syst. Tech. J.* 44, 2245–2269.
- Liu, M., Yang, D., Su, Q., Xu, L., 2018. Bi-objective approaches for home healthcare medical team planning and scheduling problem. *Comput. Appl. Math.* 37, 4443–4474. <https://doi.org/10.1007/s40314-018-0584-8>
- Liu, R., Xie, X., Garaix, T., 2014. Hybridization of tabu search with feasible and infeasible local searches for periodic home health care logistics. *Omega* 47, 17–32. <https://doi.org/10.1016/j.omega.2014.03.003>
- Lozano Murciego, A., Villarrubia Gonzalez, G., López Barriuso, A., Hernández de la Iglesia, D., Revuelta Herrero, J., 2015. Multi agent gathering waste system.
- Macik, B.M., 2016. Case management task assignment using optaplanner. Master's thesis, Masaryk University Faculty of Informatics.
- Malapert, A., 2006. *Optimisation de tournées de véhicules pour l'exploitation de Réseau Telecom*. Rapp. Stage Diplôme Master Intell. Artif. Décision Univ. Paris 6.
- Malone, T.W., Crowston, K., 1994. The Interdisciplinary Study of Coordination. *ACM Comput Surv* 26, 87–119. <https://doi.org/10.1145/174666.174668>
- Manavizadeh, N., Farrokhi-Asl, H., Beiraghdar, P., 2020. Using a metaheuristic algorithm for solving a home health care routing and scheduling problem. *J. Proj. Manag.* 5, 27–40.

- Mankowska, D.S., Meisel, F., Bierwirth, C., 2014. The home health care routing and scheduling problem with interdependent services. *Health Care Manag. Sci.* 17, 15–30. <https://doi.org/10.1007/s10729-013-9243-1>
- Marca, D.A., McGowan, C.L., 1987. SADT: structured analysis and design technique. McGraw-Hill, Inc.
- Marcon, E., Chaabane, S., Sallez, Y., Bonte, T., 2016. Caregivers routing problem in home health care: literature review, in: *International Workshop on Service Orientation in Holonic and Multi-Agent Manufacturing*. Springer, pp. 319–326.
- Marcon, E., Chaabane, S., Sallez, Y., Bonte, T., Trentesaux, D., 2017. A multi-agent system based on reactive decision rules for solving the caregiver routing problem in home health care. *Simul. Model. Pract. Theory* 74, 134–151.
- Marmion, M.-É., 2011. Recherche locale et optimisation combinatoire: de l'analyse structurelle d'un problème à la conception d'algorithmes efficaces. Université des Sciences et Technologie de Lille-Lille I.
- Martinez, C., 2020. Considération de la dimension humaine dans l'optimisation de tournées de soins et services à domicile soumises à des perturbations (phdthesis).
- Martinez, C., Espinouse, M., Mascolo, M.D., 2018. Continuity of care in home services: a client-centered heuristic for the home health care routing and scheduling problem, in: *2018 5th International Conference on Control, Decision and Information Technologies (CoDIT)*. Presented at the 2018 5th International Conference on Control, Decision and Information Technologies (CoDIT), pp. 1045–1050. <https://doi.org/10.1109/CoDIT.2018.8394968>
- Martinez, C., Espinouse, M.-L., Di Mascolo, M., 2019. Re-planning In Home Healthcare: A Decomposition Approach To Minimize Idle Time For Workers While Ensuring Continuity Of Care. *IFAC-Pap.* 52, 654–659.
- Mascolo, M.D., Espinouse, M.L., Haddadene, S.R.A., 2018. Taking patients' wishes into account for daily planning in the Home Health Care contex. *IFAC-Pap.*, 16th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2018 51, 1010–1015. <https://doi.org/10.1016/j.ifacol.2018.08.475>
- Matta, A., Chahed, S., Sahin, E., Dallery, Y., 2014. Modelling home care organisations from an operations management perspective. *Flex. Serv. Manuf. J.* 26, 295–319. <https://doi.org/10.1007/s10696-012-9157-0>
- McDonald, K.M., Sundaram, V., Bravata, D.M., Lewis, R., Lin, N., Kraft, S.A., McKinnon, M., Paguntalan, H., Owens, D.K., 2007. Closing the Quality Gap: A Critical Analysis of Quality Improvement Strategies (Vol. 7: Care Coordination). Agency for Healthcare Research and Quality (US).
- Méndez-Fernández, I., Lorenzo-Freire, S., García-Jurado, I., Costa, J., Carpenle, L., 2020. A heuristic approach to the task planning problem in a home care business. *Health Care Manag. Sci.* 23, 556–570. <https://doi.org/10.1007/s10729-020-09509-1>
- Merlo, C., Vicien, G., Ducq, Y., 2014. Interoperability modelling methodology for product design organisations. *Int. J. Prod. Res.* 52, 4379–4395.
- Mestria, M., 2018. New hybrid heuristic algorithm for the clustered traveling salesman problem. *Comput. Ind. Eng.* 116, 1–12.
- Mhamedi, A.E., Vernadat, F.B., 1999. ACNOS: A functional and socio-cognitive modelling approach to analyse industrial systems. *IFAC Proc. Vol.* 32, 145–150. [https://doi.org/10.1016/S1474-6670\(17\)56027-9](https://doi.org/10.1016/S1474-6670(17)56027-9)

- Michael, J., Mayr, H.C., 2015. Creating a Domain Specific Modelling Method for Ambient Assistance, in: 2015 Fifteenth International Conference on Advances in ICT for Emerging Regions (ICTer). Presented at the 2015 Fifteenth International Conference on Advances in ICT for Emerging Regions (ICTer), IEEE, Colombo, Sri Lanka, pp. 119–124. <https://doi.org/10.1109/ICTER.2015.7377676>
- Milburn, A.B., Spicer, J., 2013. Multi-objective home health nurse routing with remote monitoring devices. *Int. J. Plan. Sched.* 1, 242–263.
- Misir, M., Verbeeck, K., De Causmaecker, P., Berghe, G.V., 2010. Hyper-heuristics with a dynamic heuristic set for the home care scheduling problem, in: IEEE Congress on Evolutionary Computation. IEEE, pp. 1–8.
- Mladenović, N., Hansen, P., 1997. Variable neighborhood search. *Comput. Oper. Res.* 24, 1097–1100.
- Mu, W., Benaben, F., Boissel-Dallier, N., Pingaud, H., 2017. Collaborative Knowledge Framework for Mediation Information System Engineering. *Sci. Program.* 2017, 1–18. <https://doi.org/10.1155/2017/9026387>
- Muller, P.-A., Gaertner, N., 2000. Modélisation objet avec UML. Eyrolles Paris.
- Nasir, J.A., Dang, C., 2018. Solving a More Flexible Home Health Care Scheduling and Routing Problem with Joint Patient and Nursing Staff Selection. *Sustainability* 10, 148. <https://doi.org/10.3390/su10010148>
- Nasiri, M.M., Yazdanparast, R., Jolai, F., 2017. A simulation optimisation approach for real-time scheduling in an open shop environment using a composite dispatching rule. *Int. J. Comput. Integr. Manuf.* 30, 1239–1252. <https://doi.org/10.1080/0951192X.2017.1307452>
- Nastoy, B., 2016. Contribution à une méthode outillée pour la conception de langages de modélisation métier interopérables, analysables et prouvables pour l'Ingénierie Système basée sur des Modèles. Montpellier.
- Noon, C.E., Bean, J.C., 1993. An efficient transformation of the generalized traveling salesman problem. *INFOR Inf. Syst. Oper. Res.* 31, 39–44.
- OMG, 2015. Meta Object Facility (MOF) Specification 2.5.
- OMG, 2014. MDA Guide revision 2.0.
- OptaPlanner, 2021. Documentation [WWW Document]. OptaPlanner. URL <https://www.optaplanner.org/learn/documentation.html> (accessed 3.9.21).
- Paige, R.F., Kolovos, D.S., Polack, F.A., 2014. A tutorial on metamodelling for grammar researchers. *Sci. Comput. Program.* 96, 396–416.
- Parnianifard, A., Azfanizam, A.S., Ariffin, M.K.A., Ismail, M.I.S., Ale Ebrahim, N., 2019. Recent developments in metamodel based robust black-box simulation optimization: An overview. *Decis. Sci. Lett.* 17–44. <https://doi.org/10.5267/j.dsl.2018.5.004>
- Parragh, S.N., Doerner, K.F., 2018. Solving routing problems with pairwise synchronization constraints. *Cent. Eur. J. Oper. Res.* 1–22. <https://doi.org/10.1007/s10100-018-0520-4>
- Peterson, J.L., 1977. Petri nets. *ACM Comput. Surv. CSUR* 9, 223–252.
- Petite, W., 2006. Les effets de l'Allocation personnalisée d'autonomie sur l'aide dispensée aux personnes âgées 12.
- Pingaud, H., 2005. Modélisation d'entreprise. Cours. Toulouse : Université Paul Sabatier.

- Poole, J.D., 2001. Model-driven architecture: Vision, standards and emerging technologies, in: Workshop on Metamodeling and Adaptive Object Models, ECOOP. Citeseer.
- Presley, A., Liles, D.H., 1995. The use of IDEF0 for the design and specification of methodologies, in: Proceedings of the 4th Industrial Engineering Research Conference.
- PSAD, 2015. Rapport : Quelle santé à domicile pour demain ? Enjeux et perspectives. Portail Natl. Silver Econ. URL <https://www.silvereco.fr/rapport-quelle-sante-a-domicile-pour-demain-enjeux-et-perspectives/3138573> (accessed 3.31.21).
- Quintanilla, S., Ballestín, F., Pérez, Á., 2020. Mathematical models to improve the current practice in a Home Healthcare Unit. *Spectr.* 42, 43–74. <https://doi.org/10.1007/s00291-019-00565-w>
- Redjem, R., Kharraja, S., Xie, X., Marcon, E., 2011. Coordinated multi-criteria scheduling of caregivers in home health care services, in: 2011 IEEE International Conference on Automation Science and Engineering. IEEE, pp. 519–524.
- Redjem, R., Marcon, E., 2016. Operations management in the home care services: a heuristic for the caregivers' routing problem. *Flex. Serv. Manuf. J.* 28, 280–303. <https://doi.org/10.1007/s10696-015-9220-8>
- Rejeb, O., 2013. Proposition d'un cadre méthodologique pour le management de la continuité d'activité: application à la prise en charge à domicile. Ecole des Mines d'Albi-Carmaux.
- Rest, K.-D., Hirsch, P., 2016. Daily scheduling of home health care services using time-dependent public transport. *Flex. Serv. Manuf. J.* 28, 495–525. <https://doi.org/10.1007/s10696-015-9227-1>
- Rhazali, Y., Hadi, Y., Mouloudi, A., 2016. Model Transformation with ATL into MDA from CIM to PIM Structured through MVC. *Procedia Comput. Sci., The 7th International Conference on Ambient Systems, Networks and Technologies (ANT 2016) / The 6th International Conference on Sustainable Energy Information Technology (SEIT-2016) / Affiliated Workshops* 83, 1096–1101. <https://doi.org/10.1016/j.procs.2016.04.229>
- Rios de Souza, V., Martins, C.B., 2020. UMA ANÁLISE DO FRAMEWORK OPTAPLANNER APLICADO AO PROBLEMA DE EMPACOTAMENTO UNIDIMENSIONAL. *Rev. Sist. E Comput.-RSC* 9.
- Rivera, J.E., 2010. On the semantics of real-time domain specific modeling languages (phdthesis).
- Rossi, F., van Beek, P., Walsh, T. (Eds.), 2006. *Handbook of Constraint Programming, Foundations of Artificial Intelligence*. Elsevier.
- Rossi, G., Urbietta, M., Distante, D., Rivero, J.M., Firmenich, S., 2016. 25 Years of Model-Driven Web Engineering. What we achieved, What is missing. *CLEI Electron. J.* 19, 5–57.
- Rothenberg, J., Widman, L.E., Loparo, K.A., Nielsen, N.R., 1989. The nature of modeling. *Artif. Intell. Simul. Model.*
- Russell, S., Norvig, P., 2002. *Artificial intelligence: a modern approach*.
- Sarmady, S., 2012. An Investigation on Tabu Search Parameters. *Sch. Comput. Sci. Univ. Sains Malays.* 11800.
- Scheer, A.-W., 2000. *ARIS—business process modeling*. Springer Science & Business Media.
- Schmidt, D.C., 2006. Model-driven engineering. *Comput.-IEEE Comput. Soc.-* 39, 25.
- Schon, D.A., 1984. *The reflective practitioner: How professionals think in action*. Basic books.
- Seidewitz, E., 2003. What models mean. *Softw. IEEE* 20, 26–32. <https://doi.org/10.1109/MS.2003.1231147>

- Semeria, C., 2001. Supporting differentiated service classes: queue scheduling disciplines. *Juniper Netw.* 11–14.
- Sendall, S., Kozaczynski, W., 2003. Model transformation: The heart and soul of model-driven software development. *IEEE Softw.* 20, 42–45.
- SGMCAS, 2016. Parcours de soins, parcours de santé, parcours de vie pour une prise en charge adaptée des patients et usagers – lexique des parcours de A à Z. *Rapp Tech Paris Fr. Ministère Aff. Soc. Santé Droits Femmes.*
- Shan-shan, L., Min, G.E., Ping, J., Min-jie, Z., Hong, L., Jiao-ling, H., Shuai, F., De-yu, Z., Yi-min, Z., 2018. Effects of contractual services from family doctors on the healthcare-seeking behavior among community residents. *Chin. Gen. Prat.* 21, 407–410.
- Shao, Y., Bard, J.F., Jarrah, A.I., 2012. The therapist routing and scheduling problem. *IIE Trans.* 44, 868–893. <https://doi.org/10.1080/0740817X.2012.665202>
- Shi, Y., Boudouh, T., Grunder, O., 2019. A robust optimization for a home health care routing and scheduling problem with consideration of uncertain travel and service times. *Transp. Res. Part E Logist. Transp. Rev.* 128, 52–95. <https://doi.org/10.1016/j.tre.2019.05.015>
- Shi, Y., Boudouh, T., Grunder, O., Wang, D., 2018. Modeling and solving simultaneous delivery and pick-up problem with stochastic travel and service times in home health care. *Expert Syst. Appl.* 102, 218–233. <https://doi.org/10.1016/j.eswa.2018.02.025>
- Sienou, A., 2009. Proposition d'un cadre méthodologique pour le management intégré des risques et des processus d'entreprise.
- Sivakumar, A.I., 2003. Simulation-based scheduling for dynamic discrete manufacturing, in: *Proceedings of the 2003 Winter Simulation Conference, 2003.* IEEE, pp. 1465–1473.
- Solnon, C., 2013. *Ant colony optimization and constraint programming.* John Wiley & Sons.
- Song, C.-H., Lee, K., Lee, W.D., 2003. Extended simulated annealing for augmented TSP and multi-salesmen TSP, in: *Proceedings of the International Joint Conference on Neural Networks, 2003.* IEEE, pp. 2340–2343.
- Steinberg, D., Budinsky, F., Merks, E., Paternostro, M., 2008. *EMF: eclipse modeling framework.* Pearson Education.
- Strobel, G., 2021. Health in the Era of the Internet of Things: A Domain-Specific Modelling Language, in: *Proceedings of the 54th Hawaii International Conference on System Sciences.* p. 3705.
- Szander, N., Ros-McDonnell, L., de la Fuente, M.V., 2019. Algorithm for Efficient and Sustainable Home Health Care Delivery Scheduling, in: Mula, J., Barbastefano, R., Díaz-Madroñero, M., Poler, R. (Eds.), *New Global Perspectives on Industrial Engineering and Management, Lecture Notes in Management and Industrial Engineering.* Springer International Publishing, pp. 315–323.
- Taentzer, G., 2003. AGG: A graph transformation environment for modeling and validation of software, in: *International Workshop on Applications of Graph Transformations with Industrial Relevance.* Springer, pp. 446–453.
- Taentzer, G., Ehrig, K., Guerra, E., Lara, J. de, Lengyel, L., Levendovszky, T., Prange, U., Varro, D., Varro-Gyapay, S., 2005. Model transformation by graph transformation: A comparative study.
- Tang, L., Wang, X., 2008. An iterated local search heuristic for the capacitated prize-collecting travelling salesman problem. *J. Oper. Res. Soc.* 59, 590–599.

- Tang, S., Hanneghan, M., Hughes, T., Dennett, C., Cooper, S., Sabri, M.A., 2008. Towards a domain specific modelling language for serious game design, in: 6th International Game Design and Technology Workshop, Liverpool, UK.
- Thabet, R., 2020. Ingénierie dirigée par les modèles d'un pilotage robuste de la prise en charge médicamenteuse (PhD Thesis). Toulouse, France.
- Thabet, R., Lamine, E., Boufaied, A., Korbaa, O., Pingaud, H., 2018. Towards a Risk-Aware Business Process Modelling Tool Using the ADOxx Platform, in: Matulevičius, R., Dijkman, R. (Eds.), Advanced Information Systems Engineering Workshops, Lecture Notes in Business Information Processing. Springer International Publishing, pp. 235–248.
- Thompson, G.M., Goodale, J.C., 2006. Variable employee productivity in workforce scheduling. *Eur. J. Oper. Res.* 170, 376–390.
- TOGAF, 2021. TOGAF | The Open Group Website [WWW Document]. URL <https://www.opengroup.org/togaf> (accessed 7.27.21).
- Tolvanen, J.-P., 2006. Domain-specific modeling: How to start defining your own language. *DevX Com* 81.
- Torres-Ramos, A., Alfonso-Lizarazo, E., Reyes-Rubiano, L., Quintero-Araújo, C., 2014. Mathematical model for the home health care routing and scheduling problem with multiple treatments and time windows, in: Proceedings of the 1st International Conference on Mathematical Methods & Computational Techniques in Science & Engineering. pp. 140–145.
- Toth, P., Vigo, D., 2002. The vehicle routing problem. SIAM.
- Tournadre, N., 2002. La population de la France métropolitaine en 2050: un vieillissement inéluctable. *Econ. Stat.* 355, 57–71.
- Toussaint, H., 2016. IBM ILOG CPLEX Optimization Studio : Une introduction à OPL et Cplex Studio IDE 14.
- Trautsamwieser, A., Hirsch, P., 2014. A Branch-Price-and-Cut approach for solving the medium-term home health care planning problem. *Networks* 64, 143–159. <https://doi.org/10.1002/net.21566>
- Triki, N., Garaix, T., Xie, X., 2014. A two-phase approach for periodic home health care planning, in: 2014 IEEE International Conference on Automation Science and Engineering (CASE). Presented at the 2014 IEEE International Conference on Automation Science and Engineering (CASE), pp. 518–523. <https://doi.org/10.1109/CoASE.2014.6899375>
- Truyen, F., 2006. The Fast Guide to Model Driven Architecture: The Basics of Model Driven Architecture, Whitepaper. Cephas Consulting Corp. Archit. Oriented Serv.
- Van Hentenryck, P., 1999. The OPL optimization programming language. MIT press.
- Van Laarhoven, P.J., Aarts, E.H., 1987. Simulated annealing, in: Simulated Annealing: Theory and Applications. Springer, pp. 7–15.
- Van Slyke, R.M., Wets, R., 1969. L-shaped linear programs with applications to optimal control and stochastic programming. *SIAM J. Appl. Math.* 17, 638–663.
- Varró, D., Pataricza, A., 2004. Generic and meta-transformations for model transformation engineering, in: International Conference on the Unified Modeling Language. Springer, pp. 290–304.
- Vernadat, F., 1999. Techniques de modélisation en entreprise, applications aux processus opérationnels.

- Wand, M.P., Jones, M.C., 1994. Kernel Smoothing. CRC Press.
- Wang, T., Truptil, S., Benaben, F., 2017. An automatic model-to-model mapping and transformation methodology to serve model-based systems engineering. *Inf. Syst. E-Bus. Manag.* 15, 323–376. <https://doi.org/10.1007/s10257-016-0321-z>
- Williams, T.J., 1994. The Purdue enterprise reference architecture. *Comput. Ind.* 24, 141–158. [https://doi.org/10.1016/0166-3615\(94\)90017-5](https://doi.org/10.1016/0166-3615(94)90017-5)
- Wirnitzer, J., Heckmann, I., Meyer, A., Nickel, S., 2016. Patient-based nurse rostering in home care. *Oper. Res. Health Care* 8, 91–102. <https://doi.org/10.1016/j.orhc.2015.08.005>
- Yalçındağ, S., Matta, A., Şahin, E., Shanthikumar, J.G., 2016. The patient assignment problem in home health care: using a data-driven method to estimate the travel times of care givers. *Flex. Serv. Manuf. J.* 28, 304–335. <https://doi.org/10.1007/s10696-015-9222-6>
- Yang, X.-S., 2010. Firefly algorithm, stochastic test functions and design optimisation. *Int. J. Bio-Inspired Comput.* 2, 78–84.
- Yuan, B., Jiang, Z., 2017. Disruption Management for the Real-Time Home Caregiver Scheduling and Routing Problem. *Sustainability* 9, 2178. <https://doi.org/10.3390/su9122178>
- Yuan, B., Liu, R., Jiang, Z., 2015. A branch-and-price algorithm for the home health care scheduling and routing problem with stochastic service times and skill requirements. *Int. J. Prod. Res.* 53, 7450–7464. <https://doi.org/10.1080/00207543.2015.1082041>
- Yuan, Z., Fügenschuh, A., 2015. Home health care scheduling: a case study. Helmut-Schmidt- Univ., Professur für Angewandte Mathematik.
- Zefouni, S., 2012. Aide à la conception de workflows personnalisés: application à la prise en charge à domicile. Université de Toulouse, Université Toulouse III-Paul Sabatier.
- Zelm, M., Vernadat, F.B., Kosanke, K., 1995. The CIMOSA business modelling process. *Comput. Ind.* 27, 123–142.
- Zhan, Y., Wang, Z., Wan, G., 2021. Home service routing and appointment scheduling with stochastic service times. *Eur. J. Oper. Res.* 288, 98–110. <https://doi.org/10.1016/j.ejor.2020.05.037>
- Zhang, L., Bouchet, P.-Y., Lamine, E., Fontanili, F., Bortolaso, C., Derras, M., Pingaud, H., 2019a. Home Health Care Daily Planning Considering the Satisfaction of all the Stakeholders, in: 2019 International Conference on Industrial Engineering and Systems Management (IESM). Presented at the 2019 International Conference on Industrial Engineering and Systems Management (IESM), pp. 1–6. <https://doi.org/10.1109/IESM45758.2019.8948201>
- Zhang, L., Fontanili, F., Lamine, E., Bortolaso, C., Derras, M., Pingaud, H., 2021a. Stakeholders' Tolerance-based Linear Model for Home Health Care Coordination, in: 17th IFAC Symposium on Information Control Problems in Manufacturing.
- Zhang, L., Fontanili, F., Lamine, E., Bortolaso, C., Derras, M., Pingaud, H., 2020a. A Systematic Model to Model Transformation for Knowledge-Based Planning Generation Problems, in: Fujita, H., Fournier-Viger, P., Ali, M., Sasaki, J. (Eds.), Trends in Artificial Intelligence Theory and Applications. Artificial Intelligence Practices, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 140–152. https://doi.org/10.1007/978-3-030-55789-8_13
- Zhang, L., Fontanili, F., Lamine, E., Pingaud, H., Bortolaso, C., Derras, M., 2018. Une approche systémique de caractérisation des connaissances sur la coordination opérationnelle d'une prise en charge à domicile, in: 9ème Conférence Francophone En Gestion et Ingénierie Des Systèmes Hospitaliers.

- Zhang, L., Lamine, E., Fontanili, F., Bortolaso, C., Derras, M., Pingaud, H., 2019b. A Decision-Making Support System for Operational Coordination of Home Health Care Services, in: 2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA). Presented at the 2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA), pp. 1–6. <https://doi.org/10.1109/AICCSA47632.2019.9035219>
- Zhang, L., Lamine, E., Fontanili, F., Bortolaso, C., Sargent, M., Derras, M., Pingaud, H., 2020b. BLPAD. Core: A Multi-Functions Optimizer Towards Daily Planning Generation in Home Health Care, in: 2020 IEEE/ACS 17th International Conference on Computer Systems and Applications (AICCSA). IEEE, pp. 1–6.
- Zhang, L., Pingaud, H., Lamine, E., Fontanili, F., Bortolaso, C., Derras, M., 2021b. Etude prospective d'un environnement d'aide à la décision rendu configurable par des modèles, in: Conférence Internationale Génie Industriel QUALITA.
- Zhang, T., Yang, X., Chen, Q., Bai, L., Chen, W., 2018. Modified ACO for home health care scheduling and routing problem in Chinese communities, in: 2018 IEEE 15th International Conference on Networking, Sensing and Control (ICNSC). Presented at the 2018 IEEE 15th International Conference on Networking, Sensing and Control (ICNSC), pp. 1–6. <https://doi.org/10.1109/ICNSC.2018.8361373>

LISTE DES FIGURES

Figure 1 : Pyramide des âges au 01/01/2020 de la France métropolitaine dont la source provient de l'INSEE.....	2
Figure 2 : Logique de parcours de l'utilisateur (ANAP, 2020) dont la source provient de l'ARS Languedoc-Roussillon (2012).....	4
Figure 3 : Chiffres clés de l'HAD 2019 dont la source provient du FNEHAD	5
Figure 4 : Représentation synthétique de l'exercice de coordination (Lamine, 2019).....	9
Figure 5 : Couverture fonctionnelle de « BL.domicile ».....	13
Figure 6 : Alignement des défis sur les classes de problèmes de coordination opérationnelle .	15
Figure 7 : Offre et demande du système de PAD : une structuration de connaissances	26
Figure 8 : « Temps d'inactivité d'intervenant » et « temps d'attente de bénéficiaire » en cas de non-respect des fenêtres de temps de début d'intervention	30
Figure 9 : Relations entre la réalité, le modèle, le méta-modèle et le langage de modélisation (Thabet, 2020)	52
Figure 10 : Triangle du sens (Sienou, 2009)	54
Figure 11 : Définition d'un DSML adapté de (Thabet, 2020) et relation entre la modélisation, le DSML et la méthode de modélisation	55
Figure 12 : Composants de méthode de modélisation d'OMILAB (Karagiannis and Kühn, 2002)	56
Figure 13 : Pyramide à quatre niveaux du MDA selon l'OMG (OMG, 2014)	59
Figure 14 : Architecture dirigée par les modèles, la logique de transformation sous forme d'un Y (Combemale, 2008).....	60
Figure 15 : Architecture dirigée par les modèles : CIM-PIM-PSM-Code.....	61
Figure 16 : Transformation des modèles basée sur l'architecture dirigée par les modèles.....	62
Figure 17 : Architecture dirigée par les modèles pour l'aide à la décision de coordination dans la prise en charge à domicile avec le positionnement des questions de recherche.....	65
Figure 18 : Positionnement du DSML dans la pyramide en quatre niveaux d'OMG (Michael and Mayr, 2015).	71
Figure 19 : Relation entre TSP, m-TSP, VRP et HHCRSP	74
Figure 20 : Architecture et positionnement à l'OMG du DSML-HHCRSP.....	75
Figure 21 : Méta-Méta-Modèle (M ² M) du ADOxx (Fill and Karagiannis, 2013)	75
Figure 22 : Illustration de métamodèle du DSML-TSP	79

Figure 23 : Illustration du métamodèle du DSML-HHCRSP vue « offre ».....	84
Figure 24 : Illustration du métamodèle du DSML-HHCRSP vue « demande »	86
Figure 25 : Illustration du métamodèle du DSML-HHCRSP.....	90
Figure 26 : Représentation graphique d'un intervenant avec la qualification « infirmier » et les aptitudes « Malvoyant » et « Fin de vie ».....	91
Figure 27 : Illustration d'ALL et de la relation d'« héritage » dans ADOxx « Development Toolkit ».....	95
Figure 28 : Exemple du AttrRep et GraphRep dans ADOxx « Development Toolkit ».....	96
Figure 29 : Fonctionnement et composants de la méthode de modélisation.....	97
Figure 30 : Analyse SADT pour l'atelier 1 P1.1 : Modélisation des intervenants.....	98
Figure 31 : Analyse SADT pour l'atelier 1 P1.2 : Modélisation de la structure de PAD	98
Figure 32 : Analyse SADT pour l'atelier 2 P2.1 : Modélisation des bénéficiaires et leur demande d'interventions	99
Figure 33 : Analyse SADT pour l'atelier 3 P3.1 : modélisation des localisations des acteurs autour de la structure.....	99
Figure 34 : Analyse SADT pour l'atelier 3 P3.2: modélisation des exigences	100
Figure 35 : Analyse SADT pour l'atelier 3 P3.3 : modélisation des orientations de la structure et l'assemblage.....	100
Figure 36 : Modèle de besoin du TSP :4 villes à visiter avec 3 exigences et 1 orientation	101
Figure 37 : Modèle de besoin du TSPM (TSP with multiple visits).....	102
Figure 38 : Modèle de besoin du HHCRSP avec l'illustration des interfaces « <i>Notebook definition</i> » dans ADOxx : 4 interventions à réaliser par 1 intervenant, avec 3 exigences et 2 orientations	103
Figure 39 : Modèle de besoin du HHCRSP : 5 interventions à répondre par 2 intervenant, avec 5 exigences et 1 orientation.....	104
Figure 40 : Positionnement de deux méthodes de résolution dans l'architecture dirigée par les modèles.....	108
Figure 41 : Métamodèle graphe (Enriquez et al., 2015).....	111
Figure 42 : Illustration de la satisfaction des acteurs dans la PAD basée sur une fonction de régression linéaire.....	116
Figure 43 : Modèle de PLNM en langage OPL pour le TSP classique	118
Figure 44 : Méta-modèle du CSP	119
Figure 45 : Exemples de chaînes valides et invalides (OptaPlanner, 2021)	122
Figure 46 : Illustration de la relation entre quatre types de variable dans OptaPlanner	123
Figure 47 : Aperçu synthétique de mécanismes de résolution du problème par OptaPlanner (OptaPlanner, 2021).....	125

Figure 48 : Modèle du HHCRSP orienté OptaPlanner.....	126
Figure 49 : Fonction de régression linéaire pour modéliser la relation entre le score et le temps de dépassement des 3 contraintes temporelles.....	131
Figure 50 : Résultat du « benchmark » pour le cas d'utilisation « L_s130_p90_mp24_k19 » : le score de la partie « contrainte dure ».....	133
Figure 51 : Résultat du « benchmark » pour le cas d'utilisation « L_s130_p90_mp24_k19 » : le score de la partie « contrainte souple »	134
Figure 52 : Processus de traduction du score dans la méthode approchée vers la fonction objectif dans la méthode exacte	134
Figure 53 : Localisation des lieux de vie des bénéficiaires du cas ML_s55_p28_mp14_k4.....	139
Figure 54 : Succession des interventions de chaque tournée avec la méthode exacte	140
Figure 55 : Succession des interventions de chaque tournée avec la méthode approchée.....	140
Figure 56 : Diagramme de Gantt sur le respect du délai inter-intervention avec la méthode exacte	141
Figure 57 : Diagramme de Gantt sur le respect du délai inter-intervention avec la méthode approchée.....	142
Figure 58 : Diagrammes en vue de la qualification de deux solutions sur le critère du respect de capacité disponible.....	143
Figure 59 : Diagrammes en vue de la qualification de deux solutions sur le critère d'équilibrage de pénibilité des tournées	143
Figure 60 : Processus cognitif d'une prise de décision	147
Figure 61 : Principe de transformations des modèles du niveau CIM à celui du PSM.....	152
Figure 62 : Chaîne de transformations des modèles du niveau CIM à celui du PSM - application au TSP	153
Figure 63 : Modèle Graphe (M-G) sur un cas d'étude	154
Figure 64 : Modèle orienté Solution (M-S) transféré du Modèle Graphe (M-G) sur un cas d'étude	155
Figure 65 : Méta-Modèle Graphe (MM-G) au niveau PIM	157
Figure 66 : Méta-Modèle orienté Solution (MM-S) adapté de (Assouroko and Denno, 2016)	160
Figure 67 : Exemple de mapping pour l'« Arc Route »	163
Figure 68 : Extrait de la bibliothèque de patrons de formulation OPL.....	164
Figure 69 : Construction des patrons et des composants de formulations à l'aide de la bibliothèque et du catalogue.....	165
Figure 70 : Code OPL suite à l'application des règles de mapping entre MM-G et MM-S.....	167
Figure 71 : Illustration de navigation pour la formulation de contrainte « Flux Entrant »	167

Figure 72 : Codes OPL suite à l'application des règles de navigation dans le MM-S.....	168
Figure 73 : Règles de <i>parsing</i> basées sur une fonction Java pour l'affectation des mots clés et la mise en forme de fichiers CPLEX.....	169
Figure 74 : Traduction de M-C en M-G pour un cas d'étude.....	170
Figure 75 : Modèle Conceptuel (MM-C) de niveau CIM, sous ensemble du méta-modèle du DSML-TSP.....	171
Figure 76 : Traduction de M-C en M-G pour un cas d'étude : l'état partiel du M-G résultant de la seule application des règles mapping entre MM-C et MM-G.....	172
Figure 77 : Règles de mapping entre MM-C et MM-G et les règles de navigation dans MM-G	173
Figure 78 : Illustration de la visualisation de résultat par le cas d'études	174
Figure 79 : Spécification du cas d'utilisation 1 du TSP différent : les trois modèles et le résultat obtenu	175
Figure 80 : Spécification du cas d'utilisation 2 du TSP différent : les trois modèles et le résultat obtenu	176
Figure 81 : Captures d'écran de l'implémentation de l'outil de transformation BL.Autom....	177
Figure 82 : Ecran de BL.Autom: le M-C avant appel de « <i>TSP M to M transformation process</i> »	178
Figure 83 : Ecran de BL.Autom : M-G publié après transformation du M-C	178
Figure 84 : Contenus des fichiers de sortie de BL.Autom après appel de CPLEX.....	179
Figure 85 : Modélisation multi-acteurs avec la 3D et la réalité virtuel (Gordeeva et al., 2021)	185

LISTE DES TABLEAUX

Tableau 1 : Exemples de coordination dans la PAD selon les cinq catégories de dépendance définies dans la théorie de la coordination (Malone and Crowston, 1994)	8
Tableau 2 : Synthèse d'exigences de PAD.....	33
Tableau 3 : Synthèse des orientations recherchées par des structures de PAD.....	36
Tableau 4 : Synthèse sur les approches d'aide à la décision pour la coordination.....	46
Tableau 5 : Synthèse des variantes du TSP	76
Tableau 6 : Représentation graphique (syntaxe concrète) des concepts inclus dans le DSML-TSP.....	80
Tableau 7 : Exemple des aptitudes des intervenants	82
Tableau 8 : Liste d'énumération des exigences basée sur l'état de l'art.....	89
Tableau 9 : Représentation graphique (syntaxe concrète) des concepts inclus dans le DSML-HHCRSP.....	91
Tableau 10 : Représentation graphique (syntaxe concrète) des relations du DSML-HHCRSP.....	94
Tableau 11 : Concepts d'un graphe.....	110
Tableau 12 : Mécanisme d'annotations dans OptaPlanner.....	124
Tableau 13 : Classes d'information du modèle du HHCRSP.....	126
Tableau 14 : Classes d'entités de planification du modèle HHCRSP.....	127
Tableau 15 : Domaines des variables de décision du modèle HHCRSP	128
Tableau 16 : Paramètre de « satisfaction » des acteurs et de « difficulté » des interventions ...	135
Tableau 17 : Nombre de variables et de contraintes pour chaque cas d'utilisation	135
Tableau 18 : Résultat général de l'expérimentation.....	136
Tableau 19 : Résultat expérimental : 4 composantes de la fonction objectif représentant la satisfaction des acteurs	138
Tableau 20 : Comparaison entre cinq langages de transformation (MTL).....	150
Tableau 21 : Synthèse des concepts du MM-G	155
Tableau 22 : Synthèse d'une association entre contenu de contraintes et variables de décision associée.....	156
Tableau 23 : Représentation graphique des concepts inclus dans le MM-G.....	157
Tableau 24 : Synthèse des concepts du MM-S (vue « déclaration des données »)	158
Tableau 25 : Synthèse des concepts du MM-S (vue « construction des formulations »).....	159

Tableau 26 : Etapes à suivre pour la déclaration de l'« ensemble des tuples début-fin ».....	162
Tableau 27 : Etapes à suivre pour la déclaration de l'« Arc Route ».....	163
Tableau 28 : Extrait du catalogue des composants de formulation en OPL.....	164
Tableau 29 : Etapes à suivre pour la construction de la formulation de contrainte « Flux Entrant ».....	167
Tableau 30 : Etapes à suivre pour l'application des règles de mapping entre MM-C et MM-G	171
Tableau 31 : Etapes à suivre pour l'application des règles de navigation dans le MM-G.....	172

LISTE DES PUBLICATIONS

- (1) (L. Zhang et al., 2018) Zhang, L., Fontanili, F., Lamine, E., Pingaud, H., Bortolaso, C., Derras, M., 2018. Une approche systémique de caractérisation des connaissances sur la coordination opérationnelle d'une prise en charge à domicile, in: 9ème Conférence Francophone En Gestion et Ingénierie Des Systèmes Hospitaliers.
- (2) (Zhang et al., 2019a) Zhang, L., Bouchet, P.-Y., Lamine, E., Fontanili, F., Bortolaso, C., Derras, M., Pingaud, H., 2019. Home Health Care Daily Planning Considering the Satisfaction of all the Stakeholders, in: 2019 International Conference on Industrial Engineering and Systems Management (IESM). IEEE, pp. 1–6.
- (3) (Zhang et al., 2019b) Zhang, L., Lamine, E., Fontanili, F., Bortolaso, C., Derras, M., Pingaud, H., 2019. A Decision-Making Support System for Operational Coordination of Home Health Care Services, in: 2019 IEEE/ACS 16th International Conference on Computer Systems and Applications (AICCSA). IEEE, pp. 1–6.
- (4) (Zhang et al., 2020a) Zhang, L., Fontanili, F., Lamine, E., Bortolaso, C., Derras, M., Pingaud, H., 2020. A Systematic Model to Model Transformation for Knowledge-Based Planning Generation Problems, in: Fujita, H., Fournier-Viger, P., Ali, M., Sasaki, J. (Eds.), Trends in Artificial Intelligence Theory and Applications. Artificial Intelligence Practices, Lecture Notes in Computer Science. Springer International Publishing, Cham, pp. 140–152.
- (5) (Zhang et al., 2020b) Zhang, L., Lamine, E., Fontanili, F., Bortolaso, C., Sargent, M., Derras, M., Pingaud, H., 2020. BLPAD. Core: A Multi-Functions Optimizer Towards Daily Planning Generation in Home Health Care, in: 2020 IEEE/ACS 17th International Conference on Computer Systems and Applications (AICCSA). IEEE, pp. 1–6. (**Prix du meilleur article du workshop - Best Student Workshop Paper Award**)
- (6) (Zhang et al., 2021a) Zhang, L., Fontanili, F., Lamine, E., Bortolaso, C., Derras, M., Pingaud, H., 2021. Stakeholders' Tolerance-based Linear Model for Home Health Care Coordination, in: 17th IFAC Symposium on Information Control Problems in Manufacturing. In press.
- (7) (Zhang et al., 2021b) Zhang, L., Pingaud, H., Lamine, E., Fontanili, F., Bortolaso, C., Derras, M., 2021. Etude prospective d'un environnement d'aide à la décision rendu configurable par des modèles, in: Conférence Internationale Génie Industriel QUALITA. In press.

From business vision to assisted schedule generation for centralized coordination of home health care services

Abstract:

Nowadays, with the development of post-operative care, the management of disabilities, long duration disease or the frail elderly losing their autonomy, Home Health Care (HHC) services are developed increasingly by medical and social institutions. Mostly, these organizations have a lack of methods and tools to support the management of such a system of HHC. When providing care and assistance services, the demand of care beneficiaries is distributed over a large area, which means that there is a requirement for new capacities in the HHC services organization and in the real-life coordination of the caregivers who are mobilized to perform the care services. Due to the rapid increase in demand and the diversity of HHC institutions, a twofold need arises: the ability to specify their own HHC coordination problem, and the capacity to generate schedules adapted to their specific needs. Therefore, we propose a framework addressing these two aspects, based on a decision-support environment using a Model Driven Engineering (MDE) approach. On the basis of a state of the art regarding the HHC, the research work firstly aims at developing a Domain Specific Modeling Language (DSML) to capture the requirements of coordination. Then, we developed two methods for planning activities according to two formulations of the coordination problem in HHC (known as the Home Health Care Routing and Scheduling Problem - HHCRSP): one is expressed as a Mixed Number Linear Programming (MILP) formulation and solved by CPLEX, and the other is codified as a Constraint Satisfaction Problem (CSP) and solved by a metaheuristic using OptaPlanner, which is able to deal with large-scale problems. The solutions generated by these two methods are compared. Finally, we built a prototype of a digital factory for coordination in HHC that gathers our findings, which is called BL.Autom. The conception of this prototype follows the principles of Model-Driven Architecture (MDA) and its development has been achieved through the use of the meta-modeling tool ADOxx. This prototype facilitates the coordinators' work by providing them with the ability to progressively modify their requirements and preferences in terms of performance. In practice, this tool emphasizes the search for the stakeholders' satisfaction in the field of HHC (care beneficiary and caregiver).

Keywords: Model driven engineering, Domain specific modelling language, Optimization, Satisfaction of stakeholders, Home health care, Coordination

De la vision métier à la génération assistée de plannings pour la coordination centralisée de services de soins à domicile

Résumé :

Aujourd'hui, qu'il s'agisse du développement du post-opératoire en ambulatoire, de la prise en charge de handicap, d'affectations de longue durée ou des personnes âgées fragiles en perte d'autonomie, la Prise en charge A Domicile (PAD) par des structures médico-sociales se développe. Le plus souvent, l'organisation de ces services à la personne a un caractère artisanal, mobilisant peu de méthodes et d'outils pour soutenir le pilotage d'un tel système de PAD. Lors de la prise en charge de services de soins et d'accompagnement, la demande est distribuée sur un territoire. Dès lors, de nouvelles capacités sont requises dans l'organisation de la prise en charge et, tout particulièrement, dans la réalité de la coordination des intervenants qui se déplacent. Face à la montée en charge rapide de la demande et de la diversité des structures de PAD, se manifeste un double besoin : (1) celui de pouvoir spécifier leur propre problème de coordination de PAD, et (2) celui de générer des plannings adaptés à ce besoin spécifique. Nous proposons donc un cadre abordant ces deux aspects pour cerner un sujet complexe. Notre étude est basée sur un environnement d'aide à la prise de décision utilisant une approche d'Ingénierie Dirigée par les Modèles (IDM). Sur la base d'un bilan de connaissances de ces PAD, les travaux de recherche visent d'abord le développement d'un langage de modélisation spécifique au domaine pour capturer le besoin de coordination. Ensuite, nous avons développé des méthodes de planification des activités selon deux formulations du problème de coordination (dit HHCRSP - Home Health Care Routing and Scheduling Problem) : une méthode exacte en Programmation Linéaire en Nombre Mixte (PLNM) résolue par le solveur CPLEX, et une méthode approchée en satisfaction de contraintes couplée à une méta-heuristique, apte à traiter des problèmes de grande taille et résolue par OptaPlanner. Nous les avons comparées. Enfin, nous avons bâti un prototype d'une fabrique numérique de la coordination en PAD réunissant les résultats acquis, appelé BL.Autom. Sa conception suit les principes de l'architecture dirigée par les modèles, et son développement a été réalisé à l'aide de l'outil de méta-modélisation ADOxx. Ce prototype facilite l'exercice du métier du coordinateur en lui procurant la faculté de modifier au fil de l'eau ses orientations en termes de performance et ses exigences. Dans son usage, cette fabrique peut mettre l'accent sur la recherche de satisfaction des parties-prenantes (bénéficiaire et intervenant) dans ce prisme d'un service à la personne.

Mots clés : Ingénierie dirigée par les modèles, Langage de modélisation spécifique au domaine, Optimisation, Satisfaction des parties prenantes, Prise en charge à domicile, Coordination