



**HAL**  
open science

# Combinaison des techniques d'optimisation et de l'intelligence artificielle distribuée pour la mise en place d'un système de covoiturage dynamique

Manel Sghaier

► **To cite this version:**

Manel Sghaier. Combinaison des techniques d'optimisation et de l'intelligence artificielle distribuée pour la mise en place d'un système de covoiturage dynamique. Autre. Ecole Centrale de Lille, 2011. Français. NNT : 2011ECLI0021 . tel-00689957

**HAL Id: tel-00689957**

**<https://theses.hal.science/tel-00689957>**

Submitted on 20 Apr 2012

**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.

N° d'ordre : 174

**ECOLE CENTRALE DE LILLE**

## **THESE**

Présentée en vue d'obtenir le grade de

## **DOCTEUR**

en

**Spécialité : Automatique, Génie Informatique, Traitement du Signal et  
Image**

par

**Manel SGHAIER**

DOCTORAT DELIVRE PAR L'ÉCOLE CENTRALE DE LILLE

**Combinaison des techniques d'optimisation et de l'intelligence artificielle  
distribuée pour la mise en place d'un système de covoiturage dynamique**

Soutenue publiquement le 16 Décembre 2011 devant le jury d'examen :

<b>Président :</b>	<b>Robert CLAVEL</b>	Chargé de projets « Transports Innovants »- CERTU / Lyon
<b>Rapporteurs :</b>	<b>Abdellah EL-MOUDNI</b>	Professeur, Université de Technologie de Belfort-Montbéliard
	<b>Mariagrazia DOTOLI</b>	Professeur, Politecnico di Bari
<b>Examineurs :</b>	<b>Jean SENG</b>	Chargé de mission « Information multimodale »-Ministère des Transports
	<b>Nawel ZANGAR</b>	Dr, MCF, Faculté des Sciences de Tunis (FST)
<b>Directeurs :</b>	<b>Slim HAMMADI</b>	Professeur, École Centrale de Lille
	<b>Christian TAHON</b>	Professeur, Université de Valenciennes et du Hainaut-Cambrésis
<b>Co-encadreur :</b>	<b>Hayfa ZGAYA</b>	Dr, MCU, Université Lille 2

Thèse préparée dans le laboratoire LAGIS UMR 8219 à l'École Centrale de Lille  
Ecole Doctorale SPI 072 (EC Lille)

**PRES Université Lille Nord-de-France**



*A mon père et ma mère*

*Pour l'amour qu'ils ont su me prodiguer*

*A mon frère et ma sœur*

*Pour leur soutien et leur patience*

*A mes nièces*

*Pour le bonheur qu'elles apportent à ma vie*

*A moi*

*Preuve de résistance et de patience*



*Remerciements*

C'est un agréable devoir pour moi que d'exprimer ma profonde reconnaissance à mon encadrant de thèse, le Professeur Slim HAMMADI, pour ses grandes qualités humaines très appréciables. Professeur au Laboratoire d'Automatique, Génie Informatique et Signal (LAGIS) de l'Ecole Centrale de Lille, il a su me faire profiter de son expérience confirmée ainsi que de ses larges connaissances scientifiques. Il m'a ainsi orientée tout au long des trois années qu'a duré cette thèse, apportant encouragement, réconfort et compréhension en plus de son savoir et de son expertise.

Je remercie également Professeur Christian TAHON, co-directeur de cette thèse du laboratoire d'Automatique, de Mécanique et d'Informatique industrielles et Humaines (LAMIH) de l'Université de Valenciennes et du Hainaut-Cambrésis, et sans qui ce projet n'aurait peut-être pas abouti.

Je tiens à remercier tout aussi chaleureusement Docteur Hayfa ZGAYA, Maitre de Conférence Universitaire de l'Université Lille II, de m'avoir soutenue continuellement et inlassablement. Elle a fait preuve d'une présence scientifique imposante pour voir ces travaux prendre forme progressivement et enfin aboutir.

Mes remerciements s'adressent exceptionnellement à Monsieur Robert Clavel, chargé de projets « Transports Innovants » au Centre d'Etudes sur les Réseaux, le Transport, l'Urbanisme et les constructions publiques, de m'avoir honoré en acceptant de présider mon jury de soutenance.

Je suis particulièrement sensible, par ailleurs, au grand honneur dont m'ont gratifié Professeur Abdellah EL-MOUDNI de l'Université de Technologie de Belfort-Montbéliard et Professeur Mariagrazia DOTOLI de la Politecnico di Bari, pour l'intérêt qu'ils ont bien voulu porter à mes travaux en acceptant d'en être les rapporteurs, qu'ils trouvent ici l'expression de ma profonde reconnaissance.

Je souhaite également témoigner de mes vifs remerciements à Monsieur Jean SENG, chargé de mission « Information multimodale » au Ministère de l'écologie, de l'énergie, du développement durable et de l'aménagement du territoire et Docteur Nawel Zangar, docteur et Maitre de Conférence Universitaire à la Faculté des Sciences de Tunis / Tunisie d'avoir accepté de faire partie de mon jury d'Examen.

## Remerciements

Des remerciements particuliers que j'adresse au personnel du LAGIS et de l'Ecole Centrale de Lille en général témoignent de la bonne ambiance dans laquelle j'ai évolué durant ces trois années.

Je remercie également mes amis pour leur soutien moral considérable et pour m'avoir aidée à résister durant toutes ces années.

Finalement et le meilleur reste toujours pour la fin, il n'est de mot qui puisse exprimer ma profonde reconnaissance et mon amour pour ma famille, qui n'a d'égal que le leur. Je les remercie pour toute l'affection, la présence, la confiance et l'amour dont ils ont fait preuve. Leurs soutien et encouragements ont largement contribué à ce succès que je leur dédie aujourd'hui. Je remercie infiniment mes chers parents pour leur patience et pour avoir cru en moi, mon frère et ma sœur pour avoir fait de ma vie ce qu'elle est et je remercie aussi mes petites diablasses de nièces pour le bonheur et la joie qu'elles apportent.

# TABLE DES MATIERES

## Contenu

Table Des Matieres.....	7
Table des Figures .....	13
Glossaire.....	17
Introduction Generale.....	27
I. Chapitre I .....	31
Introduction au contexte.....	31
le covoiturage à la veille d'une pratique de Transport Comodal .....	31
I.1. Introduction .....	32
I.1.1. Développement durable : à l'encontre des menaces sur l'environnement .....	32
I.1.2. Transport : son rôle dans les menaces sur l'environnement .....	34
I.1.3. L'évolution du transport .....	37
I.1.4. Vers une mobilité durable et soutenable.....	38
I.2. Transport : Les différents moyens.....	39
I.2.1. Le transport classique .....	39
I.2.2. Transport en commun Versus Automobile.....	45
I.3. Transport : Les différents modes.....	47
I.3.1. Le transport monomodal.....	47
I.3.2. Le transport intermodal.....	47
I.3.3. Le transport multimodal .....	48
I.3.4. Le transport comodal .....	49
I.4. Le transport comodal : Vers un système de services de transport .....	50
I.4.1. Pour l'équité dans le choix des modes de transport.....	50
I.4.2. Complémentarité des systèmes.....	51

## Table des Matières

I.4.3. Perturbations dans les systèmes de transport.....	52
I.4.4. Transport comodal : une solution pour la régulation.....	54
I.4.5. La voiture : de l'individuel vers le collectif.....	55
I.5. La voiture partagée.....	56
I.5.1. L'autopartage.....	57
I.5.2. Le covoiturage.....	61
I.6. Proposition d'une approche optimisée pour le covoiturage dynamique.....	73
I.6.1. Contexte et Motivations.....	73
I.6.2. Description du système.....	74
I.7. Conclusion.....	77
II. Chapitre II.....	79
Les Systèmes Multi-Agents et l'optimisation :.....	79
Une alliance Vers une méthodologie de résolution efficace.....	79
II.1. Introduction.....	80
II.2. Optimisation.....	82
II.2.1. Optimisation : Concept et définition.....	82
II.2.2. Délimitation de l'espace de recherche.....	84
II.2.3. Différents critères d'optimalité pour différents types d'optimum.....	85
II.2.4. Classification des problèmes d'optimisation.....	86
II.2.5. Optimisation mono-objectif.....	87
II.2.6. Optimisation multi-objectif.....	87
II.2.7. Les méthodes d'optimisation.....	90
II.2.8. l'Optimisation : une multitude de domaines concernés.....	104
II.3. CODAC : l'optimisation au service de la qualité dans le covoiturage dynamique ....	108
II.3.1. CODAC : une optimisation au profit de la satisfaction de l'utilisateur.....	108
II.3.2. Complexité du problème de covoiturage dynamique optimisé.....	109
II.3.3. Avis et mesures.....	113

## Table des Matières

II.4. Les systèmes multi-agents au service de l'optimisation .....	114
II.4.1. SMA : le concept.....	115
II.4.2. Interaction et communication.....	121
II.4.3. Les SMA envahissent les domaines de recherche.....	126
II.4.4. Le paradigme agent dans le domaine du transport.....	127
II.4.5. Les techniques multi-agents intégrées au concept de covoiturage.....	128
II.5. CODAC : l'intelligence collective via des agents optimisateurs en coalition .....	130
II.5.1. Optimisation dans le covoiturage dynamique .....	132
II.5.2. Les SMA au service de l'efficacité technique.....	134
II.5.3. L'alliance entre les SMA et les techniques de l'Optimisation.....	135
II.6. Un Graphe Dynamique Distribué pour la modélisation .....	137
II.7. Conclusion .....	138
III. Chapitre III.....	139
Modèle Représentatif Du Système :.....	139
Mise en place d'une Architecture.....	139
Dynamique Distribuée.....	139
III.1. Introduction .....	140
III.2. Système de covoiturage dynamique optimisé .....	141
III.2.1. Optimisation dans les systèmes de covoiturage dynamique .....	141
III.2.2. Approche proposée : Critères d'Optimisation .....	145
III.3. Une modélisation graphique innovante pour la mise en place d'une architecture physique distribuée .....	146
III.3.1. Une modélisation sous forme de graphe dynamique distribué pour la mise en place d'une architecture physique évolutive.....	147
III.3.2. Subdivision du réseau géographique desservi : mise en place d'une architecture physique dynamique distribuée.....	174
III.4. Conclusion .....	189
IV. Chapitre IV.....	191

## Table des Matières

CODAC : Une plateforme de Covoiturage Dynamique Optimisé basée sur les agents communicants.....	191
IV.1. Introduction .....	192
IV.2. Le paradigme agent à la base d'une décentralisation .....	194
IV.2.1. SMA : pour une stratégie de resolution efficace .....	194
IV.2.2. Graphe Dynamique Distribué : Une modélisation au service de la distribution logicielle.....	195
IV.3. CODAC : de l'analyse à la conception .....	195
IV.3.1. Problème de Covoiturage Dynamique Optimisé : Analyse Fonctionnelle .....	196
IV.3.2. Quelle modélisation pour quel objectif ?.....	207
IV.4. CODAC : Un contexte d'Intelligence Artificielle Distribuée .....	210
IV.5. CODAC : Une architecture multi-agent.....	211
IV.5.1. A chaque agent son espace d'action .....	213
IV.5.2. A chaque agent ses propres fonctions.....	214
IV.6. CODAC : un système basé sur des agents communicants .....	224
IV.6.1. Communication Inter-Système .....	225
IV.6.2. Communication Intra-Système .....	227
IV.7. Modélisation formelle : pour un procédé d'affectation optimisé et performant .....	228
IV.7.1. A3D : Algorithme Distribué Dynamique basé sur Dijkstra.....	229
IV.7.2. ODAVe : Optimisation Distribuée de l'Affectation des Véhicules aux usagers	233
IV.8. Conclusion.....	238
V. Chapitre V .....	241
Implémentation et Déploiement du service de covoiturage dynamique .....	241
V.1. Introduction.....	242
V.2. CODAC : une plateforme de covoiturage dynamique optimisé.....	243
V.2.1. CODAC : Une solution dynamique attractive .....	243
V.2.2. Domaines d'intérêts scientifiques .....	244
V.2.3. CODAC : Le débriefing .....	247

## Table des Matières

V.3. Plateforme de développement Multi-Agent.....	250
V.3.1. Processus de développement : Agent ou objet ? .....	250
V.3.2. Différentes plateformes de développement.....	251
V.3.3. JADE : le choix d'une plateforme.....	253
V.3.4. Spécificités de la plateforme JADE .....	255
V.4. Déploiement.....	261
V.4.1. Architecture Centralisée.....	262
V.4.2. Architecture Distribuée : .....	263
V.4.3. Déploiement de CODAC : Choix d'une architecture .....	264
V.5. Architecture en couche .....	266
V.5.1. Modèle MVC .....	266
V.5.2. L'API JDBC.....	272
V.6. Cartocom.....	273
V.7. Tests et Scénarii d'exécution .....	278
V.7.1. Données sur les usagers .....	279
V.7.2. Traitements et affectations .....	283
V.7.3. Quelques calculs .....	287
V.7.4. Affichage des résultats .....	290
V.7.5. DOMARTiC : Une couche Vue.....	291
V.8. Conclusion .....	296
Conclusion Générale .....	299
Annexe 1 .....	301
Bibliographie.....	307



## TABLE DES FIGURES

Figure I.1. Evolution de la moyenne de la température entre 1850 et 2010 .....	33
Figure I.2. Evolution des catastrophes naturelles.....	34
Figure I.3. Taux de rejet de CO <sub>2</sub> et GES en Europe par secteur de service .....	35
Figure I.4. Variation des émissions de CO <sub>2</sub> en EU-27 par secteur entre 1990 et 2008.....	35
Figure I.5. Variation des émissions polluantes entre 1990 et 2009.....	36
Figure I.6. Variation en pourcentage du taux d'émission de CO <sub>2</sub> en EU-27 par secteur .....	36
Figure I.7. Évolution du nombre de voyageurs au kilomètre en interne en France et en Europe .....	37
Figure I.8. Consommation de produits pétroliers par secteur .....	38
Figure I.9. Etude statistique de l'utilisation des français des modes de transports .....	42
Figure I.10. Importance des taux d'émission de CO <sub>2</sub> par mode de transport.....	44
Figure I.11. Evolution du comportement des individus par rapport aux modes de transport adoptés.....	46
Figure I.12. Evolution des perturbations en France entre les années 2000 et 2008 .....	53
Figure I.13. Evolution de l'afflux des voyageurs aux modes de transport de 1990 à 2008 .....	55
Figure I.14. La voiture partagée : Autopartage ou Covoiturage ?.....	57
Figure I.15. Autopartage : Planification des déplacements, utilisation et tarification .....	58
Figure I.16. Transporter 30 personnes en autobus, voitures personnelles ou covoiturage ?....	63
Figure I.17. Notions de base pour le covoiturage .....	65
Figure I.18. Limites des systèmes de covoiturage existants .....	72
Figure I.19. CODAC : Un système de covoiturage dynamique optimisé compétitif à grande échelle.....	76
Figure I.20. CODAC : Les principaux modules.....	77
Figure II.1. Modélisation et Résolution pour l'optimisation.....	83
Figure II.2. Espace de Recherche (19) .....	84
Figure II.3. Espace de valeurs Réalisables (19) .....	84
Figure II.4. Maximum global, local fort ou local faible ? .....	86
Figure II.5. Optimisation multi-objectif et optimalité au sens de Pareto .....	90
Figure II.6. Méthodes d'optimisation monobjectif (56).....	91
Figure II.7. Organigramme de la métaheuristique du « Recuit Simulé » (59).....	95
Figure II.8. Méthode de recherche tabou (59).....	97

## Table des Figures

Figure II.9. Organigramme des algorithmes évolutionnaires (59).....	100
Figure II.10. Organigramme des algorithmes génétiques (59).....	101
Figure II.11. Cycle de vie d'un agent (2) .....	117
Figure II.12. Les interactions sous leurs différentes formes .....	121
Figure II.13. Structure Hiérarchique d'une OMA (82) .....	124
Figure II.14. Acte de Communication : Principe .....	125
Figure II.15. Traitement instantané : Affectation dynamique des véhicules aux usagers.....	131
Figure II.16. CODAC : Optimisation dans et autour des agents.....	136
Figure III.1. Critères identifiés pour l'optimisation des services de covoiturage .....	144
Figure III.2. Représentation des itinéraires des voitures .....	150
Figure III.3. Partage d'itinéraires : une ou plusieurs voitures par tronçon.....	151
Figure III.4. Ensembles de nœuds successeurs et prédécesseurs .....	152
Figure III.5. Itinéraires composés des voitures .....	166
Figure III.6. Succession de tronçons sur le parcours d'une voiture .....	168
Figure III.7. Composition des solutions d'affectation en fonction de l'offre disponible.....	169
Figure III.8. Terrain vierge commun à deux zones .....	177
Figure III.9. Intersection non vide de zones (point(s) et / ou route(s)) .....	178
Figure III.10. Calcul des centres pour une zone contenant un point unique .....	182
Figure III.11. Calcul des centres pour une zone contenant 2 points .....	182
Figure III.12. Calcul des centres et création des zones : ensembles de 3 voisins ou plus.....	183
Figure IV.1. Fonctionnalités du Système proposé .....	197
Figure IV.2. Diagramme FAST : Composition de fonctions .....	199
Figure IV.3. Réception et structuration des requêtes quasi-simultanées.....	200
Figure IV.4. Extraction et organisation des offres de covoiturage.....	201
Figure IV.5. Construction du modèle représentation du système .....	203
Figure IV.6. Décomposition du réseau.....	205
Figure IV.7. Génération de solutions optimisées aux requêtes des usagers.....	206
Figure IV.8. Vue sur l'architecture du système CODAC.....	210
Figure IV.9. CODAC : Un système basé sur les Agents Communicants .....	212
Figure IV.10. Agents : Une portée étendue ou limitée.....	214
Figure IV.11. Un AIU pour chaque piéton (passager potentiel) connecté.....	217
Figure IV.12. Un AIV pour communiquer avec la voiture (i.e. le conducteur).....	218
Figure IV.13. Paramètre $\Delta\epsilon$ pour l'acquisition des requêtes.....	220
Figure IV.14. Un AI pour la collecte et structuration des données .....	221

## Table des Figures

Figure IV.15. Un AD pour la subdivision du réseau.....	222
Figure IV.16. Un AF pour le dispatching des réponses .....	223
Figure IV.17. Collecte et Organisation des données sur les offres et demandes de covoiturage .....	227
Figure IV.18. Un AO pour le traitement optimisé distribué des requêtes basé sur ODAVe .	235
Figure V.1. CODAC : Une panoplie de domaines scientifiques.....	246
Figure V.2. Différents aspects pour l'élaboration de CODAC .....	247
Figure V.3. SMA VS Systèmes Orientés Objets (2) .....	251
Figure V.4. Plateforme et conteneur JADE (2).....	256
Figure V.5. Exemple d'initialisation de l'agent décomposeur.....	257
Figure V.6. Exemple d'utilisation de la méthode getAID.....	257
Figure V.7. Un exemple d'implémentation du « OneShotBehaviour » dans l'AI.....	259
Figure V.8. Exemple d'implémentation d'un 'Cyclic Behaviour' .....	260
Figure V.9. CODAC déployé sur une architecture centralisée .....	263
Figure V.10. Déploiement de notre système CODAC .....	265
Figure V.11. Insertion d'une requête de covoiturage.....	269
Figure V.12. Insertion d'une offre de covoiturage.....	270
Figure V.13. Architecture du système et implication des agents dans chacune des couches	271
Figure V.14. Interactions entre les couches du modèle MVC.....	271
Figure V.15. Positions des origines et destinations des piétons et véhicules sur une cartographie réelle .....	281
Figure V.16. Traçage de l'itinéraire de la voiture C <sub>7</sub> .....	282
Figure V.17. Traçage de l'itinéraire de la voiture C <sub>3</sub> .....	282
Figure V.18. Représentation des usagers (piétons) et leurs itinéraires .....	283
Figure V.19. Zones Primaires .....	284
Figure V.20. Spécification d'un rayon de zones .....	285
Figure V.21. Graphe pour les origines et destinations des voitures.....	285
Figure V.22. Superposition Covoitureurs & Covoiturés.....	286
Figure V.23. Superposition des plans voitures et piétons sur cartocom .....	286
Figure V.24. Zones Intermédiaires et couverture du réseau .....	287
Figure V.25. Traitement de R <sub>4</sub> .....	288
Figure V.26. Traitement de R <sub>5</sub> .....	288
Figure V.27. Traitement de R <sub>6</sub> .....	289
Figure V.28. Traitement de R <sub>3</sub> .....	289

## Table des Figures

Figure V.29. Traitement de la deuxième partie de $R_3$ .....	290
Figure V.30. Affichage des itinéraires solutions.....	290
Figure V.31. Génération aléatoire de données .....	291
Figure V.32. Un exemple d'exécution .....	292
Figure V.33. Coalition entre AO et recomposition d'itinéraires.....	293
Figure V.34. Variation des Temps d'exécution .....	294
Figure V.35. Activités des Agents du système.....	296
Figure V.36. Performances de CODAC.....	297
Figure A.1. Modèle OOSE.....	303
Figure A.2. Les vues UML (136).....	304

## GLOSSAIRE

- ACL :** Agent Communication Language
- ADEME :** Agence De l'Environnement et de la Maîtrise de l'Énergie
- AE :** Algorithme Evolutionnaire
- AG :** Algorithme Génétique
- AOT :** Autorité Organisatrice des Transports
- Autopartage :** désigne le partage d'un véhicule sur des intervalles de temps différents et dont l'usage est réservé à une personne différente à chaque fois (lors de chaque intervalle de temps spécifique)
- CDO :** Covoiturage Dynamique Optimisé
- CNIL :** Commission Nationale de l'Informatique et des Libertés
- CODAC :** Covoiturage Optimisé Dynamique basé sur les Agents Communicants
- Comodalité :** On entend par « comodalité » l'utilisation efficace des modes de transport, qu'ils soient exploités seuls ou dans le cadre d'une intégration multimodale, de façon à permettre une utilisation optimale et durable des ressources
- Covoiturage :** Désigne le partage d'une voiture par un groupe de personnes (dans la limite de la capacité du véhicule) au cours d'un même intervalle de temps pour effectuer ensemble un trajet commun
- DARP :** Dial-A-Ride Problem : problème de transport à la demande spécifique à la catégorie de personnes âgées ou handicapées.
- DPDP :** Dynamic Pickup and Delivery Problem
- Ecopartage :** désigne l'utilisation de la voiture particulière à des fins de transport collectif, cette catégorie regroupe l'Autopartage et le Covoiturage
- FIPA :** Foundation for Intelligent Physical Agents
- GDD :** Graphe Dynamique Distribué
- GES :** Gaz à Effet de Serre

## Glossaire

<b>GIS :</b>	Geographical Information Service
<b>GPRS :</b>	General Packet Radio Service
<b>GPS :</b>	Global Positioning System
<b>KQML :</b>	Knowledge Query and Manipulation Language
<b>HOV lanes :</b>	High Occupancy Vehicles lanes
<b>IAD :</b>	Intelligence Artificielle Distribuée
<b>ICARO :</b>	Increase Car Occupancy
<b>LOTI</b>	Loi d'Orientation des Transports Intérieurs
<b>NTIC :</b>	Nouvelles Technologie de l'Information et de la Communication
<b>OMA :</b>	Organisation Multi-Agent
<b>Oseo :</b>	Schématiquement appelée banque publique française de financement des PME innovantes ( <a href="http://www.oseo.fr/">http://www.oseo.fr/</a> )
<b>PA :</b>	Personal Agent
<b>PDA :</b>	Personal Digital Assistant
<b>PDP :</b>	Pick and Delivery Prolem : problème de transport, prise et depose de biens ou de personnes d'un lieu à un autre
<b>Sciences Po :</b>	Institut d'études politiques à Paris ( <a href="http://www.sciencespo.fr/">http://www.sciencespo.fr/</a> )
<b>SIAD :</b>	Système d'Information d'Aide au Déplacement
<b>SMA :</b>	Système Multi-Agent
<b>SP :</b>	Swapping Problem : problème de redistribution (dispatching) de biens de nœuds origines vers des nœuds destinations selon certaines contraintes (e.g. type d'objet désiré dans chaque nœud...)
<b>STIC :</b>	Sciences et Technologies de l'Information et de la Communication
<b>TAD :</b>	Transport A la Demande, un mode de transport collectif dont les requêtes des utilisateurs définissent les grandes lignes (itinéraires, horaire, fréquence, etc.)
<b>TMR :</b>	Tableau de Marche Réel

<b>TMT :</b>	Tableau de Marche Théorique
<b>TSP :</b>	Travel Salesman Problem (Problème du voyageur de commerce)
<b>VRP :</b>	Vehicle Routing Problem : problème de tournée de véhicule pour le transfert, la redistribution de biens ou de personnes

### **Glossaire Formulation**

Dans ce qui suit un glossaire définissant toutes les expressions et variables utilisées pour la formulation du problème de covoiturage dynamique optimisé dans le cadre du système proposé *CODAC* (c.f. *Chapitre III*).

<b>Modélisation en Graphe</b>		$T(t)$	Le triplet modélisant le système et constitué de $G(t)$ , $R(t)$ et $O(t)$
	<b>Graphe Dynamique</b>	$G(t)$	Un graphe orienté modélisant le système à l'instant $t$
		$N(t)$	L'ensemble des nœuds du graphe $G(t)$
		$A(t)$	L'ensemble des arcs de $G(t)$
		$(a,b)$	Un arc orienté dans $G(t)$
		$N_x^+(t)$	L'ensemble des nœuds successeurs de $x$ dans $G(t)$
		$N_x^-(t)$	L'ensemble des nœuds prédécesseurs de $x$ dans $G(t)$
		$\rho_{O,D}(t)$	Fonction de pondération des arcs de $G(t)$ permettant de calculer à un instant $t$ le temps de parcours d'une origine $O$ à une destination $D$
<b>Modélisation des Données</b>	<b>Requêtes des Usagers (Piétons)</b>	$U(t)$	Les piétons (potentiels passagers) ayant émis des requêtes
		$R(t)$	L'ensemble des requêtes de covoiturage émises à un instant $t$

	$R_p(t)$	La requête de l'utilisateur $p$ ( $U_p$ )
	$R_p^+$	Origine de la requête émise par l'utilisateur $U_p$
	$R_p^-$	Destination requise par l'utilisateur $U_p$
	$N_R(t)$	L'ensemble des nœuds composant les trajets demandés par les utilisateurs ( $U(t)$ )
	$P_p$	Le nombre de personnes ( $U_p$ inclus) concernées par la requête ( $R_p(t)$ )
	$[d_p, a_p]$	Indique l'intervalle durant lequel le voyage doit avoir lieu, avec pour extrémité initiale le temps de départ au plus tôt ( $d_p$ ) et pour extrémité finale le temps d'arrivée au plus tard ( $a_p$ )
	$D(t)$	Représentation matricielle regroupant l'ensemble des demandes des usagers (piétons)
	$D_{i,j}$	L'élément d'indices de $D(t)$ : demandes en attente sur la route $[a \rightarrow b]$ , $i$ et $j$ représentent respectivement $a$ et $b$
<b>Offres des voitures (Conducteurs)</b>	$V(t)$	Les usagers conducteurs de véhicules ayant spécifiés des offres de déplacement
	$O(t)$	L'ensemble des offres des conducteurs avec les spécificités de déplacements possibles à l'instant $t$
	$O_j(t)$	L'offre du véhicule $j$ ( $V_j$ )

$O_j^+$	Origine de l'offre spécifiée par le véhicule $V_j$
$O_j^-$	Destination finale du véhicule $V_j$
$N_o(t)$	L'ensemble des nœuds composant les itinéraires empruntés par les véhicules ( $V(t)$ )
$l_j$	Le nombre de places disponibles dans le véhicule $V_j$
$[d_j, a_j]$	Intervalle durant lequel le voyage doit avoir lieu, $d_j$ est le temps de départ au plus tôt et $a_j$ est le temps d'arrivée au plus tard
$ID_j(t)$	L'ensemble des nœuds de passage constituant l'itinéraire de $V_j$ à l'instant $t$
$ID_{q,j}$	Le nœud intermédiaire numéro $q$ de $V_j$ sur son chemin
$N_{ID}$	L'ensemble des nœuds relatifs aux repères (adresses de passage) des véhicules
$I(t)$	La représentation matricielle regroupant l'ensemble des offres $O(t)$ des véhicules
$r_{O,D}$	L'élément d'indices $O$ et $D$ de $I(t)$ . Il regroupe les différents paramètres des offres existantes sur la route $[O \rightarrow D]$
$l_{O,D}^j$	Le nombre de places disponibles dans $V_j$ sur la route $[O \rightarrow D]$

	$d_{O,D}^j$	Le temps de départ au plus tôt de $V_j$ au départ de la section de route $[O \rightarrow D]$
	$a_{O,D}^j$	Le temps d'arrivée au plus tard de $V_j$ à l'extrémité finale de la section de route $[O \rightarrow D]$
<b>Itinéraires</b>	$IT_{V_j}(t)$	L'itinéraire restant à parcourir à $V_j$ à l'instant $t$ , il est constitué de l'origine de son déplacement, de $ID_j$ et de sa destination
	$PI_s^j$	La section de route (Partie d'Itinéraire) $s$ composant l'itinéraire complet de $V_j$
	$\oplus$	Opérateur de conjonction de sections de routes pour constituer un itinéraire (chemin) global
	$PI_s^+$	Origine de la section (Partie d'Itinéraire) $s$
	$PI_s^-$	Destination de la section (Partie d'Itinéraire) $s$
	$[O \rightarrow D]$	Section de route commençant à $O$ et se terminant à $D$
<b>Solution</b>	$IT_{U_p}$	Le chemin solution proposé à $U_p$ pour sa demande $R_p(t)$
	$S_{U_p}(t)$	Solution fournie à l'utilisateur $U_p$ en réponse à sa requête $R_p(t)$
	$PI_{w,p}$	L'itinéraire partiel (tronçon de route) numéro $w$ composant l'itinéraire global $IT_{U_p}(t)$ de l'utilisateur $p$

	$S_{w,p}$	Solution partielle composante de $S_{U_p}$ relative à $PI_{w,p}$ et fournie à $U_p$
	$PI_{w,p}^j$	Le véhicule $V_j$ est affecté à la partie d'itinéraire $PI_{w,p}$ relativement à la solution partielle $S_{w,p}$ composant la solution $S_{U_p}(t)$ fournie à $U_p$
	$O$	Opérateur de concaténation de solutions partielles pour en composer une globale à fournir à l'utilisateur concerné
	$V_{w,p}$	Ensemble des véhicules traversant la route $w$ et satisfaisant toutes les contraintes de correspondance sur celle-ci (correspondant à des offres faisables) relativement à la demande de $U_p$
	$TD_{PI_{w,p}}^j$	Durée globale (i.e. Trip Duration) nécessaire pour traverser $PI_{w,p}$ en empruntant le véhicule $V_j$ .
<b>Optimisation</b>	$TD_{U_p}^*$	Durée globale optimisée de $IT_{U_p}(t)$
	$TD_{PI_{w,p}}^*$	Durée globale optimisée sur $PI_{w,p}$ (avec l'affectation du véhicule qui offre de passer le moins de temps tout en satisfaisant les contraintes de correspondance et en évitant les conflits).
	$W_{j,w}$	Temps passé à attendre $V_j$ pour qu'il parvienne au départ de la section $PI_w$
	$T_{j,w}$	Temps de parcours nécessaire à $V_j$

		pour parcourir $PI_w$
	$\omega_{1_p}$	Ordre de préférence de $U_p$ sur le critère de temps d'attente ( $W$ ).
	$\omega_{2_p}$	Ordre de préférence de $U_p$ sur le critère de temps de parcours ( $T$ )
<b>Expressions et Evaluations</b>	$TAE(O, D, t)$	Temps d'Arrivée Estimé à $D$ à partir de $O$ à l'instant $t$
	$\tau(O, D)$	Retard maximum possible toléré entre une origine $O$ et une destination $D$
	$Dis\ tan\ ce(O, D)$	Distance entre deux points GPS ( $O$ et $D$ )
	$VMP(O, D, t)$	Vitesse moyenne de Parcours d'un segment de route $[O \rightarrow D]$ à un instant $t$ donné
	$SRT$	Seuil de Retard Tolérable
	$d_o^j$	Le temps de départ au plus tôt de $V_j$ à partir de $O$ (temps d'arrivée au plus tôt à $O$ )
	$a_D^j$	Le temps d'arrivée au plus tard de $V_j$ à $D$
<b>Processus de Algorithme 1 Décomposition</b>	$SN$	Ensemble de nœuds sélectionnés pour le traitement
	$VN$	Ensemble de Nœuds Visités
	$Diamètre$	Diamètre des zones à tracer
	$ZP$	Zone Primaire
	$N$	Tableau contenant les ensembles de nœuds voisins identifiés

	$N_a$	Nœud sélectionné aléatoirement
	$Centre(N[i])$	Fonction pour calculer le centre $C_i$ d'une zone couvrant l'ensemble des nœuds voisins stockés dans l'élément du tableau $N[i]$
	$Etablir\_Zone\_Primaire(C_i, Diamètre)$	Identification et construction d'une zone $PZ_i$ sur la base du centre $C_i$ établi pour un ensemble de nœuds voisins $N[i]$ . Cette fonction définit le périmètre d'une zone.
<b>Algorithme 2</b>	$ZI$	Ensemble de Zones Intermédiaires
	$Plus\_Grande\_Distance(N_a, N[i])$	Retourne la plus grande distance séparant $N_a$ de l'ensemble des nœuds (nœud le plus éloigné) dans $N[i]$
	$Affecté$	Permet de vérifier l'état d'un nœud, s'il a déjà été affecté à une zone ou pas encore
<b>Algorithme 3</b>	$A3D$	Algorithme Distribué Dynamique basé sur Dijkstra
	$TPM$	Temps de Parcours Moyen
	$Poids_{AO_i}$	Un tableau à double dimension stockant pour chaque nœud traité par un $AO_i$ la pondération minimale qu'il a réussi à lui affecter en partant de la source incluse dans la zone dont il est responsable et en fonction de son prédécesseur
	$Voiture_{AO_i}$	Tableau des affectations (i.e. véhicules) réalisées par l' $AO_i$ pour chaque

## Glossaire

	tronçon représenté par un élément du tableau
$OS_p$	Solution optimisée pour la requête $r_p$
$Localiser(D)$	Identifie la zone à laquelle appartient le nœud $D$

## INTRODUCTION GENERALE

Dans l'optique de répondre aux besoins d'innovation tout en demeurant dans un cadre respectueux des conditions environnementales fondamentales pour un climat sain, la voiture partagée se présente comme la solution clé pour une évolution sans conséquences néfastes en plus d'être d'ordre économique très avantageux.

Les outils mettant en place de tels services font ainsi l'objet d'intérêt de plusieurs industriels aussi bien que des chercheurs. Depuis plusieurs décennies déjà, le thème de l'innovation propre est placé en pleine ligne de mire par les gouvernements à l'échelle internationale vu la menace que traîne avec lui tout succès technologique atteint. En effet, sous l'ombre de l'avancement réalisé sur le plan du modernisme se cachent des méfaits et impacts péjoratifs majeurs touchant la qualité de la vie ainsi que l'équilibre biologique et environnementale nécessaire à la survie des êtres. Sous la lumière des sommets mondiaux nommés « *sommets de la terre* » dont les thématiques principales comprennent entre autre le développement durable comme point d'intérêt majeur, plusieurs études et travaux de recherche ont épuisé leurs efforts à faire émerger des énergies renouvelables et des technologies vertes. Dans cette lancée, des efforts ont été déployés pour atténuer l'impact négatif de la voiture personnelle : acteur pivot dans les taux d'émission de CO<sub>2</sub> et gaz à effet de serre. Les émissions de dioxyde de carbone sont d'une ampleur remarquable et vont crescendo avec l'évolution technologique; elles constituent de ce fait un facteur loin d'être négligeable dans la dégradation des conditions de vie.

Afin de soigner l'image environnementale du véhicule particulier, les systèmes d'Ecopartage sont nés. Dans ce contexte, le covoiturage en particulier a connu un succès notable grâce aux contributions qu'il apporte réduisant principalement le nombre de voitures en circulation. En effet, faisant de la voiture personnelle un mode de transport en commun, le covoiturage participe amplement à la réduction des taux d'émission de gaz nuisibles. Ses apports sont ainsi quantifiables en termes de CO<sub>2</sub> « non émis », conjointement aux multiples avantages qu'il offre aussi bien sur le plan individuel que collectif (e.g. réduction des budgets alloués au transport, flexibilité spatio-temporel, confort, équilibre social, etc.). Il a ainsi fait d'emblée son entrée dans le domaine de la recherche et de multiples systèmes ont vu le jour. Plusieurs études ont ainsi été menées dans ce sens puisant dans les domaines de

l'informatique, de l'intelligence artificielle, des GIS (Geographical Information Systems), de l'Internet ainsi que des domaines des télécommunications, etc.

Usant des technologies nouvelles, la mise en place d'un système plus ou moins évolué prime sur tout autre objectif au niveau des approches existantes. Celles-ci se sont en effet, pour la plupart, penchées sur l'adoption de la technologie (i.e. réseau bluetooth, intégration des PDAs, etc.) et délaissent les aspects d'efficacité des traitements (i.e. automatisation, optimisation, temps réel, aide au déplacement, etc.). Des supports en ligne hébergés sur la toile sont aujourd'hui opérationnels et permettent au grand public de s'inscrire et bénéficier de services assez restreints tels que les publications et consultations d'offres et demandes ainsi que la récupération des coordonnées des covoitureurs pour une éventuelle prise de contact. Malheureusement, ce type de systèmes est le seul qui mérite d'être évoqué puisque les autres, malgré leur ouverture sur des fonctionnalités avancées telles que l'intégration du temps réel, l'automatisation des tâches d'affectation basés sur des systèmes multi-agents, etc. sont restés au stade de « *l'idée* » ou de « *l'ébauche* » tout en n'étant pas passible d'amélioration. Irrévocablement lancés sur la voie de l'amélioration, nous proposons dans le cadre de notre travail de thèse de mettre en place un système de covoiturage dynamique optimisé. Aspirant à l'élaboration d'un système opérationnel, performant et compétitif à grande échelle, nous nous sommes focalisés sur la notion d'optimisation en usant des méthodologies d'intelligence artificielle distribuée. En effet, un problème de covoiturage dynamique optimisé instaure la qualité de service requise tout en faisant émerger la problématique de la complexité exponentielle. C'est dans ce cadre que logent nos principales motivations faisant ainsi le fondement de notre approche. Une approche basée sur le principe de l'optimisation distribuée pour abolir la difficulté de résolution du problème. Pour ce faire, le système proposé se doit de mener en parallèle plusieurs processus indépendants chargés chacun d'exécuter une tâche partielle du problème initial dans un contexte de « *multi-threading* ». Deux concepts principaux méritent ici d'être évoqués en particulier ; à savoir, la modélisation du problème sous forme de graphe dynamique distribué à l'image de laquelle une architecture logicielle distribuée est établie et le déploiement d'une multitude d'entités autonomes conformément à cette architecture. Le paradigme agent aidant, les entités sont dotées d'une aptitude à mener d'écho et de manière fiable et synchrone des processus parallèles et décentralisés, à priori portant à conflit et sujets à d'importantes ambiguïtés. Faisant profit de l'autonomie des agents pour une prise de décision intelligente ainsi que de leurs capacités à communiquer via l'échange de messages, le procédé d'affectation optimisée des véhicules aux usagers

demandeurs de trajets de covoiturage est ainsi mené de manière décentralisée. Autonomie et communication vont ainsi de pair pour réaliser la décomposition du processus global en plusieurs sous processus moins complexes menés en parallèle dans une synchronisation totale grâce au concept de coalition inter-agent. Ceci nous a permis de réaliser notre objectif premier qui est celui de répondre à un maximum de demandes efficacement en fournissant des solutions optimisées dans un délai raisonnable tout en demeurant hors du piège des affectations conflictuelles.

L'alliance des systèmes multi-agent avec les fondements de l'optimisation a ainsi été mise au service de l'efficacité des traitements sous tous les angles de vue. L'efficacité du système mis en place touche en effet aussi bien à la qualité des réponses fournies (i.e. satisfaction des usagers, qualité et personnalisation du service, etc.) qu'aux propriétés d'exécution assurant ses performances techniques (i.e. délais de réponse, utilisation des ressources CPU, espace mémoire, etc.).

Ce rapport se subdivise en cinq grandes parties : la première a pour objectif d'introduire le covoiturage qui, assujetti à un descriptif détaillé d'un plus global, est présenté comme partie intégrante du phénomène de la voiture partagée. Celle-ci est venue apporter la solution à plusieurs problèmes qui ont constitué la première partie de ce chapitre. Par la suite, un état de l'art des systèmes de covoiturage existants avec un focus sur leurs limites et lacunes nous a amenés à considérer nos principales motivations pour traiter ce problème. Notre problématique ainsi définie, nous nous focalisons dans le deuxième chapitre sur la méthodologie de résolution à adopter et qui se présente sous forme d'une alliance des systèmes multi-agents avec les fondements de l'optimisation pour la mise en place d'une approche s'inscrivant dans le cadre d'une intelligence artificielle distribuée. Nous mettons ainsi à profit les aspects positifs des deux concepts faisant régner une intelligence ambiante pour l'exécution des traitements dispatchés sur un ensemble d'entités. Ce dispatching est réalisé conformément à une modélisation sous forme de graphe dynamique distribué. Cette modélisation fait l'objet du troisième chapitre où le problème d'optimisation ainsi que la structure choisie ont été définis formellement. Les spécifications formelles et algorithmes de décomposition aboutissant à une telle modélisation sont aussi décrits dans ce chapitre. Dans la quatrième partie, nous présentons l'architecture de notre système en se basant essentiellement sur le paradigme agent. Les agents composant cette architecture ainsi définis, nous en détaillons les comportements sur la base de l'analyse fonctionnelle de notre problème faite dans un premier volet de ce même chapitre. Puis, sont présentés en dernier lieu les

## Introduction Générale

algorithmes d'affectation des véhicules aux usagers appuyés du concept de l'optimisation distribuée. Dans le cinquième et dernier chapitre, les architectures logicielles et physiques de déploiement de notre application sont présentées avec les outils utilisés. Aussi différents jeux de tests ainsi que les calculs associés sont fournis dans cette dernière partie.

## I. CHAPITRE I

### INTRODUCTION AU CONTEXTE

### LE COVOITURAGE A LA VEILLE D'UNE PRATIQUE DE TRANSPORT COMODAL

## I.1. Introduction

Organisés par l'ONU chaque dix ans, les sommets de la terre sont des rencontres où des dirigeants mondiaux s'associent pour la définition des buts à atteindre dans le cadre de la réalisation d'un développement durable à l'échelle internationale. Les sommets de la terre tentent de concrétiser l'intérêt porté par des gouverneurs mondiaux visant entre autre à mettre en place une politique énergétique propre.

A l'instar du Programme des Nations Unies pour l'Environnement (PNUE) abouti de par le sommet tenu en 1972 ou de la Convention-cadre des Nations Unies sur les Changements Climatiques (CCNUCC) lancée lors du sommet de 1992<sup>1</sup>, ces rencontres ont pour objectif premier d'instaurer des programmes pour soigner l'image environnementale de par le monde. Une image ternie par le progrès et l'avancement technologique réalisés, preuve du modernisme et du pic de développement atteint.

Les rencontres organisées à l'occasion des sommets de la terre placent, de ce fait, aux premiers rangs des préoccupations écologiques touchant l'environnement et le climat et fixent les objectifs à accomplir pour y remédier. Ces objectifs ont donné lieu à l'élaboration de programmes collectifs rassemblant plusieurs états et faisant ainsi la preuve du développement d'une culture mondiale de respect de l'environnement. Ceci nous mène à considérer les causes ayant engendré de tels effets sur la détérioration de l'environnement pour essayer de considérer le problème à partir de ses origines.

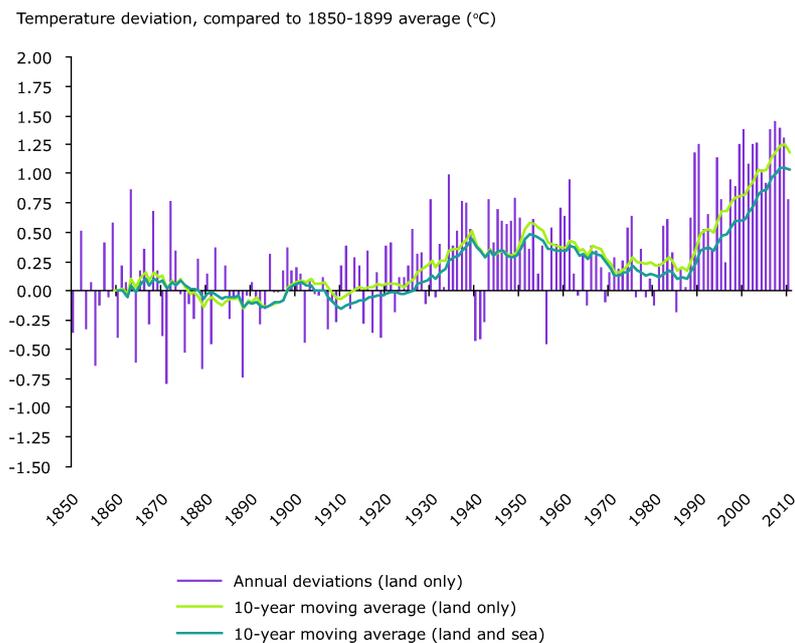
### I.1.1. Développement durable : à l'encontre des menaces sur l'environnement

Le nombre important des menaces affluant sur la terre et l'environnement ont engendré des problèmes d'une incidence assez grave sur les êtres vivants. L'impact péjoratif de ces problèmes touche principalement l'équilibre environnemental, le climat, la qualité de la vie, etc. Le réchauffement climatique est entre autre à l'origine de ces menaces engendrant un déséquilibre biologique et écologique tel que la fonte des glaciers par exemple. La Figure I.1 montre la déviation des températures moyennes entre 1850 et 2010 en Europe. Les données d'origine pour cette étude ont été extraites du Climatic Research Unit of the University of East Anglia<sup>2</sup>.

---

<sup>1</sup> <http://www.unep.org/>

<sup>2</sup> <http://www.cru.uea.ac.uk/cru/info/warming/>



*Figure I.1. Evolution de la moyenne de la température entre 1850 et 2010*

L'une des sources les plus importantes de ces problèmes se trouve être le taux élevé d'émission de CO<sub>2</sub> et gaz à effets de serre. Les trous aujourd'hui existants dans la couche d'ozone et qui s'élargissent de plus en plus témoignent du danger qui subsiste et rappellent en permanence les risques encourus. Le changement climatique et les catastrophes naturelles de plus en plus présentes sur terre et dont l'une des principales causes est l'existence de ces trous invoquent le besoin de trouver dans l'immédiat des solutions pour essayer de rétablir l'équilibre naturel dans lequel nous vivions.

Les dégâts humains et matériaux importants engendrés de par ce déséquilibre écologique ont ainsi tourné l'œil du monde entier vers l'ampleur de ces phénomènes naturels incitant les dirigeants à mettre en place les dispositifs nécessaires pour rompre cet enchaînement de mouvements naturels destructifs. La Figure I.2 classe l'ensemble de ces événements suivants leurs différents types en définissant pour chacun la proportion qu'il représente dans la totalité des mouvements naturels et en montrant la croissance importante perçue entre les années 1980 et 2009 pour chacun des types identifiés.

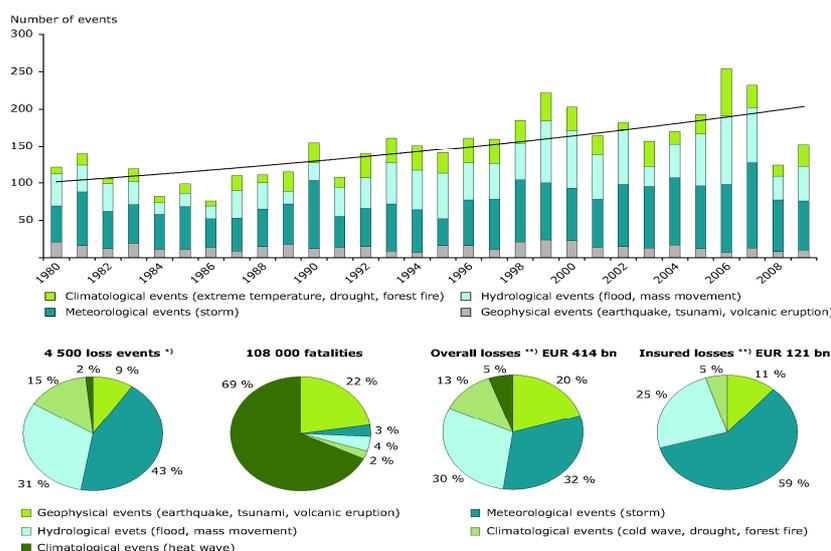


Figure 1.2. Evolution des catastrophes naturelles

Des solutions pour essayer de prévenir de telles catastrophes accaparent l'intérêt des responsables concernés et la recherche a pris une place de plus en plus encombrante dans plusieurs domaines. Allant de pair avec le secteur industriel, les recherches élaborées dans ce sens ont abouti à des énergies propres, des technologies vertes, des produits et des systèmes novateurs rivalisant d'originalité et de performance, et ce dans plusieurs secteurs.

### I.1.2. Transport : son rôle dans les menaces sur l'environnement

Plusieurs chercheurs et praticiens se sont particulièrement intéressés aux problèmes de transport. Ce domaine en particulier se trouve être un facteur pivot dans l'importance des rejets de dioxyde de carbone et émissions de gaz à effet de serre qui impliquent des conséquences péjoratives sur la qualité de la vie. En effet, les données recensées par l'Agence pour l'Environnement de l'Union européenne<sup>3</sup> en 2006 révèlent que pratiquement le quart (23,8%) des émissions totales de gaz à effet de serre et légèrement plus d'un quart du total des émissions CO<sub>2</sub> (27,9%) dans l'UE-27 sont dus aux engins utilisés dans les transports. Ce taux n'a pas considérablement changé malgré les mesures et dispositions prises dans ce sens quelques années auparavant. En effet, le secteur des transports symbolisait le principal acteur dans les émissions de gaz à effet de serre et était responsable de 28% des rejets de CO<sub>2</sub> dans l'atmosphère à l'échelle européenne en 2003 et 35% à l'échelle nationale (1). Ceci étant, il devient de plus en plus urgent d'essayer de remédier à ce problème vu la forte croissance de ces taux.

<sup>3</sup> <http://www.eea.europa.eu/fr>

La Figure I.3 (2) montre l'évolution des émissions de CO<sub>2</sub> et Gaz à Effet de Serre (GES) en Europe. Les courbes tracées au niveau de ce graphique sont relatives à l'évolution de ce taux par secteur d'activité entre les années 1990 et 2006. Un domaine en particulier qui est celui du transport attire l'attention de par l'ampleur des émissions engendrées par celui-ci ainsi que de par l'évolution croissante et très rapide des taux de rejets de CO<sub>2</sub> et GES. Cette croissance rapide et très importante suscite des inquiétudes vu l'aptitude destructive de tels rejets sur l'environnement inhibant de ce fait le processus de développement durable escompté (3).

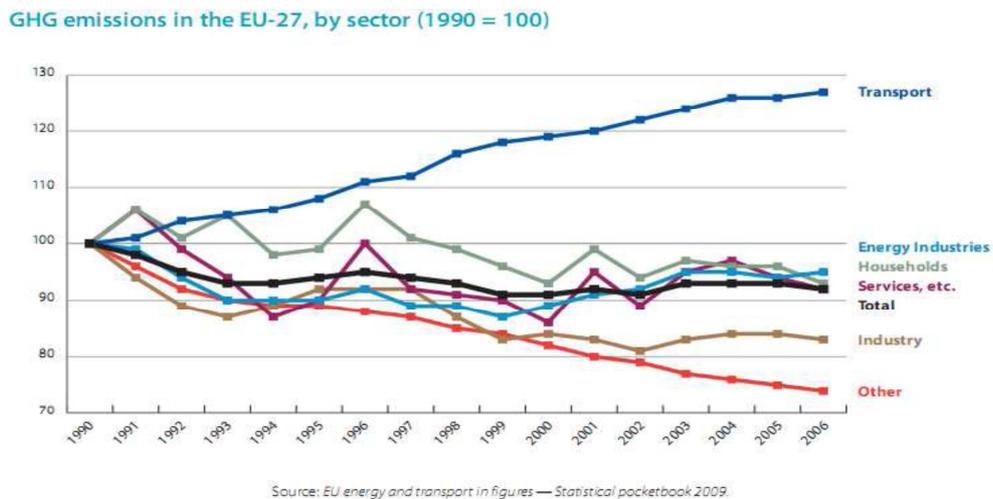


Figure I.3. Taux de rejet de CO<sub>2</sub> et GES en Europe par secteur de service

La Figure I.4 montre l'évolution des rejets de CO<sub>2</sub> quantifiés en tonnes et en pourcentage entre 1990 et 2008. Il s'agit d'une évolution par secteur qui montre une croissance ou bien une régression des taux de GES émis pour chacun des secteurs représentés.

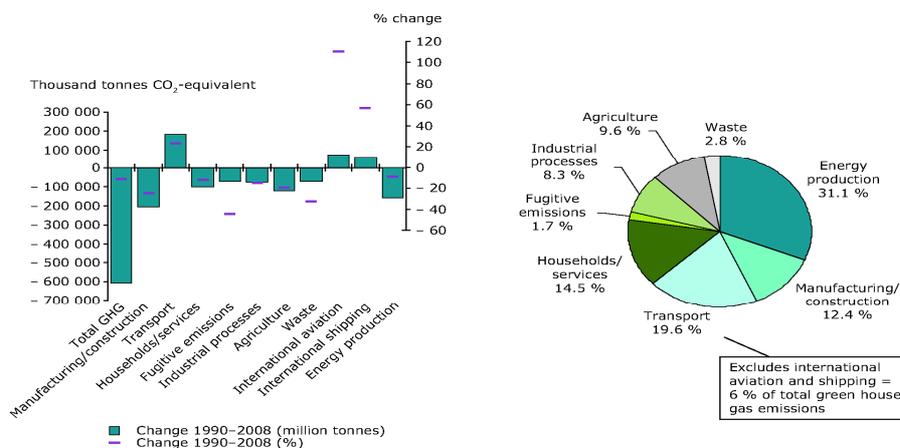


Figure I.4. Variation des émissions de CO<sub>2</sub> en EU-27 par secteur entre 1990 et 2008<sup>4</sup>

<sup>4</sup> <http://www.eea.europa.eu/data-and-maps/figures/greenhouse-gas-emissions-in-the>

En outre, malgré l'amélioration perçue à travers le monde relativement à la régression du taux d'émission de CO<sub>2</sub>, le secteur de transport est malheureusement resté hors d'atteinte de cette amélioration. La Figure I.5 montre en effet une dégradation des émissions polluantes de l'année 1990 à l'année 2009 alors que la Figure I.6<sup>5</sup> met l'accent sur la croissance continue et aussi importante de ces émissions pour le secteur du transport contrairement à tous les autres.

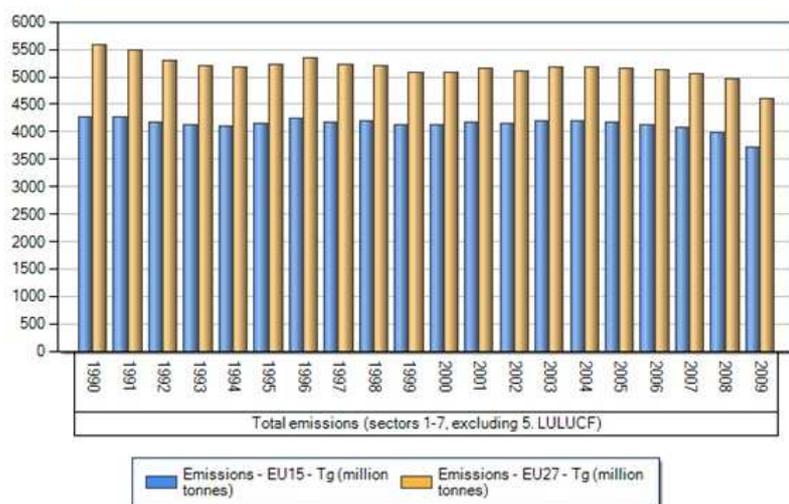


Figure I.5. Variation des émissions polluantes entre 1990 et 2009

En effet, qu'il s'agisse de transport terrestre, maritime ou encore aérien, le secteur du transport se distingue de par les chiffres des taux d'émissions de GES qu'il affiche. La Figure I.6 montre le changement de ces taux en Europe par secteur d'activité en comparaison de ceux émis en 1990. L'histogramme illustré dans cette figure met l'accent sur l'importance des rejets dus aux moyens de transport dans toutes ses formes.

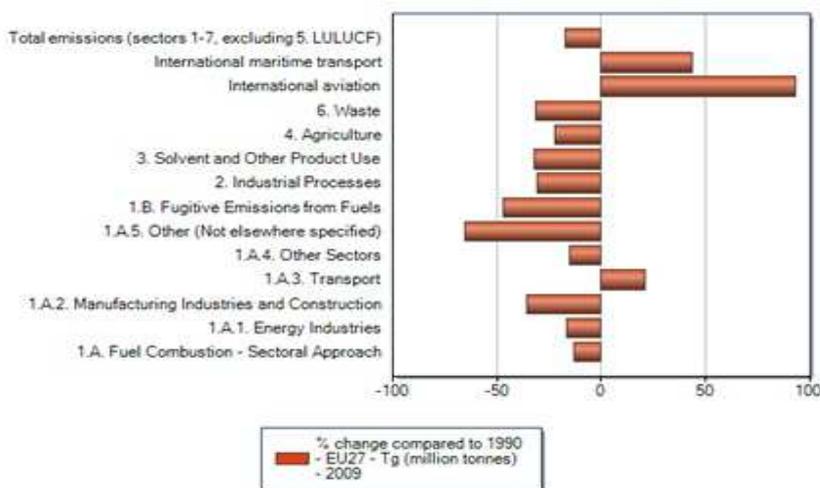


Figure I.6. Variation en pourcentage du taux d'émission de CO<sub>2</sub> en EU-27 par secteur

<sup>5</sup> <http://www.eea.europa.eu/data-and-maps>

### I.1.3. L'évolution du transport

L'étude montrée préalablement ne peut témoigner que de l'évolution rapide du secteur de transport. En effet, depuis plus de trois décennies déjà, le déplacement en interne des voyageurs, en France, mais aussi dans les différents pays de l'Europe, n'arrête pas d'augmenter. L'allongement des distances parcourues quotidiennement, qu'il s'agisse de déplacements dans le cadre de trajets quotidiens et fréquents (e.g. pour rejoindre leurs domiciles ou travaux) ou dans les trajets touristiques, est en continuelle progression. Cette évolution est illustrée au niveau de la Figure I.7 où est représentée une superposition de deux diagrammes comptabilisant le nombre de « voyageurs-kilomètre » en France et pour toute l'Europe. Pour le cas de la France, nous pouvons remarquer de par ce graphique, que l'allongement des déplacements a été marqué par une certaine stabilité. « voyageur-kilomètre » est l'unité de mesure proposée par l'INSEE équivalente au transport d'un voyageur sur un kilomètre et ce afin de pouvoir repérer l'évolution des voyageurs (2).

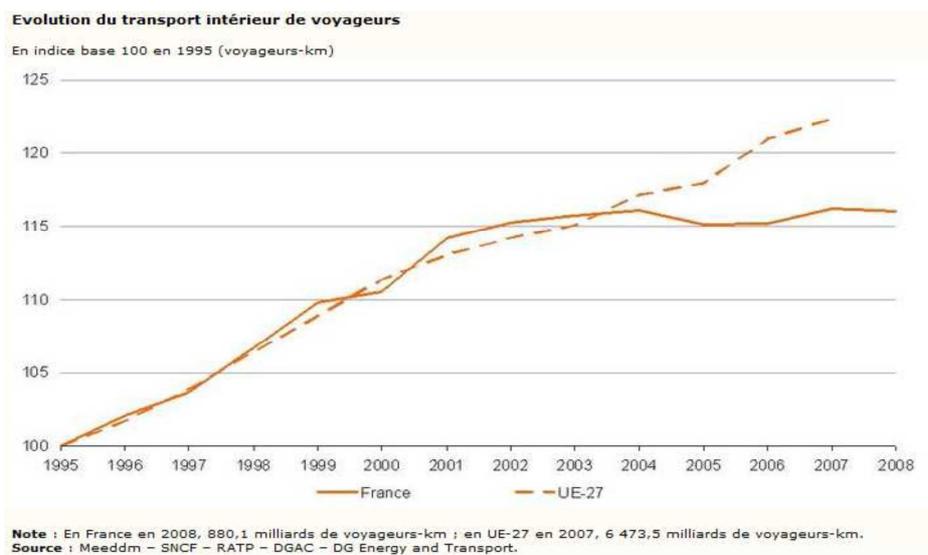


Figure I.7. Évolution du nombre de voyageurs au kilomètre en interne en France et en Europe

Cette évolution ne peut que confirmer le succès perçu dans le domaine du transport et proportionnellement la croissance des impacts péjoratifs qu'il engendre faisant s'insurger les organismes concernés par le processus de développement durable.

A cela s'ajoute la consommation importante d'énergie dans ce secteur (Figure I.8). En effet, le transport s'appuie à 97% sur l'énergie fossile (3) et accapare à ce titre le plus grand pourcentage (50%) de consommation de produits pétroliers en l'année 2002.

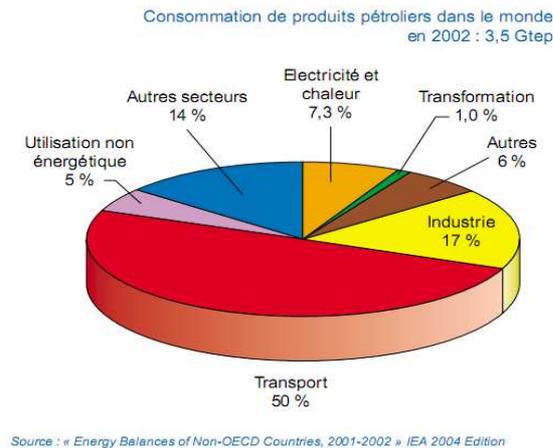


Figure I.8. Consommation de produits pétroliers par secteur

### I.1.4. Vers une mobilité durable et soutenable

A l'instar de *CIVITAS* (*City-VITALity-Sustainability*), programme européen pour encourager de nouvelles pratiques de déplacement ou encore de *PUMAS*<sup>6</sup>, projet de recherche pré-industrielle sur la communauté de l'agglomération de Rouen Elbeuf Austreberthe, plusieurs initiatives ont pour but essentiel de mettre en place des politiques innovantes et décisives dans le domaine du transport. Il en existe aussi plusieurs visant à développer des plateformes logicielles d'aide à la décision offrant des solutions économiques et flexibles. Ces plateformes offrent donc des solutions de développement de mobilité durable dans le sens où elles proposent des solutions de facilité avec les connaissances requises (i.e. informations en temps réel sur les conditions du trafic, mesures des émissions des gaz nocifs...). Aussi dans le cadre de projets innovants de mobilité avancée, certains organismes proposent d'aider les voyageurs dans leurs déplacements sur la base de calculs des temps de parcours par exemple. Leurs retombées dans ce contexte peuvent concerner une gestion plus efficace des déplacements, une maîtrise de l'impact environnemental, une incitation à l'usage des transports en commun ou encore la minimisation des coûts, etc.

Les efforts déployés dans ce sens se sont orientés vers les technologies vertes, les modes de transport propres tout en essayant de demeurer dans le cadre douillet des services de transport à la demande ou encore la voiture personnelle alliant confort et flexibilité. Ce compromis a été vu en le concept de la voiture partagée qui a marqué l'aire du développement durable et de la mobilité avancée. Chercheurs et industriels ont ainsi collaboré pour faire immerger des projets d'innovation dans ce contexte et mettre en pratique les idées énoncées de par ces

<sup>6</sup> [http://pumas.inria.fr/\\_media/public/pumas\\_20100408.pdf](http://pumas.inria.fr/_media/public/pumas_20100408.pdf)

projets alliant de ce fait le pragmatisme aux idéologies théoriques. Ce faisant, plusieurs pratiques sont nées et des systèmes offrant des services plus ou moins évolués rivalisent d'originalité et de performance (*c.f. I.5*).

Les problèmes de transport, aujourd'hui partiellement résolus, demeurent de grande envergure. La mobilité avancée est ainsi placée à un stade probatoire quant aux domaines de l'industrie ainsi que de la recherche<sup>7</sup>.

Dans nos travaux, nous abordons le problème de la voiture partagée et plus particulièrement le concept de covoiturage. Après avoir traité la problématique principale touchant le domaine du transport dans son contexte environnemental, ce premier chapitre fait l'objet d'une présentation des principaux modes de transport les plus émergents avec un focus sur le covoiturage en particulier. Les principales études élaborées dans le cadre de chacun des modes ainsi définis sont par la suite détaillées tout au long de ce chapitre. Nous présentons aussi une liste non exhaustive des systèmes existants, élaborés pour répondre à une problématique particulière énoncée dans chacun de ces modes, notamment et principalement les systèmes automatisés intégrant des supports informatiques.

## **I.2. Transport : Les différents moyens**

Les moyens utilisés pour réaliser le transport d'un bien ou d'une personne diffèrent. Plusieurs éléments et facteurs font l'explosion de cette variabilité des modes et moyens de locomotion inspirée des besoins de déplacement différant selon un contexte bien déterminé et selon le besoin.

### **I.2.1. Le transport classique**

Le transport classique implique le fait de considérer les modes de transport selon leur contexte. Nous pouvons considérer dans ce cadre les modes de transport organisé par une collectivité publique, c'est-à-dire un organisme, une société, ou un groupe bénéficiant d'une identité et un lieu d'être connus. Visant une classification de ces organisations, nous considérons plutôt la classification des moyens de transport. Vu la panoplie de modes de transport existants aujourd'hui, nous distinguons les transports publics et les transports privés, qui selon les textes européens<sup>8</sup> diffèrent de par le fait qu'ils soient organisés par une personne

---

<sup>7</sup> <http://www.iisd.org/>

<sup>8</sup> <http://www.developpement-durable.gouv.fr/Les-transports-publics-et.html>

pour le compte d'autrui, et c'est là le cas des transports publics. Par ailleurs, les transports privés sont organisés par une personne pour son propre compte.

### *1.2.1.1. Les modes de transport public*

« Sont considérés comme des transports publics tous les transports de personnes ou de marchandises, à l'exception des transports qu'organisent pour leur propre compte, des personnes publiques ou privées »<sup>9</sup>. Ainsi, les transports publics sont ceux organisés pour le compte de la collectivité. Il s'agit de ce fait de services offerts à la place ouverts au grand public et dont toutes les informations sont connues à l'avance et disponibles à tout le monde. En effet, dans le cas des services publics réguliers de transport routier de personnes, les trajets (les itinéraires), les points d'arrêt, les fréquences, les horaires et les tarifs sont fixés et publiés préalablement. Ce mode de service public implique toutes les dessertes régulières urbaines et extra-urbaines. Il inclue donc tous les moyens de transport en commun dont nous pouvons citer à titre d'exemple les métros, bus, trains, tramway, les services réguliers publics tels que les transports scolaires, etc.

Nous distinguons aussi dans cette catégorie deux services particuliers de transports :

***Le transport à la demande (TAD)*** : il s'agit des services collectifs déterminés en fonction de la demande des usagers. Leurs règles générales de tarification sont connues au préalable, par contre leurs itinéraires (origines et destinations) et fréquences sont établis en fonction de la demande lorsqu'elle existe (4). Nous pouvons citer dans ce cas, la desserte d'un marché, le transport des personnes handicapées, les urgences (5)... les véhicules utilisés dans ce contexte sont de capacité minimale fixée à quatre places incluant celles du conducteur. Ceci étant, les services publics réguliers et les services publics à la demande de transport routier peuvent considérer de manière spécifique des catégories particulières de personnes : personnes âgées, élèves et étudiants handicapés, personnes handicapées... Ainsi, dans le cadre des problèmes de tournées de véhicules (VRP : Vehicle Routing Problem) (6), un problème particulier à citer est celui relatif au ramassage et dépôt d'objets ou de personnes (Pickup and Delivery Problem : PDP). Cette classe de problèmes de tournée de véhicules consiste à transporter des « charges » d'une origine à une destination données sans transbordement ni emplacements intermédiaires (7). Des variantes des PDPs existent et nous en citons principalement trois catégories distinctes selon qu'il s'agisse de problèmes de *plusieurs-à-plusieurs* tels que le « *Swapping Problem SP* » (8) ; de problèmes de *un-à-plusieurs-à-un* où des charges existent

---

<sup>9</sup> Selon le Ministère de l'écologie, du développement durable, des transports et logement

initialement dans un entrepôt doivent être livrées aux clients et inversement (les charges chez les clients sont destinées à un entrepôt) ; ou de problèmes de un-à-un tels que le transport porte-à-porte, de courrier au autre, et où sont considérés les transports d'une origine à une destination bien déterminées.

Des travaux ont été élaborés dans ce contexte pour essayer de résoudre les différents types de PDPs qu'ils soient statiques (9) ou dynamiques (10). Ceux-ci, contrairement aux premiers, se doivent de gérer en temps réel les données en fonction desquelles les trajets des véhicules seront tracés.

Plus particulier encore est le problème de transport de personnes, tel que par exemple le transport de personnes âgées ou handicapées (Dial-A-Ride Problem : DARP) qui s'inscrit plutôt dans le contexte des PDPs dynamiques. Les modèles et algorithmes relatifs au DARP ont été étudiés dans (11).

***Le transport occasionnel*** : n'a lieu que lors d'occasions particulières (e.g. organisation d'une colonie de vacances, une visite touristique, un évènement spécifique : concert, exposition...). Cette catégorie se dissocie en deux types principaux, à savoir les circuits à la place et les services collectifs. Le premier type concerne les voyages organisés dont les places sont vendues séparément et ramènent généralement les voyageurs à leur point de départ, alors que le deuxième concerne la mise à disposition exclusive d'un ou plusieurs groupes connus, constitués à l'avance d'au moins dix personnes.

Par ailleurs, le concept de transport public ne peut se confondre avec la notion de service public de transport au sens du statut de la personne ou organe organisateur. C'est ainsi que seuls sont réellement considérés publics les services réguliers et les services à la demande de transports publics au sens de l'organisation et de la jurisprudence. C'est-à-dire les modes et pratiques qui relèvent de la notion juridique classique de services publics dans le sens où ils représentent un intérêt général ainsi que dans leur organisation et financement par une personne publique.

Dans leur diversité, les transports en commun fournissent aux usagers l'occasion de bénéficier de moyens de transport fiables, propres et économiques (i.e. de moindre coût). L'endoctrinement reçu sur les moyens de transport en commun n'a de fait que d'encourager les personnes à aller vers ces services. Cependant, les véhicules utilisés dans ce contexte restent d'usage restreint à une catégorie spécifique de personnes. En effet, l'usage des modes de transport collectifs reste le lot des personnes plus ou moins « *démunies* ». Par

« *démunies* », nous entendons une catégorie spécifique de personnes auxquelles le luxe de l'automobile privée est ôté. Cette catégorie rassemble ainsi les personnes de revenus limités, les étudiants, les personnes en recherche d'emplois ou à revenu modeste, etc. Ceci étant donné la difficulté d'acquisition et d'entretien d'une voiture pour les besoins de déplacements journaliers ou même de moindre fréquence. D'une manière générale, salariés, étudiants, personnes âgées ou autres préfèrent se déplacer dans leurs propres véhicules alliant confort et flexibilité au détriment de leurs portefeuilles.

Une étude récente vient confirmer cette hypothèse montrant que le transport automobile (voiture personnelle) reste le mode de transport le plus utilisé (12). Cette étude menée autour de la mobilité des français et recensant leur avènement vers les modes de transport, montre l'usage abusif de l'automobile particulier (de l'ordre de 81,8% du transport intérieur), pour ensuite se retourner vers les transports ferroviaires, les autobus et autocars (2). La Figure I.9 montre les différentes proportions relatives aux usages des différents modes de transport selon les statistiques relevées au terme de l'année 2006.

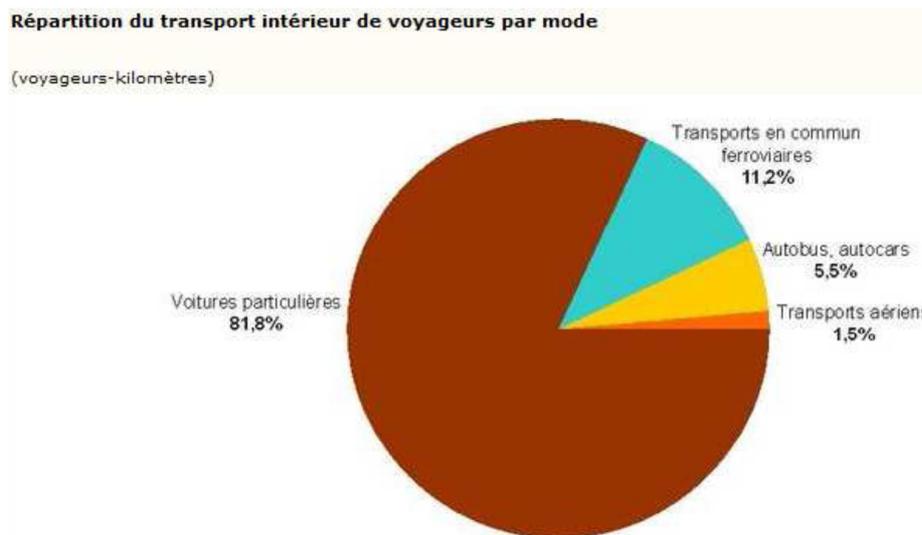


Figure I.9. Etude statistique de l'utilisation des français des modes de transports

### *1.2.1.2. Les modes de transport privés*

Parmi ces modes, nous distinguons quelques moyens de transport relatifs à la marche à pied, le vélo, l'automobile privée, etc. Il s'agit de moyens de transport privés utilisés par une personne pour son propre compte. Le mode le plus émergent dans ce cadre et relativement à ce qui précède se trouve être l'automobile (i.e. voiture particulière).

Parallèlement à l'étude menée en France, d'autres ont été faites sur l'appréhension de l'usage des véhicules particuliers de par le monde entier, démontrant ainsi le succès sans limite de ce mode. Des statistiques révélées dans (13) comptabilisent le nombre d'automobiles en l'an 2000 à 740 millions dans le monde. La plupart d'entre eux sont dans les pays industrialisés avec une moyenne énoncée de 500 voitures pour 1000 habitants et 1500 kilomètres de distance parcourue chaque année. Par ailleurs, cette prédominance de la voiture particulière n'est pas tout à fait nouvelle et ne représente pas un phénomène émergent dans la mesure où en 1999, Dupuy (14) annonçait déjà une dépendance aux transports automobiles.

En effet, grâce à ses multiples avantages : ses performances que ce soit en terme de vitesse, tenue de route, confort, sécurité et fiabilité ou encore aux grandes aptitudes d'adaptation qu'elle représente (déplacement sur une courte ou longue distance, en zone urbaine, péri-urbaine ou extra-urbaine...) ; l'automobile est devenue une nécessité dans la vie des gens. Par conséquent, joignant le confort à la facilité d'accès et à la flexibilité spatio-temporelle, les individus ont de plus en plus tendance à aller vers ce mode de transport. Un mode qui est devenu, avec le temps, nécessaire à leur équilibre psychologique et à leur tranquillité d'esprit (15). Ceci vient notamment en défaveur des autres modes de transport (essentiellement le transport en commun), pratique qui va en crescendo abolissant les projets de développement durable. En effet, malgré ses avantages déterminants allant jusqu'à modifier les comportements et modes de vie des individus (16), l'automobile privée empêche les perspectives de mobilité avancée et soutenable d'aller à bon escient. Elle se trouve de ce fait derrière bon nombre d'obstacles allant à l'encontre de la mise en place d'un processus de développement propre et durable.

### ***A. L'automobile : problèmes environnementaux***

L'usage abusif du véhicule privé engendre une consommation déroutante d'énergie extériorisée sous forme de CO<sub>2</sub> et GES. Par ailleurs, le regroupement d'une collectivité dans un seul moyen de locomotion (Bus, Train, Métro...) réduit largement les consommations et par la suite les expulsions en termes de gaz polluants dans l'air. Les transports en commun permettent ainsi de faire une réduction d'échelle et donc d'énergie.

La Figure I.10 en est la preuve tangible où, sur la base de chiffres extraits sur les rejets de CO<sub>2</sub> de par les différents modes de transports, des statistiques à l'origine de ces histogrammes révèlent que la voiture personnelle demeure le mode le plus polluant sur une plage de 18 ans (depuis l'année 1990 jusqu'en 2008).

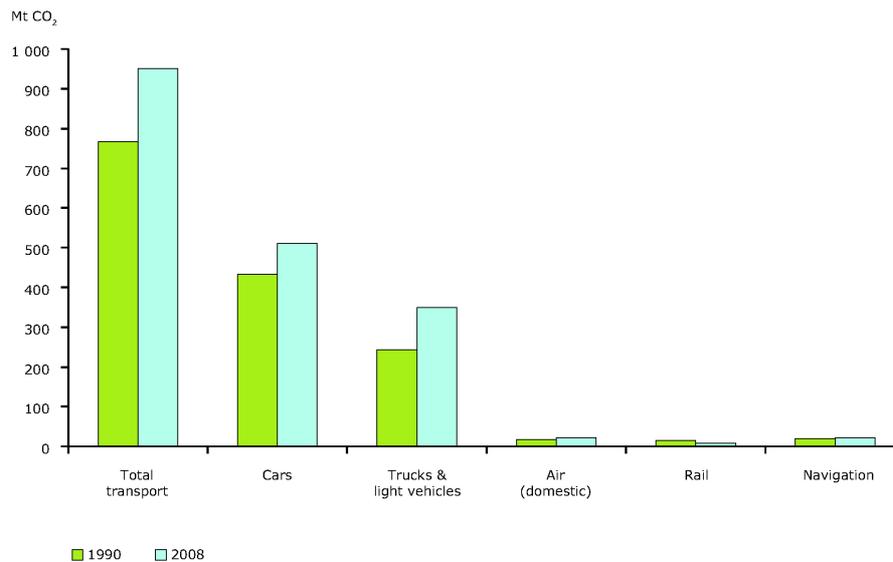


Figure I.10. Importance des taux d'émission de CO<sub>2</sub> par mode de transport

### ***B. L'automobile : problèmes de sécurité***

Le nombre important de voitures sur les routes engendre des impacts négatifs non seulement sur l'environnement mais aussi sur la fluidité des trafics créant de ce fait des bouchons et un encombrement pouvant être à l'origine d'accidents ou tout autre type d'incident. En effet, le nombre de morts du fait d'accidents de la route en union européenne était de l'ordre de 41000 individus en l'an 2000 avec plus de 1,7 millions de blessés. Le même ordre de grandeur a été observé aux états unis. Les pertes humaines et matérielles dues aux accidents de trafic routier sont de ce fait énormes et engendrent par ailleurs des coûts économiques très importants (13).

Parallèlement, les transports en commun aident à gratifier le transport routier de fluidité en transportant des nombres importants de personnes sur des voies spécifiques diminuant ainsi le nombre de voitures sur les routes. En effet, pour 60 personnes transportées par bus, la surface occupée ne dépasse pas celle destinées à supporter deux voitures. Alors qu'une voiture ne transporte en moyenne que 1,5 voyageur (2). Par ailleurs, les voies réservées destinées au transport loin du trafic routier aident considérablement à fluidifier la circulation et ainsi diminuer les accidents. Les transports en commun créent de ce fait un boom en matière de sécurité par rapport au véhicule privé.

### ***C. L'automobile : problèmes sociétaux***

Le seul hic dans ce qui précède, c'est la flexibilité aussi bien sur le plan spatial que temporel. En effet, sur ce plan spécifique de l'étude, la voiture personnelle se distingue nettement des transports en commun par son aptitude d'adaptation. Même si les transports en

commun peuvent s'avérer plus rapides que l'automobile privée quand les conditions sont favorables (e.g. réseaux équipés de voies réservées..) et beaucoup moins coûteux, ils n'ont toujours pas réussi à lui usurper sa place. En effet, il y a d'ores et déjà beaucoup de voitures en circulation dans le monde. De plus, l'usage de l'automobile a créé une sorte de dépendance chez les individus. L'adaptation des modes de déplacement et des organisations des infrastructures n'est pas loin d'être à l'origine de cette dépendance à l'automobile outre la facilité d'accès qu'elle représente. Toutefois, des problèmes d'ordre sociétal subsistent (bruit, stress, pollution, manque d'espace pour les piétons, rupture des liens sociaux...) principalement dus aux encombrements (embouteillage, difficulté de parking, etc.).

### I.2.2. Transport en commun Versus Automobile

La préservation de l'environnement de par la réduction des émissions de CO<sub>2</sub> et gaz à effet de serre perçue comme un leitmotiv dans les programmes de protection contre les menaces de changement climatique et déséquilibre écologique. Pour le domaine du transport en particulier, cet enjeu relève du défi vu l'afflux des individus sur les modes de transport individuels, essentiellement la voiture personnelle. L'usage abusif de la voiture vient ainsi à l'encontre des projets de réduction des rejets de gaz à effets de serre engendrés par le secteur d'activité relatifs aux transports. Les politiques de transport dans le monde, et spécialement en Europe, se sont ainsi focalisées sur la production de dispositifs encourageant les différents types de transports en commun. Cette propagande des modes collectifs serait en faveur de l'instauration d'un développement rentable et durable.

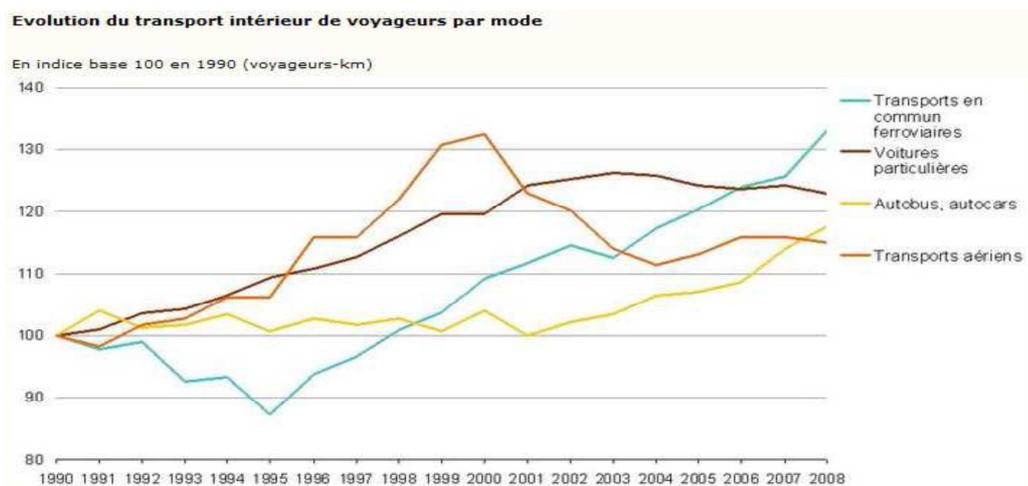
Sur la base de l'étude précédente, nous pouvons conclure que les transports en commun s'avèrent être très avantageux par rapport à l'automobile privée selon plusieurs angles de vue. Nous pouvons considérer notamment dans ce contexte cinq avantages majeurs tirés d'une plus large panoplie<sup>10</sup> qui sont ceux des transports en communs :

- La consommation d'énergie et pollution
- La fluidité du trafic routier
- Les performances, notamment la rapidité de desserte grâce aux voies réservées
- Les coûts de moindre poids sur les budgets
- La sécurité

---

<sup>10</sup> <http://chimistes-environnement.over-blog.com/article-37152156.html>

Il existe de ce fait une nette différence entre l'utilisation de la voiture personnelle et des transports en commun, le premier au détriment du deuxième qui est, cependant une pratique en train de changer. Le changement qui s'opère en France et dans le monde entier vient suite à l'incitation des organismes concernés à aller vers les transports en commun de par des campagnes de prise de conscience des avantages de ceux-ci et la mise en exergue du danger émanant de la voiture personnelle. Cette prise de conscience est croissante en France et en Europe. L'impact environnemental perçu durant ces dernières années est à la clé de ce mouvement détectable de par les changements comportementaux des individus adaptant de plus en plus les transports en commun plutôt que la voiture personnelle. La Figure I.11 (2) montre des statistiques sur la base de l'indicateur voyageur-kilomètre sur les différents modes de transport. Ces courbes, où la dernière vingtaine d'années a témoigné de l'avènement plus important des individus aux modes de transport en commun, montrent en parallèle une légère diminution de l'usage de l'automobile privée qui est une première dans l'histoire de la voiture (depuis les années soixante-dix). Ayant commencé en 2005, cette baisse continue jusqu'à présent faisant écho avec les projets de mobilité durable.



*Figure I.11. Evolution du comportement des individus par rapport aux modes de transport adoptés*

Toujours à l'horizon du développement durable, les principaux responsables continuent de s'intéresser à diminuer d'avantage la fatalité de l'impact de la voiture personnelle sur l'environnement et la vie sociétale. Dans cette perspective, un élément clé qui consiste en la combinaison du concept de transport en commun et de l'automobile particulière est né. Cette idée a contribué à faire basculer la voiture personnelle d'un moyen de transport privé vers un mode collectif. Elle se base sur le concept novateur de l'utilisation d'un véhicule à l'origine personnel et privé pour effectuer des déplacements de groupe. Nous allons dans la suite de ce

chapitre nous approfondir dans cette thématique en particulier sur la base de la problématique du véhicule partagé. Cependant, nous présentons préalablement les différents modes de transport, leurs spécificités, la notion de service de transport, d'irrégularités ou perturbations etc. Nous mettons par ailleurs l'accent sur la complémentarité des systèmes qui régissent les principaux modes les plus en vue de ce secteur d'activité afin de mieux placer nos travaux ainsi que la notion du véhicule partagé en tant que concept en soit.

### **I.3. Transport : Les différents modes**

Préalablement considérés chacun dans son propre contexte, les moyens de transport qu'ils soient publics ou privés possèdent leurs propres avantages et inconvénients. Parmi les principaux problèmes du transport en commun en particulier, la limitation des zones de desserte qui est en réalité un inconvénient majeur et un principal handicap à la mobilité des individus. Pour remédier à cet inconvénient, des combinaisons de plusieurs modes de transport permettent de fournir plus de flexibilité et de souplesse dans les déplacements rendant ainsi le réseau de transport plus fiable et beaucoup plus rentable. Dans ce contexte, les services de transport sont passés du monomodal au comodal en passant par d'autres modes de transport. Tous sont définis dans ce qui suit.

#### **I.3.1. Le transport monomodal**

Ce type de transport concerne les déplacements réalisés avec un seul mode tout au long du trajet à effectuer ; un mode de transport désignant le moyen utilisé pour déplacer un objet (un bien ou une personne) d'un lieu à un autre<sup>11</sup>. Le transport monomodal désigne ainsi le fait d'utiliser un mode particulier pour réaliser les opérations de transport et considère, par conséquent, un type spécifique de véhicule (e.g. Bus, Métro, Tram, etc.).

#### **I.3.2. Le transport intermodal**

Le transport combiné est une forme de transport intermodal de type « *rail-route* ». En effet, à l'opposé du transport monomodal, l'intermodalité désigne le fait d'adopter la possibilité de changer de mode de transport pour effectuer un déplacement d'une origine à une destination données. Ainsi, deux modes de transport ou plus peuvent être utilisés pour réaliser un trajet complet entre deux points. Cherchant à répondre à une question cruciale qui est celle de remédier aux restrictions imposées par les transports en commun par rapport aux zones hors d'atteinte pour ces réseaux, l'intermodalité se trouve être une solution bien adaptée. Elle vise

---

<sup>11</sup> [www.businessdictionary.com](http://www.businessdictionary.com)

ainsi à réduire l'usage de l'automobile privée connue pour être l'alternative la plus appropriée pour répondre au besoin de mobilité en zones extra-urbaines. L'intermodalité favorise, de ce fait, l'usage de modes de transports moins polluants en combinant les moyens de transports collectifs entre eux, mais aussi aux vélos et autres modes (17).

### I.3.3. Le transport multimodal

La multimodalité concerne toute offre de transport faisant usage de plusieurs moyens de transport pour une même demande de déplacement d'une origine vers une destination données. Plusieurs offres de plusieurs moyens de transport différents peuvent ainsi exister pour un unique trajet. Le libre choix revient donc à l'utilisateur de décider du moyen qu'il veut emprunter pour se déplacer, comme il existe aujourd'hui bien des systèmes à qui ce rôle incombe. Une multitude de travaux (18) (19) (20) traitent ainsi du sujet de la multimodalité en tant que service de transport de grande ampleur. Plusieurs systèmes d'aide au déplacement des voyageurs existent aussi et ont pour mission de fournir à leurs utilisateurs des solutions indiquant la panoplie de modes de transport qu'ils peuvent emprunter pour effectuer un trajet bien déterminé. Des systèmes particuliers peuvent aller jusqu'à choisir pour leurs usagers le mode qu'il convient de prendre selon certaines stratégies de résolution fixées au préalable. Le choix s'effectue alors selon la pertinence du service de transport et l'optimalité des combinaisons.

Dans un élan d'essai pour l'optimisation des services de transport (21) (22), de grands efforts ont été dépensés dans ce domaine afin de développer des services de transport multiples dans le but premier de satisfaire les usagers de tels systèmes. Nous citons à titre d'exemple ceux de M.F. Feki (2) qui propose une étude des systèmes de transport visant l'élaboration d'un support informatique fiable tout en faisant figure de point médiateur entre les différents opérateurs de transport, et ce afin de pouvoir extraire l'information nécessaire. Cette étude a abouti à la mise en place d'un Système d'Information d'Aide au Déplacement (SIAD) des voyageurs proposant de fournir les plus courts chemins selon les demandes des usagers du service. A ce titre, le système proposé concrétise une approche d'aide à la décision pour l'extraction d'itinéraires les plus courts selon l'intervalle de temps requis et selon l'offre existante durant cet intervalle. Tous les moyens de transport sont ici considérés et l'étude des trajets se fait sur la base des données collectées à partir des bases de données des opérateurs de transport collaborant. Toute demande aboutit à l'élaboration d'un carnet de voyages

proposant plusieurs alternatives relativement à des chemins solutions pouvant être empruntés, tout en fournissant le détail sur les différentes combinaisons possibles.

A l'aube d'une optimisation intégrée dans les systèmes de transport, la qualité de service est un facteur pivot auxquels aspirent les systèmes de transport aujourd'hui existants. Offrant des services d'aide au déplacement, les travaux existants sont ainsi une poussée d'innovation aidant à propulser le secteur du transport vers l'amélioration de la qualité du service offert tout en jouant la carte du développement durable. Les principales préoccupations quant à la bonne conduite de ces travaux étant de gérer au mieux l'offre de transport dans un contexte d'amélioration de la qualité de service du transport, chercheurs et praticiens n'oublient toutefois pas l'enjeu principal qui est de relever le challenge économique et environnemental. Dans ce contexte, les dispositifs politico-sociaux considérés conditionnent la voie dans laquelle ont été menés ces projets (21) (23) (19) (24) pour ainsi considérer un contexte d'optimisation dans l'ensemble de ces travaux. Toutes les notions relatives à ce concept seront définies et profondément détaillées dans le chapitre suivant. Nous nous intéressons effectivement à cet aspect en particulier, notre but premier étant de cadrer avec l'efficacité et l'optimisation de la qualité de service dans nos propres travaux.

### I.3.4. Le transport comodal

Dans l'optique de réunir l'ensemble des moyens de transport précédemment énoncés, un quatrième mode de transport a émergé. Ayant été introduit dans le domaine de la politique des transports en 2006 par la commission européenne (25), il s'agit d'un mode particulier adoptant le transport sous toutes ses formes afin de fournir un service souple offrant un maximum de flexibilité à l'utilisateur. C'est dans ce contexte que sont alors envisagées des combinaisons rassemblant les moyens de transports en commun (Tram, Train, Avion, etc.) et privés (automobile privée, etc.). A l'opposé, l'intermodalité incite à aller plutôt vers les transports en commun suscitant de ce fait une sorte de rivalité entre les services de transport se basant sur l'un et l'autre des pratiques collectives ou privées (25). La comodalité vient par ailleurs contrer cette concurrence pour imposer et instaurer un certain équilibre mettant en exergue la complémentarité entre les modes de transports collectifs et privés (26). Le but principal de par les systèmes de transport comodaux étant unique, seule la qualité de service est visée. Les aspirations de tels systèmes vont donc dans un seul sens qui est celui de l'optimalité du service offert. Pour bien expliquer cette initiative, la section suivante sera consacrée à une étude approfondie de ce mode émergent.

## **I.4. Le transport comodal : Vers un système de services de transport**

Extrait de l'avis de la commission : « C'est uniquement grâce à une interopérabilité réelle entre les différents modes de transport dans des conditions de marché équitables que peut être obtenue une optimisation naturelle des transports ... On ne peut améliorer la situation des transports en Europe qu'en instaurant des conditions équitables pour tous les «modes» de transport, sans en privilégier aucun » (27).

### **I.4.1. Pour l'équité dans le choix des modes de transport**

Rassemblant les moyens de transport ferroviaires, les bus, les métros, etc. pour assurer la continuité des voyages, la politique de transport multimodale a évolué depuis l'année 2006 vers une politique comodale (25). Cette dernière constitue une vision évolutive faisant un usage intelligent de la voiture personnelle tout en contrant l'effet négatif de l'afflux excessif des personnes vers ce mode unique, et ce de par une complémentarité avec les modes de transports collectifs délaissant la rivalité entre ceux-ci pour une combinaison de tous les modes de transport (28).

La comodalité étant définie comme « *la combinaison optimale des différents modes sur la chaîne de transport* » (25), elle concrétise l'idée de l'amélioration de la qualité de service du transport. Grâce à la comodalité, sont offertes toutes les possibilités de combinaisons sur la base des offres de services de transport disponibles avec notamment d'éventuels « *transbordements* », et ce afin de fournir le chemin optimal d'une origine à une destination bien déterminées. Pour cela, elle considère tout moyen de transport pouvant être emprunté considéré isolément ou en combinaison avec d'autres.

Toujours dans un souci d'optimisation des services de transport, la comodalité vient apporter une utilisation intelligente et efficace des différents modes de transport, à savoir le transport routier, maritime, ferroviaire et aérien. Elle a ainsi pour objectif premier d'optimiser les combinaisons et vient donc remplacer la multimodalité en joignant les transports individuels à ceux collectifs pour plus d'efficacité (29). En effet, passant de la concurrence à la complémentarité, le transport en général et plus particulièrement le transport des personnes ont évolué vers une qualité de service plus satisfaisante alliant les avantages propres à chacun de ces modes. Cette évolution a été ressentie tant sur le plan économique (i.e. minimisation

des coûts et budgets alloués au transport) que sur le plan environnemental (i.e. atténuant l'image polluante de certaines locomotives ainsi que des voitures).

Dans ce sens et de manière similaire à la multimodalité, la comodalité serait aussi à l'origine du développement d'infrastructures adaptées qui, conjointement à des mesures et des actions spécifiques devant être prises, assurent l'optimalité des combinaisons des différents modes de transport (30). La mesure d'optimalité ici considérée évoque plusieurs domaines. La comodalité a en effet une portée économique, environnementale, commerciale, financière, comportementale (sociétale), etc.

De ce fait, la réussite des services inscrits dans ce contexte vient entre autre de l'intégration du concept essentiel de la comodalité qui grâce à la complémentarité des moyens de transports qu'elle propose a su se distinguer de par l'aspect d'optimisation obtenue au travers de ces combinaisons.

### I.4.2. Complémentarité des systèmes

Pour plusieurs raisons, considérer les moyens de transport chacun dans son contexte s'avère toujours insatisfaisant. Ceci est dû à l'ampleur des risques encourus et qui concernent notamment l'impact sur la flexibilité, la sécurité, la continuité des trajets, etc. Aussi, parmi les facteurs importants causant cette insatisfaction, nous pouvons citer à titre d'exemple la variabilité des besoins de mobilité motivant la nécessité d'adaptation des offres de transport. Un utilisateur peut avoir à se déplacer la plupart du temps en ville ou en agglomération, dans ce cas les moyens de transport les plus préconisés sont ceux relatifs aux modes de transport en commun, moins encombrants et surtout moins polluants. Par ailleurs, l'étalement périurbain vient d'imposer la voiture comme moyen de transport essentiel. En effet, la nécessité de déplacement hors des zones d'urbanisation conjointement aux limites et restrictions des zones de desserte des transports en commun a créé un besoin récurrent de recourir à la voiture personnelle pour pouvoir satisfaire des besoins en matière de déplacement auxquels les modes de transport collectifs ne peuvent répondre. C'est ainsi que les individus sont devenus de plus en plus dépendants à leurs véhicules personnels puisqu'ils leur offrent liberté, confort et flexibilité aussi bien spatiale que temporelle. Cependant, l'usage de l'automobile privée engendre plus d'une séquelle sur le plan individuel aussi bien que collectif (*c.f. I.1*). Bénéficier de ces multiples avantages a effectivement un prix et se fait au détriment des budgets, de l'environnement dans lequel évolue la collectivité, des comportements, de la qualité de la vie, etc.

En contrepartie, les transports collectifs viennent remédier aux principaux problèmes de l'usage abusif de la voiture créant des modes de transport propres et très économes en matière d'énergie. Par ailleurs, les transports publics malgré leur grande évolution restent à eux seuls insatisfaisants à cause de non seulement les restrictions imposées vis-à-vis de l'étalement géographique mais aussi par rapport aux risques pouvant toucher les réseaux de transport collectifs. Nous parlons ici de manière spécifique des incidents et perturbations engendrant le disfonctionnement (partiel ou total) de ces réseaux et qui sont de plus en plus fréquents. Cette fréquence est devenue avec le temps inquiétante et déterminante pour la recherche d'un compromis visant l'interopérabilité entre les moyens de transport publics. Ce compromis porte le nom de la comodalité.

Loin de toute discrimination, la comodalité considère de ce fait à parts égales et dans une interopérabilité équitable (31)) tous les modes de transport possibles. La comodalité œuvre de ce fait dans le sens de l'équité entre les moyens de transports publics (collectifs ou individuels) et privés dans un contexte de combinaison favorisant plutôt la complémentarité à la concurrence (26).

### I.4.3. Perturbations dans les systèmes de transport

Pour introduire le concept de perturbation et d'irrégularité dans les services de transport, il convient tout d'abord de définir ce qu'est une perturbation :

***Perturbation*** : est considéré une perturbation tout incident survenant de manière aléatoire et pouvant aller à l'encontre de la bonne marche du réseau de transport. L'impact de tels accidents peut se voir à un écart enregistré entre le Tableau de Marche Théorique (TMT) et le Tableau de Marche Réel (TMR) (32). D'internes à externes, une panoplie de facteurs existent et sont probablement à l'origine des perturbations ayant lieu (33). Parmi les causes engendrant des perturbations, nous pouvons citer par exemple les problèmes techniques touchant le matériel ou le personnel comme facteurs internes, et les aléas dus aux changements de la demande ou des données sur le trafic comme facteurs externes. Une autre classification (34) distingue les causes de perturbations selon qu'ils sont d'ordre naturel, technique ou humain. Par ailleurs, que l'on considère l'une ou l'autre de ces classifications, les conséquences possibles des perturbations demeurent les mêmes, donnant lieu à des irrégularités plus ou moins importantes dans les services de transport.

***Irrégularité*** : Une perturbation engendre de manière générale des irrégularités ressenties dans un changement des spécifications de circulation prévue d'un mode de transport. Son

effet peut être vu par exemple dans la modification des intervalles de passage des véhicules et par conséquent à travers un prolongement des temps d'attente des usagers dans les stations. Ce qui a pour effet de détériorer la qualité du service pouvant notamment créer un déséquilibre au niveau de la répartition des charges entre les véhicules.

Qu'il s'agisse d'une interruption d'une ou plusieurs lignes, d'un changement des heures d'arrivée ou de l'arrêt de desserte d'une ou plusieurs stations, les perturbations n'ont de fait que de déranger le bon cours du service et d'inciter les usagers à annuler leurs voyages ou à favoriser d'autres services. De plus, les observations de perturbations vont en croissant, ce qui a pour effet de détériorer encore plus la qualité du service offert. Pour le cas de la France par exemple, et plus particulièrement les opérateurs de transports SNCF et RATP, la dernière décennie a témoigné d'une évolution rétrogradante observée de par une croissance de l'indice d'irrégularité<sup>12</sup>. Cette évolution est illustrée au niveau de la Figure I.12 et a été démontrée de par les statistiques faites utilisant un indice d'irrégularité se basant sur la détermination des pourcentages de trains, métros, bus et trams arrivant à leurs destinations avec cinq minutes de retard.

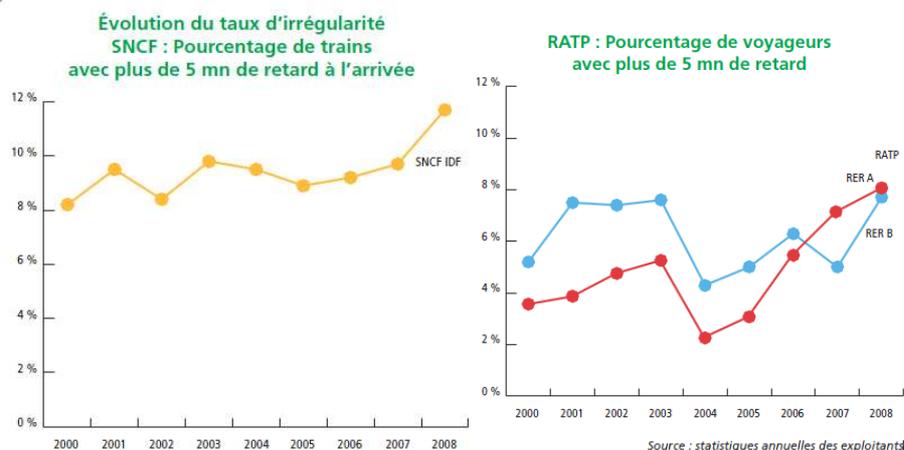


Figure I.12. Evolution des perturbations en France entre les années 2000 et 2008

Par conséquent, les systèmes de transport ayant opté pour la mise en place d'une qualité de service optimale, il est nécessaire de pouvoir gérer l'ensemble des perturbations survenant sur le réseau et pourquoi pas les anticiper. Outre les modules de maintenance améliorative et préventive mises en place par la majorité des systèmes de transport pour essayer de prévenir tout type d'incident pouvant survenir, le réseau de transport n'est toutefois jamais à l'abri des irrégularités. Ce qui va à l'encontre du but premier qui est celui d'atteindre l'optimalité requise dans les services de transport. C'est dans ce contexte que la notion de gestion en

<sup>12</sup> Les exploitants de transport utilisent plusieurs indicateurs (indice d'irrégularité) pour mesurer les perturbations

temps réel intervient pour essayer d'amoindrir au maximum les « dégâts » et offrir des services compétitifs à une grandeur d'échelle souhaitable.

C'est sur cette lancée et dans le cadre d'une gestion dynamique que s'est renforcée la notion d'« *Information Voyageur* » pour devenir de première importance offrant aux usagers un maximum de transparence et des informations actualisées en continu. Plusieurs travaux de recherche en coopération avec le secteur industriel, notamment avec les opérateurs de transport, se sont focalisés sur l'offre de tels services pour aider les usagers dans leurs déplacements (2) (35) (23). L'auteur dans (36) traite le problème de la gestion du trafic ferroviaire en cas d'aléas en adoptant une approche de modélisation et d'optimisation. Par ailleurs, gérer l'information multimodale à elle seule reste insuffisant, et c'est ainsi que la pratique comodale est devenue un maillon essentiel dans cette problématique.

### I.4.4. Transport comodal : une solution pour la régulation

La pratique comodale est aujourd'hui un concept émergent à travers le monde entier. Vu les réponses qu'elle apporte, la comodalité promet d'apporter l'équilibre escompté dans les services de transport et offrir l'optimalité non encore atteinte dans les services de transport. En effet, ce concept vient remédier aux problèmes érigés par les moyens de transport en commun ou les véhicules privés considérés séparément. Des travaux ont ainsi fait l'objet d'études approfondies en Grèce (37), en France (2) ou ailleurs. Nous pouvons citer à titre d'exemple le projet *easyway*<sup>13</sup> soutenu par la commission européenne et qui s'inscrit dans le domaine de la route intelligente sur le réseau trans-européen. La comodalité a ainsi été proposée pour optimiser le transport des personnes mais aussi le transport des marchandises (38).

Les pratiques multimodales basées sur les transports publics étant les plus en vogue grâce essentiellement à leurs avantages économiques et environnementaux, elles restent toutefois insatisfaisantes dans plusieurs cas. La comodalité vient par ailleurs promouvoir la voiture particulière dans un contexte de complétude avec le reste des services de transport pour plus de satisfaction du voyageur. Elle vient ainsi soigner l'image de l'automobile mettant l'accent sur l'utilité de son intégration dans les offres de transport, surtout en cas de perturbation ou d'allongement des distances hors d'atteinte des transports en commun.

Mettant le focus sur la complémentarité des modes de transport, les travaux de (2) incluent, en plus de l'information sur les modes de transport collectifs, les services d'Ecopartage

---

<sup>13</sup> <http://www.predim.org/spip.php?article3818>

uniquement si besoin est. L'Ecopartage réunit à ce titre les modes de partage de véhicules, à savoir l'autopartage et le covoiturage.

En effet, prétendant à une efficacité au niveau des traitements et qui se refléterait sur les performances du système, rassembler les informations à partir des sites de covoiturage et organisme d'autopartage s'avère difficile et très complexe pour être effectué en temps réel et à chaque fois qu'un utilisateur le demande. Pour cela, Feki (39) propose dans ses travaux d'intégrer de tels services uniquement en cas de perturbations faisant de la voiture partagée une solution d'appoint pour remédier aux problèmes des modes de transport en commun.

C'est ainsi qu'après les aspirations des politiciens et responsables à un monde évolué mais propre et avec la pratique comodale aujourd'hui essentielle, la voiture vient de faire d'emblée son entrée dans les services de transport publics et donc dans les systèmes d'aide au transport lui donnant une légitimité dans le processus de développement durable.

### I.4.5. La voiture : de l'individuel vers le collectif

Grâce aux pratiques comodales, la propagande des moyens de transport publics aidant, l'avènement vers ces modes est de plus en plus important. Cette montée en puissance des transports collectifs n'a de fait que de faire régresser l'usage abusif des transports privés et plus particulièrement la voiture personnelle. La Figure I.13 montre l'effet de l'évolution des services publics de par une détérioration dans l'utilisation des voitures. Une détérioration observée entre 1990 et 2008 qui reste toutefois assez légère et loin de satisfaire les buts escomptés.

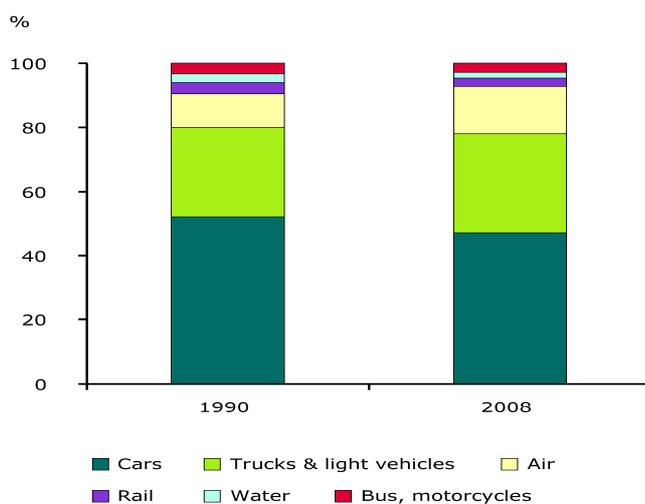


Figure I.13. Evolution de l'afflux des voyageurs aux modes de transport de 1990 à 2008<sup>14</sup>

<sup>14</sup> <http://www.eea.europa.eu/data-and-maps>

Visant à réduire d'avantage l'impact négatif du domaine du transport particulièrement sur l'environnement, tous les regards ont été dirigés dans ce sens. La voiture personnelle a été démontrée pour y être en grande partie responsable, accaparant la proportion d'émissions de CO<sub>2</sub> la plus importante (*c.f. 1.2.1.2, Figure I.10*). De plus, avec l'évolution du rapport à l'automobile il devient de plus en plus urgent de trouver un bon remède qui stopperait net cette évolution et saurait faire régresser les comportements des individus vers un usage plus raisonnable et donc des impacts de moindre incidence. Le but visé est de réduire le nombre de voitures en circulation, réduisant par la même occasion son impact péjoratif sur la fluidité du trafic et l'environnement tout en maintenant accessible la flexibilité et facilité qu'elles offrent. Une solution clé se trouve dans le concept de voiture partagée qui soigne l'image de l'automobile particulière la délogeant du contexte réquisitoire dont elle a toujours fait figure tout en préservant les avantages qu'elle présente.

## **I.5. La voiture partagée**

Les moyens de transport à la base privés devenus publics (i.e. la voiture personnelle) ont donné naissance à des systèmes de transport à priori individuels mais devenus collectifs (e.g. services de covoiturage, autopartage ...). De tels systèmes ont vu le jour suite au développement de plusieurs travaux visant l'amélioration de la qualité de vie en considérant particulièrement le développement du domaine de transport dans un contexte mélioratif. Amélioration en termes des services fournis aussi bien sur le plan individuel que collectif et dont l'objectif premier consiste essentiellement à améliorer les conditions de vie des individus d'un côté et augmenter leur satisfaction de l'autre. Cependant, aussi évolués soient-ils, ces systèmes ne sont pas passibles d'efforts supplémentaires afin d'en optimiser le fonctionnement (40). Dans ce qui suit, nous passons en revue les principales études et approches s'inscrivant dans le contexte de la voiture partagée. Cette dernière a donné lieu à deux catégories selon le type de partage (Figure I.14) du véhicule :

- **Autopartage** : et il s'agit du partage d'un même véhicule par période de temps par des personnes différentes à chaque fois.
- **Covoiturage** : partage d'un véhicule pendant un même intervalle de temps par plusieurs personnes pour parcourir une partie ou la totalité d'un trajet commun.

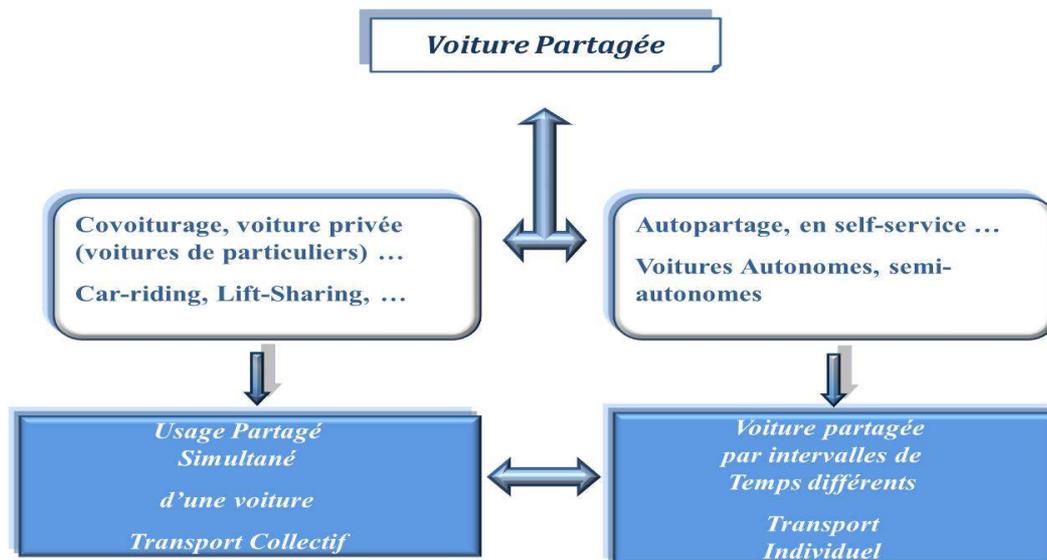


Figure I.14. La voiture partagée : Autopartage ou Covoiturage ?

Chacune de ces deux catégories sera définie et détaillée dans la suite de cette section.

### I.5.1. L'autopartage

Afin de résorber les problèmes existants engendrés de par l'utilisation massive de la voiture particulière, un nouveau concept est né. Ce concept porte le nom d'autopartage et a pour mission de rendre publique l'utilisation d'une automobile destinée à la base à un usage personnel.

#### I.5.1.1. Autopartage : Définition

Comme son nom l'indique, l'autopartage considère le partage d'un véhicule par une collectivité à des fins d'en optimiser l'usage et tente entre autre de solutionner la future pénurie mondiale de pétrole. Les chiffres et statistiques relevés en France (41) ne peuvent que le confirmer. Il s'agit en effet d'une forme de partage de véhicule particulier pour effectuer un trajet répondant à un besoin de déplacement bien déterminé. L'autopartage, ainsi défini, est un service offert par une entreprise, une association ou une agence publique à toute personne en droit d'y accéder, mettant à sa disposition une flotte de véhicules en libre-service dont elle aura à partager l'usage avec d'autres membres.

#### I.5.1.2. Autopartage : Mode d'emploi

Grâce à cette pratique, tout individu ayant un permis de conduire et qui s'est procuré ce droit peut disposer d'une voiture à volonté et sans restriction délaissant ainsi le souhait ou le

besoin d'en acquérir une pour ses déplacements personnels. L'autopartage est donc défini comme un service de mobilité prodiguant à ses usagers le droit de disposer d'une voiture « à la carte » moyennant une adhésion préalable lui donnant la possibilité d'accéder au service. Pour profiter des fonctionnalités offertes par ce dernier, tout usager doit effectuer une réservation (par téléphone, via Internet...) à l'avance pour pouvoir prendre un véhicule (Figure I.15) (42).

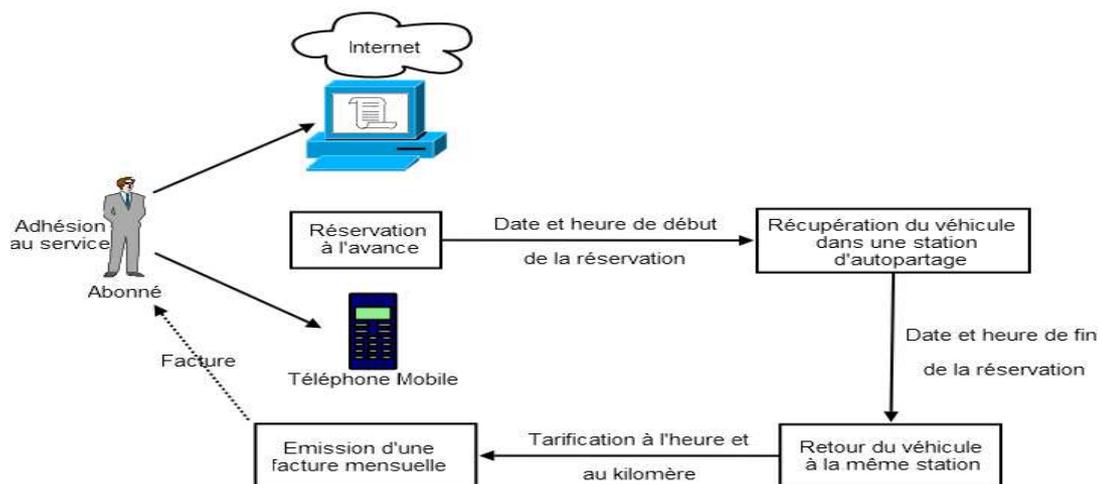


Figure I.15. Autopartage : Planification des déplacements, utilisation et tarification

Il s'agit plus précisément d'un partage de parcs de véhicules par les membres du service. L'ensemble des voitures de la flotte est ainsi disponible à une prise autonome moyennant une carte à puce utilisée pour débloquer un boîtier contenant les clés ou un ordinateur embarqué sur la voiture (43).

### I.5.1.3. Autopartage : Tarification et Mesures

Outre le coût d'adhésion, la tarification se fait selon l'usage du véhicule. Une formule basée sur la durée d'utilisation (temps) et la distance parcourue (kilomètre) a ainsi été mise en place pour la tarification du service selon l'utilisation des abonnés. Cette formule s'avère dans la plupart des cas beaucoup plus avantageuse et moins coûteuse pour les individus que les frais engendrés par l'acquisition d'une voiture personnelle (prix du véhicule, assurance, carburant, etc.). Par ailleurs, des mesures spécifiques ont été prises pour encourager les individus à aller vers ce mode de transport, telles que la gratuité des frais de parking. De plus, des routes spécifiques ont été aménagées dans ce sens et une restructuration des voiries a été élaborée en conséquence.

Sur la base de cette définition, l'autopartage promeut le véhicule particulier comme un moyen de transport propre et efficace de par la flexibilité, la liberté et la facilité d'accès qu'il offre à ses usagers. Véhiculés par cette initiative, beaucoup de projets industriels développent des idées allant dans ce sens. Les voitures électriques beaucoup moins polluantes et économiques en énergie, les smart véhicules prenant très peu de places et facilement maniables surtout en ville, les véhicules hybrides, etc. sont ainsi nés pour répondre au besoin de mettre en œuvre des systèmes de mobilité durable et avancée et sont fortement liés au concept de l'autopartage. Faisant écho avec ces systèmes grâce au grand intérêt qu'il suscite chez les industriels aussi bien que les chercheurs, l'autopartage est aujourd'hui une source intarissable d'avancées technologiques. Beaucoup d'efforts ont été déployés dans ce sens et l'autopartage a connu une propulsion stratégique englobant le monde entier.

### *1.5.1.4. Autopartage : Histoire et état de l'art*

L'autopartage a vu le jour en 1948 avec la plus ancienne organisation de partage de véhicule retracée dans la littérature : SEFAGE (Selbstfahrergemeinschaft, ce qui pourrait être traduit en " club de conducteurs "). Cette organisation a été fondée à Zurich, en Suisse. Il s'agissait essentiellement d'un club où les membres s'étaient cotisés pour acquérir une automobile. Sans aucune fin commerciale, l'objectif principal était de s'offrir le privilège de disposer d'un véhicule disponible selon le besoin (44). Les initiateurs de ce service n'étant pas conscients de son caractère novateur, SEFAGE ne s'est jamais développée davantage et n'a jamais compté plus de quelques usagers et l'expérience fut ainsi sans lendemain.

Après cette expérience, et plus particulièrement durant ces dernières années, le concept de l'autopartage a connu un grand développement donnant naissance à plusieurs nouveaux systèmes. Les travaux de recherche réalisés jusqu'aujourd'hui peuvent être subdivisés en deux catégories suivant les critères de gestion des réservations adoptés au sein du système. La première catégorie se base sur une gestion statique des réservations alors que la deuxième traite l'aspect dynamique (i.e. en temps réel).

Une multitude de systèmes très ou peu connus existent d'ores et déjà dans plus de 1000 villes<sup>15</sup>. Classés dans l'une ou l'autre de ces catégories, ils sont opérables à travers le monde entier allant même jusqu'à s'installer dans les petites communes. Nous en citons à titre d'exemple le réseau bien connu « *France Autopartage* » créé au début des années 2000 (43)

---

<sup>15</sup> <http://www.cities.worldcarshare.com/>

et qui regroupe une panoplie de projets et travaux ayant été réalisés pour la mise en place de services d'autopartage plus ou moins évolués sur le territoire français. C'est ainsi que des systèmes tels que City Roul' à Rennes, Lilas à Lille, Mobilib à Toulouse, Autolib' à Lyon, etc. ont intégré ce réseau pour engendrer une expansion très remarquable de ce concept en France allant jusqu'à s'installer dans les petites provinces. En parallèle, l'autopartage a aussi été très développé de par le monde entier, touchant les pays de l'Europe (45), les états unis, la Grande Bretagne, etc. Parmi les systèmes qui relèvent de la gestion statique des réservations, l'on peut citer : mobility qui est le leader du marché de l'autopartage en Suisse, Communauto au Canada, CityCarClub en Grande-Bretagne, CambioStadt dans la ville de Bremen en Allemagne, Autopia et Cambio qui sont les deux alternatives d'autopartage en Belgique, ZipCar et I-GO aux états unis, etc.

Par ailleurs, en matière de réservations dynamiques, quelques systèmes existent mais la plupart d'entre eux sont restés à un stade embryonnaire n'intégrant que très partiellement l'aspect temps réel dans leurs services. Ayant développé un concept innovant de nouvelle mobilité urbaine de type *VIP* pour *Véhicules Individuels Publics* sans obligation de stationnement dans des sites dédiés à cet effet ; le projet le plus en vue aujourd'hui en France est celui baptisé *Cité Vu*<sup>16</sup> implémenté à Antibes et réalisé par la société *Vu Log*. Dans le même contexte de gestion dynamique des réservations, nous pouvons aussi citer le projet *Autolib'*<sup>17</sup> implanté à Paris et dont le déploiement a été initialement prévu pendant le dernier trimestre de l'année 2010 mais qui n'est toujours pas opérationnel jusqu'à ce jour. Tous ces travaux, qu'ils soient inscrits dans le cadre d'une gestion statique ou dynamique, ont été présentés et détaillés dans (46).

### *1.5.1.5. Autopartage : Meilleur substitut pour la voiture personnelle ?*

Sur la base de tout ce qui a précédé, nous pouvons conclure que la voiture en accès libre se présente donc comme un service de mobilité avancée qui offre les avantages de l'automobile privée dont essentiellement une grande flexibilité d'usage, tout en limitant, pour ses adhérents, la nécessité d'en acquérir une (47). Dans ce sens, le véhicule partagé est souvent promu comme l'alternative adéquate venant remplacer le besoin d'acquisition d'un véhicule personnel (48). Il constitue ainsi le meilleur substitut à la voiture privée (49) dans la mesure où il permet de bénéficier des mêmes avantages tout en se débarrassant de ses inconvénients.

---

<sup>16</sup> <http://www.citevu.com/citevu/home/index.aspx>

<sup>17</sup> <http://www.autolib-paris.fr/>

Par ailleurs, l'étude approfondie du concept de l'autopartage nous a amenés à en dénicher plus d'un inconvénient, non très importants mais préservant certaines limites et restrictions quant à l'usage spontané du véhicule. En effet, le service d'autopartage, sur la base d'une tarification à l'heure et au kilomètre parcouru, n'est avantageux que dans le cas de courte distance. Dans le cas contraire, il devient très coûteux de prendre un véhicule de la flotte pour un usage personnel sur une durée plus ou moins longue (e.g. pour partir en vacances, en mission, etc.). Le service d'autopartage est ainsi fait pour des trajets de courte durée (inférieurs à une journée). Par ailleurs, les personnes dépourvues du luxe de possession d'une voiture personnelle sont celles qui bénéficient le plus de ce service. Celles qui sont déjà dotées de ce luxe préfèrent leurs véhicules personnels, plus faciles d'accès et plus souples d'usage. C'est dans ce sens, et conjointement aux problèmes de gestion logistique de la flotte de véhicules, d'encombrement des routes et autres, que l'autopartage nécessite plus d'efforts pour être vraiment compétitif. En parallèle, le covoiturage, au lieu de rajouter de nouveaux véhicules, utilise ceux des particuliers sans nécessité d'aménagement de routes ou d'infrastructures pour la mise en place de stations et parcs pour accueillir la flotte de véhicules. De plus, il n'est pas non plus nécessaire dans le concept du covoiturage de mettre en place des services de redistribution de la flotte, de gestion logistique, de maintenance technique ou autre. Certains systèmes mobilisent des staffs techniques, un personnel expert et spécialiste, des véhicules lourds, etc. pour la maintenance et la gestion de la flotte. Ceci engage l'entreprise initiatrice du service dans un labyrinthe sans issue de charges et investissements assez coûteux en matière de main d'œuvre, de produits et matériels spécifiques, etc. étant essentiellement basé sur les véhicules de particuliers, le covoiturage vient remédier à tous ces problèmes et répondre à des questions longtemps restées en suspens.

### I.5.2. Le covoiturage

Cette deuxième forme de partage de voiture particulière, pareille à sa précédente, connaît aujourd'hui un succès remarquable, grâce notamment aux avantages qu'elle a contribué à apporter. Elle a de fait soigné l'image environnementale, sociale et économique de la voiture réduisant les émissions de CO<sub>2</sub>, les dépendances à la voiture, les frais de transport engagés dans le cadre d'une mobilité individuelle, etc. Par ailleurs, outre les avantages économiques et écologiques, le covoiturage permet aussi de restaurer une certaine communication qui a disparu dans les transports en commun et qui a demeuré absente même avec le concept de l'autopartage. C'est ainsi que le covoiturage a permis, de par le regroupement de personnes se connaissant ou pas, de créer et de fortifier les liens sociaux.

Compte tenu de la montée en puissance du concept de la voiture partagée et plus particulièrement du covoiturage, mettant à profit les contributions qu'il est susceptible d'apporter à la société, nous avons choisi de diriger nos travaux vers ce contexte. Par la suite, nous proposons de nous approfondir dans ce sujet pour en détailler les axes principaux. Nous en fournissons alors une étude détaillée avec une liste non exhaustive des systèmes existants avant de définir la manière dont nous abordons le sujet de par notre approche ; laquelle se focalise essentiellement sur l'absence dans les travaux existants de certains concepts pourtant pertinents.

### *1.5.2.1. Quelques Définitions*

Etant particulièrement intéressés par la problématique du covoiturage, nous nous focalisons essentiellement sur ce concept pour en définir les grandes lignes. Une revue générale de la littérature est détaillée dans cette section, mais nous présentons tout d'abord dans ce qui suit les notions de base relatives à ce concept pour pouvoir en aborder les principaux éléments.

***Le challenge :*** avec l'allongement des distances et les durées des trajets, provoquant une explosion de la mobilité qui n'est pas sans conséquences néfastes, la voiture particulière est devenue une source d'ennui après le succès fulgurant dont elle a été couronnée. En effet, outre les nuisances sonores, la pression qu'elle exerce sur les individus (formes de dépendance), les dépenses énormes, etc. l'utilisation massive de la voiture personnelle provoque l'extermination du concept écologique propre. En grande partie responsable des problématiques actuelles, à savoir la pollution atmosphérique, les trous dans la couche d'ozone, le changement climatique, etc. il devient urgent de remédier à l'expansion hallucinante de la voiture. Chose qui n'est pas du tout facile, vue la nécessité de l'intégrer dans les déplacements pour couvrir certains territoires non accessibles par les transports en commun. Relever le défi de la réduction de l'usage excessif de la voiture particulière est d'autant plus difficile que celle-ci exerce une pression incommensurable sur les individus créant de forts liens de dépendance. Une solution réside de ce fait dans l'utilisation de celle-ci tout en en réduisant le nombre pour revenir à un usage raisonnable non abusif. L'accès de groupe et qui de surcroît fait usage des voitures de particuliers aide à réduire considérablement le nombre de voitures en circulation. Ce concept fait la définition même du covoiturage. Comme le montre la Figure I.16<sup>18</sup>, la différence entre le nombre de véhicules

---

<sup>18</sup> <http://www.naturavox.fr/societe/article/petite-histoire-du-covoiturage>

nécessaires pour transporter 30 personnes est remarquable : un autobus uniquement est nécessaire pour déplacer la totalité de ces individus et 30 pour le cas d'un véhicule individuel. Ce nombre est par contre réduit de moitié dans le cas d'un covoiturage de deux personnes au lieu que chacune emprunte son automobile personnelle. Qu'en serait-il si quatre covoiturés étaient dans le même véhicule ?

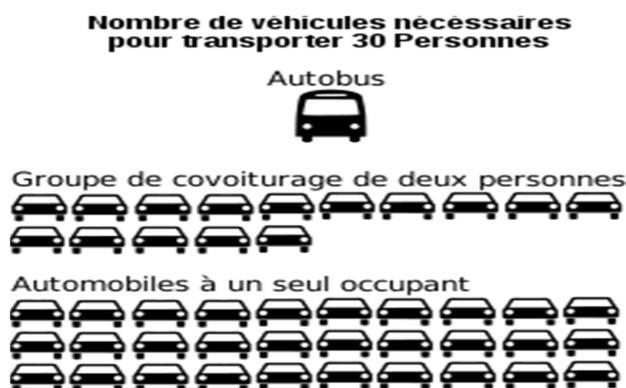


Figure I.16. Transporter 30 personnes en autobus, voitures personnelles ou covoiturage ?

Passer de un à deux passagers en voiture réduit déjà de moitié le nombre de voitures en circulation. La répercussion est sans précédent et tombe comme un verdict : le covoiturage est une solution très prometteuse répondant à la problématique actuelle pour un avenir meilleur, propre et « *mobile* ».

**Covoiturage :** Le covoiturage (dit *carpooling*, *ride-sharing* ou encore *lift-sharing* en anglais) se réfère à l'utilisation partagée d'une voiture par un conducteur non professionnel et un ou plusieurs passagers généralement pour effectuer des déplacements ensemble<sup>19</sup>. Selon le concept énoncé du covoiturage, deux façons subsistent pour utiliser le ou les véhicules dans ce contexte :

- Un véhicule différent de l'un des covoiturés est utilisé à chaque fois, les trajets sont ainsi effectués à tour de rôle en utilisant l'un des véhicules propriétaires. Chaque individu intégrant un groupe de covoiturés est ainsi alternativement une fois conducteur, les autres passager.
- Un seul véhicule, propriété de l'un des covoiturés, est utilisé. Dans ce second cas, les autres passagers doivent contribuer chacun aux frais de déplacement, à savoir le carburant et éventuellement le péage pour effectuer le trajet en question. Une participation forfaitaire est ainsi calculée selon le montant total des charges et le nombre de passagers au total sur le même trajet. Les usagers de ce service s'en voient

<sup>19</sup> <http://www.answers.com/topic/carpool>

ainsi réduire les frais de transport grâce à cette tarification avantageuse. Avantage non négligeable vu la cherté de la vie à laquelle les coûts de transport ne sont pas étrangers. Toutefois, il n'est pas envisageable que des voyages organisés dans ce cadre deviennent une source de profit pour l'utilisateur mettant son véhicule personnel à disposition des autres. En effet, le conducteur ne doit en aucun cas faire de bénéfices mais peut accepter une participation financière telle que définie précédemment. Dans le cas contraire, il prendrait le statut de transporteur au titre de la LOTI (1).

**Covoiturés :** ce terme désigne les passagers d'une voiture de covoiturage. A la base piétonnes en quête d'une éventuelle offre de covoiturage pouvant les ramener à un endroit précis, ces personnes peuvent posséder ou non leurs propres voitures. Elles sont ainsi définies comme intervenants du service initiant des demandes pour se faire conduire entre deux points donnés dans le cadre d'un déplacement souhaité.

**Demande :** un déplacement en covoiturage ne peut se faire que si une demande existe. Il s'agit ici d'une requête d'utilisateur émettant le souhait de se déplacer en voiture d'un endroit à un autre. Une demande concerne un besoin spécifique de déplacement en fonction duquel l'utilisateur détermine la date et l'heure du voyage, l'endroit où il veut aller, etc.

**Covoitureurs :** il s'agit ici des conducteurs d'un véhicule dans le contexte de covoiturage avec un ou plusieurs autres passagers. Ces usagers du service utilisent dans le cas général leur propre véhicule ou parfois des voitures de location. Dans ce dernier cas, les frais de location incombent aussi à tous les passagers du véhicule en question. Par ailleurs, l'initiateur du voyage, à savoir le conducteur, a ses propres besoins de déplacement qu'il définit à travers la soumission d'une offre de covoiturage.

**Offre :** elle désigne les paramètres de déplacement du conducteur (généralement propriétaire de la voiture utilisée pour le covoiturage). Ces paramètres définissent les spécificités du trajet à parcourir : origine, destination, date, heure de départ, nombre de places disponibles, etc.

**Contraintes de correspondance :** covoiturer n'a un sens et n'a lieu d'être que si certaines conditions sont satisfaites. Dans les systèmes de covoiturage classiques, seules des contraintes basiques sont vérifiées. Comme le montre la Figure I.17, ces contraintes se réfèrent essentiellement à la correspondance entre offre et demande<sup>20</sup> permettant ainsi de vérifier les :

---

<sup>20</sup> <http://www.carpoolglobal.com>

- Dates et heures des voyages offert et requis ; celles-ci doivent se rapprocher ou encore mieux, se confondre ;
- Coïncidences entre les trajets ; le déplacement concernant la requête du covoituré doit de ce fait appartenir à l'itinéraire à parcourir par le covoitureur pour que sa demande puisse être considérée. Ceci n'exclut pas le fait qu'un petit détour peut être toléré pour effectuer les opérations de prise ou dépose, et ce selon les motivations du covoitureur et des autres passagers ;
- Nombres de places disponibles dans une voiture de covoiturage qui doivent être suffisantes pour pouvoir accueillir les usagers demandeurs du même trajet
- Etc.

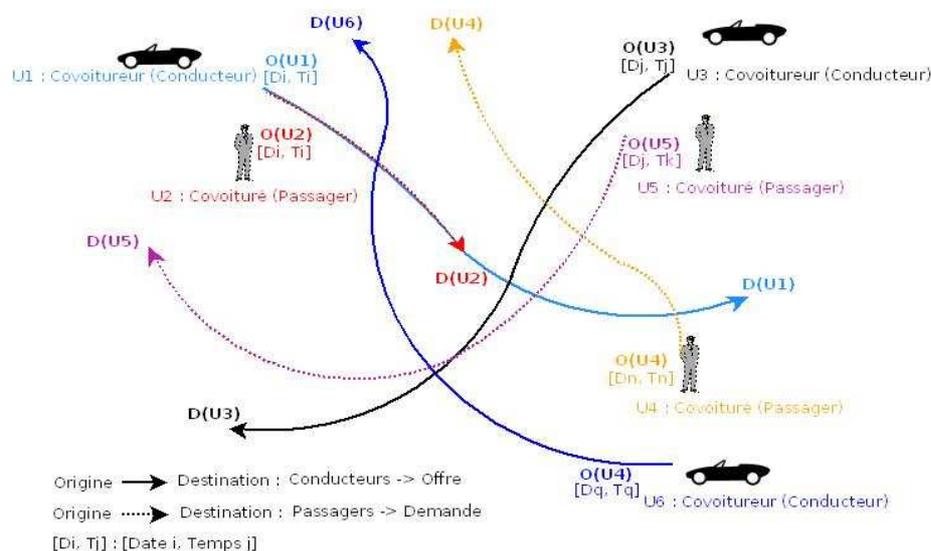


Figure I.17. Notions de base pour le covoiturage

Selon le contexte et les distances à parcourir, le covoiturage prend des formes quelque peu différentes et des appellations distinctes. Nous distinguons plusieurs types de covoiturage dont trois essentiellement retiennent notre attention et sont définis dans la suite.

### I.5.2.2. Différents types de covoiturage

Les arrangements et régimes de covoiturage impliquent des degrés divers de formalité et de régularité engendrant sa dissociation en trois types principaux<sup>21</sup>.

<sup>21</sup> <http://www.covoiturama.com>

**A. Covoiturage régulier :**

Ce type désigne les voyages et trajets qui sont d'une fréquence régulière. Il s'agit en effet des déplacements qui se font de manière hebdomadaire, c'est-à-dire au moins une fois par semaine. Le covoiturage régulier désigne aussi des voyages plus fréquents pour englober même ceux ayant lieu tous les jours (i.e. quotidiens). Nous faisons ici référence aux déplacements nécessaires pour rejoindre son domicile ou aller à son travail (i.e. domicile-travail, travail-domicile). De manière similaire, les trajets reliant le lieu de résidence à celui relatif aux études (i.e. trajets scolaires) s'inscrivent dans le contexte du covoiturage régulier. Il s'agit donc de trajets usuels prenant généralement effet à des dates et heures connues et concernent dans la plupart des cas des distances courtes. Bien avant que le covoiturage existe, ce concept de « partage de trajets » était bien développé entre le personnel d'une même entreprise, groupe d'étudiants fréquentant la même université ou école, etc.

**B. Covoiturage occasionnel :**

Ces trajets n'ont, comme leur nom l'indique, lieu que si l'occasion se présente ; c'est-à-dire qu'ils n'ont pas (ou peu souvent) vocation à se répéter dans le temps. Il s'agit essentiellement de trajets de longue distance, généralement supérieure à 50 kilomètres. Nous pouvons en citer à titre d'exemple les déplacements ayant lieu aux départs des vacances ou encore les événements à travers l'Europe (soirées, concert, festival...). Le covoiturage occasionnel est la forme la plus commune et aussi la plus connue parmi les trois types qui existent. Ce type particulier concerne principalement les voyages entre les villes (ou pays) pour des trajets de longue distance planifiés bien à l'avance.

**C. Covoiturage événementiel :**

Tel que leur nom l'indique, les trajets faisant partie de cette catégorie, n'ont lieu d'être que lorsqu'un événement particulier se produit. Les services de covoiturage sont particulièrement utiles dans ce contexte puisque permettant de déjouer les problèmes induits par la forte densité de véhicules dans les endroits accueillant de tels événements. La flexibilité dans l'accès au véhicule est tout aussi bien maintenue, tout en diminuant le nombre de voitures sur la route pour ainsi mieux en gérer la circulation, la logistique, l'accessibilité aux places de parking, etc. Le covoiturage dans ce contexte sert à diminuer l'afflux d'un grand nombre de voitures à un même endroit et ainsi fluidifier le trafic dans les endroits ou routes concernés et où une forte concentration peut engendrer de multiples problèmes. Le **covoiturage de crise** qui est nettement moins répandu et beaucoup plus informel s'inscrit aussi dans le cadre du

covoiturage événementiel. Il s'agit plus précisément dans ce cas du recours au concept de covoiturage pour pallier des problèmes d'absence d'autres alternatives. C'est-à-dire que les moyens de locomotion pris habituellement par les individus concernés ne sont plus accessibles pour des raisons externes à leurs volontés (panne, grèves des transports collectifs, accident corporel, etc.). Ceci rejoint l'idée de la nécessaire intégration du covoiturage dans un contexte de transport comodal pour remédier à la problématique des perturbations aujourd'hui d'actualité.

Le concept de covoiturage ayant connu un succès fulgurant auprès des responsables concernés par la mise en place d'un processus de développement durable, il a captivé l'attention de plusieurs chercheurs et praticiens et a été abordé dans plus d'un contexte. De multiples travaux ont de ce fait émergé aidant à en faire une évolution assez rapide mais pas toujours réussie. Dans la suite, nous passons en revue les principaux systèmes qui ont fait l'objet de recherches et investissements plus ou moins importants.

### *1.5.2.3. Covoiturage : Un état de l'art riche mais pas autant*

Sur la base de l'apport au départ théorique qu'il a prodigué aux responsables concernés par la mise en place d'une mobilité avancée et soutenable dans le cadre de la concrétisation du processus de développement durable ; le covoiturage a été, depuis un petit peu plus de trois décennies, d'un grand intérêt. Particulièrement durant ces dernières années, le concept de covoiturage a brillé de par l'afflux de plus en plus important des individus à de tels services.

En France par exemple, la prise de conscience collective de la nécessité de mettre en place un processus de développement durable a donné une impulsion supplémentaire aux modes de transport alternatifs. C'est dans ce cadre que de nombreuses initiatives localisées ont émergé sous la forme d'associations dans le cas général. Ayant bénéficié d'appui technique et financier de la part de conseils généraux et régionaux, des Autorités Organisatrices des Transports (AOT) ou encore de l'Agence de l'Environnement et de la Maîtrise de l'Energie (ADEME), ces initiatives n'ont su que porter leurs fruits. Dans ce sens, le secteur des transports étant en grande partie responsable dans les émissions de GES, accaparant la majeure proportion de rejets de CO<sub>2</sub>, ce constat alarmant a donné une impulsion stratégique aux modes alternatifs les propulsant au-devant de la scène.

Faisant désormais partie des concepts réussis de premier rang vu les avantages qu'il offre (40), le covoiturage a ainsi été abordé dans plus d'un contexte donnant lieu à une multitude de

projets industriels et travaux théoriques qui en solidifient les bases pour une meilleure évolution et incrustation en tant que facteur de mobilité avancée.

Des structures formelles de covoiturage ont commencé à se mettre en place dès le milieu des années 70s (50). Compte tenu des importantes contributions qu'elles ont apportées, des chercheurs et praticiens se sont focalisés sur les services de covoiturage. Plusieurs de ces travaux ont par la suite conduit à la mise en place de systèmes fonctionnels qui ont été déployés et testés. Cependant, il existe aussi des systèmes qui n'ont jamais abouti faute de moyens ou d'autres aspects auxquels les utilisateurs aspirent, tels que la sécurité par exemple. Des solutions innovantes ont aussi émergé faisant écho avec les innovations technologiques qui jouent un rôle principal dans leur bon fonctionnement. Parmi celles-ci, nous pouvons citer les Nouvelles Technologies de l'Information et de la Communication (NTIC), Internet, les systèmes embarqués, les systèmes d'information géographique (GIS), les logiciels d'optimisation d'itinéraires ou encore les logiciels de management environnemental (1), etc. qui contribuent à rendre ces solutions attractives et dynamiques.

Dispersées à travers le monde entier, ces solutions ont commencé à immerger aux États-Unis avec le premier choc pétrolier de 1973<sup>22</sup> donnant naissance à une multitude d'associations dont une mission principale était d'inciter les individus (étudiants et salariés) à se partager leurs véhicules sur leurs trajets quotidiens. Une adaptation de l'infrastructure routière (construction de voies réservées aux véhicules à fort taux d'occupation (HOV lanes<sup>23</sup>)) s'inscrit dans une politique incitative servant à encourager les individus à aller vers ce mode. Parallèlement un projet similaire baptisé ICARO pour Increase Car Occupancy a vu le jour en Europe mais qui a débuté assez tardivement par rapport au premier (1997) et s'est achevé en 1999 (1).

Ce dernier a débouché sur un nombre d'initiatives (51) encourageant l'implantation du covoiturage en Europe ; parmi celles-ci des parcs de stationnement aux nœuds routiers en Autriche, un programme de stationnement réservé pour les véhicules à taux d'occupation élevé en Suisse, etc. En France aussi, le covoiturage a eu son lot d'intérêt, qui malgré son arrivée tardive (avec les grèves importantes de 1995 provoquant une paralysie dans les transports en commun) a su porter ses fruits. Les résultats furent d'autant plus importants qu'à partir des années 2000, les collectivités locales ont commencé à fournir leur soutien, essentiellement financier, pour encourager l'implantation du covoiturage comme service à

---

<sup>22</sup> <http://www.retail-and-co.com/article-22248232.html>

<sup>23</sup> <http://www.mto.gov.on.ca/english/traveller/hov/>

part entière dans le secteur du transport. Parmi les initiatives prises dans ce contexte, nous pouvons citer à titre d'exemple, le projet de l'autoroute A14 devenue gratuite pour les véhicules à fort taux d'occupation.

Ces initiatives ont de ce fait réussi à faire basculer le monde des pratiques de « *covoiturage spontané* » plus connu sous le nom de « *auto-stop* » vers des pratiques beaucoup plus évoluées dans le sens où elles sont bien organisées (« *covoiturage organisé* »). Ce concept a en effet commencé à prendre forme dès les années 50 avec l'émergence de solutions telles que Taxistop en Belgique, Allostop en France (1958) ou bien avant Allo-Stop au Canada et qui consistaient concrètement en une dynamique d'organisation des pratiques de l'auto-stop.

Orientées dans cette direction, des associations, organismes, agences et collectivités publiques sont nées pour donner l'aspect formel que requiert cet aspect organisationnel. En effet, faisant le rôle d'entités intermédiaires, ces organes sont responsables de la mise en relation d'éventuels covoiturés. Ces entités peuvent aussi bien être des centrales de mobilité, ou encore des particuliers et proposent un panel de services plus ou moins complet sur la base de leurs structures, financement et aussi de leurs connaissances de l'outil support à ces services. Dans la majorité des cas, il s'agit de services fournis par téléphonie mobile ou Internet. Il existe de ce fait des réseaux sociaux tels que les forums ou autres accessibles via les canaux virtuels. Par ailleurs, les services les plus en vogue et aussi les plus répandus sont plutôt sous forme de sites Internet. Le nombre de services utilisant de tels supports en France a atteint 78 ou plus en Mars 2007 (1) réalisés entre autre par des opérateurs de covoiturage tels que Green Cove, Ecolutis, LaRoue Verte ou encore covoiturage.fr. Entre *sites de covoiturage grand public* (ouverts à tout individu) et *sites de covoiturage à accès restreint* (nécessitant un code d'accès), ces derniers mettent en place des fonctionnalités plus ou moins évoluées.

Les travaux relatés dans la littérature ont émergé dans deux formes différentes du concept : dynamique ou statique selon qu'elles offrent des services d'accès et de gestion instantanés ou pas. Des documents bibliographiques (1) (50), recensant la majorité des études menées dans le cadre du covoiturage à travers le monde entier, nous ont donc permis de relever, dans ce cadre, deux grands volets relatifs aux aspects de gestion des réservations et des accès au service de covoiturage.

**A. Le covoiturage statique :**

La majorité des systèmes existants font partie de cette catégorie et se présentent dans le cas général sous forme de systèmes non automatisés, accessibles via le web, intégrant tout au plus les fonctionnalités de gestion des offres et demandes de covoiturage. L'utilisation de tels supports en est ainsi simplifiée ; elle nécessite une inscription préalable afin de pouvoir avoir accès aux différentes fonctionnalités offertes par le biais de ces supports. Malheureusement, ces fonctionnalités se résument dans le cas général à un simple dépôt ou consultation d'une offre ou demande de covoiturage selon que l'internaute soit covoitureur ou covoituré. En effet, ces sites web, dont la plupart sont dynamiques disposant d'une base de données, ont pour fonctionnalité principale de sauvegarder les offres et demandes en vue d'une éventuelle demande de consultation. Ils doivent donc être dotés de capacité de stockages nécessaires en adéquation avec les besoins de sauvegarde.

Dans ce cadre, des systèmes tels que 123envoiture<sup>24</sup> et aide-covoiturage.com<sup>25</sup> se présentent sous forme de sites Internet permettant aux personnes intéressées de s'inscrire et de consulter/chercher des demandes/offres de covoiturage et de récupérer les données personnelles d'un utilisateur présentant une alternative qu'elles ont jugée intéressante. La prise de contact entre les usagers du service se fait généralement en dehors du contexte du site Internet support du service : plutôt via mail ou par téléphone. Ceci n'empêche pas le fait qu'un service de rencontre direct soit intégré à ces systèmes. Il existe en effet d'autres types de sites Internet, tels que comove<sup>26</sup>, qui présentent l'avantage de permettre à leurs utilisateurs de communiquer via des forums ou chatrooms. Cependant, rares sont les sites Internet qui proposent une mise en ligne de leurs abonnés leur permettant de rentrer directement en contact pour se mettre d'accord sur les détails d'un éventuel voyage commun.

Malgré leur réussite plus ou moins relative, ce type de systèmes reste néanmoins très ouvert à de possibles métamorphoses évitant de par là le piège de la stagnation. La révolution technologique qu'a connue le monde ces dernières années n'a eu de fait que de propulser ces systèmes vers une nouvelle aire, révolutionnaire et très pragmatique : l'aire du temps réel.

**B. Le covoiturage dynamique :**

Il s'agit d'une tendance tout à fait nouvelle dans le domaine du covoiturage, une tendance qui propose de gérer les accès instantanés des usagers aux offres et demandes disponibles,

---

<sup>24</sup> <http://www.123envoiture.com/>

<sup>25</sup> <http://www.aide-covoiturage.com>

<sup>26</sup> <http://www.comove.com/forum-covoiturage/index.php>

actualisées en temps réel. Le boom dont témoigne le domaine des technologies avancées n'est pas du tout étranger à ce subit intérêt porté à l'intégration de l'aspect temps réel dans des services aussi complexes que le covoiturage. En effet, avec les smartphones, les PDAs, les tablettes PC, etc. munis de technologies Blue Tooth, GPS, GPRS... il est devenu possible de bénéficier d'applications mobiles intégrées grâce à des systèmes embarqués sur son propre support. Ces technologies révolutionnaires sont très pertinentes pour la mise en œuvre de mesures de performances et d'efficacité dans un contexte de mobilité continue. Ceci étant, des fonctionnalités telles que la géolocalisation instantanée et continue, l'Internet mobile, la communication en temps réel, etc. jadis impensables, sont devenues aujourd'hui banalisées avec la réussite explosive de ces technologies.

Dans ce contexte, des projets soutenant cette nouvelle initiative ont émergé et des associations à l'instar de Covivo<sup>27</sup> en région Lorraine en France, ont vu le jour. Covivo est une start-up incubée à Sciences Po qui, le soutien d'Oseo et de l'ADEME aidant, s'est lancée sur la part du marché du covoiturage dynamique en France pour proposer à ses usagers un accès instantané à l'information disponible. Dès qu'ils sont connectés au réseau Covivo, les potentiels covoitureurs et covoiturés sont ainsi informés des possibilités de covoiturage en temps réel.

Par ailleurs, malgré l'avancement technologique et les efforts déployés dans ce sens, la majorité des approches existantes implémentant des services en temps réel tels que GoLoco, Easy-Rider, T.écovoiturage... (50) sont restées au stade embryonnaire à cause de failles de sécurité, d'automatisation ou à cause d'aspects d'optimisation leur faisant défaut. D'autant plus qu'ils sont pratiquement similaires à ceux précédemment décrits, à la seule différence qu'ils présentent l'avantage de consulter en temps réel la liste des offres des véhicules en circulation. L'intégration de services de géolocalisation est à la base de la possibilité de mise en place de tels services. Les fonctionnalités supplémentaires des services de covoiturage dynamique ainsi définis se résument globalement à la récupération de données GPS actualisées et de les fournir à leurs abonnés.

### *1.5.2.4. Covoiturage : Limites et restrictions*

Ayant plus ou moins réussi, les systèmes existants ont au moins le mérite d'avoir fait découvrir au grand public l'existence des opportunités de partager son véhicule.

---

<sup>27</sup> <http://www.covivo.eu/>

Malgré le progrès important réalisé dans le domaine du covoiturage, les systèmes existants demeurent quelque peu défaillants. Défaillance technique, informationnelle, technologique ou autre, celle-ci va à l'encontre de leur succès absolu puisque imposant des limites grotesques dans la libre utilisation de tels services. En effet, les systèmes les plus réussis jusqu'à ce jour étant ceux qui se présentent sous forme de sites Internet dynamiques (*c.f.* I.5.2.3), ils présentent un modèle commun lourds d'handicaps en termes de fonctionnalités et surtout d'automatisation de celles-ci. Ce modèle présenté à la Figure I.18, fournit aux usagers des services qui répondent à celui-ci, l'opportunité unique de pouvoir insérer ou consulter des spécificités de déplacements pour (ou sur) des trajets de covoiturage. Répondant à la qualification de concept statique, la majorité des services de covoiturage aujourd'hui dominants imposent des limites d'utilisation. Les restrictions induites de par ce concept touchent principalement aux possibilités d'extension de ces services inhibant leur aptitude à pouvoir intégrer l'instantanéité des accès et donc la gestion de ceux-ci en temps réel. Les usagers de ces systèmes se voient ainsi obligés de planifier leurs voyages plus ou moins longtemps à l'avance afin de pouvoir augmenter leurs chances de réussite à dénicher des offres ou demandes de covoiturage qui correspondent aux leurs.

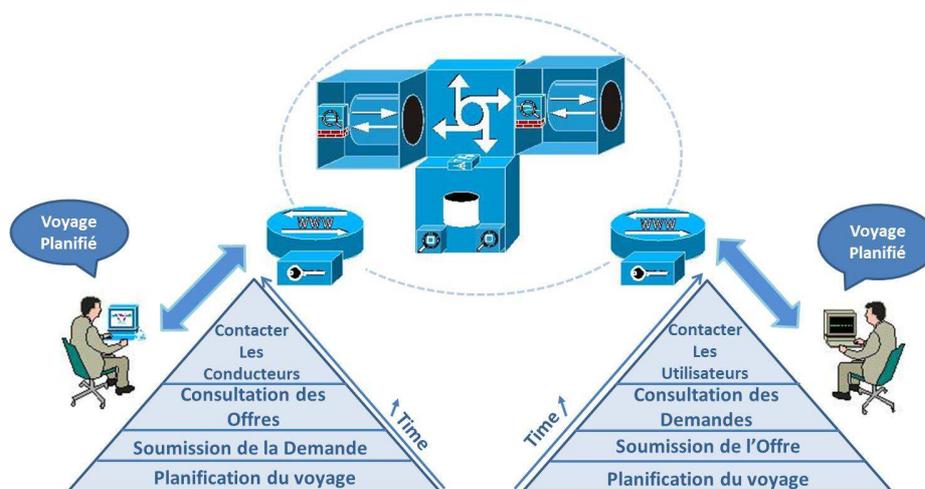


Figure I.18. Limites des systèmes de covoiturage existants

Par ailleurs, outre la dominance du concept de gestion statique dans les systèmes de covoiturage existants, de grandes questions notamment par rapport à l'automatisation des tâches demeurent sans réponse. Même si quelques systèmes dynamiques de covoiturage ont par la suite existé (e.g. eNotions en Allemagne, EasyRider à Amsterdam, RideNow à Los Angeles, T.écovoiturage en France, etc.), ils n'apportent pas non plus les réponses nécessaires à ces controverses. Si ce n'est la question du temps réel, ce sont les questions de sécurité et de traçabilité qui continuent de faire obstacle à la réussite absolue des systèmes existants et

entraver leur ascension à l'échelle. C'est à partir des fondements de ces affirmations récidives et dans une tentative d'apporter les réponses adéquates à des interrogations longtemps restées en suspens que s'inscrivent nos travaux.

## **I.6. Proposition d'une approche optimisée pour le covoiturage dynamique**

En pratique comme en théorie, le covoiturage est un concept riche en apport. De lourde incidence touchant à plusieurs aspects, le covoiturage a des portées positives sur l'environnement, les comportements des individus, leurs relations sociales, l'économie, les budgets, etc. et mérite donc d'être considéré avec grand intérêt. De plus en plus convoités, les systèmes de covoiturage font le succès des services de transport offrant la satisfaction de la complétude qui lui faisait défaut. Le concept de covoiturage accapare aujourd'hui une attention bien particulière doublée d'un grand intérêt pour l'évolution dans ce domaine. Nos travaux de recherche s'inscrivent dans ce cadre et tentent une poussée technologique pour essayer de gagner du terrain par rapport aux travaux recensés dans la littérature. Nous proposons de ce fait un système de covoiturage dynamique optimisé œuvrant à grande échelle et compétitif à l'échelle internationale. L'étude des systèmes existants nous ont amenés à en définir les grands axes conduisant à la mise en œuvre des fondements de notre approche. Ces axes principaux ont, en effet, été inspirés des problèmes et lacunes ayant émergé de par l'étude des approches existantes. Nous définissons alors dans ce qui suit nos principales motivations justifiant le grand intérêt que nous y portons. Une description globale de notre approche est par la suite fournie dans la section suivante. Optimisation et temps réel en sont les maître mots, et ce dans le but essentiel de pouvoir atteindre la qualité de service escomptée.

### **I.6.1. Contexte et Motivations**

Adopter le covoiturage comme solution de transport à part entière ou solution d'appoint pour remédier à un manque de mobilité ou irrégularité dans les modes de transport existants (nous parle ici du covoiturage de rabattement) a le mérite d'apporter les réponses à des questions longtemps restées en suspens. Nous faisons référence ici aux problèmes relatifs au secteur de transport qu'ils soient d'ordre écologique, financier, sociétal, ou autre.

Grâce au succès important et de surcroît évolutif du phénomène de covoiturage, plusieurs chercheurs ont concentré leurs efforts sur ce concept (*c.f. I.5.2.3*). En dépit de la présence

imposante de la « *Technologie* » dans les systèmes existants, la plupart d'entre eux demeurent insatisfaisants vu le manque de motivation des individus à aller vers de tels systèmes. Leurs multiples inconvénients sont notamment à l'origine de ce refoulement. En effet, la majorité des approches existantes présentent de multiples lacunes de première importance (*c.f. I.5.2.4*): disfonctionnement, manque d'automatisme, insécurité, défauts d'accessibilité, etc. De plus, la majorité des opérateurs de covoiturage lésinent sur les fonctionnalités et fournissent des services aussi basiques que possible. La qualité des services fournis en sont donc affectés et se trouvent dépourvus d'originalité, et parfois même de l'essentiel (automatisation des tâches, gestion en temps réel...).

Dans ce contexte, seule une minorité de travaux dans le domaine de la recherche dans ce contexte se sont focalisés sur l'aspect dynamique de ces services. En plus d'être peu, les travaux conçus et spécifiés dans un cadre de gestion en temps réel sont pour la plupart restés au stade de l'ébauche ou de l'idée. De plus, une grande partie de ceux qui ont abouti, n'ont pas longtemps été opérationnels et visent à peine une petite communauté. Cet échec d'une tentative d'évolution pourtant de bon augure dans le processus de développement durable a fait se précipiter le concept de covoiturage dans le piège de la stagnation.

Dans l'optique de combler le déficit et remédier aux problèmes des systèmes existants, nous introduisons une nouvelle approche que nous baptisons **CODAC**, acronyme de **Covoiturage Optimisé Dynamique** basé sur les **Agents Communicants**. Comme son nom l'indique, cette approche a pour but essentiel de mettre en place un système de covoiturage dynamique optimisé où nous proposons de nous focaliser sur une multitude de tâches. Celles-ci seront intégrées dans un système complet basé sur un support totalement automatisé. L'intégration de ces tâches serait réalisée en fonction des lacunes des travaux existants et dans le but premier d'y remédier et fournir un système aussi efficace et performant que possible. L'aspect temps réel requiert par ailleurs de répondre à une condition primordiale qui est celle de fournir aux utilisateurs des réponses optimales en termes de temps de réponse. L'optimalité des réponses fournies étant aussi une question d'actualité, nous fournissons dans ce qui suit une brève description de notre système et des fonctionnalités qu'il englobe.

### I.6.2. Description du système

En vue de répondre au mieux aux attentes des usagers du service, l'optimalité de ce dernier réside aussi bien dans leur satisfaction par rapport aux réponses fournies. Dans ce cadre, la prise en compte de critères d'optimisation (distance à parcourir, émissions de CO<sub>2</sub>...)

s'impose. Nous délaissions ici le concept d'optimisation, traité dans les *Chapitre II et III*, au gré d'une explication de notre choix par rapport à l'intégration de la dynamique des traitements dans notre approche.

Joint au concept de l'optimisation, l'aspect dynamique nécessite de disposer du matériel nécessaire pour pouvoir disposer de l'information pertinente et actualisée en continue. Parmi le matériel nécessaire, des supports équipés d'un module GPS s'avère alors de première utilité pour mener à bien cette fonctionnalité. La Figure I.19 illustre une vue réelle de notre système représentant la dispersion des covoitureurs et covoiturés, les différents liens susceptibles d'exister entre eux et aussi qui doivent les relier au système (serveur de covoiturage), etc. Cette figure montre aussi l'équipement nécessaire : entre PDA, ordinateurs, et téléphones (ou smartphones) qui doivent notamment être équipés de modules de géolocalisation GPS en référence aux systèmes GIS.

Cette utilisation quelque peu massive des technologies évolutives de l'information engagerait en temps normal un investissement important en termes de matériel et imposerait certaines contraintes aux individus pour pouvoir utiliser ce service. Par contre, nous pouvons prétendre au succès de ce système, ces contraintes étant aujourd'hui illusoire vue la présence imposante en terrain acquis de ces matériels aussi avancés soient-ils dans les sociétés. Les PDA et Smartphones ont en effet aujourd'hui envahi le monde et plusieurs d'entre eux sont abordables et ne nécessitent pas d'importants investissements.

Ceci étant, l'acquisition de l'information actualisée sur les positions géographiques des covoitureurs et covoiturés en temps réel devient accessible. Les outils GPS intégrés dans notre système et équipements rendent en effet cette action banale. Une action qui n'est toutefois possible que sous réserve que les utilisateurs du système aient accepté d'être instantanément localisés. En effet, afin de demeurer dans un contexte judiciaire propre et hors d'atteinte aux droits à la vie privée des personnes au regard de la Commission Nationale de l'Informatique et des Libertés (CNIL), une charte d'utilisation où des clauses spécifiques y faisant référence doit être signée lors de tout abonnement au service. Ces clauses symboliseront le droit cédé au système par les utilisateurs qui en acceptent les termes afin que ce dernier puisse accéder à leurs données géographiques dès leur connexion au réseau *CODAC* et jusqu'au moment où leurs équipement s'en déconnectent.

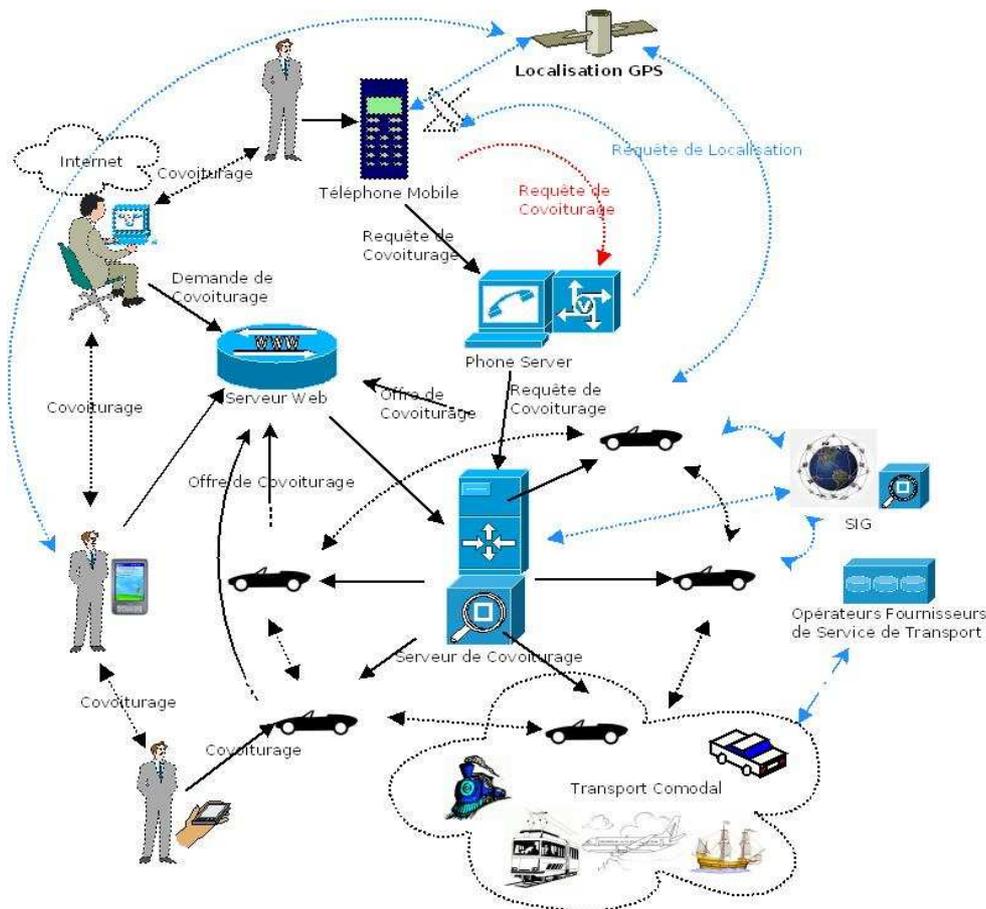


Figure I.19. CODAC : Un système de covoiturage dynamique optimisé compétitif à grande échelle

Telle qu'elle est définie, l'approche que nous proposons initie le covoiturage à un mix de concepts intégrant plusieurs volets qui, dans leur mélange, constituent une nouveauté :

- Géolocalisation en temps réel des usagers du service
- Traitements dynamiques des offres et demandes de covoiturage
- Affectation dynamique prenant en compte les voitures en cours de circulation aussi bien que les voitures n'ayant pas encore entamé leurs trajets
- Gestion des contraintes d'affectation (capacité des véhicules Vs nombre de passagers, date et heure de la demande Vs date et heure de l'offre de covoiturage, spécificité du trajet du conducteur Vs besoin de déplacement du piéton ...) pour éviter les situations de conflit
- Optimisation dans les traitements effectués par le système : aspect technique relatif aux propriétés d'exécution (i.e. délai de réponse, utilisation de l'espace mémoire, ressource CPU)

- Optimisation des affectations des covoiturés aux voitures : Quels critères pour mieux satisfaire ?
- Géolocalisation continue tout au long des trajets, et de manière plus générale tout au long de la période de connexion des usagers du service. Ce module s'avère être très utile vu le refoulement des individus des systèmes de covoiturage existants ; la sécurité leur faisant grand défaut. Outre l'aspect rassurant de ce module, qui fournit le degré de sécurité requis, la traçabilité des routes et trajets de covoiturage effectués devient abordable. De par cette traçabilité, nous visons une meilleure gestion du service pour en prévenir les variations et irrégularités sur la base de statistiques faites sur les offres et demandes de covoiturage
- Module de Gestion statique des offres et demandes de covoiturage
- Etc.

La Figure I.20 reprend globalement les principaux modules intervenant dans notre approche.

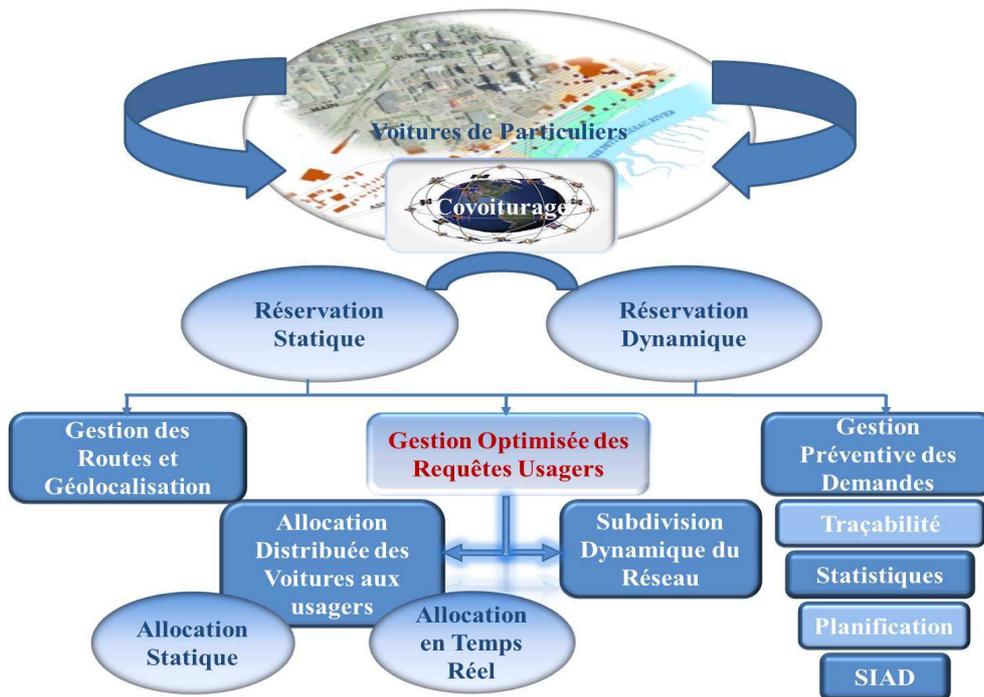


Figure I.20. CODAC : Les principaux modules

## I.7. Conclusion

Il est vrai que les systèmes de partage de véhicules aident à l'amélioration des conditions environnementales dans lesquelles évoluent les individus. Le concept de covoiturage présente

particulièrement de multiples avantages, dont la réduction du nombre de voitures en circulation au kilomètre. En effet, la voiture personnelle est devenue, grâce à ce concept, un moyen de déplacement collectif accessible au grand public. Ces systèmes représentent en ce sens un moyen efficace pour minimiser le taux d'émissions de CO<sub>2</sub> et gaz à effet de serre aussi bien que pour réduire les budgets alloués au transport.

Solution à part entière ou solution d'appoint comme dans le cas des travaux de M.F. Feki (39), le covoiturage apporte la réponse adéquate à des interrogations longtemps restées en suspens. C'est dans ce cadre que nous avons dirigé nos travaux pour proposer une approche innovante dont les fondements ont été construits sur la base d'une étude approfondie de l'histoire du covoiturage et des principes et concepts faisant défaut aux systèmes érigés de par le développement de ce phénomène.

Notre système, baptisé CODAC pour Covoiturage Optimisé Dynamique basé sur les Agents communicants, intègre comme son nom l'indique le concept d'optimisation dans une alliance avec les Systèmes Multi-Agent (SMA). Cette alliance vise la mise en place de traitements efficaces dans un système performant fournissant une qualité de service optimale. Par ailleurs, traiter une problématique d'optimisation dans un contexte d'affectation en temps réel engendre une complexité d'ordre exponentiel qui fait entrave à la conduite à bon escient d'un système doté d'une performance optimale sur tous les niveaux (*c.f. II.3.2*). Par conséquent, nous ne pouvons prétendre à l'optimalité dans tous les sens et sous tous les points de vue mais cherchons les meilleurs compromis possibles : évitant les situations conflictuelles, s'exécutant dans des délais de réponse raisonnable, fournissant des réponses optimisées, etc. Pour ce faire, nous sommes allés puiser dans les théories des graphes, les méthodologies de résolution basées sur l'intelligence artificielle distribuée, les méthodes et algorithmes de recherche opérationnelle, les algorithmes d'optimisation multi-objectif, les SMA, etc. Tous ces concepts ont été consacrés et utilisés dans un seul sens où nous avons dirigé nos travaux. Dans cette perspective, nous visons en effet de mettre en place un processus d'affectation dynamique optimisé décentralisé basé sur la subdivision du réseau de desserte. Une mesure de performance efficace pour une qualité de service optimisée et un système compétitif prime sur tout autre objectif dans la vision où nous menons nos travaux. Toutes ces notions ainsi que la manière dont elles opèrent pour la bonne marche de notre système seront détaillées dans le chapitre suivant.

II. CHAPITRE II

LES SYSTEMES MULTI-AGENTS ET  
L'OPTIMISATION :

UNE ALLIANCE VERS UNE METHODOLOGIE DE  
RESOLUTION EFFICACE

## II.1. Introduction

Sur la base de la problématique précédemment énoncée (*c.f. Chapitre I*), nous orientons notre étude vers les fondements essentiels de l'optimisation. Nous rejoignons dans cette mission les systèmes invoquant la qualité de service comme objectif indiscutablement essentiel et récurrent dans le domaine du transport. En effet, la question omniprésente de la qualité de service est indéniablement indispensable pour renchérir l'intérêt d'utiliser de tels systèmes et ainsi faire leurs réussites.

Dans ce contexte, nous avons choisi de positionner nos travaux dans un cadre riche en apports principalement relatifs à l'optimalité de la qualité de service au profit des utilisateurs de notre système. Ce dernier se présente sous forme de support informatique pour le covoiturage dynamique et dont le rôle principal est d'exécuter la recherche des possibilités d'affectations des covoiturés aux covoitureurs. A l'opposé de la plupart des travaux existants où les usagers doivent chercher par leur propre soin une offre ou demande susceptible de les intéresser, le système ici proposé se doit de le faire de manière automatique et en plus performante. Dans ce cadre de mise à l'échelle, nous essayons de joindre les performances techniques à l'optimisation des solutions d'affectation fournies aux usagers au sens d'une qualité de service optimale. L'optimalité favorise en effet la mise en place d'un système indélébile et susceptible de monter à l'échelle de surcroît.

Par ailleurs, la complexité du problème de covoiturage dynamique optimisé ayant été prouvée comme étant exponentielle (*c.f. II.3.2*) (52), nous abordons la nécessité de mettre en place une méthodologie de résolution efficace qui doit coordonner l'efficacité à la performance. Nous introduisons dans ce contexte l'approche que nous proposons pour contrecarrer la problématique de la complexité importante, laquelle se base essentiellement sur les concepts combinés de l'optimisation, du paradigme agent et de la théorie des graphes.

Nous verrons donc dans ce chapitre l'enchaînement dans lequel ont été entraînés nos travaux d'optimisation en covoiturage dynamique. Les problèmes de complexité érigés de par ce concept ont nécessité la mise en place d'une stratégie de résolution efficace afin d'inhiber les séquelles engendrées de par cette problématique. Cette efficacité a été vue en l'adoption d'un processus décomposé et décentralisé plaçant nos travaux dans un contexte de « *multi-threading* ». Plusieurs sous processus seraient ainsi responsables chacun d'une ou plusieurs tâches de complexité nettement inférieure à celle du problème initial. Ces processus légers, dits « *threads* », doivent s'exécuter en parallèle afin de bien cadrer avec les mesures de

performances escomptées. Cette stratégie même en a fait émerger plusieurs des concepts impliqués directement ou indirectement dans la mise en place d'un processus distribué. Dans notre cas, des notions, très importantes et d'actualité, telles que les Systèmes Multi-agents (SMA) et l'optimisation distribuée sont directement impliqués dans notre approche. Leur combinaison, efficace dans la réduction de la complexité du problème de covoiturage dynamique, nous a amenés à y porter l'intérêt qu'il se doit. Nous en fournissons de plus amples détails tout au long de ce chapitre et de ceux qui suivent.

Par conséquent nous nous focalisons essentiellement dans notre étude sur :

- La mise en exergue de l'adéquation du choix d'intégration du concept de l'optimisation. Nous en présentons ainsi les fondements dans la troisième section de ce chapitre, un choix qui peut par contre ne pas s'avérer très judicieux si les problèmes qu'il induit ne sont pas traités en premier lieu. Nous évoquerons de ce fait essentiellement un problème qui n'est pas des moindres vu les répercussions qu'il engendre. Il s'agit en effet de la complexité exponentielle du problème de covoiturage dynamique optimisé. Notons par ailleurs qu'il est essentiel de définir le concept d'optimisation en général avec notamment tous les aspects qui y sont liés; la deuxième section en fera effectivement l'objet.
- Les systèmes multi-agents qui constituent un concept agissant à « *part entière* » en faveur de la mise en place d'une méthodologie de résolution basée sur les notions de décomposition et d'optimisation distribuée pour lesquelles nous avons opté. Il est donc fondamental de définir le concept agent avant de procéder à la description de son implémentation (*c.f. Chapitres 4 et 5*) dans notre système. La quatrième section sera ainsi consacrée à cet effet.
- Les Graphes Dynamiques Distribués (*GDD*) qui sont l'essence même de notre approche dont l'idée motrice est la mise en place d'un procédé d'optimisation distribuée au travers d'une architecture multi-agent. Cette architecture étant en effet établie sur la base d'une modélisation du réseau géographique de desserte sous forme de graphe dynamique distribué, nous en définissons alors les fondements dans la cinquième section de ce chapitre. Une étude concise des graphes y est donnée ainsi que les algorithmes d'optimisation recensés dans la littérature et qui sont en concordance avec une telle modélisation. Nous aboutissons de par-là à une introduction à l'approche que nous avons adoptée dans nos travaux pour mettre en place un système de covoiturage dynamique efficace tout en remédiant à sa complexité combinatoire. Dans

ce contexte, le focus est alors mis sur les concepts d'optimisation et du paradigme agents dont le mariage nous a conduits sur le chemin des performances et théoriques et applicatives. Nous concluons finalement à la septième section par un bref résumé reprenant les concepts ici présentés et concrétisant la méthodologie adoptée. Toutes ces notions feront par ailleurs l'objet d'une étude beaucoup plus approfondie dans le reste de ce manuscrit.

### II.2. Optimisation

Visant un système performant et compétitif, nous nous intéressons particulièrement à l'optimisation pour en faire l'axe pivot de notre approche. Dans ce sens, offrir une qualité de service optimisée, si ce n'est optimale, prime sur tout autre objectif, concrétisant ainsi nos centres d'intérêts de par ces travaux. L'optimalité de la qualité de service sous-entend une satisfaction des usagers du système de par l'intégration des fonctionnalités nécessaires. Celles-ci seraient menées d'écho avec leurs attentes et ont pour mission d'accomplir l'affectation des « piétons » aux voitures. Cette affectation se déroule dans le cadre de deux volets essentiels décrits et formalisés dans le *Chapitre III*. Il s'agit dans le premier volet de vérifier la faisabilité des offres des voitures existantes. Par faisabilité nous entendons ici la correspondance entre le trajet offert et les exigences des utilisateurs, notamment par rapport à l'heure du voyage, le nombre de places demandées et celles disponibles en voitures, etc. Sur la base des solutions faisables ainsi extraites, notre système se doit de chercher dans le second volet celles qui optimisent le mieux certains critères fixés préalablement au lancement du processus d'affectation. Le concept d'optimisation a ainsi retenu toute notre attention et nous en étudions ici les bases en vue d'en expliquer le choix d'intégration dans notre approche.

#### II.2.1. Optimisation : Concept et définition

L'optimisation a été introduite dans un souci d'amélioration des services fournis, peu importe le domaine auquel ils s'appliquent. Un problème d'optimisation concerne l'exécution de méthodes spécifiques en quête d'un *optimum*. Ce dernier peut être une valeur maximisant ou minimisant une fonction  $f$ , dite *fonction objectif* ou *fonction de coût* ; elle est encore appelée *critère d'optimisation*.

Dans ses travaux sur l'intégration de l'optimisation multi-objectifs pour les procédés de mise en forme (53), M. Ejday définit le concept d'optimisation comme comprenant deux phases. Celles-ci, telles que illustrées dans la Figure II.1. (54), correspondent à :

- Une première phase de modélisation à travers laquelle la (ou les) fonction(s) coût seraient définies, avec en plus, la détermination des variables principales d'optimisation ainsi que des contraintes d'inégalité et égalité que nous verrons par la suite.
- Une deuxième phase de résolution responsable de la recherche de(s) valeur(s) optimisant la fonction coût ainsi définie et ce grâce à un algorithme d'optimisation élaboré à cet effet (55).

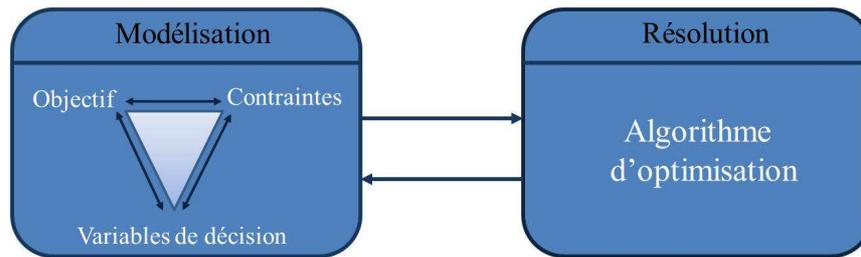


Figure II.1. Modélisation et Résolution pour l'optimisation

Le concept d'optimisation a été intégré dans des services touchant à des domaines divers. Il s'agit de fournir aux usagers de ces services le seuil de satisfaction requis et qui cadrerait avec leurs exigences. Par conséquent, la nécessité de fournir à l'utilisateur un système répondant au mieux à ses attentes est l'essence même de ce concept et dont découle le besoin d'amélioration cherchant à atteindre l'optimalité de la qualité de service. Le degré de satisfaction des usagers est quantifiable au travers de la fonction objectif  $f$  où l'on fait intervenir une ou plusieurs variables appelées *variables de décision*.

La recherche de l'optimum (maximum ou minimum) de  $f$  se fait de par la modification d'une composition de celles-ci. Pour ce qui est de la formalisation d'un problème d'optimisation, elle relève de la formulation mathématique de la minimisation d'une fonction  $f(\vec{x})$  telle que  $\vec{g}(\vec{x}) \leq 0$  et  $\vec{h}(\vec{x}) = 0$ , où :

$$\begin{cases} \vec{x} \in \mathfrak{R}^n : n \text{ Variables de décision} \\ \vec{g}(\vec{x}) \in \mathfrak{R}^m : m \text{ contraintes d'inégalité} \\ \vec{h}(\vec{x}) \in \mathfrak{R}^p : p \text{ contraintes d'égalité} \end{cases}$$

Les contraintes sont initiées à l'ensemble des points  $\vec{x}$  pour lesquelles la valeur de  $f$  peut être déterminée. Ces contraintes sont ainsi utilisées pour en délimiter le champ d'application à un ensemble de points spécifique dit *espace de recherche* ou *espace de valeurs réalisables*.

L'espace de recherche où s'appliquent les méthodes d'optimisation s'en trouve ainsi restreint aux limites imposées par les contraintes d'optimisation.

## II.2.2. Délimitation de l'espace de recherche

L'espace de recherche où s'appliquent les méthodes d'optimisation pour l'extraction d'une solution optimale est restreint et délimité grâce aux limites imposées par les contraintes d'optimisation. Nous distinguons deux types de contraintes d'inégalité :

- Celles de type  $B_{i_{inf}} \leq x_i \leq B_{i_{sup}}$  : elles définissent une plage de valeurs  $\vec{x}$  qui vérifient ces contraintes pour ainsi définir un **espace de recherche** où va s'exécuter l'algorithme d'optimisation. Un exemple en est illustré à la Figure II.2., où 2 variables de décisions sont considérées ( $n = 2$ ) (19).

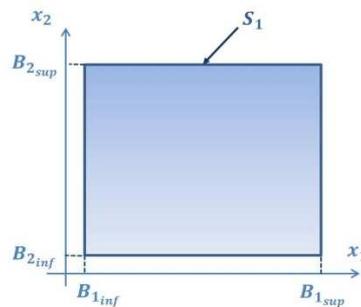


Figure II.2. Espace de Recherche (19)

- Celles de type  $c(\vec{x}) \leq 0$  ou  $c(\vec{x}) \geq 0$  : les valeurs pour lesquelles  $\vec{x}$  vérifie ces contraintes sont considérées comme l'ensemble définissant ce que l'on appelle un **espace de valeurs réalisables** (19). Un exemple où sont considérées  $n = 2$  variables de décisions, illustre ce type de contraintes à la Figure II.3.

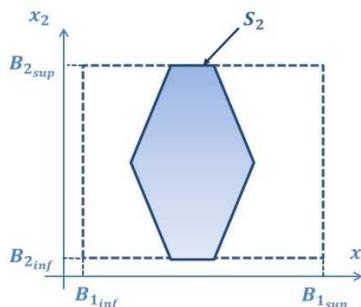


Figure II.3. Espace de valeurs Réalisables (19)

### II.2.3. Différents critères d'optimalité pour différents types d'optimum

D'*optimal* à *optimisé* il existe une différence se référant au degré d'optimalité d'une valeur de  $f$  pour un point choisi dans un espace de recherche global et l'étendue de cette optimalité par rapport aux différentes valeurs de  $\vec{x}$  telles que définies dans la précédente section. Nous distinguons alors, selon le cas, trois types différents d'optimums (minimums ou maximums). Considérons pour cela le cas d'une fonction  $f$  à maximiser, un point de l'espace de recherche  $\vec{x}^*$  est :

- Un **maximum global** de la fonction  $f$  s'il constitue la meilleure solution de façon absolue. Son optimalité s'étend de ce fait globalement à tout l'espace de recherche considéré, de manière à englober toutes les valeurs qui y sont incluses.  $\vec{x}^*$  est ainsi considérée comme la plus optimale de celles-ci maximisant  $f(\vec{x}) : f(\vec{x}^*) > f(\vec{x}) \forall \vec{x}$  tel que  $\vec{x}^* \neq \vec{x}$ . Le point  $M_1$  de la Figure II.4 est relatif à un **maximum global** dans cette représentation.
- Un **maximum local fort** de la fonction  $f$  si son optimalité s'étend uniquement à son entourage appelé plus formellement son voisinage  $V(\vec{x}^*) : f(\vec{x}^*) > f(\vec{x}) \forall \vec{x} \in V(\vec{x}^*)$  tel que  $\vec{x}^* \neq \vec{x}$ . Les points  $M_2$  et  $M_3$  sur la Figure II.4 sont des exemples de maxima locaux forts de la fonction  $f(\vec{x})$  considérée.
- Un **maximum local faible** de la fonction  $f$  si son optimalité s'étend à un ensemble restreint de l'espace de recherche. Il reprend un peu la définition d'un maximum local fort avec cependant l'existence de points de l'espace ayant la même valeur de  $f$  que lui ( $f(\vec{x}^*)$ ):  $f(\vec{x}^*) \geq f(\vec{x}) \forall \vec{x} \in V(\vec{x}^*)$  tel que  $\vec{x}^* \neq \vec{x}$  et  $V(\vec{x}^*)$  est le voisinage de  $\vec{x}^*$ . Le point  $M_4$  de la Figure II.4 vient concrétiser cette définition à travers l'exemple de  $f(\vec{x})$  qui y est donné et où est illustrée la courbe traçant les différentes valeurs de  $f(\vec{x})$  pour l'ensemble des points de l'espace de recherche.

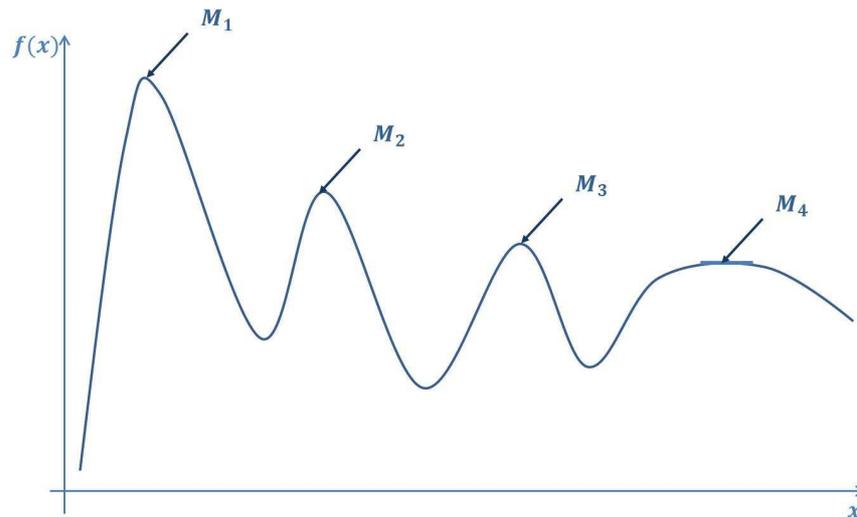


Figure II.4. Maximum global, local fort ou local faible ?

## II.2.4. Classification des problèmes d'optimisation

Nous avons tiré les différentes notions principales relatives aux définitions données dans les précédentes sections des travaux de H. Zgaya (19) où elle propose une classification des problèmes d'optimisation telle que donnée dans le Tableau II.1 suivant :

Tableau II.1. Classification des problèmes d'optimisation (19)

Caractéristiques du problème		Type du problème	
<b>Variables de Décision</b>	<b>Nombre</b>	$1$	Monovariable
		$>1$	Multivariable
	<b>Type</b>	Nombre réel continu	Continu
		Nombre entier	Entier ou discret
	Permutation sur un ensemble fini de nombres	Combinatoire	
<b>Fonction Objectif</b>	<b>Type</b>	Fonction linéaire des variables de décision	Linéaire
		Fonction quadratique des variables de décision	Quadratique
		Fonction non linéaire des variables de décision	Non linéaire
<b>Formulation du problème</b>	<b>Type</b>	Avec des contraintes	Contraint
		Sans contraintes	Non contraint

Comme le montre le Tableau II.1, il existe différents types de problèmes d'optimisation qui dépendent essentiellement des nombres et types de(s) variable(s) de décision, du type de la fonction objectif (linéaire, quadratique ou non linéaire) et de la formulation du problème (avec ou sans contrainte(s)).

Outre ces caractéristiques, le nombre de critères d'optimisation entre en considération et subdivise les problèmes placés dans ce contexte en deux types essentiellement. Nous focalisons ainsi notre étude sur deux problèmes d'optimisation, à savoir l'optimisation mono-objectif et l'optimisation multi-objectif. Nous en présentons ici les principales notions qui leurs sont relatives et définissons en plus quelques méthodes d'optimisation capables de résoudre l'un ou l'autre de ces problèmes.

### II.2.5. Optimisation mono-objectif

Implique le fait de considérer un seul critère à optimiser. Il s'agit ici d'une catégorie de problèmes d'optimisation relativement « *faciles* » à résoudre. Le terme *facile* est ici utilisé non pas pour désigner le degré de difficulté du problème mais par relativité aux problèmes multiobjectifs puisque ne présentant pas de conflit d'intérêt à privilégier l'un ou l'autre des critères d'optimisation comme c'est le cas quand une multitude de critères doivent être considérés.

Cependant, plusieurs complications peuvent exister telles que par exemple une fonction objectif non linéaire ou que l'on ne peut exprimer analytiquement en fonction des paramètres. La principale difficulté que l'on peut rencontrer en optimisation monobjectif réside ainsi dans le fait que modéliser le problème sous forme d'une équation unique peut s'avérer une tâche très difficile. Par la suite, tenter de ramener la formulation du problème à une seule fonction objectif peut biaiser la modélisation. Problème qui n'est pas rencontré dans le cas d'une optimisation multicritère où un certain degré de liberté est autorisé engendrant une certaine flexibilité qui manquait aux problèmes monobjectif (56).

### II.2.6. Optimisation multi-objectif

L'optimisation multi-objectif s'incruste dans la considération d'une multitude de critères à prendre en compte en même temps sachant qu'ils peuvent à fortiori être contradictoires et engendrer des conflits d'intérêts. Cette extension est d'autant plus nécessaire que de plus en plus de problèmes nécessitent la considération de manière simultanée d'une multitude d'objectifs. L'optimisation multicritère consiste donc à choisir parmi un ensemble infini

d'alternatives une seule (ou plusieurs) lorsqu'une multitude de critères doivent être considérés (57). L'ensemble de ces alternatives varie généralement dans un domaine continu. Depuis plus de quatre décennies déjà, la problématique multicritère n'a pas arrêté d'évoluer témoignant de la naissance et du développement d'une grande panoplie de méthodes. Deux groupes distincts ont émergé pour réaliser la classification de celles-ci; le premier est constitué des méthodes qui s'appuient sur un critère unique de synthèse, lesquelles sont dites *classiques* ; alors que le deuxième inclue des méthodes réactives au sens où elles intègrent un processus interactif de décision (58).

Les méthodes *classiques* sont des méthodes de programmation linéaire qui sont efficaces lorsque la fonction objectif et les contraintes s'expriment linéairement en fonction des variables de décision.

Formellement, la définition d'un problème d'optimisation multiobjectif reprend celle d'un problème d'optimisation classique tout en considérant un ensemble de fonctions objectifs regroupées dans  $\vec{f}(\vec{x})$  et où  $\vec{g}(\vec{x}) \leq 0$  et  $\vec{h}(\vec{x}) = 0$  :

$$\begin{cases} \vec{x} \in \mathfrak{R}^n : n \text{ variables de décision} \\ \vec{f}(\vec{x}) \in \mathfrak{R}^k : k \text{ fonctions objectif} \\ \vec{g}(\vec{x}) \in \mathfrak{R}^m : m \text{ contraintes d'inégalité} \\ \vec{h}(\vec{x}) \in \mathfrak{R}^p : p \text{ contraintes d'égalité} \end{cases}$$

Comme énoncé précédemment, un problème d'optimisation multicritère présente l'avantage de tolérer les degrés de liberté qui faisaient défaut à l'optimisation monoobjectif. Par ailleurs, cette flexibilité n'est pas sans conséquence sur l'espace de solutions qui sont passées d'une seule à plusieurs et qui dépendent fortement de la démarche d'optimisation suivie. En effet, les objectifs étant souvent contradictoires, optimiser un objectif ne peut pas être sans influencer négativement sur un ou plusieurs autres. De ce fait, une solution optimale n'existe pas dans l'absolu, nous parlons plutôt, dans ce contexte, de solutions « *optimisées* ». Ces dernières doivent ce qualificatif au fait qu'elles ne peuvent, dans la majorité des cas, optimiser tous les critères en même temps et devraient, par conséquent, en privilégier un ou plusieurs par rapport à d'autres. C'est dans ce cadre que le concept de compromis a été introduit, dans le sens où certains critères s'en verront privilégiés (optimisés) au détriment d'autres dont la qualité de performance s'en serait réduite dans l'ensemble des solutions finalement extraites. Par ailleurs, ces solutions ne sont jugées utiles et adéquates qu'au sens de l'utilisateur sur lequel se basera le décideur pour la modélisation de la fonction objectif. En

effet, ces solutions feront l'objet d'une étape de sélection constituant l'arbitrage final de l'utilisateur.

L'ensemble des solutions ainsi extraites sont dites *solutions de Pareto* et constituent la *surface de compromis*. Une solution est jugée intéressante lorsqu'elle présente une relation de dominance par rapport aux autres solutions, on dit que  $\vec{x}_1$  domine  $\vec{x}_2$  si :

$$\begin{cases} \vec{x}_1 \text{ est au moins aussi bon que } \vec{x}_2 \text{ dans tous les objectifs} \\ \vec{x}_1 \text{ est strictement meilleur que } \vec{x}_2 \text{ dans au moins un objectif} \end{cases}$$

Grâce à cette définition, la relation de dominance est ainsi considérée comme un filtre des mauvais éléments pour faire émerger seules les solutions ne pouvant être comparées entre elles. Plusieurs critères distinguent les solutions ainsi retenues leur faisant valoir le qualificatif de *solutions optimales au sens de Pareto* (ou encore *solutions non dominées*) si elles ne se dominant pas entre elles mais dominant les autres ; de *solutions optimales localement au sens de Pareto* si elles sont optimales au sens de Pareto sur une restriction de l'ensemble  $\mathfrak{R}^n$ . L'ensemble des solutions *Pareto optimales de rang 1* constituent la *Surface de Compromis* dont est extraite une solution unique dite *Optimum Global au sens de Pareto*. L'extraction de ce *point idéal* se fait par le biais d'une méthode bien déterminée telle que le *Cône Négatif* ou le *Théorème du Contact*.

La Figure II.5 reprend brièvement le principe de l'optimisation multi-objectif et de la définition de l'optimalité des solutions au sens de Pareto de par la relation de dominance.

Les solutions ainsi extraites sont obtenues de par le suivi d'une démarche précise consacrant une méthode d'optimisation spécifique parmi un large panel de techniques existant dans la littérature (59). Le choix de la méthode à appliquer doit être adéquat selon le problème d'optimisation auquel l'on se trouve confronté. Ce choix représente une difficulté qui n'est pas des moindres puisque c'est sur ce dernier que portera la responsabilité de la réussite ou non de la méthode adoptée. La sélection de la méthode d'optimisation peut se faire parmi trois grandes familles (60) distinctes de par l'instant où la décision d'effectuer ou non un compromis entre fonctions objectifs est prise. Nous retrouvons ainsi :

- Les méthodes d'optimisation a priori : le compromis est fixé avant l'application de la méthode d'optimisation.
- Les méthodes d'optimisation progressive : le compromis est fixé en interagissant avec le décideur au fur et à mesure de l'exécution de la méthode d'optimisation. Ceci afin de

pouvoir orienter la recherche vers les zones auxquelles le décideur est le plus sensible et qui satisfont le compromis qu'il souhaite opérer entre les fonctions objectifs.

- Les méthodes d'optimisation a posteriori : où après la recherche d'une sélection d'une large panoplie de solutions, différentes et bien réparties, qui se fera pour être exposée au décideur ; ce dernier pourra ainsi choisir la solution qui lui convient le plus par rapport au reste.

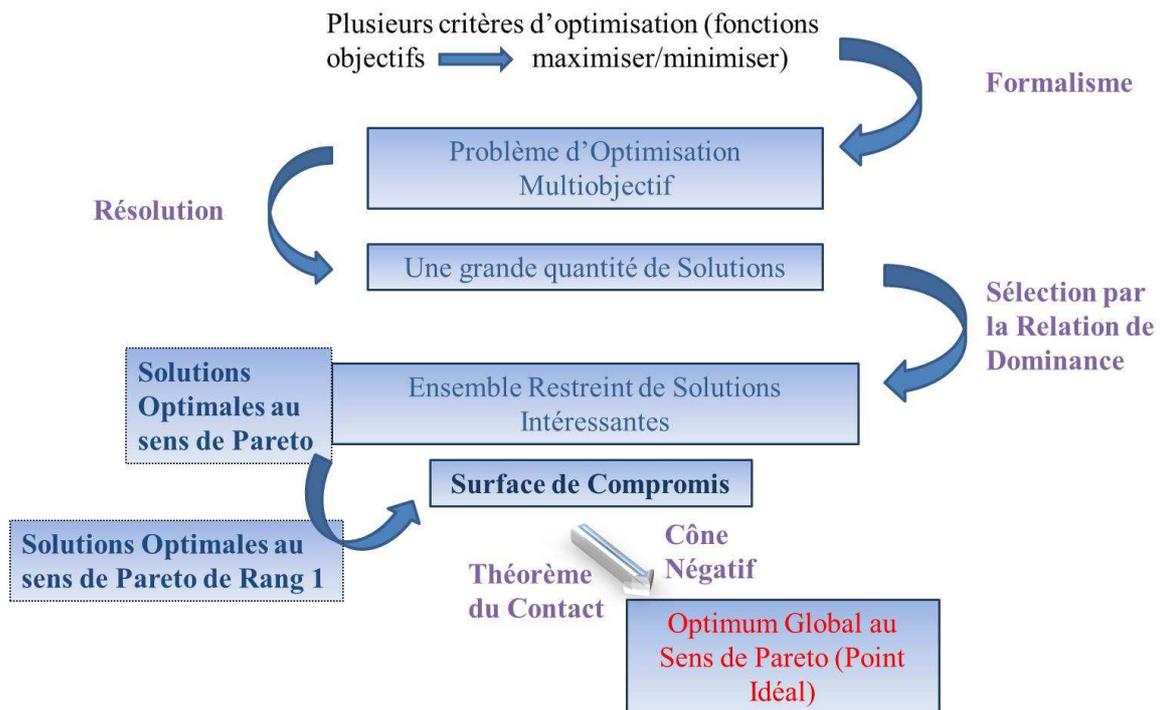


Figure II.5. Optimisation multi-objectif et optimalité au sens de Pareto

Toutes les notions ici abordées ont été largement détaillées dans l'ouvrage (56). Nous nous sommes particulièrement intéressés à ce type de problèmes d'optimisation puisque nous proposons de considérer au niveau de notre approche l'incorporation du concept d'optimisation multicritère. Ce volet sera décrit bien plus en détail dans le chapitre III.

Dans la suite, nous nous focalisons sur les méthodes d'optimisation pouvant être utilisées dans ce contexte.

### II.2.7. Les méthodes d'optimisation

Selon le cas, qu'il s'agisse d'un problème monovarié ou multivarié, continu ou discret, etc. une méthode d'optimisation adéquate est choisie avec soin afin de bien cadrer avec le contexte et résoudre le problème de manière efficace. Entre méthodes exactes, métaheuristiques, hybrides ou autres, chercheurs et praticiens ont à leur disposition un large



donnait lieu à la définition d'une fonction objectif ne possédant pas de propriété structurelle particulière, telle que la convexité par exemple. Parallèlement à l'optimisation continue, les efforts déployés dans le domaine de l'optimisation combinatoire ont abouti au développement d'un grand nombre d'*heuristiques*. Celles-ci ont pour mission principale de produire des solutions proches de l'optimum. Cependant, la plupart d'entre elles ont été conçues pour répondre de manière spécifique à un problème bien déterminé.

Compte tenu de cette spécificité handicapante des heuristiques, chercheurs et praticiens ont été amenés à considérer le développement de techniques génériques capables de s'adapter à n'importe quel domaine. Dans le but de réaliser cette genericité des méthodes, les efforts déployés dans ce sens ont mené à la conception d'une nouvelle classe de méthodes d'optimisation, nommées « *métaheuristiques* ». Ces dernières marquent une réconciliation entre les deux domaines d'optimisation continue et combinatoire puisqu'elles peuvent s'appliquer à toute sorte de problème à variables discrètes aussi bien que continues.

Plus généralement, pour la résolution des problèmes d'optimisation multi-objectif, *Colette et al.* proposent dans leur ouvrage (56) de regrouper l'ensemble des méthodes d'optimisation en cinq grandes familles brièvement décrites dans ce qui suit.

### *II.2.7.1. Les méthodes scalaires*

Elles regroupent un ensemble de méthodes dont l'approche pour la résolution d'un problème d'optimisation multiobjectif est la plus évidente. Cette approche appelée « *approche naïve* » a pour but de reformuler le problème d'optimisation de façon à revenir à une optimisation monobjectif. La manière la plus intuitive de par laquelle ce procédé est réalisé est l'*agrégation* en utilisant la sommation de l'ensemble des fonctions objectifs en une seule. Le coefficient de pondération affecté à chacune d'entre elles reflètera son poids dans la fonction objectif globale et donc l'importance du critère considéré relativement à celle-ci. Autrement, une autre méthode dite de « *Keeney-Riffa* » consiste à utiliser le même procédé d'amplification des fonctions objectifs élémentaires, mais en faisant leur produit, par la suite, pour pouvoir se ramener à un problème d'optimisation monobjectif.

### *II.2.7.2. Les méthodes interactives*

Les méthodes regroupées sous cette classe forment la famille des méthodes progressives et permettent de chercher une solution unique. Elles se basent pour cela sur une interactivité avec l'utilisateur qui a l'occasion de choisir au fur et à mesure de l'exécution de la méthode la

solution qui correspond le mieux à ses préférences par rapport au compromis qu'il souhaiterait voir effectué sur les différentes fonctions objectifs. Nous pouvons citer comme exemple correspondant à cette démarche de procédé, la méthode du *compromis par substitution*, la méthode de *Fandel*, la méthode *STEP*, etc. toutes ces méthodes sont recensées, détaillées et formalisées dans (56).

### II.2.7.3. Les méthodes floues

Ces méthodes sont inspirées de la logique floue où la logique binaire (vrai ou faux) est délaissée en faveur de l'introduction d'un certain degré de *flexibilité* ou *élasticité* en vue de pouvoir tolérer les incertitude et imprécision des connaissances humaines. La méthode de *Sakawa* ou encore la méthode de *Reardon* sont des exemples à citer dans cette catégorie.

### II.2.7.4. Les méthodes basées sur une métaheuristique

Ces méthodes, apparues depuis les années 1980, comprennent la méthode du *recuit simulé*, les *algorithmes génétiques*, la méthode de *recherche tabou*, les algorithmes de *colonies de fourmis*, etc. Elles ont pour objectif principal et commun entre elles, de résoudre au mieux les problèmes d'optimisation difficile.

En effet, les métaheuristicques ont initialement été proposées dans la recherche dédiée aux problèmes *d'optimisation difficile*. Le principal avantage de ces méthodes, constituant aussi leur principale source d'efficacité, se trouve être leur capacité à éviter les pièges des minima locaux, et ce contrairement aux méthodes d'optimisation classiques. Par conséquent, à l'opposé du procédé suivi par ces dernières qui, à partir d'une configuration initiale donnée, cherchent la meilleure valeur grâce à une suite d'itérations (au cours desquelles des mouvements, modifications élémentaires, sont effectuées), l'idée motrice des métaheuristicques repose sur le fait de tolérer une dégradation momentanée de la situation. Cette idée s'est révélée être très fructueuse et efficace pour surmonter l'obstacle érigé par le piégeage des minima (ou maxima) locaux dans la résolution des problèmes complexes, au point qu'elle est à la base de toutes les métaheuristicques dites de voisinage (recuit simulé, méthode tabou). Cette idée part du principe que l'on peut atteindre l'optimum global ou une solution qui s'en rapproche le plus à partir d'une configuration moins bonne qu'une solution extraite auparavant. En effet, il s'agit, dans ce cas, d'autoriser de temps en temps des mouvements de remontée (ou de descente dans le cas d'une maximisation) ; autrement dit, d'accepter une dégradation temporaire de la situation, lors du changement de la configuration

courante. Toutefois, afin de mener à bien le procédé de résolution du problème et éviter la divergence vers une solution moins satisfaisante, un mécanisme de contrôle des dégradations, spécifique à chaque métaheuristique est nécessaire. Il devient dès lors possible de s'extraire du piège du minimum (ou maximum) local. Ceci dans le but de permettre l'exploration d'autres vallées éventuellement plus prometteuses. A l'instar des méthodes de voisinage, les métaheuristicues dites distribuées, telles que les Algorithmes Evolutionnaires (AE), disposent elles aussi de techniques bien particulières. Ces dernières leur permettent de s'extraire d'une solution particulière, hors d'un puits local de la fonction objectif définie. Ces techniques qui affectent une solution, comme la mutation dans les AG par exemple, contribuent, dans ce cas, à seconder la technique plus collective de lutte contre les minimums (ou maximums) locaux. Dans cette dernière, est considéré en particulier, le contrôle en parallèle de l'ensemble de toutes les solutions.

Dans ce qui suit, nous passons en revue les principales métaheuristicues citées ci-dessus en essayant d'introduire les détails et principes de chacune d'entre elles. Celles-ci constituent les méthodes les plus souvent citées dans la littérature.

### A. *Le recuit simulé (Simulated Annealing)*

La méthode du *recuit simulé* est inspirée du procédé du *recuit* et a pour principe de partir d'une configuration donnée (e.g. placement initial de tous les composants d'un circuit électronique, configurations de basse énergie de matériaux magnétiques désordonnés) pour aboutir au résultat escompté. Ce résultat concerne l'optimum global, ou un résultat qui s'en rapproche, atteint via l'application de transformations élémentaires en un nombre fini d'itérations.

En pratique, cette méthode exploite l'algorithme de Metropolis, lequel permet de définir et représenter le comportement d'un système en « *équilibre thermodynamique* » à une température bien déterminée (T). Disposant initialement d'une configuration donnée, le procédé suivi par cette méthode repose sur le fait de faire subir au système des modifications élémentaires, telles que par exemple la translation d'un composant ou encore l'échange de deux composants. Si la modification en question arbore une amélioration qui converge vers l'objectif fixé (i.e. diminution de la fonction objectif ou énergie), elle est alors acceptée. Par ailleurs, même si cette même modification provoque au contraire une augmentation, il est envisageable qu'elle soit acceptée mais avec une certaine probabilité de  $e^{\frac{-\Delta E}{T}}$ . Cette démarche est par la suite réitérée en gardant la *température* T constante, jusqu'à ce que l'équilibre

thermodynamique soit atteint, concrètement au bout d'un nombre *suffisant* d'itérations (i.e. modifications). Une fois ce but atteint, toute la démarche, telle que décrite ici, est renouvelée pour une nouvelle valeur plus basse de la température. Une série de transformations est alors de nouveau effectuée sur cette valeur. Pareillement au critère d'arrêt du programme, la loi que suit la décroissance par paliers de la température est souvent empirique. La méthode du *recuit simulé* opère, lorsqu'elle est appliquée au problème de placement de composants, une transformation désordre-ordre.

Dans la Figure II.7, nous représentons l'organigramme qui reprend les principales étapes de l'algorithme du recuit simulé (59). Dans cette figure, la démarche ci-dessus décrite est reprise et récapitulée, présentant une schématisation des principales étapes suivies lors de l'application d'une méthode d'optimisation basée sur le *recuit simulé*. Il existe aussi plusieurs variantes de la méthode du recuit simulé telles que, par exemple, la *diffusion simulée*, le *recuit microcanonique*, la *méthode du seuil*, la méthode du *grand déluge* ou encore la méthode du *voyage de record en record*.

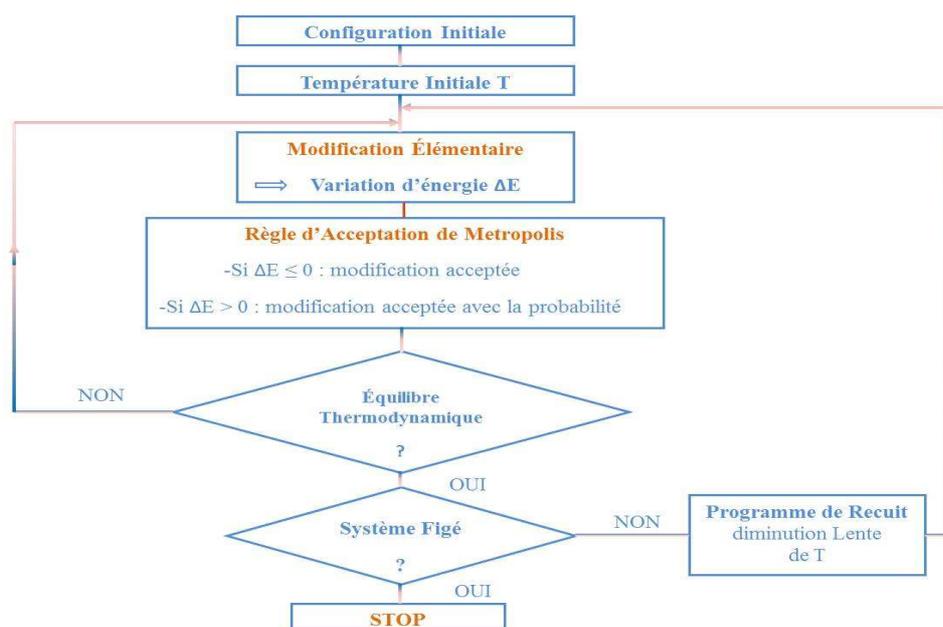


Figure II.7. Organigramme de la métaheuristique du « Recuit Simulé » (59)

### B. La recherche tabou (Tabu Search)

La méthode de recherche avec tabous, formalisée en 1986 par F. Glover dans (61), est encore appelée recherche tabou ou simplement méthode tabou (Tabu Search, TS). L'intérêt principal de cette méthode tient de la mise à profit de techniques et mécanismes inspirés de la mémoire humaine. En effet, contrairement à la méthode du recuit simulé, précédemment définie et qui est totalement dépourvue de mémoire, la recherche tabou garde en mémoire les

traces des différentes étapes de son application. Elle est, par la suite, capable de réagir en conséquence ; c'est-à-dire que, tirant les leçons du passé, les actions entreprises dans l'itération courante sont affectées de celles précédentes qui constituent l'historique de son exécution. Par ailleurs, la modélisation de la mémoire nécessite d'introduire un certain nombre de degrés de liberté qui entravent toute analyse mathématique rigoureuse de la méthode tabou (62).

Par ailleurs, semblablement à la méthode du recuit simulé, le principe de la méthode de recherche avec tabous est simple. En effet, de manière analogue à cette première méthode, le procédé suivi est le même et concerne l'application d'une série de modifications sur plusieurs itérations du processus. La méthode tabou fonctionne alors avec une seule configuration courante à la fois ; celle-ci étant obtenue de par une configuration quelconque choisie initialement et à laquelle est appliquée une mise à jour, ré-exécutée continuellement au cours du processus d'application de la méthode (i.e. au fil des différentes itérations qui se succèdent). A chaque itération, le mécanisme de passage d'une configuration donnée  $c$  à une configuration qui la succède  $c'$  suit les étapes suivantes :

- a. La construction de l'ensemble des *voisins* de  $c$ . Cet ensemble désigne toutes les configurations possibles et réalisables, et qui seraient accessibles en un seul mouvement élémentaire appliqué à  $c$ . Si cet ensemble est trop vaste, alors une technique bien déterminée est appliquée afin d'en réduire la taille. Parmi les techniques bien connues, l'on peut retrouver par exemple la sélection d'une liste de candidats ou encore l'extraction aléatoire d'un sous-ensemble de voisins de taille fixée.  $V(c)$  est l'appellation désignant l'ensemble (ou le sous-ensemble) de ces voisins,
- b. L'évaluation de la fonction objectif  $f$ , prédéfinie du problème, en chacune des configurations faisant partie de  $V(c)$ . En termes d'optimisation de la fonction objectif, la configuration  $c'$  succédant à  $c$ , dans la suite des solutions obtenues de par l'exécution de la méthode tabou, serait la meilleure parmi celles constituant l'ensemble  $V(c)$ . Nous entendons par là, la configuration pour laquelle  $f$  prend la valeur minimale, respectivement maximale, qu'il s'agisse d'un problème de minimisation ou maximisation. Par ailleurs, cette même configuration ( $c'$ ) peut être prise en compte même dans le cas où elle s'avère moins bonne que la première ( $c$ ) (i.e.  $f(c') > f(c)$ ). Nous mettons l'accent ici sur la particularité qui constitue le

principal avantage des *métaheuristiques* (notamment la méthode de recherche tabou dans ce cas), et qui consiste à éviter le piègeage des minimums locaux de la fonction objectif  $f$ .

Telle qu'elle a été présentée ici, cette procédure peut souvent s'avérer inopérante. En effet, il existe un risque fréquent et très important de retourner à une configuration précédente, déjà retenue lors d'une itération ultérieure. Ceci a pour effet d'engendrer un cycle et donc le blocage sur une boucle infinie. Afin d'éviter ce phénomène, la méthode de recherche tabou introduit le concept d'actions interdites se traduisant par la construction d'une liste de mouvements interdits. A chaque itération, cette liste, continuellement maintenue à jour, est exploitée. La méthode de recherche avec tabous tient, en effet, son appellation de cette liste qui est elle-même appelée « *liste de tabous* » ou « *liste tabou* ». Cette dernière contient un nombre fini  $m$  de mouvements interdits ( $c' \rightarrow c$ ) qui représentent les actions inverses des  $m$  derniers mouvements ( $c \rightarrow c'$ ) appliqués à  $c$  pour avoir la configuration  $c'$ . La figure II.8 présentée dans la suite montre le détail de l'organigramme relatif à l'exécution de la méthode tabou.

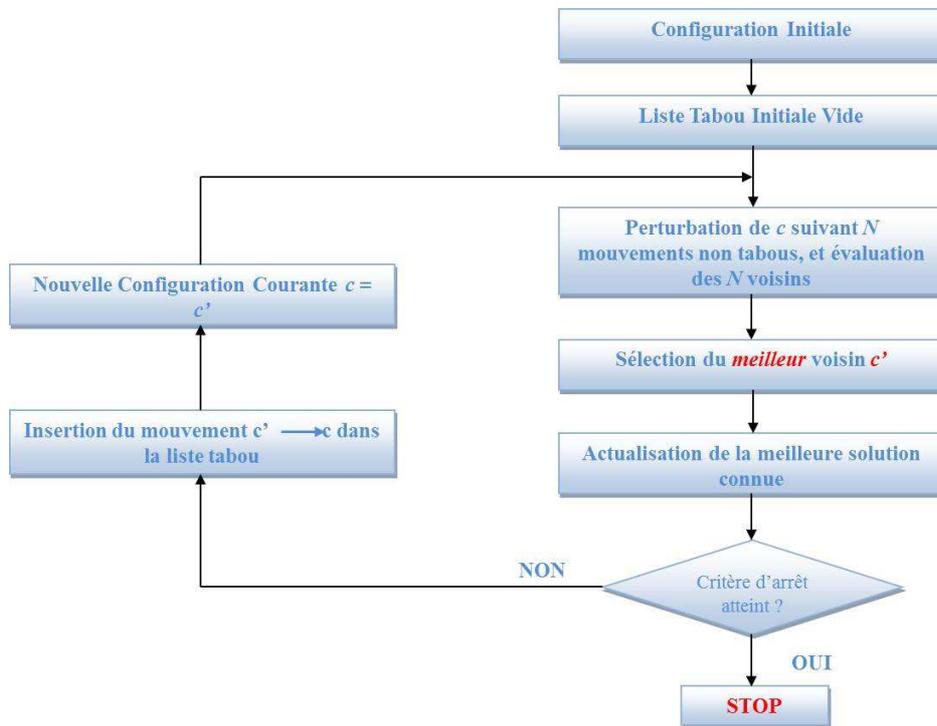


Figure II.8. Méthode de recherche tabou (59)

Cet algorithme, dit « *tabou simple* », modélise une forme rudimentaire de mémoire : la mémoire à court terme qui sauvegarde les solutions récemment visitées. Afin de remédier aux problèmes et limites susceptibles d'être engendrés par cette courte mémoire, deux

mécanismes supplémentaires, l'*intensification* et la *diversification*, sont souvent mis en œuvre afin de pouvoir doter l'algorithme d'une *mémoire à long terme*. Ils énoncent, en effet, des procédés exploitant la fréquence de l'occurrence d'évènements particuliers sur une période plus ou moins longue, plutôt que par rapport à leur proximité dans le temps. Au niveau du premier mécanisme (i.e. l'*intensification*), il s'agit de porter plus d'intérêt aux régions qui ont été identifiées comme étant particulièrement prometteuses. L'*intensification* consiste effectivement à favoriser et approfondir l'exploration de certaines régions spécifiques de l'espace des solutions jugées intéressantes. D'un autre côté, la *diversification* est au contraire la réorientation périodique de la recherche d'un optimum vers des régions trop rarement visitées jusqu'au moment considéré (itération courante).

La méthode de recherche tabou s'est avérée efficace et a fourni d'excellents résultats dans la résolution de certains problèmes d'optimisation (59). En outre, cette méthode comporte, sous sa forme de base, moins de paramètres de réglage par rapport à la méthode du *recuit simulé*. Elle est de ce fait beaucoup plus simple d'emploi que cette dernière. En revanche, si l'on rajoute les divers mécanismes annexes à cette méthode, tels que l'*intensification* et la *diversification*, nous pouvons constater une augmentation notable de la complexité.

### **C. Les algorithmes Évolutionnaires (Evolutionary Algorithms)**

Les *algorithmes évolutionnaires* (AE) sont apparus à la fin des années 1950 (63). Ils utilisent des techniques de recherche inspirées par l'évolution biologique des espèces. De premier abord, les algorithmes évolutionnaires ont suscité un intérêt minime, ceci étant principalement dû à leur important coût d'exécution. Cependant, l'intérêt porté à ces techniques a considérablement crû durant les deux dernières décennies, et ce grâce à l'augmentation vertigineuse de la performance des calculateurs, aujourd'hui très puissants. L'apparition des architectures logicielles et matérielles massivement parallèles, qui exploitent entre autre le « *parallélisme intrinsèque* » des techniques adoptées par les algorithmes évolutionnaires, a aussi largement contribué à l'augmentation de l'intérêt porté à ces techniques et donc à leur développement.

Le principe d'un algorithme évolutionnaire est simple. En effet, il s'agit de considérer un ensemble de  $N$  points, choisis à priori de manière aléatoire, dans un espace de recherche de départ. L'ensemble des  $N$  points choisis constitue la *population* initiale. Chaque individu  $x$  de la population ainsi définie possède une certaine performance, laquelle mesure son degré d'*adaptation* à l'objectif visé. Nous pouvons citer, à titre d'exemple, le cas de la minimisation d'une fonction objectif  $f$ , où l'individu  $x$  est d'autant plus performant que  $f(x)$  est plus petit.

Le principe de base d'un algorithme évolutionnaire revient à faire évoluer de manière progressive, par *générations* successives d'individus, la composition de la population tout en maintenant, toutefois, sa taille constante. Au fil des générations, l'objectif principal visé consiste à améliorer la performance globale des individus. Un procédé particulier est utilisé dans le cadre des *AE* afin de pouvoir obtenir un tel résultat, il s'agit de mimer les deux principaux mécanismes qui régissent l'évolution des êtres vivants. Ceux-ci sont définis, selon la théorie de C. Darwin, par :

- a. La *sélection* : qui favorise la reproduction et la survie des individus les plus performants,
- b. La *reproduction* : qui consiste à effectuer le brassage, la recombinaison et les variations des caractères héréditaires des parents, afin de former des descendants dotés de nouvelles potentialités.

En pratique, une représentation bien déterminée et adéquate des individus d'une population doit être choisie. Par exemple, pour les problèmes de type combinatoire, un individu peut être assimilé, de manière classique, à une liste d'entiers. Cependant, pour les problèmes numériques manipulant des variables dans des espaces continus, un individu est plutôt représenté par un vecteur de nombres réels, une chaîne de nombres binaires dans le cas des problèmes booléens, ou encore une combinaison de ces représentations dans des structures plus complexes, si besoin est. A chaque itération de l'exécution des AEs, il existe quatre phases de passage d'une génération à une autre :

- Une phase de sélection, qui consiste à désigner les individus susceptibles de participer à la reproduction. Ces derniers peuvent être choisis éventuellement à plusieurs reprises, d'autant plus souvent qu'ils sont performants. Ils seront, par conséquent sélectionnés et feront ensuite l'objet de la seconde phase de reproduction.
- Une phase de reproduction (ou de variation), qui consiste à appliquer des opérateurs de variation spécifiques sur des copies des individus sélectionnés afin d'en engendrer de nouveaux. Parmi les opérateurs les plus utilisés, nous pouvons citer le *croisement* (ou *recombinaison*) dont l'application a pour effet de produire un ou deux descendants à partir de deux parents, ainsi que la *mutation* dont le rôle principal consiste à produire un nouvel individu à partir d'un seul et unique individu. La structure des opérateurs de variation est étroitement liée à la représentation choisie pour les individus et dépend donc fortement du codage de ceux-ci.

- Une phase d'évaluation des performances qui, comme son nom l'indique, a pour objectif d'évaluer les performances des nouveaux individus générés lors de la précédente phase, sur la base des objectifs fixés.
- Une phase de remplacement qui vient finalement clore le processus de définition d'une nouvelle génération de solutions et porte sur la détermination du choix des membres de celle-ci. Par exemples, les individus les moins performants de la population peuvent être remplacés par les meilleurs individus (i.e. les plus performants) produits, en nombre égal. L'algorithme est interrompu après un certain nombre de générations, selon un critère d'arrêt à préciser.

Lorsqu'il s'agit de considérer une fonction objectif comportant plusieurs optima globaux, les algorithmes évolutionnaires sont particulièrement indiqués pour proposer un jeu de solutions diverses, du fait de la manipulation d'une population de plusieurs instances de solutions différentes dans le procédé qui leur est propre. Par conséquent, les AE peuvent fournir un échantillon de solutions de compromis, lors de la résolution de problèmes comportant plusieurs objectifs, pouvant éventuellement être contradictoires. Dans la figure II.9, est fourni le détail du processus d'exécution d'un algorithme évolutionnaire.

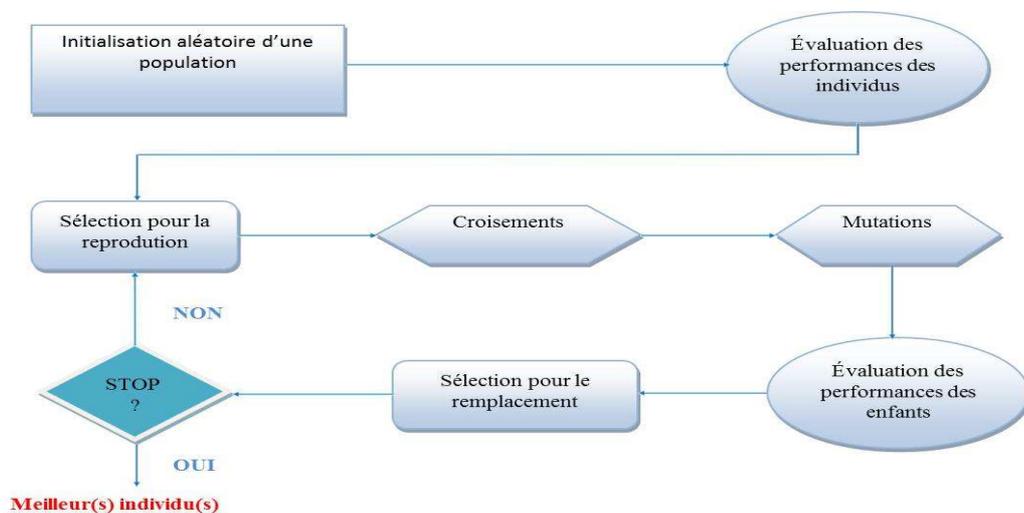


Figure II.9. Organigramme des algorithmes évolutionnaires (59)

Une variante des plus importantes sous cette catégorie est celle relative aux algorithmes génétiques. En effet, parmi la multitude d'approches connues et classées sous la catégorie des algorithmes évolutionnaires, les *Algorithmes Génétiques (AG)* sont souvent les plus cités et en constituent certainement l'exemple le plus connu. Les AG simples respectent le schéma d'un algorithme évolutionnaire mais de façon très originale. En effet, les algorithmes génétiques s'inspirent de la transcription génotype (i.e. phénotype de la génétique naturelle). Il s'agit ici

de la phase qui précède directement celle de l'évaluation des performances des individus. Un phénotype étant l'ensemble des caractères observables chez un individu, il correspond non seulement à la réalisation du génotype mais aussi à des effets du milieu, de l'environnement. Un génotype est associé à une chaîne de symboles souvent binaire, cette chaîne est décodée afin de construire une solution du problème, représentée dans son formalisme naturel (i.e. *phénotype*). Ce dernier est, par la suite, évalué, sur la base d'une fonction Fitness, pour donner une valeur de performance exploitable par les opérateurs de sélection. La figure II.10 représente le schéma d'un *algorithme génétique* simple où l'on peut remarquer qu'un opérateur de sélection proportionnelle et un remplacement générationnel sont mis en œuvre, c'est-à-dire que la population des enfants remplace complètement celle des parents. Il existe aussi une autre version *classique* qui met en place un mécanisme de *remplacement stationnaire (steady state)*.

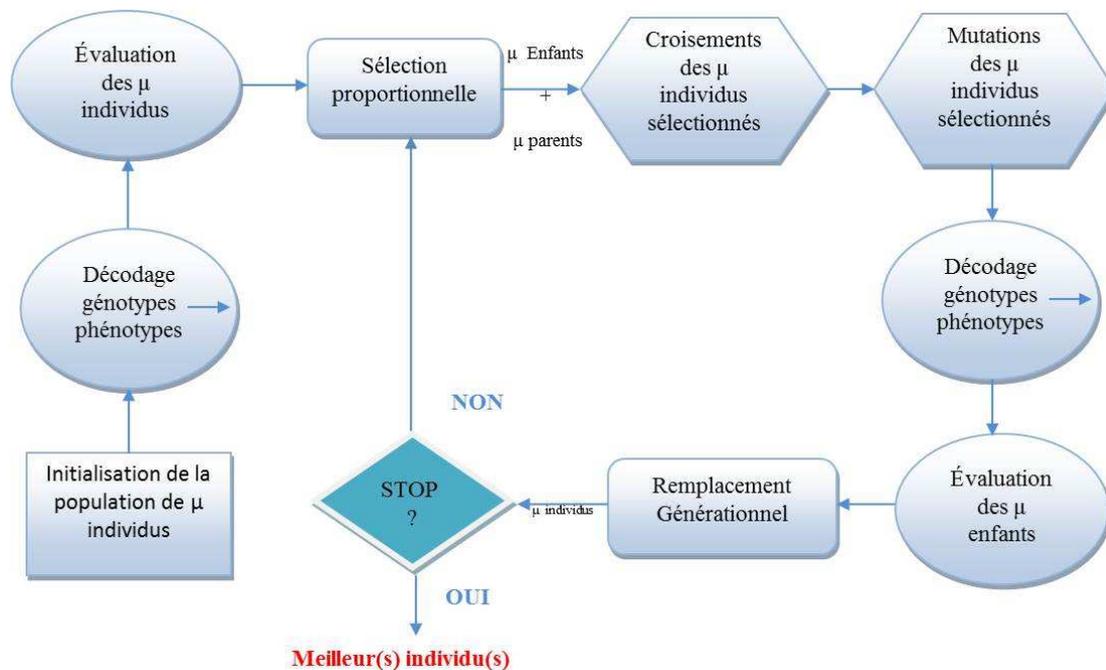


Figure II.10. Organigramme des algorithmes génétiques (59)

Les opérateurs de variation travaillent sur les génotypes. Puisque ces derniers se présentent sous la forme de chaînes binaires, l'on utilise alors les opérateurs de croisement et de mutation. Le premier opérateur est considéré comme étant l'opérateur de recherche essentiel, alors que la mutation est appliquée avec un faible taux la plupart du temps (probabilité de mutation :  $p_m$ ), de façon à maintenir un certain degré de diversité dans l'espace de recherche. La représentation étant fondée sur des chaînes binaires, la difficulté réside essentiellement dans le fait de dénicher et implémenter le bon codage, qui soit le plus en adéquation avec le

génotype, de façon à permettre aux opérateurs de variation dans l'espace des chaînes binaires de produire des descendants viables tout en respectant les contraintes du problème. Il s'agit d'une tâche non triviale dans le cas général.

Une évolution des algorithmes génétiques a considéré la concrétisation d'une multitude de suggestions de modifications qui leurs seraient apportées en vue d'améliorer leurs performances et étendre leurs domaines d'utilisation. Les chaînes binaires ont ainsi été remplacées par des représentations qui se rapprochent encore plus du formalisme des problèmes traités, évitant à la fois, l'épineuse question de la conception d'un codage efficace. Par ailleurs, la sélection proportionnelle a souvent cédé la place à d'autres formes de sélection qui peuvent s'avérer plus efficaces. Ces modifications se sont révélées d'une importance de premier ordre, suffisamment pour que les spécificités des algorithmes génétiques entravant leur expansion disparaissent par rapport à la diversité des autres approches évolutionnaires.

L'apparition des algorithmes évolutionnistes a fait l'effet d'une bombe dans l'optimisation de fonctions avec contraintes, et dans le domaine de la résolution de problèmes complexes de façon générale. L'optimisation par essaim de particules se présente comme une alternative aux algorithmes génétiques et aux colonies de fourmis pour l'optimisation de fonctions non-linéaires.

### ***D. Les algorithmes de colonies de fourmis (Ant colony)***

Énoncée par Colomi, Dorigo et Maniezzo dans (64), la méthode de colonies de fourmis se base sur le comportement des colonies de fourmis et s'efforce donc de simuler la capacité collective observée chez ces dernières pour la résolution de certains problèmes. Notons que les différents membres des colonies de fourmis sont par ailleurs dotés de facultés très limitées. Plusieurs recherches se sont orientées vers l'étude du mode de survie et des habitudes comportementales des fourmis qui sont comptés comme étant l'une des espèces les plus prospères. Le processus suivi par les fourmis, met l'accent notamment sur la faculté collective, c'est-à-dire l'aptitude et l'aisance d'une collectivité à retrouver rapidement le plus court chemin, si ce dernier se trouve fortuitement obstrué par un obstacle.

Les algorithmes de colonies de fourmis sont dotés de plusieurs caractéristiques intéressantes dont nous pouvons citer essentiellement :

- Le parallélisme intrinsèque élevé,
- La flexibilité : une colonie de fourmis est dotée d'un fort degré de flexibilité permettant son adaptation à des modifications de l'environnement,

- La robustesse : une colonie est apte à maintenir son activité si quelques-uns des individus la formant sont défaillants,
- La décentralisation : cette caractéristique met encore une fois l'accent sur l'aspect réparti (i.e. distribué) de la démarche puisqu'une colonie n'obéit pas à une autorité principale et centralisée,
- L'auto-organisation : une colonie est dotée d'un certain degré d'autonomie, elle est ainsi capable de trouver d'elle-même une solution qui n'est pas connue à l'avance.

Du fait de ses multiples avantages, ci-haut cités, et plus particulièrement le premier et le dernier, cette démarche s'avère très efficace et particulièrement indiquée pour les problèmes qui sont par nature distribués, et surtout susceptibles d'évolution dynamique, ou qui requièrent une forte tolérance aux pannes. A ce stade de développement de ces algorithmes récents, la transposition à chaque problème d'optimisation ne va cependant pas de soi, elle nécessiterait plutôt une étude préalable et devrait faire l'objet d'un traitement spécifique, qui peut s'avérer plus ou moins ardu.

Les *métaheuristiques* ont connu un immense succès grâce aux avantages qu'elles présentent et qui ont fait défaut aux méthodes classiques d'optimisation qui se sont avérées inefficaces et insatisfaisantes devant les difficultés rencontrées dans la résolution de problèmes complexes d'ingénierie. Après le triomphe qu'ont « vécu » les différentes métaheuristiques, les unes indépendamment des autres ; d'autres types de difficultés ont fait surface, mettant ainsi l'accent sur la notion de complémentarité de ces nouvelles méthodes entre elles, aussi bien qu'avec d'autres approches, détournant ainsi l'attention de certains chercheurs vers une nouvelle notion qu'est celle des *méthodes hybrides* (56).

Tous les efforts déployés dans ce sens ont pour but essentiel de fournir une qualité de service satisfaisante aidant entre autre dans le processus de prise de décision.

### *II.2.7.5. Les méthodes d'aide à la décision*

Il s'agit de méthodes d'optimisation qui permettent de répondre à plusieurs problématiques mais dont une particularité essentielle est celle de ne pouvoir travailler sur des ensembles de valeurs continues. Contrairement à la relation de dominance précédemment définie, ces méthodes permettent d'extraire un ensemble de solutions liées par une relation d'ordre : *partiel* s'il s'agit d'un ensemble de solutions ou *total* si une et une seule solution est extraite.

De la recherche opérationnelle, au traitement d'images, à l'électronique et à la conception de systèmes mécaniques, etc. l'optimisation a laissé son empreinte dans plusieurs domaines (21) (23) (19). En effet, les méthodes d'optimisation précédemment présentées sont des méthodes techniques génériques qui peuvent s'appliquer à n'importe quel domaine.

### II.2.8. l'Optimisation : une multitude de domaines concernés

L'optimisation est assez large pour inclure différentes problématiques touchant à plusieurs domaines. Des problèmes d'ordonnancement, d'affectation, de logistiques ou autres, en Informatique, automatique, robotique, etc. il en existe un large panel où le concept d'optimisation a été introduit à plusieurs niveaux impliqués dans la résolution de tels problèmes. En recherche opérationnelle par exemple, nous pouvons citer le problème bien connu du voyageur de commerce (TSP : Travel Salesman Problem) qui doit visiter  $n$  villes sans passages multiples en effectuant le plus court chemin pour revenir finalement à son point de départ. Le TSP est donc un problème discret qui a nécessité l'introduction du concept d'optimisation vue sa complexité importante. En effet, il s'agit d'un problème pour lequel il n'est pas possible de trouver une solution exacte en un temps polynomial surtout si le nombre de villes est très grand. Il a par la suite attiré l'attention de beaucoup de chercheurs et constitue l'un des problèmes les plus étudiés.

Parmi les travaux que l'on peut citer dans ce contexte, ceux de (65) qui traitent le problème du TSP en proposant une approche basée sur les algorithmes évolutionnaires où les auteurs proposent de nouvelles stratégies impliquant l'opérateur de sélection, l'opérateur de mutation et une nouvelle stratégie de contrôle. Ils se basent pour cela sur l'opérateur d'inversion dans les algorithmes génétiques pour résoudre le problème de TSP dynamique. Par ailleurs, d'autres travaux utilisant les méta-heuristiques basées sur les colonies de fourmis (66), les méthodes de recherche tabou (67), de recuit simulé (68), les algorithmes évolutionnaires ou des variantes de celles-ci existent et ont fait l'objet d'études développées dans le but de minimiser la complexité du problème initial.

#### *II.2.7.1. Optimisation dans les problèmes de transport*

Parmi les problèmes de transport qui ont fait l'objet de travaux approfondis considérant les aspects d'optimisation, nous pouvons citer ceux relatifs à la recherche d'itinéraires que ce soit dans un contexte monomodal, multimodal ou comodal, mono-opérateur ou multi-opérateur ;

les problèmes de tournées de véhicules, de ramassage et dépose de marchandises, d'objets ou de personnes, etc.

Parmi les travaux que nous pouvons citer, il y a ceux relatifs à la recherche d'itinéraires dans un contexte de transport multimodal-multiopérateur de M.A. Kammoun (18) et M.F. Feki (2), de M.M. Hizem (69) et K. Zidi (21) dont les intérêts ont balancé entre :

- L'application d'algorithmes spécifiques tels qu'une adaptation de l'algorithme de dijkstra pour les deux premiers (18) (2),
- La proposition d'un modèle dynamique des réseaux routiers intégrant l'incertitude sur son état et sur lequel des algorithmes de plus court chemin ont été appliqués (69) considérant deux visions d'optimisation monocritère et multicritères pour le troisième,
- L'intégration des méthodes d'optimisation basées sur les colonies de fourmie pour le quatrième (20). Cette intégration a été adoptée pour la mise en place d'un procédé de recherche d'itinéraires dans un contexte interactif avec l'utilisateur afin de l'aider dans ses déplacements.

Ces travaux intègrent aussi le contexte de perturbations ou de dégradation du service qu'elles soient prévues (i.e. travaux, maintenance, etc.) ou non (i.e. incident technique ou autre). Quelques uns des auteurs des travaux ci-dessus cités, proposent d'y remédier en minimisant par exemple les temps d'attente des passagers (21) ou en recalculant l'itinéraire optimal en intégrant des solutions d'échanges (autres alternatives) considérant les systèmes d'Ecopartage par exemple, comme c'est le cas dans les travaux de (2). Dans ce contexte aussi, B. Fayeche propose dans ses travaux d'utiliser les algorithmes génétiques pour la régulation du trafic (70). Aussi les travaux de (19) viennent rejoindre ces derniers dans le recours aux algorithmes génétiques pour l'optimisation des services de transport multimodal.

Un autre exemple est celui des taxis collectifs que nous pouvons très bien classer sous le thème de Transport A la Demande (TAD). Là aussi les algorithmes d'optimisation ont fait leur entrée pour pallier aux problèmes de routage de véhicules complexes essentiellement dans le cas de prise en charge instantanée de demandes dynamiques. Des heuristiques développées dans les travaux de Horn (71) ont ainsi été proposées pour les algorithmes de routage en faveur de l'optimisation de l'insertion des demandes individuelles. Une ré-optimisation régulière de tous les itinéraires est aussi proposée après l'insertion de chaque demande (origine et destination nouvelles).

Dans le même contexte, un des problèmes de transport les plus connus est le problème de tournée de véhicules qui a été considéré dans plus d'une étude et a fait l'objet de plusieurs travaux incluant notamment le concept d'optimisation vue sa complexité. Nous pouvons citer à titre d'exemple une étude d'optimisation réalisée par *Wang et al.* (72) en regard du développement important du commerce électronique et des problèmes logistiques, pour la distribution de marchandises, qu'il engendre. Les auteurs de cette étude proposent dans leurs travaux un algorithme en deux phases : basé sur la notion de classification hiérarchique dans la première pour la subdivision de l'ensemble des clients en plusieurs régions distinctes ; puis sur une version améliorée des algorithmes génétiques dans la deuxième pour rechercher les solutions.

Dans le même volet, les travaux de Diana et Dessouki (73) s'inscrivent dans le cadre de propositions pour l'optimisation des problèmes de transport à la demande de personnes handicapées ou personnes âgées (DARP : Dial-A-Ride Problem) en considérant les fenêtres de temps. Les auteurs proposent dans ce contexte une heuristique visant comme objectif principal la minimisation d'une fonction généralisée de regret considérant à la fois la distance parcourue par les véhicules, le temps supplémentaire de trajet subi par le client par rapport au trajet direct et le temps inoccupé des véhicules. Dans l'algorithme ainsi proposé, les auteurs considèrent en premier lieu l'insertion des demandes ayant les heures de départ les plus proches et les points d'origine les plus éloignés du dépôt. Par la suite, c'est le reste des demandes qui est inséré en second lieu suivant un processus spécifique minimisant l'augmentation du coût dans chaque itinéraire. Cette étude a été bien détaillée dans (74) avec une liste non exhaustive d'autres travaux recensés dans la littérature.

### *II.2.7.2. Optimisation dans les problèmes de covoiturage*

De manière spécifique à notre domaine d'intérêt, nous considérons ici les travaux relatés dans la littérature et ayant considéré un volet optimisation dans des problèmes de covoiturage. L'auteur d'une étude sur les taxis collectifs (75) introduit des algorithmes d'optimisation dans ce contexte en application au problème de correspondance (matching) entre les trajets requis dans un contexte de voyages quotidiens de type domicile-travail (plusieurs-à-un) et travail-domicile (un-à-plusieurs). Aussi dans ce cadre I.H. Dridi (76) a introduit de par ces travaux l'optimisation heuristique au problème du m-PDPTW (Pickup and Delivery Problem with Time Windows à plusieurs véhicules) prenant en compte le volet statique aussi bien que dynamique adoptant pour cela les algorithmes génétiques. Nous citons particulièrement ces

travaux, puisque le PDPTW, variante du problème de VRP se rapproche considérablement du problème de covoiturage. En effet, en plus de l'existence des contraintes sur le temps et la capacité des véhicules, le PDPTW implique un ensemble de clients (piétons dans notre cas) et un ensemble de fournisseurs géographiquement localisés (pouvant être assimilés aux offres de covoiturage) et une correspondance adéquate doit être effectuée entre les deux.

Par ailleurs, des logiciels de covoiturage se focalisent plutôt sur l'aspect optimisant du concept de covoiturage même. Sequovia<sup>28</sup> par exemple s'est introduit dans les marchés des entreprises pour rechercher les possibilités de covoiturage entre leurs salariés créant les meilleures combinaisons de covoitureurs et covoiturés sur la base des correspondances de leurs trajets. Pour cela, il intègre des outils GPS de traçage d'itinéraires. La correspondance est ainsi faite sur la base des itinéraires exacts des voitures (point par point). La vocation du logiciel est donc d'opérer uniquement dans les entreprises sur les types de trajets professionnels. Dans le même contexte, COVIVO<sup>29</sup> propose une offre aussi dédiée aux entreprises et qui vise à aborder de manière globale et intégrée la problématique de tous les déplacements d'une entreprise (salariés, clients, visiteurs, livreurs,...). L'objectif premier de ces logiciels est ainsi de rationaliser les déplacements professionnels pour un meilleur environnement et une diminution des coûts.

Peu de travaux ont réellement consacré l'effort d'intégration du concept d'optimisation au covoiturage, encore moins le covoiturage dynamique. La plupart d'entre eux étant en effet, comme l'a révélé l'étude au chapitre I, sous forme de sites Internet capables uniquement de collecter les données nécessaires sur les covoitureurs et covoiturés ainsi que sur leurs trajets respectifs. Par ailleurs, même ceux dotés de capacités de calculs automatisés pour la recherche des correspondances accordent peu d'égards à l'optimisation des parcours des usagers, qu'ils soient conducteurs ou passagers. Venant rejoindre les problèmes énoncés dans le chapitre I, ceci est à l'origine de l'intérêt bien particulier que nous portons au problème de covoiturage. Le système que nous proposons se base ainsi sur les axes fondamentaux du temps réel et de l'optimisation.

---

<sup>28</sup> <http://www.sequovia.com/>

<sup>29</sup> <http://www.europages.fr/COVIVO/>

## **II.3. CODAC : l'optimisation au service de la qualité dans le covoiturage dynamique**

Partant du but essentiel à atteindre de par nos travaux, à savoir la mise en place d'un service de covoiturage rapide et efficace, il est impératif de considérer un volet optimisation. Outre la satisfaction des usagers du service considérant leurs préférences de déplacement, ce volet doit impérativement tenir compte de la criticité dans un environnement aussi dynamique et instable que celui du covoiturage dynamique. Notre premier objectif s'inscrivant dans un cadre d'optimisation, chercher à satisfaire au mieux les utilisateurs du service et à leur offrir une qualité de service à la hauteur de leurs attentes est derrière cette quête d'optimalité.

### **II.3.1. CODAC : une optimisation au profit de la satisfaction de l'utilisateur**

En considérant le problème de covoiturage dynamique, une problématique essentielle a retenu notre attention. Laquelle se présente sous la forme de tolérance ou non d'un certain degré de flexibilité dans la construction des itinéraires à fournir aux usagers avec les affectations de véhicules qui y correspondent. Il s'agit en effet de faire le choix d'adopter une méthodologie de résolution qui se baserait sur une simple correspondance entre les trajets en considérant les demandes des covoiturés dans leur totalité ou bien qui permettrait une certaine tolérance aux changements de véhicules durant le voyage. C'est de là qu'a émané le besoin d'intégrer le concept d'optimisation et a requis un intérêt grandissant dans nos travaux pour s'étendre à une multitude de critères considérant à la fois les temps d'attente et de parcours dans un premier temps. La première approche proposée de par notre étude considère en effet des algorithmes d'optimisation basés sur une version améliorée de Dijkstra adaptée à notre contexte. Les algorithmes proposés ont pour effet de creuser dans l'espace des solutions potentielles à la recherche d'une réponse optimisant le temps de parcours global tout en permettant une grande flexibilité par rapport à la composition des solutions sans contrainte sur le nombre de changements. Dans un second temps, c'est ce dernier critère qui est privilégié donnant ainsi la priorité aux solutions « complètes » correspondantes au trajet global demandé si elles existent. Sinon, celles avec un nombre minimal de correspondances sont extraites suite à l'exécution d'un nombre d'itérations nécessaires des algorithmes proposés dans ce second volet. Dans le cas d'une multiplicité de solutions réalisables, que nous avons appelées faisables dans nos travaux, celle optimisant le mieux la durée globale du trajet (TD : Travel

Duration) est ainsi sélectionnée. Cette deuxième manière d'approcher le problème d'optimisation dans les systèmes de covoiturage dynamique vient du fait que la première, basée sur Dijkstra, peut discriminer des solutions plus coûteuses en temps (TD) mais moins en changements. En effet, celles générées de par la première approche peuvent s'avérer fort contraignantes pour un individu puisque pouvant engendrer une multitude de changements à petits intervalles. Qu'il soit court ou long, un trajet où l'utilisateur doit faire plusieurs changements est généralement délaissé pour un autre représentant moins de transferts, même si le premier permet de gagner en termes de durée de parcours globale ; le critère de confort étant d'une importance égale à celui-ci (TD).

Toutes ces notions ainsi que les méthodes et concepts adoptés dans ce cadre favorisant l'optimisation des solutions générées aussi bien que l'optimisation des performances du système sont décrits avec le niveau de détail requis dans les chapitres suivants (*c.f. Chapitres III et IV*). Ces derniers seront en effet alloués à la formulation mathématique de notre problème sur la base de la modélisation choisie. Aussi les deux méthodes ici brièvement décrites et considérées pour approcher de deux manières différentes le problème de Covoiturage Dynamique Optimisé (CDO) y seront détaillées. Les algorithmes mis en place dans le cadre de l'une et l'autre sont aussi fournis conformément à la formulation adoptée.

Nous visons de par ces algorithmes l'optimalité de la qualité des réponses fournies aux usagers dans le but premier d'en optimiser la satisfaction tout en offrant une qualité de service digne d'un système compétitif à l'échelle des systèmes dynamiques optimisés. Par contre, il n'est de doute que pour assurer une telle optimalité, il faut savoir narguer les obstacles qui vont à l'encontre de l'efficacité ainsi que de la rapidité d'exécution des traitements d'affectation. Un obstacle majeur se trouve en l'aspect combinatoire du problème considéré et qui présente une complexité exponentielle pouvant entraver nos projets de mise à l'échelle. La section suivante fait en effet l'objet d'une étude de la complexité du problème de Covoiturage Dynamique Optimisé (CDO) de par sa comparaison avec des problèmes largement étudiés dans la littérature en raison de leur difficulté de résolution.

### II.3.2. Complexité du problème de covoiturage dynamique optimisé

En matière d'optimisation difficile, deux types de problèmes d'optimisation sont discernés ; le premier est relatif à ceux dits *combinatoires* (*problèmes discrets*) et le deuxième à ceux dont les variables sont *continues* (56). La différenciation entre les problèmes discrets et

continus a été considérée pour cerner le domaine de l'optimisation difficile. En effet, n'ayant pas de définition précise et stricte, deux types de problèmes distincts lui ont valu cette description :

- Le premier concerne les problèmes d'optimisation combinatoire pour lesquels il n'existe pas d'algorithmes *exacts rapides*. C'est le cas par exemple du problème de voyageur de commerce classé dans ce contexte sous la catégorie des problèmes dits « *NP-Difficiles* ». La résolution exacte de ce type de problèmes ne peut donc se faire en un temps de calcul proportionnel à  $\aleph^n$ , où  $n$  est un nombre entier et  $\aleph$  désigne le nombre de paramètres inconnus du problème.
- Le deuxième type est relatif aux problèmes d'optimisation dont les variables sont de type continu. Pour ce type de problèmes il est connu qu'il n'existe pas d'algorithmes garantissant le fait de trouver un optimum *global* en un nombre fini de calculs et à coup sûr.

Dans un problème d'optimisation combinatoire, il s'agit de trouver la meilleure solution à partir d'un ensemble dit « *ensemble de solutions réalisables* ». Cet ensemble est décrit de manière implicite à travers l'ensemble des contraintes que doivent satisfaire les solutions qui y appartiennent, appelées « *solutions réalisables* ». Le nombre de celles-ci est généralement fini mais exponentiel (très grand), d'où la difficulté de résolution et ses mauvaises répercussions sur les performances du système considéré. Compte tenu de cet énoncé, un problème d'optimisation combinatoire peut avoir plusieurs solutions possibles (77).

L'incorporation du concept d'optimisation dans nos travaux de par les objectifs essentiels d'optimalité de la qualité de service à laquelle nous aspirons correspond parfaitement à cette description. Ayant adopté une modélisation sous forme de GDD qui épouse adéquatement la forme du réseau en temps réel, nous comparons ici le procédé d'optimisation en covoiturage dynamique par rapport à d'autres problèmes très connus et qui ont connu un large champ d'études. Cette comparaison, basée sur la mise en exergue de l'analogie du problème de Covoiturage Dynamique Optimisé (CDO) avec d'autres problèmes, se subdivise en deux études élémentaires données dans les Tableaux II.2. et II.3. Chacune des deux études présente une comparaison du CDO par rapport à un problème d'optimisation bien déterminé et connu pour sa grande complexité.

*Tableau II.2. Comparaison entre les problèmes du TSP et du Covoiturage Dynamique Optimisé (CDO)*

<i>Problème</i>	<i>Paramètres</i>	<i>Contraintes</i>	<i>Fonction Objectif</i>
<i>TSP</i>	<i>Nœuds : Villes de passage du Voyageur de Commerce ; Arcs : chemin entre les villes</i>	<i>Chaque ville doit être visitée une et une seule fois</i>	<i>Trouver une tournée légale et optimisée : le plus court chemin reliant toutes les villes</i>
<i>CDO</i> <i>(à un instant t donné pour une requête donnée)</i>	<i>Nœuds : Points de covoiturage éventuels (multiples prises et déposes) : origines / destinations Arcs : bouts de trajets formant l'itinéraire global du piéton</i>	<i>Itinéraire unique n'engendrant pas de boucle, disponibilité et positions et trajets des des voitures, nombre de places disponibles, etc.</i>	<i>Fournir l'itinéraire avec l'affectation des voitures de covoiturage qui optimise la durée du trajet</i>

Dans le premier tableau II.2, ci-dessus présenté, est fournie une étude comparative mettant en exergue les similitudes que présente le CDO par rapport au problème très connu du voyageur de commerce (TSP). Similaires en plus d'un point, ces deux problèmes sont ici énoncés comme étant analogues par rapport aux paramètres à prendre en compte (nœuds origines/destinations Vs villes...), par rapport aux contraintes (chaque nœud (ville) doit être visité(e) une et une seule fois...), et aussi par rapport à la fonction objectif où il s'agit de minimiser un coût (distance, durée du trajet, émission de CO<sub>2</sub>, etc.). Compte tenu de ces similitudes et de l'importante complexité du problème de voyageur de commerce, le CDO est ainsi considéré comme un problème combinatoire *NP-Difficile* dont la complexité s'exprime de manière similaire à celle du TSP ( $O(CDO) \equiv O(TSP)$ ).

Ceci étant, nous n'avons considéré dans cette étude que le problème du CDO du point de vue unique de l'utilisateur émettant une requête, négligeant de par-là la considération des voyageurs en tant que covoitureurs (conducteurs de véhicules) dont les contraintes sont largement prises en compte dans notre problème. Lequel s'avère beaucoup plus complexe que le TSP dans ce cas. Une étude bien détaillée dans ce sens a été publiée dans le cadre de nos travaux (52) (78).

Outre le problème du voyageur de commerce, nous nous sommes focalisés sur celui des tournées de véhicules (VRP : Vehicle Routing Problem) et plus particulièrement le PDP (i.e. problème de collecte et livraison d'objets : produits, marchandises, personnes, etc. (79)

énoncé dans le chapitre précédent. Nous considérons pour cela deux problèmes dérivés de ceux de tournée de véhicule et plus particulièrement les DPDPs (Dynamic Pickup and Delivery Problems pouvant être traduit littéralement en : problèmes de ramassage et livraison dynamiques). Ceux-ci représentent bien plus de similitudes par rapport au problème de covoiturage dynamique que le TSP. Le Tableau II.3. fournit ainsi une étude comparative de notre problème avec deux variantes des DPDPs, à savoir le Swapping Problem SP (8) et le Dial-A-Ride Problem (DARP) (11). Ce dernier se rapproche encore plus de notre problématique dans le sens où il s'agit de transport de personnes.

*Tableau II.3. Analogie DPDPs Versus CDO*

<i>Problème</i>	<i>Paramètres</i>	<i>Contraintes</i>	<i>Fonction Objectif</i>
<i>DPDPs</i>	<i>Nœuds possédant des « charges » qui correspondent à des objets : marchandises (SP) ou personnes(DARP)</i>  <i>Arcs : itinéraires</i>	<i>Un ou plusieurs véhicules <math>V_j</math> de capacités limitées <math>C_j</math>, requêtes émises en temps réel ...</i>	<i>Rechercher le plus court chemin pour réarranger les objets ou personnes</i>
<i>CDO</i>	<i>Nœuds correspondant aux Coordonnées des utilisateurs, Specifications des requêtes (Origines, Destinations...)</i>  <i>Arcs : itinéraires</i>	<i>Un ou plusieurs véhicules <math>V_j</math> de capacités limitées <math>C_j</math> (nombre de places disponibles offertes), requêtes et offres émises en temps réel ...</i>	<i>Rechercher un itinéraire optimisé au vue d'un ou plusieurs critères (Durée Globale du trajet, etc.) pour chaque requête usager</i>

Les deux problèmes (*DPDPs* et *CDO*) sont similaires à quelques détails près. Considérant le transport de personnes, le problème particulier du *DARP* dynamique classé sous la catégorie des *DPDPs* est analogue au *CDO* pratiquement en tout point. De plus, comme le montre cette étude (80), un autre problème intégrant cette catégorie qu'est le *Swapping Problem (SP)* rejoint le problème du *CDO* en plus d'un point de similitude. Ces problèmes (*SP* et *dynamic DARP*) sont connus pour leur importante complexité qui est d'ordre exponentiel. Par analogie à ces problèmes, le *CDO* est ainsi énoncé comme étant un problème d'optimisation combinatoire *NP-Difficile* :

$$O(CDO) \equiv O(Swapping Problem) \equiv O(Dynamic DARP)$$

En plus, cette étude n'a eu pour effet que de révéler les similitudes entre ces approches passant sous silence les particularités du problème de *CDO* et qui en font un problème plus

complexe qu'à ce qu'il paraît à travers cette comparaison. Plusieurs aspects n'ont, effectivement, pas été considérés dans cette étude. Ceux-ci concernent :

- Le problème du covoiturage dynamique optimisé du point de vue du conducteur (i.e. covoitureur). En effet, dans le concept de covoiturage, les services fonctionnent uniquement sur la base de l'offre de particuliers et doivent donc prendre en compte leurs spécificités de déplacement, eux-mêmes ayant des contraintes et préférences sur les trajets qu'ils souhaitent effectuer. La correspondance avec les déplacements demandés est donc de l'essor de l'offre et non pas au gré du système. Ce dernier se doit donc de respecter les exigences des conducteurs et pourquoi pas optimiser leurs parcours. Ainsi le problème de CDO n'est pas aussi restreint que les DPDPs pour ne considérer que les demandes de déplacement.
- A un instant donné, il peut exister plus d'une offre sur un même trajet ou qui peuvent très bien concerner des trajets différents. La diversité de l'offre rajoute ainsi à la complexité du problème pour essayer de trouver les correspondances adéquates, puis en extraire celle(s) qui optimise(nt) le mieux le ou les critères fixés.
- Etc.

Ces aspects sont susceptibles d'avoir pour effet d'augmenter encore plus la complexité du problème énonçant ainsi le CDO comme étant plus difficile que les problématiques de prise et dépose d'objets ou de personnes. Par la suite,  $O(CDO) \geq O(DPDPs) \equiv O(\text{Dynamic DARP})$ .

Cette étude révèle l'importante difficulté à résoudre la problématique du CDO et qui serait probablement à l'origine d'un grand handicap entravant les projets de mise en place d'un système d'optimisation performant.

### II.3.3. Avis et mesures

Se heurter à un problème de complexité importante peut fort probablement conduire à un échec du système si cette problématique n'est pas traitée. En effet, ayant des répercussions très significatives et accablantes, une complexité exponentielle induit la plupart du temps à une lenteur généralement importante dans les délais d'exécution du système. Par conséquent, afin de demeurer à l'échelle de la compétitivité et offrir un système riche en fonctionnalités tout en étant techniquement performant, nous nous intéressons à une nouvelle stratégie de résolution efficace et efficiente.

Dans cette optique, nous nous proposons de considérer un principe de décomposition qui serait à l'origine de l'élaboration de méthodes d'optimisation distribuée. Cette stratégie n'a pour but que de faire abaisser la complexité du problème tout en offrant des méthodes efficaces permettant de le résoudre dans les meilleurs délais. Nous nous sommes aidés pour cela de plusieurs concepts décrits dans la suite de ce chapitre. Parmi les concepts dont nous avons fait usage, les Systèmes Multi-Agents (*SMA*) intégrés comme prototype de base sur lequel est fondée l'architecture du système proposé. Nous en fournissons dans ce qui suit un large panel de notions et définitions qui ont contribué à en faire le succès.

Par ailleurs, il convient tout d'abord de spécifier que les *SMA*, aujourd'hui très répandus, sont bien connus pour leur efficacité à résoudre les problèmes de complexité importante et réussissent remarquablement là où les systèmes monolithiques échouent. Ceci vient soutenir notre thèse que le système proposé se doit de fournir une qualité de service optimale abolissant le problème de la complexité exponentielle. Cependant, l'idée d'incorporer le paradigme agent émane d'une idée motrice qui concerne la décomposition ainsi que la décentralisation du procédé d'affectation optimisée et sur laquelle repose tout le principe du système ici élaboré. En effet, cette idée constitue l'axe pivot autour et en fonction duquel ont été construits les différents choix de modélisation, de méthodologies de résolution basée sur la mise en place de procédés d'optimisation distribuée, d'intégration des systèmes multi-agent, d'algorithmes adaptés à ce contexte, etc. Nous définirons les principaux fondements de cette idée dans la sixième section de ce chapitre après avoir procédé à une présentation des principaux concepts qui y sont directement liés, à savoir les *SMA* et les *GDD*. Les premiers font l'objet d'une étude plus ou moins approfondie dans la section suivante avec un focus sur les principales notions dont nous ferons usage au niveau de notre approche.

### **II.4. Les systèmes multi-agents au service de l'optimisation**

Les systèmes multi-agents ont connu, depuis leur apparition aux états unis en 1970, un essor important au niveau mondial qui n'a pas encore été démenti. Faisant intervenir plusieurs entités hétérogènes pour la résolution d'un même et unique problème, et capables de communiquer de surcroît, les *SMA* ont changé la perception « classique » des méthodologies de résolution. Ce concept a par ailleurs l'avantage supplémentaire de coïncider avec une montée en puissance des nouvelles Sciences et Technologies de l'Information et de la

Communication (*STIC*) changeant ainsi la vision que l'on a habituellement des interactions entre différentes entités intelligentes.

Développés dans le cadre d'applications sophistiquées, les *SMA* favorisent la mise en place de processus distribués sur différentes entités autonomes (ou semi-autonomes) pour en garantir l'optimalité. En effet, faisant partie du domaine de *l'Intelligence Artificielle Distribuée (IAD)*, les *SMA* sont de plus en plus utilisés dans plusieurs disciplines de recherche : communication, santé, production, etc. et plus particulièrement les systèmes logistiques. Par ailleurs, à tous égards et l'avancement technologique aidant, les systèmes multi-agents s'avèrent très efficaces dans la résolution de problèmes complexes et se prêtent adéquatement au contexte distribué ; ce qui aide à leur intrusion dans tous les domaines.

### II.4.1. SMA : le concept

Un SMA est un système composé d'un ensemble d'entités interactives et autonomes, appelées « *agents* ». Un concept étroitement lié aux SMA est la notion d'organisation décrivant la manière suivant laquelle les différentes entités opèrent et interagissent entre elles au sein d'un environnement spécifique. Cette organisation décrit par ailleurs les agissements et communications entre ces entités régissant de ce fait les relations ainsi que la dynamique des interactions entre celles-ci.

Plus concrètement, un SMA est composé d'un environnement, d'un ensemble d'objets passifs (créés, perçus, modifiés et détruits par les agents), d'un ensemble d'agents actifs et avec, les relations qui les relient entre eux, leurs opérations et compétences ainsi qu'un ensemble de lois universelles (18).

Un SMA peut ainsi être vu sous forme d'une composition de plusieurs entités différentes considérées comme étant des agents intelligents et interactifs avec des processus parallèles (81). Ces agents agissent dans un environnement spécifique à la problématique que se doit de résoudre le système en question. Leurs agissements sont régis par une certaine loi imposée par l'organisation adoptée qui établit par ailleurs les règles de communication devant être suivies par ces entités. Ces règles sont convenues à l'avance selon l'organisation adoptée et doivent être respectées afin de pouvoir demeurer dans un cadre de travail collaboratif coordonné, cohérent et sans conflit entre les agents. L'objectif premier visé de par cette modélisation est de trouver une unité dans la diversité et non pas d'uniformiser le processus établi par chacune des entités.

Les SMA sont très utilisés dans les problématiques connues pour avoir une complexité importante et s'avèrent très efficaces dans ces cas vu l'aspect distribué dont est assuré le procédé de résolution adopté. En effet, se partageant les tâches d'un problème complexe à priori, les entités exécutent de manière plus ou moins autonomes des processus moins complexes menés en parallèle dans un contexte de « *multi-threading* ». Une modélisation multi-agent en est ainsi définie comme étant le terrain favorable à la répartition d'un problème en un ensemble de compétences et connaissances distribuées sur les agents la constituant. Cette modélisation incorpore aussi des moyens permettant la communication et l'interaction entre ces agents au service de l'établissement de liens et dialogues ainsi que de la recomposition. Ce sont les organisations de ces entités qui énoncent les règles de recomposition

### *II.4.1.1. Définitions élémentaires*

Plusieurs travaux se sont intéressés au concept des SMA et ont essayé de donner une définition précise à la notion d'*agent*. Nous tirons nos définitions ici d'études intégrant les SMA dans le domaine du transport distribué, notamment ceux évoqués dans l'état de l'art que nous avons réservé pour l'étude de l'existant dans le domaine du transport multimodal et comodal (2), (82), (83), etc.

Ainsi le terme agent a bénéficié de plusieurs identifications et spécificités qui se rapprochent incontestablement. Par ailleurs, la définition la plus récurrente est celle qui définit un *agent* comme étant une entité virtuelle (logicielle) ou physique autonome (84) :

- capable d'agir dans un environnement
- dotée de capacité de communication la reliant directement aux autres agents
- mue par un ensemble de tendances (sous forme d'objectifs individuels ou d'une fonction de satisfaction voire même de survie qu'elle veut optimiser)
- en possession de ressources qui lui sont propres
- dotée d'aptitudes de perception de parties limitées de son environnement
- possédant une représentation partielle, voire même nulle de l'environnement dans lequel elle évolue
- dotée d'un certain nombre de compétences et apte à offrir un ensemble de services
- capable éventuellement de se reproduire

- suivant un comportement qui lui est spécifique dans le but d'atteindre ses objectifs fixés et ce en prenant en considération ses perceptions et représentations et en fonction aussi des communications et messages dont elle est réceptive. Ceci est faisable notamment grâce aux compétences et ressources dont elle est en possession.

Selon cette définition (84), le paradigme agent ici considéré est spécifié en fonction de ses interactions avec l'environnement dans lequel il évolue. Pour reprendre les termes exacts employés par l'auteur : «*un agent est une entité capable de percevoir son environnement et de réagir en fonction de ces perceptions* ».

Le cycle de vie d'un agent en est ainsi perçu et déduit directement à partir de cette définition. Trois grandes étapes permettent de retracer ainsi l'activité d'un agent dès sa création :

- a. *La perception de l'environnement* : pour ce faire, l'agent use de ses connaissances et comportements
- b. *La prise de décision* : appelée encore *délibération* consiste à décider de l'action suivante à entreprendre. Les connaissances et l'intelligence de l'agent aidant, cette fonctionnalité constitue une des spécificités les plus importantes faisant la particularité du concept agent. Elle symbolise en effet l'*autonomie* d'un agent.
- c. *L'exécution* : consiste concrètement à effectuer l'action choisie lors de la précédente étape du cycle de vie de l'agent.

Les trois étapes relevées du cycle de vie de l'agent sont schématisées dans la Figure II.11.

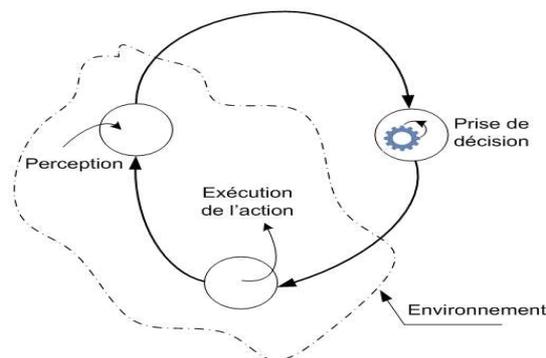


Figure II.11. Cycle de vie d'un agent (2)

D'autres chercheurs ont tenté de définir le concept agent mais finissent tous par se rejoindre sur les notions de base faisant d'un agent ce qu'il est. Essayant de synthétiser l'ensemble de ces définitions, une plus commune englobe ces notions et définit un agent

comme étant « *un système informatique doué de raisonnement et capable de communiquer avec ses semblables et qui agit sur son environnement d'une façon autonome et en fonction de sa perception pour atteindre les objectifs pour lesquels il a été conçu* » (2).

### II.4.1.2. Différents types d'agents

La différence dans la typologie des agents relève essentiellement de la *différence comportementale* par rapport à la manière et la rapidité d'exécution des actions qui leur sont spécifiques (85) (86). Deux types principaux d'agents sont distingués dans la littérature et ont fait l'objet de plusieurs travaux (87) :

- Les **agents réactifs** : très rapides, leurs actions sont assimilées à des réflexes. Ils réagissent en effet très rapidement pour répondre à des stimuli provenant du monde extérieur selon des règles pré-établies. Pour cela, ces agents sont constamment en état de veille attendant un changement probable survenant dans l'environnement. Ce type d'agents correspond à une modélisation adéquate pour la simulation de mondes et de sociétés (humaines, animales ou machines) et l'étude du comportement et propriétés individuels et leurs impacts sur le groupe.
- Les **agents cognitifs** : contrairement aux précédents, ces agents possèdent un comportement réfléchi basé sur leurs connaissances de l'environnement, des autres agents et d'eux-mêmes. Les actions qu'ils entreprennent sont généralement le fruit de programmes de planification et d'un raisonnement rationnel basé sur les connaissances acquises (88), sur leurs croyances, désirs, etc.

### II.4.1.3. Propriétés d'un agent

Tout agent correspondant à la définition précédente possède un certain nombre de propriétés lui faisant valoir le succès qu'il a reçu, ceci grâce à l'efficacité dont ce concept a fait preuve.

Nous distinguons principalement cinq propriétés typiques des agents :

- a. L'**autonomie** : se définit comme une existence indépendante : un agent autonome est, en effet, un agent dont l'existence ne dépend pas de celle des autres (89). Par ailleurs, une autre définition (90) considère l'autonomie comme étant la capacité d'agir et réagir sans pour autant qu'une intervention humaine ou d'autres agents ne soit nécessaire. Cette propriété est particulièrement mise en exergue en comparaison par rapport à un objet. A la différence de celui-ci, un agent autonome est, effectivement, une entité logicielle qui est seule maîtresse d'elle-même

puisqu'il peut répondre positivement ou négativement à un appel de méthodes ou aux compétences qu'il intègre. Il peut ainsi être l'initiateur d'un mouvement précis, réorientant, de ce fait, l'exécution d'un programme dans le sens voulu sans avoir recours à une intervention externe.

- b. La **réactivité** : elle concerne l'aptitude d'un agent à réagir aux modifications survenant dans l'environnement dans lequel il évolue et d'adapter son fonctionnement et comportement (actions et ordre dans lequel elles sont entreprises) en fonction des changements perçus.
- c. La **pro-activité** : tout agent est doté de certaines capacités dont une principale est son aptitude à agir de sa propre initiative pour essayer d'atteindre les buts et objectifs qu'il s'est fixé.
- d. La **continuité** : concerne la capacité d'un agent à demeurer en continuelle activité sans pour autant être provoqué par un stimulus externe. Les agents logiciels sont effectivement souvent implémentés sur des processus légers appelés « *threads* » qui s'exécutent en parallèle. La propriété de continuité est ainsi définie, à la différence d'un objet comme étant la persistance d'une identité et d'un état sur une longue période (91).
- e. La **sociabilité** : la notion de communication intervenant dans les SMA est le principal moteur à un comportement social évolué. En ce sens, les agents sont très sociables et communiquent directement entre eux pour pouvoir échanger des informations et interagir via des langages de communication spécifiques et convenus. Les messages de type informatif ou requête sont le principal générateur d'échanges instanciés par un agent soit pour mener à bien son fonctionnement interne, soit pour instaurer une coopération et/ou coordination de différentes actions mutuelles.

Outre cette panoplie de propriétés faisant la vertu des agents, d'autres viennent s'y ajouter faisant ainsi croître l'intérêt qu'ils apportent. Il s'agit de caractéristiques spécifiques aux SMA dont l'essence est une base communicante qui se cache derrière des formes de travail collectif.

### **L'union fait la force :**

Le règne de l'aspect d'organisation, de communication, interaction et coordination entre les différents agents d'un système apportent un surplus de performances additionnées à celles déduites des différentes propriétés ci-dessus énumérées. Dans ce contexte, Weiss définit un

SMA comme étant un « *un système composé d'un ensemble d'entités autonomes et intelligentes qui coordonnent leurs connaissances pour atteindre un objectif ou résoudre un problème* » (92). Ceci stipule que les intelligences individuelles de chacun des agents sont rassemblées dans un contexte de coopération bien organisée permettant de renchérir l'efficacité du système et ainsi atteindre des performances jamais inégalées, particulièrement quand il s'agit de problèmes complexes. Une simple addition d'agent ne peut subvenir au besoin de résolution de tels problèmes et ne peut non plus définir un SMA, mais plutôt une concordance et organisation dans un contexte de coordination, de collaboration faisant écho avec leur intelligence collective pour la mise en place de systèmes efficaces et performants

Dans ce sens une autre définition citée dans (18) (70) énonce qu' « un système multi-agents est un réseau d'agents solveurs faiblement couplés qui coopèrent ensemble pour résoudre des problèmes qui dépassent les capacités ou les connaissances individuelles de chaque agent ».

### *II.4.1.4. SMA : l'organisation au sens de la coopération*

Les entités d'un SMA doivent cohabiter ensemble dans un même environnement et s'adapter en fonction de ce dernier et des autres entités qui y évoluent. Pour que cela soit possible, une organisation bien définie doit être respectée. Elle énonce la manière dont ces entités doivent agir et interagir dans le respect des 'autres' et de l'environnement et qui leur est inculquée de par une organisation particulière. Celle-ci définit en effet pour chaque agent un rôle particulier qui lui sera par la suite attribué définissant son degré de collaboration avec les autres entités de l'environnement. En fonction de ce rôle, l'agent aura pour mission de fournir un ensemble de services au reste du groupe.

Par conséquent, la notion d'organisation a pour principal but de décomposer le problème initial en un ensemble de tâches élémentaires réparties sur l'ensemble des entités du système. Ces entités se devront d'exécuter leurs tâches en parallèle tout en coordonnant leurs actions. C'est de cette disposition qu'émane la notion d'intelligence collective faisant la puissance des SMA. Une propriété supplémentaire des SMA mérite aussi d'être citée dans ce contexte et qui est celle de l'aptitude à s'auto-organiser ou ré-organiser, provoquant ainsi des changements dynamiques sur l'organisation initialement adoptée lorsque nécessaire (i.e. par besoin de s'adapter à l'environnement). Ce changement, lorsqu'il est opéré, est généralement en faveur d'une meilleure adaptabilité pour faciliter ou bien accélérer la résolution du problème traité.

Des règles particulières régissent ce procédé de changement d'organisation (93) faisant intervenir le système dans sa globalité et non pas un ou plusieurs agents individuellement.

Dans notre étude, nous nous intéressons particulièrement à la notion de base régissant la coordination entre les agents. Interaction et communication, en subordination avec l'intelligence et les capacités des agents, constituent en effet une révolution dans le domaine de l'Intelligence Artificielle Distribuée (IAD).

## II.4.2. Interaction et communication

Les interactions, telles que définies par Ferber dans (84), correspondent à une mise en relation directe et dynamique de deux agents ou plus, et ce par le biais d'un ensemble d'actions réciproques. Telles que énoncées précédemment, ces interactions sont un concept d'ordre d'importance très grand dans les organisations et constituent un concept inhérent et indissociable de celui des SMA. En effet, afin de pouvoir modéliser les comportements sociaux des agents, il faut un minimum d'interactions entre les différents membres. La sociabilité étant une propriété principale dont doit se doter chaque agent.

Une communication est entretenue entre une société d'agents pour assurer ces actions. La notion de communication adhère fortement au concept d'interactions. Celles-ci n'étant pas aussi restreinte aux moyens seuls d'échange de message mais intègre aussi l'échange de données et d'objets tout en obéissant à des règles, des protocoles et des ontologies.

Weiss (94) propose la modélisation donnée à la Figure II.12 afin de schématiser l'ensemble des interactions possibles entre les agents.

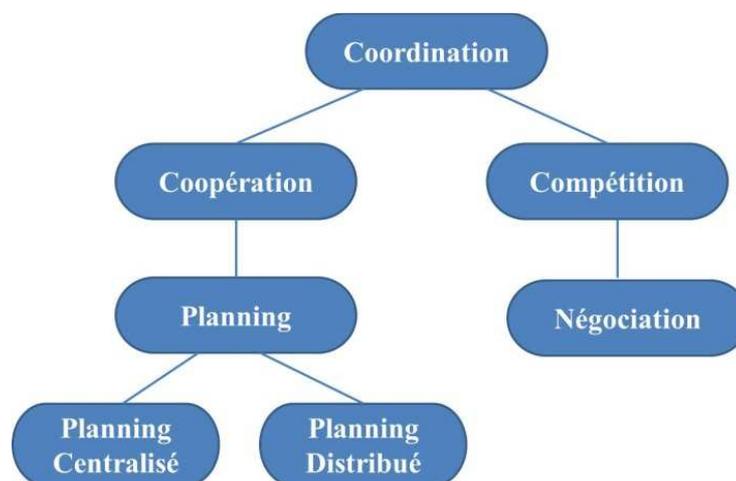


Figure II.12. Les interactions sous leurs différentes formes

Selon le même auteur, la coordination est définie comme étant la « *propriété d'un système composé d'au moins deux agents, exécutant des actions dans un environnement partagé* ». Cette définition a valu à la notion de coordination l'aspect d'obligation d'intégration pour pouvoir synchroniser les agissements et accès des agents à des ressources partagées. Les agents devraient ainsi coordonner leurs actions individuelles les uns avec les autres afin de pouvoir réaliser dans les meilleures conditions un but collectif tout en :

- Evitant les conflits éventuels engendrés par l'accès simultané à des ressources partagées ou pouvant subsister entre des *agents antagonistes*. Ces derniers sont les agents ayant des buts et objectifs contradictoires pouvant amener le système dans des situations de conflits. La négociation est utilisée dans ce contexte pour empêcher le système de converger vers ce piège.
- Améliorant les performances du système de par l'amélioration de l'efficacité et l'utilité de chaque agent. La coopération est un contexte propice pour mettre en œuvre une telle amélioration pour les agents non antagonistes.

### *II.4.2.1. Types de communication*

Conséquemment aux définitions précédentes, nous distinguons essentiellement deux types de communication inter-agents.

#### *II.4.2.1.1. négociation entre les agents*

Il s'agit d'une méthode de coordination qui se base sur le principe de la communication et l'échange de données (informations) entre les agents. Cette méthode permet à une communauté d'agents spécifiques d'atteindre, après avoir établi un processus de communication et d'échanges, un accord mutuel sur la manière d'entreprendre une certaine action bien déterminée. De par la communication ainsi établie, la négociation peut mener dans le cadre d'un accord collectif global à considérer des possibilités de concessions mutuelles, des relaxations de buts initiaux, des mensonges ou des menaces (70). Cette description lui a donc valu le nom de méthode de résolution de conflits et de recherche de consensus (18).

#### *II.4.2.1.2. Coopération*

Du fait de la visibilité limitée que les agents ont chacun pour leur part de leur environnement, leur champ d'action s'en trouve ainsi restreint aux seuls objets de cet environnement dont ils ont connaissance. Par conséquent, résoudre un problème global amène

impérativement l'ensemble des agents à échanger et communiquer en vue de coopérer dans le but de réaliser un objectif collectif. L'échange d'informations dans une communauté d'agents est ainsi consacré au but essentiel de coordination de leurs actions individuelles et locales.

Weiss définit dans ses travaux (94) la coopération comme étant la coordination entre agents non antagonistes et qui cherchent à se satisfaire mutuellement sans se 'gêner'. L'auteur fait référence ici à l'obligation de passer par les concessions et relaxations précédemment énoncées dans le cas d'une négociation. A l'opposé, une coopération, initiée de par le procédé d'échange d'informations, est souvent associée à une collaboration. Cette dernière est une forme particulière d'interactions qui a pour mission d'étudier la manière dont sera réparti le travail et d'allouer conséquemment les tâches ainsi décomposées aux différents agents.

Une fonctionnalité spécifique des SMA qu'est l'allocation des tâches peut procéder de deux manières différentes :

- Soit statiquement : se faisant dès la conception du système et courant ainsi le risque de définir une organisation non-adaptable pour la résolution de problèmes,
- soit dynamiquement : grâce à un processus de coopération entre les agents. Cette coopération a pour effet de rendre dynamique la planification des actions individuelles à entreprendre par chaque agent.

Ceci rejoint la définition donnée dans (95) et reprise par M.A. Kamoun dans (18) décrivant un SMA comme étant « *un ensemble d'entités qui coordonnent leurs connaissances, buts, expériences et plans pour agir et résoudre des problèmes, incluant le problème de coordination inter agents lui-même* ».

En fonction des définitions avancées dans cette section, la typologie des agents peut être identifiée différemment que par rapport à la nature et rapidité des actions comme définie précédemment. Dans ce sens, une classification spécifique a été inspirée de la typologie des interactions reliant les agents entre eux, deux grandes classes sont ainsi distinguées (86) (96):

- les *Organisations Multi-agents (OMA) à structure hiérarchique* : il s'agit de considérer dans ces structures rigides un agent initiateur sur lequel le contrôle est centralisé puisqu'il est le seul moniteur du reste des agents, dits exécutants, et avec lesquels il communique en donnant des ordres. Par ailleurs, parmi les exécutants, il peut y avoir des structures identiques mais à un niveau inférieur de la hiérarchie. Le système dans sa globalité opère dans le seul sens de réaliser l'objectif fixé par l'agent chef

hiérarchique. La Figure II.13 schématise ce type d'OMA reprenant les principales notions de cette structure hiérarchique.

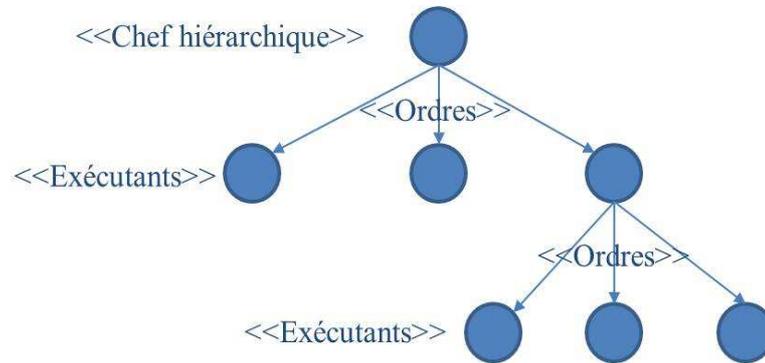


Figure II.13. Structure Hiérarchique d'une OMA (82)

- Les *Organisations Multi-agents* (OMA) à *structure hétérarchique* : elles-mêmes se dissocient en plusieurs types distincts :
  - o Les organisations de type marché : pour un ensemble d'agents exécutants, il peut exister plusieurs coordinateurs sur un même niveau. Le contrôle s'en trouve alors décentralisé et réparti sur l'ensemble de ces coordinateurs qui ont chacun son objectif opérationnel qui lui est propre les distinguant les uns des autres. La plateforme MAGIQUE (96) répond à ce type d'organisation dans laquelle chaque agent coordinateur réceptionne un ensemble d'offres suite au lancement d'une requête pour celles-ci (appel d'offre). Un contrat de transaction est par la suite établi avec l'agent qui présente la meilleure offre.
  - o Les organisations de type communauté : dans ce type d'OMA, un ensemble d'agents possédant les mêmes capacités et interagissant de par l'échange de flux de données se partagent le contrôle du système.
  - o Les organisations de type société : les agents de ce type d'architectures obéissent à certaines règles précises sous le règne desquelles un croisement entre structures hiérarchique et de communauté est mis en place.

Toutes ces notions se réfèrent à l'établissement de canaux de communication entre les agents dans le but de pouvoir échanger et ainsi coopérer et coordonner leurs actions.

#### II.4.2.2. Mise en œuvre de la communication inter-agents

Dans le but de mettre en œuvre les structures d'organisations précédemment décrites faisant la particularité des SMA et mettant en exergue l'aspect collaboratif et coopératif, une

communication entre les agents doit être établie servant le but essentiel de coordonner leurs actions consacrant une intelligence collective au service de l'efficacité. Il existe ainsi deux types de communication :

- **Indirecte** : elle se fait via l'interprétation de résultats de modifications visibles directement sur l'environnement. L'agent *émetteur* apporte ainsi les modifications en question sur l'environnement, celles-ci seront perçues et interprétées par les agents *récepteurs*. Parmi les effets de modifications voulues sur l'environnement que l'on peut discerner, il y a la propagation de signaux ou les traces laissées sur ce dernier.
- **Directe** : souvent associée à une action, elle concerne l'échange de messages directs entre les agents. L'acte de communication décrit à la Figure II.14 (82) répond à ce type de communication où une transmission d'informations directe d'*émetteur* à *récepteur* a lieu.

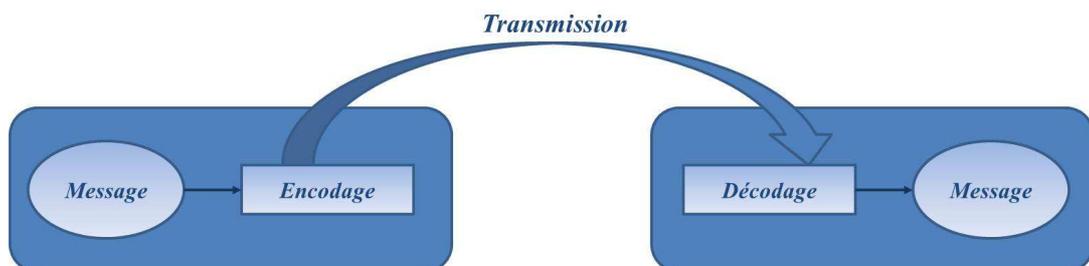


Figure II.14. Acte de Communication : Principe

Le principe de la communication inter-agent, quel qu'en soit le type, doit obéir à certaines règles et conventions régissant les échanges pour pouvoir assurer la compréhension des messages et l'interprétation aisée par tous. Pour cela, des protocoles de communication ont été établis spécifiquement aux SMA. Dans ce contexte, deux langages méritent ici d'être cités vu l'avènement des chercheurs et praticiens sur ceux-ci pour établir les communications au sein des communautés d'agents implémentés. Il s'agit des langages *KQML* (*Knowledge Query and Manipulation Language*) et *ACL* (*Agent Communication Language*). Ces deux protocoles de communication ont été développés en 1993 ; le premier à l'université de Maryland aux USA et le deuxième par la FIPA<sup>30</sup> (Foundation for Intelligent Physical Agents) (97).

### II.4.2.3. Apport des SMA : une mise au point

Tels que décrits dans les sections précédentes ainsi que dans une multitude de travaux de recherche (2) (83) (18), les SMA présentent de multiples avantages. Ces derniers ont fait leur

<sup>30</sup> <http://www.fipa.org/>

grand succès dans plusieurs domaines et le recours des chercheurs et industriels aux plateformes multi-agents pour la mise en place de systèmes performants et efficaces. Parmi leurs points forts, nous pouvons noter essentiellement :

- Leur capacité d'adaptation à n'importe quel type de problème
- L'intégration et la gestion aisée d'une quantité importante d'informations distribuées et hétérogènes faisant évoluer la littérature vers le génie logiciel distribué et l'intelligence ambiante qui règne aussi efficacement sur un ensemble d'applications réparties qu'au sein d'une application unique. Dans ce contexte, sont à noter les systèmes d'informations évolutifs et flexibles et qui intègrent des agents spécifiques dits *Agents d'Information*.
- La personnalisation des services à l'utilisateur de par l'intégration d'agents assistants de l'utilisateur dans son usage du système. Ces agents sont appelés *Agents Interfaces*. De par cette pratique, les systèmes d'information s'en trouvent ainsi personnalisés s'adaptant aux besoins des usagers.
- La prise de décision collective à travers des mécanismes de coordination et collaboration assurées de par la communication, échanges et interactions entre les agents. Dans ce cadre, les *Agents Collaborants* sont connus pour leur efficacité à interconnecter plusieurs systèmes ou résoudre des problèmes de type distribué. Ils associent pour cela leurs compétences individuelles pour mener à bien leurs tâches respectives, et ce dans le but de réaliser un objectif collectif dont la complexité dépasse leurs aptitudes individuelles. Ce domaine d'application est souvent appelé *Intelligence Artificielle Distribuée (IAD)*.
- Etc.

Nombreux sont les facteurs ayant fait la réussite des SMA dans le monde de l'Informatique et du génie logiciel. Différentes études dans plusieurs domaines ont ainsi émergé intégrant ce concept pour la résolution de problèmes variant de complexes à difficiles de point de vue architectural ou autre.

### II.4.3. Les SMA envahissent les domaines de recherche

À tous égards, les SMA sont des systèmes qui ont connu un grand succès dans n'importe quel domaine qu'ils ont incorporé. En effet, que ce soit en recherche opérationnelle, en

traitement d'image, en fouille de données, en ontologie, etc. les SMA ont su s'intégrer et bien s'adapter à la nature du problème lié, efficacement et brillamment.

De plus, l'avancement technologique qui s'inscrit dans le cadre de la révolution que connaissent les STIC depuis plusieurs années déjà ; un avancement surtout observé dans le domaine des réseaux, les appareils mobiles et légers ; a aidé les SMA à s'incruster encore plus et à affirmer leur efficacité.

### II.4.4. Le paradigme agent dans le domaine du transport

Les systèmes multiagents ont été adoptés dans plus d'une approche touchant aux problématiques liées au domaine du transport. Parmi les problèmes inhérents à ce domaine, nous pouvons citer la recherche d'itinéraires, les systèmes d'information et systèmes d'aide au déplacement des voyageurs, les services liés au transport, la gestion du trafic, l'ordonnancement et logistique pour la gestion des places de parking ou de circulation dans un carrefour, etc. les problématiques sont multiples et le nombre de travaux ayant émergé y est proportionnel, voir même beaucoup plus important. Traitant d'optimisation pour améliorer la qualité des services offerts ou tout autre concept, nous nous intéressons particulièrement ici aux études qui ont adopté le paradigme agent et nous en citons quelques uns suivant le problème traité. Nous nous focalisons particulièrement ici sur le domaine du transport où plusieurs modèles adoptant les SMA comme support de base existent. Parmi les travaux qui méritent d'être cités, ceux qui se sont intéressés aux problématiques de :

- Simulation de trafic automobile (88).
- Gestion des perturbations et régulation dans les réseaux de transport multimodal (23) (70).
- Recherche et composition d'itinéraires optimisés pour l'aide au déplacement des voyageurs (2) (18) (20).
- Surveillance des véhicules automatisés (98).
- Contrôle du trafic aérien (99).
- Transport de marchandises (100).
- Aide au pilotage des avions militaires (101).
- Gestion du trafic urbain (102).
- Service de mobilité (19).

- Etc.

#### II.4.5. Les techniques multi-agents intégrées au concept de covoiturage

Grâce à ses multiples contributions faisant la réussite d'une multitude de systèmes pour la résolution de problèmes complexes, plusieurs chercheurs ont ainsi concentré leurs efforts sur ce concept, non pas nouveau mais très innovant, et bénéficiant par la même occasion de l'apport considérable des nouvelles technologies de l'information et de la communication. La problématique de covoiturage n'a pas été épargnée non plus, donnant lieu à l'incorporation de ce concept dans quelques systèmes de covoiturage passés en revue dans (103) (104). Dans ce cadre favorable à l'efficacité et la rapidité de la communication (entre les utilisateurs et le système par exemple), chercheurs et praticiens ont ainsi cherché à tirer profit de l'évolution remarquable que connaît le domaine des réseaux et technologies de l'information et de la communication. Ceci a été enregistré dans un cadre révolutionnaire du covoiturage qui était resté, jusqu'ici, au stade de l'application pratique, sous forme de support en ligne pour la dépose et consultation d'offres et demandes.

En effet, compte tenu de ce qui a été avancé dans le précédent chapitre (*c.f. chapitre I*) présentant une étude bibliographique du concept de covoiturage, l'étude a révélé que Les systèmes existants présentent un déficit assez important. Les limites alors considérées concernent notamment et principalement l'automatisation des tâches et fonctionnalités incombant à un système de covoiturage, les aspects d'optimisation, de sécurité, de communication, etc. Quelques ébauches de systèmes basés sur le concept multi-agent (105) (104) sont ainsi venues essayer de palier à ces problèmes assurant essentiellement l'automatisation du processus de recherche grâce à des agents personnels associés aux utilisateurs émettant des requêtes. Cependant le choix final revient toujours à l'utilisateur d'opter pour l'une ou l'autre des solutions extraites et pouvant correspondre au trajet qu'il aurait demandé. Aussi, ces systèmes restent toujours limités en matière d'optimisation et de temps réel.

Comme la théorie des SMA a remarquablement marqué le domaine de l'analyse et description des systèmes de gestion de trafic (106) dans les services de transport ; elle est aussi venue s'incruster dans le concept de covoiturage. Elle a ainsi pour effet notoire de faire croître l'autonomie des composantes et acteurs intervenant dans de tels systèmes, facilitant par la même occasion l'intégration de plusieurs plateformes dédiées (104) (106). Afin de

mieux décrire ces systèmes, dans le domaine du covoiturage en particulier, nous citons à titre d'exemple les travaux qui ont tourné autour de ce concept pour la mise en place d'une architecture multi-agent au service des fonctionnalités intégrées dans les systèmes de type support classique basé sur le web (105). Les fonctionnalités principales à noter dans ce type de structures concernent essentiellement la gestion des données de covoiturage dont dispose le système mais aussi la recherche, à la place des usagers, d'offres correspondantes à leurs demandes. Cependant dans la plupart des systèmes ainsi élaborés, la prise de décision finale reste le lot de l'utilisateur demandeur du service de covoiturage. En effet, très rares sont les systèmes inscrits dans le paradigme agent en coalition avec les méthodes d'optimisation.

Ce qui est par ailleurs à noter dans les systèmes classiques de covoiturage, c'est qu'il s'agit généralement de mettre en place des agents logiciels pour représenter un utilisateur et travailler 'virtuellement à sa place'. Ces agents sont appelés « *Super Agents* » (105).

Parmi les systèmes proposés dans la littérature, nous pouvons citer la plateforme *MobiAgent*<sup>31</sup> (107) où le paradigme agent a été introduit pour bénéficier d'une facilité d'accès à certains services (e.g. recherche dans le web, applications de contrôle à distance, etc.). Ces services seront désormais accessibles grâce au concept agent pour tout utilisateur disposant d'un smartphone ou *PDA*. Pour ce qui est de la démarche adoptée, il s'agit de créer pour chaque utilisateur, qui se connecte et émet une requête de covoiturage, un Agent Personnel (*Personal Agent : PA*) dans un serveur centralisé. Cet agent spécial sera responsable de chercher à la place de l'utilisateur et en fonction de l'offre disponible, celle qui correspond aux spécificités de sa demande de déplacement. Le PA continue de ce fait son activité indépendamment de l'état de connectivité de l'utilisateur auquel il se réfère. En effet, après avoir spécifié les paramètres du déplacement requis, l'utilisateur se déconnecte et le PA continue à travailler à la recherche d'une offre adéquate ; laquelle pourra être téléchargée par l'utilisateur lors de sa prochaine connexion, notamment à la suite de la réception d'une notification par SMS de la part du PA l'informant que des résultats ont été générés. D'autres chercheurs ont aussi élaboré dans le même contexte l'idée d'un système qu'ils ont nommé *Andiamo* (104) et qui répond au même principe que *MobiAgent*. Ces auteurs proposent ainsi de chercher via des PA les offres correspondantes aux demandes émises par les usagers, assurant automatiquement le processus de '*matching*' de celles-ci. Par ailleurs, ces mêmes auteurs se focalisent particulièrement sur la problématique d'accès à la plateforme multi-agent. La gestion de l'accès au système est perçue, dans ce contexte, sous un angle '*technologique*' ;

---

<sup>31</sup> <http://www.docstoc.com/docs/58249789/Genghis---A-Multiagent-Carpooling-System>

consacrant pour cela les outils adéquats pour mener à bien cette fonctionnalité du système. En termes de technologies, les auteurs de ces travaux se réfèrent à l'usage des architectures basées sur les ToothAgent (108). Dans ce type d'architecture, une application multi-agent est accessible via des périphériques et appareils dits « *légers* » qui sont munis de la technologie « *Blue Tooth* ».

Il est vrai que, ainsi décrits, les systèmes de covoiturage présentent l'avantage de la facilité pour l'utilisateur (i.e. facilité d'accès en plus de l'automatisation du service) vu la notoriété du paradigme agent dans ce domaine. Cependant la plupart de ces systèmes passent sous silence certains aspects pourtant jugés essentiels. Ces aspects se réfèrent essentiellement à la qualité du service au profit de l'amplification de la satisfaction de l'usager. Parmi les principales notions qui s'inscrivent dans ce contexte, l'optimisation et le temps réel font partie des concepts primordiaux qui s'attachent à une qualité de service optimale nécessaire pour faire le succès d'un système de transport en général et de covoiturage de manière spécifique. De plus, des aspects tels que la non-tolérance à la flexibilité dans les trajets permettant des itinéraires composés peuvent induire à des réponses négatives, dégradant ainsi la qualité du service avec un taux de réponse positives bas, voire même nulle.

Compte tenu de l'étude de ces systèmes conjointement à l'état de l'art présenté dans le précédent chapitre (*c.f. Chapitre I*) et sur la base des notions théoriques et pratiques présentées précédemment relevant du paradigme agent et du concept d'optimisation, nous nous proposons de mettre en place un système offrant les services de covoiturage dynamique sur la base d'un axe pivot qui est l'alliance des systèmes multi-agent avec l'optimisation.

### **II.5. CODAC : l'intelligence collective via des agents optimisateurs en coalition**

L'intérêt pratique des SMA dans les systèmes logistiques optimisés relève essentiellement de critères relatifs aux propriétés d'exécution (consommation mémoire, temps de réponse et performance du système), en particulier dans les services en temps réel. C'est dans ce contexte que s'inscrivent nos travaux pour la réalisation d'affectations optimisées des véhicules aux usagers en quête de performances techniques.

A titre de rappel, la Figure II.15 vient reproduire une schématisation du concept de covoiturage tel que nous le concevons et tel que nous en percevons le cadre de fonctionnement pour fournir un service des plus attrayants.

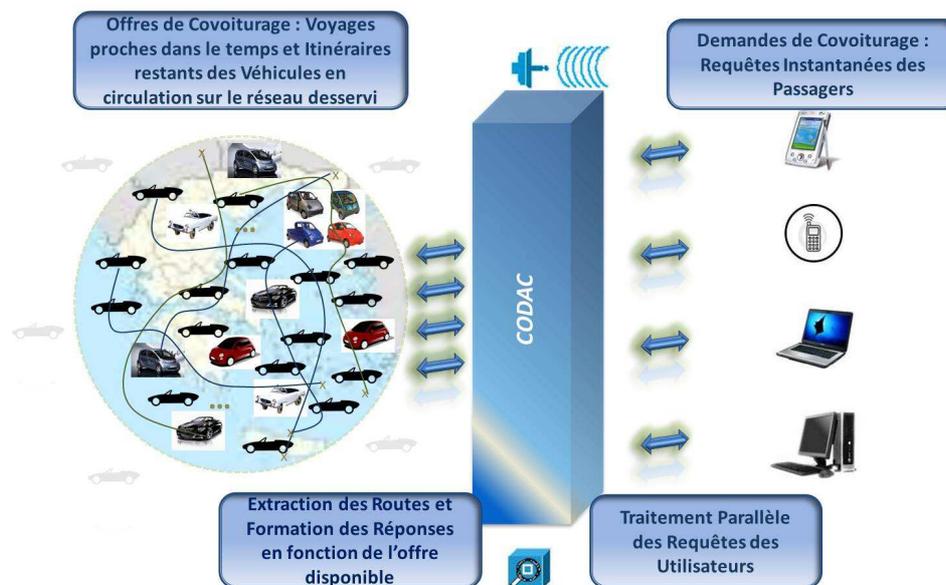


Figure II.15. Traitement instantané : Affectation dynamique des véhicules aux usagers

Le système que nous nous proposons d'implémenter constitue ainsi un axe médian assurant la communication, interaction et coordination entre deux plans de l'espace dont la fusion garantit une représentation qui épouse parfaitement la forme réelle du réseau géographique. Ce réseau est représenté concrètement dans la partie gauche de la modélisation donnée à la Figure II.15. Cette concrétisation est faite au travers d'une vision réaliste basée sur des informations 'tangibles' puisqu'elle se base sur des offres de covoiturage : concrètement les voitures déjà en cours de circulation ou pas encore (voitures grisées positionnées autour et en dehors du globe et dont le voyage n'a pas encore été entamé). En effet, du moment où un conducteur insère une offre au système et même si le voyage correspondant n'a pas encore commencé (temps de départ du véhicule non encore atteint), elle est tout de même considérée puisqu'elle peut constituer une réponse potentielle à une ou plusieurs demandes si les contraintes de correspondance en sont vérifiées.

Le fait de disposer des informations complètes sur ces voitures ainsi que les représentations de leurs itinéraires respectifs, actualisées en temps réel, a valu à cette modélisation du réseau le qualificatif de tangibles à notre sens. En contrepartie, la modélisation fournie dans la partie de droite de la Figure II.15 considère les utilisateurs émetteurs de demandes de covoiturage représentant une communauté de piétons que nous qualifions ici de modélisation virtuelle. Pourquoi virtuelle ? Parce que le traçage des itinéraires est uniquement au gré de l'offre existante. Outre les spécificités d'origines et destinations requises, les covoiturés (potentiels passagers) existent en effet pour le système au travers de leurs appareils mobiles. Leur existence 'physique' considérée au niveau du système

de covoiturage sera retracée en fonction de celle des voitures (i.e. leurs offres et trajets correspondants).

Sur la base de cet énoncé, le système que nous mettons en place propose d'élaborer des itinéraires et voyages éventuellement communs sur la base de l'offre disponible. Ceci est réalisable à condition de trouver les bonnes correspondances entre les paramètres de déplacement souhaité de chacun des utilisateurs avec ceux offerts par les conducteurs et qu'il serait judicieux de réunir au sein d'une seule voiture pour un même voyage ou parties de trajets.

Par ailleurs, nous nous proposons aussi d'implémenter de par le système CODAC une plateforme automatique capable de trouver non seulement l'offre adéquate susceptible de répondre à un ou plusieurs besoins de déplacement mais aussi la meilleure qui existe. Cet aspect des traitements n'a jamais été abordé auparavant. Notre système représente, de ce fait, une innovation dans la mesure où de tels degrés d'automatisation dans le traitement des affectations, de flexibilité, et d'optimisation n'ont jamais été réalisés au niveau des approches existantes.

Orientant nos travaux dans ce sens, nous avons choisi d'intégrer le concept d'optimisation pour offrir la qualité de service escomptée au profit des covoitureurs et covoiturés. Ainsi, pour satisfaire au mieux les usagers de notre système, nous proposons des fonctionnalités qui incorporent des méthodes d'optimisation, et ce en quête de l'optimisation du taux de réponses positives en premier lieu et de la qualité de ces réponses en second lieu. L'optimisation dans notre approche a été brièvement énoncée dans la *section II.3* de ce chapitre et va être longuement détaillée et formalisée dans le chapitre suivant (*c.f. Chapitre III*) et dont nous reprenons quelques notions dans la section suivante.

### II.5.1. Optimisation dans le covoiturage dynamique

Considérant un seul critère ou plusieurs, l'optimisation ne peut qu'être bénéfique au sens des usagers du service de covoiturage. Dans le domaine du transport en général, les systèmes les plus réussis sont ceux qui offrent une assistance au voyageur dans ses déplacements tout en lui assurant de trouver les meilleures offres qui lui correspondent. Aujourd'hui, même les opérateurs de transport publique sont en quête de performances par rapport à cet aspect d'optimisation afin d'offrir une qualité de service susceptible d'attirer les individus. Dans cette quête des performances, des méthodes d'optimisation ont été consacrées implémentant des algorithmes jugés adéquats dans le cadre des situations auxquelles ils sont confrontés.

C'est dans cette vision des choses que nous nous sommes placés pour offrir un service de covoiturage optimisé et performant conformément aux attentes des usagers. Plusieurs critères peuvent être considérés à cet effet selon que l'on considère l'intérêt des usagers individuellement ou bien l'intérêt de la collectivité dans un sens plus global. Enumérés dans le chapitre suivant afin d'introduire le concept d'optimisation dans le covoiturage de manière générale et dans notre système en particulier, les critères pouvant être considérés constituent un large panel. Le choix portant sur l'un ou l'autre de ces critères est du ressort de l'administrateur du système ou de l'utilisateur lui-même (dans le cas où une spécification des préférences peut être assurée via une plateforme interactive IHM) et est abouti au détriment d'autres au gré de l'utilisateur.

Privilégier un critère au dépend des autres s'avère toujours être un choix difficile et compromettant. Afin de répondre le plus efficacement possible au besoin en matière d'optimalité au niveau des traitements effectués et ainsi des résultats fournis, la possibilité de considérer plusieurs critères en même temps devient évidente, voire même essentielle. En effet, deux critères ou plus peuvent être considérés conjointement dans le cadre d'une optimisation multiobjectif. Les systèmes pourvus de tels moyens fournissent dans la majorité des cas des réponses optimisées, si ce n'est optimales, permettant ainsi de rechercher les solutions qui optimisent une combinaison de critères parfois contradictoires. Un compromis est dans ce cas recherché en fixant une fonction objectif en adéquation avec les exigences des utilisateurs, permettant ainsi d'agréger ou de classer l'ensemble des critères suivant un ordre de préférences (généralement défini par l'utilisateur lui-même).

Dans nos travaux, nous privilégions des critères plus en rapport avec les attentes des usagers, à savoir le confort de l'utilisateur essayant de minimiser le nombre de changements de véhicules qu'il aura à effectuer durant son voyage. Sur la base du premier ensemble de réponses valides et optimisées par rapport à ce critère, celle qui optimise son trajet en termes de durée de parcours globale sera finalement sélectionnée. Ceci notamment dans une optique d'optimisation multi-objectif incluant les temps d'attente des véhicules ainsi que les temps de parcours passés dans chacun de ces véhicules.

Ceci étant, considérer un volet optimisation des qualités de réponse (critères d'optimisation) dans des systèmes de recherche opérationnel peut induire à une complexité exponentielle, tel que c'est le cas pour le problème de covoiturage dynamique. L'étude fournie à la section II.3.2. a effectivement révélé le fort potentiel des systèmes de covoiturage dynamique optimisé à tomber dans le piège de l'inefficacité ainsi que de la complexité et la

lenteur des traitements puisqu'étant classés sous la catégorie des problèmes *NP-Difficiles*. Il est donc essentiel de considérer des stratégies de résolution efficaces afin de contrer l'explosion combinatoire, surtout dans le cadre d'un système dynamique optimisé multicritère. Comme les performances d'exécution sont d'une importance égale, ceci implique de considérer le problème de complexité à priori exponentielle du service proposé.

### II.5.2. Les SMA au service de l'efficacité technique

La complexité combinatoire est maintenue tout au long de la résolution du problème de covoiturage dynamique optimisé en pleine ligne de mire.

Dans le même temps, réunir les critères d'efficacité des traitements et de performances techniques constitue un défi difficile à relever surtout en présence de données dynamiques dans un environnement instable. Par ailleurs, l'évolution du génie logiciel vers l'intégration de composantes autonomes que sont les agents et qui sont dotées de capacités notoires a contribué à rendre ce défi relativement plus facile et notre objectif plus accessible. Largement définis dans ce qui a précédé, les agents sont un concept remarquable de par leurs propriétés, capacités, comportements et interactions les rendant capables d'opérer intelligemment et efficacement que ce soit sur le plan individuel ou collectif.

Vus sous cet angle, les *SMA* sont ainsi considérés comme la concrétisation d'une rencontre entre le génie logiciel et l'intelligence artificielle distribuée. L'apport notable des systèmes distribués est visible de par le contexte collaboratif dans lequel s'inscrivent les activités et comportements des agents mettant en place un contexte d'intelligence ambiante. L'autonomie des agents permet par ailleurs au concepteur de se concentrer sur une partie *humainement appréhendable* du logiciel.

Tentés par une telle efficacité et notre problématique s'avérant en plus des plus complexes nous avons tenu à profiter de ces avantages pour réussir un système compétitif de point de vue efficacité et performance. Présentant l'atout principal de mise à bas des problèmes de complexité des systèmes informatiques, nous considérons alors le paradigme agent pour la résolution de notre problématique. Le *chapitre IV* de ce document en fait une description approfondie présentant l'architecture multi-agent ainsi que le principe d'interaction et coalition entre les différents acteurs intervenants (ensembles d'agents initiés de par cette modélisation architecturale) pour la résolution du problème d'affectation dynamique optimisé des covoiturés aux voitures de covoiturage.

### II.5.3. L'alliance entre les SMA et les techniques de l'Optimisation

Visant une expansion de haut niveau pour une mise à l'échelle d'important calibre, nous considérons un système qui puisse intégrer un nombre d'utilisateurs aussi large que possible et capable de surcroît de traiter instantanément une quantité d'informations combinatoires. Ces informations sont en outre susceptibles de changer d'un instant à l'autre rendant ainsi l'environnement fortement évolutif et instable. Le concept des SMA s'accorde parfaitement à ce contexte et est capable d'apporter l'efficacité requise dans le cadre d'une dynamique contraignante. Compte tenu de ce constat, les agents sont donc bien appropriés vu leurs aptitudes d'intégration et d'adaptation efficaces sous ces contraintes. Ils sont en effet capables de réaliser des traitements sur des données hétérogènes et dynamiques grâce aux différentes coopérations qui peuvent subsister mais aussi aux perceptions continuellement mises à jour qu'ils ont de l'environnement dans lequel ils évoluent.

Les différentes perceptions de l'environnement sont au gré des changements et modifications pouvant survenir sur l'état d'un ou plusieurs objets intégrés dans celui-ci. Suivant le principe des SMA, détaillé précédemment (*c.f. II.4.1*), les agents évoluant et opérant dans un environnement où de tels changements surviennent, doivent pouvoir adopter la bonne pratique à suivre en changeant leur comportement et probablement en réorientant dans un sens différent leurs prises de décisions. Par ailleurs, il est essentiel que cette réorientation aille dans la bonne direction et pourquoi pas diriger les actions de l'agent vers la recherche de l'optimalité. Par conséquent, afin de faire en sorte que cette réorientation se fasse pour le mieux et optimiser leurs choix, les agents intervenant dans cet environnement peuvent intégrer des méthodes d'optimisation adaptées à leurs compétences et connaissances.

Optimisation et agents vont ainsi de pair pour réaliser un service optimisé tout en étant efficace et performant. L'alliance de ces deux concepts s'impose ainsi d'elle-même et a pour effet d'incorporer le sens de l'optimisation dans les agents (i.e. optimisation locale intégrée dans les actions et comportements des agents), eux-mêmes régis globalement par des méthodes d'optimisation distribuées. Ce contexte est d'autant plus explicite que s'il est schématisé. La Figure II.16 vient ainsi illustrer l'intégration croisée des deux concepts d'agents et d'optimisation. Ce croisement est concrétisé de par les notions de coalition, collaboration et coordination entre les agents consacrant l'optimisation locale ainsi que les

communications et interactions au profit de l'intelligence collective et donc de l'optimisation distribuée.



Figure II.16. CODAC : Optimisation dans et autour des agents

Un type particulier d'agents est important à noter ici et c'est l'Agent Optimisateur, noté AO, auquel la tâche principale de recherche d'affectations possibles et par la suite optimisées incombe. Ce type d'agent défini en détail ultérieurement (*c.f. Chapitre IV*) est doté d'un comportement spécifique principalement voué à l'exécution d'algorithmes d'affectations optimisées selon les approches proposées. Ainsi selon la méthode souhaitée ; choisie parmi les deux heuristiques spécifiques au CDO que nous avons énoncées précédemment et que nous détaillons ultérieurement (*c.f. Chapitre IV*) ; l'AO adopte le comportement adéquat pour ainsi orienter ses décisions à chaque fois vers un sens différent qu'est celui du choix de l'utilisateur (administrateur du système) par rapport à la méthode à exécuter.

L'intégration du concept d'optimisation aux SMA dans ce cadre a servi à la mise en place de services optimaux en termes de critères de satisfaction par rapport aux attentes des usagers (coût, temps de production, rentabilité, etc.). Cette intégration vient effectivement contrecarrer le problème de complexité du CDO (*c.f. II.3.2*) de par la mise en place d'un procédé d'optimisation distribuée plaçant notre système dans un contexte de *multi-threading*. Pour faire bien, nous avons opté pour la mise en place de traitements logiciels distribués à l'image de la distribution des covoiturés et covoitureurs sur le réseau de desserte. Orientant nos efforts dans ce sens, nous nous sommes focalisés sur un principe de décomposition qui mette en exergue la répartition difforme des usagers et donc les zones de concentration géographique de ceux-ci dans des groupements différents favorisant ainsi leur traitement parallèle. Cette décomposition de l'espace géographique desservi a de ce fait abouti à la mise en place d'une

modélisation spécifique sous forme de graphe dynamique distribué à l'image duquel des processus légers sont lancés en parallèle pour le traitement simultané des différents groupements érigés.

## **II.6. Un Graphe Dynamique Distribué pour la modélisation**

En s'inspirant de travaux menés dans ce sens (18) (2) (69), la modélisation sous forme de graphe dynamique distribué favorisant la mise en place d'une architecture distribuée s'est avérée efficace. Nous nous sommes donc employés à adapter cette modélisation dans le cadre de notre travail où le réseau desservi répond aux critères de dynamique dans les environnements ubiquitaires.

En orientant nos travaux dans cette direction, le choix des systèmes multi-agents y étant largement approprié, nous avons voulu concéder à la mise en place d'une architecture multi-agent des plus efficaces et bien adaptée à notre problème.

Deux énoncés spécifiques ont été considérés dans cette optique :

- Les SMA sont un concept favorable à la mise en place d'une plateforme logicielle composée d'un ensemble d'entités intelligentes communicant par le biais de langages spécifiques et assurant chacune une tâche de complexité moindre par rapport à tout le processus.
- Les données devant être traitées par ces entités sont réparties sur un réseau géographique très vaste et où les informations sur les voyageurs (covoitureurs et covoiturés) obéissent à des lois de distribution aléatoires. Selon les besoins du moment, ces informations sont ainsi réparties différemment à chaque fois que le système est lancé, donnant ainsi lieu à des représentations différentes d'un instant à l'autre et mettant en exergue une distinction entre les zones géographiques de densité plus ou moins importantes.

Prenant en compte ces données, nous nous sommes voulu efficaces en essayant de centraliser en premier lieu le traitement d'optimisation dans les zones de concentration de population des individus (dans des AO qui en seront responsables) puis de s'agrandir à une échelle plus large via les interactions entre les différents AO intervenant. L'architecture multi-agent que nous avons voulu appropriée à ce principe répond ainsi à une modélisation sous forme de GDD que nous énonçons et formulons dans le chapitre suivant.

## II.7. Conclusion

Dans ce chapitre, nous avons présenté les principaux aspects théoriques relatifs à notre approche. Nous avons en effet commencé par présenter le concept d'optimisation, les principales notions qui y sont reliées et avec, une brève revue des systèmes existant dans la littérature et qui ont traité de ce concept.

En second lieu, les systèmes multi-agents ont été présentés en mettant en exergue les contributions qu'ils sont susceptibles d'apporter notamment et particulièrement lorsqu'il s'agit de traiter des problèmes d'importante complexité, comme c'est le cas pour la problématique du covoiturage dynamique optimisé (CDO) dont l'étude fournie au niveau de ce chapitre le prouve largement.

Considérant la problématique ainsi énoncée, nous adoptons dans notre approche un croisement mutuel entre ces deux concepts. Dans le but premier de réaliser des traitements optimisés efficacement et dans des délais de réponse raisonnables, nous avons en effet opté pour une alliance des SMA et de l'optimisation dans une architecture distribuée. Celle-ci a été mise en place de par une modélisation particulière inspirée des GDD et concrétisée de par un principe de décomposition. Ceci a pour principal objectif de mettre en relief la répartition naturellement déséquilibrée des usagers sur l'aire géographique considérée et qui donnerait intuitivement lieu à des zones distinctes réparties selon la densité de chacune. Les notions de classification aidant, le principe de décomposition choisi sert ainsi à mettre en relief ces concentrations géographiques.

De la théorie à la pratique, il n'y a qu'un pas et il s'agit dans notre cas de formuler les fondements de notre approche et énoncer les méthodes et algorithmes adoptés en vue de réaliser les objectifs que nous nous sommes fixés de par ces travaux.

### III. CHAPITRE III

MODELE REPRESENTATIF DU SYSTEME :

MISE EN PLACE D'UNE ARCHITECTURE

DYNAMIQUE DISTRIBUEE

### **III.1. Introduction**

Compte tenu du problème d'explosion combinatoire dans la résolution des systèmes de covoiturage dynamique optimisé, le choix d'une méthodologie de résolution adaptée s'impose. En effet, afin de contrecarrer le piège de la complexité exponentielle, étant particulièrement placés dans un contexte de système opérant en temps réel, s'appliquer à une optimisation des problèmes difficiles s'avère généralement de première nécessité. De ce fait, dans le but de restreindre les limites de la dynamicité, ces systèmes requièrent parfois des technologies avancées et des méthodologies de résolution efficaces et non contraignantes. Une telle stratégie est généralement visée dans les systèmes d'optimisation dans le but essentiel de pouvoir respecter un seuil minimal de satisfaction de l'utilisateur ; une satisfaction évaluée en termes de qualité de service et de temps de réponse. Il est donc impératif que les solutions envisagées soient en faveur d'une qualité de service optimale et sans conséquences néfastes sur les critères d'exécution, plus particulièrement le délai de réponse.

L'objectif que nous visons en premier lieu de par nos travaux étant d'assurer la mise en place d'un système de covoiturage dynamique, les performances du système en question doivent être à la hauteur des attentes des usagers. Partant de ce principe de base, nous nous sommes focalisés sur la mise en place d'un système dynamique optimisé.

Dans cette optique, nous nous focalisons particulièrement dans nos travaux sur l'aspect optimisation afin de pouvoir conduire une démarche efficace sous tous les angles. Ainsi, nous en sommes venus à conclure que deux volets d'optimisation sont requis pour la bonne mise en œuvre d'un système de covoiturage dynamique. Le premier volet touche à la satisfaction des utilisateurs par rapport à la qualité de la réponse fournie optimisant les préférences des utilisateurs (i.e. critères voulus) sur les déplacements générés. Aussi, le second volet est-il relié à la qualité du service vu la difficulté et le manque d'aisance pour approcher un tel problème. En effet, la problématique d'une complexité combinatoire revient régulièrement comme un leitmotiv auquel nous devons faire face dans les différentes phases de construction de notre approche (i.e. analyse, conception, modélisation, implémentation, etc.). Il est donc primordial de pouvoir adopter les démarches adéquates afin d'en effacer l'empreinte sur les performances du système.

## III.2. Système de covoiturage dynamique optimisé

Dans une optique d'optimisation des services de covoiturage, nous optons pour une solution qui allie le traitement optimisé des requêtes instantanées des utilisateurs aux critères de performances du support mis à leur disposition. Vue sous cet angle, l'optimalité des réponses fournies est logée dans le rapport à l'utilisateur basé sur les critères à optimiser (i.e. choix de l'utilisateur, soit qu'il privilégie le coût, le temps de trajet, etc.) aussi bien que dans les propriétés d'exécution, à savoir un temps de réponse raisonnable et une allocation optimisée des ressources nécessaires (e.g. espace mémoire, ressources CPU, etc.). Une optimisation, en ce sens, perçue dans la qualité des solutions produites et sans conséquences nuisibles sur les performances du système. Ce dernier doit notamment cadrer avec l'instantanéité des traitements pour épouser parfaitement la forte variabilité des données du réseau de covoiturage.

### III.2.1. Optimisation dans les systèmes de covoiturage dynamique

Les moyens de transport à la base privés devenus publics (i.e. la voiture personnelle) ont donné naissance à des systèmes de transport à priori individuels mais devenus collectifs (e.g. services de covoiturage, autopartage, etc.). De tels systèmes ont vu le jour suite au développement de plusieurs travaux visant l'amélioration de la qualité de la vie en considérant particulièrement le développement du domaine de transport dans un contexte mélioratif (c.f. *Chapitre I*). Amélioration en termes de services fournis aussi bien sur le plan individuel que collectif et dont l'objectif premier consiste essentiellement à améliorer les conditions de vie des personnes d'un côté et augmenter leur satisfaction de l'autre. Cependant, aussi évolués soient-ils, ces systèmes ne sont pas passibles d'efforts supplémentaires afin d'en optimiser le fonctionnement (c.f. §I.5.2.4). Les aspects d'automatisme et d'optimisation (c.f. §II.2.7.2) étant parmi les principales lacunes des systèmes relatés dans la littérature, nous nous proposons de remédier à ces limites et ainsi combler ce déficit. En effet, dans une tentative d'amélioration de l'existant, nos travaux ont débouché sur la mise en place d'un système complètement automatisé, développé sur la base d'une optimisation des traitements sous tous ses aspects. L'« *optimisation distribuée* » est, effectivement, le maître mot dans les deux approches, solutions de covoiturage dynamique, que nous proposons. Ces approches ont fait l'objet d'un travail laborieux aboutissant à la mise en place d'un support pourvu d'une multitude de fonctionnalités pour une qualité de service satisfaisante. Le système proposé est

décrit tout au long de ce rapport avec les détails de conception et d'implémentation qui s'y rapportent et ce avec un fort accent d'optimisation. Dans cette partie en particulier, nous nous focalisons essentiellement sur les aspects d'optimisation telle que adoptée au niveau de nos travaux mais aussi sur le choix de modélisation dans une section ultérieure.

### *III.2.1.1. Le covoiturage : une solution d'optimisation en soit*

Dans le cadre d'une mobilité avancée, il est nécessaire de considérer l'aspect optimisation dans les services de transport mis à disposition des individus abonnés de ces systèmes. Plusieurs recherches ont été menées dans ce sens et convergent vers un but essentiel qui est celui d'assurer l'intérêt des personnes dans un contexte communautaire et surtout vital. En effet, vu sous cet angle, l'intérêt principal de mettre en place un système de transport, quel qu'en soit la forme, est avant tout de fournir des conditions favorables pour l'évolution des individus dans un environnement sain. Mettre le monde à l'abri des menaces (e.g. réchauffement climatique, détérioration de la qualité de la vie, etc.) qui pèsent continuellement dessus est l'un des plus grands objectifs visés à l'échelle internationale. Un objectif particulièrement difficile à atteindre quand il s'agit de menaces dangereuses et qui de surcroît n'arrêtent pas d'évoluer avec l'évolution des technologies. Plusieurs associations et organismes se sont employés à étudier la nature de ces menaces et l'ampleur de leur impact sur l'environnement et la qualité de la vie aussi bien qu'à rechercher les solutions qui permettent d'y remédier et alléger leur effet. Un effet ressenti de par la détérioration des conditions environnementales, le réchauffement climatique, etc. principalement dus aux taux importants d'émission de CO<sub>2</sub> et gaz à effet de serre. L'un des facteurs les plus notoires causant ce taux à augmenter encore plus est l'usage abusif de la voiture personnelle et sa propagation dans le monde entier (*c.f. Chapitre I*). Le domaine du transport est ainsi visé en particulier et de multiples recherches ont été développées dans ce sens. Le recours à des technologies propres ainsi que la mise en place de systèmes de transport peu polluants sont nécessaires.

Outre ces considérations d'ordre écologique, d'autres se rattachant à l'aspect économique et qui viennent dénoncer l'impact négatif et l'appauvrissement des portefeuilles budgétaires des individus sont tout aussi importants. L'acquisition et maintien d'un véhicule personnel demeurent un luxe nécessitant un budget assez important. L'ensemble des considérations prises dans cette mesure attestent de l'importance de pouvoir trouver une solution de moins lourde incidence sur la qualité de vie ainsi que sur les budgets alloués au transport. Ainsi,

dans le cadre d'une mobilité avancée, des services tels que l'autopartage et le covoiturage ont vu le jour et permettent d'assurer un service d'aide à la mobilité pratique, flexible et économique tout en satisfaisant les objectifs de réduction de la pollution.

Il est vrai que les systèmes de partage de véhicules ont aidé à l'amélioration des conditions non seulement environnementales dans lesquelles évoluent les individus mais aussi économiques. En effet, le concept de covoiturage présente particulièrement de multiples avantages qui ont le mérite d'en avoir fait un service très sollicité et compétitif. Parmi ces avantages, nous citons particulièrement la réduction du nombre de voitures en circulation au kilomètre. La voiture personnelle étant devenue, grâce à ce concept, un moyen de déplacement collectif accessible au grand public ; elle accueille de ce fait des passagers supposés voyager individuellement dans d'autres voitures. Ces systèmes représentent en ce sens un moyen efficace pour minimiser le taux d'émissions de CO<sub>2</sub> et gaz à effet de serre évacués par les voitures. De plus, ils ont aussi des répercussions positives sur les conditions financières puisqu'ils permettent de réduire les budgets alloués au transport.

Par conséquent, le concept en soit du covoiturage est de grand calibre en matière d'apport et d'optimisation. Il est ainsi considéré comme un service original et innovant pesant lourd en termes d'optimisation, essentiellement par rapport aux critères financier et environnemental. Cependant, outre ces critères et afin de pouvoir offrir une qualité de service irréprochable, d'autres aspects peuvent être considérés.

#### *III.2.1.2. Optimisation : Une panoplie de critères*

Outre l'optimisation implicite des systèmes de covoiturage, la satisfaction des usagers de tels services est comptée comme principal caractère d'attraction et peut faire intervenir un ou plusieurs critères d'optimisation au niveau des traitements pour la génération des solutions d'affectation. Il s'agit particulièrement de critères concordant avec les besoins et désirs des utilisateurs, soit qu'ils préfèrent voyager avec un moindre coût, en un minimum de temps, avec un maximum de confort (i.e. en limitant les transferts d'une voiture à une autre), etc.

Ainsi, pour garantir l'optimalité des services offerts, de tels systèmes doivent considérer les attentes de leurs usagers. En effet, la satisfaction de ces derniers prime sur tout autre objectif puisque constituant un grand attrait et permettant donc de mettre en place des systèmes déployés à grande échelle. La Figure III.1 montre l'ensemble des critères susceptibles d'être considérés dans le cadre d'une optimisation du traitement des requêtes des usagers. Ceci revient à optimiser (i.e. minimiser ou maximiser) une fonction objectif globale

qui reprend les critères spécifiés par les covoitureurs ou par l'utilisateur principal du système et ainsi répondre totalement ou partiellement aux attentes et exigences des utilisateurs du service.



Figure III.1. Critères identifiés pour l'optimisation des services de covoiturage

L'intégration de l'optimisation dans les traitements à effectuer devient, par conséquent, essentielle pour maintenir le système à l'échelle de la compétitivité. Par ailleurs, notre objectif premier étant de cadrer avec une optimisation efficace et absolue, la possibilité de considérer plusieurs critères en même temps devient évidente, voire même essentielle. En effet, deux critères ou plus peuvent être considérés conjointement dans le cadre d'une optimisation multiobjectif. Les systèmes pourvus de tels moyens fournissent dans la majorité des cas des réponses optimisées, si ce n'est optimales, permettant ainsi d'extraire à partir d'un espace de recherche étendu et infini les solutions qui optimisent une combinaison de critères parfois contradictoires. Un compromis est dans ce cas recherché en fixant une fonction objectif en adéquation avec les exigences des utilisateurs, permettant ainsi d'agrèger ou de classer l'ensemble des critères suivant un ordre de préférences (généralement précisé par l'utilisateur lui-même). Dans ce contexte, la « *multi-objectivité* » nous permet de répondre le plus efficacement possible au besoin en matière d'optimalité par rapport aux résultats fournis de sorte à en faire des réponses personnalisées pour chaque individu.

Ceci étant, considérer des méthodes d'optimisation dans les systèmes de recherche opérationnelle peut induire à une complexité exponentielle. C'est le cas pour le problème du covoiturage dynamique optimisé tel que ça a été démontré dans le chapitre précédent (c.f. §II.3.2.). Il est donc essentiel de considérer des stratégies de résolution efficaces afin de contrecarrer l'explosion combinatoire notamment dans le cadre d'un système dynamique optimisé multicritère.

### III.2.2. Approche proposée : Critères d'Optimisation

L'optimisation étant un concept aujourd'hui très convoité, essentiellement dans les systèmes dynamiques, et à dessein de réaliser cet objectif premier que nous nous sommes fixés, nous aspirons à la mise en place d'un système de covoiturage satisfaisant et avantageux. Il est de ce fait nécessaire de considérer un volet optimisation lors de la génération des « réponses solutions » aux requêtes des usagers afin d'en garantir la qualité et l'intégrité. Dans nos travaux, nous implémentons deux approches pour le traitement optimisé des requêtes des utilisateurs, entre autre une heuristique spécifique considérant l'optimisation du confort, en termes de nombres de changements de voitures. Pour l'optimisation des solutions ainsi générées mais aussi dans la première approche proposée (c.f. §IV.7), nous nous focalisons principalement sur le critère de *durée* du trajet. En effet, les usagers du service de covoiturage, qu'ils soient passagers ou conducteurs, préfèrent généralement atteindre leurs destinations finales le plus vite possible. Etant donné que plusieurs véhicules peuvent offrir un même trajet correspondant à la demande d'un ou plusieurs covoiturés, l'ensemble de ces voitures représente des solutions potentielles dites faisables puisque répondant aux contraintes de correspondance. Le choix final du véhicule à affecter porte sur la solution qui optimise au maximum le critère fixé, à savoir la durée globale du trajet, tout en évitant d'engendrer un contexte de conflits pouvant toucher le système.

Les contraintes relatives à la correspondance des offres et demandes considérées par rapport au trajet, la date, l'heure, etc. sont détaillées dans la section suivante avec leur formulation mathématique. Compte tenu de l'ensemble des solutions ainsi extraites, celle qui sera finalement affectée à l'utilisateur est donc celle qui concorde avec ses exigences tout en lui permettant de parvenir au lieu souhaité en un minimum de temps. Toujours est-il que l'optimalité calculée en ce sens reste relative aux préférences de l'utilisateur. En effet, afin de déterminer la durée globale du voyage, nous considérons tous les critères pouvant l'impacter, à savoir le temps d'attente et le temps de parcours. Une fonction objectif globale minimisant la durée totale du voyage et agréant ces deux critères en fonction des préférences des utilisateurs est ainsi adoptée pour déterminer le choix final d'optimisation de la solution.

Dans la section qui suit est fourni le détail de toutes ces notions avec le formalisme correspondant pour une modélisation conforme à la nature du problème permettant de l'approcher de manière à en réduire la complexité combinatoire.

### **III.3. Une modélisation graphique innovante pour la mise en place d'une architecture physique distribuée**

Dans le but essentiel de contrecarrer les problèmes de complexité exponentielle, nous nous proposons de mettre en place un système de résolution qui se base sur la décomposition du processus global en un ensemble de tâches moins complexes s'exécutant en parallèle. Cette idée de base se traduit par la mise en place d'une architecture logicielle distribuée où le problème initial est décomposé en plusieurs sous-problèmes auxquels sont associés des processus plus ou moins indépendants. Dans un contexte de « *multi-threading* », ces processus exécutent en parallèle et de manière autonome une ou plusieurs requêtes ou sous-requêtes favorisant ainsi le traitement instantané et parallèle des demandes des usagers du service. Ces processus parallèles distribués sont construits à la suite d'un procédé de décentralisation du problème initial.

Afin de concrétiser cette idée et mettre en place une telle architecture logicielle, nous avons opté pour la construction d'une architecture *physique* distribuée représentant le plus fidèlement possible l'aire géographique de desserte. Cette architecture qui est à l'origine de la décentralisation du traitement est conforme au modèle de graphe dynamique distribué. Nous nous sommes inspirés à cet effet des travaux réalisés par (18), (69) et (2) qui se sont concertés sur l'adoption d'un tel modèle pour la modélisation des réseaux de transport, à la base complexes, de sorte à en faire des problèmes faciles d'approche et des espaces de recherche d'accès aisé. Ce modèle s'est en effet avéré très utile pour mener des traitements à priori compliqués de recherche d'itinéraires par exemple, tel que c'est le cas pour les deux premiers travaux cités (18) et (69). La conformité aux aspects dynamiques de l'espace de recherche à représenter en plus de la simplification dont il s'imprègne, pour la modélisation de l'étendue et la complexité d'un tel espace, en font un modèle intéressant apportant des contributions majeures.

Dans cette première partie de nos travaux, nous visons la construction d'un modèle représentatif de « *l'architecture physique* » qui représenterait au mieux le champ applicatif du système. Ce modèle devrait par ailleurs délimiter le périmètre d'exécution du système pour ne traiter que l'information existante sur les passagers et conducteurs (respectivement covoiturés et covoitureurs) avec de fortes probabilités d'extension dépendant notamment des données

nouvelles. Les usagers du service de covoiturage dynamique sont ainsi représentés avec leurs itinéraires respectifs (i.e. correspondants à leurs offres et demandes) selon une modélisation graphique distribuée conforme à leur répartition de par le réseau géographique desservi. Un graphe dynamique distribué reproduisant intégralement l'image du réseau avec ses formes dynamiques et réparties dans un ordre et aspect inconstants se trouve être bien approprié et adapté à ce contexte. L'architecture logicielle établie par la suite est à l'image de cette modélisation du terrain puisque mettant en exergue les ressemblances entre les offres et les demandes entre elles pour en assurer l'optimalité des traitements et ainsi la recherche des correspondances pour l'affectation des véhicules aux usagers.

La modélisation du système de covoiturage dynamique proposée est établie dynamiquement de sorte à représenter le plus fidèlement possible le réseau géographique desservi à l'instant  $t$  considéré. Le périmètre de la zone représentée est ainsi restreint à l'aire contenant l'information sur les usagers mais tout en n'étant pas passible d'expansion puisque devant suivre l'évolution des offres et demandes dans le temps et ainsi inclure leurs spécificités en temps réel.

Dans ce qui suit, la modélisation formelle du service de covoiturage dynamique proposé. Un formalisme adapté pour répondre efficacement aux besoins de flexibilité et variabilité des informations à modéliser.

### III.3.1. Une modélisation sous forme de graphe dynamique distribué pour la mise en place d'une architecture physique évolutive

Pour la modélisation de notre système, nous nous sommes inspirés des efforts développés dans le cadre des problèmes de tournée de véhicules (109) et plus particulièrement les problèmes de collecte et de distribution en temps réel (110), (79) compte tenu des ressemblances qu'ils présentent avec le problème de covoiturage dynamique. En effet, les aspects stochastiques et dynamiques (111) des événements et données (112) définissant ces problèmes rejoignent l'aspect aléatoire des offres et demandes du problème de covoiturage en temps réel. Le problème de transport à la demande de personnes handicapées (Dial-A-Ride Problem : DARP) (113), (114), qui consiste à adapter l'offre en fonction de la demande des usagers, est ici considéré en particulier puisqu'il se rapproche encore plus de notre problème,

étant donné que nous nous plaçons dans le cadre du transport de personnes et non pas de marchandises.

### *III.3.1.1. Modélisation des données du système*

La problématique du Covoiturage Dynamique Optimisé (CDO) peut être assimilée à un problème de transport public de personnes, notamment en zone urbaine et pré-urbaine. Les efforts déployés dans le sens d'intégration des données y transitant ont abouti à la mise en place d'une architecture adéquate et ont ainsi donné lieu à une représentation innovante du réseau de desserte sous forme de graphe dynamique distribué. Vers une démarche simplifiée de traitements complexes, les graphes dynamiques sont, dans notre cas, le fondement de notre approche. Cette modélisation ayant fait l'objet de plusieurs travaux dans plusieurs domaines, notamment dans le domaine du transport n'a jamais été abordée dans les systèmes de covoiturage et encore moins dans les systèmes de covoiturage dynamique. Par ailleurs, la dynamique des modélisations basées sur ce concept se réfère essentiellement à la pondération variable des arcs du graphe. Dans notre cas, tous les aspects à représenter sont touchés par le problème de la variation continue et ce concept a ainsi été adapté au problème de covoiturage dynamique optimisé. Une démarche innovante inspirée de la notion de classification des données a ainsi aidé à l'établissement d'un modèle bien adapté à la problématique que nous nous proposons de résoudre. Le travail élaboré dans ce cadre a donné lieu à la mise en place de l'architecture graphique escomptée, détaillée et formalisée dans les sections suivantes.

Basée sur le concept de graphe dynamique distribué, l'architecture en question tient sa propriété de dynamique des spécificités des offres et demandes des usagers qu'elle représente. Ces dernières étant fortement variables, le modèle graphique représentatif du système doit en suivre l'évolution à travers le temps. Ainsi, pour  $n$  usagers émettant des requêtes de covoiturage et  $m$  véhicules offrant des trajets, qu'ils soient déjà en cours de circulation ou pas encore à un instant  $t$  donné, le système de covoiturage dynamique est défini comme étant le triplet  $T(t) = (G(t), R(t), O(t))$ . Dans cette formulation, chacun des paramètres est relatif à une composante particulière qui contribue à la bonne mise en œuvre du système :

- $G(t)$  est un graphe modélisant le réseau dynamique de covoiturage. Il s'agit d'une représentation que nous avons jugée idéale puisqu'elle épouse parfaitement la forme et l'image du terrain accueillant les covoitureurs et covoiturés ;
- $R(t)$  est l'ensemble des demandes de covoiturage à traiter à l'instant  $t$  considéré ;

- $O(t)$  est l'ensemble des offres relatives aux voitures disponibles.

Dans la suite, l'ensemble de ces composantes est décrit avec le niveau de détail requis et ce afin d'en expliquer le choix de modélisation adopté.

#### *A. Une modélisation sous forme de graphe pour illustrer l'image du réseau*

$G(t) = (N(t), A(t))$  est un graphe orienté dynamique représentant l'image du réseau de desserte où sont dispersées l'ensemble des offres et demandes des usagers du service. En effet, les principaux paramètres du graphe étant  $N(t)$  et  $A(t)$ , ces derniers sont définis dans la suite de cette section.

##### *A.1. $N(t)$ : l'ensemble des nœuds du graphe*

$N(t)$  est l'ensemble des nœuds construits, à l'instant  $t$ , sur la base des positions géographiques des usagers du service, qu'ils soient covoitureurs ou covoiturés ainsi que sur les détails des déplacements requis ou prévus (i.e. repères d'itinéraires et chemins à parcourir). Ainsi, l'ensemble des nœuds, défini plus en détail dans la suite, est identifié comme étant l'ensemble des origines des trajets demandés ou à desservir, leurs destinations finales requises ainsi que les adresses intermédiaires par lesquelles sont susceptibles de passer les usagers en question. A défaut de paramétrage des départs de voyages, l'ensemble des origines des déplacements s'identifie aux coordonnées géographiques actuelles des utilisateurs (i.e. piétons et voitures). En effet, notre système étant doté d'un module de positionnement géographique, la géolocalisation des usagers se fait en temps réel et en continu de sorte à en assurer l'acquisition facile à n'importe quel moment pour des besoins de traitement mais aussi de traçabilité. Ceci étant un facteur principal dans la mise en place d'un système sûr et efficace garantissant la sécurité des usagers.

##### *A.2. $A(t)$ : l'ensemble des arcs du graphe*

Correspondant à l'ensemble des arcs du graphe,  $A(t)$  représente essentiellement les itinéraires qui vont être desservis par les voitures. Ces arcs s'identifient, de ce fait, aux chemins restants à parcourir par les conducteurs en cours de voyage mais aussi ceux qui n'ont pas encore été entamés (i.e. voitures immobiles). Ces itinéraires sont construits et mis à jour en temps réel en fonction des points de passage des véhicules tracés par le système. Dans le cadre de prise en compte d'une image actualisée, fidèle à l'état réel du réseau, ces itinéraires déterminent ainsi les routes à emprunter afin de parvenir aux destinations finales préalablement spécifiées par les conducteurs. Si l'on se projette sur une carte routière réelle,

la représentation de ces arcs doit être conforme à la nature des routes ainsi qu'aux possibilités de desserte sur celles-ci (e.g. route en sens unique, route à double sens, etc.). Toujours est-il que le sens de chaque route n'est visible sur le graphe que lorsque celle-ci est empruntée par une ou plusieurs voitures dans la direction indiquée, conformément à son itinéraire global. Ainsi, comme le montre la Figure III.2, un arc peut être orienté soit dans un seul sens soit dans les deux si deux voitures ou plus comptent desservir la même section de route dans deux directions opposées.

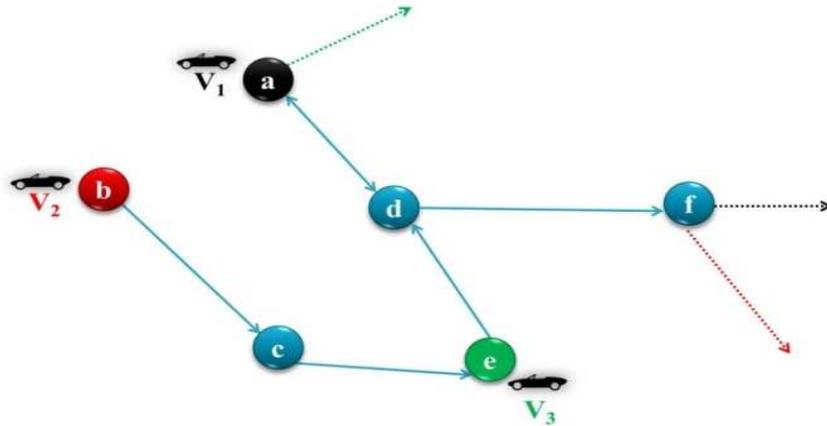


Figure III.2. Représentation des itinéraires des voitures

Les routes non desservies, c'est-à-dire ne correspondant pas à des trajets ou des parties de trajets de véhicules, ne sont pas représentées au niveau de cette modélisation. En effet, le processus d'affectation est effectué uniquement sur la base de l'offre existante pour rechercher les itinéraires répondant aux requêtes des covoiturés. Les arcs représentés au niveau de ce graphe sont ainsi formés à la suite de l'extraction de l'information nécessaire sur l'offre disponible à l'instant  $t$  considéré. Il s'agit en effet d'une modélisation des itinéraires reliant les origines des déplacements des voitures à leurs destinations finales en passant par des points repères. Ces derniers correspondent à des adresses de passage désignés par les conducteurs eux-mêmes ou calculés par le système lors d'un processus préalable comme étant des points de prise et/ou dépose de passager(s). En effet, lors de traitements antérieurs, les voitures peuvent se voir affecter des passagers qu'il faudra récupérer et déposer à des endroits spécifiques pouvant engendrer de petits détours dans leurs itinéraires initiaux. Par conséquent, ces points de prise et dépose servent à actualiser les chemins à parcourir par les véhicules.

Les nœuds origines des déplacements modélisés dans ce graphe représentent soit les positions actuelles des voitures en question si celles-ci sont déjà en cours de circulation, soit leurs départs initiaux s'il s'agit de voitures dont le voyage n'a pas encore été entamé. Une

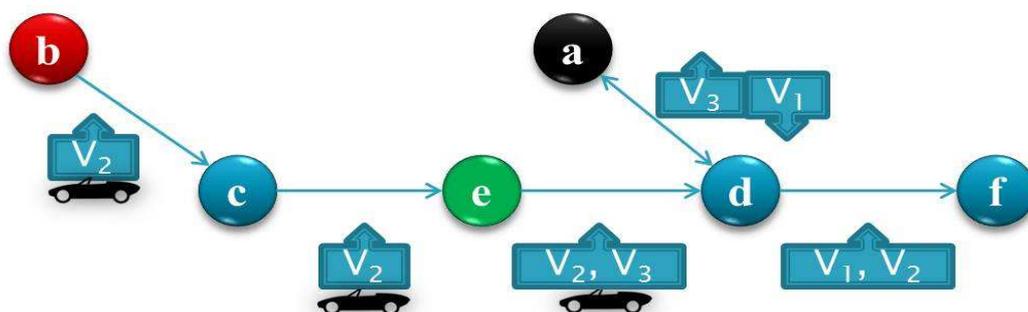
géolocalisation instantanée permet ainsi de déterminer les positions courantes des voitures et de tracer leurs itinéraires restants pour atteindre leur objectif.

Le Tableau III.1 montre un récapitulatif des itinéraires de chacun des véhicules  $V_1$ ,  $V_2$  et  $V_3$  représentés sur le graphe décrit au niveau de la Figure III.2. Ces itinéraires sont ici décrits sous forme d'une succession de points de passage de chacun de ces véhicules.

*Tableau III.1. Itinéraires des Véhicules*

<i>Véhicule</i>	<i>Points de Passage</i>
$V_1$	$a \rightarrow d \rightarrow f \rightarrow \dots$
$V_2$	$b \rightarrow c \rightarrow e \rightarrow d \rightarrow f \rightarrow \dots$
$V_3$	$e \rightarrow d \rightarrow a \rightarrow \dots$

Les itinéraires des voitures tels que représentés sur la Figure III.2 sont composés de plusieurs sections de routes. Deux voitures ou plus peuvent en partager plus d'une dans le même sens (i.e. cas du tronçon  $[d \rightarrow f]$ ) ou dans un sens contraire (i.e. cas du tronçon  $[a \rightarrow d]$ ) comme le montre la Figure III.3. Ainsi définies, ces sections de routes constituent les arcs du graphe  $G(t)$  reliant les nœuds entre eux, identifiant ainsi leur sens d'orientation en fonction des directions selon lesquelles les routes représentées sont empruntées. Les nœuds extrémités de ces parcelles de trajets peuvent donc jouer le rôle d'origine ou destination ou bien les deux à la fois.



*Figure III.3. Partage d'itinéraires : une ou plusieurs voitures par tronçon*

Partant de ce principe et sachant que les voitures peuvent se partager des parties de leurs trajets ou simplement des points d'intersection, nous identifions deux catégories distinctes de nœuds : origines (i.e. nœuds sortie) et destinations (i.e. nœuds réception). Ces deux classes de

nœuds, telles qu'elles sont illustrées sur la Figure III.4, peuvent se croiser en un ensemble de points communs (origine/destination). Par conséquent, chaque nœud peut posséder :

- Un ou plusieurs successeurs se référant à l'ensemble des nœuds destinations (extrémités finales) des sections de routes dont le nœud en question est l'origine (extrémité initiale). Les nœuds successeurs représentent ainsi des destinations finales ou intermédiaires de véhicule(s) ou usager(s).
- Un ou plusieurs prédécesseurs qui sont les extrémités initiales (i.e. origines) des arcs dont le nœud considéré est la destination.

Un nœud peut lui-même être successeur ou prédécesseur ou les deux à la fois d'un ou plusieurs autres nœuds. On distingue ainsi pour chaque nœud  $x \in N(t)$  deux ensembles :

1. L'ensemble des nœuds successeurs de  $x$  noté  $N_x^+(t)$  et défini comme suit :

$$N_x^+(t) = \{v \in N(t) / x \neq v \text{ et } (x, v) \in A(t)\} \quad (\text{III.1})$$

2. L'ensemble des nœuds prédécesseurs de  $x$  noté  $N_x^-(t)$  et défini comme suit :

$$N_x^-(t) = \{v \in N(t) / x \neq v \text{ et } (v, x) \in A(t)\} \quad (\text{III.2})$$

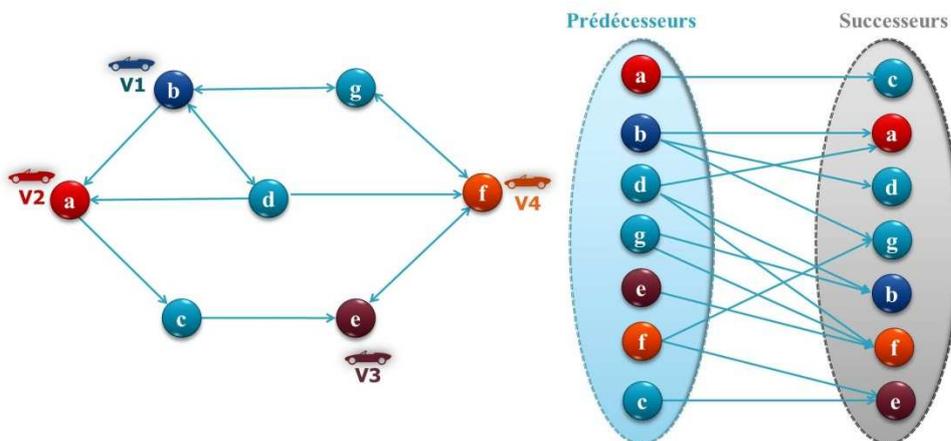


Figure III.4. Ensembles de nœuds successeurs et prédécesseurs

### A.3. Un graphe dynamique pour un réseau dynamique

Le graphe  $G(t)$  dont les éléments  $N(t)$  et  $A(t)$  dépendent fortement des données du système, à savoir l'information sur l'offre et la demande, dépend entièrement de ces données et possède une forme et des propriétés qui varient constamment. En effet, les données du système étant en continuelle variation, leur modélisation ne peut être constante. Ainsi, la représentation graphique proposée suit le mouvement assurant ainsi une image fidèle à

l'apparence du réseau variant avec les informations acquises à chaque instant  $t$  considéré. La modélisation proposée est ainsi établie en fonction du temps et est représentée sous forme de graphe dynamique dont la dynamicité d'au moins un de ses paramètres en fait la définition (69). En effet, les caractéristiques de ce graphe ne sont pas fixes et s'expriment plutôt en fonction des besoins des utilisateurs en matière de déplacement et des services fournis. De ce fait, ces caractéristiques varient constamment montrant l'évolution des trajets à travers le temps ainsi que les nouveaux changements opérés sur le réseau (i.e. nouvelles offres et demandes).

### ***B. Recueil des requêtes de covoiturage et modélisation des données***

$R(t) = \bigcup_{p=1}^n \{R_p(t)\}$  : est l'ensemble des  $n$  requêtes émises, à l'instant  $t$ , par les  $n$  usagers du service :  $U(t) = \bigcup_{p=1}^n \{U_p(t)\}$ . Seuls les utilisateurs connectés au système à l'instant  $t$  et ayant émis des requêtes de covoiturage sont pris en considération. Sachant que bien d'autres usagers abonnés du service peuvent exister (i.e. connectés ou non), ces derniers ne sont pris en compte que dès lors qu'une requête qui leur est associée parvient au système. En effet, l'ensemble de ces abonnés ne sont pas considérés lors de la construction du graphe puisqu'ils ne font pas partie intégrante du fonctionnement du service. Ils sont, de ce fait, dits « *passifs* » étant donné qu'ils ne contribuent à la réalisation des traitements et ne deviennent donc « *actifs* » que sous l'unique condition d'avoir émis préalablement des demandes de déplacement. Par ailleurs, ces utilisateurs peuvent être à l'origine d'une mise à jour du graphe s'ils envoient des requêtes alors que le processus est déjà en cours d'exécution (i.e. à un instant  $t'$  proche de  $t$  avec  $t' > t$ ). En effet, la modélisation de la zone de desserte doit représenter le plus fidèlement possible le réseau de covoiturage, prenant ainsi une forme adéquate et peut être différente d'un instant à l'autre afin de modéliser en temps réel tout changement pouvant survenir.

#### ***B.1. Structure d'une requête de covoiturage***

A dessein de mettre en place un système optimisé qui requière en premier lieu le paramétrage de l'utilisateur, nous nous proposons de structurer les requêtes des usagers de manière à ce qu'elles puissent incorporer leurs choix et préférences. A chaque utilisateur représenté dans la modélisation établie est ainsi associée une demande de covoiturage dont le formalisme est le suivant :

$$R_p(t) = \{R_p^+, R_p^-, P_p, [d_p, a_p]\} \quad (\text{III.3})$$

Chaque demande est spécifique à un utilisateur bien déterminé  $U_p$  indiquant ses préférences de déplacement, à savoir :

- o  $R_p^+$  : désigne les coordonnées géographiques de l'origine souhaitée de son déplacement. Le départ de chaque voyage peut être indiqué par l'utilisateur en spécifiant soit l'adresse soit directement les coordonnées géographiques qui lui correspondent. Si l'utilisateur omet de spécifier son lieu de départ, sa position géographique à l'instant  $t$  considéré est automatiquement détectée via l'outil GPS intégré dans notre système. La position ainsi déterminée sera considérée comme étant l'origine initiale du trajet à effectuer par  $U_p$ .
- o  $R_p^-$  : correspond à la destination finale que l'utilisateur  $p$  voudrait atteindre. Indiquée de manière similaire au départ du trajet planifié, la destination peut être insérée sous forme d'adresse ou bien de coordonnées géographiques. Les origines et destinations des trajets demandés par les utilisateurs sont représentés au niveau de la modélisation proposée par des couples de nœuds du graphe  $G(t)$ . Par conséquent,  $N_R(t)$  est l'ensemble des nœuds dans  $N(t)$  relatifs aux requêtes émises par les covoiturés (i.e. piétons, passagers) :

$$N_R(t) = \bigcup_{p=1}^n \{R_p^+, R_p^-\} \subseteq N(t) \quad (\text{III.4})$$

- o  $P_p$  : correspond au nombre de personnes,  $U_p$  inclus, à transporter de l'origine  $R_p^+$  à la destination  $R_p^-$  ( $P_p \geq 1$ ). Par défaut, ce paramètre vaut 1 considérant  $U_p$  comme l'unique passager. Cependant, l'utilisateur, émetteur de la demande, peut spécifier que le voyage concerne plusieurs individus (e.g. collègues de travail ou étudiants souhaitant voyager sur un même trajet...). L'utilisation de ce paramètre nous permet d'optimiser le traitement des requêtes. En effet, en évitant à plusieurs usagers d'envoyer séparément leurs requêtes, l'émission d'une seule demande étant suffisante, le traitement parallèle de l'ensemble de leurs demandes est ainsi favorisé pour n'en traiter qu'une seule. Faisant usage de ce paramètre, nous visons le traitement parallèle des requêtes similaires aussi bien que l'optimisation des flux d'émission et de réception des requêtes et des réponses.
- o  $[d_p, a_p]$  : désigne un intervalle de temps limité qui définit la durée globale du trajet

maximale tolérée par l'utilisateur. Ici nous nous sommes inspirés des travaux développés dans le cadre des problèmes de collecte et distribution de charges (115). Plus particulièrement encore, sont considérés ceux relatifs aux « *Dial-A-Ride problems* » qui ont eu recours au concept de fenêtre de temps. Dans ce sens, nous faisons plutôt référence à la variante de « *Dial-a-Ride problem with desired pickup and delivery times* » (116) pour ne considérer que la satisfaction des abonnés du service. Dans nos travaux,  $d_p$  désigne de ce fait le temps de départ au plus tôt possible pour l'utilisateur  $p$  et  $a_p$  le temps d'arrivée au plus tard toléré. Cet intervalle, lorsqu'il est spécifié par l'utilisateur, indique l'heure au plutôt à partir de laquelle ce dernier pourrait se présenter au point de départ spécifié ainsi que l'heure d'arrivée à ne pas dépasser indiquant implicitement la marge de retard toléré durant le trajet à effectuer. Dans le cas contraire, à défaut d'avoir spécifié ses préférences, l'usager se verra affecter des valeurs par défaut. Ces valeurs correspondent au temps système relatif à l'heure d'émission de la requête pour le temps de départ au plus tôt et au Temps d'Arrivée Estimé (TAE) pour le temps d'arrivée au plus tard :

$$TAE(R_p^+, R_p^-, t) = \rho_{R_p^+, R_p^-}(t) + \tau(R_p^+, R_p^-) \quad (\text{III.5})$$

calculé dans ce cas sur la base de deux fonctions :

1. Une fonction de pondération calculant le temps de parcours ( $\rho_{O,D}(t)$ ) requis pour traverser une route bien déterminée à partir d'une origine  $O$  vers une destination  $D$  à l'instant  $t$  considéré. Dans ce cas, le graphe  $G(t)$  est dit « valué » permettant d'attribuer une valeur (i.e. un poids) à chaque arc du graphe. Le poids d'un arc  $(O, D)$  tel que  $(O, D) \in A(t)$  vaut ( $\rho_{O,D}(t)$ ) calculé en fonction de plusieurs facteurs :

$$\rho_{O,D}(t) = \text{Distance}(O, D) * \frac{1}{VMP(O, D, t)} \text{ avec } (O, D) \in A(t) \quad (\text{III.6})$$

Notons que dans un contexte de transport urbain et pré-urbain, les pondérations des arcs ne peuvent être négatives. Ainsi,  $\forall (O, D) \in A(t), \rho_{O,D}(t) \geq 0$  (2). Les poids des arcs du graphe sont calculés en fonction de :

- La fonction  $\text{Distance}(O, D)$  qui calcule la distance entre deux points géographiques représentés par leurs coordonnées GPS (i.e. Longitude, Latitude).

Dans cette formule, la loi sphérique des cosinus est prise en compte et se présente comme suit :

$$Distance(O, D) = \text{acos} \left( \sin(\text{latitude}_O) * \sin(\text{latitude}_D) + \cos(\text{latitude}_O) * \cos(\text{longitude}_D - \text{longitude}_O) \right) * R \quad (\text{III.7})$$

Cette formule indique avec une précision quasi-totale la distance séparant deux points, même proches à moins d'un mètre de distance sur la surface de la terre<sup>32</sup> dont R est ici le rayon. Les paramètres *latitude* et *longitude* nécessaires au calcul de cette formule sont relatifs aux coordonnées GPS d'un point de l'espace et doivent être spécifiés en radian.

- Le paramètre *VMP* qui indique la Vitesse Moyenne de Parcours sur une route spécifique à un instant *t* bien déterminé. Ce paramètre varie en fonction de plusieurs facteurs, entre autre le type de la route qu'il s'agisse d'une autoroute, route secondaire ou autre. Ainsi, la vitesse moyenne de parcours est différente d'une route à une autre. Par ailleurs, pour un même type de route, ce paramètre peut varier puisque la vitesse à laquelle roule une voiture sur une même route n'est pas la même en temps pluvieux qu'en temps neigeux ou en heure de pointe qu'en période normale, etc. Par la suite, et comme le montre le Tableau III.2, ce paramètre dépend particulièrement de la nature de la route, la période de voyage et le temps.

*Tableau III.2. Vitesse Moyenne de Parcours (VMP) à l'origine d'une pondération dynamique*

<i>Période de Voyage</i>	<i>Temps</i>	<i>Type De Route</i>	<i>Vitesse Moyenne de Parcours (VMP)</i>
<i>Période Normale</i>	<i>Pluie</i>	<i>Autoroute</i>	<i>90 km/h</i>
<i>Période Normale</i>	<i>Pluie</i>	<i>Route Secondaire</i>	<i>70 km/h</i>
<i>Heure de Pointe</i>	<i>Pluie</i>	<i>Autoroute</i>	<i>60 km/h</i>
<i>Heure de Pointe</i>	<i>Pluie</i>	<i>Route Secondaire</i>	<i>40 km/h</i>

<sup>32</sup> <http://www.movable-type.co.uk/scripts/latlong.html>

<i>Période Normale</i>	<i>Normal</i>	<i>Autoroute</i>	<i>110 km/h</i>
<i>Période Normale</i>	<i>Normal</i>	<i>Route Secondaire</i>	<i>90 km/h</i>
...	...	...	...

Partant de ce principe et sur la base de la variabilité des *VMP*, nous pouvons constater que les poids affectés à une même route peuvent différer même si l'image du réseau reste figée entre deux instants différents. Ces poids dépendent du temps comme facteur principal et peuvent être à l'origine de l'inconstance des traitements et calculs. Ainsi, en plus de sa forme dynamique basée sur des paramètres variables (i.e.  $N(t)$  et  $A(t)$ ), le graphe  $G(t)$  est aussi doté d'une fonction de pondération dynamique. Un graphe est énoncé comme étant dynamique si au moins l'un de ses paramètres est fonction du temps (69) (18).  $G(t)$  est de ce fait défini comme étant un graphe « *fortement dynamique* » puisque dépourvu de toute caractéristique invariante.

2. Une fonction  $\tau(O, D)$  qui calcule le retard maximal possible sur une route  $[O \rightarrow D]$ .  $[O \rightarrow D]$  n'est considéré segment de route que s'il existe un arc  $(O, D) \in A(t)$  reliant ses deux extrémités dans le sens indiqué. Le taux de retard tolérable est calculé en fonction de la distance séparant les nœuds origine et destination de la route considérée proportionnellement à un Seuil de Retard Tolérable (*SRT*) paramétrable par l'administrateur du système :

$$\tau(O, D) = \text{Distance}(O, D) * \frac{\alpha}{\beta} \quad (\text{III.8})$$

*SRT* représente ainsi la valeur du retard toléré ( $\alpha$ ) en minutes pour un trajet de  $\beta$  kilomètres permettant de calculer l'heure d'arrivée au plus tard à l'extrémité finale de la route considérée.

### ***B.2. Une structure matricielle pour recueillir les informations sur les requêtes***

Les usagers du système spécifient leur besoin en matière de déplacement en provoquant l'émission d'un ensemble de requêtes distinctes de par leurs différents paramètres. Aussi, deux usagers ou plus peuvent émettre des requêtes qui soient susceptibles de se joindre en une

ou plusieurs routes composant leurs trajets respectifs. Cependant, les spécificités de leurs demandes respectives sur la ou les section(s) de route partagée(s) peuvent être similaires comme elles peuvent être différentes. En effet, deux usagers ou plus peuvent demander de se déplacer sur un même chemin mais dans des intervalles de temps différents, avec un nombre de compagnons différents, etc. Compte tenu de cet aspect de multiplicité des demandes sur une même route, une représentation adéquate est proposée permettant de favoriser le traitement parallèle de celles-ci, et ce en faveur de l'optimisation des propriétés d'exécution (i.e. allocation mémoire, temps de traitement, utilisation CPU, ...). A cet effet, nous nous proposons de réunir les requêtes des  $n$  utilisateurs en une seule et unique structure mettant en exergue les trajets convoités ainsi que les différentes demandes sur ceux-ci avec les paramètres qui leurs sont relatifs. Cette structure fait figure d'une matrice Origine à Destination où a lieu l'assemblage des requêtes de sorte à en montrer les ressemblances et ainsi mettre l'accent sur le traitement parallèle de celles-ci en vue d'en alléger le traitement.  $D(t)$  est une matrice ( $k * k$ ) qui représente les  $k$  demandes « partielles » associées aux requêtes des usagers à l'image des compositions de routes définies par rapport aux nœuds d'intersection. Les nœuds ici considérés sont essentiellement reliés aux origines et destinations spécifiés dans les requêtes initiales. Ces points définissant les indices de la matrice sont en effet extraits des détails des déplacements définis par les  $n$  utilisateurs (origines et destinations requises).

Par conséquent, pour  $n$  utilisateurs ayant émis des demandes de covoiturage,  $k \leq 2n$  et la représentation matricielle se présente comme suit :

$$D(t) = \begin{matrix} & \begin{matrix} 1 & \dots & i & \dots & j & \dots & k \end{matrix} \\ \begin{matrix} 1 \\ \vdots \\ i \\ \vdots \\ j \\ \vdots \\ k \end{matrix} & \left( \begin{array}{cccccc} \phi & & \dots & & \dots & \dots \\ & \ddots & & & & \\ \dots & & \phi & & D_{i,j} & \dots \\ & \dots & & \ddots & & \dots \\ \vdots & & D_{j,i} & & \phi & \vdots \\ \dots & \dots & \dots & \dots & \dots & \ddots \\ \dots & \dots & \dots & \dots & \dots & \phi \end{array} \right) \end{matrix}$$

Chaque élément  $D_{i,j}$  avec  $1 \leq i, j \leq k$  de la matrice  $D(t)$  est une modélisation de la demande existante sur une route bien déterminée  $[a \rightarrow b]$  partant d'une origine  $a$  (correspondant à l'indice  $i$  de la matrice) vers une destination  $b$  (correspondant à l'indice  $j$ ).

Les indices de la matrice se réfèrent ainsi à des nœuds du réseau représentés sous forme de coordonnées géographiques. Une liste de taille variable est ainsi utilisée au sein de chaque élément de la matrice représentant l'ensemble des demandes relatives à la route représentée par les indices de cet élément. Nous retrouvons ainsi, dans ces éléments, les utilisateurs ayant demandé à voyager par cette route ; c'est-à-dire les usagers dont le trajet complet  $[R_p^+ \rightarrow R_p^-]$  demandé ou une partie seulement concerne la route  $[a \rightarrow b]$ . Ces utilisateurs sont représentés avec toutes les spécificités de déplacement requises sur le tronçon en question (i.e. l'intervalle de temps préféré et le nombre de passagers).

$$D_{i,j} = \bigcup_{p=1}^n \{(U_p, P_p, [d_p, a_p])\} \text{ tel que } [a \rightarrow b] \prec [R_p^+ \rightarrow R_p^-] \quad (\text{III.9})$$

L'opérateur  $\prec$  désigne ici la notion d'appartenance d'un tronçon d'itinéraire à un chemin plus global dans le sens indiqué et en respectant l'ordre dans lequel il existe. Cet opérateur implique entre autre une relation de précédence entre les différents itinéraires partiels indiquant leur enchainement dans l'itinéraire global. Dans ce cas précis, il s'agit de considérer une demande comme étant une composition de deux autres ou plus, à l'image des correspondances entre les trajets requis. Par ailleurs, l'opérateur ici considéré implique aussi la possibilité d'égalité entre les itinéraires considérés ( $[a \rightarrow b]$  et  $[R_p^+ \rightarrow R_p^-]$ ).

### C. Des offres de particuliers pour satisfaire les demandes des covoiturés

$O(t) = \bigcup_{j=1}^m \{O_j(t)\}$  est l'ensemble des offres des  $m$  véhicules proposant des trajets à l'instant  $t$ .  $V(t) = \bigcup_{j=1}^m \{V_j\}$  est la flotte de véhicules ayant proposé une offre relative à un voyage bien déterminé. A l'instant  $t$  considéré, les offres disponibles sont relatives à des voyages déjà en cours ou qui vont prochainement commencer. A tout trajet de covoiturage offert spécifié correspond un couple de nœuds dans  $N(t)$  représentant ses points de départ (i.e. origine ( $O_j^+$ )) et d'arrivée (i.e. destination ( $O_j^-$ )) :

$$N_o(t) = \bigcup_{j=1}^m \{O_j^+, O_j^-\} \subseteq N(t) \quad (\text{III.10})$$

$N_o(t)$  désigne ainsi l'ensemble des nœuds origines et destinations indiqués par les membres conducteurs de véhicules (i.e. covoitureurs) soumettant des offres de covoiturage à l'instant  $t$ .

### ***C.1. Structure d'une offre de covoiturage***

Les offres de véhicules sont modélisées par notre système de manière pratiquement similaire aux requêtes des covoiturés décrites précédemment au niveau de l'équation (III.3) et se présentent donc comme suit :

$$O_j(t) = \left\{ O_j^+, O_j^-, l_j, \begin{pmatrix} ID_1 \\ \dots \\ ID_z \end{pmatrix}_j, [d_j, a_j] \right\} \quad (\text{III.11})$$

$O_j(t)$  : l'offre de trajet relative à la voiture  $V_j$  ( $1 \leq j \leq m$ ) est ainsi dotée des propriétés requises pour en optimiser le traitement et permettre la vérification des contraintes de correspondance dans le but de réaliser une affectation valable et efficace. Ainsi, pour soumettre une offre, tout conducteur doit spécifier :

- o Le point de départ de son trajet ( $O_j^+$ ) qui pareil au paramètre origine des demandes peut être spécifié par l'utilisateur lui-même ou correspondre à sa position géographique actuelle. Dans le cas des véhicules, la fonction de géolocalisation est particulièrement utile, puisque dans un contexte de covoiturage en temps réel, il est essentiel d'acquérir l'information actualisée des positions des voitures en cours de circulation. Ces voitures étant susceptibles de répondre à une ou plusieurs des demandes à traiter. La connaissance globale sur l'ensemble des véhicules avec leurs positions courantes est ainsi requise pour pouvoir rechercher les similitudes avec les trajets demandés tout en étant sûrs de manipuler une information réelle et pertinente (continuellement mise à jour) et ce en faveur de la validité et la faisabilité des réponses fournies.
- o Le point d'arrivée ( $O_j^-$ ) est spécifié par tout conducteur soumettant une offre de covoiturage dès lors qu'il se connecte et en insère les spécificités. Cette même information sera valable tout le long de son trajet et tant qu'il n'a pas encore atteint sa destination finale ( $O_j^-$ ). En effet, si un processus est créé pour traiter une ou plusieurs requêtes survenues à l'instant  $t$  considéré, ce paramètre ( $O_j^-$ ) reste inchangé

même si la voiture  $V_j$  est déjà en cours de circulation sous réserve d'absence d'annulation ou de modification de l'offre par le conducteur concerné.

- o Le nombre de places  $l_j$  disponibles dans le véhicule et qui sont susceptibles d'être affectées à des passagers voulant parcourir le même trajet emprunté par la voiture  $V_j$  ou une partie de celui-ci. Chaque véhicule  $V_j, \forall j \in \{1..m\}$ , a une capacité limitée  $C_j \in \mathbb{N}^{*+}$  qui peut être différente d'une voiture à une autre et sert à calculer le nombre maximum de passagers pouvant être transportés à bord de  $V_j$  (i.e. le nombre de places disponibles  $l_j < C_j$  est calculé en fonction de celles déjà occupées et de la capacité du véhicule). Ce nombre est automatiquement mis à jour lors de l'affectation d'un ou plusieurs passagers au véhicule sur le(s) tronçon(s) concerné(s). Le système mis en place devant être pourvu des fonctionnalités de mise à jour automatique, les données sur les véhicules sont ainsi actualisées lors de chaque prise et dépose de passagers afin d'éviter les situations de conflit. De plus, l'information sur les détails d'affectation est sauvegardée (e.g. nombre de passagers à bord pour chaque trajet ou parcelle de trajet si une décomposition de celui-ci en fonction de la demande a eu lieu, etc.) et ce notamment afin d'optimiser les traitements ultérieurs.
- o  $[d_j, a_j]$  définit les préférences de déplacement des conducteurs de voitures. Cet intervalle désigne les mêmes paramètres que ceux définis au niveau des demandes des passagers ; à savoir le temps de départ au plus tôt et le temps d'arrivée au plus tard de la voiture. A défaut de spécification par l'utilisateur lui-même, les deux extrémités de cet intervalle valent respectivement l'heure actuelle affichée sur le serveur de traitement et le temps d'arrivée estimé impliquant par ailleurs une estimation du retard possible sur le trajet considéré. Il est vrai que nous nous focalisons particulièrement sur le voyageur en tant que passager pour en optimiser le traitement des requêtes. Cependant, la satisfaction des conducteurs de véhicules nous importe autant. En effet, contrairement aux « Dial-A-Ride problems », où un ou plusieurs véhicules libres sont mis à la disposition des usagers et avec pour seule contrainte un dépôt de départ et un d'arrivée (114), les véhicules dans notre système sont les propriétés de particuliers dont l'offre dépend entièrement. Ainsi, outre le fait de rechercher les correspondances des trajets pour pouvoir en réaliser l'affectation, il convient de considérer les préférences de déplacement des conducteurs. Ces derniers

n'étant pas des employés libres du service ni des conducteurs de taxis collectifs (117), ils ont des besoins en matière de déplacement qu'il convient de respecter pour ainsi garantir leur satisfaction.

- o Un vecteur  $ID_j(t) = \begin{pmatrix} ID_1 \\ \vdots \\ ID_z \end{pmatrix}_j = (ID_{1,j}, \dots, ID_{q,j}, \dots, ID_{z,j})^T$  indiquant l'ensemble des

Destinations Intermédiaires (en anglais *Intermediate Destination ID*) par lesquelles devra passer le véhicule  $V_j$  et ce à partir de l'instant  $t$  considéré pour atteindre sa destination finale  $O_j^-$ . En effet, dans le but de permettre un certain degré de flexibilité à l'utilisateur du service, il est essentiel de s'enquérir de ses préférences de déplacement essentiellement par rapport aux itinéraires à emprunter. Etant donné la multitude des possibilités de choix des routes pouvant être traversées pour parvenir d'un nœud origine vers une destination, celle qui est finalement choisie pour traiter les affectations correspond aux exigences des usagers. Le système que nous proposons leur permet ainsi d'insérer un ensemble de points repères désignant les adresses par lesquelles va passer le véhicule et auxquelles peuvent se rajouter d'autres adresses (i.e. points ou nœuds de passage) relatifs à des emplacements de prise et dépose de passager(s) calculés lors d'un processus préalable. Le trajet du véhicule est ainsi tracé en temps réel, par l'outil GPS intégré au niveau de notre système, en fonction des différentes coordonnées géographiques relatives à ces adresses intermédiaires vers la destination finale. Ceci est particulièrement utile en matière de sécurité puisque ça nous permet, en plus du module de géolocalisation en temps réel, de suivre la trace de chacun des véhicules du service. Cette traçabilité est possible sous l'unique condition que les usagers donnent leur accord pour être localisés continuellement. Une clause est ainsi prévue dans la charte d'utilisation du système et qui mentionne une géolocalisation légale.

Grâce à cette propriété, le système que nous proposons est ainsi qualifié de sûr d'un côté et d'optimisé de l'autre étant donné que la prise en compte de l'ensemble des nœuds intermédiaires pour le traçage d'un chemin réellement emprunté nous évite l'explosion combinatoire engendrée par la recherche des itinéraires possibles pour chaque voiture. A défaut de spécification de ces points repères, un itinéraire par défaut est affecté au véhicule. Il s'agit du trajet construit automatiquement par les outils GPS et qui représente généralement le plus court chemin d'une origine à une

destination données. Plusieurs véhicules peuvent se voir partager des points intermédiaires ou même des sections de routes complètes. L'ensemble des points repères par lesquels les véhicules de la flotte doivent passer pour parvenir à leurs destinations finales  $N_{ID}(t)$  est défini par l'union des destinations intermédiaires de chacun :

$$N_{ID}(t) = \bigcup_{j=1}^m \bigcup_{q=1}^z ID_{q,j} \subseteq N(t) \quad (\text{III.12})$$

L'ensemble des points de passage des véhicules sont ainsi représentés comme étant des nœuds dans le graphe  $G(t)$ . Par la suite,  $N(t)$  est défini sur l'ensemble des départs, destinations et points de passage aussi bien des conducteurs que des passagers. L'équation (III.13) reprend celles qui la précèdent (i.e. III.4, III.10 et III.12) définissant  $N(t)$  :

$$N(t) = N_R(t) \cup N_O(t) \cup N_{ID}(t) \quad (\text{III.13})$$

### ***C.2. Une représentation matricielle pour modéliser les offres de covoiturage***

Deux véhicules ou plus peuvent avoir un ou plusieurs points en commun (i.e. leurs parcours se rejoignent en des points d'intersection distincts). Ils peuvent aussi bien se partager des trajets ou des tronçons de routes que ce soit dans le même sens ou dans des directions opposées, et ce en même temps ou durant des intervalles de temps différents. Partant de ce principe, nous nous proposons de modéliser les offres de l'ensemble des véhicules de la flotte de sorte à en faire ériger les ressemblances, essentiellement par rapport aux routes empruntées. Ces offres sont ainsi regroupées en un ensemble d'itinéraires composés ou élémentaires construits relativement aux trajets offerts. Toujours est-il qu'au niveau de cette modélisation, doit figurer toute information élémentaire sur chaque offre existante sur chaque parcelle d'itinéraire représentée. Nous nous proposons alors de réorganiser les offres par route en une matrice Origine Destination  $I(t) = (r_{o,d})_{1 \leq o,d \leq f}$ . Cette modélisation nous permet de représenter jusqu'à la plus petite parcelle de chemin sur laquelle il existe au moins une offre de covoiturage et de stocker l'information pertinente sur celle(s)-ci. Les indices  $o$  et  $d$  des éléments de  $I(t)$  se réfèrent ici aux nœuds origine et destination d'une route  $[a \rightarrow b]$  modélisée dans cette représentation matricielle. Chaque indice représente de ce fait un nœud qui peut être soit origine, soit destination soit les deux à la fois. Dans ce dernier cas, deux nœuds  $a$  et  $b$  représentent à tour de rôle des segments de route différents (e.g.  $[a \rightarrow b]$  ou  $[b \rightarrow a]$ ) en fonction de l'offre considérée. Un itinéraire complet devant être réalisé peut ainsi

être reconstruit à partir de cette représentation en recombinaison les sections de route successives offertes par une même voiture de la flotte ou des véhicules différents. La représentation matricielle proposée est faite à la base d'une décomposition de l'ensemble des trajets globaux en routes élémentaires  $[a \rightarrow b]$  telles que  $(a, b) \in A(t)$ . Cette décomposition fait figure d'un désassemblage des itinéraires des véhicules en fonction de leurs origines, destinations ainsi que de leurs points de passage :

$$a \in \{N_o(t) \cup N_{ID}(t)\} \text{ et } b \in \{N_o(t) \cup N_{ID}(t)\} \quad (\text{III.14})$$

$\forall o \in \{1..f\}$  et  $\forall d \in \{1..f\}$ , les indices  $o$  et  $d$  de la matrice  $I(t)$  se réfèrent ainsi à l'ensemble de ces nœuds, avec  $f = \text{Card}(N_o(t) \cup N_{ID}(t))$ . D'où  $f \leq 2m + Z$ , avec  $Z = \text{Card}(N_{ID}(t))$ . L'ensemble des offres est ainsi ramené à une représentation matricielle des ensembles de routes desservies considérant jusqu'à la plus petite partie de chemin  $[a \rightarrow b]$  définie par une origine  $a$  vers une destination  $b$  :

$$I(t) = \begin{matrix} & 1 & \dots & o & \dots & d & \dots & f \\ \begin{matrix} 1 \\ \vdots \\ o \\ \vdots \\ d \\ \vdots \\ f \end{matrix} & \left( \begin{array}{cccccc} \phi & & \dots & & \dots & \\ & \ddots & & & & \dots \\ \dots & & \phi & & r_{o,d} & \phi \\ & \dots & & \ddots & & \dots \\ X & & r_{d,o} & & \phi & \\ \dots & & & & & \ddots \\ \dots & & & X & & \phi \end{array} \right) \end{matrix}$$

Chaque élément  $r_{o,d}$  de la matrice représente l'offre sur la section de route représentée  $[a \rightarrow b]$ . Les indices de la matrice étant représentés par l'ensemble global des origines, destinations et points de passage de toutes les voitures, des sections de routes peuvent être représentées au niveau de cette même matrice sans pour autant être desservies ou même permettre le passage dans le sens indiqué. Les éléments de la matrice peuvent ainsi, selon le cas, avoir l'une des valeurs suivantes :

$$r_{(o,d)} = \begin{cases} \Phi_1 \\ \Phi_2 \\ \Phi_3 \end{cases}$$

avec :

$\Phi_1 = X$  : si selon le code de la route,  $[a \rightarrow b]$  ne peut être traversée dans le sens indiqué;

$\Phi_2 = *$  : si  $[a \rightarrow b]$  est praticable mais aucune offre n'existe (i.e. aucun véhicule ne la dessert);

$\Phi_3 = \bigcup_{j=1}^m \{(V_j, l_{a,b}^j, [d_a^j, a_b^j])\}$  tel que  $[a \rightarrow b] \prec [O_j^+ \rightarrow O_j^-]$  : L'ensemble des véhicules traversant la route

$[a \rightarrow b]$  avec, pour chacun, les différents paramètres de déplacement sur celle – ci

Nous pouvons remarquer que les tronçons de route aussi bien que leur recombinaison, c'est-à-dire l'information complète sur les itinéraires globaux des voitures, peuvent être retrouvés au niveau de cette matrice. Les itinéraires initiaux des véhicules ainsi recombosés sont appelés  $IT_{V_j}$ ,  $\forall j \in \{1..m\}$ , et seront définis dans la suite. Un même véhicule peut alors se trouver dans plusieurs éléments de la matrice indiquant son passage par les routes relatives aux éléments en question et qui composent l'itinéraire global lui restant à parcourir.

Quand au moins une offre existe sur un tronçon de route  $[a \rightarrow b] \prec IT_{V_j}(t)$ ,  $r_{o,d}$  contient la liste des paramètres spécifiques à chaque véhicule desservant la partie du trajet en question, à savoir :

- $l_{a,b}^j$  : qui est le nombre de places disponibles dans le véhicule  $V_j$  sur la route  $[a \rightarrow b]$ . Selon le procédé de traitement des affectations des véhicules aux passagers détaillé dans le chapitre suivant, une voiture peut être affectée à différents passagers pouvant être pris puis déposés à des endroits (i.e. nœuds) distincts de son chemin. A chaque prise et dépose, une mise à jour des paramètres de la voiture sur le chemin indiqué est nécessaire, notamment le nombre de places vides à bord. Ce nombre peut ainsi différer d'une section à une autre ou d'un bout de chemin à l'autre du moment qu'il y a lieu de récupérer ou déposer un ou plusieurs passager(s) dans l'une de ses extrémités.
- $[d_a^j, a_b^j]$  : indique pour tout véhicule  $V_j$  desservant  $[a \rightarrow b]$  le temps de départ au plus tôt  $d_a^j$  à partir de l'extrémité initiale  $a$  de  $[a \rightarrow b]$  et le temps d'arrivée au plus tard  $a_b^j$  à son extrémité finale  $b$ . Ces paramètres sont calculés par le système en fonction des spécificités de déplacement initialement indiquées par le conducteur concerné et dépendent notamment des fonctions de pondération et estimation du retard énoncées précédemment (équations III.5, III.6 et III.8).

Les éléments de la matrice  $I(t)$  sont relatifs à des sections de routes desservies ou non par un ou plusieurs véhicules. Ces routes sont ainsi obtenues de par une décomposition automatique non voulue des trajets offerts, c'est à dire qui existe implicitement conformément à la nature des routes et itinéraires. Cette décomposition met en exergue l'existence ou non d'offres sur chacun des chemins représentés et notamment leur multiplicité.

La réorganisation des itinéraires des voitures sous formes de structures composées est obtenue en considérant les nœuds intermédiaires traversés et donc les sections de routes reliant chaque paire de nœuds successifs appartenant aux points de passage du véhicule considéré. Elle est ainsi à l'origine de la modélisation des itinéraires des véhicules sous forme de chemins recomposés comme le montre la Figure III.5. Cette modélisation illustre une succession (juxtaposition) de parties d'itinéraires élémentaires dans un ordre bien déterminé pour chaque véhicule  $V_j, \forall j \in \{1..m\}$ .

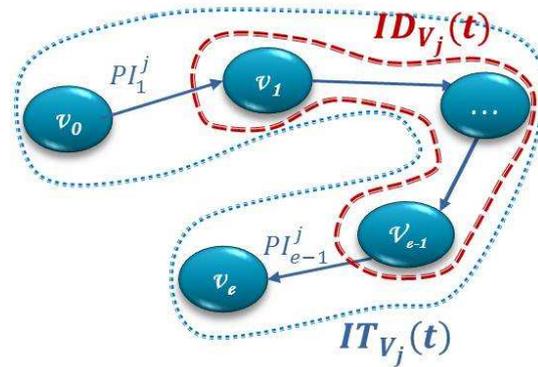


Figure III.5. Itinéraires composés des voitures

Ainsi, à tout véhicule  $V_j$  de la flotte est associé un itinéraire global  $IT_{V_j}$  mis à jour en temps réel à l'instant  $t$  considéré.  $IT_{V_j}$  se présente sous forme d'une succession de nœuds (i.e. chaîne de successeurs) commençant à son point de départ (i.e. l'origine de son déplacement, probablement la position actuelle de la voiture) jusqu'à sa destination finale. Par conséquent,  $\forall j \in \{1..m\}$ ,  $IT_{V_j}$  se définit comme une chaîne de nœuds ordonnés  $(v_0, v_1, \dots, v_e)$ , avec  $e = z + 2$  et  $z = \text{Card}(ID_j(t))$  :

$$IT_{V_j}(t) = (v_0, v_1, \dots, v_e) \text{ avec } \begin{cases} v_k \in N(t), \forall k \in [0, e] \\ v_0 = O_j^+, v_e = O_j^-, \bigcup_{k=1}^{e-1} v_k = ID_j(t) \\ (v_k, v_{k+1}) \in A(t), \forall v_k, v_{k+1} \in IT_{V_j}(t); k \in \{0..e-1\} \end{cases}$$

Chaque deux nœuds successifs de cet ensemble sont reliés par un arc dans  $A(t)$  et l'itinéraire global de la voiture est ainsi défini comme étant des routes adjacentes partant de l'origine de son déplacement vers le point d'arrivée indiqué. Dans la formulation qui suit (i.e. équation III.15), un itinéraire  $IT_{V_j}$  est défini comme étant une jonction de  $e-1$  Parties d'Itinéraires ( $PI$  : Partial Itineraries) selon qu'il se compose de  $e$  nœuds successifs :

$$IT_{V_j}(t) = (PI_1^j \oplus \dots \oplus PI_{e-1}^j) = ([O_j^+ \rightarrow ID_1], \dots, [ID_{q-1} \rightarrow ID_q], \dots, [ID_z \rightarrow O_j^-]) \quad (III.15)$$

$PI_s^j$  représente la Partie d'Itinéraire numéro  $s$  dans l'itinéraire global  $IT_{V_j}(t)$  de  $V_j$  et  $\oplus$  est ici utilisé comme opérateur de juxtaposition pour montrer la succession des sections de routes désignées par  $PI$  dans un ordre bien déterminé afin de former l'itinéraire global. Chaque  $PI$  est modélisé par l'élément qui lui correspond dans la matrice  $I(t)$  ; il possède ses propres propriétés (origine, destination, etc.) et peut faire l'objet d'une ou plusieurs offres.

Par ailleurs, pour chaque offre sur un  $PI$  en particulier, les paramètres qui lui sont spécifiques sont calculés automatiquement par le système. Tel que ça a été mentionné précédemment, le nombre de places disponibles est mis à jour lors de chaque prise et dépose de passager(s). Aussi, il est possible de calculer les extrémités de l'intervalle de temps de départ et d'arrivée sur toute partie d'itinéraire  $[o \rightarrow d]$  dès lors que l'on détient les informations nécessaires sur les temps de parcours du segment de route précédent. Par informations nécessaires, nous entendons plus précisément les temps de départ au plus tôt et temps d'arrivée au plus tard du véhicule  $V_j$  sur la section de route  $[h \rightarrow o]$  qui précède directement  $[o \rightarrow d]$ .  $[h \rightarrow o]$  précède  $[o \rightarrow d]$  si  $o \in N_h^+(t)$ ,  $d \in N_o^+(t)$ ,  $(h, o) \in A(t)$ ,  $(o, d) \in A(t)$  et  $[[h \rightarrow o] \oplus [o \rightarrow d]] \prec IT_{V_j}(t)$  comme le montre la Figure III.6 :

$$\begin{cases} d_o^j(t) = d_h^j(t) + \rho_{h,o}(t) = \text{le temps d'arrivée au plus tôt à } o \\ a_d^j(t) = d_o^j(t) + \rho_{o,d}(t) + \text{Distance}(O_j^+, d) * \frac{(a_{o_j^-}^j - d_{o_j^-}^j)}{\text{Distance}(O_j^+, O_j^-)} \end{cases}$$

Avec  $\tau(o, d) = \text{Distance}(O_j^+, d) * \frac{(a_{o_j^-}^j - d_{o_j^-}^j)}{\text{Distance}(O_j^+, O_j^-)}$  est la proportion de retard possible

pour parvenir à  $d$  en partant de  $o$ . Ce retard est estimé en fonction du retard toléré par l'utilisateur sur la totalité du trajet, calculé comme étant  $(a_{o_j^-}^j, d_{o_j^-}^j)$ . Nous avons alors

$\alpha = a_{o_j^+}^j - d_{o_j^-}^j$  la valeur du retard que le conducteur tolère pour parcourir son chemin global  
 $\beta = \text{Distance}(O_j^+, O_j^-)$ .  $a_{o_j^+}^j$  et  $d_{o_j^-}^j$  expriment respectivement le temps d'arrivée au plus tard fixé par le conducteur de  $V_j$  lui-même et le temps de départ (arrivée) au plus tôt à  $O_j^-$  calculé en fonction de l'évolution de  $V_j$ .

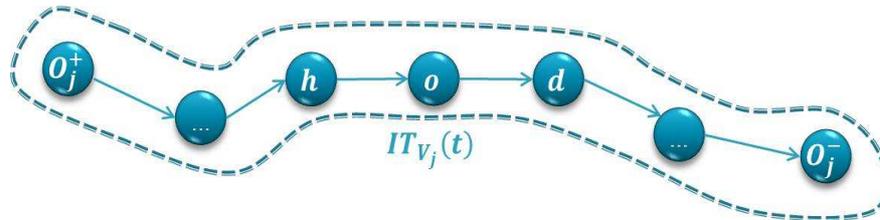


Figure III.6. Succession de tronçons sur le parcours d'une voiture

Les données transitant au niveau de notre système ainsi définies et modélisées, nous procédons dans ce qui suit à la formulation des différentes contraintes d'affectation pour la vérification des correspondances entre les offres et demandes sur la base de cette modélisation.

### III.3.1.2. Modélisation des contraintes d'affectation des véhicules aux usagers

Afin d'optimiser le processus de traitement des requêtes des utilisateurs du système, il est impératif de considérer un certain degré de souplesse par rapport à la construction des itinéraires répondant à leurs besoins en matière de déplacement. Cette flexibilité touche essentiellement les véhicules affectés à un même trajet demandé. En effet, compte tenu de la variabilité de l'offre ainsi que de l'incertitude de l'environnement, il est très peu probable qu'une offre globale, répondant à la totalité de la demande, existe. Notre système dépend ainsi entièrement et uniquement des conducteurs inscrits, connectés au système et proposant des trajets de covoiturage spécifiques à leurs propres besoins en matière de déplacement. L'approche que nous proposons doit par conséquent tenir compte des préférences de déplacement des individus demandeurs de places de covoiturage aussi bien que des conducteurs offrant des trajets dont les paramètres sont fixés au préalable mais tout de même quelque peu flexibles.

La flexibilité des traitements concerne de ce fait la possibilité de combiner plusieurs bouts d'itinéraires entre lesquels l'utilisateur aura à faire un changement de véhicule. Un changement est envisageable s'il n'existe pas de voiture capable de répondre à la totalité de sa

demande. Ceci se traduit par la création d'un itinéraire composé de plusieurs sous-chemins auxquels sont affectés des véhicules différents à chaque fois. La Figure III.7 illustre un exemple de composition d'itinéraire solution à l'utilisateur  $U_p$  à partir de la flotte disponible où plusieurs véhicules peuvent être considérés si une voiture unique ne peut satisfaire à elle seule sa demande globale.

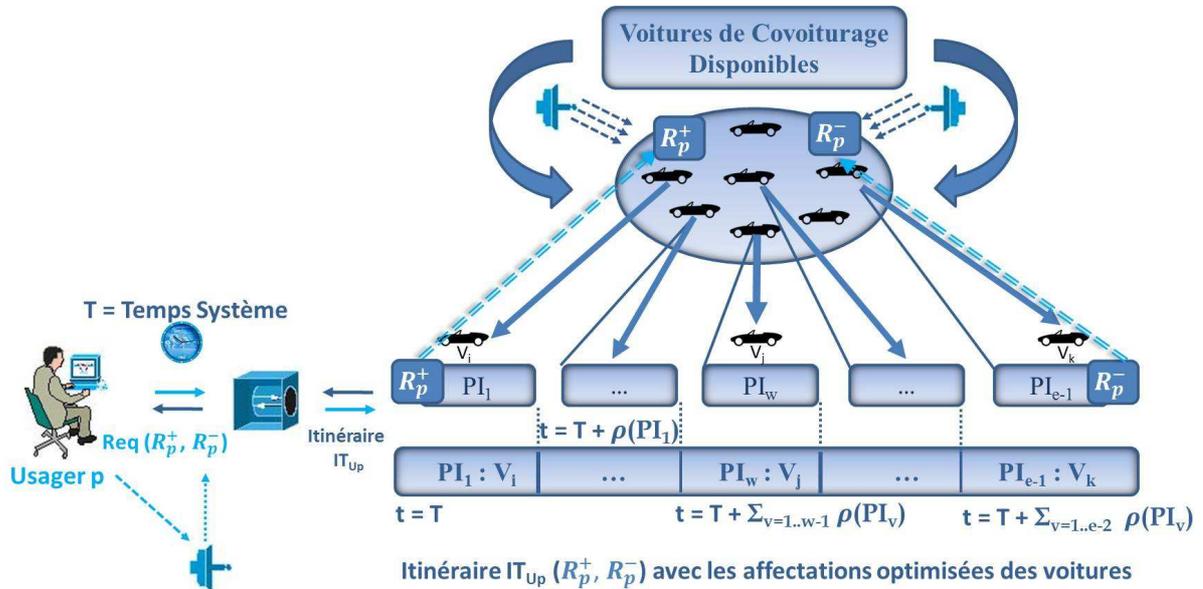


Figure III.7. Composition des solutions d'affectation en fonction de l'offre disponible

Comme le montre la Figure III.7, notre système se base essentiellement sur un module de géolocalisation permettant de suivre et capturer en temps réel les positions courantes des différents usagers (i.e. conducteurs ou piétons) et de rechercher les possibilités susceptibles de répondre à la demande d'un utilisateur  $U_p$ . La multiplicité des offres sur un même trajet nous incite à choisir celle qui minimise au maximum la durée globale du voyage : critère d'optimisation choisi au niveau de notre système comme énoncé précédemment. Toujours est-il que dans l'une des deux approches que nous proposons et détaillons ultérieurement (c.f. Chapitre 3), nous favorisons en premier lieu le critère de confort par rapport au critère de temps de parcours, minimisant en priorité le nombre de transferts entre voitures. En effet, une décomposition n'est envisageable que lorsqu'aucune offre susceptible de répondre à la totalité du trajet demandé n'existe ou ne peut être considérée faute de satisfaction des différentes contraintes de correspondance. Ces contraintes seront détaillées dans la suite de cette section. En l'absence d'une offre globale, l'ensemble des sections de routes pouvant répondre à une partie du voyage est recherché en fonction des offres disponibles. La voiture qui optimise le

mieux la durée globale du trajet sur chaque tronçon considéré est finalement sélectionnée pour construire la solution finale :

$$S_{U_p}(t) = (S_{1,p} \circ \dots \circ S_{w,p} \circ \dots \circ S_{e-1,p}) \quad (\text{III.16})$$

Où  $S_{U_p}(t)$  est une solution globale optimisée fournie à l'utilisateur  $U_p$  à la fin du processus de traitement de la requête  $R_p(t)$ . Cette solution est, comme le montre l'équation (III.16), reconstruite à partir d'un séquençement de plusieurs solutions partielles  $S_{w,p}$  relatives chacune à une partie du chemin à parcourir par  $U_p$  pour qu'il puisse atteindre son objectif.

«  $\circ$  » est ici un opérateur de combinaison des solutions partielles  $(S_{w,p})$ , indiquant le respect de l'ordre et le sens dans lequel sont affichées les solutions, et ce à l'image du séquençement des parties d'itinéraires  $(PI_{w,p})$  qui leur sont relatives. Chaque solution partielle  $S_{w,p}$ ,  $\forall w \in \{1..e-1\}$  est alors modélisée sous forme d'une structure de données contenant les informations nécessaires de parcours du bout de chemin en question. Outre les adresses de prise  $(PI_{w,p}^+)$  et dépose  $(PI_{w,p}^-)$ , cette structure fournit de ce fait les données essentielles sur la solution ; à savoir les détails sur la voiture ayant été affectée  $(V_j)$ , les temps de départ au plus tôt et d'arrivée au plus tard de celle-ci sur les extrémités du chemin :

$$S_{w,p} = (PI_{w,p}^+, PI_{w,p}^-, V_j, d_{PI_{w,p}^+}^j, a_{PI_{w,p}^-}^j) \quad (\text{III.17})$$

Ainsi, pareil aux itinéraires des voitures tels que définis précédemment, les solutions auxquelles aboutit le système pour la recherche d'affectations possibles sont construites à l'image d'un itinéraire composé répondant au déplacement demandé. Cette composition de sections de route se base essentiellement et uniquement sur l'offre disponible sur les différentes parcelles de routes desservies et susceptibles de répondre à la totalité ou une partie de la demande. L'itinéraire solution fourni à l'utilisateur  $U_p$  est par la suite formulé comme suit :

$$IT_{U_p}(t) = (PI_{1,p}^v \oplus \dots \oplus PI_{w,p}^j \oplus \dots \oplus PI_{e-1,p}^k) \text{ tel que } 1 \leq v, j, k \leq m \quad (\text{III.18})$$

Où  $PI_{w,p}^j$  est la  $w^{\text{ième}}$  Partie d'Itinéraire desservie par le véhicule  $V_j$  dans le chemin global défini comme résultat de traitement de la requête de l'utilisateur  $p$  (i.e.  $R_p(t)$ ). Ici,  $j$  se réfère à la voiture qui optimise la durée du voyage parmi  $V_{w,p}$  voitures sur la section de route

considérée  $PI_{w,p} \cdot V_{w,p}$  est l'ensemble de véhicules dont les itinéraires respectifs incluent le chemin  $PI_{w,p}$  dans le sens indiqué tout en satisfaisant les contraintes de correspondance formant ainsi un ensemble de solutions « faisables » sur la section de route considérée.

En effet, tel que énoncé ci-dessus, deux voitures ou plus peuvent se rencontrer à des points d'intersection de leurs itinéraires respectifs sans pour autant passer par les mêmes chemins, comme ils peuvent très bien se rejoindre sur un même bout d'itinéraire simultanément ou quasi-simultanément ou encore à des dates (i.e. temps) éloignées. Les parties d'Itinéraires communes sont par conséquent traitées de sorte à en extraire l'ensemble des solutions faisables en premier lieu, c'est-à-dire les voitures qui traversent les routes correspondantes dans le sens adéquat, dans la période souhaitée et tout en offrant un nombre de places suffisant. En second lieu, la voiture optimisant la durée du voyage est sélectionnée.

Les algorithmes que nous proposons s'exécutent ainsi suivant deux étapes principales : recherchant l'ensemble des solutions faisables  $V_{w,p}$  dans la première, puis la solution optimale parmi celles-ci dans la seconde. Pour être considérée comme étant faisable, une proposition relative à une offre  $O_j$  sur un tronçon de route  $PI_{w,p}$  doit satisfaire un ensemble de contraintes de correspondance dont voici la formulation :

- $PI_{w,p} \prec IT_{V_j}$  : Une voiture  $V_j$  ne peut être considérée dans le processus de recherche de possibilités d'affectation que si elle répond à la première contrainte de correspondance (i.e. matching) entre son itinéraire global et  $PI_{w,p}$ . Ceci se traduit par le fait que  $V_j$  doit parcourir la section de route de son départ ( $PI_{w,p}^+$ ) jusqu'à son extrémité finale ( $PI_{w,p}^-$ ) dans le sens approprié.
- $l_w^j \geq P_p$  : Le véhicule  $V_j$  ne constitue une solution faisable que s'il répond à la contrainte de disponibilité de places. En effet, il doit fournir un nombre de sièges vacants plus grand ou égal au nombre de passagers demandant à se déplacer sur la route  $PI_{w,p}$  tel que indiqué dans la requête insérée par l'utilisateur  $U_p$ .
- $d_{PI_{1,p}^j} \geq d_p$  : Le temps de départ au plus tôt spécifié par le conducteur de  $V_j$  (ou calculé s'il ne s'agit pas de son point de départ) sur la première parcelle de route  $PI_{1,p}$  dans  $IT_{U_p}$  doit être supérieur ou égal à celui spécifié par  $U_p$ . La section de route  $PI_{1,p}$  part

de l'origine de déplacement du voyageur  $U_p$  constituant ainsi la première partie de son itinéraire global.

- $d_{PI_{w,p}^j} \geq a_{PI_{w-1,p}^v} - \tau(PI_{w-1,p}^+, PI_{w-1,p}^-), \forall w \in \{2..e-1\}$  : Pour le reste des tronçons pouvant être combinés pour ainsi tracer l'itinéraire global, une voiture  $V_j$  n'est dite faisable sur un bout de chemin ( $PI_{w,p}$ ) que si son temps de départ au plus tôt à l'origine de celui-ci ( $PI_{w,p}^+$ ) est supérieur ou égal au temps d'arrivée au plus tôt à l'extrémité de la section de route qui le précède directement ( $PI_{w-1,p}$ ), quelque soit le véhicule  $v$  qui lui est affecté (sachant que  $PI_{w,p}^+ = PI_{w-1,p}^-$ ). Ainsi, pour chaque deux tronçons successifs, le départ sur le deuxième doit succéder (de préférence directement et sans retard) à l'arrivée sur le premier. Ces temps de départ et d'arrivée sont calculés en fonction des voitures et de leurs parcours respectifs tout en respectant la contrainte du temps réel et en prévoyant une marge de tolérance par rapport au retard qui peut survenir en cas d'imprévus possibles.
- $a_{PI_{e-1,p}^j} \leq a_p$  : cette dernière contrainte concerne notamment le temps d'arrivée au plus tard d'une voiture  $V_j$  pouvant être affectée au dernier tronçon ( $PI_{e-1,p}$ ) de l'itinéraire de l'utilisateur  $p$ . Le système proposé doit respecter la marge de retard tolérée par l'utilisateur  $U_p$ . Un véhicule  $V_j$  ne peut ainsi lui être affecté que s'il parvient à la fin du parcours à un temps inférieur ou égal au temps d'arrivée au plus tard spécifié par  $U_p$ . Par conséquent, le véhicule  $V_j$  doit atteindre l'extrémité finale du dernier tronçon ( $PI_{e-1,p}^- = R_p^-$ ) de l'itinéraire  $IT_{U_p}(t)$  à  $a_p$  au plus tard.

La vérification de ces contraintes est faite pour toutes les offres de covoiturage émises à un instant  $t$  afin d'en extraire celles qui sont susceptibles d'être considérées comme des solutions potentielles à la requête traitée. L'ensemble des solutions faisables sur une partie de l'itinéraire construit au fur et à mesure est ainsi obtenu. Cet ensemble nommé  $V_{w,p}$  désigne les véhicules qui desservent le tronçon numéro  $w$  de l'itinéraire de l'utilisateur  $U_p$  tout en satisfaisant la totalité des contraintes ci-dessus énoncées.

Puisque nous nous focalisons essentiellement au niveau de notre approche sur l'aspect optimisation des solutions fournies à l'utilisateur, une unique solution est ainsi choisie pour

chaque route  $w$  de manière à en optimiser le parcours. Le critère que nous avons déjà fixé se réfère à la Durée Globale du Trajet (i.e.  $TD$  pour global Trip Duration) qui est calculable en termes de temps d'attente et temps de parcours. Ces derniers sont calculés à leur tour grâce à la fonction de pondération précédemment définie dans l'équation III.6 ( $\rho$ ). Une fonction Fitness permet ainsi d'agréger les temps d'attente ( $W_{j,w}$ ) et de parcours ( $T_{j,w}$ ) sur une section de route ( $PI_{w,p}$ ) permettant ainsi de calculer la durée du voyage sur celle-ci pour tout véhicule  $V_j \in V_{w,p}$  :

$$TD_{PI_{w,p}}^j = \omega_{1_p} W_{j,w} + \omega_{2_p} T_{j,w} \text{ avec } V_j \in V_{w,p} \quad (\text{III.19})$$

$\omega_{1_p}$  et  $\omega_{2_p}$  étant les paramètres d'agrégation pondérant respectivement  $W_{j,w}$  et  $T_{j,w}$ . Ils sont fixés par l'utilisateur lui-même et indiquent ses préférences en matière de déplacement selon qu'il favorise l'optimisation par rapport à l'attente plutôt que la durée de parcours ou bien passer moins de temps dans la voiture même si ça se répercute sur son temps d'attente. La fonction Fitness ( $TD$ ) ainsi identifiée permet, en agrégeant les fonctions de temps d'attente et en les amplifiant des poids fixés par les usagers eux-mêmes, de trouver le juste équilibre conformément aux préférences de ces derniers.

La solution retenue à la fin,  $TD_{PI_{w,p}}^*$ , est celle qui minimise le plus la fonction Fitness sur la section  $w$  sur la base de la fonction objectif définie dans ce qui suit :

$$TD_{PI_{w,p}}^* = \text{Min}_{V_j \in V_{w,p}} (TD_{PI_{w,p}}^j) = \text{Min}_{V_j \in V_{w,p}} (\omega_{1_p} W_{j,w} + \omega_{2_p} T_{j,w}) \quad (\text{III.20})$$

Sur la base de la fonction objectif globale ainsi définie et conjointement au principe de recomposition d'une solution complète à partir de plusieurs partielles, la solution optimisant l'itinéraire global ( $IT_{U_p}$ ) est celle qui optimise la combinaison de toutes les sections  $PI_{w,p}$ , tel que  $w \in \{1..e-1\}$ , et dont la jonction constitue le chemin  $[R_p^+, R_p^-]$ . Notre objectif initial étant d'optimiser la durée sur le voyage complet effectué par  $U_p$ , la solution finale revient donc à considérer pour chaque section,  $PI_{w,p} \in IT_{U_p}$ , le véhicule minimisant la durée du trajet sur celle-ci. La somme des durées optimisées sur l'ensemble des tronçons d'itinéraires constituant le chemin complet conduit par conséquent à la minimisation du temps passé à voyager au total sur  $IT_{U_p}$  :

$$TD_{U_p}^* = \sum_{w=1}^{e-1} \text{Min}_{V_j \in V_{w,p}} (TD_{P_{j,w,p}}^j) = \sum_{w=1}^{e-1} \text{Min}_{V_j \in V_{w,p}} (\omega_{1p} W_{j,w} + \omega_{2p} T_{j,w}) \quad (\text{III.21})$$

La nécessité de cadrer avec les contraintes de correspondance entre les offres et les demandes ainsi que la gestion des conflits d'affectations possibles entre des usagers demandant un même itinéraire implique le fait que les solutions ainsi générées peuvent ne pas être optimales mais seulement optimisées. Par ailleurs, même si des solutions optimales ont pu être générées sur l'ensemble des itinéraires partiels, la recombinaison de celles-ci n'induit pas forcément une solution optimale globale pour le trajet demandé.

Sur la base de la formulation et modélisations précédemment définies pour les données du système, un procédé de décomposition est proposé dans ce qui suit pour mettre en évidence la répartition difforme de ces données sur le réseau de covoiturage. Cette décomposition sera utile pour la distribution et décentralisation du procédé d'optimisation des affectations en parallèle.

### III.3.2. Subdivision du réseau géographique desservi : mise en place d'une architecture physique dynamique distribuée

Dans le but d'optimiser les traitements effectués par un système aussi complexe que le covoiturage dynamique optimisé, énoncé selon une précédente étude comme étant un problème NP-difficile (*c.f.* §II.3.2.); nous nous proposons d'élaborer une plateforme dynamique distribuée. La mise en place d'une telle plateforme favorise la décomposition du processus global initial en plusieurs tâches moins complexes. Ces tâches correspondraient au traitement d'itinéraires partiels et seront réalisées de manière décentralisée et en parallèle afin d'atténuer la complexité des traitements effectués en vue de réaliser l'affectation des voitures aux usagers en temps réel. Le principe de décomposition que nous proposons de mettre en place permet ainsi d'alléger l'impact des traitements a priori lourds sur la qualité du service fourni et les temps de réponse.

Pour ce faire et compte tenu de la modélisation détaillée précédemment, nous nous proposons de mettre en place une architecture « physique » distribuée établie à l'image de l'état courant du réseau géographique de desserte. Pour cela, nous nous sommes inspirés de quelques travaux existants évoquant le même principe et dont les modélisations proposées sont quelque peu semblables à la nôtre. Nous citons particulièrement les travaux relatifs aux problèmes de transport tels que ceux de (18), (2) et (69). Les approches proposées par ces derniers sont basées sur le concept de graphe dynamique distribué et mettent en place des

systèmes d'aide au déplacement des voyageurs. Les services fournis de par ces travaux sont particulièrement utiles puisqu'ils proposent des systèmes d'aide à la décision distribués dans un contexte d'optimisation des systèmes de transport. Les travaux conduits dans ce contexte traitent du problème de recherche d'itinéraires pour l'aide au transport des voyageurs et proposent des modélisations d'architectures distribuées en faveur de la décentralisation de processus.

Dans ce contexte, nous nous proposons de modéliser la représentation du réseau de desserte sous forme d'un graphe dynamique distribué mettant en place un procédé de décomposition de la modélisation précédemment définie. En effet, les données du système telles que précédemment formulées comme étant fortement dépendantes de la fonction temps, la modélisation choisie suit la variation de celles-ci démontrant les propriétés de dynamique du graphe représentatif du réseau de desserte. Celles-ci ayant été prouvées, nous nous concentrons dans ce qui suit sur le choix adéquat d'une stratégie de subdivision qui mette en exergue l'aspect distribué de ces données.

### *III.3.2.1. Subdivision : Principe et objectifs*

Sur la base de l'objectif premier de nos travaux visant à mettre en place un service de covoiturage efficace et rapide, il est impératif que les performances du système soient à la hauteur des attentes des usagers. Il s'agit essentiellement de mettre à leur disposition un service qui permette de rechercher à leur place l'offre qui corresponde le mieux à leurs attentes tout en assurant l'efficacité des traitements.

Etant donné la complexité combinatoire des problèmes de covoiturage dynamiques optimisés, nous avons alors opté pour un choix qui porte sur la réduction du problème initial en un ensemble de sous-problèmes de moindre complexité. Ceci se traduit dans notre approche par une décomposition du processus d'affectation en plusieurs sous-processus légers (i.e. threads) exécutés en parallèle et envisagés comme des unités autonomes. Ces dernières sont créées conformément au principe de subdivision appliqué sur le réseau géographique de desserte. L'architecture logicielle définie par la suite est à l'image de la modélisation physique établie.

La modélisation sous forme de graphe dynamique proposée précédemment se base essentiellement sur les intrants du système, à savoir l'offre et la demande de covoiturage. Nous avons aussi choisi de nous concentrer sur les mêmes paramètres de notre système afin d'appliquer un procédé de décomposition dynamique sur le terrain où évoluent les voitures et

piétons, et ce en vue de mettre en exergue les zones de concentration des offres et demandes. Notre objectif premier étant d'en ressortir les aires géographiques impliquant plusieurs usagers, et ce en faveur du traitement parallèle des requêtes. En effet, rassemblées dans un périmètre restreint, les demandes et offres sont dotées d'une probabilité de ressemblance plus forte que si elles sont éloignées dans l'espace. Ceci nous aidera donc à mettre en évidence les similitudes entre les demandes et les offres aussi bien qu'entre les demandes elles-mêmes. Notre but premier étant d'alléger les traitements d'affectation et ainsi atténuer l'impact de sa complexité grandissante sur les performances du système au fur et à mesure que le flux des passagers et conducteurs augmente.

### *III.3.2.2. Propriétés des zones de décomposition*

Le formalisme précédemment détaillé correspond à une modélisation conforme aux graphes dynamiques. La représentation que nous proposons est choisie de manière à modéliser, adéquatement, le réseau de desserte visualisé de par une cartographie réelle à l'aide de l'outil GPS intégré au niveau de notre système. Cette modélisation doit impérativement contenir toutes les données nécessaires aux traitements des requêtes et faire état de l'image du système de covoiturage à l'instant  $t$  considéré. En effet, pour une efficacité absolue, toute information existante est susceptible d'aider à la bonne réalisation du processus d'affectation et doit donc être représentée au niveau du modèle mis en place.

Par ailleurs, pour la décomposition du réseau de desserte, considérer un procédé efficace permettant d'englober l'ensemble des données disponibles tout en évitant les redondances d'information prime sur tout autre objectif. Nous nous proposons alors d'aboutir de par ce principe de subdivision à la création d'un ensemble de zones géographiques distinctes dont les principales caractéristiques sont les suivantes (52):

#### *a. Similarité de forme :*

Toutes les zones créées représentent des aires géographiques de formes et de dimensions similaires. En effet, chaque zone est déterminée en fonction d'un diamètre préfixé à l'avance par l'administrateur du système au niveau du paramétrage initial. Par conséquent, toutes les zones auront le même diamètre et donc la même surface. Nous voulons concrétiser de par cette propriété la conformité de toutes les parties du réseau à un modèle précis couvrant l'ensemble du territoire sur lequel sont dispersées les offres et demandes de covoiturage à l'instant  $t$  considéré.

**b. Distinction des surfaces couvertes :**

Toute zone possède son propre périmètre qui englobe un ensemble d'informations relatives aux demandes des covoiturés et/ou aux offres des covoitureurs. Ceci pour répondre à l'objectif premier du principe de subdivision proposé et qui concerne la mise en place d'une architecture logicielle distribuée à l'image de la modélisation du réseau réalisée lors de cette étape. En effet, à chaque zone correspondra une composante logicielle, plus ou moins indépendante, traitant de manière locale les données incluses dans chaque zone considérée. Un ensemble de composantes autonomes (ou semi-autonomes) est ainsi créé pour les zones établies. Elles s'exécutent en parallèle de manière à optimiser le traitement global du système.

**c. Intersection sur un territoire commun :**

Deux zones ou plus peuvent avoir des zones d'intersection communes. Ces dernières peuvent représenter un point unique ou plusieurs comme elles peuvent représenter une aire géographique vide. Les intersections de ces zones, dans ce cas, représentent simplement un territoire commun vierge, dénué de toute information faisant partie des intrants du système comme le montre l'exemple au niveau de la Figure III.8.

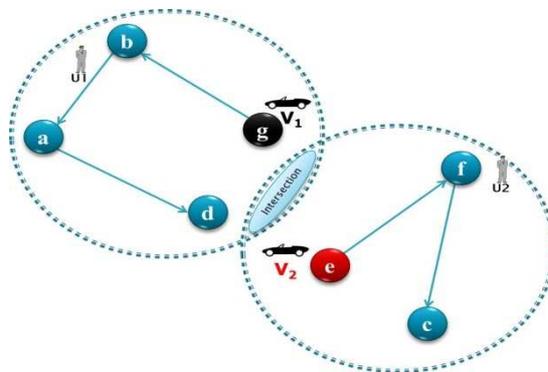


Figure III.8. Terrain vierge commun à deux zones

**d. Chevauchement des zones sur des données :**

Deux zones ou plus peuvent se croiser tout en se partageant des données susceptibles d'aider à la bonne mise en œuvre du système ; à savoir des points de départs et/ou d'arrivées de covoitureurs et/ou covoiturés, des points intermédiaires repères de passage de véhicules, des nœuds de prise ou dépose de passagers, ou même des bouts d'itinéraires en commun tel que c'est le cas pour l'exemple illustré dans la Figure III.9.

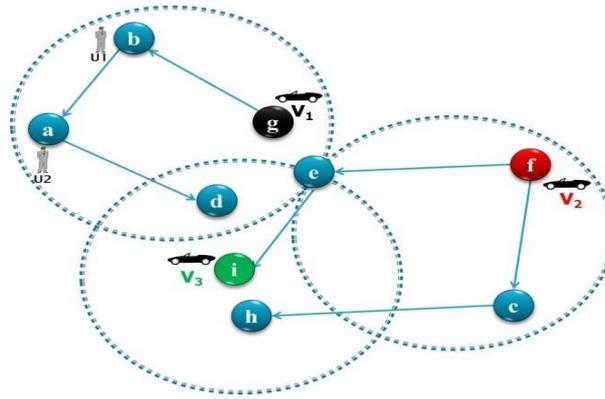


Figure III.9. Intersection non vide de zones (point(s) et / ou route(s))

**e. Traitement d'une requête exclusivement réservé à une seule zone :**

La redondance et le partage de données au niveau des zones est utile puisque chacune des composantes créées a besoin de pouvoir communiquer avec ses voisines pour pouvoir traiter au mieux les requêtes des usagers localisés au niveau des zones considérées. Cependant, elle peut être conflictuelle dans le cas où deux zones ou plus se partagent des données sur les origines de requêtes d'usagers. Dans ce cas, les composantes responsables de ces zones se verront traiter les mêmes requêtes accaparant inutilement les ressources. Ceci peut induire à une lourdeur des traitements et donc à une régression des performances du système. Aussi, un accès simultané aux mêmes données et leur traitement parallèle par des entités logicielles différentes peut engendrer des conflits au niveau des affectations réalisées. Des affectations différentes portant sur les mêmes usagers ont ainsi une forte probabilité d'avoir lieu. Afin de remédier à ce problème, les algorithmes de décomposition que nous proposons de mettre en place affectent les origines et destinations des requêtes des utilisateurs une et une seule fois, d'une part afin d'éviter les traitements inutiles, et d'autre part pour mieux cerner le problème et couvrir tout le réseau géographique avec un nombre minimum de zones géographiques. Celles-ci doivent alors être le plus dispersées possible, avoir de petites intersections tout en couvrant toutes les informations existantes. Ces informations, relatives aux données sur les usagers transitant au niveau du système font l'objet d'un traitement distribué sur les zones érigées. D'où l'utilité de distinguer l'appartenance de ces données en exclusivité à une zone bien déterminée, et ce afin d'éviter les pièges conflictuels d'affectations multiples de piétons, ou non faisables de voitures et ainsi veiller à la bonne marche du système dans le but essentiel de réaliser des affectations dynamiques optimisées et efficaces.

***f. Techniques de classification non supervisée pour la subdivision :***

Une zone est définie en fonction d'un nombre de points voisins dans le réseau spatio-temporel dans le cadre duquel le processus est conduit. En effet, à l'instant  $t$ , l'ensemble des points (i.e. nœuds précédemment définis) voisins en terme de distance et ayant lieu d'être considérés au moment de la création des zones sont à la base de l'établissement d'une zone spécifique. Dans ce contexte, nous nous sommes particulièrement intéressés aux méthodes de classification<sup>33</sup> de données suivant un ou plusieurs critères de rapprochement pour la création de nuages de points. Il s'agit dans ces méthodes et plus particulièrement dans la « *classification non supervisée* »<sup>34</sup> de créer un ensemble de  $k$  classes où chacune correspond à un ensemble d'individus. Les aires géographiques de plus ou moins forte densité sont alors à l'origine de la création de « clusters » auxquels on associe des étiquettes de classe. Les classes ainsi définies sont associées, dans notre étude, aux zones dont deux types sont discernés sur la base de la nature de l'information manipulée (c.f. § III.3.2.3). Les observations disponibles sur le réseau sont, par la suite, analysées et étudiées sur la base d'un critère bien déterminé, ici la distance ; et ce afin de pouvoir par la suite les rassembler en un ensemble de classes de voisins proches dans l'espace.

***g. Une zone est un cluster : espace de recherche limité***

Le réseau de desserte que nous modélisons est réduit à un espace bidimensionnel où l'information dont nous disposons se présente comme étant un ensemble de données relatives à des points géographiques. Ces points qui sont les individus, objets de classification, correspondent à des nœuds (adresses d'origines, destinations de déplacements, etc.) représentés par leurs coordonnées GPS (i.e. Longitude et Latitude). L'ensemble de ces données, constituant les observations du système, peut ainsi faire l'objet d'une classification où le caractère dominant de séparation est la distance entre chaque deux nœuds d'une population faisant partie d'un même cluster. Une classe réelle, sous-jacente occupe ainsi une région limitée du plan espace-temps représentant le réseau de desserte. Chaque classe ainsi constituée est relative à un ensemble de nœuds proches (i.e. voisins dans l'espace) observés de par la modélisation graphique précédemment détaillée. La représentation adoptée, détaillée dans un premier temps, est susceptible d'être décomposée et ainsi donner lieu à une multitude de clusters (i.e.

---

<sup>33</sup> <http://atlas.irit.fr/TETRALOGIE/annexeA.htm>

<sup>34</sup> [http://www.aiaccess.net/French/Glossaires/GlosMod/f\\_gm\\_classification\\_non\\_supervisee.htm](http://www.aiaccess.net/French/Glossaires/GlosMod/f_gm_classification_non_supervisee.htm)

zones). Dans notre cas, il s'agit d'un ensemble de zones bâties par le biais d'un principe de décomposition inspiré des techniques de classification et détaillé dans une section ultérieure (c.f. § III.3.2.3)

### ***h. Classification floue et dynamique***

Comme nous l'avons précédemment observé, deux zones ou plus peuvent se chevaucher et se partager un ou plusieurs individus (i.e. nœuds) avec des degrés d'appartenance différents et dont la somme est égale à 1 (118). Cela sous-entend que deux zones ou plus peuvent se partager des connaissances sur un ensemble de données communes. Par ailleurs, s'il s'agit de requêtes de covoiturés, le traitement en est exclusivement réservé à une zone et une seule ; probablement celle à laquelle appartient le degré d'appartenance le plus fort. Partant de ce principe, nous nous plaçons alors dans le cadre d'une classification non supervisée floue (119) où le nombre de groupes est à priori inconnu et doit être défini grâce à l'algorithme de décomposition. Aussi, étant placés dans le cadre d'un service de covoiturage dynamique où des offres et demandes peuvent survenir à n'importe quel moment, il est impératif de considérer l'évolution du nombre d'individus pour une continuelle mise à jour des classes ainsi établies. Le nombre même des clusters peut évoluer au fur et à mesure de l'évolution des traitements et donc au cours de l'exécution du processus d'affectation ou même durant le procédé de décomposition.

### ***i. Zone de forme circulaire :***

Chaque zone est associée à une région d'observations « *similaires* » selon le critère fixé (i.e. proches dans l'espace selon une distance fixe prédéfinie au préalable). De multiples tentatives nous ont conduits, dans la majorité des cas, à une ébauche sur la construction de zones dynamiques en matière de forme pour qu'elles puissent intégrer de nouvelles données et s'élargir en conséquence. Cependant, l'expansion dynamique et progressive des zones implique une forte tendance vers l'élargissement des espaces de recherche, le chevauchement des zones pouvant aller jusqu'à une superposition totale, et donc des traitements lourds, redondants rendant inutile le principe de décomposition. Chose qui viendra entraver notre principal objectif qui est de contrecarrer la complexité exponentielle du problème de CDO. Sur la base de ces limites et impacts négatifs d'un choix portant sur les formes dynamiques, une zone, construite à un instant  $t$  donné, est

donc assimilée à un espace géographique de forme circulaire figée et fixe. Elle est dotée des paramètres suivants :

- o Cardinal ( $N(t)$ ) : le nombre d'individus qui y sont inclus à l'instant  $t$  considéré
- o Centre ( $C_i$ ) : le centre de la zone  $Z_i$ .
- o Rayon ( $r_i$ ) : le rayon de la zone  $Z_i$ . Ce rayon est le même pour toutes les zones établies pour assurer leur conformité et aussi une bonne dispersion équitable des données entre celles-ci. La distance, terme de vérification de l'appartenance des nœuds au cluster, est à notre sens fixe. Ceci, entre autre, par souci d'équité au niveau de la distribution des tâches sur les entités logicielles responsables chacune de traiter une zone bien déterminée.

***j. Zone : calcul des centres***

A partir de l'ensemble de voisins déterminés à chaque itération de l'algorithme de subdivision détaillé plus tard, les propriétés de chacune des classes créées sont ainsi définies, essentiellement son centre et son rayon. A partir de ces paramètres, de nouvelles populations peuvent être créées déterminant ainsi l'espace de connaissances de chacune des zones (« classes ») identifiées (119). Le centre de la zone est calculé de sorte à ce qu'il soit le point représentatif (i.e. isobarycentre) des points les plus éloignés de l'ensemble considéré. Plusieurs cas de figure peuvent avoir lieu d'être et doivent donc être considérés selon que l'ensemble des voisins déterminés contient un seul, deux, trois nœuds ou plus. Ces cas sont expliqués dans ce qui suit :

- o La population de nœuds voisins déterminée contient un seul nœud éloigné du reste des points du réseau à une distance strictement supérieure au diamètre préfixé au niveau du système. La Figure III.10 montre le cas du traçage d'une zone pour un point unique qui est lui-même le centre de celle-ci.

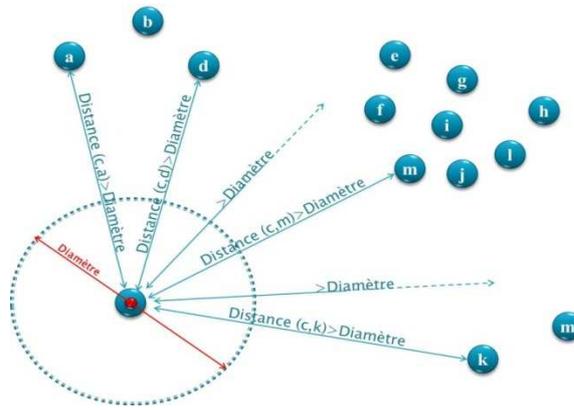


Figure III.10. Calcul des centres pour une zone contenant un point unique

- o Si les points voisins sont au nombre de deux et qu'aucun nœud n'appartient au périmètre de la zone incluant la paire de nœuds ainsi définie, le centre de la zone les incluant est alors calculé comme étant le milieu de la droite qui relie les deux points en question. Un exemple en est montré à la Figure III.11.

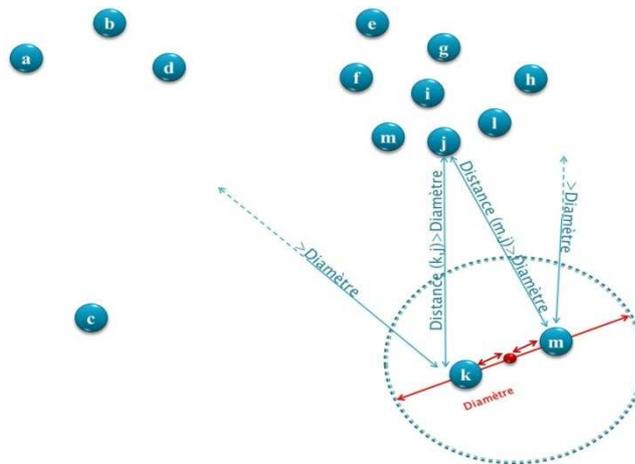


Figure III.11. Calcul des centres pour une zone contenant 2 points

La zone ainsi tracée est, par conséquent, établie sur la base du point équidistant calculé à partir des deux nœuds donnés et de leur voisinage, au départ, vierge. Cependant, l'espace de connaissance d'une zone peut évoluer pour inclure d'autres nœuds relatifs à de nouvelles offres ou demandes de covoiturage. si celles-ci arrivent à parvenir au système, elles doivent, en effet, être prise en compte instantanément et intégrées au traitement. Par ailleurs, une fois le centre établi, la zone est définitivement construite et ses propriétés (les centre et périmètre) demeurent inchangés jusqu'à la fin du processus.

- o Si l'ensemble des voisins a pour taille 3 ou plus ( $\text{Cardinal}(N) \geq 3$ ), alors le centre de la zone contenant l'ensemble de ces nœuds (i.e. individus) est le centre de gravité du triangle dont les sommets sont représentés par les points les plus distants de cet ensemble. Le centre de cette zone est donc déterminé comme tracé au niveau de la Figure III.12. Il s'agit en effet du centre de gravité identifié comme étant l'isobarycentre des trois sommets et le centre de masse de l'intérieur du triangle. Ce point se trouve aux deux-tiers de chaque médiane en partant du sommet. Les individus choisis comme sommets du triangle sont ceux qui présentent les plus grandes distances parmi celles calculées sur toute la population de nœuds voisins deux à deux.

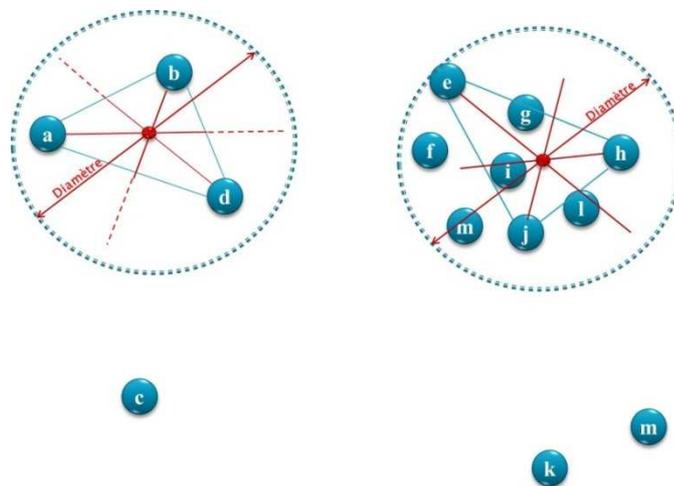


Figure III.12. Calcul des centres et création des zones : ensembles de 3 voisins ou plus

Ainsi, pour un nœud unique au sein de l'ensemble, le centre est le nœud lui-même. Dans le cas de deux nœuds, le milieu de la droite passant par les deux nœuds en question est considéré comme étant le centre de la zone. Dans le cas de trois nœuds ou plus, le centre du cercle les incluant est le centre de gravité du triangle constitué des trois points les plus éloignés les uns des autres dans l'ensemble considéré. Ainsi, tout ensemble dont la taille excède les trois nœuds est ramené à un triangle dont les sommets sont les trois points les plus distants géographiquement.

#### ***k. Zones dynamiques***

Une zone possède des propriétés dynamiques et peut être mise à jour en fonction des perceptions de l'environnement et des changements survenant sur les offres et demandes des usagers du service. Par conséquent, l'ensemble des individus au sein d'une même zone peut évoluer d'un instant à un autre : diminuer s'il y a annulation par

un ou plusieurs usagers du service ou augmenter si de nouvelles offres ou demandes surviennent dans la zone considérée. Aussi de nouvelles zones peuvent-elles être rajoutées si les informations nouvellement reçues font partie d'un terrain non encore couvert par le processus de décomposition.

### *III.3.2.3. Principe de décomposition*

Le procédé adopté pour réaliser la subdivision du réseau géographique de desserte en un ensemble de zones distinctes se dissocie principalement en deux étapes. Cette distinction aboutit à la création de zones plus ou moins différentes, libellées respectivement comme étant Zones Primaires (ZP) ou Zones Intermédiaires (ZI). L'union de celles-ci constitue l'ensemble total de zones :  $Z = ZP \cup ZI$ . Deux algorithmes ont été développés dans ce sens, implémentant les deux étapes, et sont détaillés dans ce qui suit.

#### *a. Étape 1 : Création des Zones Primaires (ZP)*

Dans cette première étape, des zones dites primaires sont créées sur la base des demandes de covoiturage uniquement. Des groupes de voisins sont ainsi déterminés en fonction des adresses relatives aux lieux de départ et d'arrivée des covoitureurs émetteurs de requêtes. Faisant référence au concept de clustering, il s'agit ici d'effectuer une classification non supervisée de l'ensemble des observations acquises initialement. Cette classification aboutit à un ensemble de classes dites « zones primaires » établies sur les ensembles de voisins définis sur la base du diamètre préfixé au préalable. Ces zones sont appelées Zones Primaires parce qu'elles sont initiatrices du traitement d'affectation. En effet, nous verrons par la suite (*c.f. chapitre IV*), que les entités logicielles créées sur les ZP sont les premières à être lancées automatiquement au début du processus de gestion des requêtes. Les Zones Primaires sont, en effet, les seules à contenir des origines de requêtes de covoiturage et sont donc seules concernées par les traitements délocalisés et distribués uniquement sur celles-ci en début d'exécution du système. Grâce à cette distinction, le lancement de traitements et donc de processus supplémentaires sur des zones non sollicitées -au moins dans le premier stade de départ d'exécution du système et probablement durant tout le processus- est évité permettant de gagner en ressources mais aussi en temps de traitement.

L'Algorithme 1 décrit les étapes logicielles suivies au niveau de notre système pour la création des Zones Primaires. Ces étapes font plutôt référence à un ensemble d'itérations durant lesquelles les ensembles de voisins sont identifiés et leurs isobarycentres respectifs

calculés ; sur la base de quoi, les zones sont établies. Un isobarycentre correspond au centre des zones identifiées et qui sont construites en fonction de ce dernier et du diamètre préfixé. Les zones ainsi établies peuvent être différentes pour chaque nouvelle exécution (i.e. relancement sur les mêmes données brutes) de l'Algorithme1, même s'il s'agit de traiter le même ensemble d'informations. Le procédé qui y est décrit est, effectivement, basé sur le concept de tirage aléatoire des nœuds : première étape dans l'établissement des Nœuds voisins ( $N(t)$ ). Ce procédé de choix de nœuds tirés aléatoirement laissé au gré du hasard et du système peut jouer un rôle crucial pour déterminer l'image du réseau distribué à établir afin de pouvoir déterminer les entités logicielles responsables du traitement des requêtes des usagers. Ceci peut aussi avoir une répercussion sur la distribution des tâches sur l'ensemble des entités identifiées de manières analogues aux zones érigées.

Algorithme 1 : Création des Zones Primaires (CZP)

**Données :**  $SN = N_R(t)$  : ensemble des nœuds origines et destinations des requêtes, Diamètre

**Résultat :**  $ZP$  : Ensemble des Zones Primaires

**Initialisation :** Nombre  $k = 0$ ,  $Distance = 0$ ,  $VN$  : Ensemble des nœuds visités =  $\phi$ ,  $N$  : Tableau d'ensembles de voisins =  $\phi$

1. **Tant que**  $VN \neq SN$  **Faire**
2.     Sélection aléatoire d'un nœud  $N_a$  de  $\{SN - VN\}$
3.      $N[k] \leftarrow N[k] \cup \{N_a\}$
4.      $VN \leftarrow VN \cup \{N_a\}$
5.     **Pour Tout** nœud  $N_1$  **tel que**  $N_1 \in \{SN - VN\}$  **Faire**
6.         **Pour Tout** nœud  $N_2$  **tel que**  $N_2 \in N[k]$  **Faire**
7.              $Distance = Distance(N_1, N_2)$
8.             **Si**  $Distance \leq Diamètre$  **Alors**
9.                  $N[k] \leftarrow N[k] \cup \{N_1\}$

```

10.           $VN \leftarrow VN \cup \{N_1\}$ 

11.          Fin Si

12.          Fin Pour

13.          Fin Pour

14.           $k \leftarrow k + 1$ 

15. Fin Tant que

16. Pour Tout nombre  $i$  tel que  $i \in \{0..k\}$  Faire

17.           $C_i \leftarrow Centre(N[i])$ 

18.           $ZP_i \leftarrow Etablir\_Zone\_Primaire(C_i, Diamètre)$ 

19.           $ZP \leftarrow ZP \cup \{ZP_i\}$ 

20. Fin Pour

21. Retourner  $ZP$ 

```

Comme le montre l'Algorithme 1, même si nous nous focalisons dans cette étape sur les composantes informationnelles initiatrices d'un processus de traitement dans notre système, à savoir les origines des requêtes ; nous considérons tout aussi bien les destinations de celles-ci au niveau de cette étape. Ceci afin de mieux travailler la dispersion des zones en vue d'en ressortir le caractère de forte densité et plus particulièrement éviter un fort degré de chevauchement entre elles en considérant un maximum de quantité informationnelle. Le procédé suivi au niveau de cette étape se base sur le fait de faire ériger les ensembles de nœuds proches (i.e. voisins) dans le graphe espace temps construit préalablement de manière à ne prendre en compte que les données sur les piétons. Pour ce faire, un nœud est tiré aléatoirement de l'ensemble des nœuds sélectionnés pour le traitement ( $SN$ ), un nouvel ensemble de voisins est ainsi construit ( $N[k]$ ) où seront stockés le nœud sélectionné ainsi que tous les nœuds qui lui sont proches et qui sont aussi proches deux à deux en termes de distance (i.e. Diamètre). Chaque nœud faisant partie de l'ensemble des voisins ainsi défini est par la suite stocké dans un ensemble de nœuds visités ( $VN$ ) pour éviter son affectation et traitement multiple. Tous les nœuds ayant été traités, un centre est calculé pour chaque ensemble de voisins selon le procédé décrit précédemment ( $Centre(N[i])$ ) et une Zone

Primaire ( $ZP_i$ ) est construite en fonction de ce dernier et du diamètre fourni en paramètre.  $Etablir\_Zone\_Primaire(C_i, Diamètre)$  vient ainsi délimiter le périmètre de la zone couvrant les nœuds faisant partie de l'ensemble de voisins ainsi que les possibilités d'affectation de nouveaux nœuds.

**b. Étape 2 : Création des Zones Intermédiaires (ZI)**

Dans la deuxième étape de subdivision du réseau, toute information (*nœud*) n'ayant pas été traitée dans la première est prise en compte. Ces informations sont considérées dans le but d'en chercher l'affectation aux zones déjà établies ou en créer de nouvelles sur la base du même principe de voisinage. Ces zones sont dites zones secondaire ou Zones Intermédiaires (*ZI*) puisque n'entrant dans le processus d'affectation que si besoin est. L'ensemble global *Z* rassemble toutes les zones créées qu'elles soient primaires ou secondaires et qui sont susceptibles d'être interrogées, mises à jour ou peut être même supprimées suivant le flux informationnel reçu. Ceci dans une optique d'optimalité des traitements, évitant la redondance des informations et/ou traitement tout en couvrant l'espace de recherche considéré.

Algorithme 2 : Création des Zones Intermédiaires (CZI)

**Données :**  $SN = N_o(t) \cup N_{ID}(t)$  : ensemble des nœuds origines, destinations et points de passage des véhicules, Diamètre, *N* : Tableau des ensembles de voisins

**Résultat :** *ZI* : Ensemble des Zones Intermédiaires

**Initialisation :** Nombre  $f, g = Card(N) - 1$ , Distance = 0, *VN* : Ensemble des nœuds visités =  $\phi$ , Booléen *Affecté* = *Faux*

1. **Tant que**  $VN \neq SN$  **Faire**
2.     Sélection aléatoire d'un nœud  $N_a$  de  $\{SN - VN\}$
3.      $VN \leftarrow VN \cup \{N_a\}$
4.     *Affecté*  $\leftarrow$  *Faux*
5.     **Pour Tout** nombre  $x$  **tel que**  $x \in \{0..g\}$  **Faire**
6.         **Si**  $x \leq f$  **Alors**

```

7.           $C_x \leftarrow Centre(Z_x)$ 

8.           $Distance \leftarrow Distance(N_a, C_x)$ 

9.          Sinon

10.          $Distance \leftarrow Plus\_Grande\_Distance(N_a, N[x])$ 

11.         Fin Si

12.         Si  $Distance \leq Diamètre$  Alors

13.          $N[x] \leftarrow N[x] \cup \{N_a\}$ 

14.          $Affecté \leftarrow Vrai$ 

15.         Fin Si

16.         Fin Pour

17.         Si  $Affecté = Faux$  Alors

18.          $g \leftarrow g + 1$ 

19.          $N[g] \leftarrow N[g] \cup \{N_a\}$ 

20.         Fin Si

21. Fin Tant que

22. Pour Tout  $i$  tel que  $i \in \{f..g\}$  Faire

23.      $C_i \leftarrow Centre(N[i])$ 

24.      $ZI_i \leftarrow Etablir\_Zone\_Intermédiaire(C_i, Diamètre)$ 

25.      $ZI \leftarrow ZI \cup \{ZI_i\}$ 

26. Fin Pour

27. Renvoyer  $ZI$ 

```

Le principe suivi dans cet algorithme est le meme que celui décrit dans la première étape (Algorithme 1). La seule différence réside dans le fait de chercher s'il existe une affectation possible de chaque nœud sélectionné aléatoirement à l'ensemble des zones déjà établies ou aux groupes de voisins en cours de construction. En effet, dans cet algorithme, l'établissement

de zones intermédiaires de par le rassemblement de nœuds voisins en groupes est fait au fur et à mesure. La recherche de possibilités d'affectation est ainsi faite en parallèle à ce procédé et implique éventuellement de considérer la probabilité d'appartenance d'un nœud à un ensemble de voisins en train d'être formé. Un nœud sélectionné est ainsi identifié comme en faisant partie si la distance qui le sépare du nœud de l'ensemble qui en est le plus éloigné (*Plus\_Grande\_Distance*( $N_a, N[x]$ )) est inférieure au diamètre.

### III.3.2.5. Un principe de décomposition dynamique

Pour revenir à la notion de rapprochement et création de clusters (nuages d'individus) ; il s'agit dans notre travail de permettre une classification des données nouvellement perçues lorsque le processus est en cours et de les inclure dans la représentation déjà établie. Ceci vient rappeler les méthodes de « *classification supervisées* »<sup>35</sup> qui, partant d'un ensemble de classes existantes, doivent inclure les nouvelles observations. Par observations, nous entendons les changements perçus au niveau de l'environnement de covoiturage et les connaissances nouvellement acquises qui sont directement prises en compte au niveau de l'ensemble des classes (i.e. zones) déjà créées. Ceci se traduit par une mise à jour automatique de l'état du réseau en temps réel.

## III.4. Conclusion

Placés dans le contexte d'un système doté de la propriété primordiale de temps réel, les performances de l'approche que nous proposons doivent impérativement cadrer avec les mesures d'efficacité de tels systèmes tout en satisfaisant les contraintes imposées par l'aspect de forte dynamicité d'un environnement aussi inconstant que celui des services de covoiturage : image d'un réseau de desserte représentant une carte réelle sur laquelle sont placés les voitures et les usagers dont les offres et demandes de déplacement varient au gré de leurs besoins.

Partant de ce principe de base, nous nous sommes concentrés dans ce chapitre sur la modélisation formelle du problème de covoiturage en temps réel de sorte à en suivre l'évolution à chaque instant au fur et à mesure que de nouveaux changements surviennent et qui seront bien évidemment instantanément pris en compte pour une mise à jour automatique du modèle établi et ce afin d'en optimiser le traitement dans un processus ultérieur.

---

<sup>35</sup> [http://docs.happycoders.org/orgadoc/artificial\\_intelligence/classification\\_documents/classification.pdf](http://docs.happycoders.org/orgadoc/artificial_intelligence/classification_documents/classification.pdf)

La modélisation en question se présente sous forme de graphe dynamique distribué dont l'image sera reflétée sur le processus de traitement des requêtes : traitement distribué parallèle en vue d'en réduire la complexité. Pour ce faire, un procédé de décomposition du réseau de desserte géographique en un ensemble de zones de surface limitée a été adopté. Sur les aires géographiques ainsi construites seront créés des processus indépendants ou semi-indépendants correspondant à des tâches considérablement moins complexes et menés en parallèle. Ces tâches seront exécutées en parallèle et de manière distribuée de sorte à optimiser les temps de réponse et l'efficacité de notre système.

Parmi les approches adoptées dans les travaux relatés dans la littérature, le concept multi-agent se trouve être parfaitement adapté à notre système. En effet, les systèmes multi-agent sont connus pour leur efficacité dans les environnements distribués et inconstants. Dans le chapitre suivant, le détail de l'architecture multi-agent établie au niveau de notre système est fourni avec l'ensemble des algorithmes élaborés au niveau de notre approche pour la réalisation d'une affectation dynamique distribuée optimisée des véhicules aux usagers.

## IV. CHAPITRE IV

# CODAC : UNE PLATEFORME DE COVOITURAGE DYNAMIQUE OPTIMISE BASEE SUR LES AGENTS COMMUNICANTS

## IV.1. Introduction

Etant placés dans un environnement non stable et fortement évolutif, la mise en place d'une modélisation appropriée qui cadre bien avec son degré de dynamique s'impose. En effet, la forte variabilité des contraintes, données et paramètres de l'environnement s'oppose à la mise en place d'un service efficient de covoiturage dynamique optimisé et performant. Les données du système de covoiturage sont éparpillées sur un réseau géographique d'une étendue assez large. La distribution de ces données est fortement aléatoire et inconstante. Elle nécessite de ce fait une modélisation qui en suit le mouvement et s'imprime de leur variabilité afin de représenter efficacement et en temps réel l'évolutivité de l'environnement qui les incorpore.

Le besoin de la mise en place d'un procédé efficace qui se marie avec l'architecture « *physique* » proposée du terrain est d'autant plus imminent qu'il contribue à l'efficacité des traitements tout en suivant l'évolution de l'image du réseau de covoiturage. Cette contribution réside essentiellement dans le fait d'établir un processus d'optimisation des requêtes décentralisé et délocalisé au niveau de plusieurs entités logicielles travaillant en parallèle. Ces entités, que nous plaçons dans un contexte de Systèmes Multi-Agents, sont assimilées à des agents logiciels « *distribués* » de manière similaire à la modélisation graphique proposée dans le précédent chapitre. Ils ont ainsi pour rôle principal de mener à bien un processus d'affectation optimisée et efficace des données du système à la manière dont elles sont distribuées sur le réseau géographique de desserte.

La mise en place d'un processus d'optimisation distribuée de recherche et de composition d'itinéraires pour les usagers du service découle ainsi de l'inconstance et l'accroissement continu des sources d'information distantes et distribuées sur les réseaux. Cette stratégie de résolution est d'autant plus efficace que s'il s'agit de systèmes déployés à grandes échelles impliquant des réseaux géographiques de grande étendue pour assurer même le transfert « inter-pays ». Les systèmes d'information distribués sont directement impliqués par ce problème d'optimisation qui nécessite le choix et la recherche de voitures offrant des trajets en adéquation avec ceux demandés. La diversité de la flotte de véhicules avec des itinéraires et coûts différents, éventuellement en concurrence sur un même trajet, implique une grande variété de choix et donc un espace de recherche combinatoire. Une solution optimisée, voire optimale, parmi ces choix doit être trouvée et par la suite affectée à l'utilisateur considéré.

Par ailleurs, les requêtes utilisateurs étant probablement nombreuses et simultanées, et comme nous nous proposons de les traiter en parallèle ; ceci peut être une source de conflit par rapport à l'affectation des véhicules aux usagers et un point d'interrogation par rapport à l'optimalité des solutions générées. La nécessité d'un compromis satisfaisant tous les usagers tout en tenant compte de l'ensemble des contraintes de correspondance précédemment énoncées rajoute encore plus à la complexité du problème traité. Par conséquent, le système que nous proposons doit faire preuve d'efficacité non seulement par rapport à la qualité des traitements effectués mais aussi par rapport à la « légèreté » des processus en dépit de la complexité exponentielle du problème de covoiturage dynamique.

Pour ce problème, nous proposons alors une solution à trois niveaux d'optimisation, dissimulée dans un système d'information multi-agent ouvert et dynamique. Ces propriétés sont intégrées dans le sens où toute donnée parvenant au système doit être prise en compte immédiatement et traitée de manière instantanée. En plus, le système doit aussi être doté d'une facilité d'accès non seulement à ses abonnés mais aussi à d'autres systèmes d'information. L'optimisation réside en premier lieu dans l'étude de l'espace de recherche disponible, scruté en vue d'en extraire des solutions d'affectation optimales par rapport à une fonction objectif globale et adaptées aux besoins des utilisateurs tout en étant faisables par rapport à un ensemble de contraintes. L'accent est mis en second lieu sur la gestion des flux entre les agents actifs, mettant en place des canaux de communication où seuls les agents impliqués sont considérés. En effet, il est vrai que le système proposé se base essentiellement sur une plateforme communicante faisant intervenir des agents en interaction pour mener à bien une tâche un tant soit peu importante. Cependant, une communication n'est établie entre deux agents ou plus que si le traitement dévie obligatoirement vers une collaboration ; ce qui requière des interactions moyennant des échanges de messages qui peuvent engendrer des coûts supplémentaires. D'où la nécessité de contrecarrer ce problème en optimisant les flux d'échange et demeurer dans les mesures d'efficacité requises. Le troisième niveau d'optimisation découle des deux premiers où une optimisation de requêtes nécessite parfois une décomposition de celles-ci et donc la décentralisation du processus de traitement pour une collaboration au sein d'une communauté d'entités logicielles impliquant la notion de coalition d'agents.

La suite de ce chapitre est organisée comme suit : l'architecture du système proposé est introduite dans la deuxième section. Les détails d'implantation du concept multi-agent dans notre approche sont, par la suite, fournis au niveau des trois sections la succédant directement

(§IV.3, §IV.4 et §IV.5). Tout le procédé suivi dès la réception de la première requête de covoiturage jusqu'à la génération des solutions d'affectation optimisée est détaillé à travers ces sections. Par la suite, le problème d'optimisation des affectations considéré est décrit et formalisé dans les sections IV.6 et IV.7. Deux procédés ont été proposés pour approcher le problème de covoiturage dynamique optimisé et sont présentés au niveau de cette section. Finalement, la section IV.8 vient résumer et conclure ce chapitre.

## **IV.2. Le paradigme agent à la base d'une décentralisation**

De la phase d'analyse jusqu'à la conception et l'implémentation d'un système, il est impératif de suivre un procédé efficace pour la bonne mise en œuvre de l'application en question. C'est dans ce contexte que nous nous proposons, à ce stade de description de nos travaux et compte tenu du principe de subdivision adopté, d'établir l'analyse fonctionnelle ainsi que l'architecture du système. Une modélisation de la structure décrivant l'ensemble des entités et objets pouvant intervenir à un instant  $t$  donné, précédée d'un descriptif des fonctionnalités de l'approche proposée, fait entre autre l'objet de ce chapitre. Ces étapes de modélisation et conception seront décrites avec une connotation « *Agent* ».

### **IV.2.1. SMA : pour une stratégie de résolution efficace**

Les Systèmes Multi-Agents, tels que décrits précédemment (*c.f.* § II.4), sont très propices à la mise en place d'un service doté des mesures d'efficacité ainsi que des performances d'exécution recherchées. Adopté au niveau de notre approche, le paradigme agent vient contrecarrer le problème de la complexité exponentielle (*c.f.* § II.3.2). Un choix qui s'avère être judicieux vu la décision portant sur une stratégie de résolution dont la décomposition et la décentralisation font le fondement. Ceci est à l'origine de la mise en place d'un procédé de traitement distribué faisant intervenir un contexte de « *multi-threading* » : un contexte auquel les SMA se prêtent fort adéquatement. Ces derniers sont, en effet, favorables à la mise en œuvre de processus distribués parallèles exécutant des tâches de moindre complexité. En plus, la modélisation adoptée n'est pas sans y contribuer. Celle-ci met effectivement l'accent sur une distribution physique des données du système et leur groupement en plusieurs classes distinctes éparpillées de manière difforme sur le réseau géographique desservi.

## IV.2.2. Graphe Dynamique Distribué : Une modélisation au service de la distribution logicielle

Compte tenu du formalisme précédemment présenté, le concept multi-agent est favorable à la mise en place d'une architecture dynamique distribuée. En effet, la modélisation du réseau géographique sous forme de Graphe Dynamique Distribué est propice au déploiement d'une architecture logicielle composée d'une multitude d'entités autonomes dont les fonctionnements sont menés en parallèle. Ce concept se trouve être en faveur de la décomposition du processus initial. Une décomposition qui donne lieu à un nombre fini de tâches exécutées en parallèle et qui favorise, par conséquent, l'allègement des traitements, la réduction de la complexité du problème et permet donc un gain de temps. Ces tâches sont dispatchées sur l'ensemble des entités en question : composantes logicielles responsables chacune d'exécuter un rôle bien défini en fonction de ses perceptions de l'environnement ainsi que des flux d'échange avec les autres entités du système.

Ainsi, dans le but d'établir un traitement optimisé efficace et léger en termes de complexité, nous nous proposons de mettre en place un système multi-agent. Ce système baptisé CODAC pour Covoiturage Optimisé Dynamique basé sur les Agents Communicants est pourvu d'une multitude de fonctionnalités impliquées dans le processus d'optimisation des affectations des véhicules aux usagers. Parmi ces fonctionnalités, nous retrouvons entre autre le traitement parallèle des requêtes usagers, la décomposition du réseau géographique en vue de mettre en exergue la distribution des données et l'affectation optimisée des véhicules aux usagers. Toutes ces fonctions sont détaillées dans ce qui suit pour une analyse fonctionnelle du problème de covoiturage dynamique vu sous un angle d'optimisation et de décentralisation du procédé d'affectation. Ceci nous mènera par la suite à définir une modélisation adéquate du système et ainsi établir l'architecture correspondante.

## IV.3. CODAC : de l'analyse à la conception

Le problème de covoiturage dynamique a été considéré dans une multitude de travaux relatés dans la littérature (*c.f.* § I.5.2.3). Il s'agit, en effet, d'un problème largement abordé sous plusieurs angles de vue, et ce dans le domaine de l'industrie aussi bien que celui de la recherche. Cependant, malgré sa forte croissance, l'intérêt jusqu'ici porté au problème de covoiturage est resté au stade de « l'ébauche » étant donné le nombre important de projets non aboutis ou insatisfaisants surtout en termes de fonctionnalités. En effet, plusieurs des systèmes

existants présentent de multiples lacunes, notamment par rapport à l'automatisation ainsi que des fonctionnalités mises en place. Ces dernières sont effectivement réduites dans la majorité des cas à la collecte et l'archivage des données sur les covoitureurs et covoiturés ainsi que leurs spécifications de déplacement en vue d'en faciliter l'accès pour une éventuelle consultation. Ceci nous a conduits à porter un œil intéressé sur les fonctions utiles dans un système de covoiturage dynamique optimisé et qui seraient susceptibles d'en alléger les traitements.

### IV.3.1. Problème de Covoiturage Dynamique Optimisé : Analyse Fonctionnelle

Dans une tentative d'amélioration de l'existant, nous arborons une image forte en matière d'automatisme et de fonctionnalités évoluées du système proposé. En effet, comme énoncé précédemment, le principe sur lequel nous nous basons est fondé sur le concept de décomposition pour une meilleure approche du problème. Ce concept implique plusieurs étapes établies, dans un ordre bien défini, dans le but essentiel d'aboutir à la génération de solutions optimisées d'affectation des véhicules aux usagers.

Comme le montre la Figure IV.1, les différentes étapes du processus optimisé de traitement des requêtes des covoiturés débutent par une acquisition et assemblage de celles-ci. Le traitement réalisé par la suite a pour but de parvenir aux différentes possibilités d'affectation de véhicules pour chacun des utilisateurs concernés et le choix d'une solution optimisée parmi celles générées.

Les étapes décrites ici sont établies afin de concrétiser l'idée de l'optimisation au niveau de notre système. Nous faisons référence, dans ce contexte, aux premier et troisième niveaux d'optimisation (*c.f. §IV.1*) relatifs au traitement parallèle des requêtes tout en se basant sur un assemblage de celles-ci afin d'en ressortir les ressemblances. En effet, le principe de décomposition vient, dans ce sens, se mettre au service du traitement parallèle. La génération de solutions « *optimisées* » pour une demande usager peut éventuellement en toucher plusieurs autres présentant des similitudes avec celle-ci et dont les paramètres concordent ; notamment par rapport aux trajets ou partie(s) de trajets voulus, les périodes de voyage, etc.

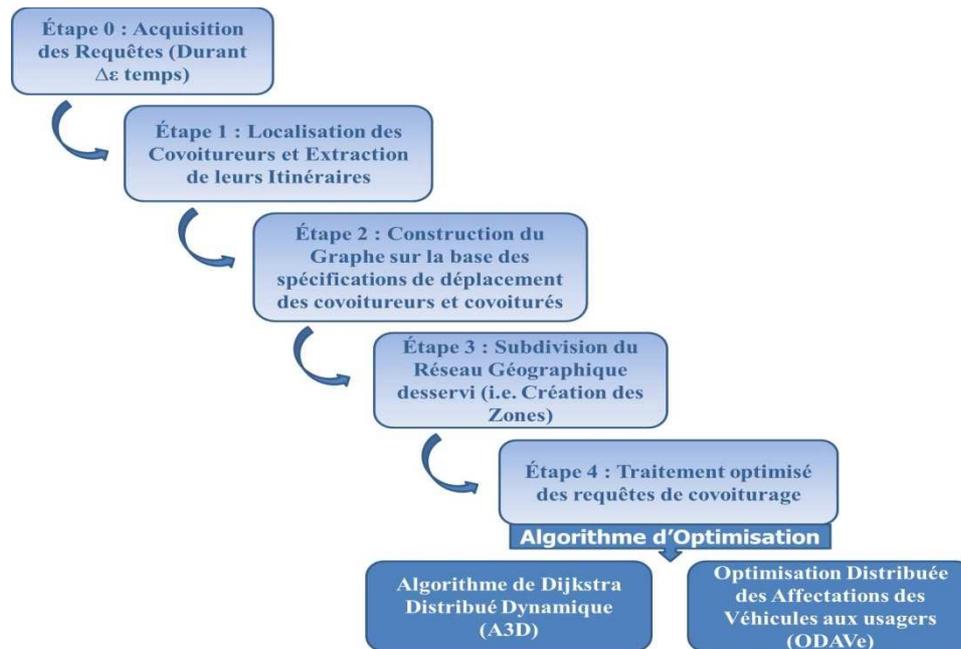


Figure IV.1. Fonctionnalités du Système proposé

Cinq grandes étapes sont ainsi conduites avec pour but final de garantir une qualité de service irréprochable avec un taux de réponses positives maximal. Ceci se traduit par la mise en place d'un support pour la collecte des données brutes disponibles sur le réseau, et leur traitement pour en rechercher les similitudes en passant par leur structuration dans une organisation adéquate (*c.f. Chapitre III*). Sur la base des ressemblances ainsi érigées, la décomposition de l'espace de recherche considéré est concrétisée mettant en relief les regroupements des données dispatchées sur le réseau. Il s'agit là des différentes étapes préliminaires précédant l'application des algorithmes d'affectation des véhicules (i.e. covoitureurs) aux usagers (i.e. covoiturés). Ces algorithmes sont dits d'optimisation distribuée compte tenu de la subdivision réalisée du réseau à l'image de laquelle l'exécution des différentes approches proposées est menée. Ces dernières sont détaillées dans une section ultérieure à la fin de ce chapitre (*c.f. §IV.7*).

Chacune des fonctionnalités considérées au niveau de ces différentes étapes a un but bien déterminé et met en place différentes méthodes pour aboutir à un objectif bien déterminé. Le but visé de par une fonction constitue généralement un produit intermédiaire permettant de construire des états de transition d'une étape à une autre dans le processus global.

Dans ce qui suit, nous présentons des vues plus ou moins détaillées des différentes fonctionnalités de notre système, relatives notamment aux étapes ci-dessus décrites. Cependant, la description que nous faisons de ces fonctionnalités présente un niveau

d'abstraction assez important et sont plutôt présentées à la manière dont elles sont intégrées dans le processus global d'optimisation distribuée du procédé d'affectation. Nous reviendrons sur celles-ci pour en faire une description beaucoup plus détaillée dans une étape ultérieure, notamment lors de la description « *pratique* » du système, à savoir le niveau architectural et logiciel (*c.f.* §IV.5.2).

Par ailleurs, afin d'établir l'analyse fonctionnelle du système que nous nous proposons de mettre en place, nous avons eu recours à l'une des techniques les plus utilisées à cet effet, à savoir les diagrammes FAST. FAST, acronyme de Function Analysis System Technique, est une méthode utilisée lors de la phase d'analyse fonctionnelle pour la spécification des besoins et ce afin de pouvoir aboutir à une modélisation adéquate du système. Elle concerne une méthodologie de représentation des différentes fonctions du système basé sur un graphisme adéquat. Ce dernier a pour finalité de montrer la hiérarchie et la composition de fonctionnalités techniques emboîtées, parallèles ou séquentielles. Les fonctions techniques incombant à une fonctionnalité bien spécifique du système (i.e. fonction de service) sont réalisées à l'aide d'un ensemble d'outils bien déterminés appelés dans ce contexte « *solution constructive* ».

La Figure IV.2 présente une esquisse d'un diagramme FAST où des fonctions techniques composent une fonction service dite principale. Les fonctions de service sont celles qui ont un lien avec le milieu externe. Un diagramme FAST en est donc la traduction en une ou plusieurs fonctions techniques dites internes au produit<sup>36</sup>, puis matériellement en solutions constructives ou encore « solutions technologiques » et qui représentent les éléments du milieu environnant. Ces derniers représentent le moyen de concrétisation de l'objectif visé et sont donc employés afin d'aider à la réalisation des fonctions techniques et donc des fonctions de service.

Un diagramme FAST est construit généralement de gauche à droite, commençant par la spécification de la fonction principale à réaliser pour aboutir à la fin aux moyens concrets mis en place pour aider à la réalisation de celle-ci. Différentes méthodes secondaires de moins en moins complexes sont spécifiées au niveau des étapes intermédiaires dans le passage du service vers les solutions pratiques de réalisation. Par ailleurs, la lecture de ce diagramme peut se faire dans les deux sens ; soit de gauche à droite et dans ce cas elle indique la façon de (i.e. comment) réaliser une fonction principale, soit de droite à gauche pour indiquer le but visé

---

<sup>36</sup> <http://www.jdotec.net>

(i.e. le pourquoi) par l'utilisation ou la réalisation d'une solution technologique ou d'une fonction technique.

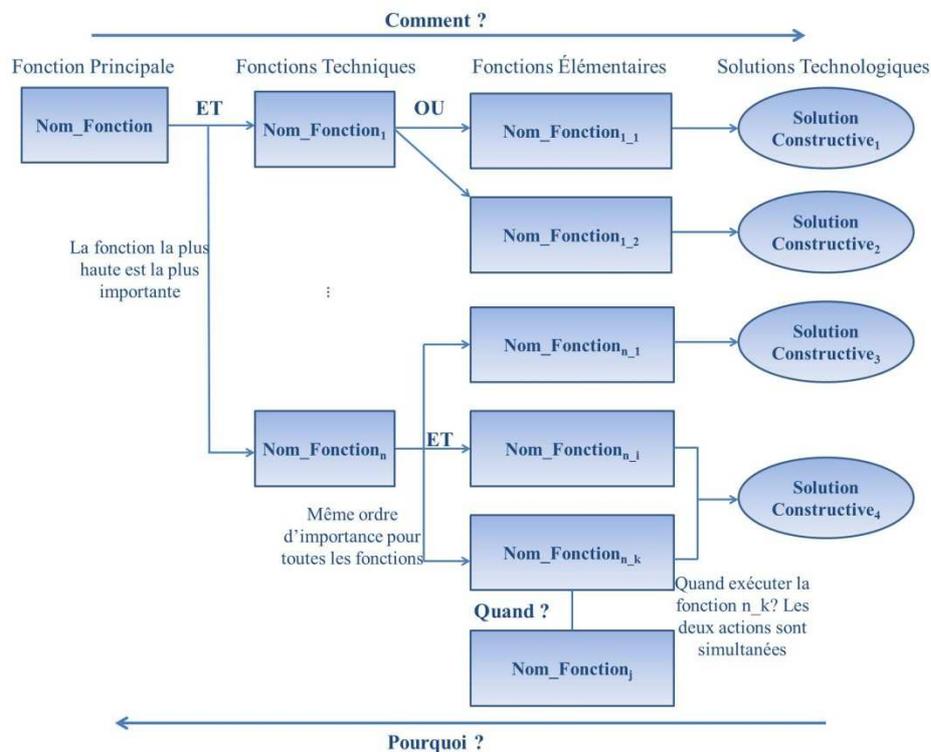


Figure IV.2. Diagramme FAST : Composition de fonctions

Dans ce qui suit, est fournie une description des différentes fonctions principales de l'approche que nous nous proposons de mettre en œuvre. Pour établir cette description, nous nous sommes basés sur les diagrammes FAST.

#### IV.3.1.1. Étape 0 : Acquisition des requêtes

Il s'agit de la première étape du processus de traitement à appliquer au niveau de notre approche. En effet, c'est la toute première phase qui s'exécute lors de l'arrivée d'une requête utilisateur provoquant le déclenchement du système. Le but essentiel que nous visons en premier lieu est de pouvoir traiter les requêtes des usagers en parallèle. Pour cela, il s'agit de pouvoir faire l'acquisition, en même temps ou presque, des spécifications des covoiturés émettant un ensemble de requêtes instantanées et simultanées ou quasi-simultanées.

Nous avons introduit à cet effet un paramètre nommé  $\Delta\epsilon$ , de par lequel l'administrateur peut indiquer un laps de temps négligeable durant lequel le système se mettra en attente d'éventuelles demandes de covoiturage. Le système est ainsi réceptif durant ce temps et se met à l'écoute du réseau pour la détection de nouveaux événements externes pouvant se

produire, et ce avant de pouvoir enclencher le traitement de recherche d'affectations sur la base des données acquises. La valeur fixée pour ce paramètre doit pouvoir garantir l'acquisition des requêtes arrivant en parallèle ou en pseudo-parallèle tout en étant des plus négligeables (e.g.  $\Delta\epsilon = 3$  secondes) de sorte à ne pas affecter les temps de traitement et demeurer dans le cadre des performances d'un système dynamique et instantanément réactif.

Comme le montre la Figure IV.3, deux fonctions pivots régissent le traitement parallèle des données sur les piétons et leurs besoins en matière de déplacement :

- La première concerne l'acquisition des requêtes émises durant le laps de temps indiqué (i.e.  $\Delta\epsilon$ ).
- La seconde est relative à l'organisation de l'ensemble des requêtes reçues et leur structuration suivant une représentation adéquate afin d'en ressortir les similitudes et ainsi favoriser leur traitement parallèle. Il s'agit ici de pouvoir mettre en place la structure de données évoquée, détaillée et formalisée dans le chapitre précédent (c.f. *Chapitre III*) où une représentation matricielle ne peut que convenir à l'organisation visée et ainsi mettre en exergue la composition des demandes des usagers.

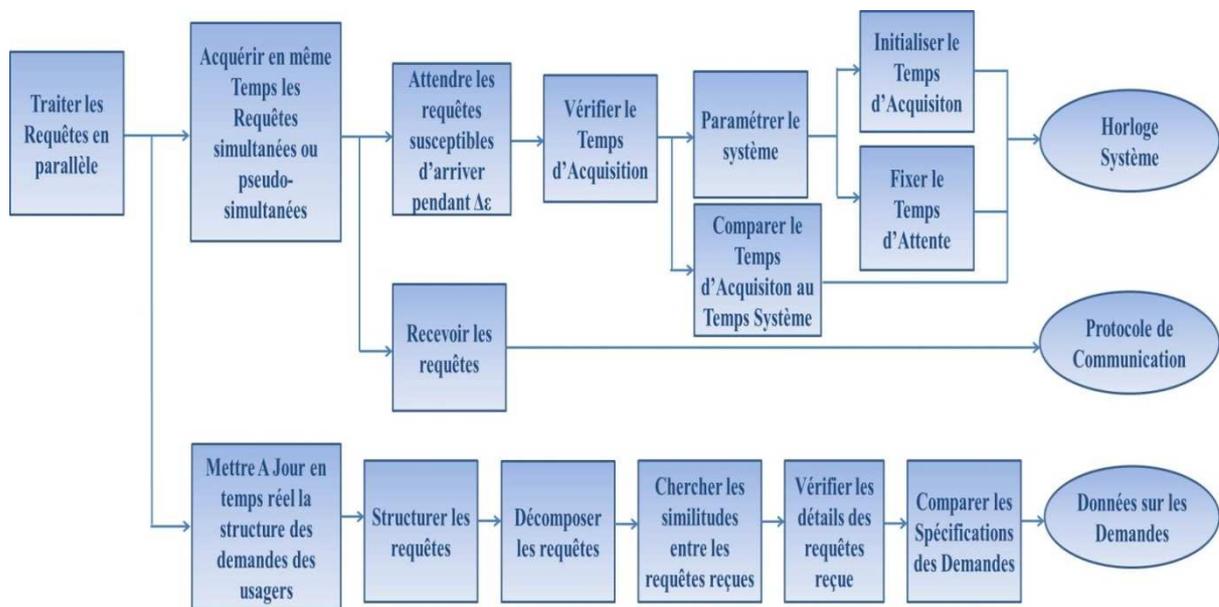


Figure IV.3. Réception et structuration des requêtes quasi-simultanées

#### IV.3.1.2. Étape 1 : Extraction des offres de covoiturage

Le processus de traitement ayant été enclenché par l'arrivée d'une ou plusieurs requêtes usagers, le système doit pouvoir assurer l'affectation de voitures aux piétons demandeurs de

trajets en covoiturage. Pour cela, le système doit disposer des informations nécessaires sur toutes les offres disponibles, relatives aux voitures en circulation sur le réseau. La Figure IV.4 illustre le fonctionnement du système pour l'identification des offres de covoiturage.

Outre le fait d'établir une structure de données pour organiser les offres de covoiturage selon la représentation matricielle détaillée précédemment (*c.f. Chapitre III*), le système doit être capable de récupérer toute nouvelle offre parvenant alors que le processus est déjà en cours d'exécution et l'intégrer dans les traitements. Il est de ce fait impératif de prendre en considération toute information susceptible d'aider dans le traitement des requêtes des covoiturés. Par conséquent, toute offre de trajet prenant effet lors de la réalisation des traitements doit être reçue instantanément et automatiquement prise en compte au niveau de ces traitements. Le système proposé se doit donc de traiter en temps réel les offres de trajets en mettant continuellement à jour l'espace de recherche pour y inclure les spécificités nouvelles sur l'offre considérée. Une mise à jour continue et dynamique a pour but principal d'intégrer toute offre nouvellement perçue afin de pouvoir la prendre en considération dans le processus d'affectation et éventuellement la proposer comme solution possible si c'est le cas.

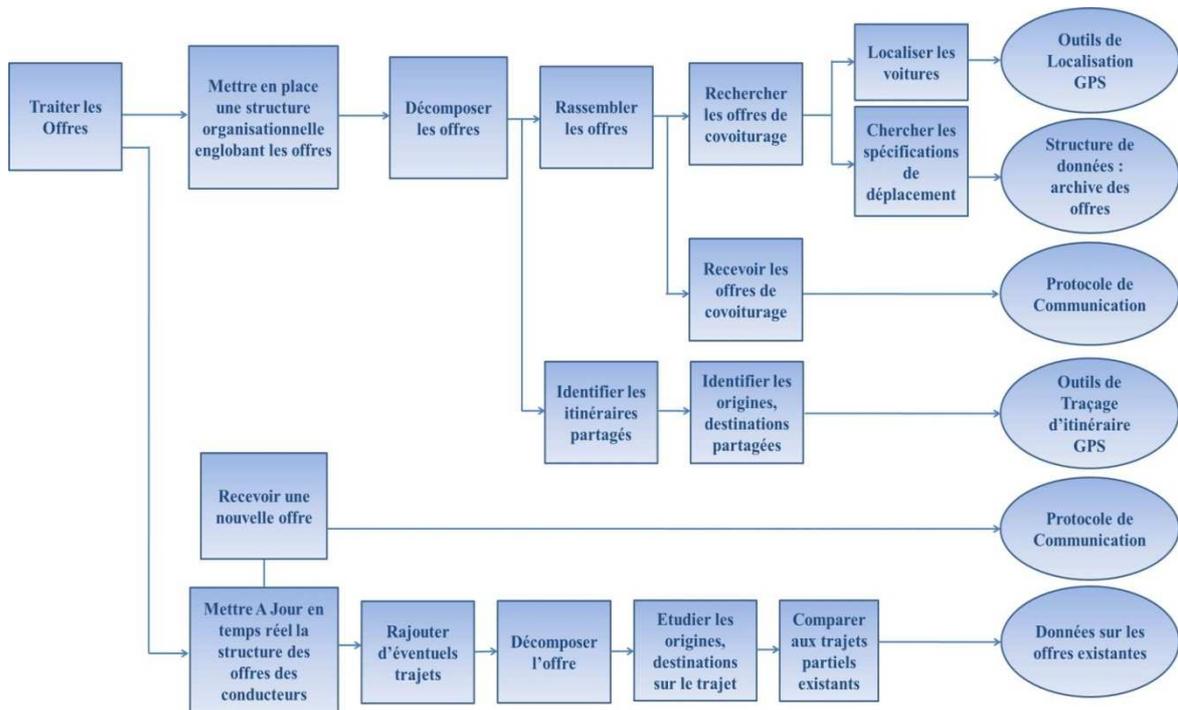


Figure IV.4. Extraction et organisation des offres de covoiturage

La Figure IV.4, où est illustré le fonctionnement du module relatif à l'extraction des offres, présente la mise à jour dynamique comme fonction principale faisant partie intégrante de ce module. Par ailleurs, une première ébauche de la structure de données proposée est

préalablement mise en place dès le lancement du système, et ce sur la base des offres déjà reçues. Ces offres sont relatives à des véhicules ayant déjà entamé leurs trajets respectifs (i.e. voitures en cours de circulation) ou pas encore. Il s'agit dans cette fonctionnalité première de mettre en place une organisation appropriée où les données sur les offres y sont structurées de sorte à en montrer les spécificités communes, notamment et essentiellement les trajets communs. Les tâches relatives au traitement des correspondances entre les offres et les demandes pour la recherche des possibilités d'affectation en sont ainsi facilitées.

#### *IV.3.1.3. Étape 2 : Construction du Graphe*

Compte tenu de la modélisation proposée pour la représentation des données du système, le module ici présenté concerne l'établissement de la structure graphique relative à cette modélisation. Tel qu'illustré au niveau de la Figure IV.5, il s'agit effectivement de mettre en place un graphe dynamique contenant les informations fournies en entrée au système proposé (c.f. *Chapitre III*). Ces informations vont donner lieu aux différents paramètres du graphe à construire, notamment les nœuds représentant les origines, destinations et points de passage aussi bien des véhicules (i.e. covoitureurs) que des passagers (i.e. covoiturés) ainsi que les arcs relatifs aux routes empruntées par les véhicules reliant ainsi les nœuds concernés entre eux.

A partir des deux étapes précédentes, des représentations matricielles ont été établies pour rassembler l'ensemble des demandes des covoiturés ainsi que l'ensemble des offres de covoitureurs. Ces demandes et offres constituent les données transitant au niveau de notre système et dont l'extraction a été réalisée à travers les deux premières étapes décrites au niveau des diagrammes FAST fournis dans les Figures IV.3 et IV.4. Les deux matrices ainsi établies ont donné lieu à l'émergence du concept de graphisme basé sur la théorie des graphes puisque celle-ci se trouve être la plus en adéquation avec le problème de covoiturage dynamique. Nous faisons donc usage de ce concept afin d'illustrer une vue réelle du réseau de desserte. Cette troisième phase en est donc la concrétisation et est responsable de la construction d'un graphe dynamique qui concerne l'illustration graphique des structures organisationnelles résultant des deux premières étapes.

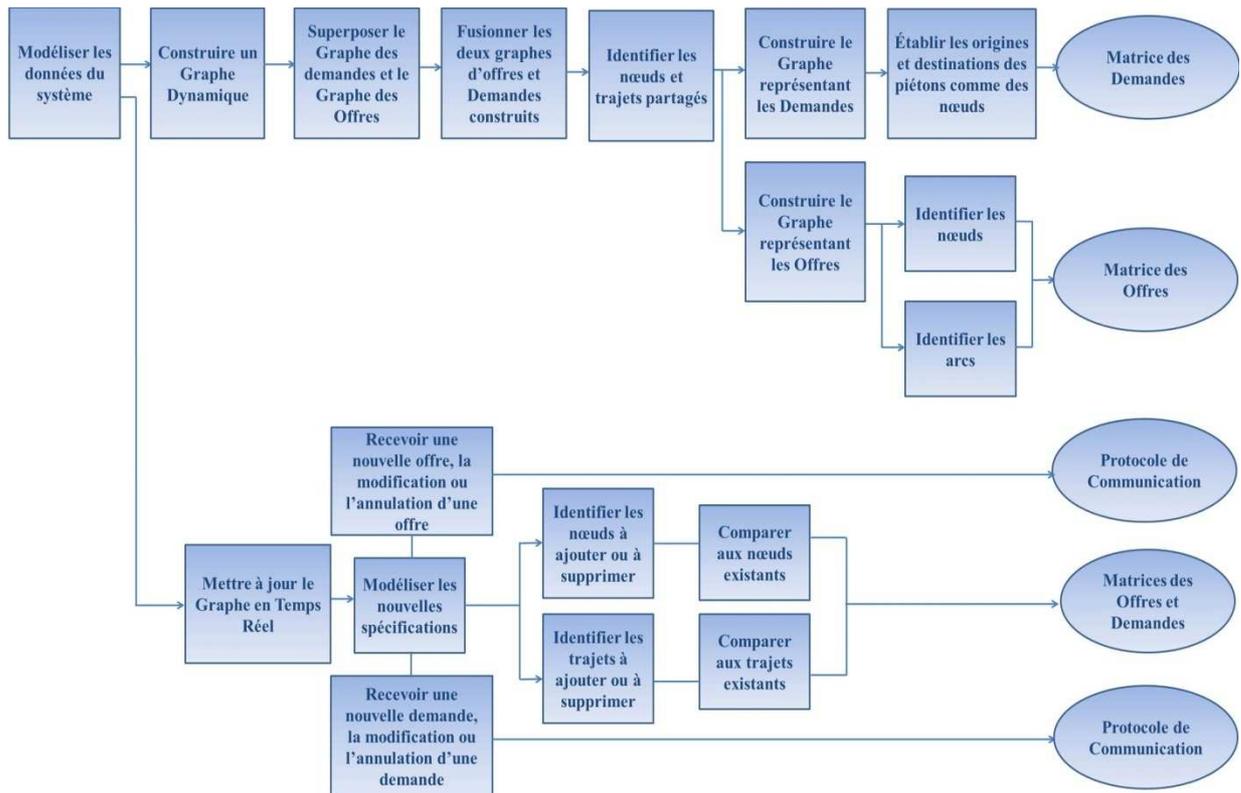


Figure IV.5. Construction du modèle représentation du système

Pareillement aux deux étapes précédentes, deux fonctions principales sont à l'origine de la modélisation du système informationnel sous forme de graphe dynamique :

- La mise en place de la structure graphique sur la base des informations détenues à priori. Cette étape concerne la construction d'un graphe global intégrant toutes les données du système. Ce processus se résume en l'établissement d'un graphe de fusion où est réalisée une superposition de deux plans de l'espace, le premier représentant les demandes des piétons et le second les offres des conducteurs.
- La continuelle mise à jour en temps réel du graphe ainsi établi. D'éventuelles modifications peuvent donc prendre effet dès lors que le système détecte un évènement quelconque parvenant à travers les canaux de communication établis avec ses différents abonnés. L'évènement en question peut être relatif à la réception d'une ou plusieurs demandes aussi bien que d'offre(s) de covoiturage ou encore l'annulation ou la modification d'une donnée existante sur un trajet spécifique. Ces évènements, reprenant l'ensemble des actions pouvant être entreprises par les usagers du système, constituent les principaux facteurs aboutissant à la réalisation du graphe à mettre en place. Si, par ailleurs, l'on se positionne dans une étape ultérieure à celle initiale ; le graphe ayant déjà été établi dans un premier temps subit les modifications nécessaires

pour pouvoir inclure l'information nouvellement perçue. En effet, le diagramme FAST illustré au niveau de la Figure IV.5 montre le processus de déploiement des différentes données d'entrée du système sous forme de graphe. Des modifications prennent effet sur ce dernier dès lors qu'un évènement externe survient, à savoir une annulation ou modification d'une offre ou demande déjà reçue et modélisée ou bien l'arrivée d'une nouvelle requête ou offre de covoiturage. Le système se doit alors de vérifier les paramètres nouvellement identifiés et de les comparer à ceux déjà inclus et représentés au préalable afin de pouvoir décider de l'action à entreprendre :

- Rajouter de nouveaux paramètres (e.g. nouvelles demandes ou offres) engendrant entre autre une modification des propriétés du graphe (e.g. nouveaux nœuds, arcs, etc.),
- Supprimer des paramètres existants devenus inutiles et portant à conflit (annulation, modification, etc.),
- Ou bien ne rien faire dans le cas où les paramètres concernés impliquent d'autres demandes et/ou offres existantes et déjà représentées (e.g. nouvelle demande sur un trajet préalablement sollicité, etc.).

#### *IV.3.1.4. Étape 3 : « Diviser pour régner »*

Le principe même de l'optimisation au niveau de notre approche se base sur la notion de décomposition introduite à des fins d'amélioration du processus d'exécution et ainsi garantir une qualité de service satisfaisante si ce n'est optimale. Nous nous proposons alors de subdiviser le réseau géographique de desserte afin de montrer les zones de densité plus ou moins forte. En effet, l'identification des aires géographiques de concentration d'individus se trouve derrière une décentralisation des traitements globaux par rapport à celles-ci. Le processus de recherche d'affectations optimisées est par conséquent décomposé et délocalisé au niveau des zones extraites et est donc mené à l'image du réseau après sa subdivision.

La Figure IV.6 reprend de ce fait le principe de décomposition énoncé dans le Chapitre III et présente ainsi les différentes étapes ou méthodes à accomplir afin de mener à bien le procédé de décomposition du réseau de desserte, toutefois limité géographiquement pour ne prendre en compte que l'aire incluant les emplacements d'individus dans l'espace ainsi que la totalité de leurs trajets.

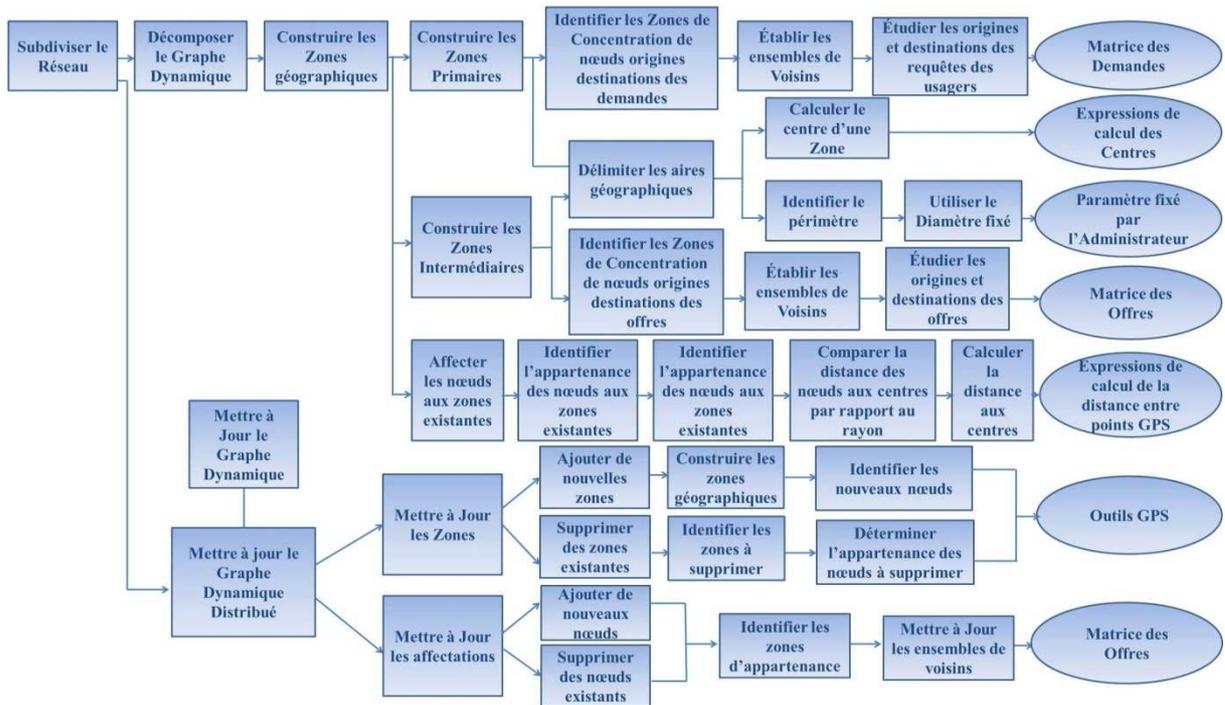


Figure IV.6. Décomposition du réseau

Il s'agit dans cette figure d'une reprise schématique du principe de décomposition détaillé dans une étape antérieure de l'élaboration de nos travaux. En effet, la Figure IV.6 n'est autre que l'illustration graphique des algorithmes de décomposition du réseau géographique de desserte présentés dans le chapitre précédent. Dans ces spécifications formelles, il s'agissait de pouvoir mettre en place une architecture « physique » distribuée pour modéliser le réseau de covoiturage. L'architecture en question est établie sur la base du graphe dynamique existant a priori et est affinée par la suite au fur et à mesure de l'élaboration des traitements et dès qu'une mise à jour quelconque du réseau est détectée. Le graphe représentant le système évolue, effectivement, dans le temps ainsi que l'image décomposée exhibée.

#### IV.3.1.5. Étape 4 : Traitement optimisé des requêtes utilisateurs

Outre la mise en place d'un traitement décentralisé, l'optimisation au niveau de notre système réside en plus dans le fait de considérer des critères d'optimisation. Ces critères sont pris en compte pour l'évaluation, sur la base d'une fonction Fitness, des solutions générées par le processus d'optimisation en premier lieu (i.e. solutions répondant aux contraintes de correspondance). L'évaluation d'une solution sur une même route avec un même véhicule peut ne pas être la même pour deux usagers différents puisque la fonction Fitness utilisée peut différer d'une requête à une autre. Elle est en effet établie en fonction des besoins et préférences des usagers. Leurs indications concernant leurs préférences de déplacement sont à

même d'être utilisées comme paramètres de pondération pour l'agrégation des fonctions élémentaires à optimiser ; à savoir le temps de parcours et le temps d'attente des véhicules. Chaque utilisateur se verra donc attribuer un trajet optimisant une fonction Fitness établie sur la base de ses propres préférences en matière de déplacement. La Figure IV.7 illustre le procédé d'optimisation utilisé au niveau de notre approche.

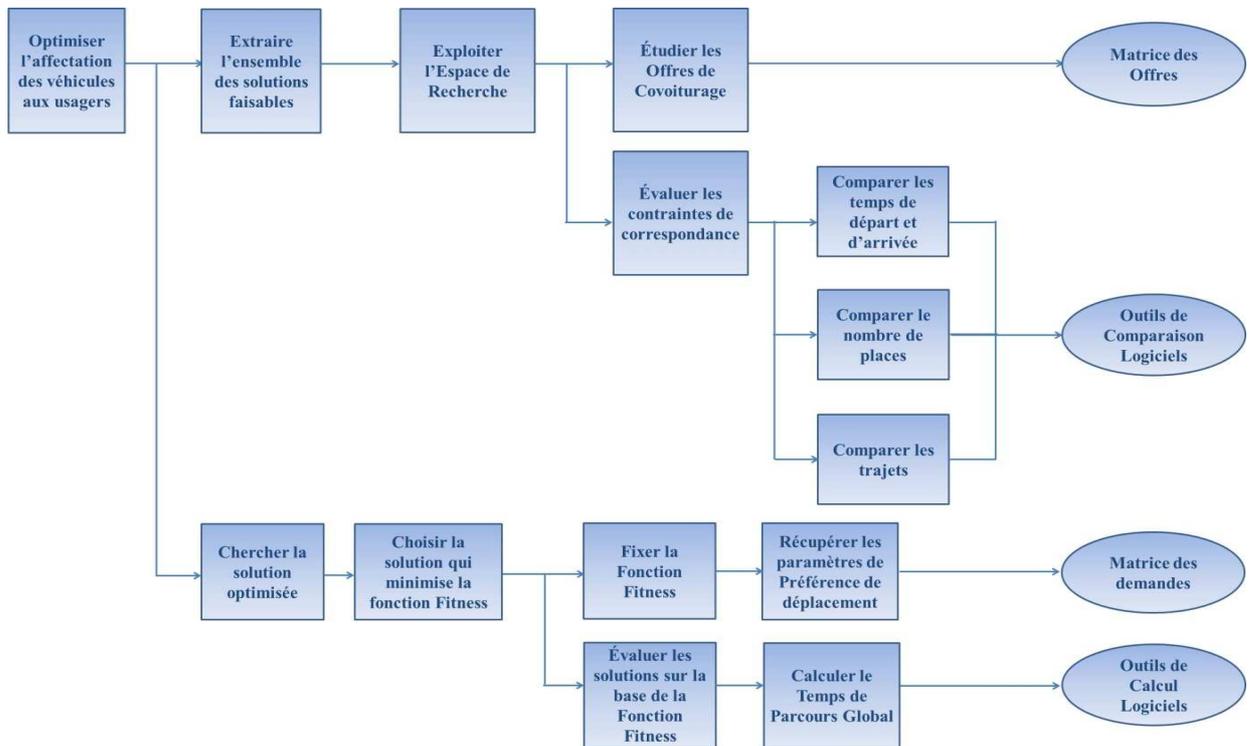


Figure IV.7. Génération de solutions optimisées aux requêtes des usagers

Dans le diagramme FAST présenté au niveau de cette figure, nous définissons les concepts de base liés au traitement optimisé des requêtes des usagers avec un plus ou moins haut degré d'abstraction. Il est de ce fait élaboré de manière à n'en montrer que le procédé adopté pour la génération de solutions optimisées, passant par une première étape où les solutions faisables sont extraites et une seconde pour la recherche de celle optimisant la fonction Fitness. Aucune indication sur les méthodes et algorithmes mis en place n'est ici donnée.

Nos travaux se subdivisent essentiellement en deux approches basées sur des concepts distincts. Deux procédés sont ainsi proposés pour approcher le problème de covoiturage dynamique d'une manière assez différente. Les concepts sur lesquels nous nous basons pour l'élaboration de ces deux approches sont détaillés dans la suite de ce chapitre.

Comme le montre le diagramme FAST illustré dans la Figure IV.7, les approches proposées ne convergent que dans le sens où les solutions générées sont des solutions

faisables minimisant la fonction Fitness fixée. La faisabilité des solutions considérées étant déterminée en fonction de la satisfaction des contraintes de correspondance établies en fonction des exigences des utilisateurs en matière de déplacement.

Chacune des fonctionnalités décrites dans cette section est la raison d'être d'une entité logicielle à laquelle la mise en œuvre de la fonctionnalité en question incombe. L'ensemble des composantes logicielles mises en place dans le cadre d'une architecture multi-agent sont détaillées dans la suite de ce chapitre et sont présentées dans leur contexte applicatif avec les différentes possibilités de coalition pouvant avoir lieu entre elles pour la bonne mise en œuvre du système proposé.

### IV.3.2. Quelle modélisation pour quel objectif ?

Dans nos travaux, nous nous intéressons particulièrement à la technologie *AUML* qui est un langage de modélisation favorisant la conception et la programmation orientées objet aussi bien que la modélisation agent ; passage obligatoire pour la définition d'une architecture logicielle appropriée. Dans ce qui suit, nous procédons à une brève introduction des différents aspects théoriques et pratiques qui viennent se loger dans le cadre d'une conception adéquate. Ceci nous permettra d'introduire l'architecture proposée pour la modélisation de notre système et par la suite détailler l'implantation du concept multi-agent. Les comportements des différentes entités ainsi définies, évoluant dans l'environnement propre au problème de covoiturage dynamique, sont présentés à l'aide de la modélisation ici adoptée.

#### *IV.3.2.1. Des techniques objets au service des agents*

Outre le fait de considérer le paradigme agent, une base objet se trouve être bien adaptée pour être intégrée au niveau de la modélisation du problème de covoiturage dynamique optimisé, et ce pour différentes raisons. Parmi nos principales motivations dans ce contexte :

- La forte croissance que subit le domaine du transport et plus particulièrement le transport comodal ;
- Les systèmes inscrits dans ce contexte montent rapidement à l'échelle et ont besoin d'une modélisation qui suit la forte évolution de l'environnement ;
- La nécessité de mettre en place des structures de données appropriées pour représenter les informations pertinentes et persistantes ;
- La distribution des connaissances ;

- Etc.

A ceci se rajoute le problème de l'optimisation incorporée dans les fonctionnalités du système qui en plus de la nécessité de considérer la dynamique de l'environnement nous a conduits à considérer le paradigme « *Agent* ». Ce choix nous a permis de mettre en place un système distribué où règne un contexte d'intelligence ambiante faisant intervenir l'ensemble des entités dans une ou plusieurs coalitions.

#### *IV.3.2.2. Vers une modélisation Agent*

Tel qu'énoncé précédemment, nous nous focalisons particulièrement dans nos travaux sur le concept « *Agent* » : acteur pivot dans les recherches en intelligence artificielle distribuée et plus généralement dans les sciences de l'Informatique (120) (121). Il s'agit en effet d'un concept émergeant et très intéressant utilisé dans nombre de champs d'application pour la modélisation de sociétés, allant jusqu'aux sciences humaines.

Une conception basée sur les agents peut être vue comme un modèle architectural qui dérive de la modélisation objet détaillée en annexe (*c.f. Annexe I*). En effet, l'architecture agent est le produit du passage du composant objectif vers le composant projectif. Un objet étant défini comme étant un composant passif puisqu'il est pourvu d'un ensemble d'attributs et de méthodes à travers lesquelles il peut être manipulé. Il offre ainsi un ensemble de services et utilise d'autres objets afin de réaliser ses propres fonctionnalités.

De manière quelque peu similaire, un agent interagit d'une façon intelligente avec d'autres entités du système, mettant à son service leurs comportements spécifiques, et ce pour atteindre ses propres objectifs. Parallèlement à l'échange de messages entre objets pour établir les interactions possibles pouvant les relier, les agents communiquent de par l'établissement d'un support pour dialoguer. Ce dernier est mis en place grâce à des outils spécifiques et est élaboré essentiellement moyennant un protocole de communication facilitant ainsi l'envoi et la réception de messages, le traitement collaboratif et l'échange d'information dans un contexte de coalition inter-agents.

Par conséquent, dans l'architecture que nous proposons pour la représentation de notre système, nous ne pouvons faire abstraction de l'un ou l'autre des deux concepts, puisque agents et objets évoluent en parallèle dans le même environnement qui est celui du covoiturage dynamique. Dans le sens de leurs définitions respectives, les premiers « *actifs* » sont seuls responsables des actions à entreprendre pour mener à bien le fonctionnement du

système et font donc usage des deuxièmes, dits « *passifs* ». La section suivante fournit de plus amples détails sur l'architecture proposée.

#### IV.3.2.3. CODAC : Vue sur l'environnement architectural

Au niveau de la Figure IV.8 est illustrée une vue figée de l'architecture dynamique que nous proposons pour modéliser notre approche. Comme le montre cette figure, des objets manipulables de par un système de covoiturage dynamique sont modélisés. Plus concrètement encore, les objets ici définis contribuent entre autre à former le champ d'application des différentes entités intelligentes impliquées dans la mise en œuvre de l'application à élaborer. L'architecture ici proposée réunit de ce fait deux types de composantes essentielles au niveau de notre système. Par conséquent, la structure ainsi définie jongle entre deux concepts qui se rapprochent mais qui restent tout de même très différents :

- a) Le concept « *Objet* » : toute entité classée sous cette catégorie représente une composante logicielle modélisant un aspect du monde réel. Dans le contexte où nous nous plaçons, ces entités se réfèrent à des routes, à des nœuds correspondant aux positions géographiques identifiées, à des itinéraires et chemins à emprunter, à des zones géographiques concrètes modélisant le réseau de desserte, etc. Les nœuds correspondants à des points bien spécifiques de l'espace sont définis essentiellement de par leurs coordonnées GPS et sont associés à des objets. Des instances de l'ensemble de ces entités sont alors manipulables de par les attributs et méthodes qu'elles incorporent. Elles font ainsi une base applicative des agents évoluant au niveau de notre système. Ces derniers intègrent, effectivement, l'ensemble des entités, instances d'objets, à partir desquelles ils perçoivent les changements opérés au niveau de l'environnement et sur lesquels ils peuvent agir. Ceci en vue de mener à bien tout le processus incombant au système de covoiturage proposé et ce de manière coordonnée et synchrone.
- b) Le concept « *Agent* » : il s'agit de composantes intelligentes et autonomes, dotées chacune d'un rôle bien particulier. Ces composantes évoluent en parallèle dans le même environnement et interagissent entre elles de manière à éviter toute sorte de conflit de par leurs agissements parallèles sur les données de l'environnement en question. Elles se partagent ainsi les tâches relatives aux différentes fonctionnalités à exécuter et se coordonnent pour synchroniser leurs actions et aboutir à un fonctionnement cohérent.

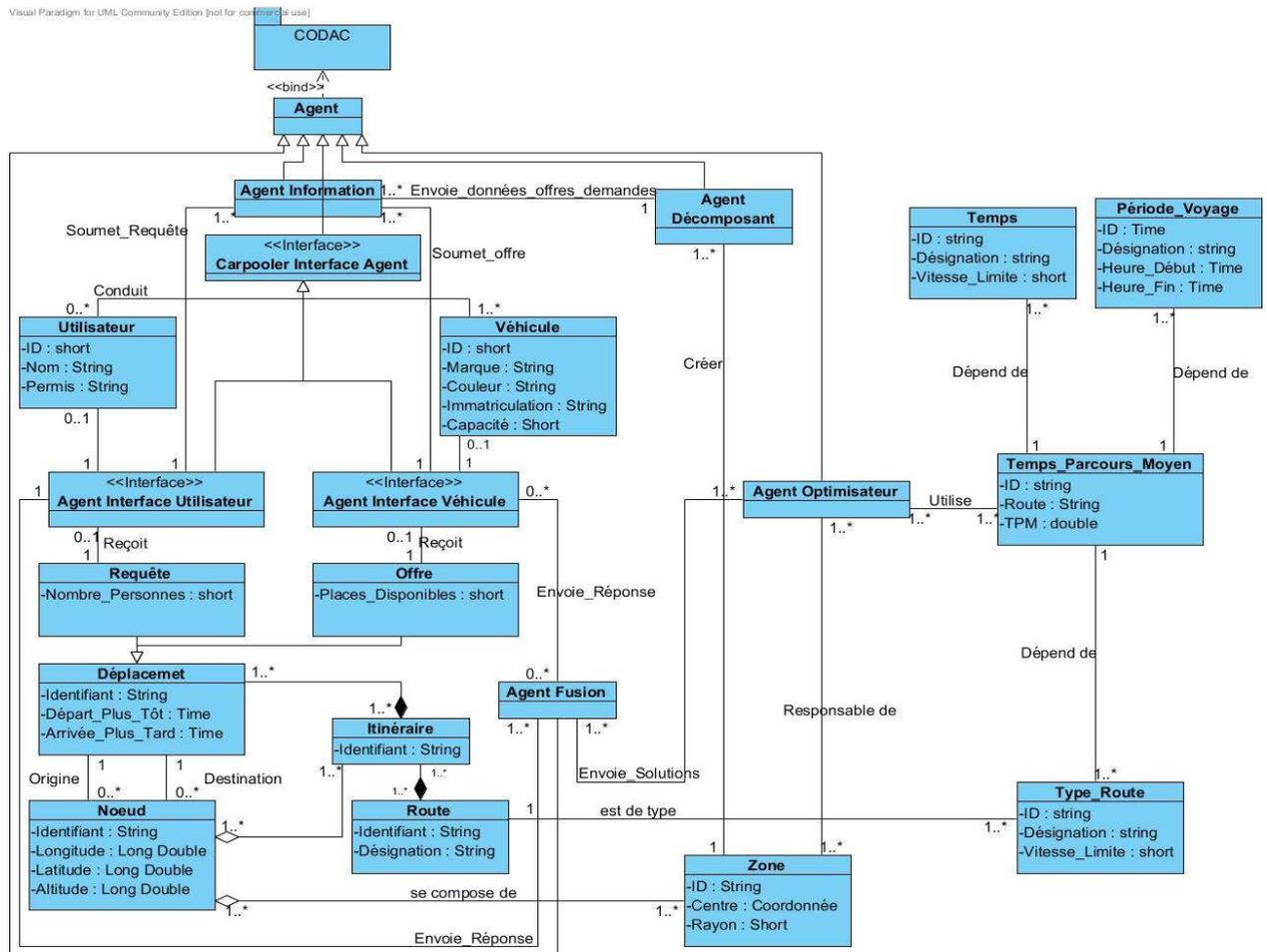


Figure IV.8. Vue sur l'architecture du système CODAC

Dans le reste de ce chapitre, une attention toute particulière sera prêtée au concept multi-agent et son implantation dans notre système.

## IV.4. CODAC : Un contexte d'Intelligence Artificielle Distribuée

Grâce au concept des systèmes multi-agents, il est plus aisé d'approcher un problème difficile dissimulé derrière un programme monolithique d'*Intelligence Artificielle (IA)*. La simplicité et l'aisance proviennent, dans ce cas, de la conception de programmes relativement « légers » exécutés par un ensemble d'agents en interaction. En effet, dans le but d'alléger la complexité d'un problème donné, la décomposition et la décentralisation du programme initial en une multitude de sous programmes, auxquels sont associés des processus parallèles, se prête parfaitement au contexte de l'*Intelligence Artificielle Distribuée (IAD)* (122). Les sous-programmes ainsi érigés sont exécutés par des entités intelligentes qui se synchronisent et coordonnent leurs actions dans un environnement dynamique et incertain.

Il s'agit, en effet, d'entités plus ou moins autonomes dont le fonctionnement varie uniquement en fonction des perceptions qu'elles ont de l'environnement ou des autres entités. Chaque agent est doté d'un contrôle total sur son comportement et d'une autonomie lui permettant de s'adapter dynamiquement aux changements imprévus qui interviennent dans l'environnement. Outre la forte variabilité de l'environnement, l'importante complexité du problème de covoiturage dynamique optimisé vient confirmer le choix de mettre en place un système multi-agent allouant ainsi un intérêt bien particulier à l'optimisation distribuée.

Ainsi placés dans un contexte d'intelligence artificielle distribuée, nous verrons dans ce qui suit la manière dont les fonctionnalités du système proposé ont été dispatchées sur les différentes entités impliquées. Notre principal objectif étant de nous départir du problème de complexité des programmes monolithiques afin de pouvoir demeurer dans des mesures de performances dignes des systèmes s'exécutant en temps réel. Pour cela, une modélisation basée sur les agents est préalablement spécifiée. L'architecture du système précédemment établie (définie au niveau de la Figure IV.8) est ainsi décortiquée afin d'en faire ressortir une sous-structure fondée uniquement sur la base du paradigme agent.

## **IV.5. CODAC : Une architecture multi-agent**

Le problème du covoiturage dynamique optimisé étant de complexité d'ordre exponentiel, nous nous sommes penchés sur cette problématique en particulier pour essayer d'aboutir à un système efficace et performant. Partant du principe fondamental de la division sous l'enseigne du contrôle distribué, nous avons opté pour une décentralisation des tâches sur un ensemble de composantes logicielles. Ces composantes faisant figure d'entités intelligentes autonomes s'exécutant en parallèle collaborent au sein d'un environnement fortement évolutif et instable qu'est le réseau de covoiturage dynamique. Ces entités évoluent ainsi au niveau de l'application que nous nous proposons de mettre en place avec chacune son propre degré d'indépendance. Une architecture multi-agent illustrée de par la Figure IV.9 est ainsi configurée sur la base des entités ayant émergé de la modélisation préalablement élaborée. Cette modélisation a été fondée principalement sur la subdivision des tâches et l'allocation des différents rôles établis à travers la décomposition des fonctionnalités du système.

De ce fait, la modélisation que nous proposons au niveau de cette section est une modélisation agent où l'environnement en continu mouvement est conçu comme un ensemble d'entités plus ou moins indépendantes mais dont l'existence et le fonctionnement

sont étroitement lié à la forte variabilité des données du système. C'est ainsi que nous distinguons :

- Des *Agents Interfaces Véhicules (AIV)* et des *Agents Interfaces Utilisateurs (AIU)* regroupés sous un type particulier d'agents appelés *Agents Interfaces Covoiturage (AIC)*. Ces entités sont une représentation logicielle des usagers du système. Le premier type (*AIV*) est ainsi associé aux conducteurs des véhicules et le second (*AIU*) aux passagers (i.e. piétons) émetteurs de requêtes de covoiturage (c.f. §IV.5.2.1);
- Des *Agents Informations (AI)* pour la collecte des informations sur les demandes et offres de covoiturage émises par les usagers du service et transmises par les *AIC* (c.f. §IV.5.2.2).
- Des *Agents Décomposants (AD)* pour la subdivision du réseau dont le comportement détaillé dans la section VI.5.2.3 reprend le principe de décomposition décrit dans le chapitre III ;
- Des *Agents Optimisateurs (AO)* pour le traitement des requêtes de manière distribuée (c.f. §IV.5.2.4). L'ensemble des entités définies dans ce contexte sont dotées de deux comportements différents selon qu'elles exécutent l'une ou l'autre des approches proposées (c.f. IV.7) ;
- Des *Agents Fusions (AF)* capables de recueillir les solutions générées et de les dispatcher par la suite aux usagers concernés. Leur comportement est détaillé dans la section IV.5.2.5.

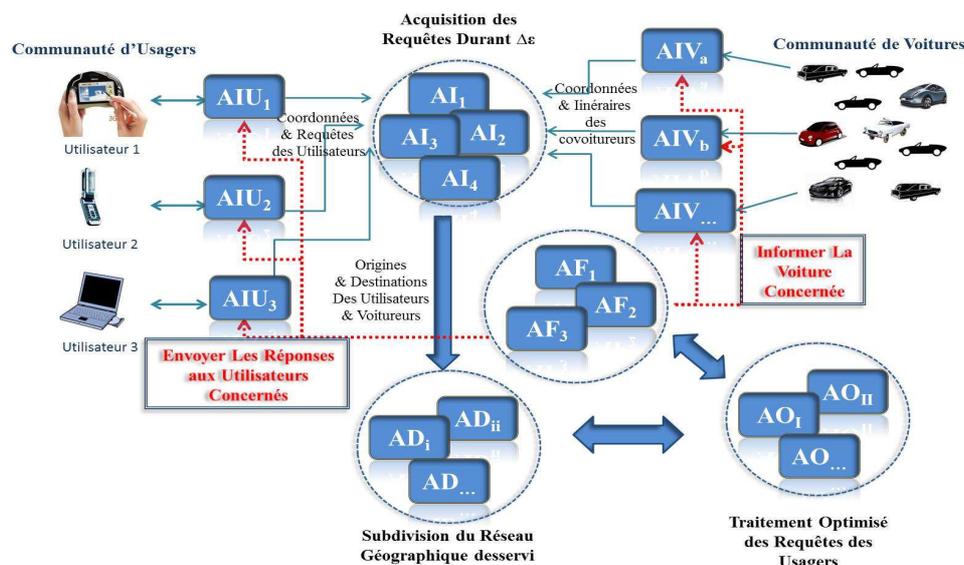


Figure IV.9. CODAC : Un système basé sur les Agents Communicants

Les agents décrits au niveau de cette architecture font l'objet de création (instanciation), probablement multiples, d'un ensemble d'entités relatives à un même type d'agent et évoluant en même temps au sein de l'environnement ici représenté. Cet énoncé vient mettre l'accent sur l'utilité de faire intervenir, pour l'élaboration des traitements, une multitude d'entités, même si elles sont pourvues d'un même comportement, et ce afin de mieux gérer les flux entrants au système aussi bien que ceux qui en sortent. En effet, ayant placé la montée à l'échelle comme étant l'un des objectifs principaux que nous visons, il convient de considérer les situations d'encombrement et de surcharge pouvant survenir suite à la réception d'un nombre assez important de demandes et offres. L'architecture que nous proposons est alors dotée de la propriété de dynamique dans le sens où les agents évoluant dans notre système sont créés et détruits au besoin. La création de nouvelles instances intervient dès lors que des événements externes ou internes le nécessitent. C'est le cas aussi lorsqu'une société d'agents préalablement créée et responsable d'assurer le traitement a atteint son seuil limite de charge en terme de nombre de demandes. Ceci afin d'éviter l'allourdissement des traitements et donc la dégradation des performances. Par ailleurs, dès lors qu'une communauté d'agents, ou des agents qui s'y trouvent, terminent leurs traitements, ils s'endorment attendant de nouveaux événements susceptibles d'arriver et de les réveiller; faute de quoi ils s'autodétruisent.

Par ailleurs, comme le montre la Figure IV.9, des flux de données sont échangés entre les différents agents aidant à la bonne marche du système dans un contexte de synchronisation évident. Chacune des entités impliquées dans ce processus est pourvue d'un rôle bien défini dans le processus global. L'ensemble de ces entités est défini dans la suite, avec un descriptif détaillé de leurs fonctionnements.

### IV.5.1. A chaque agent son espace d'action

Le principe de subdivision « *logicielle* » énoncé de par la mise en place d'un système multi-agent va de pair avec la subdivision « *physique* » annoncée dans le chapitre précédent. Nous faisons référence ici à la modélisation d'une architecture multi-agent dans le premier cas et une architecture physique sous forme de graphe dynamique distribué dans le deuxième. En effet, cette dernière se prête favorablement à une décomposition du processus à l'image des parcelles de réseau ainsi établies. Par conséquent, nous avons concentré nos efforts sur une architecture agent qui mette en exergue le bienfondé d'une telle modélisation (i.e. graphe dynamique distribué).

La Figure IV.10 définit un angle de vue des différents agents précédemment définis et ce compte tenu du principe de subdivision du réseau géographique. Les Agents Décomposants (*AD*), Information (*AI*) et Fusion (*AF*) ont de ce fait une vue quelque peu globale sur la totalité de l'architecture géographique considérée. Ces agents sont en effet concernés par toute donnée circulant dans l'environnement de covoiturage et dont le fonctionnement dépend. Par ailleurs, les Agents Interface Covoiturage (*AIC*) et Optimisateurs (*AO*) ont une portée assez restreinte, limitée aux aires géographiques incluant les usagers concernés.

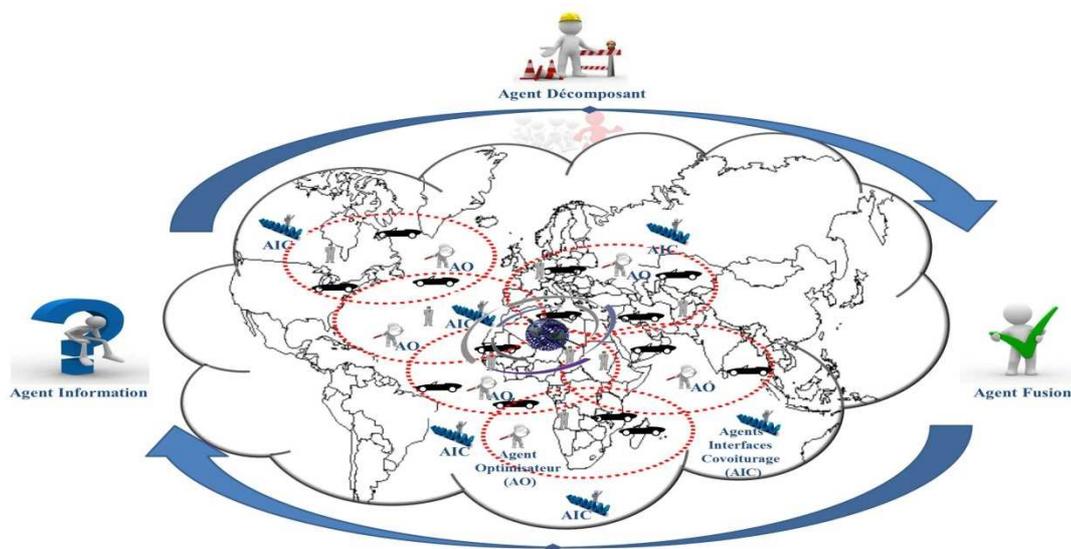


Figure IV.10. Agents : Une portée étendue ou limitée

Les *AO* et *AIC* sont ainsi qualifiés d'agents locaux. Ces agents tiennent cette qualification de leur emplacement sur le réseau et leur appartenance aux zones géographiques. Etant relatifs aux utilisateurs situés dans une aire géographique bien déterminée, les *AIC* sont localisés dans les zones correspondantes. Pareillement, tout *AO* a pour fonction principale de traiter les demandes d'utilisateurs appartenant à une même zone, celle sur laquelle il est responsable. Par conséquent, un *AO* a une portée limitée sur l'espace réservée à la zone en question par rapport aux demandes mais qui peut être étendue pour inclure des offres situées en dehors du périmètre de cette zone.

#### IV.5.2. A chaque agent ses propres fonctions

Les résultats de la phase d'analyse ayant abouti à l'établissement d'une multitude de fonctionnalités, nous verrons dans cette section comment sont alloués les résultats obtenus de par l'analyse fonctionnelle.

Chaque entité telle que définie précédemment correspond à un agent possédant des objectifs qui constituent ses intentions. En effet, chaque entité de la modélisation préalablement établie agit pour jouer un rôle spécifique dans ce système. Nous présentons de ce fait les différents agents mis en place pour la bonne conduite du processus d'affectation optimisée des véhicules aux usagers. Ces agents seront présentés chacun à part ; une description des spécificités qui leurs sont propres en est alors faite. Aussi, nous donnons de plus amples détails sur leurs fonctionnements et leurs contributions dans le processus global.

#### *IV.5.2.1. Des Agents Interface Covoiturage pour assurer la communication avec les usagers*

Notre objectif premier étant de pouvoir satisfaire au mieux les abonnés du service, il est nécessaire de mettre en place des canaux de communication maintenue en continu avec ces derniers, et ce durant toute la période de connexion et d'utilisation du système. Cette nécessité provient essentiellement du fait que d'une part il faudrait pouvoir récupérer leurs requêtes et/ou offres de covoiturage, leurs préférences en matière de déplacement, des mises à jour possibles ou annulations, etc. En effet, le système que nous nous proposons de mettre en place se doit de prodiguer à ses usagers les degrés de flexibilité et de souplesse requis pour être le plus attrayant et efficace possible. D'autre part, afin de pouvoir suivre les usagers continuellement, notamment et essentiellement durant leurs déplacements, un module GIS est intégré dans le système CODAC, et ce pour assurer la géolocalisation instantanée et continue de ses abonnés : Une fonctionnalité qui s'avère des plus efficaces en matière de sécurité. Le défaut de sécurité étant l'une des raisons les plus évidentes ayant causé l'échec d'une multitude de systèmes de covoiturage, particulièrement dans le contexte de temps réel, il est essentiel de considérer, tout particulièrement, cet aspect. Pour ce faire, le maintien d'un contact continu avec les utilisateurs sert à les rassurer et à les mettre en confiance. Outre le fait d'user d'un outil pour la géolocalisation GPS continue, utilisé au niveau de notre système pour réaliser le suivi et la traçabilité de leurs mouvements, des interfaces maintenues actives viennent raffermir et consolider le lien avec l'utilisateur.

Deux types particuliers font partie de cette famille d'*Agents Interface Covoiturage*. Dans le jargon de la modélisation objet, on dit que ces types héritent de l'*AIC*. L'héritage est défini dans le sens où des objets sont définis à partir d'autres plus « abstraits ». Dans le concept agent, des entités peuvent étendre un type bien défini redéfinissant son comportement à quelques détails près. Dans le même contexte et plus particulièrement dans le cadre de notre

système, deux types d'agents spécifiques sont placés sous la catégorie interface. Il s'agit des *Agents Interface Utilisateur (AIU)* et *Agents Interface Véhicule (AIV)* qui sont dotés de comportements similaires, à savoir la transmission des données de et vers les usagers, mais dont les vues telles qu'elles sont implémentées au niveau de notre système diffèrent. Ces agents interfaces sont, en effet, spécifiques à des usagers covoiturés (cas des *AIU*) ou des covoitureurs (cas des *AIV*) et dont les paramètres de déplacement diffèrent.

#### **A. Agent Interface Utilisateur : AIU**

Dès sa connexion au système, tout usager du service est pris en charge par un *Agent Interface Utilisateur (AIU)*. Ce dernier en est responsable durant toute sa période de connexion et prend en charge toute action entreprise par celui-ci. Ces actions sont assimilées à des événements externes provoqués par les agissements (e.g. clique sur un bouton, insertion de données, etc.) de l'utilisateur sur l'*Interface Homme Machine (IHM)* mise à sa disposition.

Dès sa création, un *AIU* se met en veille attendant tout événement susceptible de le réactiver et de déclencher un processus qui lui est propre. L'événement en question peut parvenir de l'intérieur (réception d'un message de la part de l'*AF*) ou bien de l'extérieur du système (i.e. de la part du covoituré). Le processus relatif au fonctionnement d'un *AIU* est décrit au niveau de la Figure IV.11 : de la création à la destruction de toute entité instance de l'agent en question. Comme le montre cette figure, les fonctionnalités d'un *AIU* se résument globalement en :

- La réception d'une requête de covoiturage : ce qui revient à réceptionner les spécificités de déplacement relatives aux différents paramètres du trajet voulu avec les préférences personnalisées de l'utilisateur ; d'en vérifier la coordination et d'en définir les valeurs à défaut de spécification par l'utilisateur lui-même. Tel est le cas par exemple pour l'origine du déplacement qui devrait prendre pour valeur les coordonnées courantes de l'utilisateur (i.e. sa position géographique) si ce dernier a omis de spécifier son adresse de départ ;
- La transmission des données sur la demande de l'utilisateur : son vis-à-vis avec le cœur intelligent du système étant l'*Agent Information (AI)*, l'*AIU* lui communique toutes les informations requises pour le traitement de la requête par envoi de message ;
- La transmission de la réponse à l'utilisateur : dans un sens inverse, l'*AIU* doit pouvoir communiquer à l'utilisateur les données sur une éventuelle solution ou une alerte générée par le système. Ces informations doivent faire l'objet d'un affichage sur le

support de l'utilisateur identifié par l'AF. Une alerte peut éventuellement aboutir à un re-paramétrage de la requête selon le motif de l'alerte et la proposition de la nouvelle demande à l'usager ;

- Assurer l'échange avec l'Agent Fusion (AF) pour la validation ou bien le rejet d'une solution bien déterminée.

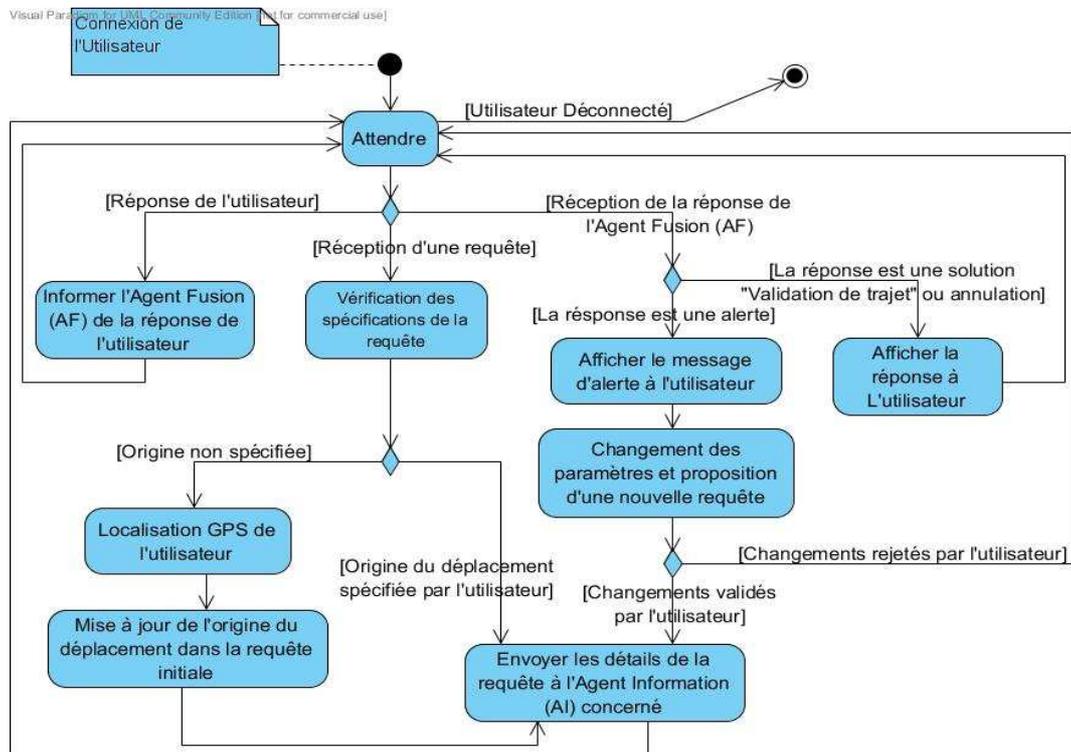


Figure IV.11. Un AIU pour chaque piéton (passager potentiel) connecté

### B. Agent Interface Véhicule : AIV

Il s'agit de mettre à la disposition du système un moyen lui permettant l'échange avec l'extérieur considérant, dans ce second cas, les conducteurs des véhicules. Ceci se traduit notamment par le déploiement d'une interface à travers laquelle il pourra communiquer avec les voitures, qu'elles soient déjà en cours de circulation ou pas encore. En effet, parallèlement à la communication avec les piétons, passagers potentiels, le système doit pouvoir échanger avec les covoitureurs et interagir avec eux. Ces échanges seront au service de la coordination entre les conducteurs et les passagers pour pouvoir organiser leurs voyages et synchroniser automatiquement les prises et déposes.

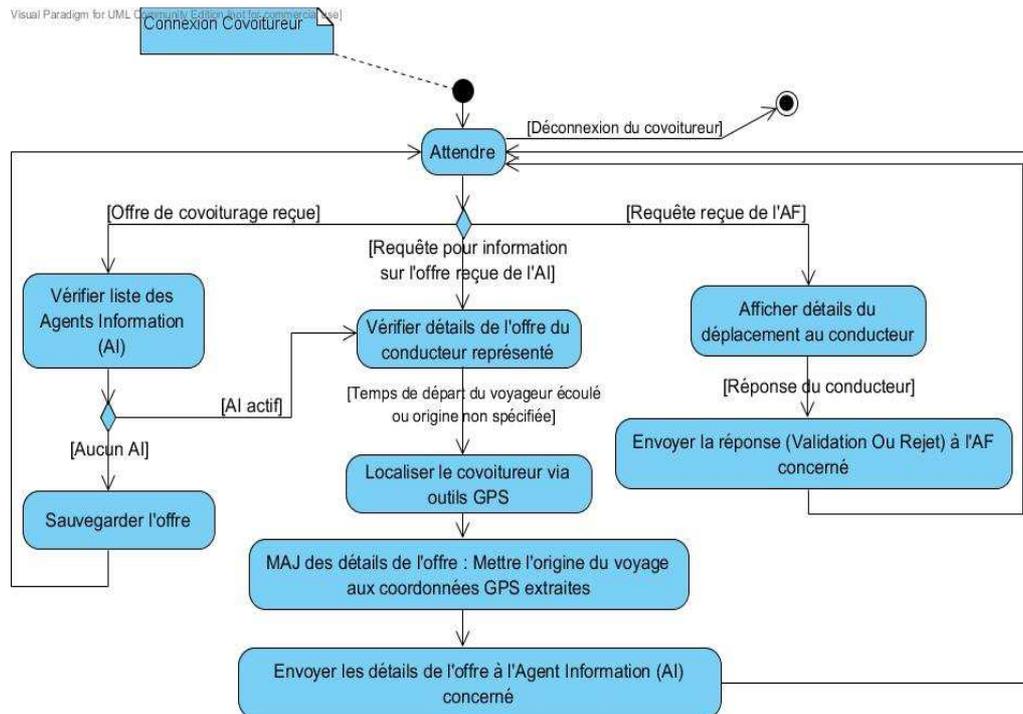


Figure IV.12. Un AIV pour communiquer avec la voiture (i.e. le conducteur)

Faisant intervenir les conducteurs dans ce processus, le système leur transmet les détails d'un trajet de covoiturage et en récupère les réponses de par l'Agent Interface Véhicule (AIV). Ce dernier devra également :

- Recueillir les informations nécessaires sur leurs offres ;
- En vérifier le paramétrage ;
- Les transmettre directement à l'AI si le processus est déjà en cours. Par processus, on entend ici le traitement des requêtes des usagers qui aura déjà été entamé. Toute offre parvenant alors que le programme est en train de s'exécuter doit pouvoir être transmise instantanément. En effet, une voiture peut entrer en réseau et être disponible pour transporter des passages, l'offre qu'elle génère doit être prise en compte au niveau des traitements en cours et ainsi intervenir dans la réalisation des affectations.
- Ou bien les sauvegarder en attendant qu'une demande soit émise par l'AI pour récupérer ces informations. Une relocalisation de la voiture s'impose dans le cas où le covoitureur a déjà entamé son voyage (i.e. voiture en cours de circulation). L'outil de localisation GPS s'avère être efficace et très utile dans ces cas puisque aidant à la manipulation de données réelles (vraies), actualisées de manière instantanée et continue. Autrement, l'AIV peut toujours procéder à une estimation de la distance

parcourue en fonction du *Temps de Parcours Moyen (TPM)*, de l'heure de départ annoncée et de l'heure actuelle (temps système) :

$$Distance = (Heure\ Système - Heure\ Départ) * TPM$$

Cette évaluation est faite afin de pouvoir approcher la distance parcourue et ainsi déterminer la position actuelle du véhicule sur l'itinéraire fixé initialement. Cette formule vient rappeler celle de l'équation III.6 où il s'agit d'évaluer le temps d'arrivée à un nœud extrémité donné.

L'ensemble des fonctionnalités assurées par un *AIV* est illustré dans la Figure IV.12.

#### *IV.5.2.2. Agent Information : AI*

Cet agent a pour fonction principale de collecter les données nécessaires pour la bonne marche du système, à savoir toute information existante susceptible de contribuer à la description de l'état de l'environnement. Un environnement fortement évolutif et instable et dont l'inconstance est au gré de ses acteurs. En effet, la variabilité de leurs besoins et désirs est à l'origine des changements provoqués sur le réseau de covoiturage. Même si le processus de traitement des requêtes a déjà été enclenché, tout changement survenant susceptible de changer le cours du processus en question doit être pris en compte en temps réel.

Le fonctionnement de l'*AI* revient essentiellement à collecter les données essentielles pour le traitement optimisé des requêtes et offres de covoiturage. Créé dès la réception de la première demande de covoiturage, l'*AI* entame ses actions de par la réception des requêtes parallèles ou pseudo-parallèles et de les organiser dans la structure de données adéquate. Le diagramme donné à la Figure IV.13 montre le procédé d'acquisition mené par l'*AI*.

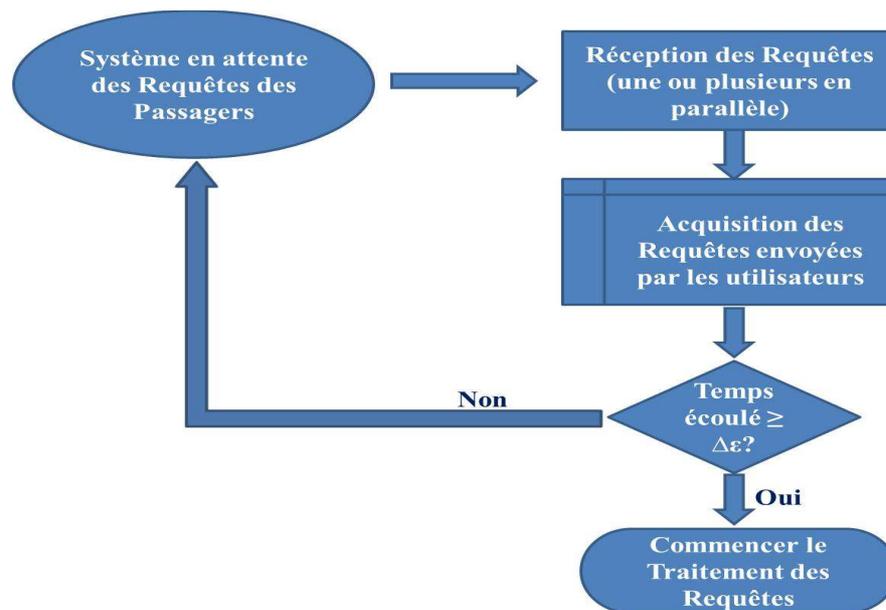


Figure IV.13. Paramètre  $\Delta\epsilon$  pour l'acquisition des requêtes

Nous avons précédemment introduit un paramètre appelé  $\Delta\epsilon$ , correspondant à un temps d'attente négligeable, et ce en faveur du traitement parallèle des requêtes par le système.  $\Delta\epsilon$  concrétise, effectivement, l'acquisition des requêtes sur un laps de temps peu important (de l'ordre de quelques secondes). Il permet alors de favoriser la récupération « *parallèle* » des requêtes parvenant quasi-simultanément au système, tout en étant sans risque sur les performances de ce dernier.

Bien détaillé dans le diagramme d'activité visible de par la Figure IV.14, le fonctionnement de l'AI tourne autour de l'objectif essentiel qui est de mettre en place les structures de données appropriées au niveau desquelles sont organisées offres et demandes.

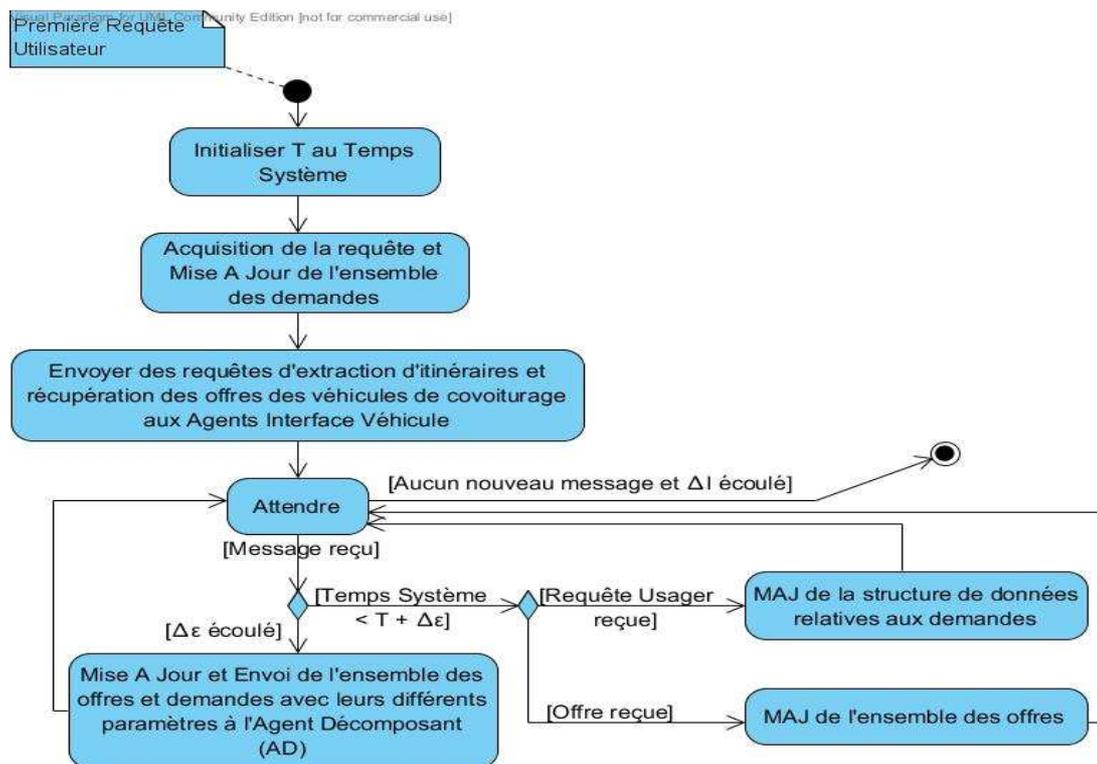


Figure IV.14. Un AI pour la collecte et structuration des données

#### IV.5.2.3. Agent Décomposant : AD

Compte tenu de la modélisation précédemment établie, un agent spécifique dit de décomposition est responsable de mettre en place l'architecture physique escomptée (i.e. GDD). Cet agent, nommé *Agent Décomposant (AD)* est ainsi responsable de mener à bien le processus de subdivision du réseau de desserte (c.f. III.3.2). Les algorithmes de décomposition préalablement définis sont ainsi conduits par cet agent pour l'établissement des ensembles de nœuds voisins. Ces nœuds étant constitués sur la base des données qu'il aura reçues de la part de l'AI sur les trajets offerts et demandés, l'AD en extrait les « communautés » de nœuds proches dans l'espace. Ces derniers aideront à définir les zones qu'elles soient primaires ou secondaires et leur étendue géographique grâce au procédé de calcul des centres et du diamètre prédéfini.

De manière similaire aux autres agents et comme le montre la Figure IV.15, l'AD doit rester actif durant tout le temps que requiert l'accomplissement des traitements des requêtes. A la fin de ce processus, l'AD devra par ailleurs attendre pendant  $\Delta I$  secondes, qualifié de temps d'inactivité. Il s'agit d'un laps de temps durant lequel l'agent se met en attente d'un éventuel changement (i.e. modification ou annulation d'une demande ou d'une offre, réception de nouvelles données, etc.) en conséquence duquel l'infrastructure du réseau établie

devrait changer. Si à la fin de  $\Delta I$ , aucun évènement ne survient et le processus ayant été terminé, alors l'agent s'autodétruit.

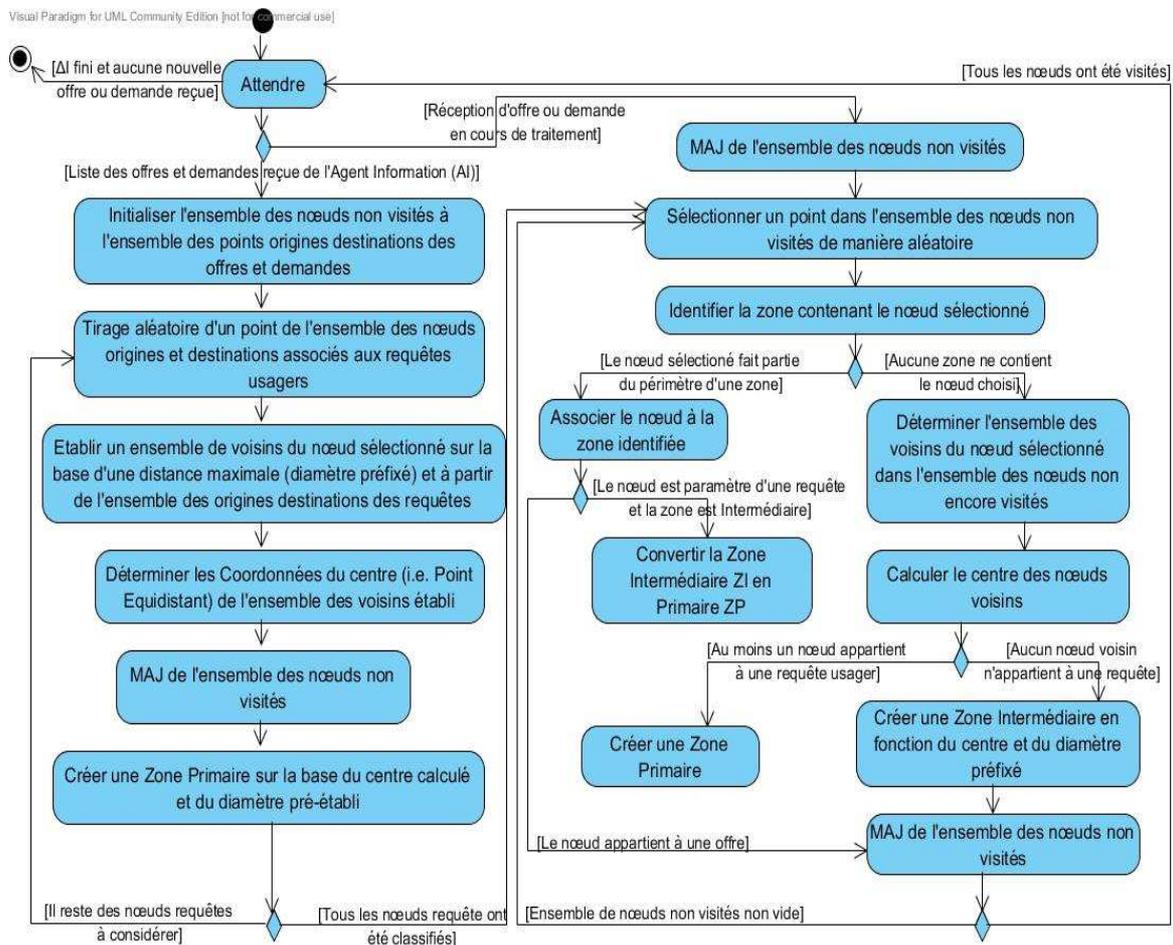


Figure IV.15. Un AD pour la subdivision du réseau

#### IV.5.2.4. Agent Optimisateur : AO

Un descriptif détaillé de cet agent fera l'objet de la prochaine section. Celle-ci traite essentiellement des approches d'optimisation proposées au niveau de nos travaux. Deux procédés spécifiques et distincts ayant été identifiés, pour approcher le problème d'optimisation, deux comportements différents ont alors été implémentés au niveau des agents optimisateurs. Chaque comportement est relatif à l'une ou l'autre des deux approches. Ces dernières sont elles-mêmes relatives à des algorithmes d'optimisation bien définis et distincts qui sont implantés dans le cœur des agents optimisateurs. Par ailleurs, les deux comportements, même si basés sur des démarches et principes totalement différents, ont en commun le fait de départager les tâches d'optimisation sur les différentes instances initiées de l'agent optimisateur ; lesquelles sont établies et lancées conformément à la modélisation de

Graphe Dynamique Distribué et ce à l'image de la répartition des données. Les entités ainsi créées exécutent leurs tâches localement à la zone dont elles sont responsables pour ensuite se rassembler dans un contexte de coalition et ainsi reconstruire des solutions complètes et cohérentes. Les instances en question, qu'elles exécutent l'une ou l'autre des deux approches, convergent dans le sens où leurs comportements s'inscrivent dans le contexte d'une optimisation distribuée. De plus amples détails sur la manière suivant laquelle est mené le traitement optimisé des affectations sont donnés, dans la suite, avec une description complète des algorithmes alloués à cet effet.

#### IV.5.2.5. Agent Fusion : AF

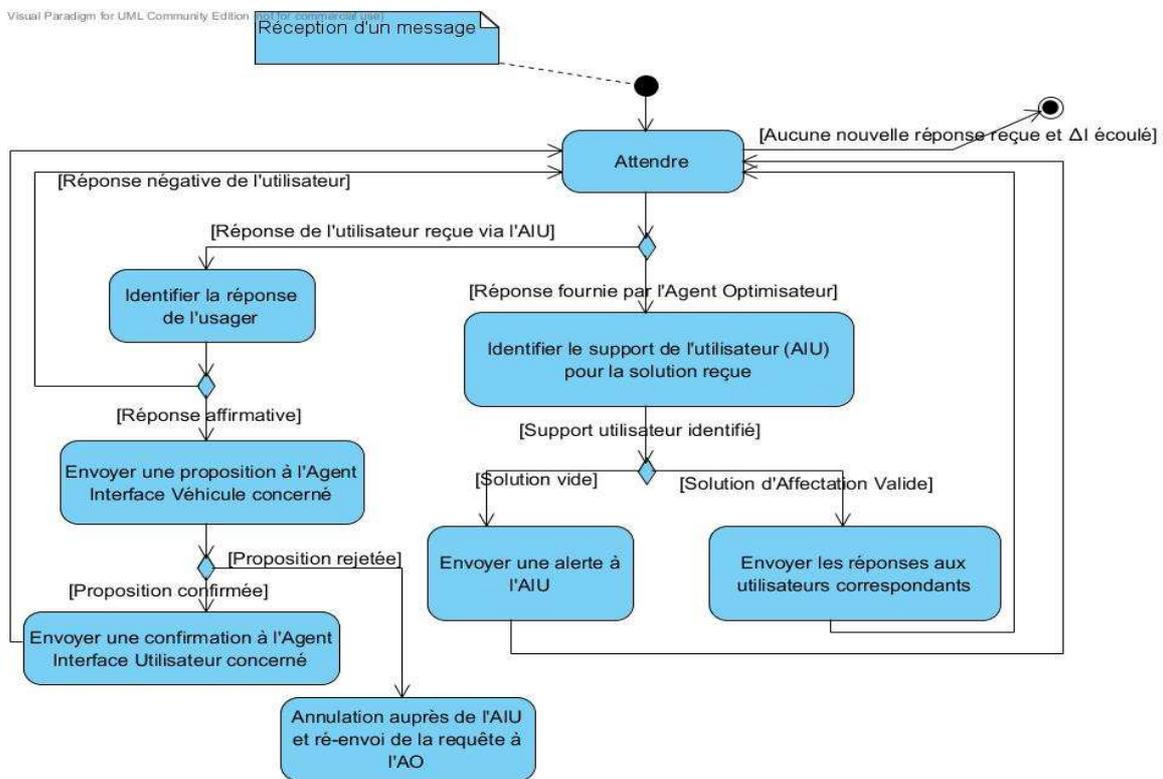


Figure IV.16. Un AF pour le dispatching des réponses

Le comportement de l'Agent Fusion (AF) est illustré de par la Figure IV.16 indiquant ainsi les actions entreprises par toute instance de celui-ci. Ces actions sont enclenchées en temps voulu, c'est à dire lors de l'arrivée d'un message d'un AIU ou bien d'un AO. Qu'il s'agisse d'une validation ou d'un rejet de la part de l'utilisateur concernant une réponse déjà fournie ou bien de la solution générée par l'AO, un traitement particulier doit être fait. De ce fait, un AF est instancié dès lors qu'un ensemble de solutions est généré par le système. Il aura pour responsabilité d'identifier le support usager concerné par chaque solution reçue, identifier la

nature de la réponse (vide ou bien valide) et agir en conséquence. Il est aussi pourvu de la capacité de jouer le rôle d'agent médian entre l'*AIU* et l'*AIV* pour la validation ou non d'un trajet rassemblant les usagers concernés.

## **IV.6. CODAC : un système basé sur des agents communicants**

Les entités établies dans notre système ont pour mission de jouer leur rôle dans une synchronisation totale mettant ainsi l'environnement à l'abri de toute sorte de conflit et faisant régner une intelligence ambiante sur tout le système. Pour cela, l'établissement de canaux de communication entre les agents en question s'impose. Le langage de modélisation *AUML* que nous avons utilisé pour la conception de l'architecture de notre système ainsi que pour la modélisation des comportements des agents se trouve être bien approprié à ce contexte. En effet, *AUML* met à la disposition des concepteurs et développeurs des outils permettant de représenter les flux de communications entre des objets actifs dits « *threads* », de manière concurrente (*c.f. Annexe I*). Il s'agit d'une extension des diagrammes de collaboration qui permettent notamment de représenter des communications entre processus ou l'exécution de « *threads* ».

Les diagrammes de séquences et les diagrammes d'état-transitions catégorisés sous la classe des diagrammes d'interaction sont les vues dynamiques les plus importantes d'*AUML*. Dans nos travaux, nous nous sommes particulièrement intéressés au premier type de diagrammes de cette catégorie dont la représentation se concentre sur l'expression des interactions à travers l'échange de messages. Les diagrammes de séquences modélisent par ailleurs les collaborations entre agents selon un point de vue temporel. L'accent y est mis sur la chronologie des envois mettant ainsi en exergue la synchronisation des actions entreprises par les différents agents. Dans notre contexte, nous nous sommes plutôt focalisés sur les comportements des différentes entités pouvant intervenir dans le processus d'affectation. Visant comme objectif principal la description des buts et intentions des agents, leurs comportements sont alors modélisés de par les échanges ayant lieu d'être entre eux. Les objectifs en question sont plutôt relatifs, dans ce cadre, à des buts collectifs dans le sens où certaines des fonctions incombant au système sont réalisées de par la collaboration, à un instant  $t$  bien déterminé, d'une communauté de  $n$  agents intelligents et semi-autonomes  $A(t) = \{A_1, A_2, \dots, A_n\}$ . Ces derniers se partagent  $m$  tâches  $T(t) = \{T_1, T_2, \dots, T_m\}$  où  $T(t)$  est

relatif à la fonctionnalité à réaliser, à l'instant  $t$ , dans le contexte de la coalition dans laquelle interviennent ces agents. Pour la réalisation d'une tâche  $T_j$ , un agent  $A_i$  est doté d'un vecteur de capacités réelles non négatives  $B_i(t) = \langle B_1^i, \dots, B_r^i \rangle$ . Tout le processus est accompli au travers d'un certain nombre ( $k$ ) d'objectifs  $G(t) = \{g_1, g_2, \dots, g_k\}$  qui modélisent un objectif plus global, celui visé par la communauté  $A$  des agents en coalition. Chacun de ces objectifs ( $g \in G$ ) requière un ensemble de capacités  $B_g = \langle B_1^g, \dots, B_f^g \rangle$  pour pouvoir être réalisé. Cette formulation, inspirée des travaux de shehory et al. dans (123) et (124) implique aussi la définition d'une fonction d'évaluation  $V$  de la coalition qui définit son utilité. Utilité en matière d'apport de la collaboration des agents, à l'instant  $t$ , pour la réalisation du but de la coalition en question ( $G_C(t)$ ). Une coalition  $C(t)$  est ainsi pourvue d'un ensemble de capacités  $B_C(t)$  et qui est la somme des capacités des agents  $A(t)$  dans cette coalition.

Dans le cadre de la modélisation des coalitions ayant lieu d'être dans le système que nous proposons, nous nous intéressons essentiellement aux flux de communication. Nous en définissons deux types : Le premier étant relatif à l'échange avec le monde externe est nommé « *communication inter-système* » et le deuxième est appréhendé d'un point de vue interne au système et est appelé « *communication intra-système* ». Chaque contexte est décrit dans le cadre d'une coalition d'un ensemble d'entités, instances des agents impliqués.

#### IV.6.1. Communication Inter-Système

Dans le système que nous nous proposons de mettre en place, nous avons jugé utile de mettre à la disposition des entités logicielles impliquées un support de communication avec l'extérieur. Ce support favorise l'écoute de manière continue du « monde réel » symbolisé par les acteurs humains intervenant au niveau du système. Cette écoute sera utile en matière de rapide prise en compte de tout changement émis par les usagers ; à savoir, des modifications dans leurs préférences de déplacement, des annulations de voyage, l'arrivée de nouvelles offres ou demandes de covoiturage, etc.

Aussi, à un stade assez avancé de l'élaboration d'un système de covoiturage efficace, un support de communication établi en continu aide à la prise de décision en fonction des données du réseau. Dans ce contexte, nous faisons particulièrement allusion aux imprévus pouvant survenir. Parmi ceux-ci nous pouvons citer à titre d'exemple, des perturbations sur les routes (e.g. embouteillage, accidents, etc.) ou le désistement d'un conducteur, la panne

d'un véhicule, un piéton qui ne s'est pas présenté au rendez-vous, etc. Transmises de par les canaux de communication établis avec le monde extérieur, toutes ces informations aident à la mise à jour continue des données dont dispose le système pour prendre les décisions appropriées, réaliser des affectations cohérentes et éviter toute possibilité de conflit. Vers un risque zéro, le maintien d'une écoute continue aide ainsi à la manipulation de données réelles et actualisées réduisant ainsi la probabilité d'erreur à néant.

Le but ici est d'acquérir les données du système, la Figure IV.17 montre un scénario basique de coalition entre des instances d'*AIU*, *AIV* et *AI* pour l'objectif final de mettre en place les structures de données adéquates rassemblant toutes les données recueillies.

Tel qu'il est montré sur cette figure, les différentes entités exécutent des tâches différentes dépendant de leurs rôles et d'objectifs spécifiques réalisés dans le cadre d'un objectif global auquel devrait aboutir le système de par cette coalition :

- Les *AIU* ont pour mission de collecter toutes les informations et veiller au bon paramétrage et synchronisation des données sur celles-ci.
- Les *AIV* sont plutôt concernés par les voitures en veillant à en suivre les déplacements ; celles-ci étant généralement en continu mouvement et ce à une vitesse beaucoup plus rapide que celles des piétons (à quelques exceptions près, en cas de perturbations).
- Les *AI* sont dotés d'une capacité organisationnelle des informations collectées de par les différentes instances d'*AIU* et *AIV*. L'objectif principal étant de mettre en place des structures de données pour l'organisation des celles brutes collectées et de pouvoir continuellement les mettre à jour en fonction des perceptions de l'environnement.

Par souci de visibilité dans le diagramme fourni à la Figure IV.17, le fonctionnement des *AIU* et *AIV* n'est détaillé que pour le premier acteur (i.e. *Utilisateur 1*) de par le fonctionnement de l'agent *AIU 1* qui en est responsable. En effet, les actions de réception des offres et demandes, de vérification, de géolocalisation et de re-paramétrage sont des actions dont les *AIU* et *AIV* se partagent la responsabilité. Ceci dans le but de fournir des informations cohérentes et actualisées à l'*AI* qui sera responsable de les organiser à la manière dont les structures de données correspondantes ont été définies dans la chapitre précédent (c.f. *Chapitre III*). En effet, le but de cette coalition est de rassembler les informations et de les structurer de sorte à ce qu'elles soient manipulables par les *AD* et *AO* et préparant ainsi le terrain pour le traitement des affectations des véhicules aux usagers.

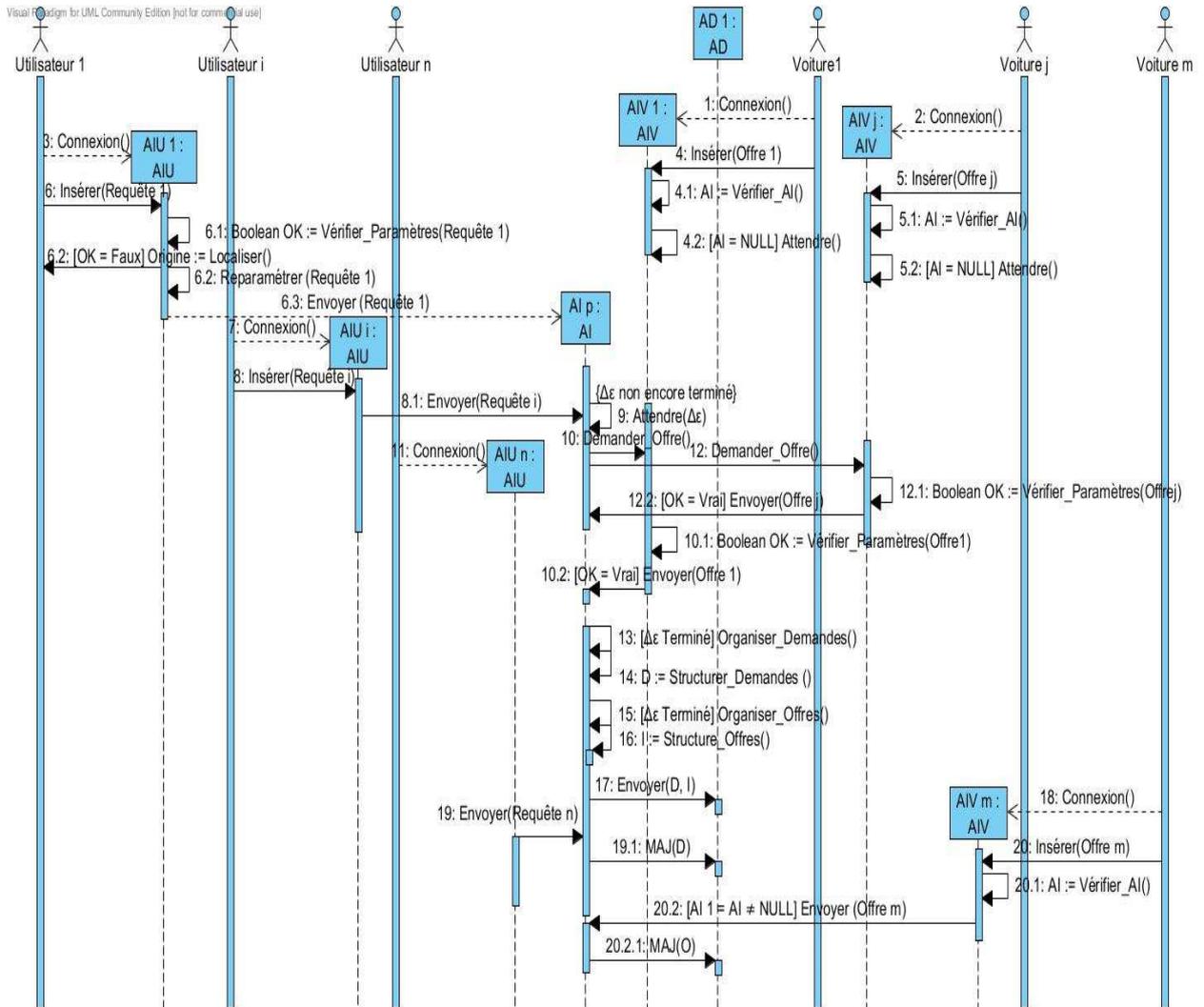


Figure IV.17. Collecte et Organisation des données sur les offres et demandes de covoiturage

Un autre exemple de communication incluant des flux sortants et entrants au système concerne la coordination entre piétons (potentiels passagers) et conducteurs pour l'organisation d'un voyage collectif. Un exemple qui fait intervenir une communauté d'agents *AF*, *AIV* et *AIU* dans le cadre d'une coalition dont le but principal est d'obtenir un accord final satisfaisant les acteurs concernés.

#### IV.6.2. Communication Intra-Système

Tout le système proposé repose sur le principe de la distribution : distribution des tâches, des traitements des affectations, etc. dans le but essentiel de réduire la complexité du problème. Les coalitions inter-agents se prêtent fortement à ce contexte et en modélisent le partage et la distribution des tâches. Parmi les collaborations à considérer dans le cadre des

communications internes au système, nous pouvons citer celles relatives à la décomposition du réseau géographique et la distribution des données sur les zones, le traitement d'une même requête de manière distribuée sur un ensemble d'*AO*, etc.

## **IV.7. Modélisation formelle : pour un procédé d'affectation optimisé et performant**

La modélisation présentée dans la précédente section se base sur la coopération de plusieurs agents afin de mener à bien le processus de traitement des requêtes utilisateurs. Notre but premier de par ces travaux étant de satisfaire un nombre aussi large que possible d'abonnés du service, aussi bien les piétons que les conducteurs, les contraintes de correspondance entre leurs trajets respectifs sont considérées dans un premier temps. Ces contraintes ont été évoquées dans le précédent chapitre et détaillées avec les formulations mathématiques adéquates afin de signaler le respect des préférences des voyageurs en premier lieu et en priorité. La notion de faisabilité des offres passe ainsi au premier degré dans le traitement effectué, donnant lieu à un ensemble de solutions potentielles susceptibles de répondre à la requête de l'utilisateur. Après l'extraction de l'espace de solutions possibles, le système proposé passe au second degré de son fonctionnement ; à savoir, la recherche de la meilleure solution parmi celles-ci compte tenu du critère d'optimisation fixé, ici établi comme étant la durée globale du trajet. Deux approches différentes ont été proposées sur la base des deux principes de résolution adoptés et implémentés au niveau de notre système. La première correspond à une adaptation des algorithmes de Dijkstra et la seconde consiste en une méthode approchée d'optimisation. Il s'agit au niveau du procédé décrit dans cette dernière de considérer l'implémentation d'une heuristique spécifique qui intègre de manière implicite un critère supplémentaire d'optimisation, à savoir le confort, ou plus concrètement le nombre de transferts d'un véhicule à un autre. Pourquoi implicite ? Parce qu'il ne s'agit pas d'inclure le critère de confort ici considéré dans l'expression de la fonction Fitness pour l'évaluation des solutions générées, mais plutôt dans le comportement même des algorithmes implémentés dans l'*AO*. Cette approche est venue s'imposer suite à la discrimination de la méthode de Dijkstra de solutions attrayantes et de la génération de celles parfois beaucoup plus contraignantes, ne considérant que le critère de durée globale du voyage au détriment de celui du nombre de transferts qui peut parfois s'avérer être assez important au sein d'un même trajet. Les deux approches sont détaillées dans la suite de cette section.

### IV.7.1. A3D : Algorithme Distribué Dynamique basé sur Dijkstra

Dans cette première approche, nous nous sommes inspirés des travaux de Feki (125) et Kammoun (126) ; leurs approches étant fondées sur le principe des algorithmes de Dijkstra distribués pour la recherche d'itinéraires optimisés dans le cadre de l'aide au transport des voyageurs. Les performances de ces algorithmes ayant été démontrées comme étant efficaces et d'une complexité d'ordre polynomial, nous avons opté pour l'implémentation de cette approche. Le choix d'adapter le principe d'exécution des algorithmes de Dijkstra nous a conduits à en distribuer l'exécution, essentiellement sur les *AD* et *AO*. Toujours dans le contexte d'une coalition entre les différentes entités impliquées pour la réalisation des affectations optimisées visées, l'*AD* s'en trouve être l'initiateur, responsable entre autre de lancer les requêtes aux autres agents pour pouvoir former une coalition et identifier les agents avec qui il peut collaborer. Il a de ce fait pour mission de réaliser les initialisations nécessaires relatives, notamment, aux pondérations initiales à affecter aux nœuds du graphe établi de par la modélisation du système. Par la suite, l'*AD* se doit d'identifier et de lancer les différents *AO* responsables sur les zones considérées lors du déclenchement du processus d'exécution ; c'est-à-dire les zones qui contiennent les origines des requêtes de covoiturage (i.e. *ZP*). Ce processus correspond à l'exécution de l'Algorithme 3 fournit dans ce qui suit :

#### Algorithme 3 : Algorithme Distribué Dynamique basé sur Dijkstra (*A3D*)

**Données :**  $G = (N, A)$  : Graphe modélisant le réseau,  $R$  : Requêtes des utilisateurs,

**Résultat :**  $OS$  : Ensemble des Solutions Optimisées

1. **Initialisation :**

2. Nœud  $D \leftarrow NULL$

3. **Pour Tout** nœud  $N_a$  **tel que**  $N_a \in N$  **Faire**

4.  $Poids[N_a] \leftarrow \infty$

5. **Fin Pour**

6. **Pour Toute** requête  $r_p$  **tel que**  $r_p \in R$  **Faire**

7.  $Poids[r_p^+] \leftarrow d_p$

8. **Fin Pour**

9. **Pour Tout**  $AO_i$  **Faire** {seuls les  $AO_i$  sur les *ZP* contenant des sources, origines de

requêtes, sont initialement lancés}	
10.	$Poids_{AO_i}, Voiture_{AO_i} \leftarrow A3D(G_i, TPM, I(t), R_i)$ {Lancer l'algorithme <i>ADO</i> : <i>Algorithme 4</i> , sur chaque $AO_i$ }
<b>11. Fin Pour</b>	
<b>12. Pour Toute</b> requête $r_p$ <b>tel que</b> $r_p \in R$ <b>Faire</b>	
13.	$D \leftarrow r_p^-$
<b>14. Tant que</b> ( $D \neq r_p^+$ ) <b>Faire</b>	
15.	Insérer $D$ au début de $OS_p$
16.	Déterminer la zone qui contient $D$ pour considérer les résultats fournis par l' <i>AO</i> concerné
17.	$Z_j \leftarrow Localiser(D)$
18.	$OS_p[Pr\ éd\ ec\ esseur[D], D] \leftarrow (Pr\ éd\ ec\ esseur[D], D, Voiture_{AO_i}[Pr\ éd\ ec\ esseur[D], D])$
19.	$l_{Voiture[Pr\ éd\ ec\ esseur[D], D]} \leftarrow l_{Voiture[Pr\ éd\ ec\ esseur[D], D]} - P_p$
20.	$D \leftarrow Pr\ éd\ ec\ esseur[D]$
<b>21. Fin Tant que</b>	
<b>22. Fin Pour</b>	
23.	$OS \leftarrow OS \cup \{OS_p\}$
24.	Envoyer $OS$ à $AF$

L'initialisation ayant été effectuée et les agents entrant dans le processus de traitement de l'ensemble des requêtes identifiés ; l'*AD* lance en parallèle les différents *AO* impliqués pour une exécution de l'Algorithme de Dijkstra Optimisé (*ADO*), sur chacun, localement aux zones dont ils sont responsables. *ADO* est associé à l'Algorithme 4 défini dans ce qui suit.

L'Algorithme 4 fournit le détail d'application de l'algorithme Dijkstra localement à chaque zone par l'*AO* qui en est responsable. Cet algorithme est effectué dans le but de chercher les affectations possibles et particulièrement celles qui optimisent le mieux les temps d'attente et temps de parcours compte tenu des contraintes de correspondance ainsi que des affectations conflictuelles. Dans ce sens, les solutions générées sont celles qui favorisent l'arrivée des usagers au plus tôt à la destination souhaitée. Dans ce contexte, partant d'une source donnée localisée au niveau de la zone dont il est responsable, chaque *AO* se doit de parcourir les chaînes de nœuds successifs leur affectant au fur et à mesure les poids qu'il aura calculé

préalablement en fonction de l'offre disponible et ce, éventuellement, après avoir vérifié les contraintes de correspondance entre les trajets offerts et ceux demandés. Par ailleurs, même s'ils ne sont pas fiables au regard de la demande (source) considérée, les calculs effectués au fur et à mesure peuvent s'avérer utiles dans plusieurs cas de figures ; comme par exemple dans le cas où d'autres requêtes partagent les tronçons actuellement en cours de traitement mais qui ont leurs origines en dehors de la zone considérée, à laquelle la portée de l'AO est limitée. D'où l'idée motrice de garder toute trace de calcul effectué afin de disposer d'une information fiable, rapidement et efficacement au cas où l'AO en question est sollicité de manière à traiter les mêmes sections de routes avec des paramètres différents, pouvant probablement concorder avec les solutions d'affectation déjà calculées. La mise en place de structures de données spécifiques stockant temporairement l'ensemble des informations sur les calculs effectués permet d'y accéder rapidement et aisément et donc d'éviter des traitements supplémentaires inutiles allourdissant le système.

Algorithme 4 : Algorithme de Dijkstra Optimisé (ADO) pour l'affectation des véhicules

**Données :**  $G_i = (N_i, A_i)$  : Graphe modélisant la zone  $Z_i$ , TPM,  $R_i$  : Requêtes dont les origines sont dans  $Z_i$ ,  $I(t)$  : matrice représentant les Itinéraires des véhicules.

**Résultat :** *Poids* : Tableau des poids des nœuds, *Voiture* : Matrice contenant les voitures affectées par tronçon d'itinéraires, *Prédécesseur* : Tableau contenant les prédécesseurs des nœuds

1.  $Q \leftarrow N_i$
2. **Tant que**  $Q \neq \emptyset$  **Faire**
3.      $p \leftarrow \min(\text{Poids})$  {choisir le nœud de poids minimal}
4.      $Q \neq Q - \{p\}$
5.     **Pour Tout** nœud  $q$  **tel que**  $(p, q) \in A_i$  **Faire** {nœuds de l'ensemble des voisins de  $p$ }
6.     **Pour Tout**  $o_j \in I(t)$  **Faire**
7.         **Si** (*Vérifier\_Constraintes*( $o_j, r_p$ )) **Alors**

```

8.       $W_t \leftarrow \omega_1 * Distance(o_j^+, p) * TPM(o_j^+, p, d_j)$ 
9.      Si ( $W_t + t - Poids[p] > 0$ ) Alors
10.      $W_t[j] \leftarrow W_t$ 
11.     Fin Si
12.     Fin Si
13.     Fin Pour
14.      $c \leftarrow \min(W_t)$  {Choisir la voiture ayant un temps d'attente minimal}
15.      $a \leftarrow Poids[p] + W_t[c] + \omega_2 * Distance(p, q) * TPM(p, q, Poids[p] + W_t[c])$ 
16.     Si  $a < Poids[q]$  Alors
17.          $Poids[q] \leftarrow a$ 
18.          $Pr\ ed\ e\ c\ e\ s\ s\ e\ u\ r[q] \leftarrow p$ 
19.          $Voiture[p, q] \leftarrow c$ 
20.     Fin si
21. Fin Pour
22. Fin tant que
23. Envoyer Poids aux zones adjacentes
24. Retourner Poids et Voiture

```

Après avoir terminé l'exécution locale de *ADO*, chaque *AO* transmet l'ensemble des solutions générées à l'*AD* qui devient apte à partir de ce moment à recomposer pour chaque requête une solution globale optimisée. Ceci est mené, tel que décrit dans la seconde partie de l'Algorithme 3 (*A3D*), de sorte à trouver les différentes jonctions de routes possibles pour reconstruire un itinéraire global. Pour cela, partant d'une destination souhaitée par l'utilisateur, l'*AD* recherche en remontant les conjonctions possibles de routes, en parcourant les prédécesseurs des nœuds trouvés successivement jusqu'à parvenir à la source correspondante à l'origine de la demande considérée. Ce procédé est conduit tout en réalisant au fur et à mesure les différentes affectations et mises à jour conséquentes des paramètres des offres sur les tronçons alloués aux usagers pour ainsi éviter les pièges d'affectations conflictuelles.

Cependant, tel que nous l'avons énoncé précédemment, ce procédé peut être à l'origine de la discrimination de solutions moins restreintes sur le critère de durée globale du trajet mais par contre plus admissibles et attrayantes pour l'utilisateur. Celles-ci peuvent, en effet, faire intervenir considérablement moins de changements de véhicules et présenter, dans le même temps, une différence minimale par rapport à leur évaluation en termes de durée de parcours des trajets souhaités par rapport à celles ici générées. D'où l'idée d'introduire le critère de confort qui a été à l'origine d'une seconde proposition apportant une pointe d'originalité à nos contributions dans la considération de la problématique du covoiturage dynamique optimisé.

#### IV.7.2. ODAVe : Optimisation Distribuée de l'Affectation des Véhicules aux usagers

Contrairement à la précédente approche, la méthode d'optimisation distribuée présentée ici, ODAVe, intègre le critère de confort en termes de nombre de transferts entre voitures au sein d'un même trajet. Nous nous concentrons, par ailleurs, dans cette première partie sur le détail relatif au travail collaboratif entre les AO, avant de procéder à la description de l'approche que nous proposons.

Chaque AO responsable d'une zone contenant l'origine d'une requête  $R_p^+$  initie une coalition qui fait intervenir un ou plusieurs autres AO dans une coopération pour pouvoir mener à bien le traitement distribué et optimisé d'une requête  $R_p(t)$  donnée. Celle-ci constitue le but visé par la communauté d'agent  $A_p(t) = \{AO_1 \dots AO_n\}$  intervenant, à l'instant  $t$ , dans le cadre de la coalition initiée dans le but de réaliser le traitement de la requête  $R_p(t)$ . La recherche d'une solution globale optimisée d'affectation de véhicule(s) à l'utilisateur  $U_p$  constitue l'objectif global  $G_p(t)$  de l'ensemble des entités intervenantes. Cette coalition  $C_p(t)$  est concrétisée de par la décomposition probable d'une requête utilisateur  $R_p(t)$  en plusieurs sous-requêtes partielles constituant les tâches affectées aux différents AO impliqués :  $T_p(t) = \{T_{p_1} = R_{p_1}, \dots, T_{p_m} = R_{p_m}\}$ . Chacune d'entre elles est traitée par l'une des entités (AO) identifiées comme répondant positivement à celle initiatrice pour pouvoir former la coalition souhaitée. Il s'agit généralement, dans ce cadre, des différents AO responsables sur les zones intégrant les sources des requêtes partielles dans leurs périmètres respectifs. L'objectif global de toute coalition  $C_p(t)$ , initiée dans le cadre de l'exécution de chaque

requête  $R_p(t)$  considérée lors du lancement du processus incombant au système à un instant  $t$  donné, se traduit par l'aboutissement à la génération d'une solution optimisée sur celle-ci. Cette solution respecte, par ailleurs, les contraintes de correspondance imposées par la formulation de la ou les sous-requête(s) élaborée par l'AO initiateur de la coalition. Ce dernier, fournit effectivement les différents paramètres nécessaires pour le traitement des requêtes partielles incombant aux AO impliqués dans la coalition en question. Les différentes réponses sur les requêtes partielles ayant été reçues, le premier AO intervenant se doit de recomposer une solution globale répondant à la totalité du trajet demandé.

Dans le but d'améliorer les performances du système, essentiellement par rapport à la qualité de la réponse fournie en termes d'optimisation par égard aux attentes des usagers, cette seconde approche, nommée *ODAVE*, a été proposée. Par ailleurs, outre cet aspect d'amélioration et de contribution nouvelle impliquant l'intégration implicite du caractère d'optimisation relatif au confort, il s'agit aussi de pouvoir répondre aux critères d'optimalité par rapport aux performances d'exécution. Nous nous sommes alors concertés sur l'élaboration d'une méthode approchée qui adopte pour support exécutif un principe d'interruption dès lors qu'une solution complète optimisée est trouvée ; si toutefois, elle existe. Dans le cas contraire, une décomposition de la requête initiale est envisageable, considérant pour cela le nœud le plus proche de la destination souhaitée pouvant être desservi à partir de l'origine du déplacement. C'est dans ce cadre que loge principalement le critère d'optimisation par rapport au nombre de transferts. La recherche du nœud le plus proche sous-entend implicitement la considération de solutions prévoyant de longues distances de parcours dans un même véhicule en priorité, et donc celles présentant à fortiori le moins de changements. Ceci n'empêche pas le fait que les décompositions peuvent se succéder dans la mesure où n'étant pas capables d'assurer la continuité en l'absence d'offres complètes sur l'une ou l'autre des parties du trajet (requête), le même procédé est réappliqué jusqu'à l'aboutissement à une solution faisable complète et optimisée. Sur la base de ce principe et tel que le montre la Figure IV.18, il s'agit :

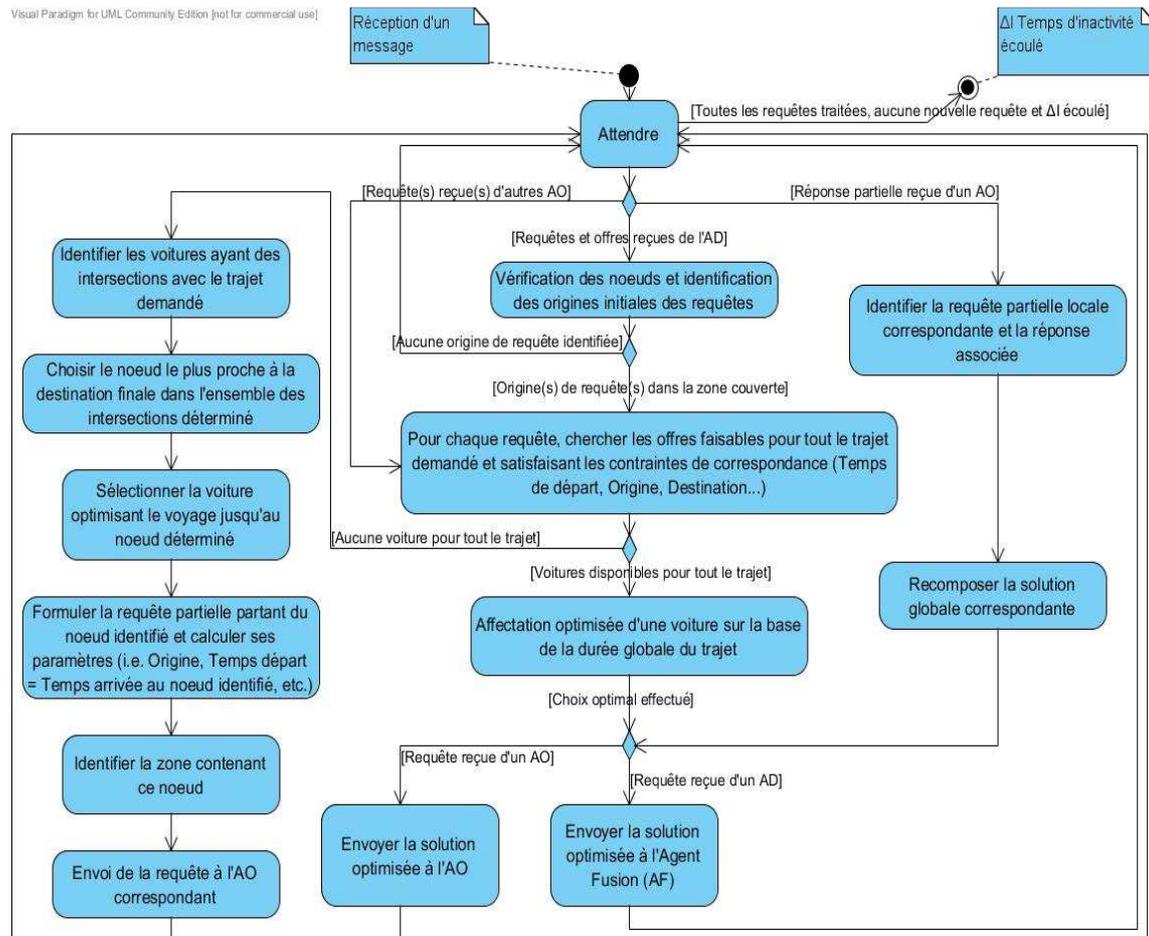


Figure IV.18. Un AO pour le traitement optimisé distribué des requêtes basé sur ODAVe

1. De chercher le nœud le plus proche desservi
2. Décomposer la requête
3. Résoudre la première partie de la requête
4. Envoyer un message de type « Request » à l'AO responsable sur la zone contenant le nœud le plus proche considéré
5. Répéter sur l'ensemble des nœuds les plus proches desservis jusqu'à ce qu'une solution concrète, faisable et optimisée soit générée ou jusqu'à l'épuisement de toutes les alternatives envisageables (i.e. jusqu'à ce qu'il n'existe plus de nœuds à visiter)

Tout ce fonctionnement avec la modélisation appropriée est détaillé au travers des Algorithmes 5 et 6 ainsi que la Figure IV.18.

Algorithme 5 : Optimisation Distribuée de l'Affectation des Véhicules aux usagers (ODAVe)

**Données:**  $R_{AO_p}$  : Requêtes dont les origines sont localisées dans la zone représentée  $Z_p$  incombant à  $AO_p$ ,  $I(t)$ : Matrice des offres, SRT: Seuil de Retard Tolérable, VMP: Vitesse Moyenne de Parcours,  $t$  : temps système,  $\{\omega_1, \omega_2\}_R$

**Resultat:**  $S$  : Ensemble de solutions pour  $R$

1. **Initialisation:** Solution  $S \leftarrow \phi$

2. **Pour Toute** requête  $r_i$  **telle que**  $r_i \in R_{OA_p}$  **Faire**

3.  $S_i \leftarrow OGI(r_i)$

4. **Si**  $S_i = \phi$  **Alors**

5.  $IN \leftarrow IN \cup \{Noeud\_Plus\_Proche(I(t), r_i^-)\}$

6. **Répéter**

7.  $NP \leftarrow Noeud\_Plus\_Proche(IN, r_i^-)$

8.  $IN \leftarrow IN - NP$

9.  $r_{i_1} \leftarrow Etablir\_Requete(r_i^+, NP, P_i, d_i, a_{NP})$

10.  $S_{i_1} \leftarrow OGI(r_{i_1})$

11.  $Z_c \leftarrow Zone\_Contenante(NP) \{ NP \in Z_c \}$

12.  $r_{i_2} \leftarrow Etablir\_Requete(NP, r_i^-, P_i, d_{NP}, a_{r_i^-})$

13. Envoyer  $r_{i_2}$  à  $AO_c$  {  $AO_c$  est l' $AO$  responsable sur  $Z_c$  }

14.  $S_{i_2} \leftarrow ODAVe_{AO_c}(r_{i_2})$

15. **Si**  $S_{i_2} = \phi$  **Alors**

16.  $S_i = \phi$

17. **Sinon**

18.  $S_i = S_{i_1} \circ S_{i_2}$

19. **Fin Si**

20. **Jusqu'à**  $S_i \neq \phi$  Ou  $IN = \phi$

21. **Fin Si**

22.  $S \leftarrow S \cup \{S_i\}$

23. **Fin Pour**

#### 24. Envoyer S à AF

L'Algorithme 6 ci-dessous décrit est relatif à la fonction de recherche d'une offre susceptible de répondre au trajet voulu dans sa totalité. Parmi les principaux rôles de l'AO dans ce contexte : la vérification de la satisfaction des contraintes de correspondance entre chaque offre considérée et la demande traitée. Ayant validé la correspondance d'une ou plusieurs offres, entre autre et essentiellement l'appartenance du trajet (i.e. points de départ et d'arrivée) à l'itinéraire emprunté par le véhicule, etc. celles-ci sont stockées dans une structure de données adéquate. Il s'agit d'un espace alloué pour garder l'ensemble des possibilités (i.e. offres faisables) sur le tronçon considéré avec le paramétrage adéquat pour chaque véhicule considéré.

Une possibilité dite *SP* (*Solution Possible*) dans nos algorithmes est ainsi considérée comme étant :  $SP = (Point\_Depart, Point\_Arrivee, V_j, l_j, Duree, d, a, TD)$ .

Où  $V_j$  est le véhicule correspondant à l'offre considérée sur le tronçon de route  $[Point\_Depart, Point\_Arrivee]$ . Toutes les informations concernant les données de déplacement de  $V_j$  sur l'itinéraire considéré sont ainsi gardées ; à savoir, le nombre de places disponibles, le temps que le véhicule mettra à voyager (inclue  $W_t$  et  $T_t$ ) ainsi que son temps de départ au plus tôt à partir de  $Point\_Depart$ , le temps d'arrivée au plus tard estimé à  $Point\_Arrivee$  et son évaluation sur la base de la fonction *Fitness* préalablement définie et formulée (*TD*).

Cette structure est gardée en mémoire uniquement le temps que dure le processus et peut être (re)visité par un AO pour récupérer une information susceptible de l'intéresser et ainsi lui éviter de faire des calculs inutiles. Toute visite d'un quelconque AO engendrant des actions de mise à jour des données de celle-ci entraîne une modification automatique instantanée pour en assurer la cohérence et l'actualisation. Les bénéfices ainsi engendrés sont comptabilisés en termes de gain de temps, occupation de ressources CPU, de minimisation des risques de conflit au niveau des affectations, etc. Le traitement parallèle des requêtes, que ce soit d'une seule requête par plusieurs AO ou de plusieurs en même temps par le même AO ou par plusieurs est ainsi réalisé de manière synchrone entre toutes les entités intervenantes au niveau du processus global. La synchronisation est essentiellement établie par la coordination des accès des AO aux différentes possibilités gérant l'interrogation ou la mise à jour conflictuelle de la structure mise à leur disposition.

Algorithme 6 : Itinéraire Global Optimisé (IGO)

**Données:**  $R_p$  : Une requête spécifique incluse dans la zone représentée  $Z_p$  incombant à  $AO_p$ ,  $I(t)$ : Matrice des offres,  $t$  : temps système,  $\{\omega_1, \omega_2\}_R$

**Resultat:**  $S_p$  : Solution optimisée globale pour  $R_p$

1. **Initialisation:** Solution  $S_p \leftarrow \phi$ ,  $SP \leftarrow \phi$  : ensemble de solutions possibles
2. **Pour Toute** offre  $o_j$  **telle que**  $o_j \in I(t)$  **Faire**
3.     **Si**  $R_p^+ \in IT_{V_j}(t)$  **Alors**
4.         **Si**  $R_p^- \in IT_{V_j}(t)$  **Alors**
5.             Calculer  $(W_t, T_t, Duree, d, a, Fitness)$
6.             **Si**  $(Vérifier\_Contraintes(d_p, Pl_j, d, a))$  **Alors**
7.                  $SP \leftarrow SP \cup \{(R_p^+, R_p^-, V_j, Pl_j, Duree, d, a, Fitness)\}$
8.             **Fin Si**
9.     **Fin Si**
10. **Fin Si**
11. **Fin Pour**
12.  $S_p \leftarrow Choisir SP_o$  de  $SP$  qui optimise la Fonction *Fitness*
12. Retourner  $S_p$

## IV.8. Conclusion

L'apport principal de l'approche que nous proposons dans cet ouvrage réside essentiellement dans la migration d'un contrôle centralisé vers un contrôle distribué.

Nous avons pour cela défini une architecture distribuée mariant les systèmes multi-agent à une modélisation de graphe dynamique distribué établie sur la base d'algorithmes de

décomposition pour la classification des données réparties sur le réseau de desserte de manière difforme. Cette modélisation définie et détaillée dans le précédent chapitre (*c.f. Chapitre III*) a été à la base de l'architecture multi-agent présentée à cette étape de notre étude. Celle-ci ayant été implantée principalement et uniquement au service de la mise en place d'un système de covoiturage dynamique optimisé et donc incorporant des algorithmes d'optimisation dans le cœur de ses agents.

Les tâches associées au système CODAC ont par la suite été réparties sur les différentes composantes logicielles ainsi définies de par cette architecture incombant chacune d'un rôle bien précis pour l'optimisation du service dans sa globalité. Deux grandes approches relativement à l'optimisation des affectations des voitures aux usagers ont été érigées de par l'étude des méthodes d'optimisation et de recherche de plus court chemin dans des structures de graphes. La première a été inspirée des travaux de Feki (125) et Kamoun (126) sur la recherche d'itinéraires dans un contexte d'aide au déplacement des voyageurs faisant intervenir des moyens de transport comodal pour le premier et multimodal pour le deuxième. Une version modifiée des algorithmes de *Dijkstra* de manière à ce qu'elle soit adaptée à notre problème d'affectation, tout en prenant en compte les différents paramètres et contraintes formulées dans ce chapitre, a ainsi été élaborée.

Par contre, la deuxième a donné lieu à une heuristique spécifique à un objectif que nous nous sommes fixés et où nous faisons intervenir de nouveaux paramètres de par l'intégration d'un nouveau critère d'optimisation relatif au confort de par le nombre de changements à effectuer. En effet, ayant choisi de tolérer une certaine flexibilité dans les trajets afin d'augmenter le taux de réponses positives du système, il n'empêche qu'il serait judicieux de travailler aussi dans le sens contraire. C'est-à-dire de minimiser le nombre des transferts d'une voiture à une autre, ceux-ci s'avérant toujours contraignants pour les passagers.

La formulation du problème définie et nos algorithmes fixés, la concrétisation de nos objectifs est complétée par une implémentation de ceux-ci. Nous présentons alors dans le chapitre suivant les détails relatifs aux outils utilisés pour le développement de la plateforme *CODAC*. Différents jeux de tests ainsi que des détails sur le déploiement y sont fournis aussi.



## V. CHAPITRE V

# IMPLEMENTATION ET DEPLOIEMENT DU SERVICE DE COVOITURAGE DYNAMIQUE

## V.1. Introduction

Après avoir introduit le concept de covoiturage, dans un premier temps, comme étant placé dans un contexte de résolution des problèmes actuels qu'ils soient environnementaux, financiers ou de mobilité ; nous nous sommes focalisés sur les notions théoriques dans un second temps. En effet, un état de l'art des systèmes de transport en général et du covoiturage en particulier a été fourni dans le premier chapitre de ce document donnant lieu à l'énoncé de nos objectifs de par une critique de l'existant mettant en exergue les défaillances des systèmes aujourd'hui opérants en France ou ailleurs. Dans le deuxième chapitre nous nous sommes alors focalisés sur la description des aspects théoriques des choix et outils méthodologiques adoptés. Se sont alors succédées, dans les chapitres III et IV, une définition et une formalisation bien détaillées de la méthodologie de résolution adoptée avec le niveau de spécification requis. Les principaux concepts liés à cette méthodologie ayant été définis et formulés, ce chapitre vient concrétiser toutes ces notions fournissant ainsi le niveau pragmatique requis pour la validation de notre approche.

Toutes les notions et fondements de base relatifs à la phase de développement, notamment l'implémentation et tests, y sont décrits. Nous présenterons pour cela en premier lieu un bref récapitulatif pour reprendre de manière abstraite ce qui a fait l'objet des chapitres précédents avec un focus sur les domaines d'intérêts scientifiques impliqués dans notre travail. Un descriptif des différents outils et de la plateforme utilisés pour le développement de notre application feront par la suite l'objet de la deuxième section. Dans la troisième section, une description élaborée des différentes architectures de déploiement possibles est fournie pour pouvoir décrire celle que nous avons finalement choisie pour le déploiement de notre système et la base sur laquelle ce choix a été fondé. La quatrième section vient en complément à celle qui l'a précédée pour y décrire l'architecture logicielle interne adoptée et qui se base sur une séparation des classes de développement en trois couches essentielles selon le modèle décrit dans cette section. La cinquième section sera consacrée au module GIS de notre système introduit de par l'intégration dans notre application d'un logiciel propriétaire de notre équipe et qui est nommé Cartocom. Ses fonctionnalités principales seront ainsi décrites en concordance avec nos besoins.

Suivront enfin les résultats générés au travers des différents tests effectués sur la plateforme. Celle-ci, comme le montreront les différents jeux de tests, porte ici le nom de **DOMARTiC** : une première appellation que nous avons donnée à notre approche, de l'anglais

*Distributed Optimized approach based on the Multi-Agent concept to set up a Real Time Carpooling service.*

La dernière partie de ce chapitre sera effectivement consacrée à tout ce qui touche à la mise en œuvre du système (*CODAC*) montrant différents scénarii d'exécution de la plateforme ainsi établie (*DOMARTiC*). Par ailleurs, nous avons aussi intégré dans les différents jeux de tests, la manipulation de l'outil logiciel intégré CARTOCOM pour la représentation d'une cartographie réelle du réseau géographique desservi. En dernier, une brève conclusion fera office d'un récapitulatif de toutes les notions détaillées dans ce dernier chapitre.

## **V.2. CODAC : une plateforme de covoiturage dynamique optimisé**

Afin d'aboutir à la présentation de notre système sur le plan pragmatique, nous reprenons brièvement dans cette section les différents aspects relatifs à la méthodologie de résolution adoptée. Celle-ci ayant été introduite dans les deux premiers chapitres, elle a fait l'objet d'une spécification formelle dans les deux suivants. En effet, le covoiturage introduit en premier lieu dans son contexte bibliographique a montré un succès notoire vu l'intérêt considérable qui y a été porté. Variant d'environnementaux à financiers, les impacts d'un tel usage ont été remarquablement riches en apport aidant à réduire la pollution, améliorant ainsi la qualité de la vie et réduisant aussi les budgets alloués au transport ; ce qui n'a pas été sans bénéfices pour les individus utilisant ces services.

### **V.2.1. CODAC : Une solution dynamique attractive**

Le concept de la voiture partagée a connu un succès de grande envergure vu les répercussions positives dont il a fait preuve sur plusieurs niveaux.

Concept à l'air du temps, le covoiturage a ainsi fait d'emblée son entrée dans les domaines de la recherche et de l'industrie. Plusieurs travaux ont été développés dans ce sens impliquant le secteur public ou privé, des projets régionaux ou autres et des investissements importants à l'échelle de l'état. Par ailleurs accablées de critiques assaillantes, plusieurs de ces approches ont connu un échec déplorable et n'ont jamais été opérationnelles. D'autres ont en contrepartie été fonctionnelles et ont abouti à des systèmes opérationnels et réussis. Cette réussite reste toujours relative comme le montre la critique que nous en avons faites (*c.f. §I.5.2.4.*). Ces systèmes présentent quelques lacunes, qu'il s'agisse des contraintes de temps pour les réservations mettant en cause l'aspect purement statique de ces systèmes, du manque

d'optimisation ou encore de la défaillance marquée de plusieurs de ces systèmes de par l'absence d'automatisation des tâches, etc.

Projetant de bénéficier de l'apport du covoiturage et marqués par ces différents aspects dénonçant les défaillances des systèmes existants, nous avons positionné nos travaux dans un concept de jointure entre l'optimisation formelle et celle pratique. Par ailleurs, nous nous sommes particulièrement intéressés à l'intégration de l'aspect temps réel dans ce concept. Les systèmes placés dans ce cadre étant très attractifs, ils sont généralement réussis mais sous l'unique contrainte de faire preuve des performances nécessaires. Aspirant à une telle attractivité, nous nous sommes volontairement placés dans ce contexte pour considérer la dynamique des environnements de transport. En contrepartie, ces considérations exigent de trouver le bon compromis aussi bien théorique que pratique pour demeurer compétitif sur le marché et anéantir des répercussions négatives probables de celles-ci. En effet, Un environnement mobile et instable rend encore plus dur le concept d'optimisation dans des considérations d'efficacité du service fourni.

Placée en ligne de mire tout au long de ces travaux, la complexité de la problématique traitée a orienté nos recherches vers des domaines où creuser en valait la peine.

### V.2.2. Domaines d'intérêts scientifiques

Afin de rétablir l'équilibre entre l'aspect amélioratif et les exigences pour la satisfaction et la faisabilité de point de vue pratique, nous avons pu considérer plusieurs concepts théoriques ainsi que les outils nécessaires pour en bénéficier. Un choix méthodologique et stratégique a ainsi été le fruit d'efforts dépensés dans ce sens pour l'élaboration de cette thèse. Nos travaux ont alors été orientés dans cette direction et ont naturellement abouti à une combinaison des deux domaines d'optimisation et du paradigme agent pour atteindre l'optimalité escomptée. En plus, cette thèse a été menée dans une équipe de recherche où les thèmes de systèmes multi-agent et d'optimisation sont imposants. En ce sens, le choix de l'alliance des SMA et de l'optimisation s'est avéré être un choix judicieux. En effet, l'étude du problème de covoiturage dynamique optimisé a révélé une importante complexité et par la suite une difficulté de résolution qui peut induire à des défaillances peut être plus importantes que celles évoquées dans la littérature. La méthodologie ainsi adoptée travaille dans le sens de réduire la complexité du problème, offrir le niveau d'automatisme requis, fournir des réponses positives optimisées à un maximum d'utilisateurs tout en respectant les délais et rester dans un

cadre d'optimalité et d'une qualité de service que nous avons voulu à la hauteur des systèmes qui fonctionnent en temps réel.

Cette méthodologie est jugée efficace d'autant plus qu'elle porte sur une modélisation du réseau géographique de desserte adoptant les graphes dynamiques distribués. Toutes ces notions menées d'écho ont été mises à profit pour mettre en place un processus décentralisé à travers la mise en place d'un système Informatique où une multitude d'agents logiciels impliqués travaillent en parallèle pour effectuer chacune une tâche de complexité moins importante que celle du problème initial.

Dans son alliance avec les systèmes multi-agent et s'appliquant de surcroît sur une modélisation en GDD du réseau, l'optimisation fait intervenir plusieurs domaines notamment relatifs à l'optimisation distribuée. Dans une aire où l'Intelligence Artificielle Distribuée est à son apogée et où les Nouvelles Technologies de l'Information et de la Communication (NTIC) ont fait un pic dévolution révolutionnaire, nous avons déployé des efforts d'analyse et de conception de sorte à en profiter. Nous classons ainsi nos travaux à un point de rencontre de domaines variés mais très évolués et l'approche à laquelle nous avons abouti fait appel à différents concepts. Ceux-ci, mis en œuvre de par les différentes étapes élaborées pour la réalisation de nos objectifs préalablement fixés, font à chaque fois intervenir un ou plusieurs domaines d'intérêt. Ces derniers conduisent dans l'ensemble à un système d'aide à la décision mettant à profit l'IAD pour la recherche d'affectations optimisées dans un contexte de covoiturage dynamique en faveur de la concrétisation du projet de développement durable (Figure V.1).

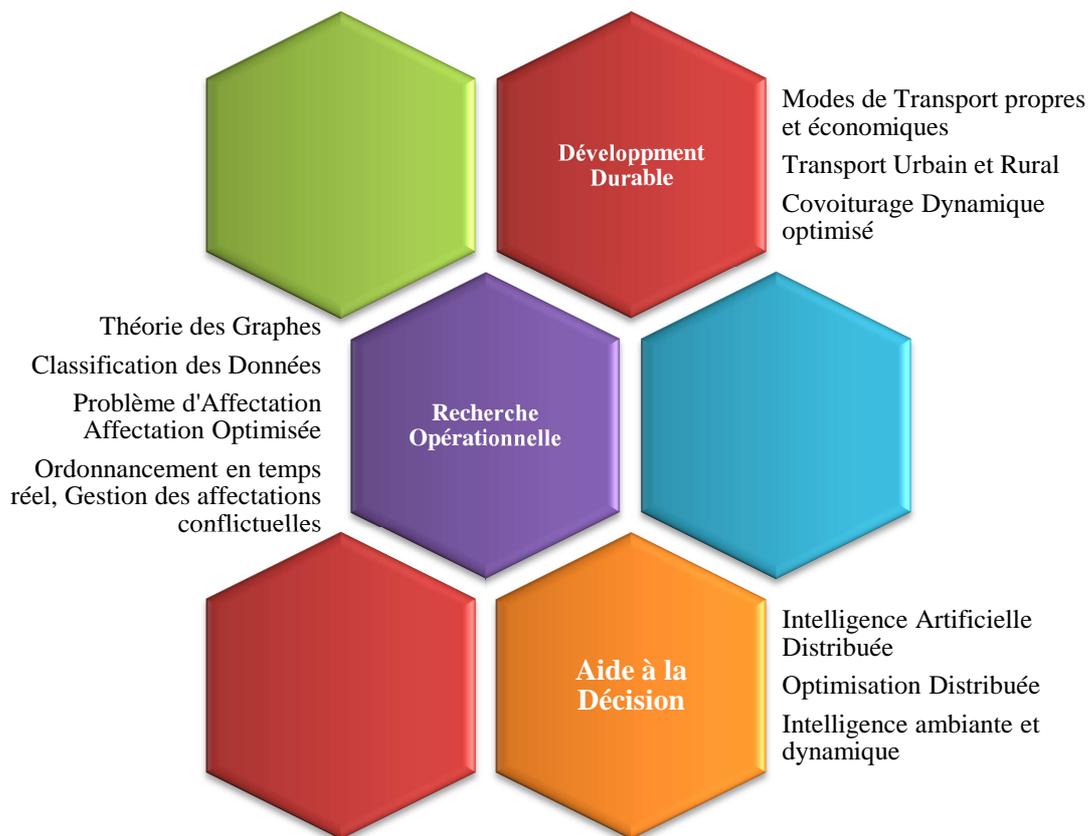


Figure V.1. CODAC : Une panoplie de domaines scientifiques

Le système de covoiturage tel que visionné de par notre approche fait intervenir une multitude de domaines scientifiques assez variés et très divers de points de vue zones d'intérêt. Consacrés tous au niveau de l'analyse, la conception et la modélisation de notre système, ils se réunissent sous une seule thématique qu'est l'intelligence artificielle distribuée (*IAD*) pour une mobilité avancée.

En relation avec la modélisation, la méthodologie de résolution choisie ou encore le développement et déploiement du système, les outils et méthodes consacrés font référence à un large panel de domaines scientifiques. Selon les exigences, nous nous sommes intéressés à l'un ou l'autre de ces domaines avec des degrés d'investissement relativement importants. Ces domaines ont ainsi constitué des centres d'intérêt plus ou moins élaborés dans notre vision du problème et l'implication dans ce contexte a consacré des notions théoriques et pratiques pour l'optimisation en Recherche Opérationnelle (*RO*). Cette dernière est la discipline des méthodes scientifiques perçues et utilisées comme le chemin menant à l'élaboration de meilleures décisions. Le mérite lui revient d'avoir fait avancer les recherches dans les fonctionnements et architectures des systèmes de production et d'organisation pour leur rationalisation, simulation et optimisation. La *RO* où s'inscrit notre problématique s'est

imposée en ce sens comme une discipline carrefour associant les mathématiques, l'économie et l'informatique. C'est pour cette raison qu'elle a été considérée comme révolutionnaire et a aidé à faciliter la résolution de beaucoup de problèmes. La *RO* a ainsi été naturellement adoptée en prise directe sur l'industrie et a joué et joue encore un rôle-clé dans le maintien de la compétitivité.

### V.2.3. CODAC : Le débriefing

Après avoir présenté les tenants et les aboutissants expliquant nos choix pour l'adoption d'une approche de covoiturage dynamique, nous nous sommes intéressés par la suite à en expliquer les concepts et fondements résumés dans la Figure V.2.

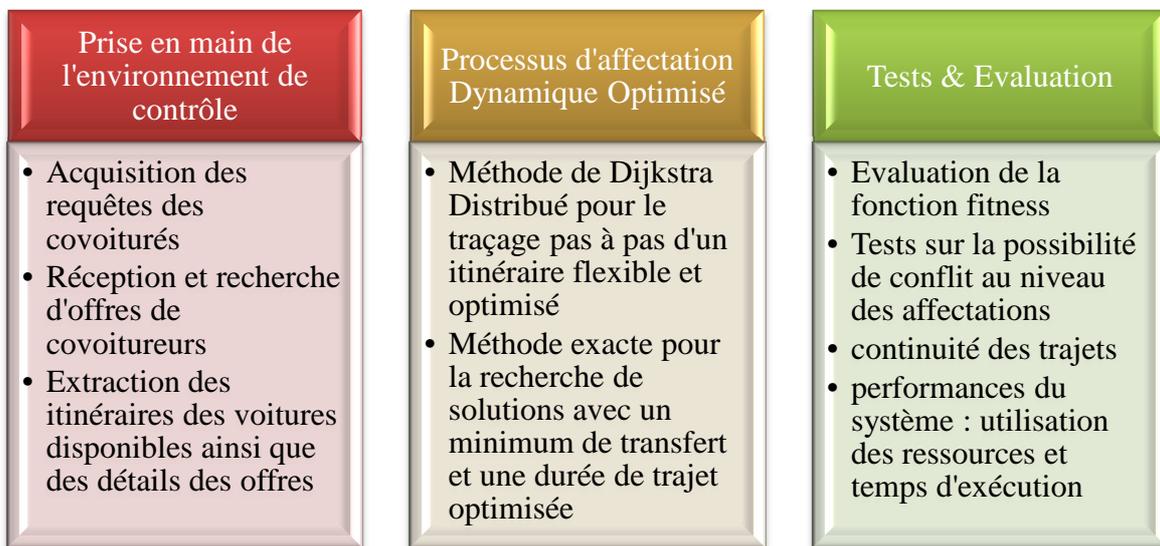


Figure V.2. Différents aspects pour l'élaboration de CODAC

Cette figure montre par ailleurs une vue globale des différentes étapes de réalisation de nos travaux où trois essentiellement sont à noter :

#### A. *Prise en main de l'environnement de contrôle*

La première étape que nous avons nommée *prise en main de l'environnement de contrôle* ou tout simplement *prise de contrôle de l'environnement* concerne la nécessité d'adopter une modélisation adéquate de l'environnement de covoiturage suivant son mouvement. Il est effectivement impératif de pouvoir adopter la bonne représentation du terrain desservi vu la rapidité de l'évolution dont il est marqué dans un contexte de temps réel ainsi que l'instantanéité des acquisitions des données avec l'avènement des nouvelles technologies qui ont révolutionné le monde des réseaux. Après avoir réceptionné toutes les données, l'extraction des itinéraires restant à poursuivre est une fonctionnalité qui incombe à un

module spécifique de notre système. Il s'agit de l'intégration du logiciel CARTOCOM dont nous détaillons ultérieurement les différentes spécificités et contributions, essentiellement techniques, dans notre système. Les fonctions élaborées via ce module intégré constituent l'aspect *GIS (Geographical Information System)* de notre système. Ces fonctionnalités intégrées étant basées sur des outils GPS, elles concernent essentiellement la capture des positions des usagers, le traçage de leurs itinéraires respectifs, etc.

### ***B. Processus d'affectation dynamique optimisé***

Dans cette deuxième étape nous nous sommes appliqués à chercher la bonne stratégie de résolution pour le problème de covoiturage dynamique tout en mettant à profit la modélisation établie au travers de la première étape. Le principe des graphes distribués dynamiques aidant et dans l'optique de réduire la complexité exponentielle du problème de CDO, nous avons investi tous nos efforts dans la mise en place d'un processus décomposé, décentralisé sur plusieurs composantes logicielles. Point de rencontre de plusieurs disciplines réunis sous la seule thématique de l'intelligence artificielle distribuée, notre approche fait appel à une alliance des SMA et de l'optimisation pour la concrétisation de la décentralisation du processus à laquelle nous aspirons. Placée sous le signe de l'efficacité et de l'optimalité aussi bien technique (par rapport aux propriétés d'exécution) que pratique (par rapport aux traitements et objectifs d'amélioration de la qualité de service), cette alliance a été considérée et mise en œuvre au travers de deux approches. Celles-ci étant spécifiques au problème de covoiturage dynamique optimisé distribué, se présentent comme une adaptation des algorithmes de Dijkstra pour la première et une approche dynamique pour l'affectation optimisée (*ODAVE*) pour la deuxième.

### ***C. Tests et Evaluation***

Tout au long des textes précédents nous avons pu aborder notre problème du point de vue théorique aussi bien que pratique. Nous avons ainsi introduit les concepts et méthodes adoptés et dont nous avons pris soin de bien expliquer le choix. Évaluée au fur et à mesure de l'élaboration de ce document, notre approche a jusqu'ici été validée sur le plan théorique. En effet, étant positionnés dans une jointure d'un mode personnel mais en même temps partagé, nous profitons d'un avantage à l'aube de la complémentarité entre tous les moyens de transport et essentiellement de l'intégration du transport comodal comme mode à part entière.

Par ailleurs les aspects de temps réel, d'optimisation, de SMA et de GDD font un mélange quelque peu 'contradictoire'. Les deux premiers allant à eux seuls à l'encontre d'un procédé

efficace et donc d'un système performant, les trois derniers viennent y remédier de par le mariage des systèmes multi-agents et de l'optimisation sur la base d'une solution de modélisation inspirée des GDD. Les SMA connus pour leur résolution des problèmes complexes qui ne peut se faire par les systèmes monolithiques, apportent donc un grand plus à l'efficacité voulue du système. Par ailleurs, cette efficacité est d'autant plus importante que les algorithmes proposés dans les deux approches sont exécutés de manière distribuée, les SMA aidant à établir des liens de coordination via une coalition des agents évoluant dans notre système. Au détour d'une optimalité de la qualité de service placé sous le thème du « multithreading », le problème initial a ainsi été décomposé en plusieurs tâches de complexité négligeable et a intuitivement donné lieu à des processus légers s'exécutant en parallèle. Aussi, les algorithmes de Dijkstra sont bien connus pour être polynomiaux et apportent donc une garantie sur l'optimalité des temps de réponse du système.

Pourtant n'est valable qu'une démonstration pratique apportant la crédibilité nécessaire par rapport à l'ensemble de ces énoncés restés jusqu'ici au stade de la validation théorique. Nous nous sommes donc employés à mettre en œuvre tous les concepts précédemment détaillés et formulés dans une phase ultime de notre travail. Celle-ci concerne le développement de notre système et la génération de tests pour en valider l'opérabilité dans des conditions propices à l'optimisation et au temps réel.

Nos efforts ont été récompensés par la mise en œuvre d'une plateforme multi-agent opérationnelle pour le covoiturage dynamique optimisé. Celle-ci a été testée et validée sur des exemples de données 'à priori' synthétiques, mais qui concernent toutefois des informations sur des données (i.e. adresses) réelles (qui existent sur la carte géographique). Seulement les demandes et offres de covoiturage visualisées dans ces tests relèvent de notre propre choix et les données fixées de notre seul essor. Le système ici élaboré n'a pas encore été mis sur le marché et nécessite une période plus ou moins longue pour pouvoir être testé réellement sur des individus avec des besoins réels.

Avant de présenter les exemples et scénarios d'exécution de notre approche, il convient de présenter tout d'abord les outils dont nous avons fait usage pour réaliser ce support (DOMARTiC). Notamment et essentiellement la plateforme de développement ; nous en présenterons quelques-unes et choisiront celle qui est la plus en adéquation avec notre problème. Aussi nous évoquerons l'outil logiciel CARTOCOM avec un bref descriptif de ses principales fonctionnalités pour en expliquer l'utilité et à quelles fins exactement nous en avons fait usage.

## V.3. Plateforme de développement Multi-Agent

Une fois l'architecture du système choisie, nous nous intéressons au processus de développement et la mise en œuvre de celui-ci. Nous avons en effet présenté dans les différents chapitres nos principaux centres d'intérêt et qui se résument en une alliance des méthodes d'optimisation et des SMA. Une architecture multi-agent particulière et spécifique à notre problématique est ainsi née faisant intervenir une multitude d'entités logicielles dont nous citons particulièrement les *Agent Décomposant* et *Optimiseurs (AD et AO)* auxquels les principales fonctions d'optimisation incombent.

### V.3.1. Processus de développement : Agent ou objet ?

Pour ce qui est du développement de notre approche, nous nous sommes intéressés aux différentes étapes composant le processus de développement d'un système multi-agent. Par ailleurs, une similitude est à noter dans le processus de développement des systèmes orientés objets et celui des SMA. Ceci vient rejoindre l'architecture statique donnée à la Figure IV.11 du précédent chapitre (*c.f. Chapitre IV. Figure IV.11*) qui représente un mélange entre le paradigme agent et l'orienté objet. D'objet à agent il n'y a qu'un pas ; en effet la seule différence réside dans la quatrième phase du processus de développement, les deux paradigmes étant en effet composés de quatre phases chacun (127). Cette ultime phase se réfère à la dernière étape de maintenance pour les structures orientées objets, et à une phase de déploiement dans le cas des SMA.

Les trois premières phases relatives à l'analyse, la conception et le développement sont pratiquement équivalentes aux systèmes usuels alors que la dernière diffère largement d'un paradigme à un autre. Elle est de ce fait consacrée dans les systèmes orientés objets à la vérification de la bonne marche du système dans le sens voulu, c'est-à-dire qu'elle aura pour effet de corriger les anomalies probables émergées de par différents tests de différents scénarii et d'intégrer ainsi les évolutions souhaitées par l'utilisateur. A l'opposé, une phase de déploiement s'impose dans la conception d'un SMA (128). Elle a pour effet de définir l'évolution des agents (i.e. création de nouveaux agents, modification, suppression ou destruction) au cours de l'exécution du système. La Figure V.3 reprend les quatre phases principales du processus de développement dans des systèmes orientés objets et les SMA.

En concordance avec nos projets de mise en place d'un système basé sur les agents, nous penchons naturellement pour un processus de développement des SMA. Le concept agent

englobe par ailleurs toutes les notions relatives au concept objet en y rajoutant les fonctionnalités de communication via l'établissement de collaborations ou négociations pour la coordination de leurs actions... Un autre avantage des SMA se réfère à la dynamique de l'architecture, c'est-à-dire la création et destruction des agents au besoin. Dans ce contexte, un *AIU (Agent Interface Utilisateur)* par exemple n'a lieu d'être que si un piéton se connecte au système et émet le désir de voyager de par l'envoi d'une requête de covoiturage. Nous avons mis à profit toutes ces notions de par la proposition d'une architecture multi-agent concordant avec notre problème.

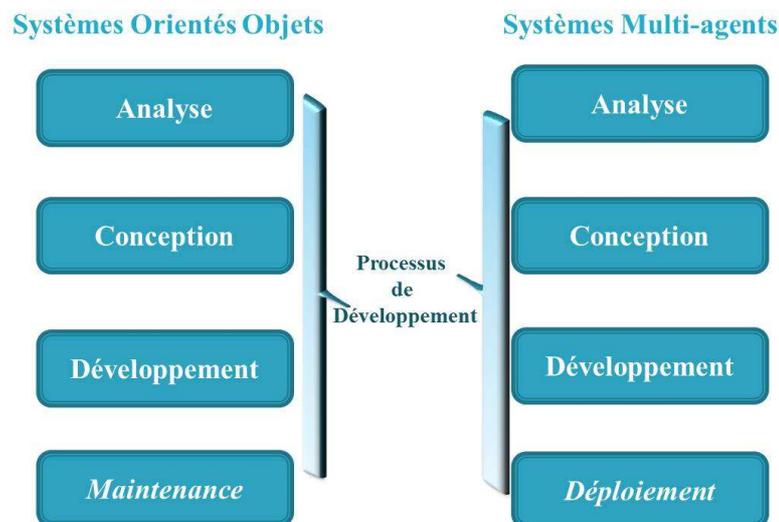


Figure V.3. SMA VS Systèmes Orientés Objets (2)

La contribution des SMA s'est aussi manifestée de par la proposition des deux approches essentiellement basées sur la coalition inter agent mettant en exergue l'intelligence collective dont ils ont fait preuve de par l'implémentation d'une optimisation distribuée. Reste à voir se concrétiser l'efficacité de ces notions dans un volet pratique déployé au travers d'une plateforme logicielle développée à cet effet. Pour ce faire, nous nous sommes intéressés aux différentes plateformes de développement intégrant les modélisations agents et notre choix a porté sur une spécifique que nous décrivons dans la suite. Par ailleurs, avant d'expliquer notre motivation à adopter ce choix spécifique, il convient de procéder à une présentation et description des principales plateformes multi-agent existantes.

### V.3.2. Différentes plateformes de développement

Dans cette section, nous présentons quelques-unes des plateformes multi-agent pouvant être utilisées pour l'implémentation de notre système. Il existe dans ce contexte une multitude de plateformes telles que :

### V.3.2.1. CORMAS :

Abréviation de *COmmon Resources Multi-Agent System*, CORMAS est une plateforme libre (open source) basée sur SmallTalk comme langage de programmation pour le développement de systèmes multi-agent. Spatialisée, cette plateforme est plutôt orientée vers des problématiques de recherche en sciences du développement et de négociation entre acteurs.

### V.3.2.2. Zeus

Il s'agit d'une plateforme logicielle (129) développée par British Telecom System Research Lab où des systèmes collaboratifs peuvent être élaborés en quatre phases : analyse, conception, réalisation et support à l'exécution. Elle définit les rôles au début de l'élaboration, pour ensuite décrire l'organisation et enfin la coordination. Elle présente par ailleurs l'inconvénient majeur d'être très complexe et difficile de maîtrise et nécessite donc de longs apprentissages.

### V.3.2.3. Jade

Acronyme de *Java Agent DEvelopment*, *JADE*<sup>37</sup> est un framework basé sur le langage *Java* initié pour le développement des SMA et répond aux normes *FIPA*. A ce stade, il convient tout d'abord de définir la norme *FIPA*<sup>38</sup> (Foundation for Intelligent Physical Agents) avant de procéder à la description de *JADE*.

Etant donnée la diversité remarquable des travaux de recherche touchant au domaine du multi-agent, une normalisation s'imposait et la *FIPA* a ainsi été initiée, en 1996, comme projet de normalisation du concept agent. Ce projet a été mis en place, et plein d'autres encore, en vue d'instaurer une interopérabilité tangible entre agents et concepteurs. A signaler dans ce contexte, les projets *AUML*<sup>39</sup> (*Agent UML*) et *FIPA* à laquelle nous nous intéressons particulièrement. Celle-ci promet en effet une meilleure interaction et communication inter agents via des moyens standardisés, sans pour autant porter atteinte à leur sens initial.

Pour en revenir à *JADE*, développé à l'Université de Parme en Italie, celui-ci répond effectivement aux normes de la *FIPA* et utilise donc le langage *FIPA ACL* pour

---

<sup>37</sup> <http://jade.tilab.com/>

<sup>38</sup> <http://www.fipa.org/>

<sup>39</sup> <http://www.auml.org/>

l'établissement de liens de communication entre ses agents. *JADE* est ainsi composé de trois modules notables dans le cadre d'une conformité à la norme *FIPA* :

Le '*Director Facilitator*' (DF) : est un registre centralisé d'entrées souvent associé à un service de pages jaunes (annuaire téléphonique). Les agents fournissant un service s'y inscrivent et ceux qui ont besoin d'un service en particulier y font la recherche correspondante<sup>40</sup>.

Le '*Agent Communication Channel*' (ACC) a pour fonction de gérer les communications entre les agents.

Les '*Agent Management System*' (AMS) sont responsables de la supervision des agents : leur enregistrement, authentification, accès et utilisation du système.

#### V.3.2.4. *Madkit*

Créé en 1996 par Olivier Gutknecht et Michel Ferber au LIRMM (*Laboratoire d'Informatique, de Robotique et de Micro-électronique de Montpellier*), *Madkit*<sup>41</sup> (Multi-Agent Development Kit) est une plateforme multi-agent basée sur un modèle organisationnel de type (Agent/Groupe/Rôle). Entièrement basée sur le langage Java, elle assure les liens de communications via un mécanisme de *peer to peer* (P2P). En étant fondée sur les principes du paradigme agent, elle permet donc un développement rapide des applications distribuées. *Madkit* fait usage de plusieurs outils mis à profit des concepteurs et développeurs parmi lesquels l'on évoquerait nécessairement l'interface graphique permettant de supporter différents modes d'utilisation et d'exploitation de la plate-forme

### V.3.3. JADE : le choix d'une plateforme

Comme nous l'avons signalé dans la précédente section, les usages sont multiples et les plateformes sont autant variées, se focalisant sur l'un ou l'autre des principes des SMA orientant les choix d'implémentation. Pour en décider, nous avons fixé les objectifs recherchés au niveau de l'implémentation en fonction des objectifs de réalisation d'un SMA tel que implanté dans le cadre théorique de notre approche. Celle-ci étant spécialement focalisée sur le caractère collaboratif entre les différentes entités logicielles intervenant dans le processus global à assurer dans le cadre de la gestion des demandes et offres de covoiturage, l'affectation optimisée dynamique des véhicules aux usagers profondément

---

<sup>40</sup> <http://www.iro.umontreal.ca/~vaucher/Agents/Jade/primer5.html>

<sup>41</sup> <http://www.madkit.org>

marqué par ce caractère a servi à orienter nos choix. Le procédé d'optimisation est en effet lui-même fondé sur un principe de coalition intégrant la notion de communication à part entière comme moteur principal pour la mise en place des algorithmes spécifiés. Ceux-ci étant fondés sur la coordination entre les agents et plus particulièrement encore sur une collaboration des différents *Agents Optimisateurs (AO)*, cette collaboration a initié des liens forts pour la synchronisation de leurs processus. Elle a ainsi été initiée à des fins de réalisation de l'efficacité escomptée. Consolidée par les échanges et communications via des messages spécifiques, l'optimisation vue sous cet angle fait ainsi le fondement de notre approche. Les liens ainsi établis entre les agents de notre système aident à mener à bien le procédé d'affectation tout en maintenant l'exécution, dans le sens des résultats générés, à l'abri des conflits.

Dans cette optique de réalisation de nos projets d'implémentation, la décision portant sur l'adoption de la plateforme JADE comme outil de développement s'avère être un choix judicieux qui épouse parfaitement nos idées et principes. Celle-ci étant l'un des outils les plus en concordance avec nos objectifs et la méthodologie de résolution que nous avons adoptée. De par l'utilisation de l'outil JADE, nous mettons de ce fait à profit les avantages normatifs qu'il exhibe en plus des mécanismes consacrés pour mener à bien les liens de communication. JADE consacre effectivement les normes essentielles (*FIPA, FIPA ACL*) pour la bonne mise en œuvre du choix stratégique que nous avons adopté pour l'élaboration de notre approche favorisant de ce fait la communication au service de la coordination et synchronisation des actions de nos agents.

D'un autre point de vue, la nécessité d'assurer l'interopérabilité ainsi que la compatibilité de notre système avec d'autres approches vient consolider ce choix. Nous parlons essentiellement dans ce contexte des travaux réalisés ou en projet d'être réalisés au sein de notre équipe (*OSL : Optimisation des Systèmes Logistiques*) en particulier et du laboratoire (*LAGIS : Laboratoire d'Automatique, de Génie Informatique et Signal*) où ces travaux ont été conduits de manière plus générale. L'objectif principal visé de par plusieurs des travaux de recherche élaborés dans le laboratoire *LAGIS* concernent en effet le développement d'une plateforme générique regroupant plusieurs services qui s'inscrivent dans le domaine du transport. Afin d'optimiser l'interopérabilité avec ces services, le plus judicieux serait d'assurer une conformité à une norme de développement multi-agent.

En effet, menée de pair avec d'autres projets, cette thèse s'inscrit dans un contexte de complémentarité avec ceux-ci, les premiers ayant déjà été réalisés traitent de services d'aide

au transport de personnes. De grandes thématiques dans ce domaine telles que le TAD, les SIAD pour la recherche d'itinéraires optimisés et la gestion des perturbations ont déjà été traités. Plus particulièrement encore, deux projets, le premier réalisé par M.F. Feki et l'autre en cours de réalisation par K. Jeribi s'inscrivent dans un contexte de transport comodal et dans lesquels un module entier est consacré à la voiture partagée. Orientés dans le sens de la comodalité et de l'EcoPartage, ces travaux intègrent notamment le concept de covoiturage à part entière dans les systèmes de transport tel que énoncé par Jeribi ou en complémentarité avec d'autres pour combler les besoins restés insatisfaits en cas de perturbation par exemple comme c'est le cas pour Feki.

Enfin, pour résumer et décrire nos principales motivations à adopter *JADE* comme plateforme de développement de notre approche *CODAC* (ou *DOMARTiC*), compatibilité, portabilité, interopérabilité, échanges et communication inter agents sont les maîtres mots. Les avantages décisifs de *JADE* dans ce contexte, implémenté entre autre entièrement en Java nous a aidé à trancher et faire le bon choix.

### V.3.4. Spécificités de la plateforme JADE

Dans cette section, nous définissons les particularités de la plateforme *JADE*. Nous décrivons alors dans la suite le fonctionnement de celle-ci ainsi que la déclaration des agents.

#### V.3.4.1. Principe des conteneurs des agents

Est appelée '*conteneur*' (*container* en anglais) chaque instance de *JADE* qui s'exécute de manière simultanée sur un ou plusieurs postes différents. Un conteneur spécial s'appelle le '*main container*' pour '*conteneur principal*' et il en existe toujours un et un seul dans chaque plateforme. Le reste des conteneurs de la plateforme doivent se déclarer dans ce conteneur spécial qu'est le *main container*. L'ensemble des conteneurs actifs constituent ainsi la plateforme agent. Toutes ces notions sont illustrées au travers de la Figure V.4, et nous pouvons y voir aussi que chaque conteneur principal (*main container*) instancie dès lors qu'il démarre deux types d'agents. Ces derniers interviennent pour reprendre deux des exigences de la FIPA et il s'agit de :

#### A. L'agent AMS

*AMS* est l'acronyme de '*Agent Management System*' qui constitue un agent dont la principale fonctionnalité est d'attribuer des noms différents à chacun des agents créé dans la

plateforme. Il s'agit en effet du service de nommage qui est de l'obligation de l'AMS de le fournir.

### B. L'agent DF

C'est le DF ou 'Directory Facilitator' qui se doit d'offrir le service de pages jaunes défini précédemment. Chaque agent passe à travers le DF pour pouvoir consulter la liste des entrées (entités enregistrées dans celui-ci) et ainsi récupérer la liste des services offerts par les agents actifs de la plateforme ou bien chercher un service spécifique susceptible de répondre à son besoin.

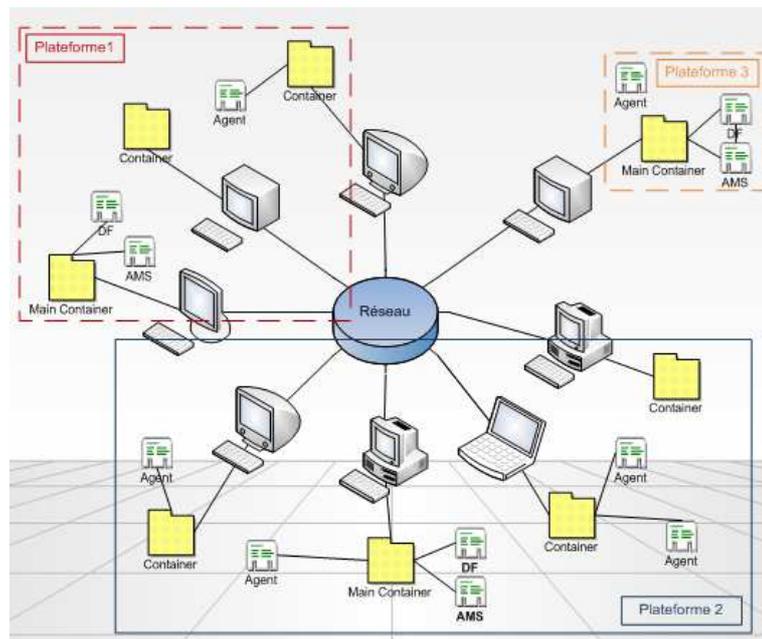


Figure V.4. Plateforme et conteneur JADE (2)

#### V.3.4.2. Les agents dans JADE

Dans cette section, nous définissons particulièrement tout ce qui se réfère à la déclaration et à la spécification des agents dans la plateforme JADE, notamment leurs structures et les principales méthodes implémentées dans le cœur des agents.

Dans l'outil que nous avons choisi pour l'implémentation de notre système, un agent est une classe spécifique qui hérite d'une classe mère implémentée dans la plateforme JADE sous le nom de '*jade.core.Agent*'. Dans cette classe, il doit impérativement exister une méthode spéciale appelée '*setup()*' et implémentée différemment pour chaque type d'agent selon le besoin de spécification pour l'agent en question. Elle concerne effectivement la partie relative à l'initialisation de chaque agent correspondant à celle-ci et s'exécute lors de son

instanciation. Dans la Figure V.5 est donné le fragment de code Java relatif à un exemple d'implémentation de cette méthode pour la déclaration d'un agent dans la plateforme JADE. Cet exemple est relatif à une des manières possibles pour l'initialisation d'un *Agent Décomposeur AD* dans notre approche.

```
import jade.core.Agent;
import carpooling.agents.GenAgent;

public class AgentDecomposeur extends GenAgent {

    Logger logger = LoggerFactory.getLogger(AgentDecomposeur.class);

    public static int Diametre = 6000;
    public Zone TabZone[];
    Chrono chrono = new Chrono();

    protected void setup() {
        super.setup(this, AgentDecomposeur.class.getName(), false);
        logger.info("[AgentDecomposeur] je suis l'agent Décomposeur, je crée les zones et je distribue les requêtes aux AO");
    }
}
```

Figure V.5. Exemple d'initialisation de l'agent décomposeur

Outre cette méthode dont la fonction est d'initialiser un agent, une autre spécifique à la plateforme JADE et pré-implémentée dans le package `jade.core` est appelée '`getAID()`' peut être appliquée à n'importe quel agent instancié dans la plateforme. Elle permet d'obtenir son identifiant comme une instance de la classe `jade.core.AID`. Seul cet identifiant permet de faire référence à l'agent, puisque contenant un nom unique associé à celui-ci. Dans la Figure V.6 est fourni un exemple d'utilisation de celle-ci pour un exemple d'initialisation de l'*AIV* appelé ici *CarAgentAssistant*.

```
import jade.core.Agent;
import carpooling.agents.GenAgent;

public class CarAgentAssistant extends GenAgent {

    protected void setup() {

        System.out.println( "Je suis l'agent"+getAID().getName()+ " et je suis prêt");
    }
}
```

Figure V.6. Exemple d'utilisation de la méthode `getAID`

Dans ces deux exemples, nous pouvons noter que nous avons fait appel au package `carpooling.agents` : un package spécifique qui contient tous les agents que nous avons utilisé dans notre approche et qui sont relatifs naturellement à l'architecture multi-agent énoncée,

décrite et détaillée dans le précédent chapitre (*c.f. Chapitre IV*). Par ailleurs, nous pouvons remarquer aussi que les agents ici instanciés sont implémentés héritant d'une classe mère appelée *GenAgent* que nous avons-nous même implémentée par héritage de la classe *Agent* de *jade.code.Agent*. Dans *GenAgent* sont implémentées des prototypes de méthodes que nous utilisons régulièrement dans les agents de notre système.

En outre, pour définir les actions pouvant être entreprises par les agents, il y a ce qu'on appelle les '*behaviours*' (ou comportements des agents) dont les prototypes de base sont implémentés dans un package spécifique de JADE nommé '*jade.core.behaviours*'. Les *behaviours* permettent donc aux agents d'agir dans une plateforme ; chaque *behaviour* (comportement) définit de manière spécifique les tâches à réaliser en fonction du temps (à quel moment exactement) et des contraintes. Tous les '*behaviour*' héritent d'une classe mère nommée '*jade.core.behaviours.Behaviour*' et possèdent implémentés dans leurs *corps* au moins deux méthodes essentielles :

- « *action()* » qui est une méthode dont le rôle est de définir les actions qui seront lancées dès que le '*behaviour*' sera invoqué.
- « *done()* » est la méthode indiquant l'état du '*behaviour*'. Elle a pour valeur de retour un booléen qui spécifie si le '*behaviour*' a terminé son exécution ou non.

D'autres méthodes comme *Block()*, *getBehaviourName()*, *onStart()*, *onEnd()*, *reset()* ou encore *restart()*... sont aussi spécifiques au '*behaviours*'.

Les '*behaviours*' comptent deux types principaux dans l'ensemble mais sont largement plus variés :

#### **A. *One-shot behaviours***

Les actions implémentées dans ce '*Behaviour*' s'exécutent une et une seule fois. De ce fait après l'appel à sa méthode « *action()* », ce '*Behaviour*' se termine immédiatement. La méthode « *done()* » est implémentée au préalable dans le *jade.core.behaviours.OneShotBehaviour*. Le fragment de code Java dans notre plateforme (Figure V.7) montre un exemple où un '*OneShotBehaviour*' est invoqué dans la méthode '*SetUp()*' de l'*Agent Information (AI)* de notre système. La méthode « *action()* » a pour effet de lancer l'affichage d'un simple message qui s'exécuterait si l'*AI* est invoqué. Par ailleurs, nous pouvons très bien à travers ce fragment que outre ce '*behaviour*', il en existe d'autres tels que ceux importés dans cet exemple : les '*TickerBehaviour*' et '*WakerBehaviour*'.

```

import carpooling.agents.GenAgent;
import jade.core.behaviours.OneShotBehaviour;
import jade.core.behaviours.TickerBehaviour;
import jade.core.behaviours.WakerBehaviour;

public class AgentInformation extends GenAgent {

    Logger logger = LoggerFactory.getLogger(AgentInformation.class);

    protected void setup() {
        logger.info("[AgentInformation] je suis l'agent Information, je reçois les requêtes et offres et je
        fournis les itinéraires préalables à l'agent décomposeur");
        // l'ajout d'un one-shot behaviour pour afficher un Hello world
        addBehaviour(new OneShotBehaviour(this) {
            public void action() {
                logger.info("Bonjour tout le monde je suis l'agent "+ getLocalName());
            }
        });
    }
}

```

Figure V.7. Un exemple d'implémentation du « OneShotBehaviour » dans l'AI

Par ailleurs il existe un autre Behaviour important à noter et c'est le 'Cyclic Behaviours'.

### B. Cyclic behaviours

Sa spécificité à lui c'est qu'il ne se termine jamais. En effet, les *behaviours* répondant à ce type s'exécutent en boucle, de manière continue, jusqu'à ce que l'agent où ils sont implémentés meure. Ces comportements sont généralement utilisés dans des situations spécifiques où le 'CyclicBehaviour' dont le fonctionnement est assimilé à celui d'un serveur, se doit de vérifier en continu les messages reçus. La vérification porte essentiellement sur la forme et le destinataire de ces messages et si les critères en sont conformes, alors cela signifie qu'il possède bien les compétences nécessaires pour pouvoir les gérer. A partir de ce moment-là, il commence donc à traiter le ou les message(s) conforme(s) et renvoie une réponse à leur expéditeur. Un 'OneShotBehaviour' qui s'exécute sous certaines conditions peut envoyer un message à un 'CyclicBehaviour' en attente.

Dans l'optique de la mise en place d'un système d'économie en ressources, une particularité des systèmes multi-agent est que toute entité ne possédant pas de comportement à exécuter est mise en instance. Dans cet aspect des choses, le *thread* (processus léger) qui gère l'agent en question 's'endort'. Ceci pour éviter de consommer inutilement des ressources CPU (temps processeur). L'agent concerné n'est 'réveillé' que lorsqu'il y a un 'Behaviour' dans sa liste d'attente.

Les morceaux de code fournis à la Figure V.8 illustrent un exemple d'implémentation d'un 'CyclicBehaviour' dans notre système et qui concerne l'aspect optimisation distribuée que

nous avons adoptée via l'instauration d'échanges (ici à travers les *ACLMessage* entre *AO*). Cette illustration concerne en effet des bribes de code relatif à l'implémentation de la méthode *ODAVE* dans les différents *AO* de notre système. Dans l'exemple ici montré, figurent quelques instructions qui ont été extraites de *ODAVE* dans sa globalité et montrent l'exemple de test sur l'existence ou non d'un chemin solution partiel dont la recherche incombe à un *AO* appelé dans le cadre d'une collaboration avec un (ou plusieurs) autres *AO*. Ceux-ci forment entre eux ce que nous avons appelé *conversation*.

```

public class AttenteReponseAutreOptimisateurBehaviour extends CyclicBehaviour {
    MessageTemplate message_template_attendu = null;
    AgentOptimisateur localAgent = null;
    Troncon troncon = null;
    String conversationIdAvecParent = null;
    AID parentAgent = null;
    int pietonConcerne = -1;

    public AttenteReponseAutreOptimisateurBehaviour(AgentOptimisateur agent, Troncon troncon, String
    conversationIdAvecParent, AID parentAgent, String conversationIDAttendu, int pietonConcerne) {
        message_template_attendu = MessageTemplate.and(MessageTemplate.MatchPerformative(ACLMessage.INFORM),
        MessageTemplate.MatchConversationId(conversationIDAttendu));
        this.troncon = troncon; this.conversationIdAvecParent = conversationIdAvecParent;
        this.parentAgent = parentAgent; this.pietonConcerne = pietonConcerne;
    }
    int step = 0;
    public void action() {
        switch (step) {
            case 0:
                ACLMessage msg = receive(message_template_attendu);
                if (msg != null) {
                    chemin = (CheminGlobalePourUnPieton) msg.getContentObject();
                    if (chemin != null && chemin.isAgentMort()) {
                        refreshGui("Information received : " + msg.getSender().getLocalName() + " is dead");
                        //Finalisation du traitement et envoi de l'infotmation à l'agent Parent
                        step = finalisationTraitementEtInformerAgentParnt(pietonConcerne, localAgent, parentAgent,
                        conversationIdAvecParent, chemin, step);
                    } else if (chemin != null) {
                        //Une solution partielle existe, le chemin est reconstruit et l'AO initiateur en est informé
                        chemin.add(troncon);
                        solutions.put(pietonConcerne, chemin);
                        if (!localAgent.grandPa) {
                            refreshGui("chemin du pieton " + pietonConcerne + " reconstruit : \n" + chemin);
                            informerAgentParent(parentAgent, conversationIdAvecParent, chemin, "Envoi du chemin reconstruit à ");
                        }
                    }
                    ...
                }
            }
    }
}

```

Figure V.8. Exemple d'implémentation d'un 'Cyclic Behaviour'

La méthode *refreshGui* (*String s*) ici invoquée est une méthode qui a pour rôle de mettre à jour l'interface graphique. Elle a été définie et implémentée dans la classe 'GenAgent' pour 'Generic Agent' que nous avons énoncée précédemment comme classe mère dont hérite tous

les agents implémentés dans notre système. Par ailleurs, outre *refreshGui* (*String s*), '*GenAgent*' contient aussi d'autres méthodes.

Après l'implémentation et avant de procéder à l'illustration des différents jeux de tests réalisés sur notre approche et la plateforme mise en place, il convient de spécifier les détails de déploiement de notre application.

## V.4. Déploiement

Dans cette phase, il est nécessaire de spécifier comment seront déployés les agents intervenants dans le processus global relatif à notre approche. La répartition des agents étant d'un ordre de première importance puisqu'elle affecte directement le coût des communications, la monopolisation des ressources ou l'optimisation de la consommation en ressources... il s'agit donc de spécifier la bonne manière de procéder dans leur déploiement de manière à optimiser l'utilisation des ressources et gagner en temps CPU. L'impact direct sur les coûts en termes de communication provient du fait que la distribution des agents peut favoriser leur optimisation ou au contraire en dégrader qualité (c'est-à-dire les faire croître). Outre le fait que l'envoi et la réception d'un message dépendent en premier lieu de la typologie du réseau où sont déployés les agents, il est donc nécessaire d'optimiser par ailleurs les coûts (temps) de transmission des messages entre les agents eux-mêmes en trouvant la bonne répartition de ceux-ci.

Notre approche telle que nous l'avons définie hérite des caractéristiques des différents agents qui la composent. Dans les principaux algorithmes d'optimisation faisant le cœur des deux procédés suivant lesquels nous approchons la problématique du *CDO*, nous pouvons très bien remarquer qu'un agent n'est créé et n'intervient que si le besoin en est proclamé. A ce titre, nous pouvons citer notre stratégie à ne faire intervenir dans notre plateforme que les *AIU* relatifs à des usagers ayant émise des requêtes de covoiturage. Aussi, dans cette optique des choses, le principe de décomposition de la manière dont il a été approché, vient rejoindre cette idée d' '*économie d'agent*' en créant des *AO* sur les seules *ZP* abouties au tout début du processus. Une *ZI*, par contre, n'entrerait en jeu que si besoin est ; et un *AO* responsable de celle-ci est créé dans le seul cas où une décomposition d'une requête impose l'invocation de celui-ci (cas où le point intermédiaire joignant les deux requêtes partielles se trouve dans la *ZI* concernée). Par la suite, outre les avantages apportés par le concept agent (e.g. agent inactif en attente ne consomme pas ou consomme très peu de ressources CPU), la méthodologie de résolution adoptée que ce soit par rapport au principe de décomposition (c.f. *Chapitre III*) ou

par rapport aux méthodes d'optimisation distribuées (*c.f. Chapitre IV*) apporte un plus notable dans l'optimisation des coûts de communication.

Par ailleurs, indépendamment de la topologie du réseau ainsi que de sa nature, la capacité de communication dont sont dotés les agents dans notre système confère à notre système la possibilité d'adaptation à une grande diversité de configurations de déploiement possibles. Sont présentées dans ce qui suit deux architectures possibles dont nous choisirons une.

### V.4.1. Architecture Centralisée

Dans cette configuration, tous les agents se trouvent sur un même serveur central auquel toutes les requêtes et offres de covoiturage sont redirigées. Une telle configuration présente l'avantage de réduire considérablement les coûts subis par la communication entre les agents de notre système se trouvant sur un même lieu et évitant ainsi des échanges au gré du débit sur un réseau distant. En effet, les échanges réalisés dans le cadre d'une telle architecture se font au sein d'une même plateforme contenant tous les agents leur évitant ainsi des acheminements probablement très coûteux sur des réseaux 'non maîtrisables' de point de vue des débits naturellement exposés à des problèmes et perturbations (i.e. différents aléas du réseau). De plus, cette architecture apporte la sûreté requise dans les échanges en diminuant les risques de perte de messages. En contrepartie de ces différents avantages, l'architecture centralisée présente l'inconvénient de faire fonctionner les agents (leurs threads respectifs) de manière séquentielle, pourtant supposés travailler en parallèle pour gagner en temps d'exécution et donc en termes de performances. En effet, fonctionnant sur un serveur unique (i.e. une même machine dotée d'un seul processeur), les *threads* initiés à l'image de ces agents ne peuvent occuper en même temps ce processeur unique. Ceci a pour effet de limiter l'apport d'une telle architecture par rapport au gain réalisé de par l'optimisation des coûts de communication. La Figure V.9 montre un exemple de déploiement des agents implémentés dans notre approche dans une même plateforme sur la base d'une architecture centralisée sur un seul serveur.

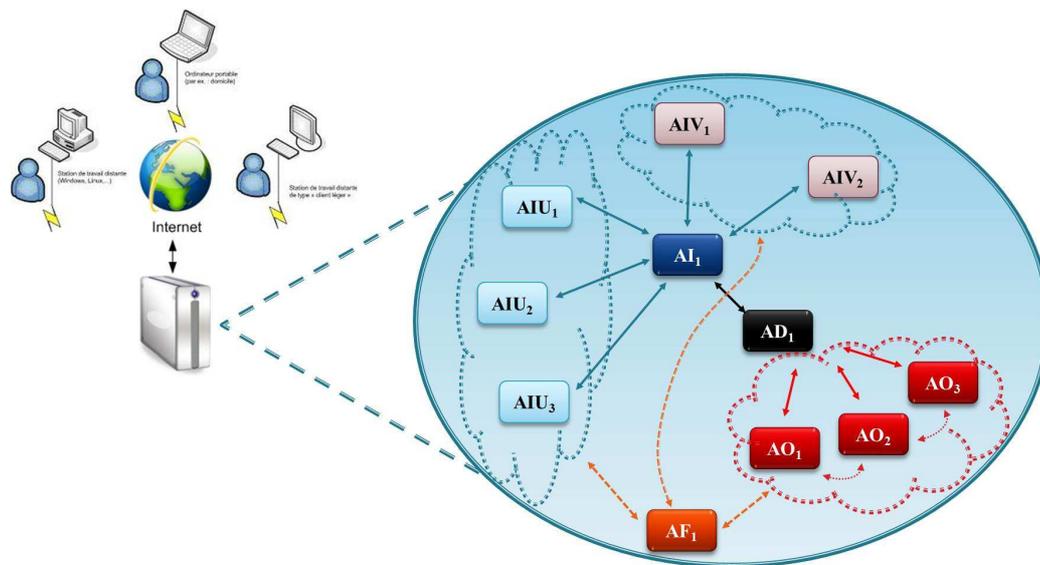


Figure V.9. CODAC déployé sur une architecture centralisée

Comme le montre cette figure, une machine unique sur laquelle s'exécutent plusieurs *threads* impose une file d'attente aux requêtes en provenance de plusieurs utilisateurs et énonce ainsi la nécessité d'adopter un algorithme de gestion approprié. Une telle solution n'est pas sans montrer de grands obstacles à l'élaboration d'une bonne gestion et pour l'atteinte des performances et efficacité escomptées puisqu'elle montre rapidement ses limites face à un grand nombre de requêtes. Ceci va à l'encontre de l'aspect temps réel des traitements dans la problématique adoptée. Chaque requête monopolisant totalement la plateforme a pour effet de dégrader encore plus les temps d'exécution retardant ainsi les délais de réponse. Cette répartition des agents reste par ailleurs d'un intérêt majeur par rapport aux coûts faibles de communication mais aussi elle ne nécessite pas la monopolisation de plusieurs machines pour l'exécution du système optimisant ainsi les ressources matérielles nécessaires. Ce qui n'est pas le cas pour les architectures de type distribuée.

#### V.4.2. Architecture Distribuée :

Afin de contribuer à l'amélioration des performances réalisées dans les architectures centralisées et combler le déficit que la seule existence d'une machine unique a fait émerger, sont nées les architectures distribuées. L'objectif principal de celles-ci s'inscrit donc dans l'optimisation du temps d'exécution de par le lancement de plusieurs processus parallèles pour la réalisation des différents traitements ; notamment l'affectation optimisée dynamique dans notre cas. Ces caractéristiques constituent un gain considérable par rapport au temps d'exécution de notre système nous permettant de rester dans le cadre de la dynamique de l'approche adoptée et de réaliser des performances à la hauteur des systèmes temps réel.

### V.4.3. Déploiement de CODAC : Choix d'une architecture

Compte tenu des différents avantages que présentent les architectures distribuées dans le cadre des systèmes dynamiques, nous avons alors opté pour une telle configuration. Ce aidant, nous avons par ailleurs tenu à préserver les avantages essentiels du paradigme agent en général et de notre approche en particulier. Ces avantages sont de calibre lourd dans le sens où la parallélisation des tâches ou plutôt des processus légers associés aux agents constitue un apport dont nous ne pouvons faire abstraction. Autrement, tout le procédé de par lequel nous approchons le problème du CDO et qui se résume en la mise en place d'une idée motrice qu'est la décentralisation du processus n'aurait plus aucun sens et nos contributions perdraient leur bien fondé.

Par ailleurs, nous ne voulons dénigrer l'aspect communication très présent dans notre approche, instaurant de multiples liens d'échanges ou de collaboration entre les différents agents de notre système. Pour ce faire, et ainsi profiter des avantages de l'une et l'autre des deux approches, nous considérant conjointement les deux configurations centralisée et distribuée en une seule '*architecture hybride*'. Comme il apparaît sur la Figure V.10, le déploiement de notre application mobilise plusieurs stations reliées entre elles à travers un réseau local, évitant ainsi de gaspiller le temps gagné de par l'instauration de processus parallèles en de longs échanges entre des agents dispatchés sur un réseau de plus grande taille. Les coûts de transferts des messages entre les différents agents impliqués sont effectivement nettement moins importants dans le cadre d'un réseau Intranet plutôt que Internet. Outre la parallélisation des traitements et les échanges en réseau local qui favorisent l'optimisation du temps d'exécution et ainsi augmentent les performances du système, nous limitons aussi les ressources matérielles utilisées. Un groupe formé de trois machines uniquement par exemple (Figure V.10) montre en effet beaucoup plus de performances que s'il était plus grand puisque celles-ci hébergent chacune plusieurs agents et les échanges se faisant ainsi en interne affectent positivement les performances de l'application ainsi déployée. En effet, l'avancement révolutionnaire réalisé dans les nouvelles *STIC* aidant, les équipements matériels (i.e. ordinateurs) sont aujourd'hui équipés de processeurs *multi-core* (e.g. *Dual-Core* pour les processeurs à double cœurs, *Quad-Core* : processeur à quatre cœurs) témoignant d'une énorme avancée sur le plan rapidité et efficacité des calculs effectués par un ordinateur répondant à ces caractéristiques. Le déploiement de plusieurs agents sur une même et unique machine est ainsi devenu possible tout en mettant à profit en plus la propriété essentielle d'exécution de leurs processus collectifs en parallèle. En effet, dotés de la capacité

de détecter l'existence de plusieurs processeurs (ou processeur multi-core) sur une machine et de faire leurs choix sur le processeur à utiliser, les agents d'un SMA font aussi preuve d'intelligence à niveau. Dès lors qu'une telle particularité est détectée sur la machine où ils sont logés, les agents s'organisent de manière à ce que chacun d'eux puisse exploiter un seul processeur. Localement à une machine, les échanges de message effectués dans ce cadre entre des agents pour lesquels il n'y a pas de risque de conflit (exploitant chacun un processeur) seront assurés rapidement et efficacement. Nous mettons à profit dans ce contexte les différents avantages énoncés par le descriptif d'une architecture centralisée. Conjointement à la répartition des groupes d'agents sur des machines distribuées sur un réseau local (architecture distribuée), les performances ainsi réalisées sont dignes des systèmes temps réel.

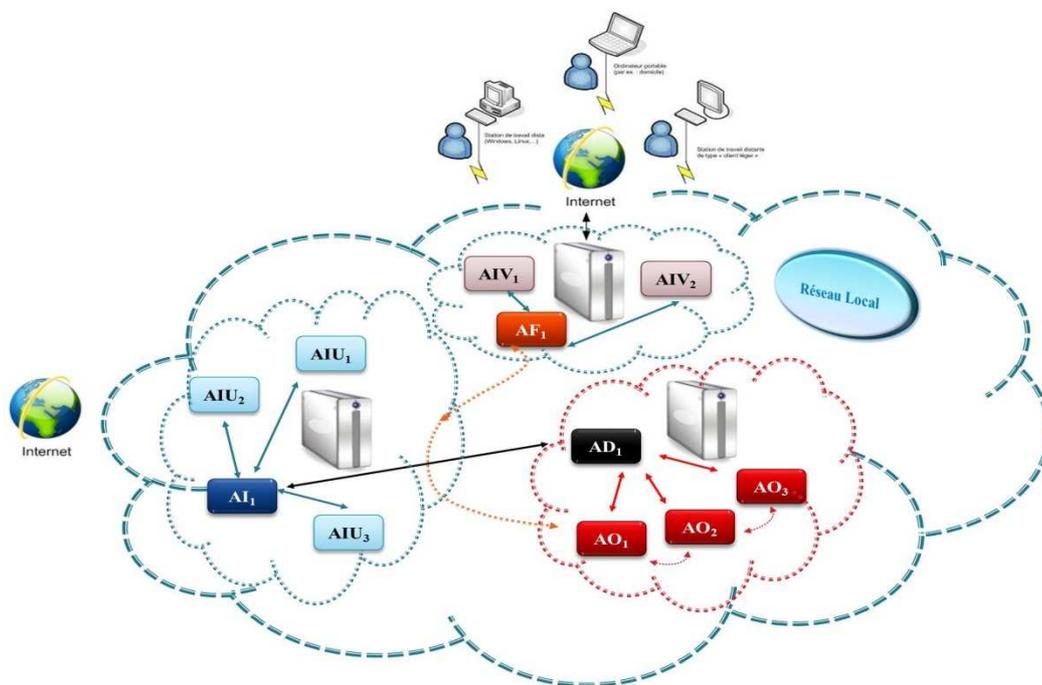


Figure V.10. Déploiement de notre système CODAC

Dans l'architecture ainsi établie, un exemple de déploiement d'agents instanciés dans le cadre de la réalisation de CODAC montre un dispatching de ceux-ci sur trois machines différentes reliées en réseau local et dotée chacune d'un processeur quad-core. Un principe et une technologie que nous mettons à profit pour optimiser les flux en en réduisant le coût.

Sur la base de cette architecture ainsi que de la plateforme de développement, l'architecture logicielle a été bien définie et répond d'une séparation entre les couches présentation, traitement et données tout en mettant en place différentes formes d'interaction entre celles-ci. Pour en démontrer la composition, nous présentons le modèle MVC auquel répond notre système.

## V.5. Architecture en couche

Dans cette section nous présentons l'architecture logicielle du système représentant sa composition de point de vue composantes ou entités intervenant dans son fonctionnement global. Nous proposons pour la mise en place de notre système d'adopter le modèle architectural *MVC* dont les particularités à la base de notre choix sont expliquées dans la section suivante.

### V.5.1. Modèle MVC

Notre système répond d'une modélisation sous forme d'une architecture en couche faisant intervenir trois différentes couches d'implémentation de notre système, à savoir une couche Modèle, une couche Vue et une couche Contrôleur. Plus généralement appelé *MVC* (*Modèle-Vue-Contrôleur*), il s'agit d'un patron de conception désigné plutôt comme patron architectural basé sur la séparation des données (le *modèle*), l'*Interface Homme-Machine* (la *vue* : *IHM*) et la logique de contrôle (le *contrôleur*). Les trois couches ainsi distinguées sont :

#### A. Le modèle

Spécifique aux données d'une application, il en représente la structure mais aussi définit les interactions avec la base de données ainsi que le traitement de celles-ci.

Dans notre application de manière spécifique, nous avons défini les particularités de chacun des agents intervenant dans la mise en œuvre du procédé d'affectation dynamique optimisé. Il est à noter que dans le contexte du *MVC*, nous pouvons remarquer que l'*Agent Information (AI)* de par ses fonctionnalités détaillées dans un précédent chapitre (*c.f. Chapitre IV*) se doit de récolter toutes les données actualisées nécessaires aux traitements devant être exécutés. Il assure ainsi dans son rôle de représenter toutes les données relatives aux requêtes et offres de covoiturage. Il convient à ce stade de noter que sur la base du comportement de l'*AI*, détaillé précédemment, et en concordance avec celui des *Agents Interface Véhicule (AIV)*, deux formes pour la collecte des données se présentent :

A travers une base de données où sont stockées toutes les informations sur les offres de véhicules ayant été reçues dans un processus préalable. Dans ce cas particulier, nous évoquons le fait qu'un conducteur ne prévient pas le système alors qu'il est déjà en cours de circulation ou même généralement n'attend pas le début de son voyage pour émettre une offre. Quel qu'en soit le contexte et le moment : voyage non encore entamé ou à son début ou encore voiture déjà en cours de circulation, l'information sur le trajet entrepris doit bien être

sauegardée si un *AI* n'existe pas dans le système pour la traiter. En effet, un *AI* n'est créé que lorsqu'il y a lieu de lancer le processus d'affectation des voitures aux piétons ; c'est-à-dire lorsqu'une demande de covoiturage a été reçue par le système. C'est à ce moment-là seulement qu'un *AI* est instancié et commence la collecte des données éventuellement à partir d'une base de données où sont préservées l'ensemble des offres ayant déjà été émises et réceptionnées par le système. Les différents traitements sur celles-ci, notamment une éventuelle mise à jour des paramètres et spécifications des déplacements offerts est ainsi envisageable si le voyage a déjà été entamé (e.g. mise à jour des origines des voyages qui deviennent les positions actuelles des voitures). Par ailleurs, ces informations même si elles ont subi préalablement un traitement quelconque aboutissant à des affectations ou non, elles restent tout de même nécessaires et sont potentiellement envisageables pour des traitements ultérieurs. Ceci est réalisable du moment où un voyage ne s'est toujours pas achevé, dans ce cas il interviendra au niveau des traitements des affectations prenant effet dans des processus prochains. Le voyage en question sera toujours pris en compte dans ces derniers tant qu'il demeure dans la mesure de la satisfiabilité requise par rapport aux contraintes de faisabilité (e.g. destination non encore atteinte, places disponibles...). Prenons le cas ici à titre d'exemple des trajets plus ou moins longs faisant intervenir le covoiturage dans sa forme occasionnelle ou événementielle pour des voyages entre des villes ou pays... plusieurs processus peuvent être lancés à la venue de requêtes de covoiturés à des moments différents alors que la voiture est toujours en cours de route. Comme il existe toujours une probabilité de concordance entre les chemins demandés et celui offert, l'information sur l'offre est préservée ainsi que les différents traitements qu'elle a subis (i.e. affectations de passagers). Elle peut donc être considérée dans la mesure du possible (i.e. disponibilité de places). De plus, nous faisons référence dans notre système à la nécessité de garder une trace sur les différents aboutissements des traitements effectués. Nous parlons notamment ici de conserver les combinaisons d'affectations réalisées pour pouvoir éventuellement générer des statistiques sur les fréquences des voyages, les usages par profils d'utilisateur, la nature des trajets demandés ou offerts, etc. Ceci nous permettra d'envisager dans un second temps de nos travaux d'introduire un aspect prédictif ou plutôt anticipatif des offres et demandes afin d'optimiser encore plus les traitements. Ces projets s'inscrivent dans une gestion améliorative préventive pouvant reposer entre autre sur différentes statistiques générées après avoir déployé le système à une échelle réelle et après qu'il ait fonctionné un temps suffisant. En effet, une durée minimale est requise pour faire ressortir les usages et les profils se basant sur une

utilisation réelle et un nombre important d'accès pour plus de crédibilité et de sûreté. Ces projets font naturellement partie des perspectives futures à nos travaux.

A travers les structures de requêtes et offres des usagers du service où les données sont maintenues préservées le temps que durent les traitements. En effet, étant placés dans le contexte du temps réel, les requêtes de covoiturage parvenant instantanément concernent dans la plupart des cas des voyages immédiats. L'AI fonctionne dans ce cas sur des données existantes dans des structures spécifiques contenant les paramètres des demandes. Ces structures sont préservées juste le temps que prendra le système pour en chercher les différentes affectations possibles et notamment celles optimisées.

### ***B. La vue***

Elle concerne tout ce qui touche à l'interfaçage avec l'utilisateur (*IHM : Interface Homme Machine*), c'est-à-dire ce avec quoi il interagit. Aucun traitement n'est effectué à ce niveau sauf l'affichage simplement des données qui lui parviennent de par la couche modèle. Plusieurs vues peuvent être initiées à fin de représenter les données d'un même modèle. Dans notre cas par exemple, il s'agit essentiellement des vues relatives aux interfaces spécifiques aux covoiturés, c'est l'AIU qui intervient à ce moment, et celles relatives aux covoitureurs et c'est l'AIV qui est concerné dans ce cas. Par ailleurs, d'autres vues d'interfaces pour l'inscription des usagers, piétons et conducteurs sont discernées de celles relatives à la spécification des besoins en matière de déplacement (i.e. requêtes et offres). A chaque fois une nouvelle interface est appelée et est différente selon le besoin de l'utilisateur (inscription ou réservation) et sa nature (conducteur ou piéton).

La Figure V.11 montre l'interface relative à l'insertion des différents paramètres pour l'émission d'une nouvelle requête. Dans l'exemple montré, les champs sont remplis par défaut par des données générées aléatoirement spécifiant des informations sur le nombre de passagers, l'heure d'arrivée au plus tôt et l'heure d'arrivée au plus tard, etc. dont le choix est laissé au gré du système dans un premier temps. Cela nous a aidés à générer rapidement des paramètres et données pour pouvoir tester le système faute de données réelles et aboutir à des résultats. Ceci n'empêche pas le fait que les données générées peuvent être assimilées à des données réelles, puisqu'étant conformes aux formats de dates et contraintes sur les nombres de places maximal autorisé. Aussi, les positions géographiques spécifiées sont relatives à des adresses existantes sur une cartographie réelle. Par ailleurs dans un contexte et positionnement à l'échelle réelle, cette interface qui est appelée lorsque l'utilisateur concerné en émet le

souhait sera notamment réinitialisée lui donnant la main pour insérer les paramètres qu'il souhaite.

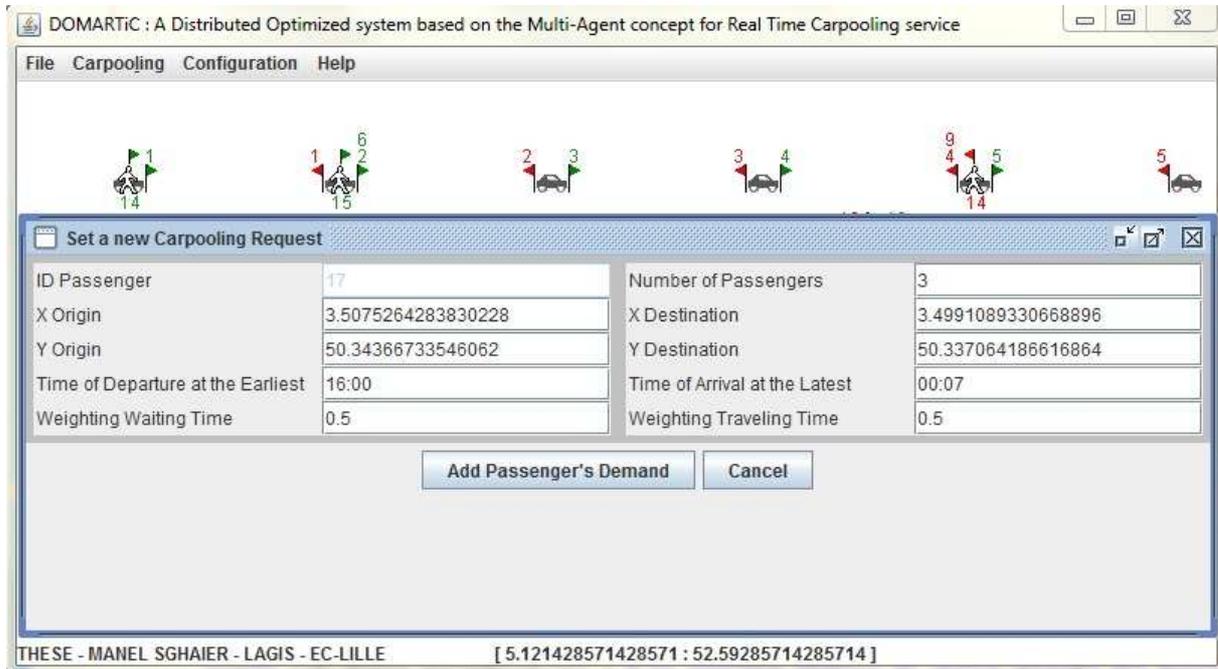


Figure V.11. Insertion d'une requête de covoiturage

La Figure V.12 montre un exemple d'insertion d'offre. Pareillement à l'exemple précédent pour l'insertion d'une requête, la spécification d'offres concerne ici une génération aléatoire pour des besoins de tests. Par ailleurs, nous pouvons noter ici l'option de mise à jour du temps d'arrivée au plus tard qui se manifeste de par l'existence d'un bouton nommé 'Update the arrival time' sur la capture d'écran ci-dessous montrée. En effet, l'utilisateur peut spécifier ce paramètre par lui-même comme il peut générer un calcul automatique de celui-ci en fonction des règles et formules spécifiées dans un chapitre précédent (*c.f. Chapitre III*, règle de calcul du temps d'arrivée au plus tard :  $a$ ). Celles-ci dépendent entre autre de la vitesse moyenne de parcours ( $VMP$ ) calculée selon différents facteurs (e.g. temps, période horaire durant laquelle sera effectué le voyage, vitesse du véhicule selon le type de la route). Le temps d'arrivée au plus tard ainsi automatiquement calculé sera fonction de la  $VMP$  et de la distance séparant l'origine de la destination spécifiées par l'utilisateur lui-même et en incluant bien évidemment la marge de retard tolérable.

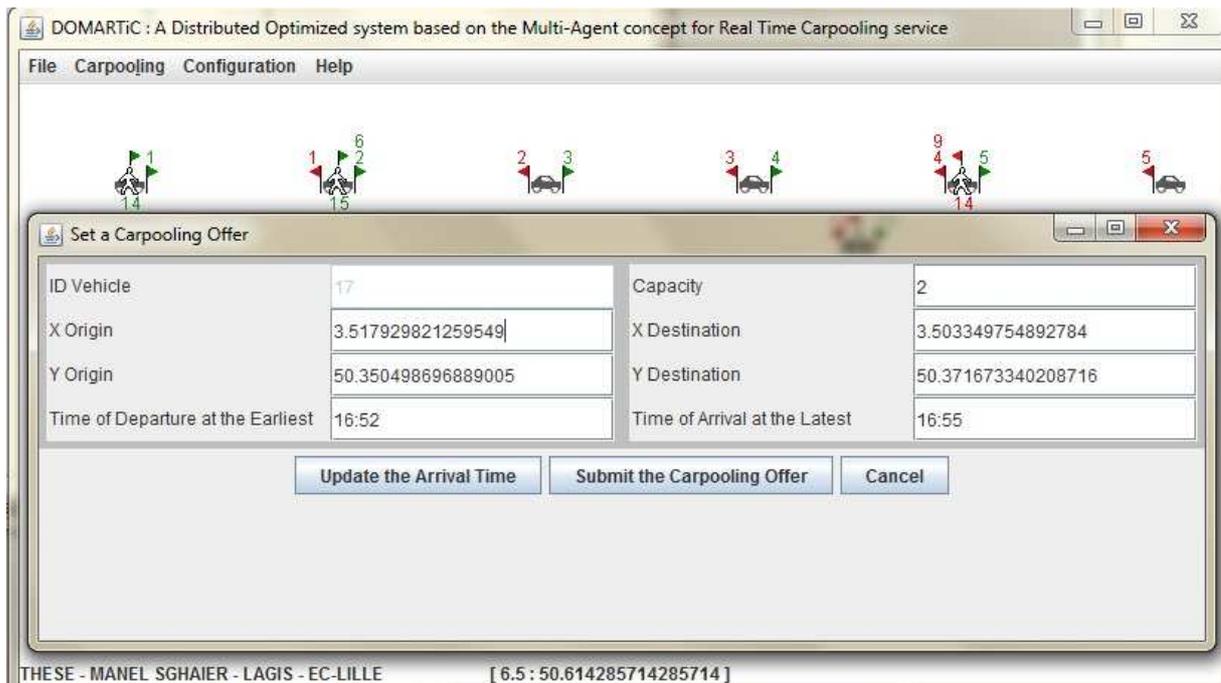


Figure V.12. Insertion d'une offre de covoiturage

Bien d'autres interfaces intégrant la couche *Vue* de notre application existent, notamment celles affichant les résultats d'exécution des algorithmes que nous verrons plus tard lors de la description des différents tests de ceux-ci. D'autres interfaces sont aussi à noter telles que celle montrant le positionnement des voitures et usagers (leurs positions actuelles ainsi que leurs destinations) sur une même interface comme il apparaît sur celles données en background des Figures V.11 et V.12.

### C. Le contrôleur

Cette couche spéciale est responsable de gérer l'interfaçage et les échanges entre le *Modèle* et le client. Elle est ainsi responsable d'interpréter la requête de ce dernier afin de lui envoyer la vue qui y correspond. Le contrôleur effectue ainsi la synchronisation entre les couches modèle et vue et permet donc de générer des évènements lors d'une modification du modèle et de lancer une mise à jour de la vue déjà en cours d'affichage. C'est le cas par exemple de la méthode *'refreshGui(String s)'* de l'agent *'GenAgent'* de notre application et qui a été précédemment évoquée. C'est la pattern Observer qui est responsable de cette synchronisation. Dans ce contexte, nous pouvons signaler que c'est l'*Agent Fusion (AF)* qui dispose de l'information sur toutes les modifications générées sur les données de la couche modèle (gérée par l'*AI*). Ces modifications, opérées par les *AD* et *AO* seront par la suite dispatchées aux différentes instanciations des *AIU* et *AIV* avec une identification nécessaire des supports utilisateurs correspondants et afin d'ordonner l'interfaçage adéquat. Par ailleurs,

les *AD* et *AO* jouent aussi un rôle de contrôle sur les vues dont nous ne pouvons faire abstraction. Ils sont effectivement responsables de la décomposition du terrain pour le cas du premier et de la génération d'itinéraires solutions optimisés pour le cas du deuxième. Différentes représentations des modifications générées sur le graphe représentant les données du réseau sont ainsi émergées à la suite de la réalisation de ces différentes étapes de décomposition et de recherche d'affectations. Une vue différente du réseau peut ainsi être visualisée sur une interface distincte.

Une architecture à trois couches : Interface (*Modèle*), Application (*Contrôleur*) et Données (*Modèle*) appliquée à notre modélisation multi-agent est visible sur la Figure V.13. Cette dernière montre effectivement une représentation du modèle à couches selon la modélisation Agent proposée dans *CODAC*. Cette représentation concorde avec les spécifications de notre problème, la modélisation choisie et des principes et traitements énoncés et réalisés. En effet, conformément aux explications données précédemment, chacun des agents fait partie d'une couche unique suivant le fonctionnement qui lui incombe.

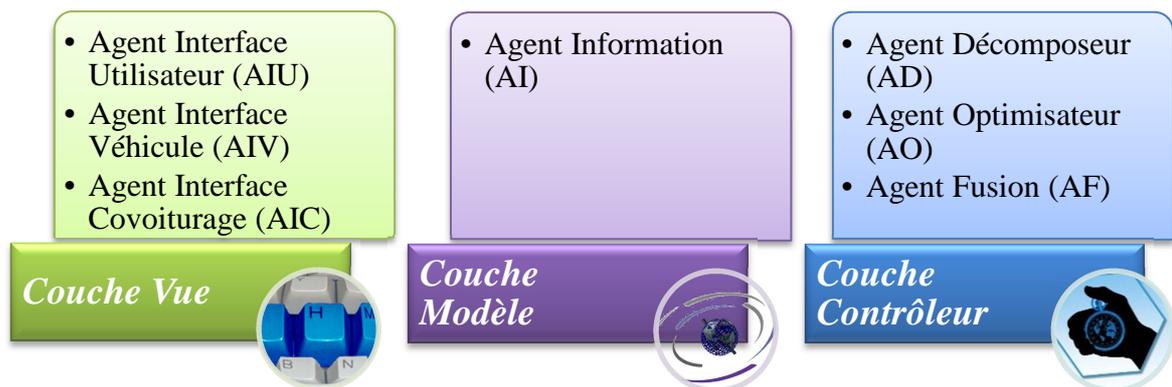


Figure V.13. Architecture du système et implication des agents dans chacune des couches

Dans la Figure V.14 est illustré le modèle de représentation des flux montrant les différentes possibilités d'échange entre les différentes couches<sup>42</sup>.



Figure V.14. Interactions entre les couches du modèle MVC

<sup>42</sup> <http://ootips.org/mvc-pattern.html>

En implémentant notre application suivant ce type architectural, nous bénéficions de ses principaux avantages qui concernent essentiellement :

- La possibilité de changer une couche sans pour autant altérer les autres puisque celles-ci sont clairement séparées et peuvent être modifiées sans porter atteinte au reste des couches (e.g. nous pouvons changer Swing par SWT dans la couche Vue sans toucher au modèle ni au contrôleur).
- La synchronisation des vues.

### V.5.2. L'API JDBC

Pour l'accès aux données du système nous avons fait usage de l'API (*Application Programming Interface*) *JDBC* qui permet à notre application d'accéder à une base de données où toutes les informations requises pour le traitement courant sont stockées.

La technologie *JDBC* (*Java DataBase Connectivity*) est utilisée en vue de pouvoir connecter des applications informatiques spécifiques à un serveur de données dans le cadre de manipulation de données existantes sur celui-ci. Elle est constituée d'un ensemble de classes permettant le développement de ces types d'applications les dotant de l'aptitude à se connecter à des Serveurs de Gestion de Bases de Données (*SGBD*). L'accès effectué grâce à *JDBC* peut se faire en adoptant soit un *modèle à deux couches* soit un *modèle à trois couches*. ce dernier fait intervenir un *serveur d'applications* qui vient s'intercaler entre l'application Java et la base de données pour faire transiter les requêtes *SQL* et les résultats retournés. Outre l'avantage de présenter un contrôle d'accès possible, ce modèle engendre des coûts inutiles pour faire transiter les données et requêtes entre les trois couches. En contrepartie, dans la deuxième modélisation qui est celle que nous adoptons, l'application Java est intimement liée à la base de données instaurant des rapports directs pour l'interrogation de la base aussi bien que pour la réception des résultats générés par celle-ci. Dans ce cas, la base de données peut par ailleurs être exécutée sur la même machine où est hébergée l'application Java (i.e. machine locale) ou bien sur un ordinateur relié en Intranet ou Internet.

Cependant, pour instaurer des liens directs, il faut impérativement disposer du pilote *JDBC* adéquat pour pouvoir manipuler la base de données en question. Nous utilisons dans ce contexte le pilote *ODBC* (*Open DataBase Connectivity*) qui convertit les appels de données Java en appels *ODBC* valides pour les exécuter par la suite grâce au pilote. Malgré la diversité des *SGBD*, les types de *BD* et leurs façons de manipuler les données et traiter les requêtes *SQL*, *ODBC* exhibe l'avantage de fournir des résultats identiques peu importe le type

de la *BD* et sans pour autant modifier le programme transmettant la requête. De plus, à l'heure actuelle, pratiquement toutes les bases de données disposent d'un pilote *ODBC*, la plupart d'entre elles ayant été développées à partir d'interfaces Microsoft.

Après avoir introduit toutes ces notions et concepts architecturaux pour la mise en place de notre système, nous procédons dans les deux dernières parties à en montrer les fonctionnalités qui ont été théoriquement détaillées au préalable à travers les précédents chapitres. Cependant, avant de montrer les exemples de tests de ces fonctionnalités générés sur le système ainsi élaboré, nous présentons dans la section suivante la partie *GIS (Geographical Information System)* : module essentiel dans notre système intégré de par l'interfaçage avec un logiciel spécifique nommé Cartocom.

### V.6. Cartocom

Le covoiturage dynamique optimisé vu de notre œil est une approche opérée en temps réel pour la recherche des affectations possibles des voitures offrant des trajets en covoiturage aux utilisateurs piétons en quête de trajets spécifiques dans ces voitures (si correspondance est).

Le contexte temps réel dans cette vision des choses s'avère très contraignant non seulement de point de vue optimalité des traitements mais aussi par rapport à la qualité des réponses fournies qui risquent fort de ne plus être fiables et réalisables lors de leur dispatching aux usagers. En effet, l'aspect dynamique fort imposant dans le contexte d'une mobilité totale et continue relativement au domaine du transport est très propice au piège des conflits des affectations qui ne concorderaient plus lors de leur émission aux covoiturés et covoitureurs concernés. Traiter des données aussi évolutives et inconstantes d'un instant à l'autre est un choix que nous assumons très bien dans notre système. Le risque de non-conformité est ainsi géré d'une part de par l'intégration de marges de retards tolérables dans le calcul des durées de parcours de distances bien déterminées. Les formules utilisées à cet effet ayant été détaillées dans le *Chapitre III*, sont particulièrement utiles dans ce contexte pour déterminer par exemple les temps d'arrivée de véhicules à un point bien défini (par exemple l'origine du déplacement d'un utilisateur où il se trouve à l'heure actuelle où le processus est en train de s'exécuter). Cette durée concerne le temps d'attente du passager et repose sur une forme prévisionnelle impliquant une première valeur qui indique la date d'arrivée au plus tôt, qui est en fait la date de départ au plus tôt du véhicule concerné à partir de ce point. Une deuxième valeur concernerait la date d'arrivée au plus tard incluant la marge de retard possible du véhicule et servant à faire patienter le piéton. Le véhicule en question ne pourra en aucun cas

arriver avant la date au plus tôt ainsi transmise. Celle-ci est notamment supérieure à l'heure système correspondant à l'émission de la réponse naturellement faite et validée après avoir prévenu le conducteur qu'il y aura prise de passagers à l'endroit indiqué. De ce fait, le véhicule concerné s'arrêtera automatiquement à cet endroit pour pouvoir récupérer le piéton. Tous ces détails de réception, discussion et validation des réponses générées par le système sont visibles de par les comportements des Agent Interface Vehicule (AIV), Agent Interface Utilisateur (AIU) et Agent Fusion (AF) décrits précédemment (*c.f. Chapitre IV*).

D'autre part ce même risque de dissolution est aboli de par la manière dont a été structuré, implémenté et déployé notre système dont les performances sont à la hauteur du contexte temps réel dont il s'imprime. **CODAC** (ou encore **DOMARTiC**) dispose d'une plateforme logicielle qui s'exécute en des temps très négligeables dans une optimalité de son processus présentant ainsi les performances nécessaires dans le cadre d'un système dynamique. Les architectures multi-agent logicielles et physiques choisies sont on ne peut plus favorables à ce contexte réduisant considérablement les coûts en termes de temps d'exécution essentiellement. Nous faisons particulièrement allusion ici aux coûts engendrés par les communications et échanges entre les différents agents de notre système. Celui-ci est en effet un véritable moteur de travail collaboratif faisant intervenir des coalitions d'agents dans les deux approches proposées. Les architectures adoptées ayant réussi à inhiber le coût d'échanges probables pas très rétrogradant par ailleurs vu la parallélisation des tâches ; reste le problème d'acquisition des données.

En effet, l'intégrité des données manipulées en temps réel entre en grande partie en considération dans la fiabilité des calculs effectués et dans leur efficacité et convenance lorsque les réponses fournies seront dispatchées aux usagers. Le challenge énoncé ici est de pouvoir manipuler des données réelles actualisées et '*vraies*'. C'est-à-dire les données courantes des usagers, qu'il s'agisse de piétons ou de voitures même si elles sont déjà en cours de circulation. Ceci concerne de manière spécifique les clients mobiles ; nous entendons par là les usagers en train de bouger et qui selon leurs mouvements changent de positions d'un instant à l'autre. Plus spécifique encore est le problème des voitures ayant déjà entamé leurs trajets, leur mobilité est bien plus grande que les piétons puisque se déplaçant à des vitesses nettement supérieures. Le besoin imminent de pouvoir disposer de données actualisées en continu n'en est que plus grand. C'est dans ce cadre que nous avons fait appel aux services particuliers d'un progiciel spécifique nommé **Cartocom** développé par la société *Bayo*<sup>43</sup>

---

<sup>43</sup> <http://www.bayo.com/>

spécialement pour subvenir aux besoins de notre équipe en matière d'information géographique.

Etant munie d'outils GPS, cette plateforme acquise par l'équipe, facilite en effet l'acquisition des données géographiques et assure la localisation continue des usagers connectés au système. Cartocom a ainsi été ancré dans notre plateforme comme un module *GIS* responsable de récupérer les données transit des usagers de leurs origines jusqu'à leurs destinations en offrant en parallèle de multiples services dont nous avons fait profit pour la réalisation de nos objectifs. Ceux-ci tournant principalement autour de la volonté de situer notre système à une échelle réelle, la cartographie mise à disposition de par le logiciel *Cartocom* nous a permis d'atteindre les objectifs escomptés de ce point de vue dans un premier temps et d'intégrité et d'efficacité dans un second temps. En effet, les services offerts par ce logiciel sont forts propices à une manipulation 'juste' à l'abri des conflits, nous aidant ainsi à relever le défi du temps réel. Parmi ces services, notons principalement :

- La géolocalisation de manière continue des usagers de notre système, qu'ils soient piétons ou conducteurs. Ce service a pour effet de mettre à la disposition de notre système l'information nécessaire et continuellement mise à jour sur les coordonnées géographiques en cours des utilisateurs connectés. En effet, lorsqu'il aura été enclenché, *CODAC* devra pouvoir accéder directement et instantanément à l'information sur les covoitureurs et covoiturés connectés à cet instant au système pour connaître leurs positions réelles. Pour ce faire, leurs coordonnées géographiques courantes sont actualisées en continu grâce à l'outil *GPS* dont est menu le logiciel *Cartocom*. Après l'extraction des données sur les positions actuelles, les données qui y sont relatives sont stockées dans une base de données gérée par *Cartocom* et à laquelle *CODAC* pourra accéder directement et interroger les données qui s'y trouvent. Ces actions entreprises par le logiciel auquel nous faisons référence ici (*Cartocom*) sont réalisées bien évidemment aussi fréquemment que possible. A priori, une capture des données sur les positions géographiques est faite toutes les 30 secondes ; une donnée qui reste par ailleurs paramétrable selon les besoins de traitement et au gré de l'administrateur du système. Une fréquence importante est d'autant plus nécessaire que la nature de notre problème l'impose. En effet, dans des manipulations réelles, l'on énonce la problématique d'accès aléatoire dépendant uniquement des besoins des utilisateurs et des moments des émissions des requêtes desquels nous ne répondons pas. Notre plus grand objectif étant par ailleurs une mise à l'échelle rapide dans un sens

large incluant la compétitivité, contribue de surcroît au besoin de fréquence de grand calibre. Dans le traitement des requêtes, ces coordonnées sont utilisées comme les origines des déplacements dans deux cas : si l'utilisateur est en cours de déplacement et que le temps de départ spécifié par ses soins est déjà passé ; sinon dans le cas contraire, cette géolocalisation servira à définir l'origine de son mouvement s'il a omis de la spécifier.

- La géolocalisation dans la fréquence de son exécution permet par ailleurs de disposer d'une information complète sur les déplacements des usagers en faveur de la mise en place d'une traçabilité totale des différents trajets entrepris en covoiturage ou non. Ceci pour préparer le terrain à la réalisation des projets futurs dans un second temps; notamment à relever des statistiques sur les routes les plus fréquentées, les déplacements les plus fréquents, les destinations les plus souhaitées par période, par profil utilisateur ou autre, etc.
- Pour rester dans ce même cadre, il est à noter que nous faisons le bénéfice d'apporter aux usagers l'assurance requise en matière de sûreté et surtout de sécurité. En ces termes, notre système s'avère favorable à la mise en place d'un dialogue continu avec les usagers mettant à profit non seulement le rôle des agents dans ce sens mais aussi le fait de disposer continuellement des informations essentielles sur leurs déplacements. Cette spécificité qui a longtemps fait défaut aux systèmes et projets de covoiturage existants est remarquablement appréciée par les individus qui sont intuitivement en quête de sûreté. La peur de l'inconnu motivant toujours leurs préférences de déplacement entre autre, ils refoulent les systèmes qui n'en portent pas la garantie, même si c'est au dépend du bénéfice apporté par des avantages qu'ils offrent et au détriment du succès de ceux-ci. En effet, ces penchants appuyés pour la sécurité ont souvent porté préjudice à plusieurs des systèmes existants conduisant souvent à leur échec. C'est pour cela que nous nous focalisons particulièrement sur l'intégration de cet aspect largement démontré dans nos services y apportons l'aspect confiance qui servira à rassurer les usagers. En effet, étant connecté en continu avec les utilisateurs via des canaux de communication établi de par les agents en plus de disposer de leurs coordonnées actualisées, le système est ainsi impliqué dans des aspects de garantie de la sécurité de ceux-ci. Parallèlement, cet aspect des choses peut aussi être mis à profit pour récupérer par exemple des informations sur le trafic ou autre... au service de l'intégration de nouveaux services comme la gestion des risques et perturbations

probablement dans des projets futurs. La localisation instantanée et continue joue donc un rôle pivot dans notre système : entre récupération et actualisation des données, et une traçabilité jointe à la sécurité des usagers, cette fonctionnalité assure la mise en œuvre des modules complémentaires abordés dans le premier chapitre de ce mémoire. Par ailleurs, en référence à la CNIL<sup>44</sup> évoquant les droits de l'homme et nous soumettant au devoir de respecter les droits des usagers de notre service à une vie privée ; une charte spécifique a été conçue à cet effet. Tout utilisateur du service ayant souscrit à notre système et subissant la fonctionnalité de géolocalisation doit avoir signé au préalable (lors de son inscription) cette charte où figure une clause particulière faisant référence à ce service et ce afin d'instaurer un environnement de confiance où règne une transparence totale avec l'utilisateur. La signature de tout abonné fait office d'accord donné au système *CODAC* pour pouvoir récupérer ses coordonnées du moment où il s'y connecte.

- Un autre service fourni par *Cartocom* est celui qui concerne la mise à la disposition du système *CODAC* d'une cartographie réelle sur laquelle peuvent être visualisés les usagers, leurs origines ainsi que leurs destinations. Plusieurs cartes du monde sont ainsi intégrées dans ce logiciel et dont nous profitons pour offrir une vue réelle aux usagers de notre système, à l'instar de Google Maps ou Mappy.
- Sur cette cartographie peuvent aussi être retracés des itinéraires des individus demandant ou offrant des trajets de covoiturage, en fonction uniquement de leurs origines et destinations souhaitées aussi bien qu'en intégrant des points repères spécifiés par les usagers eux-mêmes. Nous faisons référence ici aux points de passage (points repères ou destinations intermédiaires) faisant partie intégrante des spécifications définissant le paramétrage des offres des véhicules (*c.f. Chapitre IV*). Ces points peuvent aussi bien être associés à des points de prise et dépose spécifiés par le système lui-même dans ce cas et sont tirés des affectations réalisées lors de processus préalables. Ces spécifications et particularités dont nous faisons usage dans notre système ont été initiées depuis le départ afin de respecter les préférences des conducteurs dans leurs déplacements et ne pas leur imposer de trajets peut être méconnus d'eux. Sinon, si aucune spécification n'a lieu, l'itinéraire généré est dans ce cas le même que celui construit de par les outils *GPS* et qui est notamment relatif au plus court chemin existant.

---

<sup>44</sup> <http://www.cnil.fr/>

- Cette fonctionnalité de traçage d'itinéraire fournit encore deux grands avantages à l'élaboration de notre système et dont nous avons tiré profit. Il s'agit dans le premier de disposer de l'information totale sur ces itinéraires dans des scripts manipulables à travers notre système. L'information en question se trouvant dans ces scripts est structurée sous forme de points ou nœuds géographiques se succédant dans une chaîne représentant l'itinéraire et dont la longueur (nombre de nœuds spécifiés) est laissée au gré de l'administrateur du système. Cette disposition est particulièrement utile pour déterminer les nœuds de croisement des itinéraires relatifs aux voitures dans le cadre de l'élaboration des algorithmes de Dijkstra adoptés dans notre première approche (*A3D*). Ces structures s'avèrent en plus très utiles dans la deuxième méthode de recherche d'affectations optimisées dynamiques (*ODAVE*) où il faut chercher les intersections des demandes et offres d'itinéraires afin d'extraire les nœuds les plus proches desservis. Le deuxième avantage concerne le fait de partir dans le sens inverse, c'est-à-dire disposant des nœuds constituant les itinéraires, probablement les itinéraires solutions, nous pouvons les retracer sur la carte réelle représentant le terrain (réseau de covoiturage) via l'outil *Cartocom*.

D'autres services très intéressants et utiles sont aussi fournis au niveau de cet outil, concernant notamment la mise à disposition de feuilles de routes, d'interfaces graphiques affichant les différents usagers (piétons ou voitures) appelés mobiles, avec leurs statuts de connexion (e.g. connecté, déconnecté...), etc. Nous n'avons cité parmi cette large panoplie de services que ceux qui sont directement liés à l'exploitation des outils cartographiques et qui sont en relation direct avec notre système et essentiellement pour mener à bien les différentes tâches et traitements qui lui incombent. Des exemples de manipulation des fonctionnalités et services offerts au niveau du progiciel *Cartocom* seront entre autre fournis dans la suite, lors de l'élaboration des jeux de tests réalisés sur notre plateforme.

### **V.7. Tests et Scénarii d'exécution**

Dans cette section nous procédons à la démonstration de l'usage pouvant être fait de notre système élaboré dans les conditions préalablement spécifiées. Nous mettons ainsi en exergue son fonctionnement de par des scénarios d'exécution que nous appuyons avec des détails de calculs que nous avons pris soin de réaliser pour montrer la validité des résultats générés à travers l'appel aux services fournis par la plateforme *DOMARTiC*.

Nous donnons dans un premier temps des résultats générés et visualisés directement sur la plateforme *Cartocom* et nous en montrons nous-mêmes les étapes du processus avec les calculs nécessaires effectués par nos soins. Par la suite nous générerons quelques résultats d'exécution directement de par la plateforme *DOMARTiC*. Celle-ci étant en effet équipée de classes interfaces affichant la vue adéquate à l'utilisateur pour une manipulation directe par le biais des interactions possibles à travers l'interface en question.

Nous présentons alors en premier lieu quelques données relatives à un scénario d'exécution spécifique faisant intervenir un ensemble d'utilisateurs et de voitures avec les paramètres de leurs demandes et offres. Ces données seront prises en compte lors de la réalisation des calculs nécessaires pour générer les résultats. Quelques exemples de ces calculs seront fournis à la section suivante pour enfin montrer dans celle qui suivra quelques captures d'écran montrant l'affichage des résultats directement sur *Cartocom*.

### V.7.1. Données sur les usagers

Dans cette section nous présentons les différentes données intervenant dans le test de la plateforme ici effectué. Dans le *Tableau V.1* figure l'ensemble des utilisateurs 'piétons' non encore affectés en référence à un ensemble de requêtes pseudo-simultanées reçues à l'instant  $t=9h10$ . C'est l'AI qui en a assuré l'acquisition comme il a été spécifié dans la description de son fonctionnement (*c.f. Chapitre IV*). Après avoir été créé à l'instant même où la première requête usager a été reçue, l'AI en attente durant  $\Delta\epsilon = 3 \text{ secondes}$  récolte les informations sur les requêtes susceptibles d'arriver durant ce temps. Nous supposons qu'elles sont au nombre de 6 et nous en présentons les spécifications touchant essentiellement aux nombres de passagers, temps de départ au plus tôt et temps d'arrivée au plus tard souhaités. Par ailleurs, nous faisons abstraction dans le *Tableau V.1* des spécifications des données sur les origines et destinations des voyageurs par manque d'espace. En effet, les  $U_1^+$  et  $U_1^-$  font référence à des adresses réelles non spécifiées ici vu leurs longueurs et à fin d'éviter le débordement dans ce tableau.

*Tableau V.1. Requêtes d'utilisateurs à un instant  $t=9h10$*

<i>Identifiant Usager</i>	<i>Origine Requête</i>	<i>Destination Requête</i>	<i>Nombre de Passagers</i>	<i>d</i>	<i>a</i>
$U_1$	$U_1^+$	$U_1^-$	1	9h10	10h40

$U_2$	$U_2^+$	$U_2^-$	<b>2</b>	<b>9h10</b>	<b>10h</b>
$U_3$	$U_3^+$	$U_3^-$	<b>1</b>	<b>9h10</b>	<b>10h15</b>
$U_4$	$U_4^+$	$U_4^-$	<b>2</b>	<b>9h10</b>	<b>10h10</b>
$U_5$	$U_5^+$	$U_5^-$	<b>2</b>	<b>9h13</b>	<b>10h00</b>
$U_6$	$U_6^+$	$U_6^-$	<b>4</b>	<b>9h10</b>	<b>10h30</b>

En contrepartie, le Tableau V.2 montre les différents paramètres relatifs aux véhicules émettant des offres de covoiturage. Les spécificités des déplacements affichées dans ce tableau sont actualisées et concernent les données courantes sur ces offres à l'instant  $t$  considéré. Ceci notamment pour éviter les conflits provenant de données corrompues ne prenant pas en compte les affectations préalablement associées à ces véhicules ou une information non mise à jour mettant en doute la position réelle de la voiture.

**Tableau V.2. Offres de covoiturage à un instant  $t=9h10$**

<i>Identifiant Voiture</i>	<i>Origine Offre</i>	<i>Destination Offre</i>	<i>Nombre de Places</i>	<i>Destinations Intermédiaires</i>	<i>d</i>	<i>a</i>
$C_1$	$C_1^+$	$C_1^-$	<b>1</b>	$(C_5^+, C_8^-, U_3^-, C_7^-, C_5^-)$	-	<b>10h15</b>
$C_2$	$C_2^+$	$C_2^-$	<b>3</b>	$(U_4^+, ID_1, U_6^+, U_4^-)$	-	<b>13h</b>
$C_3$	$C_3^+$	$C_3^-$	<b>4</b>	$(U_1^+, U_2^+, ID_1)$	-	<b>11h</b>
$C_4$	$C_4^+$	$C_4^-$	<b>3</b>	$(U_6^+, ID_1)$	-	<b>12h15</b>
$C_5$	$C_5^+$	$C_5^-$	<b>7</b>	$(C_8^-, U_3^-, C_7^-)$	-	<b>11h15</b>
$C_6$	$C_6^+$	$C_6^-$	<b>8</b>	$(U_5^+, U_6^+, U_5^-)$	-	<b>12h10</b>
$C_7$	$C_7^+$	$C_7^-$	<b>2</b>	$(C_4^-, ID_1, U_1^-, U_2^-)$	-	<b>12h</b>
$C_8$	$C_8^+$	$C_8^-$	<b>1</b>	$(U_1^+, U_3^+, C_5^+)$	-	<b>9h45</b>
$C_9$	$C_9^+$	$C_9^-$	<b>3</b>	$\left( \begin{array}{l} C_4^-, ID_1, U_1^-, U_2^-, \\ C_7^-, U_6^+, U_5^- \end{array} \right)$	-	<b>13h15</b>
$C_{10}$	$C_{10}^+$	$C_{10}^-$	<b>2</b>	-	-	<b>13h</b>

La Figure V.15 montre la disposition des nœuds sur une carte réelle telle que générée via l'outil *Cartocom*.

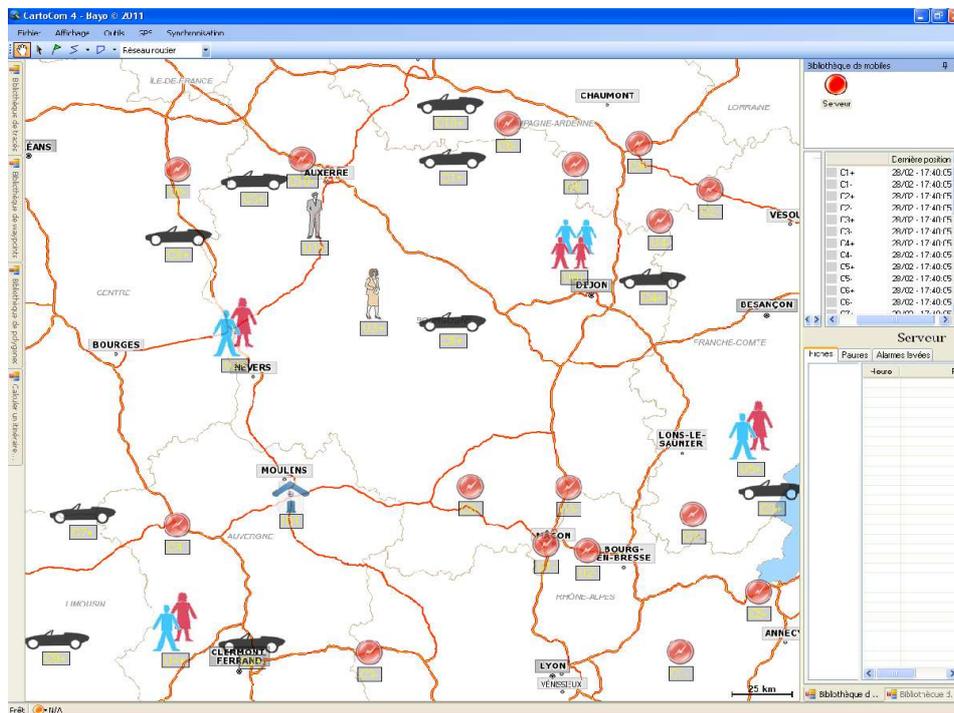


Figure V.15. Positions des origines et destinations des piétons et véhicules sur une cartographie réelle

Les conducteurs n'ayant pas spécifié ou presque pas de nœuds adresses de passage pour leurs itinéraires, nous avons générés les champs qui leur sont relatifs (*Destinations Intermédiaires*) de par des comparaisons faites sur les itinéraires construits à l'aide de l'outil *Cartocom* pour ces voitures avec l'ensemble des nœuds dont nous disposons. Il s'agit des nœuds origines et destinations de l'ensemble des covoitureurs et covoiturés en plus d'un nœud de passage (adresse intermédiaire) ( $ID_1$ ) qui a par ailleurs été spécifié par le conducteur de la voiture  $C_7$  comme faisant partie de son chemin. La construction de l'itinéraire associé à cette voiture a ainsi pris en compte cette adresse comme point repère pour le traçage de son chemin convenablement à son choix.

La capture d'écran montrée à la Figure V.16 illustre un exemple de traçage d'itinéraire dans *Cartocom* en spécifiant les origines, destinations et adresses intermédiaires respectivement appelés *départ*, *arrivée* et *étapes* sur cette figure et dont les spécifications insérées sont relatives à la voiture  $C_7$ . Ces champs spécifiques à l'interface *Cartocom* sont notamment spécifiés via un interfaçage direct entre ce logiciel et notre plateforme. L'itinéraire colorié en vert sur cette figure est celui de la voiture  $C_7$ .

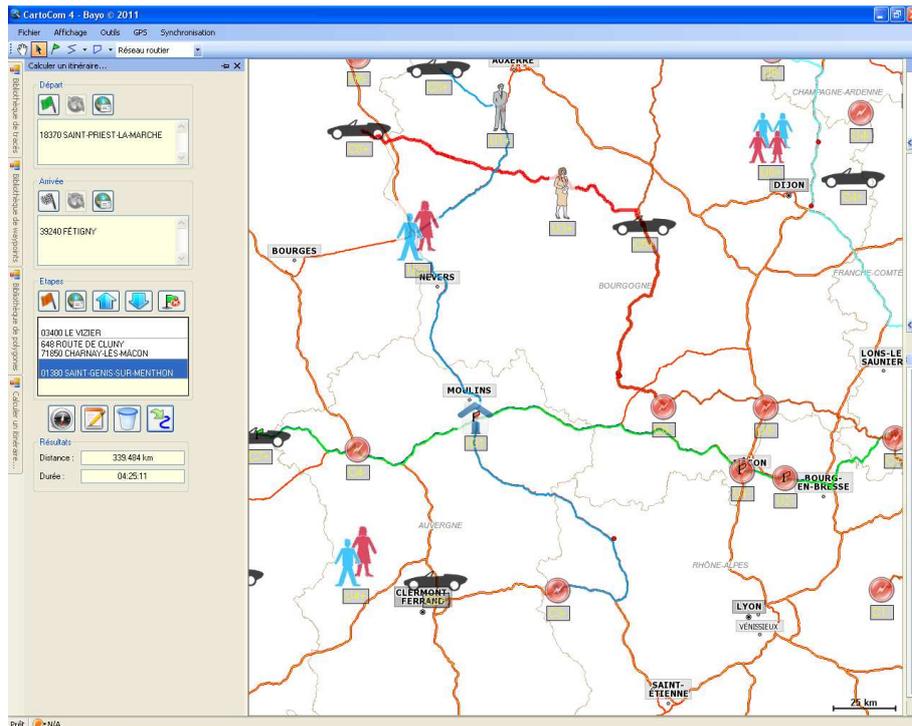


Figure V.16. Traçage de l'itinéraire de la voiture C<sub>7</sub>

La Figure V.17 montre l'exemple de traçage de l'itinéraire relatif à la voiture C<sub>3</sub>.

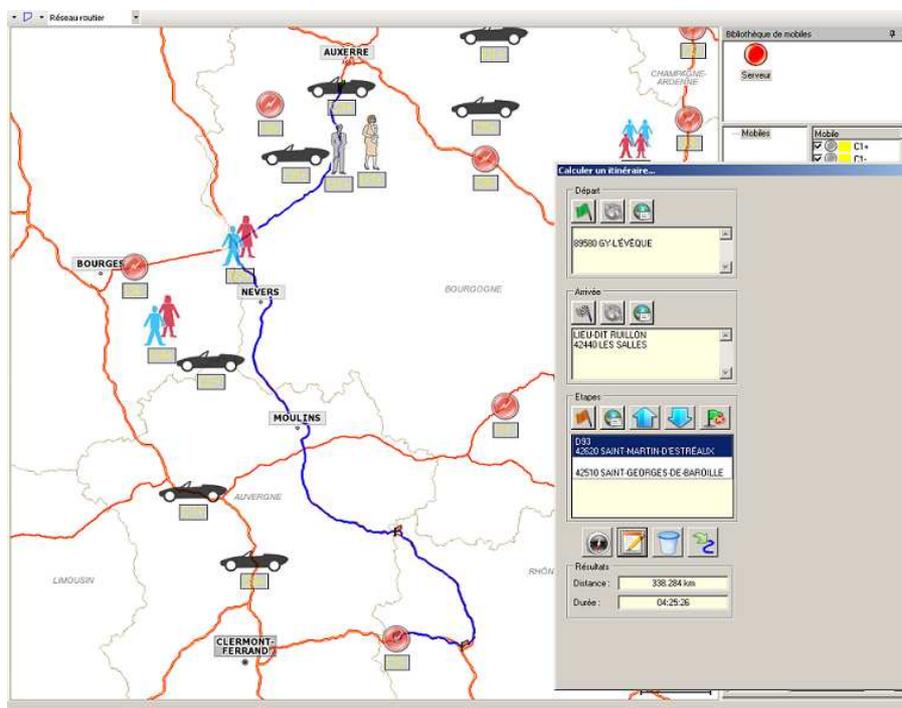


Figure V.17. Traçage de l'itinéraire de la voiture C<sub>3</sub>

Tous les itinéraires ayant été tracés sur cette base, avec ou sans étapes (*Destinations Intermédiaires*), nous disposons alors des données nécessaires pour effectuer les traitements internes à notre système.

## V.7.2. Traitements et affectations

Dans cette section nous présentons les principaux traitements réalisés sur les données récoltées et structurées par l'AI. Ces traitements commencent réellement à partir du moment où les différentes données sont envoyées par l'AI à l'AD pour subir en premier lieu une classification mettant en exergue les zones de concentration des données et où il convient de lancer le processus d'affectation les favorisant à d'autres beaucoup moins denses et ne contenant pas d'origines de requêtes. En effet, relativement à cette stratégie de résolution 'économique' en termes de processus, le procédé de décomposition détaillé et formulé dans le *Chapitre III* converge vers la concentration sur la donnée unique relative aux requêtes dans sa première phase.

### V.7.2.1. Requêtes et Zones Primaires

N'est considérée dans la première étape du principe de décomposition du réseau géographique que l'information sur les piétons. Pour la réalisation de celle-ci et conformément à la spécification formelle relative à la mise en place d'un GDD pour la représentation du terrain, un premier graphe est construit. Ce dernier contient toutes les données sur les requêtes des piétons pouvant être schématisées. Le graphe illustré dans la Figure V.18 est relatif à une modélisation des informations sur les piétons (données dans le Tableau V.1). Dans ce graphe les itinéraires reliant les origines aux destinations souhaitées ont été construits via l'outil *Cartocom* donnant lieu aux plus courts chemins dans ce cas puisqu'aucune étape n'est spécifiée.

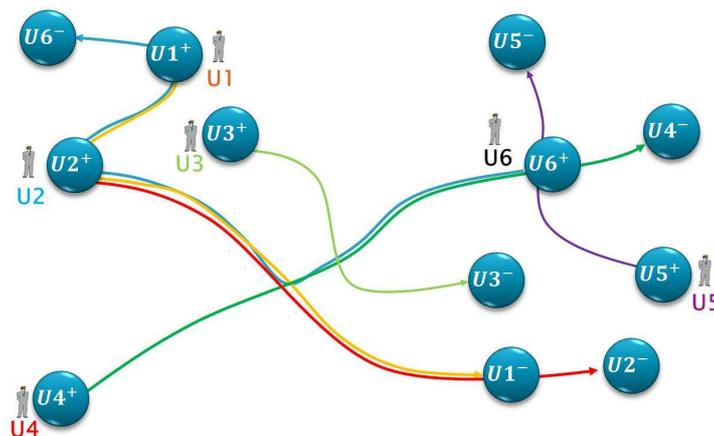


Figure V.18. Représentation des usagers (piétons) et leurs itinéraires

Partant de cette illustration, nous définissons toujours graphiquement, à travers la Figure V.19 les Zones Primaires (ZP) établies de par l'exécution des algorithmes implémentés à cet

effet. Ces algorithmes nommés *CZP* (*Création des Zones Primaires*) ont été spécifiés dans le *Chapitre III*. Compte tenu des données géographiques relevées sur les adresses fournies par les utilisateurs indiquant leurs lieux de départ et d'arrivée, les distances les séparant ont été calculées via la formule précédemment spécifiée (*c.f. Chapitre III. Equation III.7*) :

$$Distance(O, D) = \text{acos} \left( \begin{matrix} \sin(\text{latitude}_O) * \sin(\text{latitude}_D) + \cos(\text{latitude}_O) * \\ \cos(\text{latitude}_D) * \cos(\text{longitude}_D - \text{longitude}_O) \end{matrix} \right) * R$$

Les distances ainsi calculées sont utilisées pour déterminer l'appartenance d'un ensemble de nœuds ou pas à une même zone en comparaison du diamètre fixé ainsi que pour le calcul de la position des centres des zones.

La Figure V.19 montre une disposition des *ZP* mettant en exergue les ensembles de voisins dans l'espace. Notons que *CZP* implique un choix aléatoire du nœud à traiter dans chacune de ses itérations engendrant peut être des décompositions (et donc des *ZP*) différentes à chaque fois. Cependant, si les nœuds (ou plutôt ensemble de nœuds voisins entre eux) sont assez éloignés les uns des autres ( $Distance > Diamètre$ ), cette disposition reste alors la même quel que soit le tirage.

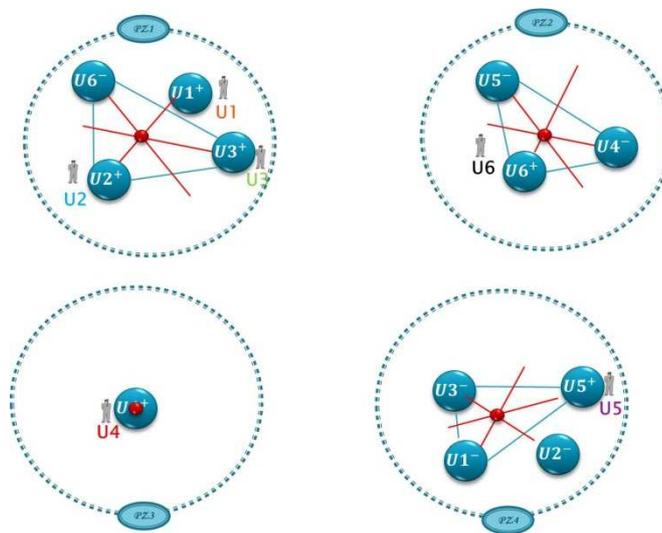


Figure V.19. Zones Primaires

Il convient de remarquer à cette étape que la classification des données et donc la disposition des zones dépendent entièrement du diamètre (ou rayon) dont la spécification est du ressort unique de l'administrateur du système *DOMARTiC* et peut être modifiée selon le besoin. Une fenêtre spécifique dans notre plateforme le permet, elle est montrée à la Figure V.20.

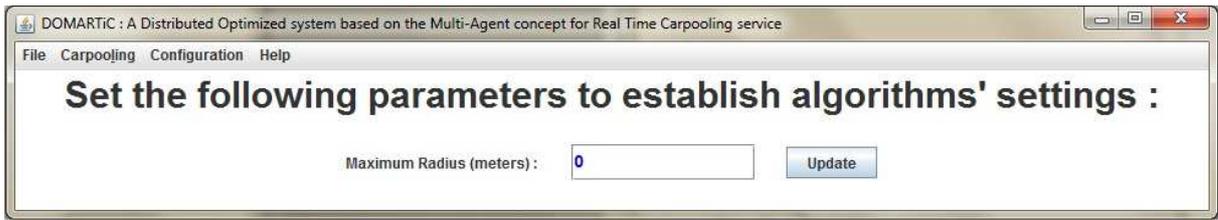


Figure V.20. Spécification d'un rayon de zones

Il est aussi judicieux de remarquer que le rayon (ou diamètre) ainsi spécifié ne doit pas être très grand afin de ne pas porter préjudice à l'aspect parallélisation : apport principal de notre approche et contribuant fortement à la réalisation de nos objectifs, relativement aux performances et efficacité de notre système. En effet, des processus lancés en même temps pour le traitement distribué et parallèle des requêtes de covoiturage y jouent un grand rôle.

### V.7.2.2. Voitures et Zones Intermédiaires

Cette partie est consacrée à montrer la phase de prise en compte des voitures et leurs données de déplacements incluant notamment les paramètres de leurs offres. Comme il a été réalisé pour les requêtes de covoiturage, le même procédé ou presque est suivi dans cette seconde étape de décomposition. Les nœuds et arcs sont ainsi définis de la même manière que pour les piétons prenant en considération les coordonnées géographiques des origines, destinations et adresses de passage pour l'identification des nœuds et les itinéraires extraits par le biais de *Cartocom* comme référence aux arcs (Figure V.21).

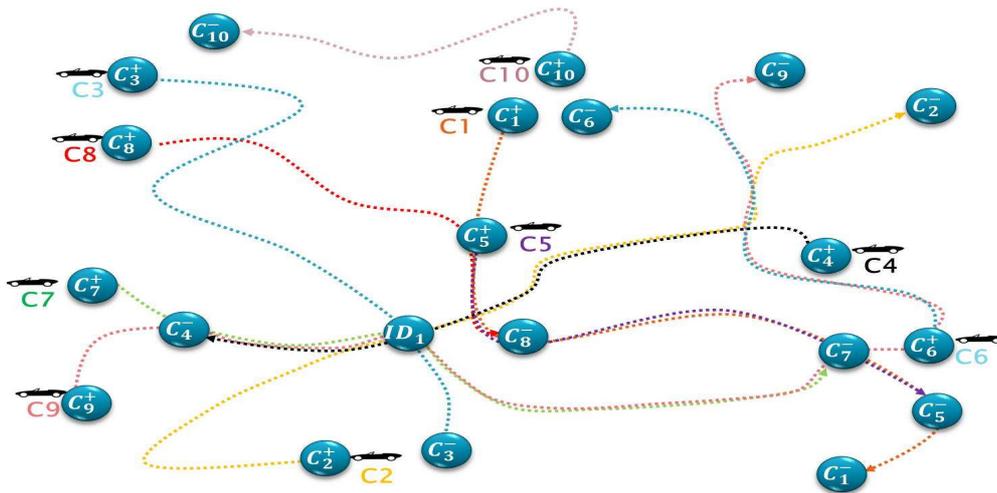


Figure V.21. Graphe pour les origines et destinations des voitures

Par la suite un graphe superposition des deux plans (plan piéton et plan voiture) de l'espace est réalisé marquant une fusion des deux graphes ainsi construits pour en faire ressortir les similitudes et notamment la correspondance entre les trajets.

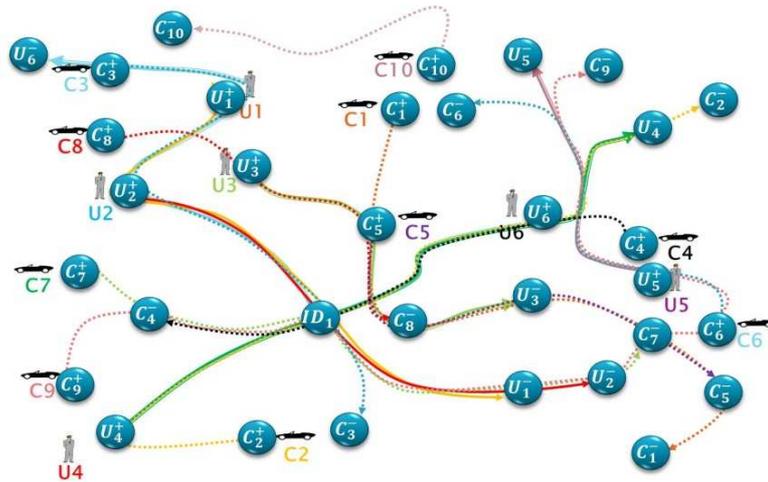


Figure V.22. Superposition Covoitureurs & Covoiturés

Il s'agit ici d'une visualisation graphique illustrée dans la Figure V.22 (et la Figure V.23 via *Cartocom*) démontrant la validité et fiabilité des résultats fournis dans la section suivante.

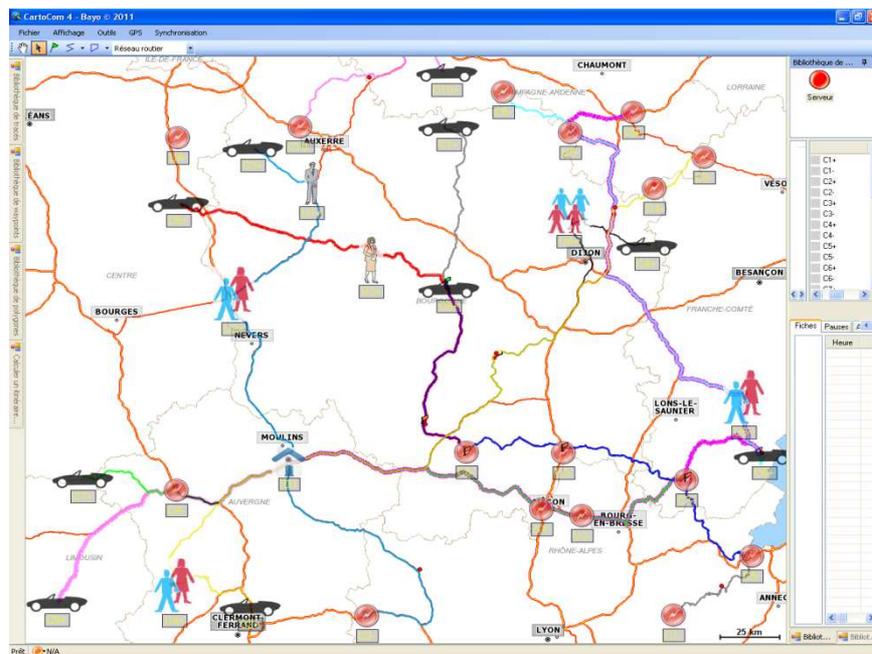


Figure V.23. Superposition des plans voitures et piétons sur cartocom

Sur la base de cette modélisation et des zones primaires créées de par la première phase de décomposition du terrain en fonction des demandes des usagers uniquement, les zones intermédiaires *ZI* seront développées au fur et à mesure selon les algorithmes précédemment établis et développés *CZI* (*Création de Zones Intermédiaires*). Ces derniers cherchent en premier lieu de possibles affectations des nœuds nouvellement construits aux *ZP* déjà élaborées et créent de nouvelles zones pour ceux non affectés sur le même principe de voisinage inspiré des méthodes de classification des données (Figure V.24).

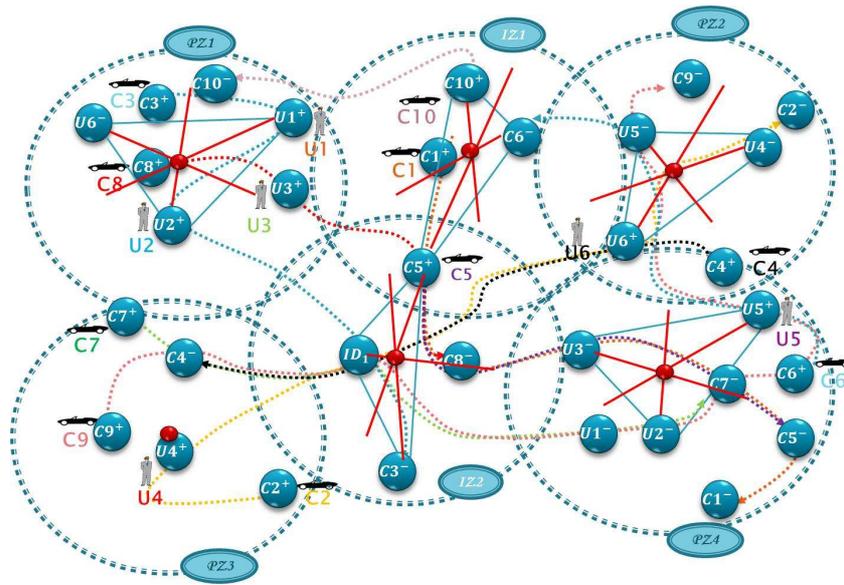


Figure V.24. Zones Intermédiaires et couverture du réseau

### V.7.3. Quelques calculs

Dans cette section nous montrons quelques exemples de calculs générés sur des agents ayant intervenu dans le traitement des affectations optimisées dynamiques des véhicules aux piétons sur la base des spécifications de l’algorithme *ODAVE*. Ces traitements se basent sur la donnée essentielle sur les vitesses et temps moyen de parcours figurant dans le *Tableau V.3*.

Tableau V.3. *TPM & VMP*

<i>Période de voyage (/Plage Horaire)</i>	<i>Temps</i>	<i>Type de la Route</i>	<i>Temps de Parcours Moyen (TPM per 100 km)</i>
<i>Période Normale</i>	<i>Pluie</i>	<i>Autoroute</i>	<i>VMP = 100km/h TPM=75 min</i>
<i>Période Normale</i>	<i>Pluie</i>	<i>Route Secondaire</i>	<i>VMP = 70km/h TPM=120 min</i>
<i>Heure de Pointe</i>	<i>Pluie</i>	<i>Autoroute</i>	<i>TPM=100 min</i>
<i>Heure de Pointe</i>	<i>Pluie</i>	<i>Route Secondaire</i>	<i>TPM=150 min</i>
<i>Période Normale</i>	<i>Normal</i>	<i>Autoroute</i>	<i>TPM=67 min</i>
<i>Période Normale</i>	<i>Normal</i>	<i>Route Secondaire</i>	<i>TPM=86 min</i>
...	...	...	...

Tel que définit auparavant le procédé de traitement des affectations débute à partir d’Agents Optimisateurs (AO) lancés sur les ZP contenant des origines de requêtes de covoiturage. Les quatre ZP établies dans le cas de l’exemple ici détaillé sont concernées et



Les exemples dans les deux Figures 25 et 26 sont assez simples, le premier disposant d'un choix unique et qui par coïncidence vérifie les contraintes de correspondance. Le deuxième dispose par contre de deux choix dont un ne répond pas à la faisabilité de l'offre.

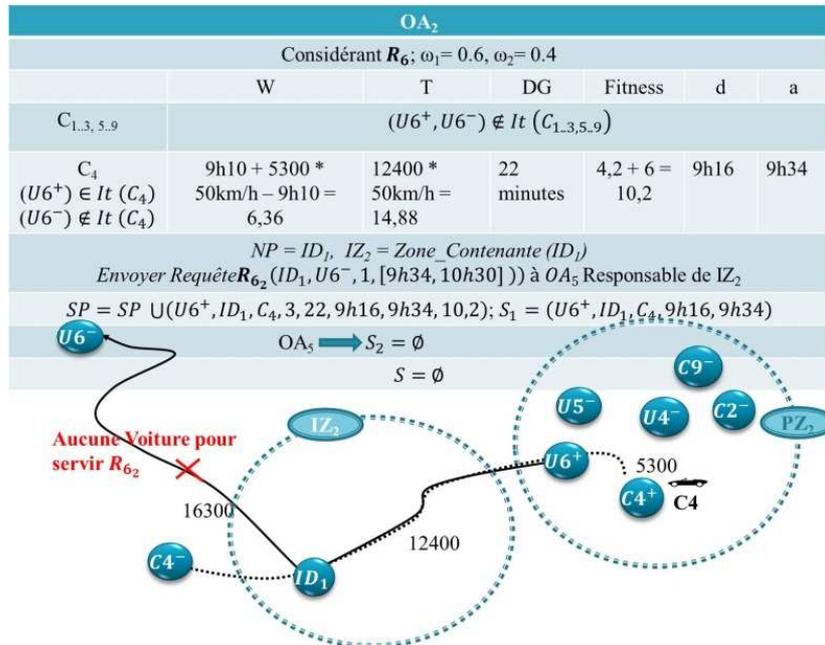


Figure V.27. Traitement de R<sub>6</sub>

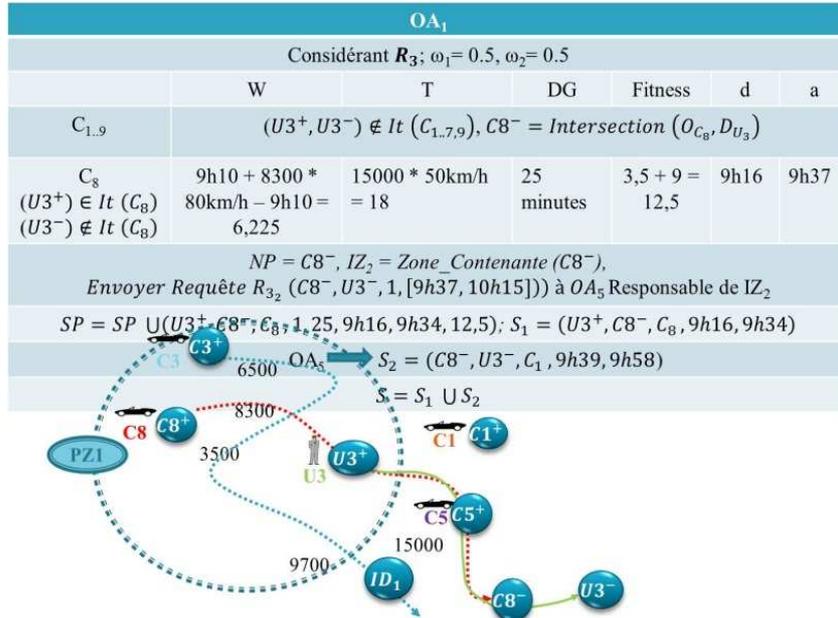


Figure V.28. Traitement de R<sub>3</sub>

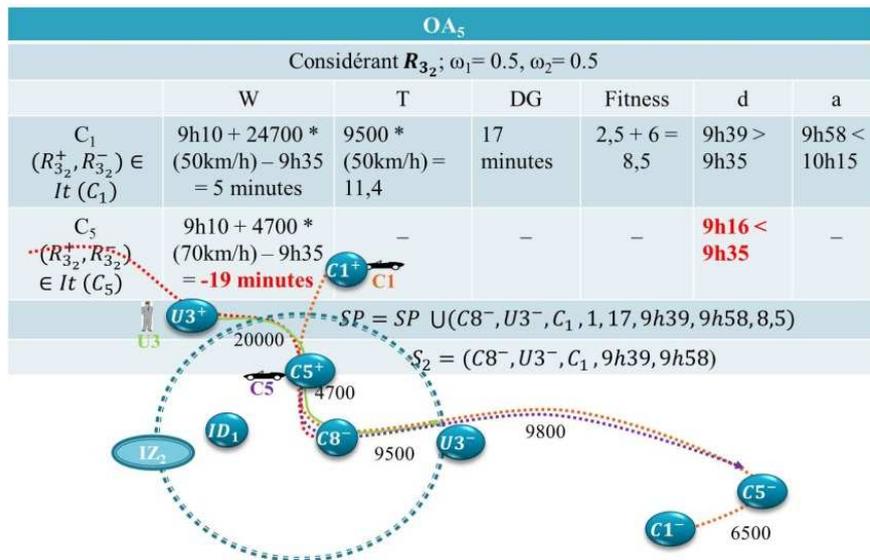


Figure V.29. Traitement de la deuxième partie de R<sub>3</sub>

Les trois dernières figures mettent en relief le fonctionnement collaboratif des agents pour le traitement de R<sub>6</sub> et R<sub>3</sub> faisant intervenir en coalition les agents OA<sub>2</sub> et OA<sub>5</sub> pour la première et OA<sub>1</sub> et OA<sub>5</sub> pour la deuxième.

### V.7.4. Affichage des résultats

La Figure V.30 montre l’affichage des solutions générées représentant les itinéraires correspondants via Cartocom.

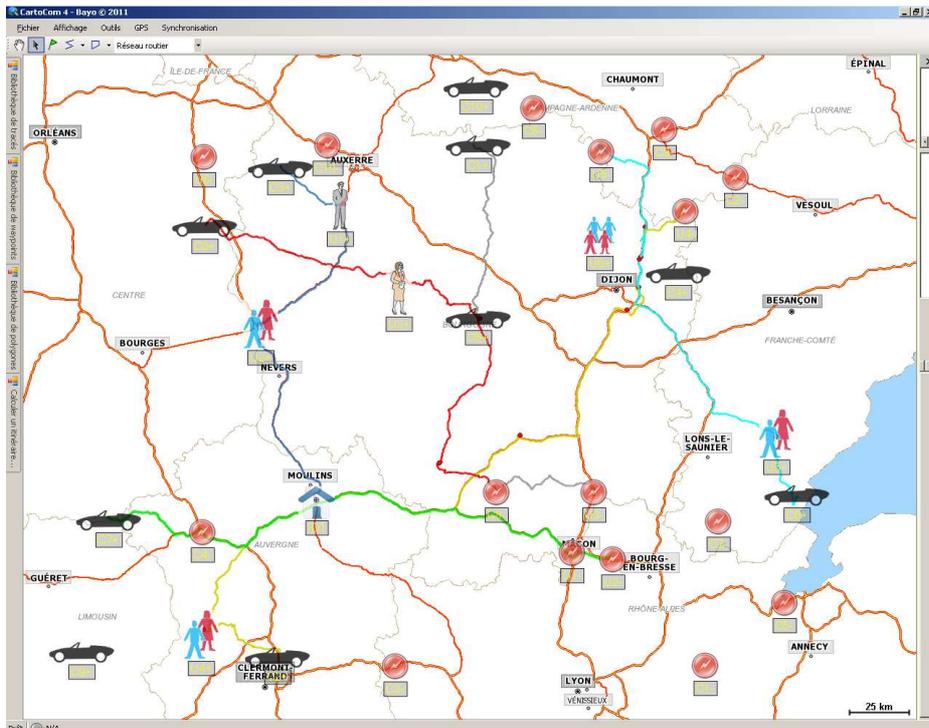


Figure V.30. Affichage des itinéraires solutions

### V.7.5. DOMARTiC : Une couche Vue

Les différentes captures d'écran réalisées sur notre système et visibles de par les Figures V.31-33 sont relatives aux interfaces générées directement à travers de multiples tests réalisés sur la plateforme de l'application *DOMARTiC*. Nous avons choisi de mettre en place, entre autre, un module de génération aléatoire de *données synthétiques* (piétons et voitures). Ceci permettant une faculté au niveau des générations des données permet de tester rapidement et efficacement le système *CODAC*. En effet, les données manipulées dans des systèmes tels que ceux des transports en général et du covoiturage en particulier dépendent entièrement des besoins et préférences des usagers en matière de déplacement. D'où le choix de laisser au gré du hasard la génération des données pour effectuer des tests de manière efficace et en plus assez rapide. Ceci n'empêche pas le fait que les données manipulées sont conformes à celles traitées dans des cas réels, à savoir des coordonnées géographiques réelles pour les nœuds et spécificités de déplacement, aussi bien que pour le restant des paramètres (i.e. nombre de places, date et heure, etc.).

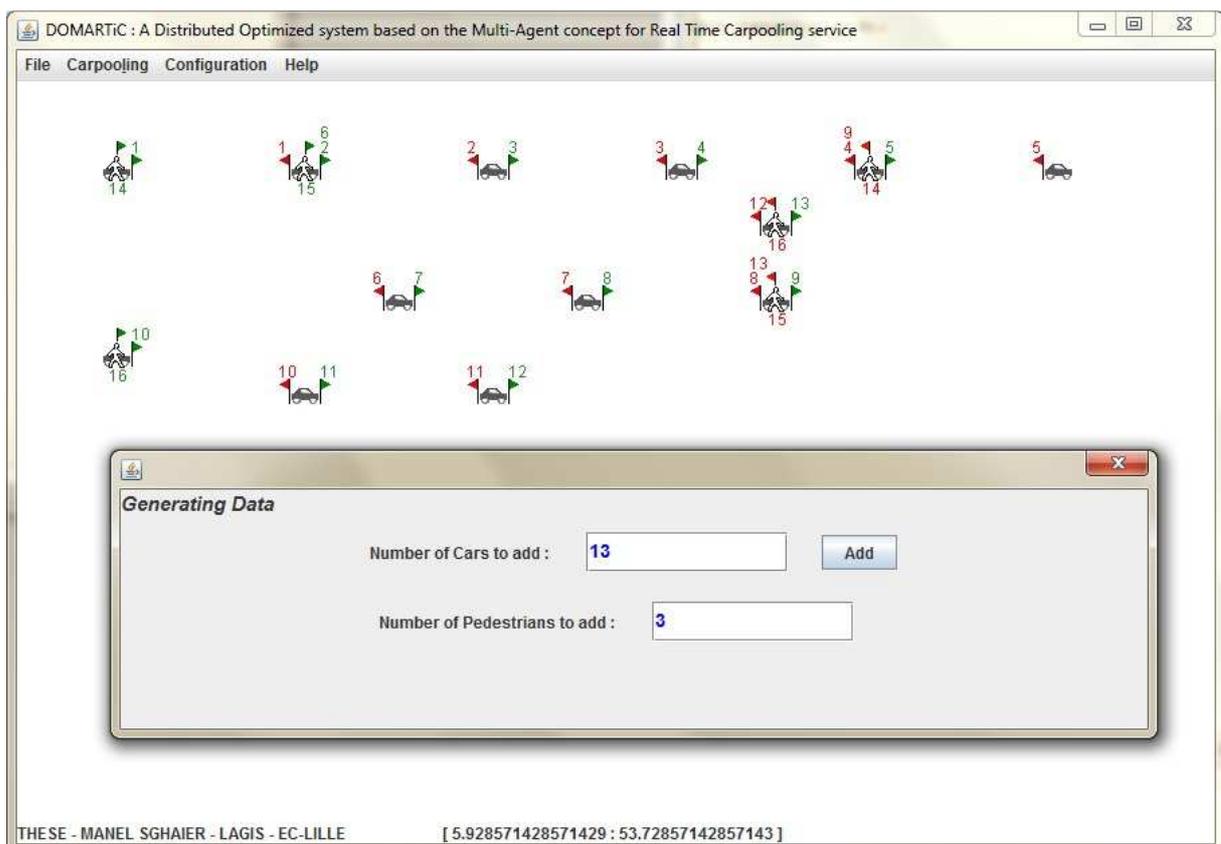


Figure V.31. Génération aléatoire de données

La Figure V.32 montre deux interfaces différentes, la première constituant la partie du haut de la figure est relative à la vue générée par le système affichant une fenêtre de choix qui

permet à l'utilisateur, à fortiori l'administrateur du système, de décider de laquelle des deux approches proposées (i.e. A3D ou ODAVe) exécuter. L'utilisateur peut ainsi jongler entre les deux méthodes pour pouvoir enfin décider laquelle lui convient le mieux; selon qu'il préfère intégrer le critère d'optimisation relatif au confort (i.e. minimiser le nombre de transferts) ou de se restreindre à celui exclusif de la durée globale du parcours.

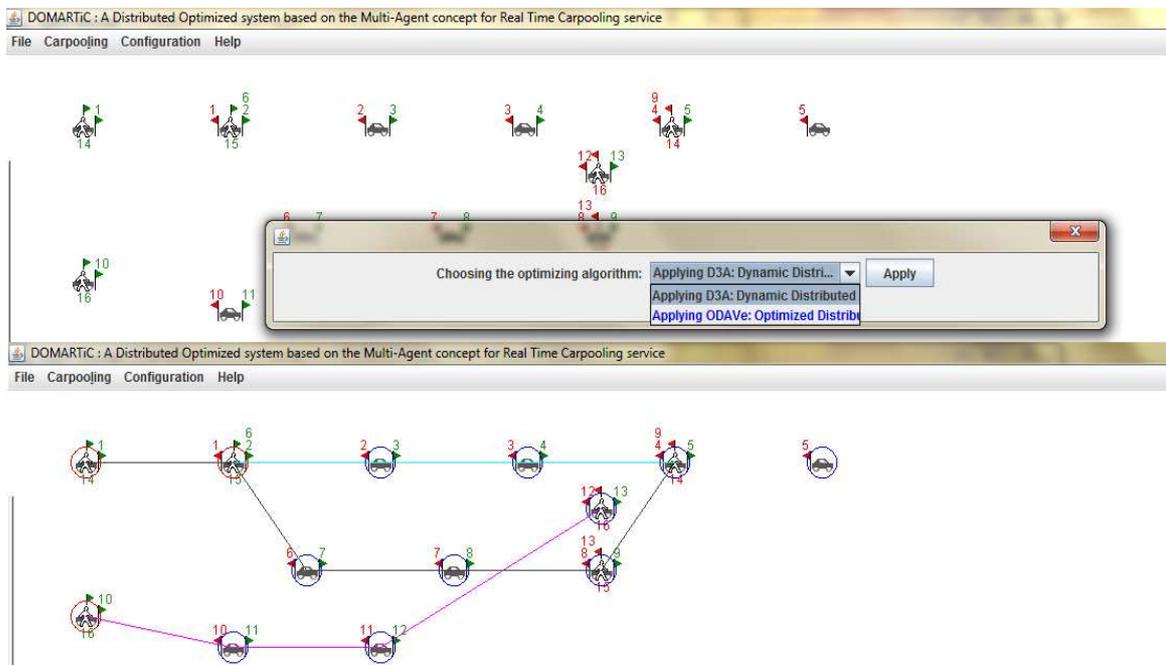


Figure V.32. Un exemple d'exécution

La seconde interface illustrée dans la deuxième partie en bas de la Figure V.32 permet de voir la manière dont sont visualisés les itinéraires solutions fournis aux usagers à la fin de l'exécution du système. Ceux-ci sont représentés par des tracés de routes reliant chaque origine de déplacement demandé à la destination qui lui correspond relativement à une demande spécifique de covoituré. Chacun des itinéraires ainsi généré est colorié de manière différente.

Par ailleurs, notons qu'au niveau de la Figure V.33, nous pouvons visualiser les différents échanges entre les agents intervenant particulièrement dans le contexte de coalition. En effet, un agent Sniffer spécifique à la plateforme multi-agent est créé pour pouvoir visualiser les différentes interactions prenant effet entre les différentes instances, entités d'agents, créées. Ces dernières se trouvent dans le conteneur principal « *Main container* ». Les multiples échanges réalisés lors de l'exécution portent essentiellement sur les envois de messages de type *Inform* et *Request* ayant eu lieu dans le cadre des interactions, collaborations et coalitions réalisées pour mener à bien tout le procédé exécuté au niveau de la plateforme *DOMARTiC* et

ce pour le traitement optimisé des requêtes usagers. Dans ce cadre, des flux informationnels aussi bien que des demandes lancées entre entités pour enclencher un comportement spécifique prennent effet entre les entités intervenantes (i.e. AI, AIC, AD, AOs).

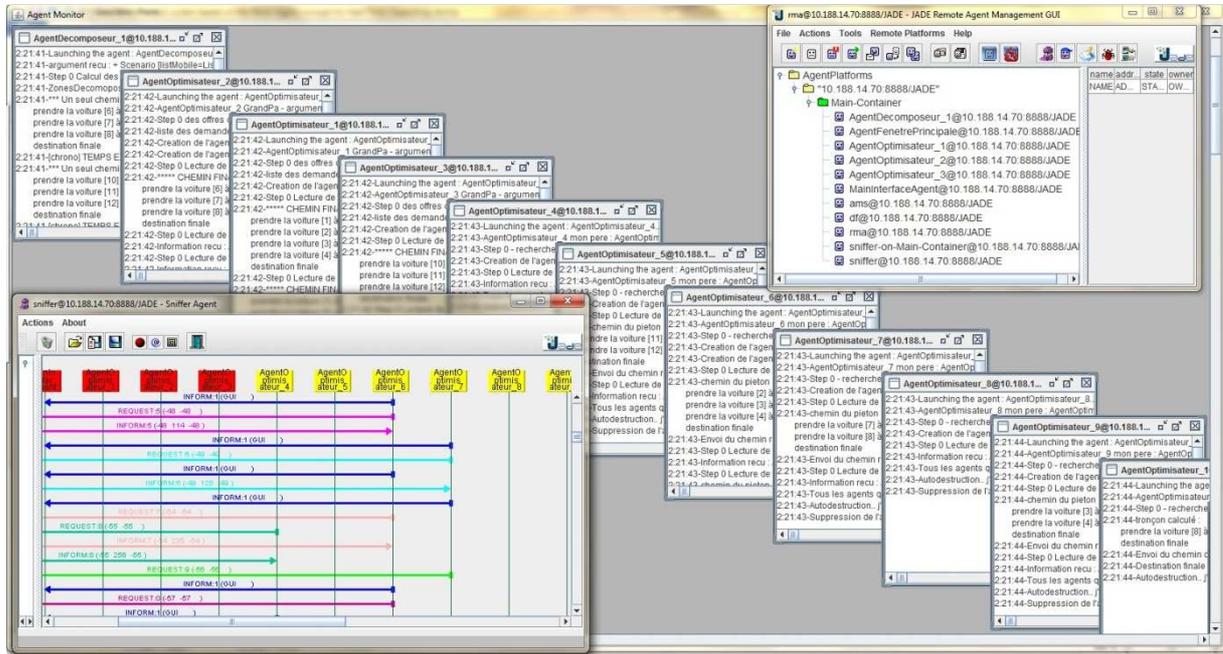


Figure V.33. Coalition entre AO et recombinaison d'itinéraires

Les différents tests de validation des résultats, par rapport à la génération de solutions réponses aux utilisateurs, ainsi générés; nous avons ainsi pu juger des performances réalisées par la plateforme DOMARTiC.

**Performances d'exécution :**

Sur la base des différents scénarios établis et testés au niveau de notre système, nous avons pu remarquer que les performances d'exécution de la plateforme DOMARTiC sont assez satisfaisantes dans la mesure où les temps d'exécution demeurent raisonnables pour les différents jeux de tests effectués. La Figure V.34 illustre effectivement une courbe où sont retracés les résultats de performances du système, par rapport au temps qu'a nécessité l'exécution de traitements spécifiques et variés. Bénéficiant d'un module de génération aléatoire des données à traiter, la variation vient de cet aspect de sélection aléatoire en premier lieu et de par la génération de celles-ci sur un ensemble de valeurs différentes, relativement au nombre d'individus intervenant au niveau du système à chaque instant  $t$  ici considéré. Nous avons, à cet effet, testé notre système CODAC sur une multitude de valeurs, les faisant augmenter au fur et à mesure afin de pouvoir valider les performances d'exécution de DOMARTiC à grande échelle. Tel qu'il est notable sur cette figure, les résultats d'exécution

réalisés cadrent bien avec le contexte où nous nous sommes placés et demeurent dans des mesures de performances et d'efficacité des systèmes dynamiques. En effet, faisant varier le nombre d'utilisateurs au total, notamment en faisant varier l'ensemble des conducteurs ainsi que celui des piétons, le système demeure dans des mesures de performances raisonnables par rapport au temps d'exécution. Notons, par exemple, que le temps requis pour le traitement de 130 véhicules et 70 piétons est de l'ordre de 15,509 secondes.

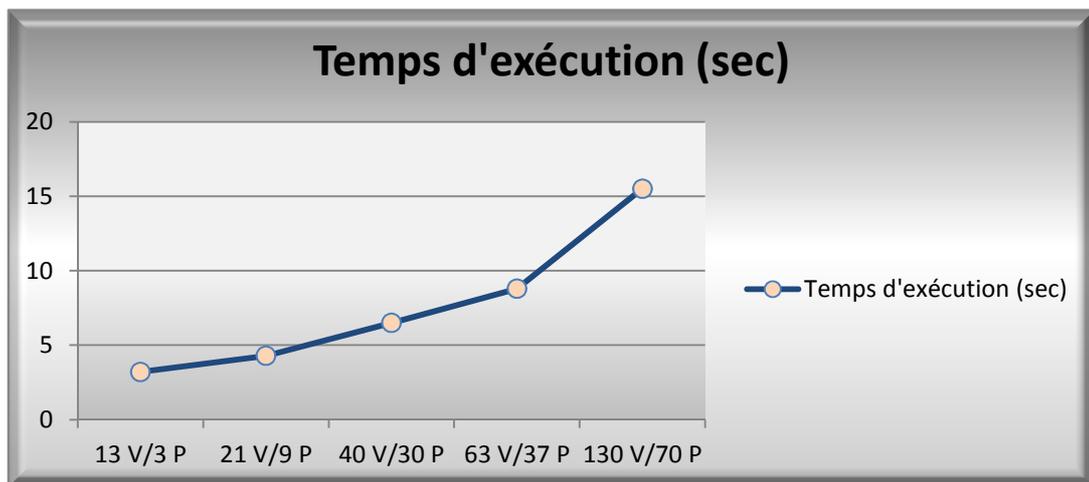


Figure V.34. Variation des Temps d'exécution

Outre cet aspect des performances d'exécution du système, nous avons focalisé notre étude sur l'aspect dominant au niveau de notre approche, à savoir la décentralisation du processus incombant au système. Dans ce contexte, et compte tenu des détails d'exécution, notamment relatifs aux différentes entitésinstanciées ainsi que les interactions ayant eu lieu entre elles (Figure V.33), nous avons fait ressortir différents types d'activité des agents. La multiplicité des catégories sous lesquelles sont classées les activités des agents a été érigée relativement à la stratégie de résolution adoptée au niveau des deux approches proposées mais aussi sur la base de la méthodologie de subdivision implémentée. A l'image de ces dernières, les activités des différentes entitésinstanciées peuvent varier.

#### Activité des agents

Une étude plus ou moins élaborée de l'ensemble des entités intervenant lors de l'exécution du système nous a permis de déceler différents types d'activité des agents que nous modélisons au niveau de la Figure V.35 et où nous introduisons la composante temps pour mettre l'accent sur les délais de réponses disproportionnels et différents d'une entité à une autre. Cette disproportion vient du fait que les charges incombant aux entités intervenantes peuvent différer, tel que c'est le cas pour la dernière courbe mais aussi et surtout par rapport

aux aspects de parallélisme qui peut être dans certains cas, non absolu, mais seulement restreints à des périodes de temps spécifiques, tel que c'est le cas au niveau de la deuxième courbe par exemple ainsi que la troisième. Les aspects de parallélisme absolu sont, par ailleurs, flagrants au niveau de la première courbe désignée par '*activité parallèle*'. Des agents peuvent, en effet, être lancés en même temps et ainsi mener leurs activités de paire, sur une même période de temps et dans une simultanéité absolue. C'est généralement le cas des différentes entités intervenant au niveau de la plateforme DOMARTiC et où il s'agit de considérer, particulièrement, la décomposition du processus en faveur de l'exécution d'une multitude de tâches menées en parallèle. Ceci est surtout le cas des *AO* lancés sur les *ZP* et qui sont enclenchés en même temps et continuent de travailler en parallèle jusqu'à l'accomplissement de leurs tâches respectives. Par ailleurs, le travail collaboratif mené dans le cadre des coalitions, aspect dominant dans nos travaux, impose de considérer une synchronisation des tâches des différentes entités, ce qui engendre une activité décalée des unes par rapport aux autres. A titre d'exemple, nous pouvons citer ici le fait qu'un *AO* n'est lancé sur une *ZI* que s'il est sollicité dans une coalition par un autre *AO* ayant déjà entamé son processus ou si un changement des perceptions de l'environnement implique son enclenchement. Ceci est le cas aussi des agents ayant fini leurs traitements préalablement aux autres mais qui ont dû être lancés après une période d'interruption dont l'arrivée d'un nouvel événement externe (e.g. requête ou offre, etc.) ou interne (e.g. requête partielle d'un autre *AO* ou *AD*, etc.) a provoqué la fin. Dans le même contexte, et à l'image de la dispersion difforme et déséquilibrée des données dans les zones érigées de par le procédé de décomposition, les différents *AO* qui en sont responsables s'en sont accommodés et sont dotés de charges plus ou moins importantes. Relativement à la quatrième courbe représentée au niveau de la Figure V.35, les entités représentées possèdent alors des activités plus ou moins importantes et assez différentes, et ce proportionnellement à la densité des individus, et spécialement les requêtes, au niveau des zones dont ils sont responsables.

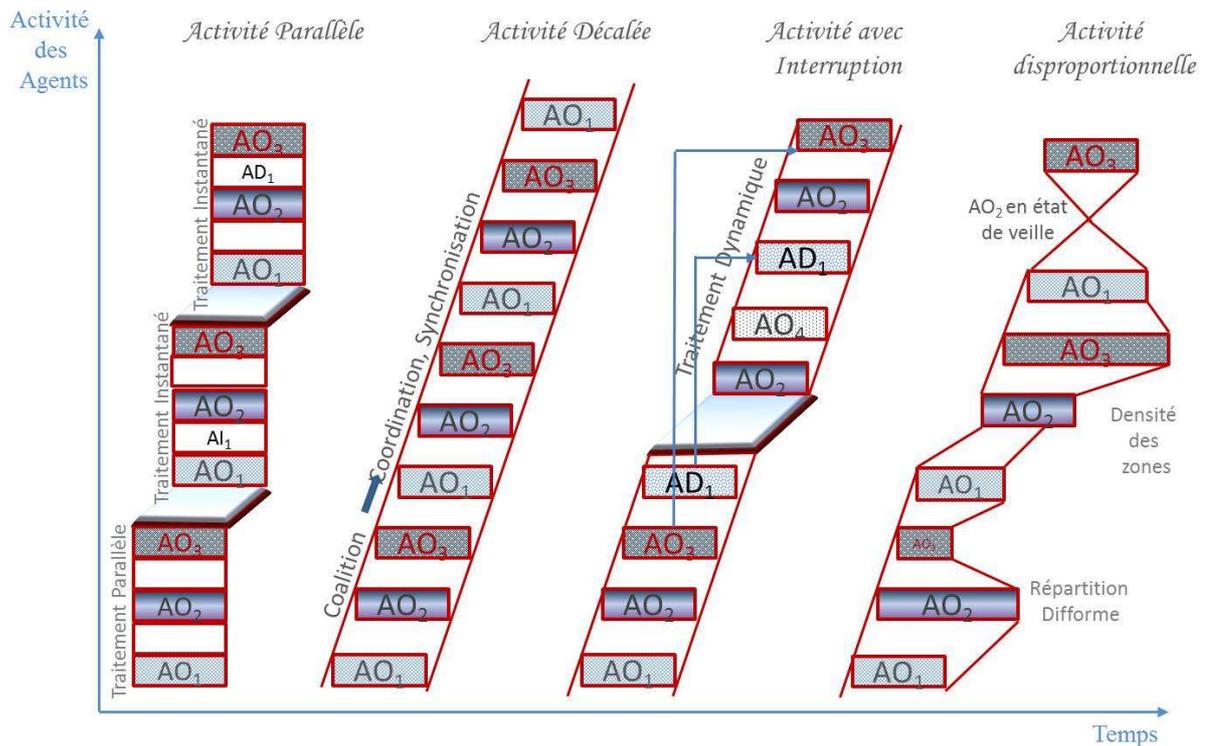


Figure V.35. Activités des Agents du système

## V.8. Conclusion

Dans le contexte des coalitions, nous cherchons à déterminer tout un jeu de solutions optimisées pouvant être recombinaées et non plus un optimum unique. De par leur nature distribuée, les algorithmes que nous proposons s'avèrent particulièrement bien adaptés à cette tâche. C'est dans ce contexte que s'inscrivent les grandes lignes qui ont fait l'objet de ce chapitre. En effet, pour faire coïncider les performances théoriques avec les performances pratiques, nous avons bien pris soin de faire les choix techniques, technologiques et architecturaux adéquats et qui ne contrarieraient pas nos projets d'optimisation de la qualité de service.

Selon nos choix stratégiques et méthodologiques d'analyse, de conception, de modélisation et de spécification de notre système, nous avons fait intervenir à tour de rôle les concepts de *GDD*, d'optimisation et d'agent avec un appui spécifique sur les collaborations et communications ayant lieu d'être dans un contexte de coalition inter agents. Afin de consolider ces choix dans la quête de la performance qui fera de **CODAC** un système compétitif à large échelle, des choix techniques appropriés s'imposaient. A signaler en particulier dans ce contexte : une plateforme *JADE* facilitant les échanges, un modèle de déploiement faisant intervenir une architecture hybride jonglant entre l'architecture

## Chapitre V : Implémentation et déploiement du service de covoiturage dynamique (CODAC)

centralisée et celle distribuée en réseau local, un modèle *MVC* prévenant toute possibilité de changement ou d'intégration de nouveaux modules (i.e. statistiques, planification et anticipation des voyages, etc.) et un module *GIS* intégré via l'interfaçage avec le progiciel *Cartocom*, etc. Toutes ces techniques et technologies réunies, elles mènent le système ainsi développé sur ces bases (*DOMARTiC* ou *CODAC*) vers l'efficacité et l'efficience (Figure V.36).

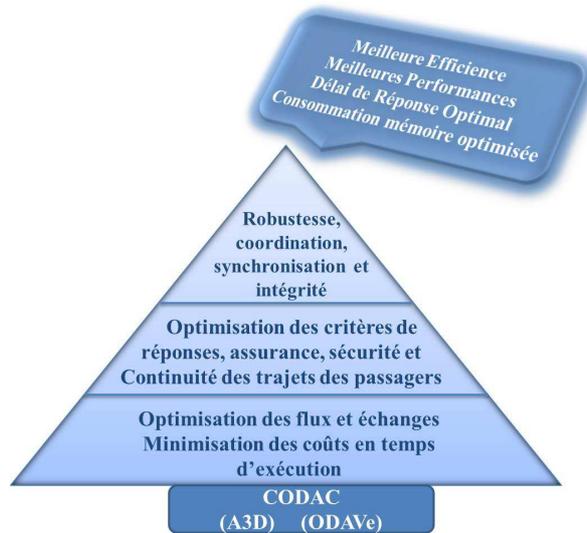


Figure V.36. Performances de CODAC



## CONCLUSION GENERALE

L'utilisation des moyens de transport collectifs offrant énormément d'avantages à l'utilisateur, les individus ont de plus en plus tendance à aller vers ces moyens les préférant parfois à leur propre voiture. Toutefois, l'automobile occupe une grande place dans leurs vies et reste toujours une source de mobilité très importante et essentielle. Ceci n'empêche pas le fait qu'elle présente plusieurs inconvénients de plusieurs points de vue (financier, environnemental, sociétal, etc.). Le covoiturage s'annonce alors comme une solution clé alliant les avantages de l'un et l'autre et a ainsi fait l'objet de plus d'un projet. Compte tenu d'une étude élaborée de ce concept, nous avons fait ressortir les grandes lignes de nos travaux se basant sur les limites de ceux existants. Nous projetons alors de par cette thèse de remédier à l'absence des aspects temps réel et automatisation qui font grand défaut aux systèmes existants. En effet, ayant placé en pleine ligne de mire l'objectif premier de mettre en œuvre un système de covoiturage efficace, performant et compétitif à grande échelle, nous considérons ses aspects dynamique, fonctionnel, automatique et essentiellement interactif avec l'utilisateur ou avec d'autres systèmes. Nos intérêts, étant entre autre d'ordre pratique, visent à mettre à la disposition du grand public, usagers particuliers ou chercheurs et praticiens une application opérationnelle et conviviale. Mais aussi une application qui peut communiquer avec d'autres systèmes facilitant ainsi l'intégration du covoiturage comme mode de transport à part entière. Par ailleurs, l'émergence notoire de la comodalité révolutionnant les habitudes et comportements dans le transport vient mettre encore plus l'accent sur la nécessité de mettre à profit la complémentarité entre le covoiturage et les autres moyens de transport. Pour appuyer l'intérêt de nos travaux, nous convoitons l'efficacité de par la considération d'objectifs essentiels visant l'optimalité de la qualité du service offert et qui tournent autour du grand axe de l'optimisation. Par ailleurs la complexité exponentielle de cette problématique démontrée, nous focalisons nos efforts sur la mise en place d'une stratégie de résolution efficace mettant à profit une mixture de concepts ; à savoir les systèmes multi-agents et l'optimisation. Une alliance de ces deux concepts riches en fondements a constitué le principal apport de nos travaux puisqu'elle exploite les profits de l'un et l'autre dans une combinaison parfaite et harmonieuse faisant régner un contexte d'intelligence artificielle distribuée. Nous faisons ainsi intervenir une optimisation tantôt locale en considérant les tâches incombant à chaque agent indépendamment des autres, tantôt distribuée sur l'ensemble des agents intervenants dans un contexte d'intelligence collective où coalition

## Conclusion Générale

inter-agents, collaboration, coordination, communication et synchronisation sont les maîtres mots. Deux approches que nous proposons au sein de notre système (CODAC) répondent à ce même principe. Une première proposition implémentant un algorithme d'affectation dynamique distribuée basée sur Dijkstra (D3A) a été réalisée pour contrer les problèmes énoncés auparavant et fournir un service de complexité d'ordre polynomial. Toujours dans une optique d'optimisation multi-objectif, un second volet concerne une heuristique combinant encore plus de critères d'optimisation. ODAVe met ainsi en place une optimisation distribuée avec les mêmes buts que la première (i.e. réduire la complexité) prenant en compte l'ordre lexicographique sur l'ensemble des critères d'optimisation (confort et durée global du trajet).

Outre les fonctionnalités supplémentaires envisageables pour étendre encore plus le champ d'application de notre système et dont nous avons parlé dans le dernier chapitre de ce rapport (e.g. module de gestion préventive et améliorative, statistiques, SIAD : tableau de bord intelligent, etc.), le concept de grappes d'agents ou sociétés d'agents dispatchés en réseau et communiquant pour faire coordonner leurs actions et résultats peut être considéré. Ceci est particulièrement utile pour faciliter le déploiement à grande échelle de notre système faisant ainsi intervenir des réseaux de covoiturage répartis à travers le monde entier. Dans ce contexte en particulier, la duplication des agents intervenant dans notre système serait utile pour pouvoir optimiser la gestion des flux et éviter le débordement.

## ANNEXE 1

Dans cet annexe sont représentés les ensembles d'aspects théoriques se rapportant à la conception et à la modélisation. Dans ces aspects, la méthodologie objet et le langage UML sont particulièrement abordés.

### o Quelques définitions élémentaires

« **Objet** » : un objet est toute entité ayant un sens dans une application quelconque du monde réel. Il s'agit d'un conteneur symbolique qui peut représenter une entité physique (e.g. un individu, un périphérique d'ordinateur ...), un concept (e.g. un processus, un procédé d'analyse de données, ...) et toute « chose » visible et tangible (un matériel, une substance, un produit...) qui possède sa propre existence (130) (131). Il incorpore des informations et des mécanismes en rapport avec l'entité du monde réel manipulée. Toute la technologie objet, la modélisation et programmation orientées objet tournent autour de ce concept central.

« **Modélisation Objet** » : La modélisation objet consiste à représenter selon un modèle bien défini en informatique les éléments de l'environnement réel du problème représenté par le système (132). Cette modélisation est, dans le cas général, indépendante des outils de programmation mis à profit pour le développement logiciel du système en question. La modélisation ainsi que la conception objet reviennent donc à déterminer les objets présents (i.e. représentant les éléments évoluant dans l'environnement) et d'isoler leurs données et les fonctions qui les utilisent et d'en créer des prototypes ou classes pour les représenter.

De multiples langages ont ainsi émergé pour définir les modèles et patrons de conception pouvant être utilisés pour modéliser un problème quelconque. Plus de 50 méthodes objets ont été proposées par différents analystes entre 1970 et 1994. Cependant, trois seulement se sont réellement distinguées. Il s'agit des trois méthodes définies dans la suite : OOSE, Booch et OMT qui ont donné lieu par la suite au Langage de Modélisation Unifié (UML : Unified Modeling Language).

« **Architecture Orientée Objet** » : une norme particulière appelée **CORBA**, acronyme de Common Object Request Broker Architecture, est un standard pour les architectures orientées objets. Ce sont des architectures logicielles établies pour représenter des applications et systèmes modélisés selon la technologie objets. Elles représentent donc les différents composants du système en question tels que produits lors de la modélisation avec

les différentes corrélations qui existent entre eux. Une architecture objet décrit de ce fait comment un système doit être conçu en se concentrant sur l'organisation de ses éléments, leurs interrelations ainsi que leurs interactions de manière à répondre aux spécifications tout en faisant abstraction des fonctionnalités propres au système en question.

Une architecture orientée objet présente l'avantage d'être dynamique puisque mettant en œuvre des applications en continuelle évolution et susceptibles d'amélioration. Le fait que ce type d'architecture soit centré sur les composantes du système et non pas sur les fonctionnalités qu'il est sensé réaliser en est la principale raison.

« **Programmation Orientée Objet** » : En programmation orientée objet, il s'agit de reproduire un objet ayant déjà été créé de par la modélisation du système en code source suivant le langage de programmation choisi. Il s'agit ici d'une méthodologie d'implémentation qui favorise la réutilisation de ce qui a déjà été implémenté. La programmation orientée objet favorise ainsi la création de programmes plus fiables à partir de composants existants. Des composants implémentés sous forme de classes appelés encore prototypes et qui sont pourvus des caractéristiques et comportements communs des entités du monde réel qu'ils représentent. La réutilisabilité des systèmes grâce au concept objet (133) présente ainsi une multitude d'avantages, notamment par rapport aux coûts et efforts, plutôt que de devoir repartir de zéro, comme c'est généralement le cas avec la programmation événementielle (i.e. approche fonctionnelle appelée encore approche structurée).

« **Classe** » : une classe est un modèle (template), une représentation abstraite d'un ensemble d'objets ayant des attributs et méthodes communs. Un objet est dit instance d'une classe. Il hérite des caractéristiques (i.e. attributs) et comportements (i.e. méthodes) d'un prototype bien déterminé. Une classe représente ainsi toutes les « choses » ayant les mêmes caractéristiques et suivant les mêmes règles. Elle se réfère donc à un ensemble d'objets ayant :

- des propriétés similaires
- des comportements communs
- et les mêmes relations

o UML : Langage de Modélisation Unifié

Une association américaine nommée **Object Management Group (OMG)** a vu le jour en 1989. Il s'agit d'une association à but non lucratif qui a pour mission de standardiser et

promouvoir le *modèle objet* sous toutes ses formes. C'est ainsi qu'elle fut l'origine des standards :

- UML : *Unified Modeling Language* ou langage de modélisation unifié qui permet de modéliser un problème de manière standard. Il s'agit d'un langage mettant en place nombre de diagrammes pour la modélisation graphique de l'ensemble des données et traitements intégrés au niveau du système à concevoir. Ce langage a été proposé pour réunir, unifier et fusionner trois autres méthodes de modélisation objet, à savoir :
  - o **OOSE** de Ivar Jacobson : OOSE pour Object Oriented Software Engineering, appelée aussi **Objectory**, est une méthode de conception utilisée à des fins d'analyse des usages de logiciels. Il s'agit d'un logiciel qui se base essentiellement sur les « *cas d'utilisation* » (134) et le cycle de vie des logiciels. Pour cela, et comme le montre la Figure IV.8, l'auteur de la méthode OOSE propose cinq modèles<sup>45</sup>.

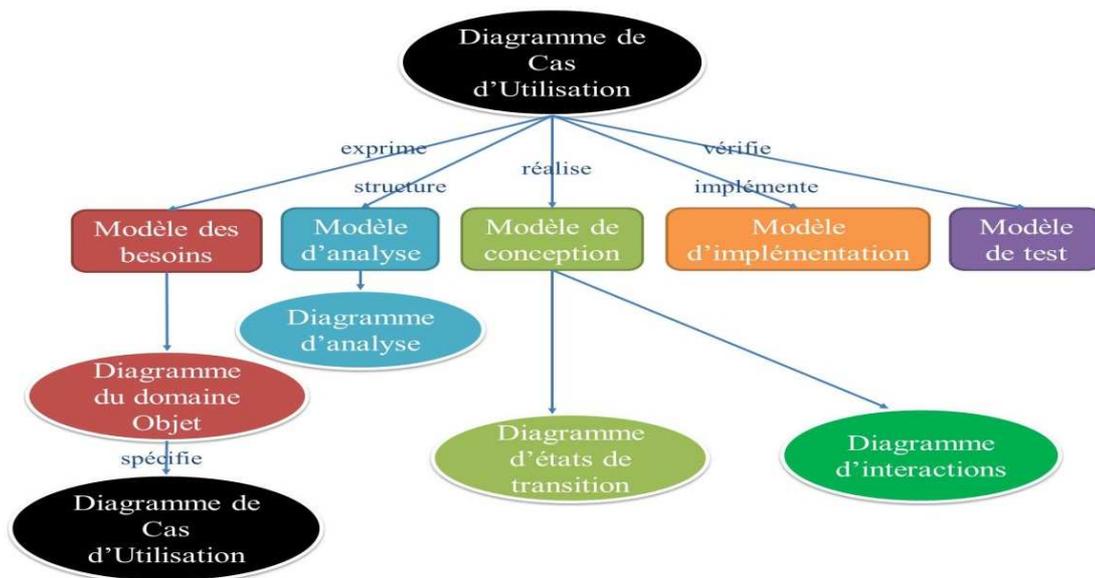


Figure A.1. Modèle OOSE<sup>46</sup>

- o **Booch** de Grady Booch : Il s'agit d'une méthode d'analyse et de conception orientée objet. L'une des principales spécificités de la méthode Booch est qu'elle permet de faciliter l'implémentation de programmes dans des langages de programmation orientée objet. Aussi permet-elle de représenter les différentes phases du développement d'un projet. La modélisation d'une solution est généralement indépendante de son implémentation dans un langage particulier (e.g. Java, C++, PHP...). L'aspect itératif ainsi que la définition d'un formalisme de conception sont à

<sup>45</sup> <http://cs-exhibitions.uni-klu.ac.at/index.php?id=448>

<sup>46</sup> <http://www.smartdraw.com/resources/tutorials/jacobson-oose-diagrams/>

retenir au niveau de cette méthode où les phases d'analyse et de conception s'entremêlent. En effet, les objets découverts lors de la phase d'analyse sont énoncés comme des candidats potentiels pour devenir des classes.

OMT (Object Modeling Technique) : est un langage orienté objet (135) pour la modélisation et la conception de systèmes logiciels élaboré par James Rumbaugh. Ce dernier a donné naissance aux concepts de base relatifs aux classes, instances et objets mais aussi les attributs et méthodes.

- MOF : Meta-Object Facility
- CORBA : Common Object Request Broker Architecture
- IDL : Interface Definition Language.

Normalisé par l'OMG en 1997, UML définit formellement l'approche objet tout en lui donnant une dimension méthodologique en se basant sur une multitude de vues du système illustrées à travers la Figure IV.9.



Figure A.2. Les vues UML (136)

Les vues décrivent l'ensemble du système d'un point de vue organisationnel, architectural, dynamique, temporel, logique, géographique, etc. Chacune comporte un ou plusieurs diagrammes qui sont au nombre de 13 dans le langage UML dans sa version 2.3. Chaque diagramme est classé sous une catégorie bien particulière, nous en distinguons 3 en tout :

1. Diagrammes structurels ou statiques : réunissent au total 6 diagrammes : de classes, de composants, d'objets, de structure composite, de déploiement et de paquetage.
2. Diagrammes comportementaux : regroupent les diagrammes de comportement, d'activités et d'états-transitions.
3. Diagrammes d'interaction ou dynamiques : modélisent les échanges inter-objets et se réfèrent aux diagrammes de séquences, de communication, de temps ou encore le diagramme global d'interaction.

UML n'étant pas une méthode, le choix des diagrammes revient à l'appréciation de chaque concepteur. Nous avons ainsi opté pour la modélisation de notre approche à un diagramme de chacune des trois catégories. Ainsi, parmi les 13 diagrammes d'UML, nous nous intéressons particulièrement au diagramme de classes placé sous le signe de la structure statique. Ce type de diagramme sera mis à profit afin de définir l'architecture logicielle de notre système.

Aussi un deuxième type de diagrammes relatif aux diagrammes comportementaux est utilisé pour la modélisation du fonctionnement des différents agents inclus dans l'architecture en question. Ces agents correspondent à des entités évoluant dans l'environnement du système et qui agissent en fonction des perceptions qu'ils en ont. Ils sont de ce fait impliqués dans la mise en œuvre de l'application de covoiturage dynamique optimisé proposée et ont chacun un rôle à jouer dans le processus englobant les différentes fonctionnalités du système. Afin de modéliser le comportement (i.e. rôle) de chacune de ces entités, nous avons opté pour les diagrammes d'activité pour en faire ressortir les différents reliefs. Un troisième type de diagramme UML correspondant aux diagrammes de séquences classé sous la catégorie des diagrammes d'interaction sera mis à profit pour définir les échanges et communications inter-agents afin d'en modéliser les coalitions ayant lieu d'être.

### o Pourquoi UML et pourquoi la modélisation objet ?

Parce qu'un tel concept favorise largement la notion de réutilisabilité induite de par les différentes propriétés de polymorphisme, encapsulation des données et traitements, généricité, héritage (137), etc. Des propriétés dont un système modélisant un problème tel que le covoiturage dynamique optimisé a besoin de s'en vêtir pour suivre l'évolution de l'environnement dans le temps (138).

UML est un langage unifié rassemblant les méthodes de modélisation objet les plus émergentes dans la littérature. Il est à ce jour le langage le plus récent et le plus complet. Dans nos travaux, nous avons donc eu recours au langage UML pour la modélisation de notre système profitant ainsi de tous les avantages de la modélisation objet. Cette modélisation se basant sur une approche modulaire, elle favorise la réutilisabilité, une bonne gestion de la mémoire de par la création et destruction d'objets au besoin, la distribution des objets, la persistance, des bases de données objet, l'héritage et le polymorphisme, l'encapsulation des données et traitements, le typage dynamique, la généricité et bien d'autres avantages. Tout cela mène à une fiabilité logicielle partant d'une conception par contrats (assertions, préconditions et postconditions, invariants de classes...) et sur la base d'une programmation

## Annexe 1

concurrente (133) (137). Des avantages qui ne sont pas aussi faciles d'accès de par une approche structurée qui se base plutôt sur les fonctionnalités du système et leur évolution.

Une architecture orientée objet est ainsi faite de modules autonomes qui facilitent l'implémentation multi-plateforme et permet une flexibilité technique accrue et une meilleure ouverture aux nouvelles technologies (telles que le multimédia)<sup>47</sup>. Chose qui n'était pas du tout évidente avec une approche fonctionnelle (139).

---

<sup>47</sup> [http://bendescamps.free.fr/cours\\_pdf/MSI\\_07a.pdf](http://bendescamps.free.fr/cours_pdf/MSI_07a.pdf)

## BIBLIOGRAPHIE

1. **Clavel, Robert.** *Le covoiturage en France et en Europe, Etat des Lieux et Perspectives.* CERTU. 2007. 88. CERTU.
2. **Feki, Mohamed Firas.** *Optimisation distribuée pour la recherche des itinéraires multi-opérateurs dans un réseau de transport co-modal.* Lille : Ecole Centrale de Lille, 2010. p. 160, Thèse de doctorat. tel-00604509, version 1.
3. **DG Energie et transport,** *A sustainable future for transport: Towards an integrated, technology-led and user friendly system.* [éd.] European Commission. Luxembourg : s.n., 2009, Office of the European Union, p. 32. 978-92-79-13114-1.
4. **Jean-François Cordeau, Gilbert Laporte, Jean-Yves Potvin, and Martin W.P. Savelsbergh.** *Transportation on demand.* [éd.] Transportation. North-Holland, Amsterdam : s.n., 2004, Handbooks in Operations Research and Management Science, Vol. 14, pp. 429-466.
5. **Alexandre Beaudry, Gilbert Laporte, Teresa Melo and Stefan Nickel.** *Dynamic transportation of patients in hospitals.* 1, 2009, OR Spectrum, Vol. 32, pp. 77-107.
6. **Manuel Iori, Juan José Salsazar Gonzalez, Daniele Vigo.** *An Exact Approach for the Vehicle Routing Problem with Two-Dimensional Loading Constraints.* 2, 2007, Transportation Science, Vol. 41, pp. 253-264.
7. **M. W. P. Savelsbergh, M. Sol.** *The general pickup and delivery problem.* [éd.] CiteSeerX. 1995, Transportation Science.
8. **S. Anily, R.Hassin.** *The swapping problem.* 1992, Networks, Vol. 22, pp. 419-433.
9. **Calvao, Roberto Diéguez.** *Comments on: Static pickup and delivery problems: a classification scheme and survey.* [éd.] Springer. 1, Juillet 2007, TOP: An Official Journal of the Spanish Society of Statistics and Operations Research, Vol. 15, pp. 37-40.
10. **Gerardo Berbeglia, Jean-François Cordeau, Gilbert Laporte.** *Dynamic Pickup and delivery problems.* 1, 2010, European journal of operational research, Vol. 202, pp. 8-15.

## Bibliographie

11. **Jean-François Cordeau, Gilbert Laporte.** *The dial-a-ride problem: models and algorithms.* 2007, Annals of Operations Research, Vol. 153, pp. 29-46.
12. **Mer, Ministère des Transport de l'Équipement du Tourisme et de la.** La situation des transports et de la mobilité au terme de l'année 2006. [En ligne] Janvier 2007. <http://www.developpement-durable.gouv.fr/Etude-transports-et-mobilite-au,3403.html>.
13. **Fraichard, Thierry.** *Cybercar: l'alternative à la voiture particulière.* 209, Paris : s.n., 2005, Navigation, Vol. 53, pp. 53-75.
14. **Dupuy, Gabriel.** *La dépendance automobile : symptômes, analyses, diagnostic.* 614-615, 1999, Anthropos, Ed. Economica, p. 160.
15. **Rocci, Anaïs.** *Les évolutions du rapport à l'automobile.* 2008. Journée ATEC-ITS, regards croisés sur l'offre de véhicules en libre service.
16. **Tortel, Lucie.** *La voiture, cet incontournable objet du désir : le rapport de l'individu à la voiture : approche psychologique, approche sémiologique, approche philosophique, approche sociologique.* Transport/Transportation, Centre d'études sur les réseaux, les transports, l'urbanisme et les constructions publiques (CERTU). 2001. p. 36.
17. **ARENE.** *Intermodalité, transport collectif et vélo : exemple de deux pôles vélo.* [éd.] ARENE – ADEME. Janvier 2002.
18. **Kammoun, Mohamed Amine.** *Conception d'un système d'information pour l'aide au déplacement multimodal : Une approche multi-agents pour la recherche et la composition des itinéraires en ligne.* Lille : Ecole Centrale de Lille, 2007. p. 181, Thèse de doctorat. tel-00142340, version 2.
19. **Zgaya, Hayfa.** *Conception et optimisation distribuée d'un système d'information d'aide à la mobilité urbaine : Une approche multi-agent pour la recherche et la composition des services liés au transport.* Ecole Centrale de Lille. Lille : s.n., 2007. p. 240, Thèse de doctorat. tel-00160802, version 1.
20. **Zidi, Kamel.** *Système Interactif d'Aide au Déplacement Multimodal (SIADM).* Ecole centrale de lille. Lille : s.n., 2006. Thèse de doctorat. tel-00142159, version 2.
21. **Kamel Zidi, Slim Hammadi.** *Algorithme génétique avec contrôle des opérateurs pour l'optimisation multicritère d'un déplacement dans un réseau de transport multimodal.* 2005, Revu électronique e-STA, Vol. 2.

## Bibliographie

22. **Hayfa Zgaya, Slim Hammadi.** *Transport Services System Integration and Optimization in Agent Based Model.* 2006, Computational Intelligence for Modelling, Control and Automation, 2006 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce (CIMCA/IAWTIC 2006), p. 29.
23. **Sidi, M.M. Ould.** *Contribution à l'amélioration des systèmes d'aide à la décision dans le domaine du transport.* Ecole Centrale de Lille. Lille : s.n., 2006. Thèse de doctorat.
24. **Mohamed Firas Feki, Slim Hammadi.** *The fastest paths in time-dependent decentralized travel information system with time-window as departure time.* [éd.] WSEAS Press. 2009, Computer and Simulation in Modern Science, pp. 129-135.
25. **Commission, European.** *European transport policy for 2010 : time to decide.* Bruxelles : Commission of the european communities, 2001. p. 124, White paper.
26. **Gille, Alain.** *La co-modalité outil du développement durable.* 2006, Transports, p. 16.
27. **Journal officiel de l'Union européenne.** *Examen à mi-parcours du livre blanc sur les transports publié en 2001.* 2007. Avis du Comité des régions.
28. *Les transports en Nord-Pas de Calais.* s.l. : Conseil économique et social régional Nord-Pas de Calais, 2008. Séance plénière d'information.
29. **Gille, Alain.** *La co-modalité outil du développement durable.* [éd.] Mendeley. 2006, Transports, p. 16.
30. **Giannopoulos, G. A.** *The Application of Co-modality in Greece: a Critical Appraisal of Progress in the Development of Co-modal Freight Centres and Logistics Services.* 2, September 2008, Transition Studies Review, Vol. 15, pp. 289-301.
31. **Philip, Christian.** *Examen à mi-parcours du livre blanc sur les transports publié en 2001.* 2007, Journal officiel de l'Union européenne, Avis du Comité des régions.
32. **Zidi, Salah.** *SARR : Système d'aide à la régulation et la reconfiguration des réseaux de transport multimodal.* Ecole Centrale de Lille. Lille : s.n., 2007. Thèse de Doctorat.
33. **Duc, Khoat Nguyen.** *Contribution à l'étude des problèmes de ré-ordonnement en cas de perturbation.* INSTITUT NATIONAL POLYTECHNIQUE DE GRENOBLE. Grenoble : s.n., 2007. Thèse de doctorat. tel-00232626, version 1.
34. **Coquio, Julien.** *La performance adaptative des systèmes de transports collectifs : Modélisation, mesures de vulnérabilité et évaluation quantitative du rôle de l'information des*

## Bibliographie

*voyageurs dans la régulation des situations perturbées.* UNIVERSITÉ FRANÇOIS - RABELAIS DE TOURS. TOURS : s.n., 2008. Thèse de doctorat.

35. **Pierre Borne, Besma Fayeche, Slim Hammadi and Salah Maouche.** *Decision Support System for Urban transportation networks.* 1, 2003, IEEE SMC Part C : Applications and Reviews, Special Issue on Decision Technologies in honour of Prof Madan Singh, Vol. 33, pp. 67-77.

36. **Laurent, GÉLY.** *Modélisation et Optimisation de la Gestion Opérationnelle du Trafic Ferroviaire en cas d'Aléas.* École doctorale de Mathématiques et Informatique - Université de Bordeaux. Bordeaux : s.n., 2010. Thèse de doctorat. tel-00551419, version 1.

37. **Giannopoulos, Georgia A.** *The Application of Co-modality in Greece: a Critical Appraisal of Progress in the Development of Co-modal Freight Centres and Logistics Services.* 2, September 2008, Transition Studies Review, Vol. 15, pp. 289-301.

38. **COMMISSION DES COMMUNAUTÉS EUROPÉENNES.** *La logistique du transport de marchandises en Europe, la clé de la mobilité durable.* Bruxelles : s.n., 2006.

39. **Mohamed Firas Feki, Slim Hammadi.** *Disturbance Management in distributed Travel Information System.* 1, September 2011, 18th IFAC World Congress (IWC), Vol. 18.

40. **Jean, M.** *Covoiturage, car sharing, taxi, etc. : Partager l'usage de l'automobile en France aussi?* [éd.] d'environnement et de circulation Association pour le développement des techniques de transport. Paris, France : s.n., Mars 2000, T.E.C. TRANSPORT, ENVIRONNEMENT, CIRCULATION, Vol. 158, pp. 22-29.

41. **Girault Maurice, Fosse Michel, Jeger François.** *Circulation et consommation de carburant en France : Estimation détaillée par type de véhicules.* [éd.] Paris-la-Défense, FRANCE (1996-2005) (Revue) Direction des affaires économiques et internationales. 131, 2000, pp. 15-22.

42. **Manel Sghaier, Hayfa Zgaya et Slim Hammadi.** Architecture basée sur les agents communicants pour la gestion optimisée des véhicules partagés. *Workshop international Logistique et Transport 2009 (LT'09).* Mars 2009.

43. **Robert Clavel, Muriel Mariotto, Benjamin Arzac.** *L'autopartage en France et en Europe : état des lieux et perspectives.* Centre d'études sur les réseaux, les transports, l'urbanisme et les constructions publiques (CERTU). 2008. p. 59.

## Bibliographie

44. **Robert, Benoît.** Histoire de l'autopartage. *Communauto*. [En ligne] <http://www.communauto.com/historique01.html> dwelles1951.
45. **Uytven, A. Van.** *Carsharing throughout europe: situation and trends*. [éd.] Mobiel 21 vzw. 2008, ATEC Technical Workshop.
46. **Manel Sghaier, Hayfa Zgaya, Slim Hammadi.** *Vue d'ensemble sur le phénomène de l'Autopartage et proposition d'un nouveau système de gestion flexible basé sur les agents*. [éd.] Man & Cybernetics Society IEEE Systems. 2009, Workshop International : Logistique et Transports (LT 2009).
47. **Benoît Robert, Marc Viviani.** *L'auto-partage et le transport en commun, ensemble pour une mobilité durable*. 2003, Intelligent Transportation Systems Conference.
48. **Karl Steininger, Caroline Vogl, Ralph Zettl.** *Car-sharing organizations: The size of the market segment and revealed change in mobility behavior*. 4, 1996, Transportation Policy, Vol. 3, pp. 177-185.
49. **Catherine Morency, Martin Trépanier, Bruno Agard, Basile Martin, Joel Quashie.** *Car sharing system: what transaction datasets reveal on users' behaviors*. 2007, Intelligent Transportation System Conference (ITSC IEEE), pp. 284-289.
50. **Robert Clavel, Philippe Legrand.** Le covoiturage dynamique, Etude préalable avant expérimentation. *Centre d'études sur les réseaux, les transports, l'urbanisme et les constructions publiques (CERTU)*. [En ligne] Janvier 2009. <http://lara.inist.fr/handle/2332/1463>.
51. **Commission Européenne.** Car-pooling : recommandations techniques pour la mise en oeuvre d'une politique de covoiturage (ICARO). *Mobilité en Wallonie*. [En ligne] 31 Décembre 1999. [Citation : 11 Octobre 2011.] [http://mobilite.wallonie.be/opencms/export/sites/be.wallonie.mobilite/fr/formation\\_informati\\_o\\_n\\_sensibilisation/cddm/catalogue/cddm\\_1371.html](http://mobilite.wallonie.be/opencms/export/sites/be.wallonie.mobilite/fr/formation_informati_o_n_sensibilisation/cddm/catalogue/cddm_1371.html).
52. **Manel Sghaier, Hayfa Zgaya, Slim Hammadi, Christian Tahon.** *ORTiC : A novel Approach towards Optimized Real Time Carpooling with an advanced Network Representation Model on siblings*. 1, Lille : s.n., Juillet 2010, 12th LSS Symposium, Large Schale Systems : Theory and application (LSS2010), Vol. 9.
53. **Ejday, Mohsen.** *Optimisation Multi-Objectifs à base de Métamodèle pour les procédés de mise en forme*. École des MINES ParisTech. Paris : s.n., 2011. p. 136, Thèse de doctorat.

## Bibliographie

54. **Bonte, M. H. A., Boogaard, A. H. Van den et Huétink, and J.** A metamodel based optimisation algorithm for metal forming processes. [éd.] Banabic D. *Advanced methods in material forming*. 2007, pp. 55-72.
55. —. An optimisation strategy for industrial metal forming processes : Modelling, screening and solving of optimisation problems in metal forming. *Structural and Multidisciplinary optimization*. 2008, Vol. 35, pp. 571–586.
56. **Yann Collette, Patrick Siarry.** *Optimisation multiobjectif*. [éd.] Eyrolles. 2212111681.
57. **Othmani, Imed.** *Optimisation multicritère : fondements et concepts*. Université Joseph Fourier de Grenoble. Grenoble, France : s.n., 1998. p. 118, Thèse de doctorat.
58. **Bouyssou, B. Roy and D.** Aide multicritère à la décision : méthodes et cas. *Economica*. 1993.
59. **J. Dréo, A. Pérowski, P. Siarry, E. Taillard.** *Métaheuristiques pour l'optimisation difficile*. 2003. 11368.
60. **Veldhuizen, D. A. Van.** *Multiobjective evolutionary algorithms : classification, analyses and new innovations*. Graduate School of Engineering, Air Force Institute of Technology, Wright Patterson AFB. Ohio, USA : s.n., 1999. Thèse de doctorat.
61. **Glover, F.** Future paths for integer programming using surrogate constraints. *Decision Sciences*. 1977, Vol. 8, 1, pp. 156-166.
62. **Laguna, F. Glover et M.** Tabu Search. *Kluwer Academic Publishers*. 1997.
63. **Fraser, A. S.** Simulation of genetic systems by automatic digital computers. *Australian Journal of Biological Sciences*. 1957, Vol. 10, pp. 484-491.
64. **A. Colorni, M. Dorgio et V. Maniezzo.** Distributed Optimization by Ant Colonies. [éd.] édité par F. Valera et al., Elsevier Publishing. *ECAL'91 – First European Conference on Artificial Life*. 1992, pp. 134-142.
65. **Xue-song Yan, Han-min Liu, Jia Yan, Qing-hua Wu.** A Fast Evolutionary Algorithm for Traveling Salesman Problem. *Third International Conference on Natural Computation, 2007. ICNC 2007*. Août 2007, pp. 85 - 90.
66. **Xiaojun Bi, Guangxin Luo.** The Improvement of Ant Colony Algorithm Based on the Inver-over Operator. *IEEE International Conference on Mechatronics and Automation*.

## Bibliographie

67. **Guangyuan Liu, Yi He, Yuhui Qiu and Juebang Yu.** Research on influence of solving quality based on different initializing solution algorithm in tabu search. *IEEE 2002 International Conference on Communications, Circuits and Systems and West Sino Expositions.* 2002, Vol. 2, pp. 1141 - 1145.
68. **Joshua W. Pepper, Bruce L. Golden, and Edward A. Wasil.** Solving the Traveling Salesman Problem With Annealing-Based Heuristics:A Computational Study. *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART A: SYSTEMS AND HUMANS.* Janvier 2002, Vol. 32, 1.
69. **Hizem, Mohamed Mejdj.** *Recherche de chemins dans un graphe à pondération dynamique : application à l'optimisation d'itinéraires dans les réseaux routiers.* Lille : Ecole Centrale de Lille, 2008. p. 155, Thèse de doctorat. tel-00344958, version 1.
70. **Fayech, Besma.** *Régulation des réseaux de transport multimodal : systèmes multiagents et algorithmes évolutionnistes.* Université des sciences et technologies de Lille (I3D) et l'Ecole Centrale de Lille (LAIL). Lille, France : s.n., 2003. Thèse de doctorat.
71. **Horn, M.E.T.** Multi-modal and demand-responsive passenger transport systems : a modelling framework with embedded control systems. *Journal of Software.* 2002, Vol. 36, 2, pp. 167-188.
72. **Wang Xiao-bo, Li Yi-jun, Sun Jin-ying.** Research on VRP of Optimization Based on Improved Two Phase Algorithm under Electronic Commerce . *International Conference on Management Science and Engineering, 2006. ICMSE '06.* . 2006, pp. 493 - 497.
73. **Dessouky, M. Diana and M.** A new regret insertion heuristic for solving large-scale dial-a-ride problems with time windows,. *Transportation Research Part B* 38. 2004, pp. 539-557.
74. **Lioris, Eugénie.** *Évaluation et optimisation de systèmes de taxis collectifs en simulation.* École Nationale des Ponts et Chaussées. Paris : s.n., 2010. p. 113, Thèse de doctorat.
75. **Tao, Chi-Chung.** Dynamic Taxi-sharing Service Using Intelligent Transportation System Technologies. *International Conference on Wireless Communications, Networking and Mobile Computing. WiCom 2007.* 2007, pp. 3209 - 3212 .
76. **Dridi, Imen Harbaoui.** *OPTIMISATION HEURISTIQUE POUR LA RÉSOLUTION DU m-PDPTW STATIQUE ET DYNAMIQUE.* Lille, France : s.n., 2010. p. 223, Ecole Centrale de Lille.

## Bibliographie

77. **Khalifa, Ismahène Hadj.** *Approches de modélisation et d'optimisation pour la conception d'un système interactif d'aide au déplacement dans un hypermarché.* LAGIS - École Centrale de Lille. Lille : s.n., 2011. p. 136, Thèse de doctorat.
78. **Manel Sghaier, Hayfa Zgaya, Slim Hammadi et Christian Tahon.** DyCSMA : A Based Multi-Agent Concept Approach for the Management of an optimized Real Time Carpooling Service. *Conférence IEEE: 18Th Mediterranean Conference on Control and Automation (MED 2010).* 2010.
79. **M. W. P. Savelsbergh, M. Sol.** *The general pickup and delivery problem* . [éd.] CiteSeerx. 1995, Transportation Science, Vol. 29, pp. 17-29.
80. **Manel Sghaier, Hayfa Zgaya, Slim Hammadi et Christian Tahon.** A Distributed Optimized Approach based on the Multi Agent Concept for the Implementation of a Real Time Carpooling Service with an Optimization Aspect on Siblings. [éd.] CSC Journals. *International Journal of Engineering (IJE).* Mai-Juin 2011, Vol. 5, 2, pp. 176-241.
81. **W. Jiao, Z. Shi.** A dynamic architecture for multi-agent systems. *In Proceedings of the 31st International Conference on Technology of Object-Oriented Language and Systems.* 1999, pp. 253-260.
82. **Kamoun, Mohamed Amine.** *Conception d'un système d'information pour l'aide au déplacement multimodal : Une approche multi-agents pour la recherche et la composition des itinéraires en ligne.* Ecole Centrale de Lille. Lille : s.n., 2007. Thèse de doctorat.
83. **Zgaya, Hayfa.** *Conception et optimisation distribuée d'un système d'information d'aide à la mobilité urbaine : Une approche multi-agent pour la recherche et la composition des services liés au transport.* Ecole Centrale de Lille. Lille : s.n., 2007. Thèse de doctorat.
84. **Ferber, J.** *Les systèmes Multi-Agents : vers une intelligence collective.* Paris, France : InterEdition, 1995.
85. **Bradshaw, J.M.** An introduction to software agents. [éd.] AAAI Press / The MIT Press. *Software agents.* 1997, pp. 3-46.
86. **Strugeon, E. Le.** *Une méthodologie d'auto-adaptation d'un système multi-agents cognitifs.* Université de Valenciennes et du Hainaut-Cambrésis. Valenciennes, France : s.n., 1995. Thèse de doctorat.
87. **Nwana, H. S.** Software agents : An overview. *Knowledge Engineering Review.* 1996, Vol. 11, pp. 1-40.

## Bibliographie

88. **R.Mandiau, E.G.L.Strugeon, A. Péninou.** *Organisation et applications des SMA.* [éd.] Lavoisier. 2002.
89. **Y. Demazeau, J. Muller.** Decentralised artificial intelligence. *Elsevier science publisher.* 1990.
90. **Russell, SJ. et Norvig, P.** *Artificial intelligence: a modern approach.* Englewood Cliffs (New Jersey) : Prentice Hall, 1995.
91. **Florez, R.** Towards a standardization of multi-agent system frameworks. *ACM Crosswords student magazine.* 1999.
92. **WEISS, G.** *Multiagent Systems. A Modern Approach to Distributed Artificial Intelligence.* Cambridge, Massachusetts : The MIT Press, 1995.
93. **DI MARZO SERUGENDO, G., et al.** *Engineering Self-Organising Systems : Nature-Inspired Approaches to Software Engineering.* 2004, Lecture Notes in Artificial Intelligence (LNAI), Vol. 2977.
94. **Weiss, G.** *Multiagent systems, a modern approach to distributed artificial intelligence.* s.l. : The MIT Press, 2000.
95. **Danfous.** *Déploiement national des systèmes d'information multimodale : DELFI, l'exemple allemand.* Centre d'Etudes sur les Réseaux, les Transports, l'Urbanisme et les constructions publiques (CERTU). 2000. Rapport technique.
96. **Adam, E.** *Modèle d'organisation multi-agent pour l'aide au travail coopératif dans les processus d'entreprises: application aux systèmes administratifs complexes.* Université de Valenciennes et du Hainaut-Cambrésis. Valenciennes, France : s.n., 2000.
97. **Roze, C. P.** *Organisation multi-agents au service de la personnalisation de l'information.* Université de Valenciennes et du Hainaut-Cambrésis. Valenciennes, France : s.n., 2003. Thèse de doctorat.
98. **E.H.Durfee, V.R.Lesser.** Partial global planning: A coordination framework for distributed hypothesis formation. *IEEE transactions on Systems, Man and Cybernetics.* Septembre - octobre 1991, Vol. 21, 5, pp. 1167-1183.
99. **M.Ljungberg, A.Lucas.** The oasis air-traffic management system. *In Proceedings of the 2nd Pacific Rim International Conference on Artificial Intelligence, PRICAI'92.* 1992.

## Bibliographie

100. **K.Fischer, J.P.Müller, M.Pishel, D.Schier.** A Model for Cooperative Transportation Scheduling. *In proceedings of the 1st International Conference on MAS. AAAI Press/MIT Press.* 1999, pp. 109-116.
101. **B.Chaib-draa.** Industrial Applications of Distributed AI. *Communications of the ACM.* Novembre 1995, Vol. 38, 11, pp. 49-53.
102. —. Interaction between Agents in Routine, Familiar and Unfamiliar Situations. *International Journal of Intelligent & Cooperative Information Systems,* . 1996, Vol. 5, 1, pp. 1-25.
103. **M. Sghaier, H. Zgaya, S. Hammadi, C. Tahon.** DOMARTiC: A Distributed Optimized approach based on the Multi Agent concept to set up a Real Time Carpooling Service. *12ème congrès annuel de la société française de recherche opérationnelle et d'aide à la décision, ROADEF2011.* 2011.
104. **F. Sottini, S. Abdel-Naby, P. Giorgini.** *Andiamo: A Multiagent System to Provide a Mobile-based Rideshare Service,* . University of Trento, Ingegneria e Scienza dell'Informazione,. Rapport Technique DIT-06-097.
105. **Kothari, Amit B.** *Genghis - a multiagent carpooling system.* Mai 2004, B.Sc. Dissertation work in Computer Science, submitted to the University of Bath.
106. **G., M., B., B., A., H.** Applications of multi agent systems in traffic and transportation. *In the IEE Transactions on Software Engineering.* 1997, Vol. 144, 1, pp. 51-60.
107. **C. Carabelea, M. Berger.** Agent negotiation in ad-hoc networks. *In Proceedings of the Ambient Intelligence, Workshop at AAMAS'05 Conference.* 2005, pp. 5–16.
108. **B. Volha, G. Paolo, F. Stefano.** *Toothagent: a multi-agent system for virtual communities support.* . Informatica e Telecomunicazioni, University of Trento. 2005. Technical Report DIT-05-064.
109. **Manuel Iori, Juan José Salazar González, Daniele Vigo.** An exact approach for the vehicle routing problem with two-dimensional loading constraints. [éd.] *Comuter Science Journals. Transportation Science.* Mai 2007, Vol. 41, 2, pp. 253-264.
110. **Gerardo Berbeglia, Jean-François Cordeau, Gilbert Laporte.** *Dynamic Pickup and delivery problems.* [éd.] *The DBLP Computer Science Bibliography.* 1, Avril 2010, *European Journal of Operational Research,* Vol. 202, pp. 8-15.

## Bibliographie

111. **Jianjun Wu, Yubo Tan.** *A particle Swarm Optimization Algorithm for Grain Logistics Vehicle Routing Problem.* 2009, ISECS International Colloquium on Computing, Communication, Control, and Management, pp. 364-367.
112. **ZHIHAI XIANG, CHENGBIN CHU, HAOXUN CHEN.** *The study of a dynamic dial-a-ride problem under timedependent and stochastic environments.* [éd.] Elsevier. 2, 2008, European journal of operational research, Vol. 185, pp. 534-551.
113. **Atahran Ahmed, Christophe Lente, Vincent T'kindt.** *Transport à la Demande et "Dynamic Dial a Ride Problem" : liens et état de l'art.* Toulouse : s.n., 2010, ROADEF 2010, 11e Congrès annuel de la société française de Recherche Opérationnelle et d'Aide à la Décision .
114. **Jean-François Cordeau, Gilbert Laporte.** *The dial-a-ride problem (DARP) : Variants, modeling issues and algorithms.* 2003, 4OR, A Quarterly Journal of Operations Research, Vol. 1, pp. 98-101.
115. **Yvan Dumas, Jacques Desrosiers, Francois Soumis, .** *The pickup and delivery problem with time windows.* [éd.] Elsevier. 1, Septembre 1991, European Journal of Operational Research, Vol. 54, pp. 7-22.
116. **Psaraftis, Harilaos N.** *An exact algorithm for the single vehicle many-to-many dial-a-ride problem with time windows.* 3, 1983, Transportation Science, Vol. 17, pp. 351-357.
117. **LIORIS, Eugénie.** *Évaluation et optimisation de systèmes de taxis collectifs en simulation.* s.l. : École Nationale des Ponts et Chaussées, 2010. Thèse de doctorat. pastel-00565617, version 1.
118. **Ammor.O, Raiss.N, Slaoui.K.** *Détermination du nombre optimal de classes présentant un fort degré de chevauchement.* Décembre 2007, La revue MODULAD, Vol. 37, pp. 31-42. [http://www-rocq.inria.fr/axis/modulad/archives/numero-37/Ammoretal-37/37\\_res\\_ammoretal.htm](http://www-rocq.inria.fr/axis/modulad/archives/numero-37/Ammoretal-37/37_res_ammoretal.htm).
119. **Jihane Alami, Abdelhakim El Imrani.** *Algorithme culturel parallèle pour l'optimisation multimodale.* Juin 2006, Journal électronique d'intelligence artificielle. <http://jedai.afia-france.org/detail.php?PaperID=63>.
120. *On agent-based software engineering.* 2, 2000, Artificial Intelligence, Vol. 117, pp. 277-296.

## Bibliographie

121. **Jennings, Nicholas R.Ferber, Jacques.** *Les Systèmes multi-agents: Vers une intelligence collective.* [éd.] Dunod. 1997. p. 522. 978-2729606657.
122. **Weiss, Gerhard.** *Multiagent systems: a modern approach to distributed artificial intelligence.* 1999.
123. **Onn Shehory, Sarit Kraus.** *Formation of Overlapping Coalitions for Precedence-Ordered Task-Execution Among Autonomous Agents.* [éd.] CiteSeer. 1996.
124. **Onn M. Shehory , Katia Sycara , Somesh Jha.** *Multi-agent Coordination through Coalition Formation.* s.l. : CiteSeer, 1998.
125. **Feki, Mohamed Firas.** *Optimisation distribuée pour la recherche des itinéraires multi-opérateurs dans un réseau de transport co-modal.* Lille : Ecole Centrale de Lille, 2010. Thèse de Doctorat. tel-00604509, version 1.
126. **Kammoun, Mohamed Amine.** *Conception d'un système d'information pour l'aide au déplacement multimodal : Une approche multi-agents pour la recherche et la composition des itinéraires en ligne.* Lille : Ecole Centrale de Lille, 2007. p. 181, Thèse de Doctorat. tel-00142340, version 2.
127. **Manh Hung, Nguyen.** *Génie logiciel orienté agent.* L'INSTITUT DE LA FRANCOPHONIE POUR L'INFORMATIQUE. 2006. TRAVAIL D'INTÉRÊT PERSONNEL ENCADRÉ (TIPE).
128. **LAICHOIR, Hakim.** *Modelisation Multi-Agent et Aide à la décision : Application à la régulation des correspondances dans les réseaux de transport urbain.* Lille 1. 2002. Thèse de doctorat.
129. **Nwana, H., et al., et al.** *ZEUS: A tool-kit for building distributed multi-agent systems.* 1999, Applied Artificial Intelligence, Vol. 13 , pp. 129–186.
130. **Sally Shlaer, Stephen Mellor.** *Object oriented System Analysis : modeling the world in data.* s.l. : Prentice Hall, 1988.
131. **David E. Monarchi, Gretchen I. Puhr.** *A research typology for object-oriented analysis and design.* [éd.] ACM Digital Library. 9, New York, USA : s.n., Septembre 1992, Analysis and modeling in software development, Vol. 35.
132. **Rebecca Wirfs-Brock, Brian Wilkerson, Lauren Wiener.** *Designing Object-Oriented Software.* s.l. : Prentice Hall, 1990. p. 341. 978-0136298250.

## Bibliographie

133. **Meyer, Bertrand.** *Conception et programmation orientées objet.* s.l. : Eyrolles, 2008. 9782212122701.
134. **Jacobson, Ivar.** *Object-Oriented Software Engineering: A Use Case Driven Approach.* s.l. : Addison Wesley, 1992. p. 552. 978-0201544350.
135. **James Rumbaugh, Michael Blaha, William Premerlani, Frederic Eddy.** *Object oriented modeling and design.* s.l. : Prentice Hall International, 1991.
136. **P-A Muller, N. Gaertner.** *Modélisation objet avec UML.* [éd.] Eyrolles. 2003. p. 202.
137. **Hugues, Anne-Marie.** *Analyse et conception objets.* 1996.
138. **Serge Demeyer, Stéphane Ducasse, Oscar Nierstrasz.** *Object Oriented Reengineering Patterns.* 2003.
139. **Audibert, Laurent.** *UML 2 : de l'apprentissage à la pratique.* [éd.] Ellipses Marketing. s.l. : Ellipses Marketing, 2009. p. 298. 978-2729852696.
140. **Wilber, Ken.** *A Theory of Everything : An Integral Vision for Business, Politics, Science and Spirituality.* Shambhala : s.n., 2001.
141. **Tranier, John.** *Vers une vision intégrale des systèmes multi-agents.* Université Montpellier II. Montpellier : s.n., 2007. Thèse de doctorat.
142. **Zidi, Kamel.** *Système Interactif d'Aide au Déplacement Multimodal (SIADM).* Ecole Centrale de Lille. Lille : s.n., 2006. Thèse de doctorat.
143. **Viroli, Mirko, Ricci, Alessandro et Omicini, Andrea.** *Operating instructions for intelligent agent coordination.* 2006, The Knowledge Engineering Review, pp. 49-69.
144. **Demazeau, Y.** *Multi-Agent Systemes : MAS.* Paris : s.n., Novembre 2000, Journée Industrie-Recherche. Les Systèmes multi-agents (SMA) et leurs applications.
145. **Ricordel, P.M.** *Programmation orientée multiagent : Développement de systems multi-agents Voyelles.* Institut National Polytechniques de Grenoble. Grenoble : s.n., 2001. Thèse de doctorat.
146. **Sabas, Arsène, Badri, Mourad et Delisle, Sylvain.** *A Multidimensional Framework for the Evaluation of Multiagent System Methodologies.* Orlando (Florida, USA) : s.n., 2002a, 6th World Multiconference on Systemics, Cybernetics and Informatics (SCI-2002), Vol. I, pp. 211-216.

## Bibliographie

147. **Sabas, Arsène, Delisle, Sylvain et Badri, Mourad.** *A Comparative Analysis of Multiagent System Development Methodologies: Towards a Unified Approach.* Vienne (Autriche) : s.n., 2002, Third International Symposium "From Agent Theory to Agent Implementation" (AT2AI-3), Sixteenth European Meeting on Cybernetics and Systems Research, Vol. II, pp. 599-604.
148. **Sabas, Arsène, Delisle, Sylvain et Badri, Mourad.** *Vers une unification des méthodologies de développement des systèmes multiagents.* Montréal (Québec, Canada) : s.n., 2001, Actes des Journées francophones pour l'intelligence artificielle distribuée et les systèmes multi-agents, pp. 327-330.
149. **Lahlouhi, A.** *Méthodologie De Développement D'environnements de développement de SMA.* 2001, Techniques et sciences informatiques, Vol. 1.
150. **Laichour, H.** *Modélisation Multi-Agent et Aide à la Décision : Application à la régulation des correspondances dans les réseaux de transport urbains.* Université des sciences et technologies de Lille. Lille : s.n., 2002. Thèse de doctorat.
151. **Bauer, B. et Mueller, J.** Methodologies and modeling languages. [auteur du livre] M. Luck, R. Ashri et M. d'Inverno. *Agent-Based Software Development.* Londo : Artech House, 2004, pp. 77-131.
152. **Zambonelli, F., Jennings, N.R. et Wooldridge, M.J.** *Developing Multiagent Systems: the Gaia Methodology.* 3, July 2003, ACM Transactions on Software Engineering and Methodology, Vol. 12.
153. **Wooldridge, M., Jennings, N.R. et Kinny, D.** *The Gaia Methodology For Agent-Oriented Analysis and Design.* 2000, Journal of Autonomous Agents and Multi-Agent Systems, Vol. 3 (3), pp. 285-312.
154. **Schreiber, A., et al.** *CommonKADS : A comprehensive Methodology for KBS Development.* 6, 1994, IEEE Expert: Intelligent Systems and Their Applications, Vol. 9, pp. 28-37.
155. **Carlos A. Iglesias, Mercedes Garijo, Jos'é C. Gonz'alez, and Juan R. Velasco.** *A methodological proposal for multiagent systems development extending CommonKADS.* [éd.] M. Musen B. Gaines. Banff (Canada) : s.n., November 1996, Proceedings of the 10th Banff Knowledge Acquisition for Knowledge-Based Systems Workshop. Track Agent-Oriented Approaches To Knowledge Engineering, Vol. 1, pp. 25–1/17.

## Bibliographie

156. **I. Jacobson, M. Christerson, P. Jonsson, and G. O. vergaard.** *Object-Oriented Software Engineering*. 1992, ACM Press.
157. *CCITT specification and description language (SDL)*. 1994. Technical report. ITU-T. Z100 (1993).
158. **C. Iglesias, M. Garrijo, J. Gonzalez and J. R. Velasco.** *Analysis and Design of multiagent systems using MASCommonKADS*. [éd.] Springer-Verlag. 1997, Intelligent Agents IV : Agent Theories, Architectures and Languages.
159. **BOOCH, Grady.** *Object-Oriented Analysis and Design with Applications (2nd Edition)*. s.l. : Addison-Wesley Professional, 1993. p. 608. 978-0805353402.
160. —. *Analyse & conception orientées objets*. s.l. : ADDISON-WESLEY (PARIS), 1994. p. 606. 978-2-87908-069-7.
161. —. *Des solutions objets : gérer les projets orientés objets*. s.l. : THOMSON (PARIS), 1997. p. 342. 978-2-84180-992-9.
162. **Rumbaugh, James.** *Modélisation et conception orientées objet*. s.l. : Masson, Prentice Hall International, 1995.
163. **James Rumbaugh, Michael Blaha, Fredrick Eddy, William Premerlani, William Lorensen.** *Object modeling technique*. s.l. : MASSON, Paris et Prentice Hall, Londres, 1995.
164. **Corporation, Rational Software.** *UML Semantics*. Santa Clara : s.n., 1997.
165. **Grady Booch, Robert A. Maksimchuk, Michael W. Engel, Bobbi J. Young, Jim Conallen, Kelli A. Houston.** *Object-oriented analysis and design with applications*. Avril : eBook, 2007.
166. **KACEM, Mohamed HADJ.** *Modélisation des applications distribuées à architecture dynamique : Conception et Validation*. Toulouse : s.n., 2008. p. 165, Thèse de Doctorat.
167. [En ligne] [Citation : 10 Septembre 2011.] <http://users.polytech.unice.fr/~hugues/GL/chapitre8.pdf>.
168. Introduction à la programmation orientée objet. [En ligne] [Citation : 10 Septembre 2011.] <http://www.besoindaide.com/ccm/poo/poointro.htm>.
169. **Bellifemine, F., Poggi, A. et Rimassa, G.** *JADE – A FIPA-compliant agent framework*. 1999, Proceedings of the The Fourth International Conference and Exhibition on The Practical Application of Intelligent Agents and Multi-Agents.

## Bibliographie

170. **Bellifemine, F., et al.** *JADE*. 3, 2003, A White Pape, Vol. 3.

171. **Barker, K.J., et al.** *A performance evaluation of the Nehalem quad-core processor for scientific computing*. 2008, Parallel Processing Letters, pp. 453-469.

## Publications Scientifiques

[1] Manel Sghaier, Hayfa Zgaya et Slim Hammadi. « *Architecture basée sur les agents communicants pour la gestion optimisée des véhicules partagés* ». Workshop international Logistique et Transport 2009 (LT'09). Mars 2009, Sousse, Tunisie.

[2] Manel Sghaier, Hayfa Zgaya, Slim Hammadi et Christian Tahon. « *DyCSMA : A Based Multi-Agent Concept Approach for the Management of an optimized Real Time Carpooling Service* ». Conférence IEEE: 18Th Mediterranean Conference on Control and Automation (MED 2010). Juin 2010, Marrakech, Maroc.

<https://controls.papercept.net/conferences/scripts/abstract.pl?ConfID=16&Number=318>

[3] Manel Sghaier, Hayfa Zgaya, Slim Hammadi et Christian Tahon. « *ORTiC : A novel Approach towards Optimized Real Time Carpooling with an advanced Network Representation Model on siblings* ». In proceeding of the 12Th LSS symposium, Large Scale Systems : Theory and Applications (LSS 2010), volume 9, Part 1. Juillet 2010, Lille, France.  
<http://www.ifac-papersonline.net/Detailed/45887.html>

[4] Manel Sghaier, Hayfa Zgaya, Slim Hammadi et Christian Tahon. « *A distributed dijkstra's algorithm for the implementation of a real time carpooling service with an optimized aspect on siblings* ». In proceedings of the 13Th International IEEE Conference on Intelligent Transportation Systems, On the Way to Intelligent Sustainable Mobility (ITSC 2010). Pages : 795 – 800. ISSN : 2153-0009. Septembre 2010, Maderia Island, Portugal.

[5] Manel Sghaier, Hayfa Zgaya, Slim Hammadi et Christian Tahon. « *DOMARTiC: A Distributed Optimized approach based on the Multi Agent concept to set up a Real Time Carpooling Service* ». In the 12ème congrès annuel de la société française de recherche opérationnelle et d'aide à la decision ROADEF2011. Mars 2011, Saint-Étienne, France.

[6] Manel Sghaier, Hayfa Zgaya, Slim Hammadi et Christian Tahon. « *A Distributed Optimized Approach based on the Multi Agent Concept for the Implementation of a Real Time Carpooling Service with an Optimization Aspect on Siblings* ». International Journal of Engineering (IJE). Volume: 5, Issue: 2, Pages: 176-241, ISSN (Online): 1985-2312. May / June 2011. CSC Journals, Kuala Lumpur, Malaysia.

[7] Manel Sghaier, Hayfa Zgaya, Slim Hammadi et Christian Tahon. « *A novel approach based on a Distributed Dynamic Graph modeling set up over a subdivision process to deal with distributed optimized real time carpooling requests* ». In proceedings of the 14Th

International IEEE Annual Conference on Intelligent Transportation Systems (ITSC 2011).  
Octobre 2011, Washington, USA.

[8] Manel Sghaier, Hayfa Zgaya, Slim Hammadi et Christian Tahon. « *An optimized dynamic carpooling system based on communicating agents operating over a distributed architecture* ». In proceedings of the 11Th International IEEE Conference on Intelligent Systems Design and Applications (ISDA 2011). Novembre 2011, Cordoue, Espagne.

## **Combinaison des techniques d'optimisation et de l'intelligence artificielle distribuée pour la mise en place d'un système de covoiturage dynamique**

### **Résumé**

Dans le but de remédier aux problèmes aujourd'hui omniprésents dans le secteur du transport, qu'ils soient financiers, environnementaux ou autres, nous nous intéressons à l'établissement d'un système de covoiturage dynamique optimisé. La voiture partagée est venue subvenir à des besoins restés insatisfaits en matière de déplacement (flexibilité spatiotemporelle...) encourageant l'émergence d'un mode de transport révolutionnaire qu'est la comodalité. Le focus est alors mis sur la complémentarité entre les modes collectifs et individuels et vient considérer la voiture partagée et plus particulièrement le covoiturage comme des modes de transport à part entière. Placés dans ce cadre, nous nous intéressons à l'aspect temps réel dans les systèmes de covoiturage et développons nos travaux dans ce sens. Ce problème ayant une complexité qui n'est pas des moindres, tous nos efforts sont dirigés dans le but de contrecarrer cet obstacle et mettre en œuvre une application logicielle compétitive à grande échelle offrant satisfaction et qualité de service. Pour ce faire, nous considérons une alliance des systèmes multi-agents et des techniques d'optimisation donnant lieu à des agents optimisateurs répartis selon une modélisation de graphe dynamique distribué. Celui-ci est établi sur la base d'un principe de décomposition du réseau géographique desservi inspiré des techniques de classification pour la mise en exergue des zones de concentration des abonnés. Cette modélisation favorise le traitement parallèle des requêtes de par la décentralisation et décomposition du processus initial sur une multitude d'agents optimisateurs chargés chacun d'une ou plusieurs tâches de moindre complexité.

**Mots-Clés :** Covoiturage dynamique, Systèmes Multi-Agents, Optimisation, Classification et Décomposition, Graphe Dynamique Distribué, Coalition et Communication, Mobilité avancée

### **A combination of optimization and distributed artificial intelligence techniques to set up a dynamic carpooling service**

#### **Abstract**

In an attempt to address the transportation problems now ubiquitous, may them be financial, environmental or any, we are mainly involved with the establishment of a dynamic optimized carpooling service. Shared cars came to remedy these problems and meet the longtime remained unsatisfied needs (spatiotemporal flexibility...) and so promote the comodal practice. The stress is then put on the complementarity between collective and individual means of transportation and comes to confirm the shared car and more particularly the carpooling as a transport mode as a whole. Based on this, we are mainly interested in setting up a real time ridesharing service providing the needed efficiency in such a context. In fact, the problem we tackle has a complexity of exponential order which must be wiped out preventing from adverse impacts. Blending the agent paradigm with the optimization technics helped reach our goals of implementing a large-scale competitive and fully automated support and providing the necessary efficiency and quality of service. The proposed alliance is realized through communicating optimizing agents spread according to a distributed dynamic graph modeling. The latter is established through a subdivision process of the served geographic network and has been inspired from clustering technics to put the stress on limited and intersecting areas of high density. This helps to promote the parallel requests treatment over a decentralized process. Thus, each optimizing agent firstly manage the requests parts included within the zone it is responsible for and then recompose global responses in coalition with concerned agents in a distributed artificial intelligence context.

**Key-Words:** Dynamic Carpooling, Multi-Agent Systems, Optimization, Clustering and Decomposition, Distributed Dynamic Graph, Coalition and Communication, Advanced mobility

