

Ministère de l'enseignement supérieur et de la recherche scientifique

Université Abderrahmane Mira Béjaïa

Faculté des sciences exactes

Département d'Informatique



Projet de fin d'étude

En vu d'obtention d'un Master professionnel en Informatique

Option : Génie logiciel

Thème

Proposition d'une solution pour aider à la régulation des réseaux du transport multimodal

Réalisé par :

✓ BOUAKLINE Farouk

Encadré par :

✓ Mr. OUZEGGANE Redouane

Devant le jury

Mr. MIR Foudil

Président

Université de Béjaïa

Mr. NAFI Mohammed

Examineur

Université de Béjaïa

Mme. KESSIRA Dalila

Examineur

Université de Béjaïa

Année universitaire : 2016/2017

“Les thèses les plus fausses sont souvent les plus belles.”

Pierre Daninos

Remerciements

Ce travail s'est réalisé grâce au soutien et à l'encouragement de plusieurs personnes qui ont contribué chacune à leur façon à l'atteinte de l'objectif que constitue le dépôt de ce mémoire. Je tiens à leur adresser un très grand merci.

Je voudrais tout d'abord adresser toute ma gratitude à mon encadreur, pour sa patience et surtout ses judicieux conseils, qui ont contribué à la réalisation de ce projet.

Je désire aussi remercier mon père et ma mère pour leur patience et leur soutien moral.

Mes remerciements sont aussi chaleureux pour mes frères et sœurs, qui m'ont guidé avec leurs conseils précieux.

Enfin, finalement, à toutes les autres personnes qui ont gravité autour de moi, qui m'ont supporté et encouragé, Je vous dis un grand merci.

J'espère que ce mémoire saura combler vos attentes.

Dédicaces

En mémoire de ma grand mère que dieu l'accueil dans sont vaste paradis.

Je dédie ce modeste travail à

Ma chère mère, tout mon respect et affection en témoignage de ton soutien, patience et sacrifice, mon très cher père, mes frères, mes soeurs, mes petits neveux, et mes amis, qui ont ménagé tous les efforts afin de rependre à mes besoins ;

A tous ceux qui me sont chers ;

A toi Samou

TABLE DES MATIÈRES

Remerciements	ii
Dédicaces	iii
Sommaire	iv
Liste des figures	vii
Liste des tableaux	viii
Abréviations	ix
Introduction générale	1
1 Transport multimodal et le problème de régulation	3
1.1 Introduction	3
1.1.1 Classification des modes du transport	3
1.2 Transport multimodal	5
1.2.1 Configuration d'un réseau de transport	7
1.2.1.1 Configuration physique	7
1.2.1.2 Configuration horaire	8
1.2.2 Configuration d'un réseau multimodal	9
1.2.3 Exploitation d'un réseau de transport urbain	10
1.3 Problème de régulation	11
1.3.1 La recherche en transport pour aider à la régulation	11
1.3.2 Processus de planification	12
1.3.3 Perturbation	12
1.3.4 Régulation du trafic	13
1.3.4.1 Définition	13
1.3.4.2 Objectif de la régulation	14

1.4	Conclusion	14
2	Algorithmes de colonies de fourmies	15
2.1	Introduction	15
2.2	Les métaheuristiques	15
2.2.1	L'approche constructive	16
2.2.2	L'approche de recherche locale	16
2.2.3	L'approche évolutive	16
2.3	Comportements collectifs des insectes sociaux	17
2.3.1	Auto-organisation et comportement	17
2.3.1.1	Auto-organisation	17
2.3.1.2	Stigmergie	18
2.3.1.3	Contrôle décentralisé	18
2.3.1.4	Hétérarchie dense	18
2.3.2	Optimisation naturelle : piste de phéromones	19
2.4	Les fourmis artificielles et les fourmis naturelles	20
2.4.1	Ressemblances [BEN13]	20
2.4.2	Différences [BEN13]	21
2.4.3	Avantage et inconvénients des fourmis artificielles	22
2.4.3.1	Avantages	22
2.4.3.2	Inconvénients	23
2.5	Application des algorithmes de colonies de fourmis	23
2.6	Algorithme de colonies de fourmis(ACF) pour l'optimisation	24
2.6.1	Variantes	24
2.6.1.1	Algorithme de base(AS)	24
2.6.1.2	Ant System & élitisme	26
2.6.1.3	ANT_Q	26
2.6.1.4	ANT COLONY SYSTEM (ACS)	27
2.6.1.5	ACS & 3-opt	28
2.6.1.6	MIN-MAX ANT SYSTEM (MMAS)	28
2.6.2	Paramètres d'un algorithme ACO	29
2.6.3	Intensification et diversification	29
2.6.4	Représentation du problème	30
2.6.5	Comportement des fourmis	30
2.7	Conclusion	31
3	Algorithme de colonies de fourmis pour la reconfiguration spatiale	32
3.1	Introduction	32
3.2	ACF pour la reconfiguration spatiale	32
3.3	Implémentation et Tests	36
3.3.1	Outils informatiques utilisés	36
3.3.2	Codage des données	37
3.3.3	Codages des solutions	37
3.3.4	Testes	38

3.3.4.1	Nombre de fourmis	38
3.3.4.2	Probabilité de choix de transition	39
3.3.4.3	Quantité initiale de phéromone et taux d'évaporation . .	41
3.4	Conclusion	42
4	Proposition d'une amélioration	43
4.1	Introduction	43
4.2	Présentation de la proposition	43
4.3	Solutions	44
4.4	Implémentation	45
4.4.1	Classes et méthodes de l'algorithme	45
4.4.2	Résultats	46
4.4.2.1	Pour la recherche de la première solution	46
4.4.2.2	Résultats sur la probabilité de transition :	47
4.4.2.3	Résultats sur le nombre de fourmis utilisées :	47
4.5	Conclusion	48
	Conclusion générale et perspectives	49

TABLE DES FIGURES

1.1	Structure d'une ligne[AMR12]	7
1.2	Exemple de réseau multimodal[ZID06]	9
1.3	Exploitation des réseaux multimodals[ZID07]	10
1.4	Processus classique de planification[ZID06]	12
1.5	Fonctionnement d'une ligne en temps réel[AMR12]	13
2.1	Hiérarchie (a) et heterarchie dense (b) : deux concepts opposés [BEN13] .	19
2.2	Experiance de sélection des branches les plus courtes par une colonie de fourmis [ZID07]	20
3.1	Un exemple d'un réseau de transport	37
3.2	La matrice d'adjacence du réseau précédent	37
3.3	Représentation d'une solution	38
3.4	L'influence de nombre de fourmi sur la convergence Benchmark 1	39
3.5	L'influence de nombre de fourmi sur la convergence Benchmark 2	39
3.6	influence de la probabilté de transition sur les pistes de phéromone	40
3.7	L'influence de la probabilité de transition sur la convergence	41
4.1	La quantité de phéromone finale lors de la recherche de la solution	44
4.2	ACFRS amélioré	45
4.3	La quantité de phéromone finale	46
4.4	Influence de nombre de fourmis sur la convergence après amélioration . . .	48

LISTE DES TABLEAUX

3.1	influence de nombre de fourmi sur la convergence	38
3.2	influence de la probabilté de transition sur les pistes de phéromone	40
3.3	influence de la probabilté de transition sur la convergence	40
3.4	Influence de tau d'évaporation sur la convergence	41
3.5	Influence de la quantité initiale de phéromone sur la convergence	41
4.1	influence de la probabilté de transition sur la convergence après amélioration	47
4.2	influence de nombre de fourmi sur la convergence	47

ABRÉVIATIONS

ACF	Algorithme de colonies de fourmis
ACFRS	Algorithme de colonies de fourmis pour la régulation spatiale
ACO	Ant Colony Optimization
PVC	Problème de Voyageur de Commerce
SAE	Systemes d'Aide à l'Exploitation
SAD	Systemes d'Aide à la Décision
SI	Système d'Information
TM	Tableau de Marche

INTRODUCTION GÉNÉRALE

De nos jours, des milliers d'usagers utilisent quotidiennement les réseaux du transport. Dans des zones urbaines à fortes densité de population, ces réseaux transportent un nombre de plus en plus important de personnes et deviennent ainsi un service indispensable. En contre partie, cette dernière augmentation entraine une augmentation dans le nombre de véhicule ainsi des problèmes aigus sont posés.

Un réseau du transport multimodal est caractérisé par la présence du plusieurs modes du transport. La régulation d'un réseau du transport est un ensemble de méthodes et de mesures qui esseyent de mettre de l'ordre dans le réseau perturbé afin de se rapprocher de la table de marche initiale(TM). Cette dernière devient de plus en plus difficile, lorsque plusieurs perturbations sont apparues, pour le régulateur, d'où la nécessité d'apparition des solutions informatiques .

Les métaheuristiques sont des algorithmes d'optimisation visant à résoudre des problèmes d'optimisation difficile. Il existe un grand nombre de métaheuristiques différentes ; l'une d'elles consiste à utiliser un algorithme issu de l'étude des comportements des colonies de fourmis.

Cet algorithme est appliqué sur plusieurs problème d'optimisation, l'une de ses variante sert à effectuer une reconfiguration spaciale sur le réseau perturbé. Dans ce travail, nous étudions cet algorithme de prés afin que nous puissions apporter des améliorations.

Pour la réalisation de ce projet et le mener à bien nous avons opté pour un plan de travail s'articulant sur quatre chapitres.

- ✓ Le premier chapitre est consacré à la présentation de quelques définitions sur le transport multimodal, sa configuration et sa planification.
- ✓ Le chapitre suivant présente une brève étude sur les algorithmes de colonies de fourmis, ses principales variantes et leurs applications.
- ✓ Dans le troisième chapitre nous avons présenté l'algorithme de colonie de fourmis pour la reconfiguration spatiale.
- ✓ Notre proposition fera l'objet du dernier chapitre. Nous terminons ce mémoire avec une conclusion générale et perspectives.

CHAPITRE 1

TRANSPORT MULTIMODAL ET LE PROBLÈME DE RÉGULATION

1.1 Introduction

La notion de multimodalité dans les réseaux de transport collectif fait l'objet de différentes thématiques de recherche liées à l'aménagement des réseaux, à l'information multimodale et à la planification et la régulation du trafic. Les systèmes informatiques dans le domaine du transport sont nombreux. Ces systèmes sont divisés en deux catégories, les systèmes pour les exploitants et les systèmes pour les clients. Nous trouvons des systèmes d'aide à la décision côté régulateur (SAD), des systèmes d'aide à l'exploitation (SAE) pour le côté exploitant et des systèmes d'information (SI) pour le côté client.

1.1.1 Classification des modes du transport

Un mode de transport, ou moyen de transport, ou système de transport, est une forme particulière de transport qui se distingue principalement par le véhicule utilisé, et par conséquent par l'infrastructure qu'il met en œuvre. Un mode de transport désigne le

moyen utilisé pour transporter un objet (ou une personne) d'un lieu à un autre. Lorsque plusieurs modes de transport sont associés pour concourir à la réalisation d'une opération de transport, c'est le terme multimodalité qui est utilisé.

il existe différents moyens de transport, tel que la voiture, la motocyclette et le vélo, qui sont des moyens de transports individuels. Il y a également les moyens de transport en commun, tels que l'autocar, le métro, le train, le taxi,... Il y a également le bateau et l'avion, qui sont en général des moyens de transport de masse utilisés pour le déplacement sur de longues distances.

Par ailleurs, nous pouvons distinguer les transports selon différents critères : le nombre de personnes transportées (les transports individuels et les transports collectifs), la nature du service rendu, et le nombre de modes utilisés [MEJ12].

1. Classification fonctionnelle des moyens de transport :

- (a) **Les transports individuels** : C'est le cas où une seule personne utilise, pour son déplacement, un moyen de transport particulier et pour son propre compte tels que l'automobile privée, les taxis publics...
- (b) **Les transports collectifs** : Les transports collectifs ou en commun, consiste à transporter plusieurs personnes ensemble sur un même trajet. Ils incluent tous les moyens de transport en commun dont à titre d'exemple les métros, bus, trains, tramway, les services réguliers publics tels que les transports scolaires.

2. Classification du transport selon la nature du service rendu :

- (a) **Le transport à la demande** : Il s'agit des services collectifs individualisés 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.
- (b) **Le transport occasionnel** : Ce type de transport n'est utilisé que lors d'occasions particulières (e.g. organisation d'une colonie de vacances, une visite touristique, un évènement spécifique : concert, exposition...).

3. Classification des transports selon le(s) mode(s) adopté(s) :

- (a) ***Le transport uni-modal (ou monomodal)*** : Ce type de transport implique l'usage lors d'un déplacement, d'un seul mode de transport. Il désigne le fait d'utiliser un type spécifique de véhicule (Bus, Métro, Tram, voiture particulière...).
- (b) ***La complémentarité modale (le transport comodal)*** : La comodalité est définie comme étant « la combinaison optimale des différents modes sur la chaîne de transport ». 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) ***Le transport multimodal*** : Ce mode de transport sera notre domaine d'étude, il sera expliqué dans la section suivante.

1.2 Transport multimodal

Un mode de transport peut être assuré par plusieurs opérateurs, ainsi un itinéraire donné peut être intra ou multi opérateur et donc exploité de différentes manières par un seul ou par plusieurs opérateurs. De ce fait découle le concept de la multimodalité qui se définit par la possibilité de plusieurs modes de transport (bus, métro, tramway, TGV, etc.) pour un même déplacement entre une origine et une destination. Ceci explique pourquoi il existe plusieurs combinaisons de modes de transport pour tracer un seul trajet entre une origine et une destination [ZGA07].

L'information multimodale : Elle sert à renseigner tout « utilisateur » d'un quelconque mode de transport (route, fer, air, mer, fleuve, vélo ...) sur les modalités et conditions d'un déplacement ou d'un voyage qu'il souhaite effectuer ou qu'il effectue. Cette notion d'utilisateur doit être prise au sens large : il peut s'agir d'un voyageur « utilisateur potentiel » de l'un des modes de transport, mais aussi de tout organisme (public ou privé) intéressé pour ses besoins propres par l'exploitation de cette information (agence de voyage, office de tourisme, agence d'urbanisme, collectivité locale, localisation de voiture, opérateurs télécoms, fournisseurs de service à valeur ajoutée ...)[ZID06].

Les services d'information multimodale Ils sont diffusés par différents canaux et supports. Ils fournissent des fonctions aussi bien de calcul d'itinéraires sur le Web ou sur le minitel, que des affichages ou diffusions d'horaires (guichet, Web, téléphone, bornes, panneaux lumineux ...), que des conditions de circulation transmises par radio ou sur le Web. Ils diffusent des informations sur les réservations, sur les tarifs, sur la disponibilité de places de parking, sur des temps d'attente, et ce aussi bien « en théorie » qu'en temps réel[ZID06].

Le transport multimodal regroupe différentes formes et aspects de transport. On note le transport intermodal et le transport combiné[MEJ12].

1. Le transport intermodal :

Le transport intermodal représente un cas particulier du transport multimodal du fait qu'il s'agit d'utiliser plus qu'un mode de transport pour se déplacer mais dans ce cas sans rupture de charge. Il constitue un système de transport en vertu duquel deux modes de transport ou plus sont utilisés pour permettre une chaîne de transport de porte à porte. Ceci en utilisant des unités de charge spécifiques (pour le transport de marchandises) et des pôles d'échange (pour le transport de personnes).

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 donnée. Ainsi, deux modes de transport ou plus peuvent être utilisés pour réaliser un trajet complet entre deux points.

2. **Le transport combiné :** Le transport combiné est une forme de transport intermodal. Par rapport au transport multimodal et intermodal, le transport combiné se limite à l'utilisation de deux modes de transport seulement. Ce mode de transport combine généralement, les modes route-fer et route-mer. Lorsque seuls les modes de transport routier et ferroviaire sont utilisés, on parle de transport combiné rail/route ou ferroulage. Le transport combiné permet d'allier la productivité du fer et de la voie d'eau avec la flexibilité de la route.

1.2.1 Configuration d'un réseau de transport

1.2.1.1 Configuration physique

Physiquement, un réseau de transport est constitué d'un ensemble de lignes et de dépôts. Une ligne contient des arrêts. Parmi ces arrêts, on distingue plusieurs types :

Arrêt simple : caractérisés par des horaires de départ et des horaires d'arrivées des véhicules.

Arrêt de régulation : caractérisés en plus des arrêts simples, par la localisation des véhicules (arrêts équipés de capteurs) et la possibilité de commander des manoeuvres de régulation.

Arrêts terminus caractérisés, en plus des arrêts de régulation, par des véhicules (vidés) des clients.

Arrêts noeud de correspondances : un arrêt de correspondance est un endroit d'échange de voyageurs entre deux modes de transport à une date donnée. Donc, une correspondance est par définition est spatio-temporelle.

On trouve sur une ligne au moins deux arrêts terminus. Lorsqu'une ligne comporte plus que deux arrêts terminus, elle est dite à antennes. Les structures de ligne X ou Y forment des antennes. Certains arrêts d'une ligne offrent la possibilité de commander des demi-tours et des possibilités de garage de véhicules [AMR12].

La liaison entre deux arrêts consécutifs est appelée « **inter arrêts** ». Les structures qu'on peut trouver dans une ligne sont représenté dans la figure (1.1).

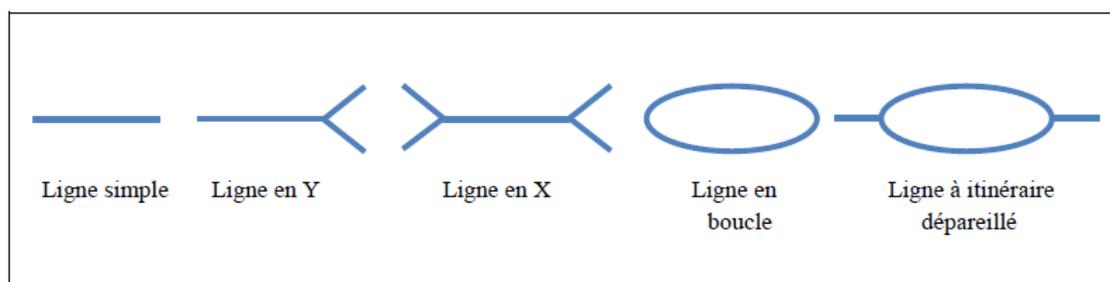


FIGURE 1.1: Structure d'une ligne[AMR12]

Dans un réseau de transport urbain. On trouve deux types de dépôts [AMR12] :

Dépôt d'exploitation : assure la préparation et le contrôle d'un service sur une ligne.

Dépôt de remisage : assure l'affectation des conducteurs, l'entretien des véhicules et le remisage proprement dit qui permet d'acheminer un véhicule du dépôt à l'itinéraire commercial et réciproquement.

Logiquement, une ligne contient plusieurs itinéraires [AMR12] :

- ✓ Des itinéraires commerciaux.
- ✓ Des itinéraires spéciaux.

1.2.1.2 Configuration horaire

La conception de l'horaire d'une ligne est une tâche assez complexe qui nécessite la définition de tous les itinéraires de cette ligne. Le travail manuel peut être assez lourd puisque la vitesse peut varier en cours de journée (heure de pointe) et que l'offre varie suivant les jours (jours ouvrables ou week-ends) et les périodes (scolaires, vacances ...).

Afin de pouvoir définir l'offre d'une ligne, une représentation en graphique de l'ensemble des parcours au cours d'une journée est nécessaire. Cette opération, appelée graphichage, définit l'heure de passage des véhicules aux arrêts, la vitesse, les temps de battements, le nombre de véhicules nécessaires et le nombre d'heures de conduite nécessaire à la satisfaction de la demande sur ligne.

Lorsque ces données sont définies, il faut répartir les heures de travail à effectuer par les conducteurs, c'est l'opération dite d'habillage¹ des services.

1. L'habillage est un exercice fondamental pour le fonctionnement de toute entreprise de transport urbain, puisqu'elle détermine, d'une part, les plannings de travail des conducteurs et doit, d'autre part, veiller à minimiser les coûts et ainsi construire une solution acceptable pour l'entreprise.

1.2.2 Configuration d'un réseau multimodal

Un réseau de transport multimodal est caractérisé par la présence simultanée de différents modes de transport des personnes tels que les bus, les métros et les tramways. La liaison entre ces modes est traduite par la présence de pôles d'échange ou de noeuds de correspondance, au niveau desquels il y a des échanges de voyageurs entre deux ou plusieurs véhicules à différents horaires de la journée. Les noeuds de correspondances sont constitués d'arrêts situés à même lieu et liés par les horaires d'arrivées et de départs des véhicules. Ce genre de réseau devient indispensable dans les zones à haute densité de population qui nécessitent alors un plus grand nombre de déplacements.

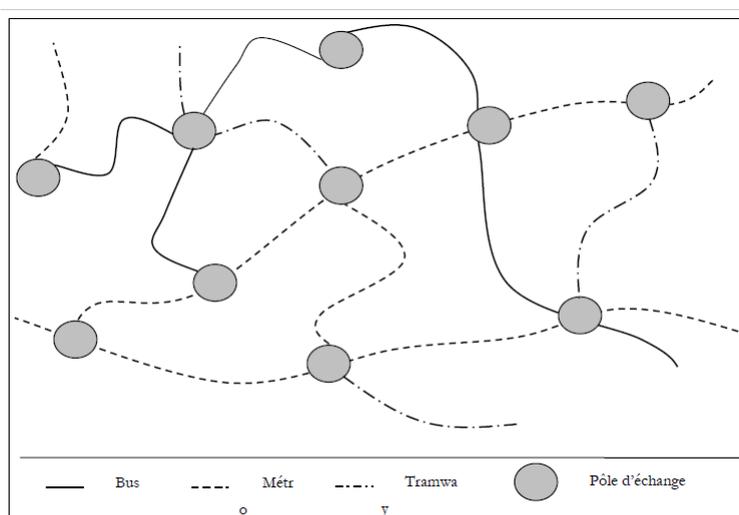


FIGURE 1.2: Exemple de réseau multimodal[ZID06]

Ces systèmes de transport peuvent être guidés ou non guidés. Le mode non guidé le plus répandu est l'autobus. Le choix des itinéraires des réseaux de transport par bus est nettement plus flexible et ces réseaux permettent de desservir des zones à faible densité avec un coût d'exploitation peu important.

Cependant, le principal inconvénient de ces réseaux réside dans leur dépendance des conditions de circulation, ce qui n'est pas le cas des systèmes de transport guidé tels que le métro. Par contre, ce dernier nécessite une importante infrastructure. Le tramway est aussi un mode de transport guidé mais qui dépend plus ou moins des conditions de circulation puisqu'il emprunte généralement les voies routières. Ainsi, l'exploitation

simultanée de ces différents modes résulte en un réseau fortement maillé et donc, plus difficile à gérer en hors-ligne et en ligne [ZID07].

1.2.3 Exploitation d'un réseau de transport urbain

Dans l'ensemble des réseaux de transport urbain, l'exploitation d'une ligne se décompose en deux phases distinctes :

- ✓ Une phase de conception et d'élaboration du programme de production concrétisée par le plan théorique appelé Tableau de Marche² (TM) de base. Ce TM est défini pour une période donnée tout au long d'une journée d'exploitation donnée.
- ✓ Une phase d'adaptation du programme de production (TM) aux conditions réelles d'exploitation concrétisée par le Tableau de Marche commandé qui est perpétuellement actualisé.

Les données issues de l'exploitation en temps réel (phase d'adaptation) viennent alimenter en permanence un tableau de marche (phase de conception) et sont ainsi susceptibles de se traduire par des modifications du tableau de marche de base.

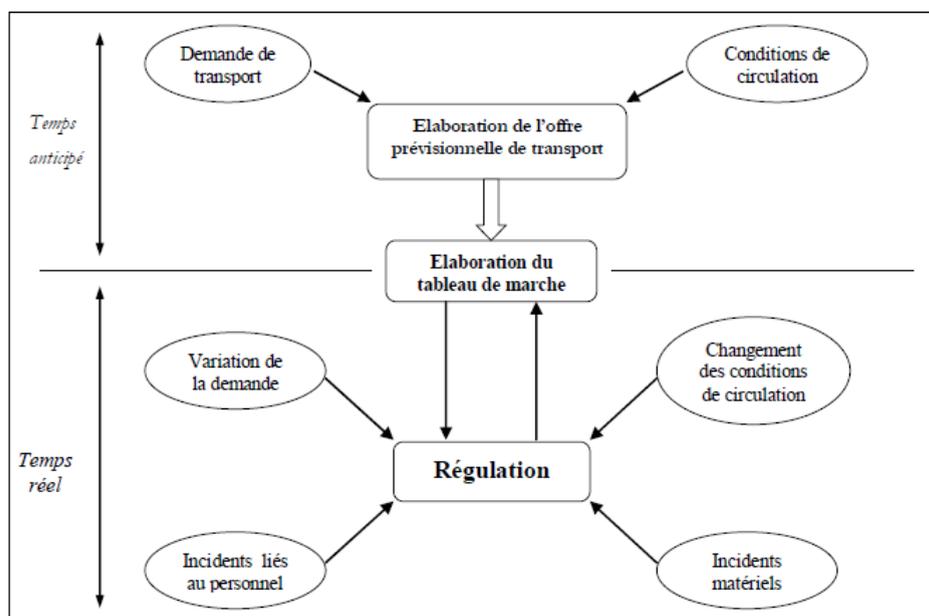


FIGURE 1.3: Exploitation des réseaux multimodals[ZID07]

2. Le tableau de marche (TM) est le résultat d'une optimisation de l'offre de service en fonction des objectifs et en respectant les contraintes d'exploitation.

Mais l'optimisation de l'offre de service étant réalisée en "temps anticipé", elle nécessite en particulier la formulation d'hypothèses sur les conditions de circulation et la demande de transport. Or, ces deux facteurs sont des phénomènes complexes, de nature aléatoire. C'est pourquoi le TM ne peut définir les modalités du fonctionnement optimal de la ligne que pour des conditions moyennes d'exploitation. Dès que l'on s'écarte de ces conditions moyennes, il devient nécessaire de réguler pour réagir à la dégradation de la qualité du service qui découle de cette dérive [LA02].

1.3 Problème de régulation

1.3.1 La recherche en transport pour aider à la régulation

Un réseau de transport multimodal est caractérisé, dans le cas des systèmes de transport des personnes et non des marchandises, par la présence simultanée de différents modes de transport tels que les bus, les métros et les tramways. La liaison entre ces modes est établie au niveau de pôles d'échange ou de noeuds de correspondance, où il y a des échanges de voyageurs entre deux ou plusieurs véhicules à différents horaires de la journée.

Dans un réseau multimodal, la qualité de l'offre est principalement perçue par les voyageurs lors des correspondances entre les différentes lignes d'un même mode de transport ou entre les différents modes.

Plusieurs questions liées à la nature des perturbations et aux manœuvres de régulation peuvent se poser lors de la gestion en temps réel du trafic d'un réseau de transport urbain. Comment et quand peut-on détecter une perturbation ? Quel est son impact sur le réseau ? Comment peut-on mesurer l'importance de l'incident ? Nécessite-t-il une régulation ? Si oui, existe-t-il des mesures qui lui sont adaptées ? Quelle mesure faut-il choisir ? Selon quel(s) critère(s) faut-il faire ce choix ? Et enfin, comment peut-on évaluer son impact pour pouvoir décider ? [ZID06]

1.3.2 Processus de planification

Le processus classique de planification est illustré par la Figure (1.4). La configuration initiale des lignes du réseau et leurs fréquences ainsi que les durées des trajets entre les différents points du réseau sont fixés comme paramètres d'entrée du processus, qui génère en premier lieu des tableaux d'horaires correspondant aux temps et lieux d'arrivée et de départ des véhicules, appelés courses. L'ensemble des courses à affecter à chaque véhicule représente une tâche d'ordonnancement statique pour générer des blocs de véhicules. Ensuite, l'ordonnancement du personnel fixe les fiches horaires journalières de service des conducteurs et des régulateurs, ainsi que les différentes relèves[LA02].

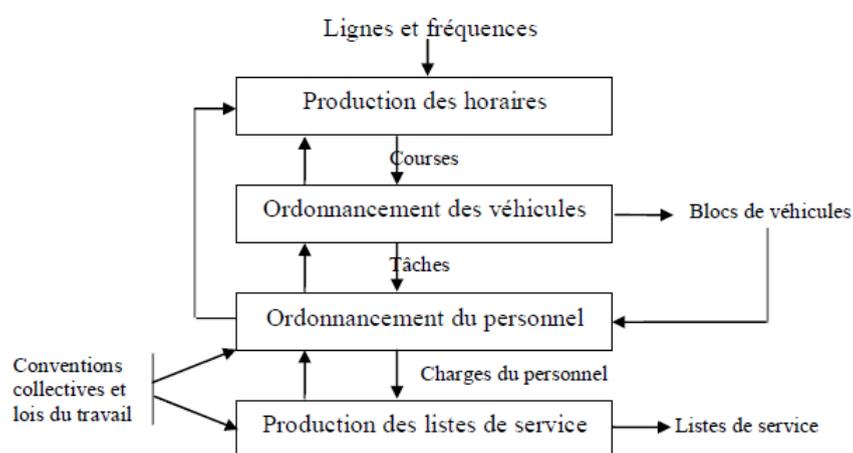


FIGURE 1.4: Processus classique de planification[ZID06]

1.3.3 Perturbation

Les écarts entre la réalité et les objectifs de la qualité de service résultent de différents types de perturbations qui peuvent affecter le fonctionnement correct de la ligne. Les perturbations pour véhicules (bus) peuvent être classées selon leurs sources. Le dysfonctionnement peut être dû à un problème interne ou externe. Les différentes perturbations sont[AMR12] :

- ✓ Des perturbations liées aux « **conditions de circulation** » et se répercutant sur les temps de parcours (retards)

- ✓ Des perturbations dues aux variations dans la configuration de la « **demande de transport** ».
- ✓ Des perturbations liées aux « **matériels roulants** » (voitures indisponibles ...).

1.3.4 Régulation du trafic

1.3.4.1 Définition

La régulation est un ensemble de méthodes permettant de se rapprocher au plus près du « **TM** » initial, malgré les perturbations, on se rend compte que la régulation est une batterie de mesures ou de méthodes (actions de régulation) qui essaient de remettre de l'ordre dans un réseau perturbé afin de se rapprocher de ce qui a été annoncé aux voyageurs. Le fonctionnement en temps réel d'une ligne est caractérisé par un système qui peut être schématisé selon la figure(1.5).

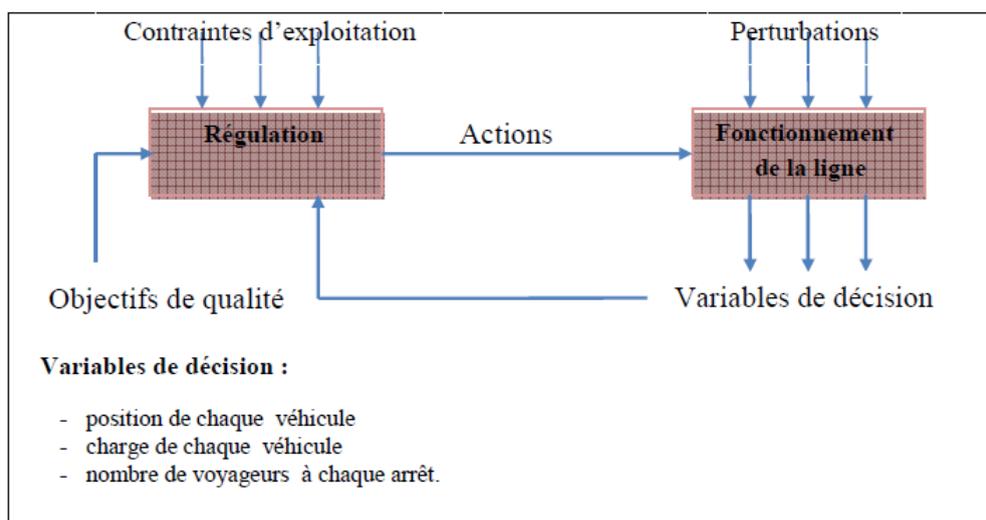


FIGURE 1.5: Fonctionnement d'une ligne en temps réel[AMR12]

Ce système se compose de deux éléments :

- ✓ Le sous système de fonctionnement de la ligne, dont l'état à un instant donné est décrit par un ensemble de variables (position et charge des voitures, voyageurs en attente aux arrêts,...).
- ✓ Le sous système de régulation qui analyse les écarts existants entre les objectifs de qualité de services et la réalité, et qui met en œuvre des actions susceptibles de

les comblé, compte tenu des contraintes liées à l'exploitation (niveau des moyens disponibles et règles d'utilisation de ces moyens)[AMR12].

1.3.4.2 Objectif de la régulation

L'objectif de régulation consiste à minimiser l'écart entre les valeurs mesurées et celles déterminées préalablement en mode de fonctionnement non perturbé et cela principalement en ce qui concerne la position et la vitesse.

Le processus de régulation contient différentes tâches allant de la détection de la perturbation jusqu'à la prise de décision, figure (1.4). Ces tâches sont regroupées en deux phases : phase de diagnostic et phase de décision. La phase de diagnostic consiste à détecter et analyser les perturbations afin d'évaluer leur gravité. Cette phase se base sur les données fournies par le Système d'Aide à l'Exploitation « SAE » qui représente un support de surveillance du réseau.

La phase de décision est entamée par le régulateur quand il doit choisir les mesures appropriées de régulation en tenant compte des contraintes d'exploitation, de la nature des perturbations, ainsi que des critères qui conviennent aux objectifs d'exploitation[AMR12].

1.4 Conclusion

Ce chapitre nous a permis de comprendre les notions liées au transport ainsi que la multimodalité. Ainsi nous pouvons comprendre leurs complexités en exploitation. La planification se fait pour optimiser l'utilisation des moyens existants dans le but de mieux servir les clients et assurer leur confort. Avec l'apparition des perturbations, les régulateurs doivent effectuer des régulations en temps réel ce qui les oblige à avoir recours à des systèmes d'aide à la décision. Dans le chapitre qui suit nous présentons les algorithmes à colonies de fourmis.

CHAPITRE 2

ALGORITHMES DE COLONIES DE FOURMIÉS

2.1 Introduction

Les colonies de fourmis dans la nature montrent une fascinante capacité à trouver des « **chemins** » entre des sources de nourriture et la fourmilière. A l'échelle d'une fourmie, le problème résolu est complexe tel que le plus court chemin. L'intelligence qui permet la résolution du problème est dite « collective » et est en fait le résultat émergent d'un grand nombre d'interactions élémentaires. Les fourmis résolvent des problèmes complexes par des mécanismes assez simples à modéliser. Il est ainsi assez simple de simuler leur comportement par des algorithmes informatiques.

2.2 Les métaheuristiques

Une métaheuristique est un algorithme d'optimisation visant à résoudre des problèmes d'optimisation difficile (souvent issus des domaines de la recherche opérationnelle, de l'ingénierie ou de l'intelligence artificielle) pour lesquels on ne connaît pas de méthode classique plus efficace.

Les métaheuristiques peuvent être classées de nombreuses façons. On peut distinguer celles qui travaillent avec une population de solutions de celles qui ne manipulent qu'une seule solution à la fois. Les méthodes qui tentent itérativement d'améliorer une solution sont appelées méthodes de recherche locale ou méthodes de trajectoire. Ces méthodes construisent une trajectoire dans l'espace des solutions en tentant de se diriger vers des solutions optimales.

Les exemples les plus connus de ces méthodes sont : La recherche Tabou et le Recuit Simulé. Les algorithmes génétiques, l'optimisation par essaim de particules et Les algorithmes de colonies de fourmis présentent les exemples les plus connus des méthodes qui travaillent avec une population.

Trois approches fondamentalement différentes qui ont déjà été appliquées avec succès ces dernières années sont présentées.

2.2.1 L'approche constructive

Les méthodes constructives produisent des solutions admissibles en partant d'une solution initiale vide et en insérant, à chaque étape, une composante dans la solution partielle courante. Cette décision n'est jamais remise en question par la suite[WID01].

2.2.2 L'approche de recherche locale

Les méthodes de recherche locale sont des algorithmes itératifs qui explorent l'espace X en se déplaçant pas à pas d'une solution à une autre. Une méthode de ce type débute à partir d'une solution $s_0 \in X$ choisie arbitrairement ou alors obtenue par le biais d'une méthode constructive[WID01].

2.2.3 L'approche évolutive

Les sciences de la vie et les processus naturels ont de tout temps fasciné les ingénieurs. Ces derniers n'hésitent pas à s'inspirer des structures et des mécanismes du monde vivant

pour développer des objets artificiels utilisables dans des contextes variés. Dans le domaine de l'optimisation combinatoire, la complexité des phénomènes naturels a servi de modèle pour des algorithmes toujours plus sophistiqués ces vingt-cinq dernières années. On peut citer les algorithmes génétiques, la méthode de recherche distribuée, les algorithmes de la fourmi qui fera l'objet de notre étude[WID01].

2.3 Comportements collectifs des insectes sociaux

2.3.1 Auto-organisation et comportement

2.3.1.1 Auto-organisation

L'auto-organisation est un processus dans lequel un modèle de niveau globale émerge uniquement d'un grand nombre d'interactions entre les composants de bas niveau du système. De plus, les règles spécifiant les interactions entre composants du système sont suivies en utilisant uniquement les informations locale, sans références au modèle global.

La question cruciale est donc de comprendre comment les composants d'un système interagissent entre eux pour produire un modèle complexe (au sens relatif du terme, i.e. plus complexe que les composants eux-mêmes). Un certain nombre de phénomènes nécessaires ont été identifiés : ce sont les processus de *rétroaction* et la gestion des *flux d'informations* [JD03].

La rétroaction (ou feed-back) qui est obtenue lorsque les résultats (obtenus par transformation des données d'entrée) sont retransmis au système sous la forme de nouvelles données d'entrée. Lorsque les nouvelles données agissent dans le même sens que l'action principale, c'est-à-dire amplifier la transformation, la rétroaction est dite positive. Dans ce cas, ces effets sont cumulatifs. Dans le cas contraire la rétroaction est dite négative, et le système doit être stable[BEN13].

2.3.1.2 Stigmergie

Elle est précisément définie comme une “forme de communication passant par le biais de modifications de l’environnement”, mais on peut rencontrer le terme “interactions sociales indirectes” pour décrire le même phénomène. Les biologistes différencient la stigmergie “quantitative” de celle “qualitative”, mais le processus en lui-même est identique. La grande force de la stigmergie est que les individus échangent des informations par le biais du travail en cours, de l’état d’avancement de la tâche globale à accomplir [BEN13].

2.3.1.3 Contrôle décentralisé

Dans un système auto-organisé, il n’y a pas de prise de décision à un niveau donné suivie d’ordres et d’actions prédéterminés. En effet, dans un système décentralisé chaque individu dispose d’une vision locale de son environnement, et ne connaît donc pas le problème dans son ensemble. Bien que, d’une manière générale, cette discipline tende à utiliser des modèles de comportements plus complexes, fondés notamment sur les sciences de la cognition. Les avantages d’un contrôle décentralisé sont notamment la robustesse et la flexibilité. Systèmes robustes, car capables de continuer à fonctionner en cas de panne d’une de leurs parties ; flexible car efficaces sur des problèmes dynamiques [JD03].

2.3.1.4 Hétérarchie dense

L’hétérarchie dense est un concept issu directement de la biologie utilisé pour décrire l’organisation des insectes sociaux, et plus particulièrement des colonies de fourmis. Le concept d’hétérarchie décrit un système où les propriétés des niveaux globaux agissent plus ou moins sur les propriétés des niveaux locaux, mais également un système où une activité dans les unités locales influence en retour les niveaux globaux. L’hétérarchie est dite dense dans le sens où un tel système forme un réseau hautement connecté, où chaque individu peut échanger des informations avec n’importe quel autre. Ce concept est en quelque sorte opposé à celui de hiérarchie où, dans une vision populaire mais erronée ;

la reine gouvernerait ses sujets en faisant passer des ordres dans une structure verticale, alors que, dans une hétéarchie, la structure est plutôt horizontale (figure.2.1).



FIGURE 2.1: Hiérarchie (a) et heterarchie dense (b) : deux concepts opposés [BEN13]

On constate que ce concept recoupe celui de contrôle décentralisé, mais aussi celui de stigmergie, en ce sens que l'hétéarchie décrit la manière dont le flux d'information parcourt le système. Cependant, dans une hétéarchie dense, tout type de communication doit être pris en compte, tant la stigmergie que les échanges directs entre individus[BEN13].

2.3.2 Optimisation naturelle : piste de phéromones

Les algorithmes de colonies de fourmis sont nés à la suite d'une constatation : les insectes sociaux en général, et les colonies de fourmis en particulier, résolvent naturellement des problèmes relativement complexes. Les biologistes ont étudié comment les fourmis arrivent à résoudre collectivement des problèmes trop complexes pour un seul individu, notamment les problèmes de choix lors de l'exploitation de sources de nourriture.

Les fourmis ont la particularité d'employer pour communiquer des substances volatiles appelées phéromones. Elles sont très sensibles à ces substances, qu'elles perçoivent grâce à des récepteurs situés dans leurs antennes. Ces substances sont nombreuses et varient selon les espèces. Les fourmis peuvent déposer des phéromones au sol, grâce à une glande située dans leurs abdomens, et former ainsi des pistes odorantes, qui pourront être suivies par leurs congénères. Les fourmis utilisent les pistes de phéromones pour marquer leur trajet, par exemple entre le nid et une source de nourriture. Une colonie est ainsi

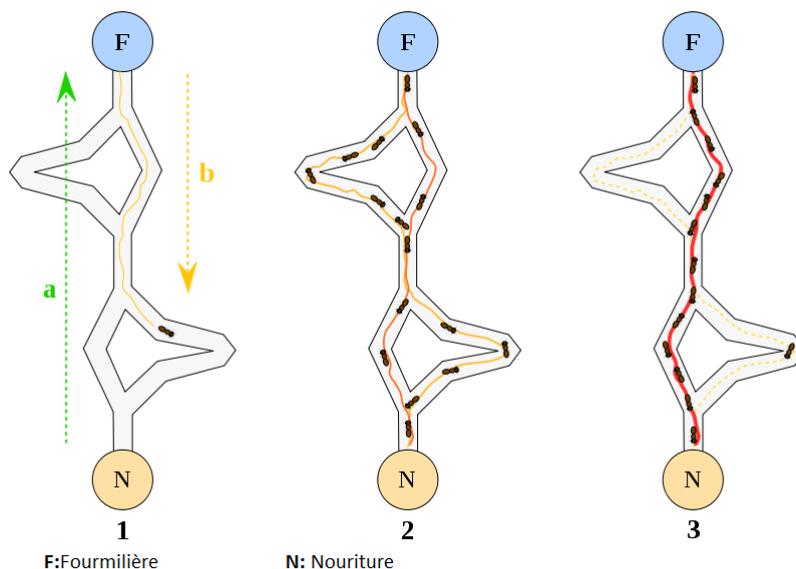


FIGURE 2.2: Expérience de sélection des branches les plus courtes par une colonie de fourmis [ZID07]

capable de choisir (sous certaines conditions) le plus court chemin vers la source à exploiter, sans que les individus aient une vision globale du trajet.

2.4 Les fourmis artificielles et les fourmis naturelles

L'étude sur le comportement des fourmis a donné une naissance à plusieurs nouvelles méthodes de résolution de problèmes. Certains mécanismes de résolution collective de problèmes chez les fourmis ont été transformés par des informaticiens en méthodes utiles pour l'optimisation et le contrôle qui peuvent s'appliquer aujourd'hui à tout un ensemble de problèmes scientifiques et techniques [BEN13].

2.4.1 Ressemblances [BEN13]

Colonie d'individus coopérants : Comme pour les fourmis réelles, une colonie virtuelle est un ensemble d'entités non-synchronisées, qui se rassemblent ensemble pour trouver une "bonne" solution au problème considéré. Chaque groupe d'individus doit pouvoir trouver une solution même si elle est mauvaise.

Pistes de phéromones : Ces entités communiquent par le mécanisme des pistes de phéromone. Cette forme de communication joue un grand rôle dans le comportement des fourmis : son rôle principal est de changer la manière dont l'environnement est perçu par les fourmis, en fonction de l'historique laissé par ces phéromones.

Évaporation des phéromones : La méta-heuristique ACO comprend aussi la possibilité d'évaporation des phéromones. Ce mécanisme permet d'oublier lentement ce qui s'est passé avant. C'est ainsi qu'elle peut diriger sa recherche vers de nouvelles directions, sans être trop contrainte par ses anciennes décisions.

Recherche du plus petit chemin : Les fourmis réelles et virtuelles partagent un but commun : recherche du plus court chemin reliant un point de départ (le nid) à des sites de destination (la nourriture).

Déplacements locaux : Les vraies fourmis ne sautent pas des cases, tout comme les fourmis virtuelles. Elles se contentent de se déplacer entre sites adjacents du terrain.

Choix aléatoire lors des transitions : Lorsqu'elles sont sur un site, les fourmis réelles et virtuelles doivent décider sur quel site adjacent se déplacer. Cette prise de décision se fait au hasard et dépend de l'information locale déposée sur le site courant. Elle doit tenir compte des pistes de phéromones, mais aussi du contexte de départ (ce qui revient à prendre en considération les données du problème d'optimisation combinatoire pour une fourmi virtuelle) [AMR12].

2.4.2 Différences [BEN13]

Les fourmis virtuelles possèdent certaines caractéristiques que ne possèdent pas les fourmis réelles :

Elles vivent dans un monde non-continu : Leurs déplacements consistent en des transitions d'état.

Mémoire (état interne) de la fourmi : Les fourmis réelles ont une mémoire très limitée. Tandis que nos fourmis virtuelles mémorisent l'historique de leurs actions. Elles peuvent aussi retenir des données supplémentaires sur leurs performances.

Nature des phéromones déposées : Les fourmis réelles déposent une information physique sur la piste qu'elles parcourent, là où les fourmis virtuelles modifier des informations dans les variables d'états associées au problème. Ainsi, l'évaporation des phéromones est une simple décrémentation de la valeur des variables d'états à chaque itération.

Qualité de la solution : Les fourmis virtuelles déposent une quantité de phéromone proportionnelle à la qualité de la solution qu'elles ont découverte.

Retard dans le dépôt de phéromone : Les fourmis virtuelles peuvent mettre à jour les pistes de phéromones de façon non immédiate : souvent elles attendent d'avoir terminé la construction de leur solution. Ce choix dépend du problème considéré bien évidemment.

Capacités supplémentaires : Les fourmis virtuelles peuvent être pourvues de capacités artificielles afin d'améliorer les performances du système. Ces possibilités sont liées au problème et peuvent être :

1. *l'anticipation* : la fourmi étudie les états suivants pour faire son choix et non seulement l'état local.
2. *le retour en arrière* : une fourmi peut revenir à un état déjà parcouru car la décision qu'elle avait prise à cet état a été mauvaise.[AMR12]

2.4.3 Avantage et inconvénients des fourmis artificielles

2.4.3.1 Avantages

- ✓ Très grande adaptabilité.
- ✓ Parfait pour les problèmes basés sur des graphes.

2.4.3.2 Inconvénients

- ✓ Un état bloquant peut arriver.
- ✓ Temps d'exécution parfois long.
- ✓ Ne s'applique pas à tous type de problèmes [BEN13].

2.5 Application des algorithmes de colonies de fourmis

Application aux problèmes d'optimisation : Les algorithmes d'optimisation sont regroupés sous le terme général métaheuristique « optimisation par les colonies de fourmis (OCF) » ou « *Ant Colony Optimization (ACO)* ». D'une manière générale, l'utilisation de OCF nécessite de choisir une représentation formelle du problème d'optimisation à traiter et de définir le processus de construction de solutions par les fourmis en utilisant cette représentation [BEN13].

Application au tri d'objets : Deneubourg apparaît comme le pionnier dans le domaine du tri automatique d'objets en utilisant des fourmis artificielles [BEN13].

Application à la classification : Le pas qui sépare le tri d'objets de la classification a ensuite été franchi dans l'article de Monmarché. Il existe plusieurs algorithmes de classification automatique basée sur les fourmis artificielles. On peut aussi introduire un modèle à base de fourmis pour la classification utilisant le système d'identification chimique des fourmis[BEN13].

Application en traitement d'images : Des études plus récentes ont étendu les algorithmes de colonies de fourmis aux problèmes de la classification pour les appliquer au domaine du traitement d'images [BEN13].

2.6 Algorithme de colonies de fourmis(ACF) pour l'optimisation

Le problème du voyageur de commerce¹ a fait l'objet de la première implementation d'un algorithme de colonies de fourmis : le "Ant System". Le passage de la métaphore à l'algorithme est ici relativement facile à faire et le problème du voyageur de commerce est bien connu et étudié. Il est intéressant d'approfondir le principe de ce premier algorithme pour bien comprendre le mode de fonctionnement des algorithmes de colonies de fourmis.

Il y a deux façons d'aborder ces algorithmes. La première, la plus évidente au premier abord, est celle qui a historiquement mené au "Ant System" original. La seconde est une description plus formelle des mécanismes communs aux algorithmes de colonies de fourmis[JD03].

Plusieurs versions de l'algorithme à colonie de fourmis qui ont été appliquées au PVC(problème du voyageur de commerce), nous allons expliquer quelques uns.

2.6.1 Variantes

2.6.1.1 Algorithme de base(AS)

Dans l'algorithme AS(Ant System), à chaque itération $t(1 \leq t \leq t_{max})$, chaque fourmi k ($k = 1, \dots, m$) parcourt le graphe et construit un trajet complet de $n = |N|$ étapes(on note $|N|$ le cardinal de l'ensemble N). Pour chaque fourmi, le trajet entre une ville i et une ville j dépend de :

1. la liste des villes déjà visitées, qui définit les mouvements possibles à chaque pas, quand la fourmi k est sur la ville i : J_i^k .
2. l'inverse de la distance entre les villes : $\eta_{ij} = \frac{1}{d_{ij}}$, appelé *visibilité*. Cette information statique est utilisée pour diriger le choix des fourmis vers des villes proches, et éviter les villes trop lointaines.

1. Dans ce problème, un voyageur de commerce doit visiter plusieurs villes (ou clients) en passant une et une seule fois par chacune d'entre elles, et en minimisant la distance totale parcourue[SBI11]

3. la quantité de phéromone déposée sur l'arête reliant les deux villes, appelée *l'intensité de la piste*. Ce paramètre définit l'attractivité d'une partie du trajet global et change à chaque passage d'une fourmi. C'est en quelque sorte une mémoire globale du système, qui évolue par apprentissage.

La règle de déplacement est la suivante :

$$p_{ij}^k = \begin{cases} \frac{(\tau_{ij}(t))^\alpha \cdot (\eta_{ij})^\beta}{\sum_{i \in J_i^k} (\tau_{ij}(t))^\alpha \cdot (\eta_{ij})^\beta} & \text{si } j \in J_i^k \\ 0 & \text{si } j \notin J_i^k \end{cases} \quad (1)$$

Où α et β sont deux paramètres contrôlant l'importance relative de l'*intensité* de la piste, $\tau_{ij}(t)$, et la *visibilité*, η_{ij} . Avec $\alpha = 0$, seul la visibilité de la ville est prise en compte ; la ville la plus proche est donc choisie à chaque pas. Au contraire, avec $\beta = 0$, seules les pistes de phéromones jouent. Pour éviter une sélection trop rapide d'un trajet, un compromis entre ces deux paramètres, jouant sur les comportements de *diversification* et d'*intensification*, est nécessaire. Après un tour complet, chaque fourmi laisse une certaine quantité de phéromones $\Delta\tau_{ij}^k(t)$ sur l'ensemble de son parcours, quantité qui dépend de la qualité de la solution trouvée :

$$\Delta\tau_{ij}^k(t) = \begin{cases} \frac{Q}{L^k(t)} & \text{si } (i, j) \in T^k(t) \\ 0 & \text{si } (i, j) \notin T^k(t) \end{cases} \quad (2)$$

Où $T^k(t)$ est le trajet effectué par la fourmi k à l'itération t , $L^k(t)$ la longueur du tour et Q un paramètre fixé.

L'algorithme ne serait pas complet sans le processus d'*évaporation* des pistes de phéromones. En effet, pour éviter d'être piégé dans des solutions sous-optimales, il est nécessaire de permettre au système d'oublier les mauvaises solutions. On contrebalance donc l'additivité des pistes par décroissance constante des valeurs des arêtes à chaque itération. La règle de mise à jour des pistes est donc :

$$\tau_{ij}(t+1) = (1 - \rho) \cdot \tau_{ij}(t) + \Delta\tau_{ij}(t) \quad (3)$$

Où $\Delta\tau_{ij}(t) = \sum_{k=1}^m \Delta\tau_{ij}^k(t)$ et m est le nombre de fourmis. La quantité initiale de phéromone sur les arêtes est une distribution uniforme d'une petite quantité $\tau_0 \geq 0$

```

Pour  $t = 1, \dots, t_{max}$ 
  Pour chaque fourmi  $k = 1, \dots, m$ 
    Choisir une ville au hasard
    Pour chaque ville non visitée  $i$ 
      Choisir une ville  $j$ , dans la liste  $J_i^k$  des villes restantes, selon la formule (1)
    Fin Pour
    Déposer une piste  $\Delta\tau_{ij}^k(t)$  sur le trajet  $T^k(t)$  conformément à l'équation (2)
  Fin Pour
  Évaporer les pistes selon la formule (3)
Fin Pour

```

Algorithme de colonies de fourmis de base(AS) [JD03]

2.6.1.2 Ant System & élitisme

Une première variante du "Système de fourmis" a été proposée dans l'introduction de fourmis "élitistes". Dans cette version, la meilleure fourmi(elle qui a effectué le trajet le plus court) dépose une quantité de phéromone plus grande, dans l'optique d'accroître la probabilité des autres fourmis d'explorer la solution la plus prometteuse[JD03].

2.6.1.3 ANT_Q

Dans cette variante de AS, la règle de mise à jour locale est inspirée du "*Q-learning*". Cependant, aucune amélioration par rapport à l'algorithme AS n'a pu être démontrée. Cette algorithme n'est d'ailleurs, de l'aveu même des auteurs, qu'une pré-version du "*Ant Colony System*" [JD03].

2.6.1.4 ANT COLONY SYSTEM (ACS)

Pour améliorer les performances de AS (Ant System) les auteurs de Dorigo & al ont proposé le Ant Colony System (ACS) surtout pour les problèmes de grandes tailles. Les modifications par rapport à AS sont :

1. ACS introduit une règle de transition dépendant d'un paramètre q_0 ($0 \leq q_0 \leq 1$), qui définit une balance *diversification/intensification*. Une fourmi k sur une ville i choisira une ville j par la règle :

$$j = \begin{cases} \operatorname{argmax}_{u \in J_i^k} [(\tau_{iu}(t)) \cdot (\eta_{iu})^\beta] & \text{si } q \leq q_0 \\ J & \text{si } q \geq q_0 \end{cases} \quad (4)$$

où q est une variable aléatoire uniformément distribuée sur $[0, 1]$ et $J \in J_i^k$ une ville sélectionnée aléatoirement selon la probabilité :

$$p_{iJ}^k(t) = \frac{(\tau_{iJ}(t)) \cdot (\eta_{iJ})^\beta}{\sum_{l \in J_i^k} (\tau_{il}(t)) \cdot (\eta_{il})^\beta} \quad (5)$$

En fonction du paramètre q_0 , il y a donc deux comportements possibles : si $q > q_0$ le choix se fait de la même façon que l'algorithme AS, et le système tend à effectuer une diversification ; si $q \leq q_0$, le système tend au contraire vers une intensification. En effet, pour $q \leq q_0$, l'algorithme exploite plus d'information récoltée par le système, il ne peut pas choisir de trajet non exploré.

2. La gestion des pistes est séparée en deux niveaux : une mise à jour locale et une mise à jour globale. Chaque fourmi dépose une piste lors de la mise à jour locale :

$$\tau_{ij}(t+1) = (1 - \rho) \cdot \tau_{ij}(t) + \rho \cdot \tau_0 \quad (6)$$

où τ_0 est la valeur initiale de la piste. À chaque passage, les arêtes visitées voient leur quantité de phéromone diminuer, ce qui favorise la diversification par la prise en compte des trajets non explorés. À chaque itération, la mise à jour globale s'effectue comme ceci :

$$\tau_{ij}(t+1) = (1 - \rho) \cdot \tau_{ij}(t) + \rho \cdot \Delta \tau_{ij}(t) \quad (7)$$

où les arêtes (i, j) appartiennent au meilleur tour T^+ de longueur L^+ et où $\Delta\tau_{ij}(t) = \frac{1}{L^+}$. Ici, seul la meilleur piste est donc mise à jour, ce qui participe à une intensification par sélection de la meilleur solution.

3. le système utilise une liste de candidats. Cette liste stock pour chaque ville les v plus proches voisins, classées par distances croissantes. Une fourmi ne prendra en compte une arête vers une ville en dehors de la liste que si celle-ci a déjà été explorée. Concrètement, si toutes les arêtes ont déjà été visitées dans la liste de candidats, le choix se fera en fonction de la règle de probabilité précédente, sinon c'est la plus proche des villes non visitées qui sera choisie [JD03].

2.6.1.5 ACS & 3-opt

cette variante est une hybridation entre le ACS et une recherche locale de type 3-opt[Dorigo et al. 97b]. Ici, la recherche est lancée pour améliorer les solutions trouvées par les fourmis (et donc les ramener à l'optimum local le plus proche) [JD03].

2.6.1.6 MIN-MAX ANT SYSTEM (MMAS)

Cette variante est fondée sur algorithme AS et présente quelques différences notables :

1. Seul la meilleure fourmi met à jour une piste de phéromone ;
2. Les valeurs des pistes sont bornées par τ_{min} et τ_{max} ;
3. Les pistes sont initialisées à la valeur maximum τ_{max} ;
4. La mise à jour des pistes se fait de façon proportionnelle, les pistes les plus fortes étant moins renforcées que les plus faibles ;
5. Une ré-initialisation des pistes peut être effectuée.

Les meilleurs résultats sont obtenus en mettant à jour la meilleure solution avec une fréquence de plus en plus forte au cours de l'exécution[JD03].

2.6.2 Paramètres d'un algorithme ACO

Les principaux paramètres d'un algorithme ACO sont les suivant :

- Le nombre de fourmis $nbAnts$, qui détermine le nombre de combinaisons construites à chaque cycle avant chaque étape de mise à jour de phéromone.
- α , qui détermine le poids du facteur phéromonal dans la probabilité de transition.
- β , qui détermine le poids du facteur heuristique dans la probabilité de transition.
- ρ , qui détermine la vitesse d'évaporation des traces phéromonales.

La valeur du paramètre β change très clairement d'une application à l'autre, en fonction notamment de la fiabilité du facteur heuristique. Les valeurs des paramètres $nbAnts$, α et ρ sont moins dépendantes de l'application considérée mais plus de temps dont on dispose pour la résolution du problème [SOL08].

2.6.3 Intensification et diversification

Quand on explore un espace de recherche, avec ACO comme avec d'autres méta-heuristiques, il s'agit de trouver un bon compromis entre, d'une part, intensifier la recherche aux alentours des zones les plus prometteuses, contenant les meilleures combinaisons trouvées, et d'autre part, diversifier la recherche afin de découvrir de nouvelles zones de l'espace de recherche dont on espère qu'elles contiendront de meilleures combinaisons.

L'intensification d'une recherche ACO est faite par le biais de la phéromone. En déposant de la phéromone sur les composants des meilleures combinaisons trouvées, on augmente la probabilité de choisir ces composants lors de la construction de nouvelles combinaisons, et on intensifie donc l'effort de recherche aux alentours de ces composants.

Si l'on intensifie trop la recherche, en récompensant trop fortement les composants des meilleures combinaisons et/ou en donnant trop de poids au facteur phéromonal dans la stratégie de choix d'un composant, on risque d'observer un phénomène de stagnation prématurée, où les fourmis ont convergé autour d'une combinaison suboptimale et ne sont plus capables d'explorer d'autres zones de l'espace de recherche. Afin d'éviter ces phénomènes de stagnation, il est nécessaire de contrebalancer l'intensification avec des mécanismes de diversification.

Dans un algorithme ACO, la diversification de la recherche est assurée par la stratégie de choix probabiliste des composants, sous réserve que les probabilités de choix ne puissent devenir nulles. Ces probabilités étant définies proportionnellement aux produits de facteurs phéromonaux et heuristiques, il s'agit de s'assurer que ces facteurs ne peuvent devenir nuls. On fixe généralement pour cela une borne minimale $\tau_{min} > 0$ pour les traces phéromonales [SOL08].

2.6.4 Représentation du problème

Le problème est représenté par un jeu de solutions, une fonction objectif assignant une valeur à chaque solution et un jeu de contraintes. L'objectif est de trouver l'optimum global² satisfaisant les contraintes. Les différents états du problème sont caractérisés comme une séquence de composants. On peut noter que, dans certains cas, un coût peut être associé à des états autres que solutions.

Dans cette représentation, les fourmis construisent des solutions en se déplaçant sur un graphe $G = (C, L)$, où les nœuds sont des composants de C et où l'ensemble L connecte les composants de C . les contraintes du problème sont implémentées directement dans les règles de déplacement des fourmis.

2.6.5 Comportement des fourmis

Les fourmis peuvent être caractérisées comme une procédure de construction stochastique construisant des solutions sur le graphe $G = (C, L)$. En générale, les fourmis tentent d'élaborer des solutions faisables, mais si nécessaire, elles peuvent produire des solutions infaisables. Les composants et les connexions peuvent être associés à des pistes de phéromones τ (mettant en place une mémoire adaptative décrivant l'état du système) et à une valeur heuristique η (représentant une information a priori sur le problème, ou venant d'une source autre que celle des fourmis ; c'est bien souvent le coût de l'état en

2. C'est la meilleur solution réalisable avec une valeur objective aussi bonne ou meilleur que toutes les autres solutions.

cours). Les pistes de phéromones et la valeur de l'heuristique peuvent être associées soit aux composants, soit aux connexions.

Chaque fourmi dispose d'une mémoire utilisée pour stocker le trajet effectué, d'un état initial et de conditions d'arrêt. Les fourmis se déplacent d'après une règle de décision probabiliste fonction des pistes de phéromone locales, de l'état de la fourmi et des contraintes du problème. Lors de l'ajout d'un composant ou à la connexion correspondante. Une fois la solution construite, elles peuvent mettre à jour la piste de phéromone des composants ou des connexions utilisées. Enfin, une fourmi dispose au minimum de la capacité de construire une solution au problème.

2.7 Conclusion

Dans ce chapitre, nous avons présenté l'algorithme de colonies de fourmis, proposé par Dorigo en 1992, appliqué au problème de voyageur de commerce. Depuis ce premier algorithme, de nombreuses extensions ont été proposées. Ces algorithmes ont été appliqués à un grand nombre de problèmes différents. Dans le chapitre qui suit nous présentons une de ces variantes.

CHAPITRE 3

ALGORITHME DE COLONIES DE FOURMIS POUR LA RECONFIGURATION SPACIALE

3.1 Introduction

La complexité des réseaux de transport ne cesse d'augmenter, les nombres de véhicules ainsi que des passagers sont de plus en plus importants. A cette complexité s'ajoute le caractère de multimodalité des réseaux de transport. Ainsi la tâche de régulateur devient de plus en plus dure et impossible dans certains cas très complexes

Après avoir présenter les algorithmes de colonies de fourmis, nous présenterons l'approche ACFRS(algorithme de colonies de fourmis pour la reconfiguration spatiale) fait par Mr Saleh Zidi [ZID07].

3.2 ACF pour la reconfiguration spatiale

Un réseau de transport peut être représenté par un graphe dont les nœuds sont les stations et les arcs sont les itinéraires suivis par les véhicules. Plusieurs problèmes entraînent une perturbation de circulation tels que le blocage d'un itinéraire ou plusieurs,

un nombre important de passagers dans une zone. Le régulateur doit trouver une solution afin d'absorber le flux de passagers et de desservir rapidement le plus grand nombre d'arrêts possibles.

Dans certains cas, des perturbations simultanées rendent la tâche du régulateur complexe ou impossible, d'où l'importance d'un système informatique pour aider à la reconfiguration, qui permet de déterminer en temps réel les chemins optimaux en temps de parcours, distance de parcours et nombre de passagers servis.

La fonction de coût de l'arc i concerné par la reconfiguration est :

$$Cout_f = a \times \left(\frac{T_i}{C_i}\right) + b \times \left(\frac{D_i}{C_i}\right) \tag{1}$$

Tel que : C_i est le nombre de passagers à servir sur l'arc i

$$C_i = Min((Cap_i + Cd_i), Cm_i) + Cd_i \tag{2}$$

Cap_i : La capacité du véhicule concerné par la reconfiguration sur l'arc i .

Cd_i : Le nombre de passagers qui vont descendre à la station d'arrivée A_i .

Cm_i : Le nombre de passagers en attente dans la station d'arrivée A_i de l'arc i .

et T_i le temps de parcours, D_i la distance de parcours, a et b les poids des critères.

Un graphe de transport concerné par la reconfiguration sera présenté par un tableau dont chaque colonne est un vecteur des données d'un arc, donnant la station de départ, la station d'arrivée, le coût et la quantité de phéromone sur cet arc.

Notation :

N : Le nombre de fourmis.

$\tau_i(t)$: La dernière quantité de phéromone déposée sur l'arc i avant la mise à jour.

ρ : Le taux de persistance des traces de phéromone ($0 < \rho < 1$).

Ω_s : L'ensemble des arcs qui ont comme station de départ la station s .

P_1 : La probabilité de la transition aveugle.

P_2 : La probabilité de la transition intelligente.

$Tabou_f$: la liste taboue de la fourmi f , elle contient l'ensemble des stations visitées par f .

P : La probabilité de choix de type de la transition (aveugle ou intelligente).

$Cout^f(t)$: Le coût total d'un chemin emprunté par la fourmi f $Cout^f(t) = \sum_i Cout_i$.

I_{max} : Nombre de répétition de la meilleur solution.

I_{1max} : Nombre d'itérations maximales pour trouver la première solution

I_{2max} : Nombre d'itérations maximales pour trouver la deuxième solution.

Dans ce qui suit, nous allons décrire brièvement les étapes de l'algorithme de colonies de fourmis pour la reconfiguration spatiale.

- ✓ Initialisation des traces de phéromones sur les arcs du graphe (la quantité est identique sur tous les arcs).
- ✓ Chaque fourmi se déplace d'une station à l'autre jusqu'à l'arrivée à la station finale déterminé par le régulateur.
- ✓ La fourmi mémorise les stations visitées dans une liste taboue afin qu'elle ne passe pas par la même station deux fois.
- ✓ Le choix d'un arc de transition se fait aléatoirement ; la fourmi choisit avec une probabilité P une première transition aléatoire et avec une probabilité $(1 - P)$ une seconde méthode plus intelligente (elle dépend de la quantité de phéromone sur cet arc et de ses caractéristiques heuristiques : le temps et la distance de parcours à minimiser et la charge à maximiser) :

$$P_1 = \begin{cases} \frac{1}{k} & \text{si } i \in \Omega_s \text{ et } i \notin tabou_f \\ 0 & \text{sinon} \end{cases} \quad (3)$$

$$P_2 = \begin{cases} \frac{[\tau_i(t)]^\alpha [\frac{1}{D_i}]^\beta [\frac{1}{D_i}]^\gamma [C_i]^\delta}{\sum_{j=1}^k [\tau_j(t)]^\alpha [\frac{1}{D_j}]^\beta [\frac{1}{D_j}]^\gamma [C_j]^\delta} & \text{si } i \in \Omega_s \text{ et } i \notin tabou_f \\ 0 & \text{sinon} \end{cases} \quad (4)$$

avec α , β , γ et δ les poids respectivement de : phéromone, temps de parcours, distance de parcours et la charge.

- ✓ Une fourmi peut emprunter un chemin qui ne mène pas à la station d'arrivée, nous ne considérons pas la solution qu'elle a construite.

- ✓ Lorsque toutes les fourmis ont terminé, une évaluation des solutions se fait ainsi que la mise à jour des phéromones d'une manière à ce que nous repectons l'évaporation de la phéromone :

$$\tau_i(t + 1) = \rho \cdot \tau_i(t) + \sum_{f=1}^n \Delta\tau_i^f(t) \tag{5}$$

ρ est le facteur de persistance.

$\Delta\tau_i^f(t)$ est la quantité de phéromone ajoutée par la fourmi f sur l'arc i à l'itération t .

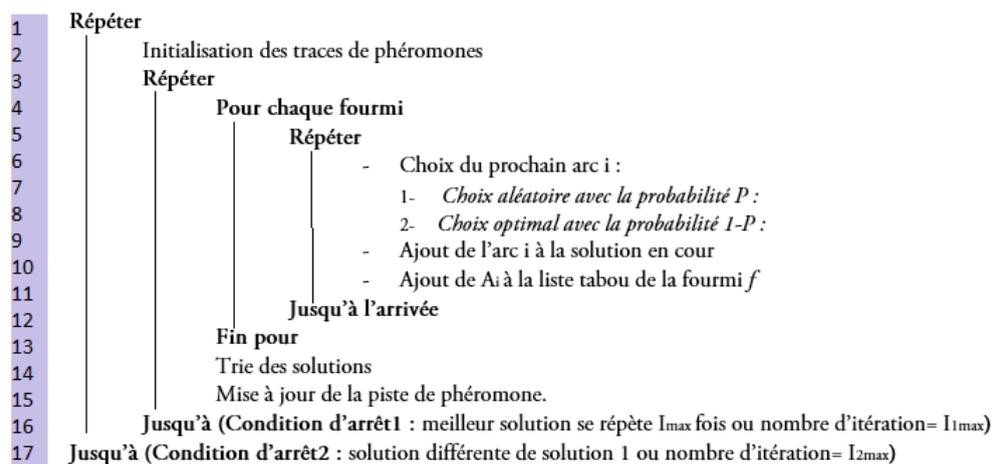
La quantité de phéromone ajoutée est inversement proportionnelle au coût du dernier chemin (solution) trouvée par la fourmi $f(Cout^f(t))$ afin de renforcer les chemins les plus courts plus que les autres.

$$\Delta\tau_i^f(t) = \begin{cases} \frac{1}{Cout^f(t)} & \text{si } i \in tabou_f \\ 0 & \text{si non} \end{cases} \tag{6}$$

- ✓ Lors de la recherche d'une deuxième solution différente de la première :
 - Mettre à zéro la quantité de phéromone.
 - Vider les listes taboues.
 - Relancer les instructions de recherche en diminuant la quantité de phéromone sur le premier chemin emprunté à chaque fois qu'une fourmi l'emprunte selon la formule suivante :

$$\tau_i(t + 1) = \varepsilon \cdot \tau_i(t); \tag{7}$$

La figure (3.1) montre les étapes de déroulement de l'algorithme.



ACF pour la reconfiguration spatiale

3.3 Implémentation et Tests

3.3.1 Outils informatiques utilisés

Le langage Java est un langage de programmation informatique orienté objet créé par James Gosling et Patrick Naughton, employés de Sun Microsystems, avec le soutien de Bill Joy (cofondateur de Sun Microsystems en 1982), présenté officiellement le 23 mai 1995 au SunWorld.

La société Sun a été ensuite rachetée en 2009 par la société Oracle qui détient et maintient désormais Java.

La particularité et l'objectif central de Java est que les logiciels écrits dans ce langage doivent être très facilement portables sur plusieurs systèmes d'exploitation tels que UNIX, Windows, Mac OS ou GNU/Linux, avec peu ou pas de modifications. Pour cela, divers plateformes et frameworks associés visent à guider, sinon garantir, cette portabilité des applications développées en Java.

Le langage Java reprend en grande partie la syntaxe du langage C++, très utilisé par les informaticiens. Néanmoins, Java a été épuré des concepts les plus subtils du C++ et à la fois les plus déroutants, tels que les pointeurs et références, ou l'héritage multiple contourné par l'implémentation des interfaces. Les concepteurs ont privilégié l'approche

orientée objet de sorte qu'en Java, tout est objet à l'exception des types primitifs (nombres entiers, nombres à virgule flottante, etc.).

3.3.2 Codage des données

Un graphe de transport est représenté par sa matrice d'adjacence.

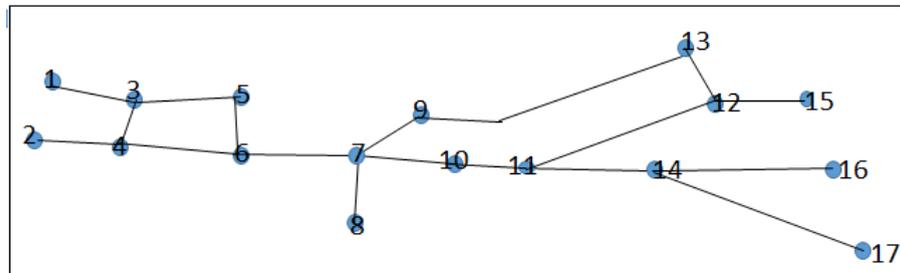


FIGURE 3.1: Un exemple d'un réseau de transport

$$\left\{ \begin{array}{cccccccccccccccc} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{array} \right\}$$

FIGURE 3.2: La matrice d'adjacence du réseau précédent

3.3.3 Codages des solutions

La solution du problème est représenté dans un tableau tel que les éléments de ce tableau sont les stations de la solutions ordonnée à partir de la station de départ jusqu'à la station d'arrivé (la station de départ est 1, la station d'arrivé est 6).

1	2	3	5	6
---	---	---	---	---

FIGURE 3.3: Représentation d'une solution

3.3.4 Testes

Après l'implémentation de la méthode précédente sous le langage de programmation java, nous avons effectué les testes sur une machine Intel(R) Core(TM) i5-5200U CPU @ 2.20 GHz de 4 GO de RAM. Les données d'entrés sont :

- La matrice de temps de parcours, La matrice des distances de parcours et la matrice de capacité qui sont lus à partir d'un fichier.
- La station de départ et la station d'arrivé : introduites par le clavier.

A la sortie nous aurons : Le chemin optimal. Les testes sont effectué sur deux benchmarks à différentes densité et nombre de stations. Le premier contient 17 stations avec 18 arcs, et le deuxième a 30 station avec 133 arcs. Nous utiliserons les données suivantes :

$$Cap = 40, Cm = 20, Cd = 20$$

$$\alpha = 1, \beta = 1, \gamma = 1, \sigma = 1 \text{ et } \rho = 0.9. I_{max} = 10, I_1max = 1000, I_2max = 10.$$

Pour le premier banchmark, nous cherchons le chemin optimal entre la station 12 et la station 8. Pour le deuxième benchmark, nous cherchons le chemin optimal entre la station 1 et la station 25.

3.3.4.1 Nombre de fourmis

Benchmark 1							
Nbr de fourmi	1	2	4	8	12	16	18
Nbr d'itération	31	28	19	13	12	10	10
Benchmark 2							
Nbr de fourmi	1	15	30	45	60	75	100
Nbr d'itération	107	14	11,7	10,5	10,3	10,1	10

TABLE 3.1: influence de nombre de fourmi sur la convergence

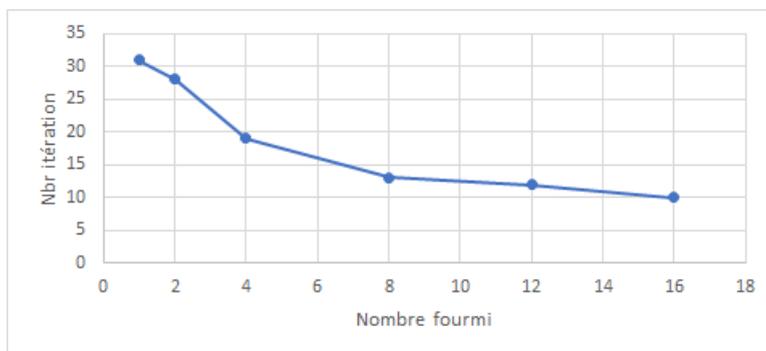


FIGURE 3.4: L'influence de nombre de fourmi sur la convergence Benchmark 1

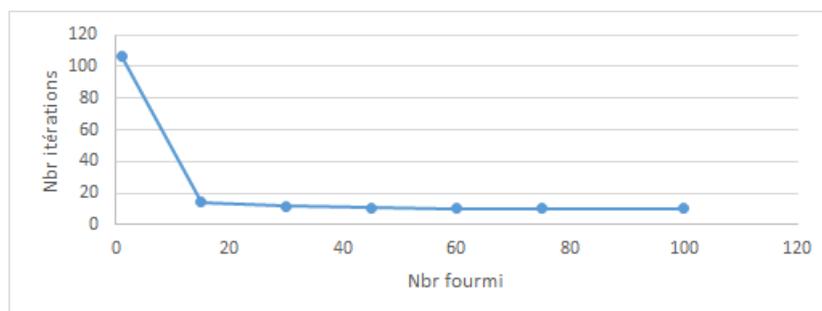


FIGURE 3.5: L'influence de nombre de fourmi sur la convergence Benchmark 2

Remarque 3.1. On remarque que le nombre d'itération diminue au fur et à mesure que le nombre de fourmi augmente (la convergence est atteinte rapidement lorsque le nombre de fourmi ≥ 16 pour le premier benchmark et ≥ 100 pour le deuxième benchmark).

Conclusion : Pour que la convergence soit atteinte plus rapidement, le nombre de fourmi doit être proche de nombre d'arc du réseau. Les auteurs de ACF suggèrent d'utiliser autant de fourmis que le nombre de nœuds.

3.3.4.2 Probabilité de choix de transition

Comme expliqué précédemment dans la section (2.6.3), le choix d'une probabilité de transition entre la probabilité aveugle et intelligente détermine le rapport exploration¹/exploitation². On peut remarquer que cela a une influence sur les pistes de phéromone

1. La capacité d'un algorithme à explorer le domaine des variables pour rechercher la meilleure vallée, c'est à dire celle qui contient l'optimum global.

2. La capacité d'un algorithme à converger rapidement vers le minimum d'une vallée donnée à partir d'un point de départ.

c'est à dire si on favorise l'exploitation, les pistes de phéromone les plus renforcées sont celle issues des solutions optimales locales.

P	0	0.2	0.4	0.6	0.8	1
Phéromone issue de solution locale	38,71	111,85	163,77	273,34	302,98	360,29
Phéromone issue de solution globale	283,27	213,41	171,95	123,94	118,87	81,88

TABLE 3.2: influence de la probabilité de transition sur les pistes de phéromone

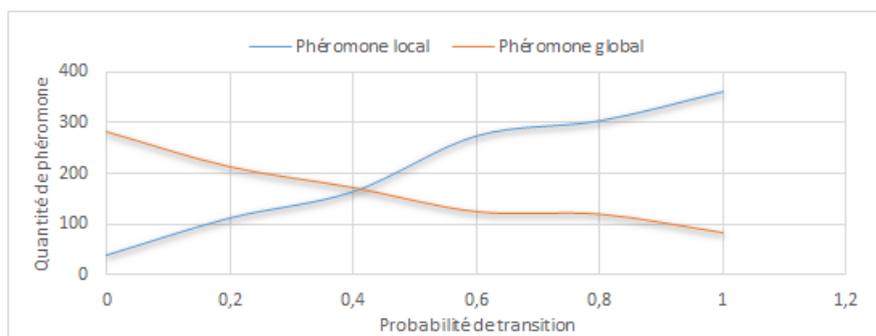


FIGURE 3.6: influence de la probabilité de transition sur les pistes de phéromone

Remarque 3.2. On remarque que la courbe des pistes de phéromone issue des solution local diminue au fur et à mesure que la probabilité de transition augmente, par contre la courbe des pistes de phéromone issue des solution globale augmente au fur et à mesure que la probabilité de transition augmente.

Pour conclure, il faut utiliser un rapport exploration/exploitation adéquat afin d'arriver à un résultat plus proche de la réalité.

Le tableau suivant montre les résultats du test effectué sur l'influence de P :

P	0.0001	0.001	0.01	0.1	0.2	0.5	1
Convergence benchmark 1	309	262	43	12	11	10	10
Convergence benchmark 2	665	331,9	48,4	13,3	10,1	10	10

TABLE 3.3: influence de la probabilité de transition sur la convergence

Remarque 3.3. On remarque que dans les deux benchmarks, la convergence diminue au fur et à mesure que la probabilité de transition augmente.

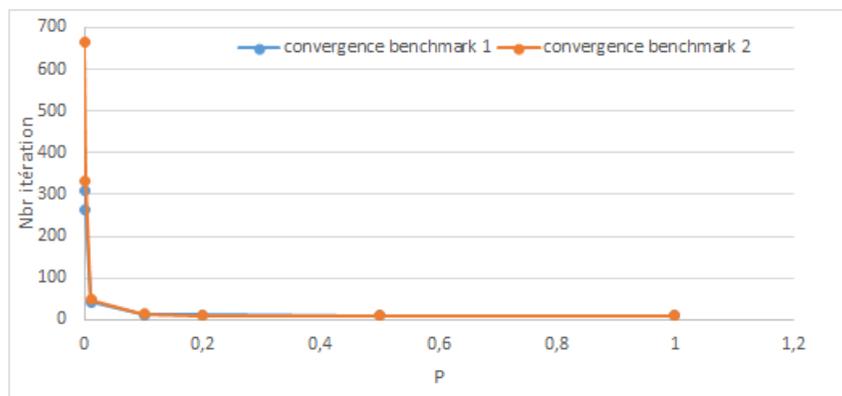


FIGURE 3.7: L'influence de la probabilité de transition sur la convergence

Conclusion : La probabilité de transition influence sur la convergence .

3.3.4.3 Quantité initiale de phéromone et taux d'évaporation

Nous avons remarqué que la quantité initiale de phéromone déposée sur tous les arcs n'as pas une influence sur la convergence ainsi que le taux d'évaporation (La quantité initiale est la même sur tous les arcs et le taux d'évaporation est le même). On outre cela a un effet sur la quantité finale.

le tableau suivant montre les résultats :

ρ	0	0.2	0.4	0.6	0.8	0.9
Convergence benchmark 1	10	10	10	10	10	10
Convegence benchmark 2	10	10	10	10	10	10

TABLE 3.4: Influence de tau d'évaporation sur la convergence

Quantité initiale	0.2	0.5	1	1.5	2	2.5
Convergence benchmark 1	10	10	10	10	10	10
Convegence benchmark 2	10	10	10	10	10	10

TABLE 3.5: Influence de la quantité initiale de phéromone sur la convergence

3.4 Conclusion

Dans ce chapitre nous avons présenté l'algorithme de colonies de fourmis pour la reconfiguration spatiale ce qui revient à retrouver les meilleurs chemins en fonction des données présentes en temps réel.

Durant les tests sur la convergence de l'algorithme, nous avons constaté que cette dernière n'est atteinte que lorsque certaines paramètres sont bien ajustés. Les pistes de phéromone sont renforcées à chaque fois qu'une solution est trouvée.

Cette analyse nous a fait comprendre le fonctionnement de cet algorithme, ce qui nous a permis de faire une proposition qui sera présentée dans le chapitre suivant.

CHAPITRE 4

PROPOSITION D'UNE AMÉLIORATION

4.1 Introduction

Cette partie du travail est réservée à la présentation de notre proposition pour contribuer à une amélioration. Dans la nature, les fourmis suivent les chemins optimaux, ces chemins sont les plus renforcés en phéromone. Au cours des tests effectués précédemment nous avons remarqué que les pistes de phéromone sont renforcées à chaque fois qu'une solution est trouvée. Nous nous sommes fait comprendre que des améliorations peuvent être effectuées.

4.2 Présentation de la proposition

L'exécution des algorithmes de colonies de fourmis se termine par le renforcement des pistes de phéromone. Durant la recherche de la solution optimale, les quantités de phéromone sont mises à jour pour toutes les solutions que ce soit une solution locale ou une solution globale. La quantité de phéromone finale observée doit logiquement être plus forte sur les arcs de la solution globale (naturellement les fourmis suivent le chemin le

plus optimal et par conséquent le plus renforcés), alors que dans certaines cas cette quantité est moins forte par rapport aux arcs des solutions locales.

La figure(4.1) montre les résultats sur la quantité de phéromone lors de la recherche de la solution(le chemin optimal entre la station 12 et la station 6) :

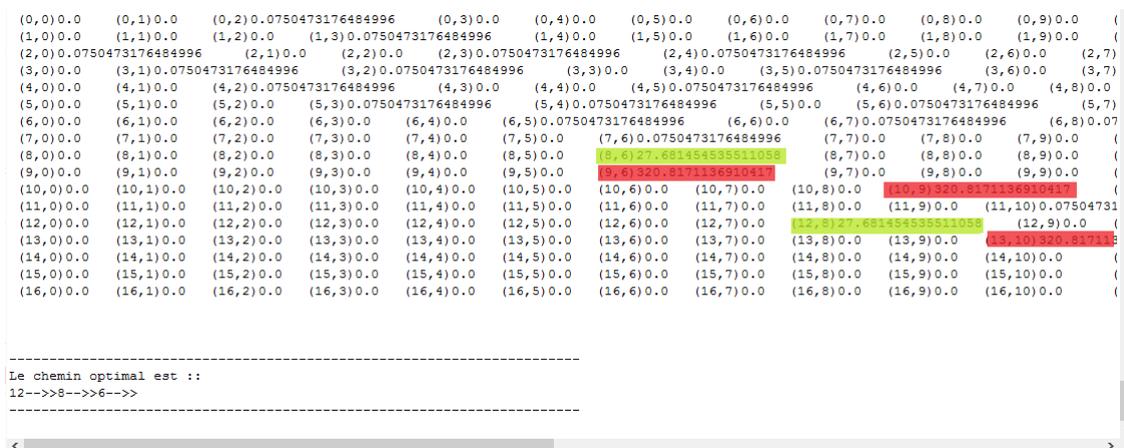


FIGURE 4.1: La quantité de phéromone finale lors de la recherche de la solution

Avec la couleur verte : la quantité de phéromone sur le chemin optimal. Avec la couleur rouge : la quantité de phéromone sur un autre chemin.

Remarque 4.1. On remarque que la quantité de phéromone finale est moins forte sur les arcs de la solution optimale.

4.3 Solutions

Pour remédier à la remarque définis dans la section précédente, nous proposons d'apporter une modifications sur l'algorithme original de la manière suivante :

- Seul la meilleure fourmi peut mettre à jour les pistes de phéromone, en d'autre terme on ne renforce que les pistes de la meilleure solution(Ce principe est utilisé dans l'algorithme *MIN- MAX AS*).

Ainsi l'algorithme devient comme décrit dans la figure(4.2) :

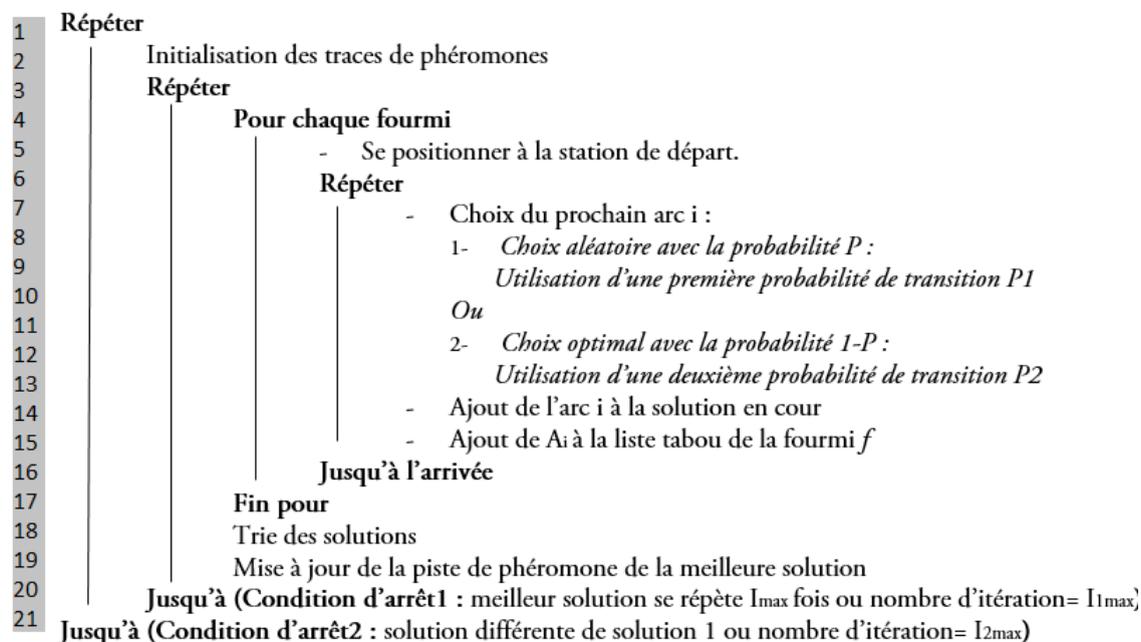


FIGURE 4.2: ACFRS amélioré

4.4 Implémentation

4.4.1 Classes et méthodes de l'algorithme

La classe ACFRS : Contient la méthode *main()*, exécute l'ensemble des méthodes de l'algorithme .

- La méthode *GetAntsNumber()* : cette méthode calcule le nombre de fourmi nécessaire en fonction de nombre d'arc de réseau.
- La méthode *initParametres()* : Avant d'appliquer l'algorithme, il est impératif d'initialiser les différents paramètres suivants :
 - Quantité de phéromone : les valeurs de phéromone sont initialisée à une valeur τ_0 fixé par l'utilisateur.
 - La probabilité de transition, les paramètres $\alpha, \beta, \gamma, \sigma, \rho$.
 - La station de départ et la station d'arrivée.

La classe Ant : Contient la liste de tabou, la solution en cours et une méthode pour calculer la quantité de phéromone ajouté par chaque fourmi.

La classe Graphe : Contiens les méthode suivantes :

- Méthode *GetDonnee()* : récupère les données d'entrée (les trois matrices T_i, D_i, C_i).

- Méthode *GetStationVoisines()* : retourne les station voisine de la station actuelle.
- Méthode *GetProbabilité()* : qui retourne un tableau de probabilité pour chaque station voisine.
- Méthode *GetNextStation()* : retourne la station suivante à visiter.

La calsse Paramètres : Contient l'ensemble des paramètres de l'algorithme

La classe Solution : contient la liste des meilleures solutions trouvées, une méthode pour le tri des solutions.

La classe Phéromone : contient les méthodes suivantes :

- *InitializePh()* : Sert à initialiser la quantité de phéromone sur tous les arcs.
- *Update()* : Sert à mettre à jour la quantité de phéromone.

4.4.2 Résultats

Après avoir effectuer les améliorations précédentes nous avons obtenu les resultats suivants :

4.4.2.1 Pour la recherche de la première solution

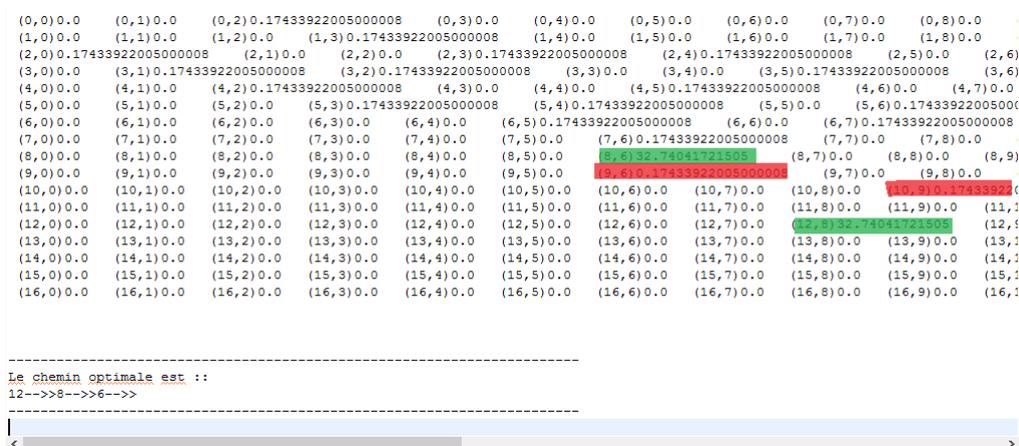


FIGURE 4.3: La quantité de phéromone finale

Avec la couleur verte : la quantité de phéromone sur le chemin optimal. Avec la couleur rouge : la quantité de phéromone sur un autre chemin.

Nous pouvons remarquer que seul la meilleure piste de phéromone est renforcé. Cela nous approche de la réalité(les fourmis naturelles suivent les pistes les plus renforcés).

4.4.2.2 Résultats sur la probabilité de transition :

P	0.0001	0.001	0.01	0.1	0.2	0.5	1
Convergence benchmark 1	10	10	10	10	10	10	11.3
Convergence benchmark 2	10	10	10	10	10	10	10

TABLE 4.1: influence de la probabilité de transition sur la convergence après amélioration

Dans ce tableau, on voit que la convergence est atteinte pour un nombre d'itération= $Imax = 10$ pour toutes les valeur de P sauf pour $P = 1$, le nombre d'itération >10 .

Conclusion : La modification apportée a influencé positivement sur la convergence.

4.4.2.3 Résultats sur le nombre de fourmis utilisées :

Benchmark 1							
Nbr de fourmi	1	2	4	8	12	16	18
Nbr d'itération	34	17,41	11,34	10,09	10,04	10	10
Benchmark 1							
Nbr de fourmi	1	15	30	45	60	75	100
Nbr d'itération	83,53	12,15	10,8	10,4	10,26	10,09	10,04

TABLE 4.2: influence de nombre de fourmi sur la convergence

Remarque 4.2. On remarque que la convergence est atteinte plus rapidement au fur et à mesure que le nombre de fourmis augmente. En comparant avec les résultats de premier algorithme(le nombre de fourmi doit être proche de nombre d'arcs), le nombre de fourmis nécessaire pour la convergence est plus petit.

Conclusion : La modification apporté sur l'algorithme a influencé positivement sur le nombre de fourmis utilisée.

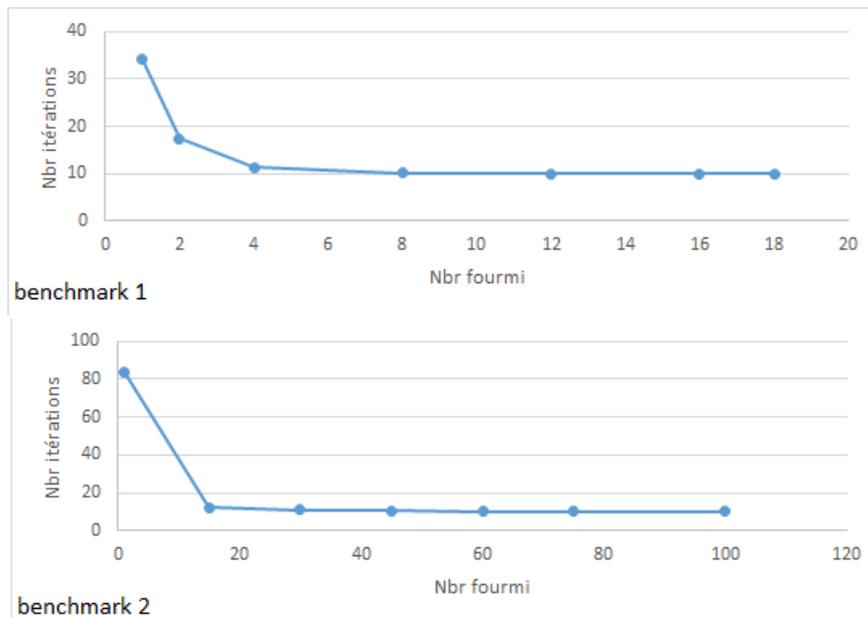


FIGURE 4.4: Influence de nombre de fourmis sur la convergence après amélioration

4.5 Conclusion

Cette partie a fait l'objet de notre contribution. En effet nous avons pu effectu e une modification sur l'algorithme original. Cette modification s'est r ev el e efficace en terme de rapidit e de la convergence. Un autre apport se voit dans le nombre de fourmi utilis e durant la recherche tel que le nombre n ecessaire pour la convergence est plus petit que dans l'algorithme original.

CONCLUSION GÉNÉRALE ET PERSPECTIVES

Le travail effectué dans ce projet traite les problèmes liés à la régulation des réseaux de transport multimodals. Il s'agit d'un besoin majeur pour les exploitant de ces réseaux afin d'améliorer la qualité de service offerte aux usagers du transport en commun.

Le point de départ de la réalisation de ce projet était une récolte des informations nécessaires pour mieux appréhender la problématique. Nous avons commencé par une étude bibliographique sur les réseaux de transport multimodal ce qui nous a permis de comprendre les difficultés actuelles dans ces réseaux. Ensuite nous nous somme penchés sur les algorithmes à colonies de fourmis, nous avons vu que ces insectes résoud des problèmes relativement complexes. Plusieurs algorithmes sont issus de l'algorithme de base "AS", chaque variante est utilisée pour résoudre certains problèmes.

Ensuite, nous avons présenté un algorithme utilisé pour la reconfiguration spatiale, nous l'avons étudié de près et nous avons tiré des conclusions. A partir de ces dernières, nous avons pu apporter une amélioration. L'apport de cette amélioration se voit dans la mise à jour des pistes de phéromones.

Après l'implémentation de cet algorithme, nous avons effectué des testes, les résultats se sont améliorés par rapport à la convergence et au nombre de fourmis. Nous pouvons dire que notre but est atteint.

L'apport de ce travail a été d'une importance très considérable, en effet, Le système a exigé un maximum de connaissances en programmation ce qui nous a permis de mettre en pratique les stratégies étudiées dans notre cursus universitaire, dans des livres, sur internet.

Néanmoins les résultats de ce modeste travail constituent les bases d'un travail à poursuivre et à améliorer pour une étude beaucoup plus approfondie.

Ainsi, les perspectives futures sont dans un premier temps d'améliorer l'autre algorithme pour la régulation horaire afin d'arriver à un système d'aide à la décision complet et utilisable.

En deuxième lieu, nous souhaitons développer un simulateur pour les réseaux de transport intégrant les deux algorithmes ,l'utilisation des données réelles pour valider notre amélioration .

BIBLIOGRAPHIE

- [AMR12] Fouzia AMRANI. *Techniques de routage par automate cellulaire : Adaptation à la régulation d'un réseau de transport urbain*. PhD thesis, Université d'Oran Mohamed Boudiaf, 2012.
- [BEN13] Ahmed BENYAMINA. *Application des algorithmes de colonies de fourmis pour l'optimisation et la classification des images*. PhD thesis, Université d'Oran Mohamed Boudiaf, 2013.
- [JD03] Patrick Siarry Éric Taillard Johann Dréo, Alain Pétrowski. *Métaheuristiques pour l'optimisation difficile*. EYROLLES, 2003.
- [LA02] Hakim LAÏCHOUR. *Modélisation multi-agent et aide à la décision : application à la régulation des correspondances dans les réseaux de transport urbain*. PhD thesis, Université des sciences et technologies de Lille U.F.R. I.E.E.A, 2002.
- [MEJ12] Hinda MEJRI. *Un système d'aide à la régulation d'un réseau de transport multimodal perturbé : réponse au problème de congestion*. PhD thesis, Ecole Centrale de Lille, 06 2012.
- [SBI11] Gilles Coppin Sahbi Ben Ismail, François Legras. *Synthèse du problème de routage de véhicules*. Télécom Bretagne, 2011.
- [SOL08] Christine SOLNON. *Optimisation par colonies de fourmis*. Lavoisier, 2008.

-
- [WID01] Marino WIDMER. Les metaheuristiques : Des outils performants pour les problemes industriels. *Troyes (France)*, pages 2,3,4, Avril 2001.
- [ZGA07] Hayfa ZGAYA. *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*. PhD thesis, Ecole Centrale de Lille, 2007.
- [ZID06] Kamel ZIDI. *Système Interactif d'Aide au Déplacement Multimodal (SIADM)*. PhD thesis, Ecole Centrale de Lille Université des Sciences et Technologies de Lille, 2006.
- [ZID07] Salah ZIDI. *SARR.Système d'aide à la régulation et la reconfiguration des réseaux de transport multimodal*. PhD thesis, Université des Sciences et Technologies de Lille, 2007.

Résumé

Les biologistes ont fait une constatation sur les insectes sociaux en général et les colonies de fourmis en particulier tel qu'elles résolvent naturellement des problèmes relativement complexe, les algorithmes de colonies de fourmis sont nés suite à cette constatation. Nous avons fait recours à un algorithme de colonies de fourmis pour aider à la reconfiguration spatiale des réseaux de transport multimodaux qui font face à des perturbations complexes et aléatoires, afin d'avoir des traitements rapides et en temps réel. Une amélioration de cet algorithme a été apporté, cette dernière a fait preuve de son apport comme l'ont montré les résultats obtenus.

Mots clés : Transport multimodal, perturbation, ACF, planification, reconfiguration, Piste de phéromone.

Abstract

Biologists have made a finding on social insects in general and ant colonies in particular as they naturally solve relatively complex problems, the algorithms of ant colonies are born as a result of this finding. We have used an ant colonization algorithm to assist in the spatial reconfiguration of multimodal transport networks that face complex and random disturbances in order to have rapid and real-time processing. An improvement of this algorithm has been made, the latter has shown its contribution as shown by the results obtained.

Key words : Multimodal transport, disturbance, ACF, planning, reconfiguration, Pheromone track.
