

Thèse

Présentée à

L'université d'Evry Val d'Essone

Préparée à

l'institut national de recherche sur les transports et leur sécurité

Par

Sofiene Kachroudi

Spécialité : Automatique et Mathématiques appliquées

Commande et optimisation pour la régulation du trafic
urbain multimodale sur de grands réseaux urbains

Soutenance Janvier 2010

Devant le jury composé de :

Didier Dumur	Professeur, Ecole Supérieure d'Electricité (Supélec)	examinateur
Jean-Pierre Quadrat	Directeur de recherche, INRIA	examinateur
Jean Patrick Lebacque	Ingénieur général des Ponts, Eaux et Forêts	examinateur
Damien Koenig	HDR, Ecole nationale supérieure en Systèmes Avancés et Réseaux	rapporteur
Florian de Vuyst	Professeur, Ecole Centrale de Paris	rapporteur
Saïd Mammar	Professeur, Université d'Evry Val d'Essone	directeur de thèse
Neila Bhourì	Chargée de recherche, INRETS	encadrant

A la mémoire de mon grand-père Mhammed Kachroudi,

A mes parents Ajmi et Mhenia Kachroudi,

A mes soeurs Nadia, Najla, Nedra, Nessrine et Safa,

A tous les Kachroudis

Remerciements

Je tiens à remercier Said Mammar, mon directeur de thèse, pour sa bonne humeur, sa disponibilité, ses encouragements et son sens de l'écoute.

Je remercie aussi Neila Bhourri, mon encadrante à l'INRETS, pour son suivi continue, ses conseils utiles et les discussions animées.

En parlant de l'encadrement, je ne peux oublier les apports et les conseils de Gérard Scemama et Jean Patrick Lebacque. Sous le suivi du comité de thèse, leurs conseils et leurs suggestions m'ont permis de prendre les bonnes décisions.

Je voudrais remercier mes rapporteurs Florian de Vuyst et Damien Koenig pour l'intérêt qu'ils ont porté à mon travail, les remarques et les suggestions qu'ils m'ont faites.

Je souhaite aussi remercier les autres membres de jury, Didier Dumur qui m'a fait l'honneur de présider mon jury de thèse et Jean-Pierre Quadrat.

Mais cette thèse s'est également inscrite dans un environnement humain particulièrement chaleureux. Je pense à toutes ces personnes grâce à qui je ne me suis jamais senti seul, personnes qui m'ont accompagné au long de ces années de travail. Je conclurai donc en remerciant mes amis : Salim Mammar, Fabien Badeig, Fatma Boustelitane, Flore Larcher et Krytyna Biletska. Aussi je tiens à remercier Mahdi Zargayouna pour sa bonne humeur bien qu'il soit mkachekh

J'aimerais aussi remercier les membres du laboratoire GRETIA dans leur ensemble pour leur bonne humeur et les discussions agréables qu'on a eu.

Last and not least, je tiens à remercier mes parents Ajmi et Mhenia sans qui je serai en train de ruminer de l'herbe dans un pré (!!). Un grand merci à mes soeurs Nadia, Najla, Nedra, Nessrine et Safa, mes cousins Saif et Siwar et mes grand-parents Mhammed, Kablouti, Fatma et la grande Salouha.

Et je termine enfin en remerciant tous ceux que j'ai pu oublier.

Tables des matières

1	Une revue sur les systèmes de contrôle du trafic	18
1.1	Introduction	18
1.2	Les systèmes de régulation du trafic général	19
1.2.1	L'affectation du trafic	19
1.2.2	La régulation via les feux de signalisation	19
1.3	Les systèmes de régulation des transports en commun	24
1.3.1	Régulation au niveau des stations	24
1.3.2	Régulation entre les stations	25
1.4	Les systèmes de priorité aux véhicules de transport en commun	26
1.4.1	Priorité passive	27
1.4.2	Priorité active à base de règles	29
1.4.3	Priorité active basée sur l'optimisation	40
1.5	Les systèmes de régulation mixte : trafic général et priorité aux transports en commun	42
1.6	Conclusion	45
2	Analyse et modélisation du trafic des véhicules particuliers	47
2.1	Introduction	47
2.2	Représentation mathématique des variables du trafic	47
2.2.1	réseau routier urbain	47
2.2.2	Les feux de signalisation	50
2.2.3	Respect des contraintes	57
2.3	Modélisation du trafic urbain	61
2.3.1	Revue des modèles du trafic	61
2.3.2	Modèle <i>store-and-forward</i>	62
2.4	Modèle d'écoulement du trafic	63
2.4.1	Équation de conservation	64
2.4.2	Étude en simulation	67
2.5	Conclusion	74
3	Modèle de progression des véhicules de transport en commun de surface	75

3.1	Introduction	75
3.2	Modèle simple	76
3.2.1	Algorithme pour le passage d'une station	77
3.2.2	Algorithme pour le passage d'un carrefour à feux	79
3.3	Modèle complet	87
3.3.1	Calcul de la longueur de la file d'attente devant le bus	88
3.3.2	Algorithme pour le passage d'une station	90
3.3.3	Algorithme pour le passage d'un carrefour à feux	101
3.4	Étude en simulation	105
3.4.1	Description du réseau et des flux	105
3.4.2	Influence de la charge du trafic	107
3.4.3	Influence des paramètres des plans de feu	108
3.5	Conclusion	110
4	La régulation du trafic multimodal	112
4.1	Introduction	112
4.2	Formulation générale d'un problème d'optimisation	112
4.2.1	Problème standard de l'optimisation mono-objectif	112
4.2.2	Problème standard de l'optimisation multi-objectif	114
4.3	Optimisation par essais particuliers	115
4.3.1	Algorithme original de l'optimisation par essais particuliers	116
4.3.2	Algorithme modifié de l'optimisation par essais particuliers	121
4.3.3	Les essais particuliers pour l'optimisation multi-objectif	124
4.4	Stratégie primaire	128
4.4.1	La régulation un problème d'optimisation	128
4.4.2	Architecture	131
4.5	Stratégie secondaire	142
4.5.1	Commande linéaire quadratique	142
4.5.2	Architecture	148
4.6	Conclusion	151
5	Étude en simulation de la stratégie de régulation	152
5.1	Introduction	152
5.2	Le réseau routier urbain	152
5.2.1	Caractéristiques géométriques	152
5.2.2	Caractéristiques des flux des véhicules particuliers	153
5.2.3	Caractéristiques des lignes de transport en commun	155
5.2.4	La stratégie des plans de feux fixes	157
5.3	Étude de sensibilité par rapport aux paramètres de l'algorithme d'optimisation	159

5.3.1	Sensibilité par rapport aux paramètres	161
5.3.2	Sensibilité par rapport au nombre de particules et nombre maximal d'itérations	169
5.4	Sensibilité par rapport aux horizons de simulation et de contrôle	173
5.5	Influence des objectifs de régulation	179
5.6	Analyse des performances de la stratégie	183
5.6.1	Analyse selon les consignes de régularité données aux transports en commun	183
5.6.2	Analyse selon la priorité : véhicules particuliers ou transport en commun	189
5.6.3	Comportement de la stratégie sous l'effet d'une perturbation simple	193
5.7	Conclusion	195

Tables des figures

1	Décomposition du temps de course	14
1.1	plan de feu d'un carrefour simple	20
1.2	Classification selon l'étendue des zones de régulation	26
1.3	Classification selon les actions de régulation	27
1.4	Rajout de ligne de saut et phase spéciale	30
2.1	Représentation graphique d'un carrefour réel	49
2.2	Plan de feu interdit : 2 phases de vert disjointes pour le même flux	51
2.3	Plans de feu autorisés	52
2.4	Les deux variables des plans de feu pour deux configurations différentes	53
2.5	Les deux configurations (1) et (2)	54
2.6	Carrefour élémentaire à 4 branches	55
2.7	Carrefour compliqué à 3 phases	56
2.8	Déplacement faisant sortir les variables de leur domaine de définition	58
2.9	Domaine de définition finale du couple (G_1, G_2)	59
2.10	flux sortant durant un cycle	63
2.11	nombre de véhicule dans un arc	64
2.12	réseau virtuel de test	67
2.13	Évolution du nombre de véhicules dans l'arc 1	70
2.14	Évolution du nombre de véhicules dans les arcs 2 et 7	72
2.15	Évolution du nombre de véhicules dans les arcs 1, 2 et 7	73
3.1	Bus au passage d'une station	78
3.2	Bus au passage d'un carrefour à feux	80
3.3	Un bus derrière une file d'attente	80
3.4	temps de passage moyen du bus sur les 10 carrefours pour 1000 essais	82
3.5	temps de passage du bus sur les 10 carrefours pour un seul essai	83
3.6	Scénario de test	86
3.7	Position du bus à la fin du cycle	86
3.8	Situation spatiale et temporelle du bus	89

3.9	Bus devant une station	90
3.10	Configuration 2 des plans de feu	99
3.11	Réseau de test virtuel	105
3.12	Influence de la charge du trafic	107
3.13	Différence entre les deux modèles	108
3.14	Influence des durées des phases de vert	109
3.15	Différence entre les deux modèles	109
3.16	Influence des instants de début des phases de vert	110
4.1	Minimum global et local d'une fonction	113
4.2	Relation de dominance et front de Pareto	115
4.3	Différentes façons pour le voisinage d'une particule	117
4.4	Prise en compte des contraintes dans l'algorithme <i>PSO</i>	119
4.5	fonctions de test	120
4.6	Minimum des fonctions de test pour l'algorithme original	120
4.7	Convergence de la 1 ^{re} composante du vecteur x pour l'algorithme <i>PSO</i>	121
4.8	Minimum des fonctions de test pour l'algorithme modifié	123
4.9	Convergence de la 1 ^{re} composante du vecteur x pour l'algorithme <i>GCPSO</i>	124
4.10	Estimateur de la densité des solutions	126
4.11	Schéma de principe de la commande prédictive	133
4.12	Trajectoire souhaitée et trajectoire réelle estimée	135
4.13	Architecture du module de régulation	141
4.14	Architecture complète de la stratégie de régulation	142
4.15	Réseau de la ville de Paris modélisé sous Dynasim	145
4.16	Nombre de véhicules et valeurs des durées des feux verts pour le carrefour 4	147
4.17	Architecture complète de la stratégie secondaire	150
5.1	Réseau virtuel de test	153
5.2	Réseau virtuel de test	156
5.3	Carrefour simple à deux phases	158
5.4	Critère C_{VP}	162
5.5	Critère C_{TC}	162
5.6	Moyenne des critères C_{VP} et C_{TC}	163
5.7	Evolution du critères C_{VP}	164
5.8	Evolution du critères C_{TC}	164
5.9	Moyenne des critères C_{VP} et C_{TC}	165
5.10	Evolution du critères C_{VP} et C_{TC}	166
5.11	Moyenne des critères C_{VP} et C_{TC}	166
5.12	Evolution du critères C_{VP} et C_{TC}	167

5.13	Moyenne des critères C_{VP} et C_{TC}	167
5.14	Evolution du critères C_{VP} et C_{TC}	168
5.15	Moyenne des critères C_{VP} et C_{TC}	169
5.16	Evolution du critères C_{VP} et C_{TC}	170
5.17	Moyenne du critères C_{VP} et C_{TC} et temps d'exécution	170
5.18	Evolution du critères C_{VP} et C_{TC}	171
5.19	Moyenne du critères C_{VP} et C_{TC} et temps d'exécution	172
5.20	Temps d'exécution pour PSO et $GCPSO$ en fonction de la taille de l'essaim	173
5.21	Critères VP et TC pour différents horizons de simulation pour l'algorithme PSO	175
5.22	Critères VP et TC pour différents horizons de simulation pour l'algorithme $GCPSO$	175
5.23	Critères VP et TC pour différents horizons de contrôle pour l'algorithme PSO	176
5.24	Critères VP et TC pour différents horizons de contrôle pour l'algorithme $GCPSO$	177
5.25	Critères VP et TC pour différents horizons de contrôle pour l'algorithme PSO	178
5.26	Critères VP et TC pour différents horizons de contrôle pour l'algorithme $GCPSO$	179
5.27	Critères VP et TC pour différentes valeurs de p pour l'algorithme PSO	180
5.28	Moyennes des critères VP et TC pour différentes valeurs de p pour l'algorithme PSO	181
5.29	Critères VP et TC pour différentes valeurs de p pour l'algorithme $GCPSO$	182
5.30	Moyennes des critères VP et TC pour différentes valeurs de p pour l'algorithme $GCPSO$	182
5.31	Progression moyenne des 4 lignes de bus pour une consigne de 100 m	184
5.32	Progression moyenne des 4 lignes de bus pour une consigne de 200 m	184
5.33	Progression moyenne des 4 lignes de bus pour une consigne de 300 m	185
5.34	Progression moyenne des 4 lignes de bus pour une consigne de 400 m	185
5.35	Progression moyenne des 4 lignes de bus pour une consigne de 500 m	186
5.36	Progression moyenne des 4 lignes de bus pour une consigne de 600 m	186
5.37	Progression moyenne des 4 lignes de bus pour une consigne de 650 m	187
5.38	Nombre de VP et durée moyenne de vert pour les deux groupes d'arcs	188
5.39	Nombre de VP et durée moyenne de vert pour les arcs 49 et 51	189
5.40	Progression moyenne des 4 lignes de bus pour le scénario 1 : VP et TC	190
5.41	Progression moyenne des 4 lignes de bus pour le scénario 2 : VP	190
5.42	Progression moyenne des 4 lignes de bus pour le scénario 3 : TC	191
5.43	Évolution du nombre total de véhicules pour chaque scénario	192
5.44	Évolution du nombre total de véhicules avec une priorité mixte et consigne de 350 m	193
5.45	Progression moyenne des 4 lignes de bus avec un trafic perturbé	194
5.46	Évolution du nombre de VP et durée moyenne de vert pour les arcs 18 et 31	195

Introduction

L'urbanisation accélérée des populations ainsi que le développement des échanges socio-économiques dans les grandes agglomérations ont eu pour conséquence une croissance rapide de la mobilité urbaine. La facilité économique à la propriété d'un véhicule personnel d'une part et le manque d'investissement dans les transports urbains collectifs d'autre part ont engendré une forte croissance du trafic urbain asphyxiant même plusieurs zones urbaines en allongeant les congestions. Les conséquences de cette croissance sont néfastes pour les collectivités locales et pour les personnes. Elles peuvent être désastreuses pour l'environnement dans la mesure où les congestions résultent en l'émission de quantités de plus en plus importantes des gaz à effet de serre.

Les plans des déplacements urbains (PDU), qui ont été formalisés en 1982, ont pour but l'organisation du transport urbain. Tous les modes sont concernés et les modes alternatifs comme les transports collectifs urbains sont particulièrement favorisés. Dans ce cadre, plusieurs actions sont envisagées et nous pouvons citer quelques unes :

- L'aménagement et l'exploitation des réseaux et des voiries afin de les rendre plus efficace.
- Le développement des transports collectifs et l'encouragement à favoriser ce mode et le covoiturage.
- La mise en place de stratégies pour réduire les congestions et réguler le trafic général.
- La mise en place de tarifications et billettiques pour les déplacements.

De notre point de vue, pour organiser le trafic urbain, il est nécessaire d'insister sur deux points. Le premier concerne la circulation des véhicules particuliers et consiste à rendre plus efficace le réseau routier urbain en particulier au niveau des carrefours à feux. C'est au niveau des lignes de feu où nous observons des marges importantes pour améliorer la circulation. Le second point concerne les véhicules de transport en commun. Un service de haut niveau des transports collectifs de surface peut être une alternative pour l'usage des véhicules particuliers. Pour la région parisienne par exemple, le parc des transports collectifs de surface se caractérise par une offre abondante et des véhicules confortables. Cependant l'abondance de l'offre et le confort ne peuvent pas seuls garantir un service de très bonne qualité. Les temps d'attente des usagers dans les stations, la régularité de la ligne et les temps de parcours des véhicules sont des critères primordiaux pour juger de la qualité d'une ligne de transport collectif de surface. Vu de cet angle, l'offre des transports

collectifs reste en deçà de ce qui peut être réellement obtenu. Dans une étude réalisée par le Syndicat des Transports d'Ile de France (STIF) en 2001, il s'avère que 40% du temps de course d'un véhicule de transport en commun est concerné par l'état des feux de signalisation au niveau des carrefours (figure 1). De notre point de vue, améliorer le service des transports en commun de surface passe par une gestion optimisée des feux de signalisation.

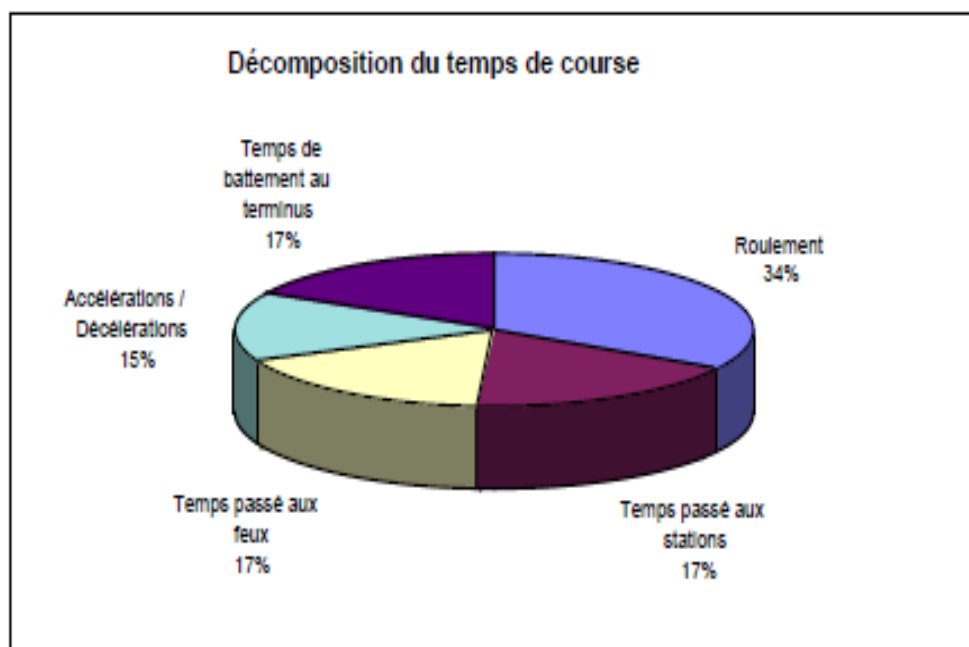


FIGURE 1 – Décomposition du temps de course

Nos travaux de thèse s'inscrivent dans le cadre des travaux sur la régulation du trafic urbain dans l'unité *GRETIA* à l'*INRETS*. La thèse a pour objectif la régulation en temps réel du trafic général et de la circulation des transports en commun de surface à travers une gestion efficace et optimisée des feux de signalisation au niveau des carrefours. Le besoin de stratégies de régulation du trafic urbain a été établie depuis des décennies. Le manque de puissants outils informatiques et mathématiques et de modèles de trafic simples ont longtemps empêché le développement de stratégies globales efficaces sur de larges réseaux urbains. Avec la croissance rapide des capacités calculatoires des ordinateurs et les nouveaux développements mathématiques théoriques, la régulation sur de larges réseaux a pris le pas sur la micro-régulation qui ne s'intéresse qu'aux carrefours isolés. Pour les transports en commun de surface, la nécessité de systèmes gérant la régularité des lignes est un constat qui fait l'unanimité pour les professionnels d'autant plus que la fiabilité de ce mode se dégrade fortement avec la croissance du trafic général. Cependant très peu de systèmes permettent de garantir la régularité d'une ligne. La plupart se limitent à donner une priorité locale au niveau des feux de signalisation. Pour améliorer la régularité sur de larges réseaux, il est nécessaire de prendre des actions

globales qu'on peut appliquer localement au niveau des carrefours. Il est nécessaire aussi de tenir compte des conditions du trafic général. Pour cela il faut disposer de modèles capables de décrire simplement et fidèlement la circulation des véhicules de transport en commun sur de larges réseaux traversés par des flux de véhicules particuliers.

La thèse tente de résoudre un problème de régulation ambitieux. Ce problème consiste à réguler globalement le trafic général sur de larges réseaux urbains et à garantir la régularité de toutes les lignes de transport en commun du réseau. Les variables de contrôle sont les paramètres des feux de signalisation du réseau. La thèse est organisée en 5 chapitres :

Chapitre 1

Le chapitre 1 passe en revue les systèmes de régulation du trafic qui sont répertoriés en trois catégories. La première catégorie concerne les systèmes qui ont pour seul objectif de réguler seulement le trafic des véhicules particuliers. Nous y présenterons les stratégies classiques cycliques et acycliques ainsi que quelques nouvelles stratégies. Les systèmes de la seconde catégorie ne s'intéressent qu'à la régulation des véhicules de transport en commun. Nous présenterons les techniques les plus utilisées et quelques résultats numériques. La dernière catégorie de systèmes a pour objectif la régulation simultanée des véhicules particuliers et des véhicules de transport en commun. Deux stratégies seront particulièrement passées en revue et qui sont la stratégie *NeTPrior* et le système *DARVIN*.

Chapitre 2

Le chapitre 2 est une manière douce pour entrer dans le vif du sujet. Deux thèmes composent le chapitre. D'abord, la représentation mathématique des variables qui nous intéressent dans la thèse. Ensuite, la modélisation du trafic des véhicules particuliers en milieu urbain. Les réseaux urbains seront représentés comme des graphes directs. Les plans de feu des carrefours seront décrits mathématiquement. Une opération qui sera très utile pour le chapitre 4 sera décrite en détail. Il s'agit de la manière de respect des contraintes sur les variables des plans de feu. Pour la modélisation du trafic, nous passerons rapidement en revue les modèles les plus connus avant de décrire plus en détail le modèle *store-and-forward*. Nous nous intéresserons ensuite à la description du modèle que nous avons développé. Ce modèle reprend les mêmes équations de conservation que les modèles rencontrés dans la littérature et l'approximation du modèle *store-and-forward*. Mais il a été modifié pour décrire, en plus, les conditions sous saturées du trafic. Un cas d'étude simple et des résultats de simulation bouclent le chapitre.

Chapitre 3

Le chapitre 3 est dédié au développement du modèle des transports en commun. Chaque véhicule de transport en commun est représenté par sa position dans le réseau. Le modèle décrit l'évolution des positions des véhicules avec un pas de temps égal à la durée de cycle d'un feu. L'approche de modélisation est fondée sur le principe des événements qui seront de deux types : arrêt dans une station et passage d'un carrefour à feux.

Dans un premier temps, nous décrivons le modèle simple. Ce modèle se distingue par deux points :

- La file d’attente devant une ligne de feu en aval d’une station ne peut pas remonter jusqu’à la station.
- Seules les durées des phases d’un cycle de feu sont considérées. L’instant de début de ces phases sont considérées inconnues et représentées comme des variables aléatoires.

Dans un deuxième temps, nous décrivons le modèle complet qui ne considère plus les limites du modèle simple. Il est, de ce point de vue, bien plus précis mais aussi bien plus complexe.

Le chapitre se termine par une étude en simulation sur un réseau virtuel de 9 carrefours.

Chapitre 4

C’est le chapitre de la synthèse. Il est constitué de deux grandes parties : d’abord, des généralités sur l’optimisation et l’algorithme par essais particuliers, ensuite la description de l’architecture des stratégies de régulation.

Dans la première partie, nous décrivons les problèmes standards de l’optimisation mono et multi objectif. Nous nous intéressons plus particulièrement à l’algorithme d’optimisation des essais particuliers. Des résultats numériques pour deux versions de l’algorithme viennent clôturer cette partie.

Dans la seconde partie, nous proposons deux architectures pour la stratégie de régulation. Les deux architectures s’appuient sur la commande des systèmes en boucle fermée et utilisent la commande prédictive généralisée. La première architecture utilise, pour la commande, directement l’algorithme des essais particuliers. La deuxième utilise la commande linéaire quadratique pour limiter les dimensions de l’espace de recherche de l’algorithme d’optimisation.

Chapitre 5

Le chapitre 5 est entièrement consacré à la mise en place de la stratégie de régulation sur un réseau de carrefours. Le réseau comporte 16 carrefours et traversé par 4 lignes de transport en commun. Dans un premier temps, nous décrivons les caractéristiques géométriques du réseau ainsi que les caractéristiques des flux de véhicules. Nous effectuons ensuite une étude sur les paramètres de l’algorithme d’optimisation qui permettent de déterminer les réglages les plus efficaces pour l’optimisation. Enfin une étude détaillée sur les performances de la stratégie sera décrite en comparaison avec une stratégie de plans de feux fixes.

Au cours de cette thèse, les articles suivants ont été présentés dans différentes conférences :

Revue Internationale : en cours de soumission

- [1] S. Kachroudi and N. Bhouiri, A predictive urban traffic control strategy. *Transportation Research Part B*.

Conférences Internationales

[2] S. Kachroudi and N. Bhourì, A multimodal model for an urban traffic control policy. *IFAC*, Séoul, Juillet 2008.

[3] S. Kachroudi and N. Bhourì and S. Mammar and G. Scemama. Modèle de prédiction semi-macroscopique pour les véhicules de transport en commun en milieu urbain, *CIFA*, Bucharest, Septembre 2008.

[4] S. Kachroudi and N. Bhourì. Multiobjective particle swarm optimization for traffic regulation, *12th Symposium on Control in Transportation Systems (IFAC)*, California, Septmebre 2009.

Congrès National

[5] S. Kachroudi and N. Bhourì. Communication interprocessus entre Scilab et Dynasim pour l'évaluation d'une stratégie de commande optimale, *ATEC-ITS*, Paris, Janvier 2007.

Chapitre 1

Une revue sur les systèmes de contrôle du trafic

1.1 Introduction

Pour les véhicules particuliers, l'objectif principal des systèmes de contrôle et de régulation est de rendre le trafic plus fluide de sorte que les usagers passent le moins de temps possible (temps de parcours) sur le réseau routier. Pour les exploitants des transports en commun, l'objectif principal n'est pas de minimiser le temps de parcours de ces véhicules mais de maintenir une certaine régularité des lignes. La régularité d'une ligne peut se mesurer de deux façons selon les caractéristiques de la ligne et les objectifs des exploitants. La première consiste à garantir un intervalle régulier entre les véhicules successifs d'une même ligne. La seconde vise à ce que les temps de passage d'un véhicule respectent une table horaire fixée à l'avance.

Pour les deux modes de transport en milieu urbain, les principaux retards se situent au niveau des carrefours à feux. C'est pour cela que le réglage des paramètres des feux de signalisation représente le "levier" le plus important pour réguler le trafic général et la circulation des véhicules de transport en commun. L'intégration de ces deux objectifs est un problème difficile étant donné les différences dans les objectifs de régulation et les différences de caractéristiques entre les deux modes : les véhicules particuliers (VP) et les véhicules de transport en commun (TC).

Dans ce chapitre, nous passons en revue trois types de systèmes. La première famille concerne les systèmes régulant seulement le trafic général. La deuxième famille s'intéresse seulement à la régulation des transports en commun. La troisième et la dernière concerne la régulation mixte des deux modes. Une attention quasi-totale est portée aux systèmes de régulation via les feux de signalisation. D'autres approches comme l'affectation du trafic pour les VP ou la régulation au niveau des stations pour les TC sont décrites sommairement.

1.2 Les systèmes de régulation du trafic général

Les premiers travaux sur les congestions en milieu urbain datent du début des années 1960 grâce aux travaux de Gazis et Potts. Depuis, plusieurs stratégies ont vu le jour dans le but de réduire les congestions et minimiser les temps de parcours des véhicules dans les zones urbaines. L'objectif de cette section n'est pas de décrire toutes les stratégies existantes et les différentes approches mais de mettre la lumière sur les plus importantes stratégies. Nous décrivons brièvement les stratégies d'affectation dynamique et nous concentrons notre attention sur les stratégies agissant sur les feux de signalisation.

1.2.1 L'affectation du trafic

L'affectation du trafic est une stratégie qui permet d'équilibrer les flux de véhicules entre les différentes zones du réseau, d'éviter les congestions sur certains axes et de minimiser les temps de parcours. L'affectation du trafic se réalise par le biais du guidage des utilisateurs qui s'effectue le plus souvent par l'intermédiaire de panneaux à messages variables ou par des informations en temps réel sur l'état du trafic via des systèmes d'informations sophistiqués tels qu'internet ou la téléphonie mobile. Papageorgiou en 1999 [56] a établi une classification des stratégies de guidage selon que l'information parvient à l'utilisateur avant qu'il prenne la route ou continuellement jusqu'à sa destination. D'autres critères de classification ont été établis selon différents aspects : nature de l'objectif, l'approche de résolution, etc ... L'affectation du trafic, en tant que moyen de régulation, est une mesure relativement récente. Les premiers travaux ont commencé au début des années 1990 avec Papageorgiou [55], Charbonnier et al. [16], Mahmassani et Peeta [50] (stratégies itératives) etc...et Bolleli et al. [13], Hawas et Mahmassani [37] (stratégies réactives). Les premiers travaux théoriques et résultats expérimentaux montrent qu'il s'agit d'un type de stratégies prometteuses. Cependant il reste à développer de puissantes méthodes de calcul et de contrôle capable de résoudre ce problème pour de larges réseaux urbains et pour des conditions de trafic variables.

1.2.2 La régulation via les feux de signalisation

Ce sont les stratégies les plus traitées dans la littérature et les plus utilisées sur le terrain. Un plan de feu d'un carrefour est déterminé par la spécification de quatre variables : la longueur du cycle, le plan de phase, la durée de chaque phase et enfin le décalage. La figure (1.1) illustre un plan de feu d'un carrefour simple. Le décalage est l'écart entre les cycles de deux carrefours successifs. Il permet d'établir "*une onde verte*" pour les véhicules traversant successivement deux carrefours.

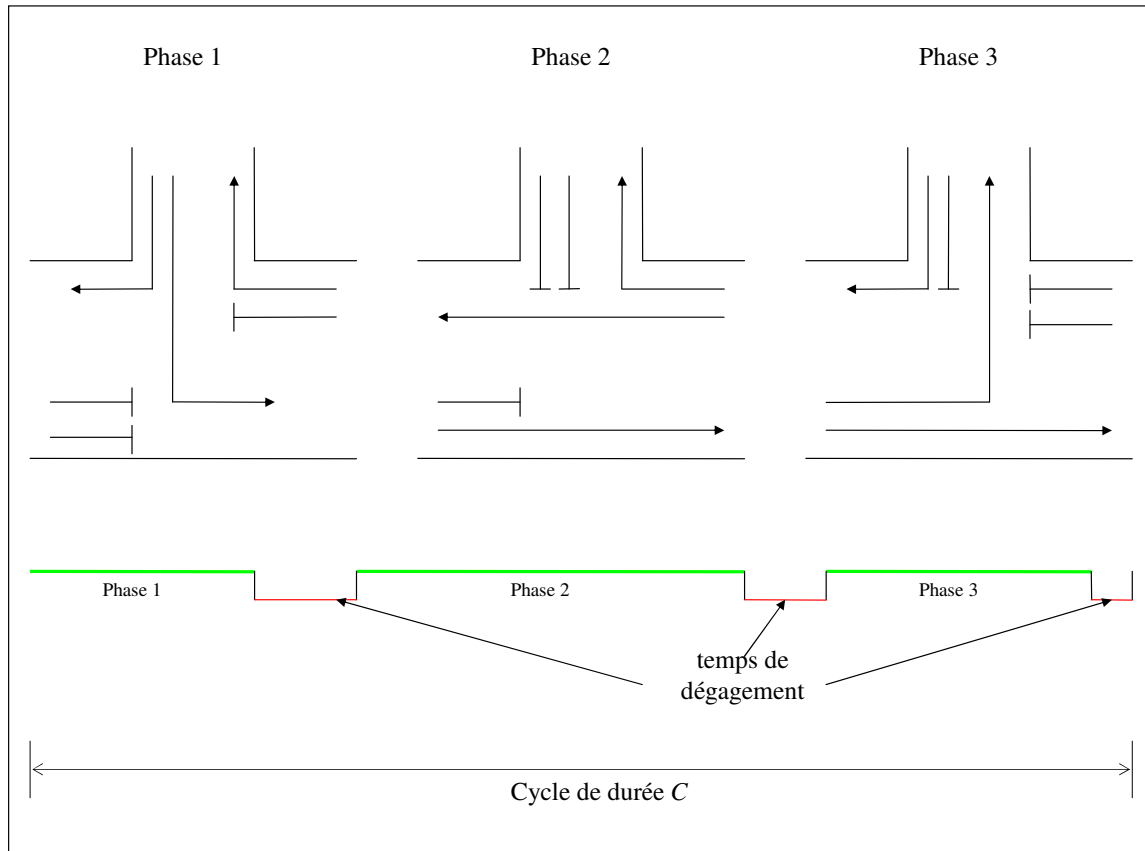


FIGURE 1.1 – plan de feu d’un carrefour simple

Papageorgiou [56] a établi plusieurs classifications des stratégies de régulation selon plusieurs aspects. Les plans de feux fixes sont opposés aux plans de feux adaptatifs (ou en temps réels) qui s’adaptent aux conditions générale du trafic. Les plan de feux fixes sont déterminés à partir de données statistiques recueillies sur les sites réels. Le plus célèbre des systèmes de plan de feux fixes est TRANSYT ([61]).

Les stratégies de régulation en temps réel s’appuient généralement sur trois modules ([57]). Le premier module est un module de collection et de traitement des mesures des variables du trafic. Plusieurs types de capteurs peuvent être utilisées pour mesurer les variables du trafic comme par exemple les boucles magnétiques qui permettent de mesurer la densité, le flux ou le nombre de véhicules dans un tronçon de route; les caméras vidéos qui peuvent en plus évaluer la longueur de file d’attente devant une ligne de feu; etc... Le deuxième est généralement un module de prédiction qui s’appuie sur un modèle de trafic plus ou moins sophistiqué. À partir des variables mesurées, ce module prédit les futures valeurs de ces mêmes variables ou/ et d’autres. Enfin le module d’optimisation détermine les variables optimales des plans de feux qui permettent de minimiser un critère donné. Le critère peut être le temps de parcours, l’inverse d’un flux (revient à maximiser le flux), les temps d’attente aux carrefours ou le nombre de voitures dans le réseau. Chaque stratégie a ses propres variables mesurées, son propre module de prédiction et son propre module d’optimisation. Les

stratégies diffèrent aussi par les approches de résolution (programmation dynamique, heuristiques, réseaux de neurones, etc ...) et même par la philosophie de l'approche (notion de cycle fixe, instants de basculement du vert au rouge, etc...).

Les différences d'approche, de philosophie, de méthodes de résolution et de modèles de prédiction ont donné naissance à plusieurs stratégies de régulation en temps réel du trafic urbain. Dans la suite, nous établissons une distinction selon que la stratégie utilise la notion de cycle ou pas. Nous parlons alors de stratégies cycliques si la notion de cycle existe et de stratégies acycliques dans le cas contraire.

1.2.2.1 stratégies cycliques

La première famille considère les cycles comme fixes pendant une période donnée. Ceci n'empêche pas que les durées des cycles peuvent changer d'une période à une autre. Dès le début des années 1980, deux systèmes ont émergé. Il s'agit de *SCOOT* et *SCATS*. Le premier a été développé au Royaume Uni par Hunt et al. [39]. Il s'inscrit dans la continuité du système TRANSYT ([61]). Le système détermine au niveau de la zone de régulation la durée optimale du cycle commune à tous les carrefours. Les décalages sont calculés pour chaque carrefour en tenant compte des carrefours voisins. Enfin les durées des phases de vert sont déterminées au niveau du carrefour indépendamment des autres carrefours. Le second, *SCATS*, a été proposé par Lowrie en 1982 en Australie [48]. La particularité de ce système réside en l'existence de deux niveaux : un niveau stratégique et un niveau tactique. Le niveau stratégique détermine les plans de feux adéquats pour tout le réseau considéré. Le niveau tactique agit au niveau local du carrefour tout en respectant les contraintes établies par le niveau stratégique. Plus récemment, une nouvelle stratégie a été proposée par Diakaki et al. en 2002 [23]. Il s'agit de la stratégie *TUC* un composant d'une stratégie plus globale appelée *IN-TUC*. *TUC* a été proposée en réponse à certains inconvénients des stratégies *SCOOT* et *SCATS* qui concernent principalement les actions prises au niveau global qui ne sont pas assez réactives aux changements rapides des conditions du trafic. Cette stratégie ne comporte qu'un seul niveau qui permet de traiter tous les carrefours du réseau. Les cycles et les décalages étant fixes, *TUC* permet de déterminer les durées des phases du vert pour tous les carrefours dans le but de minimiser le nombre total des véhicules sur le réseau. La simplicité du modèle utilisé et sa linéarité permettent d'utiliser la commande linéaire quadratique, une puissante méthode du contrôle automatique des systèmes linéaires. Encore plus récemment Dotoli et al. [25] ont proposé en 2005 une stratégie de régulation qui détermine les durées des phases de vert et les décalages en tenant en compte les contraintes liées aux traversées piétonnes. Le modèle utilisé est très fin mais serait très gourmand en temps de calcul.

1.2.2.2 stratégies acycliques

Cette famille de stratégies de régulation ne considère pas explicitement la notion de cycles, de décalages ou de durées des phases de vert. Les successions de phases de vert et de rouge sont aperçues comme des instants de commutation. Sur un horizon de planification déterminé à l'avance, ces stratégies déterminent si, au pas de temps actuel, une commutation est nécessaire. Le choix des commutation s'effectue suivant un critère de performance donné.

Les premiers systèmes de ce type sont l'américain *OPAC* ([36]) et le français *PRODYN* ([32] ; [43]). *PRODYN* est composé de deux niveaux. Un niveau supérieur convertit le complexe problème d'optimisation de départ en de petits sous-problèmes correspondant chacun à un niveau inférieur qui traite un seul carrefour. La décomposition du problème de départ se base sur les méthodes de *décomposition-coordination*. La résolution des sous-problèmes au niveau des carrefours fait appel à la programmation dynamique. Au niveau local et sur un horizon de planification donné, l'algorithme détermine à chaque pas de temps (5 secondes) s'il est optimal d'effectuer une commutation entre les phases. L'optimalité est mesurée à l'aide d'un critère qui approxime le temps d'attente total de tous les usagers.

En Italie, la stratégie *UTOPIA* a été introduite en 1989 par Mauro et Di Taranto [51]. Cette stratégie s'appuie sur deux niveaux de contrôle : un niveau régional et un niveau local ([14]). Le niveau régional fait des estimations sur les changements dans le réseau et détermine les plans de feu de référence et certaines variables de réglage qu'il transmette au niveau local. Le niveau local détermine les réglages optimaux à appliquer sur les feux au niveau des carrefours. Le critère d'optimisation est constitué des temps d'attente des véhicules, du nombre d'arrêts et des longueurs de files d'attente pour tous les tronçons du carrefour en tenant compte des carrefours voisins. Pour le contrôleur régional, l'horizon d'optimisation est de 30 minutes. pour le niveau local, les décisions sont prises chaque 6 secondes.

Trois ans plus tard, le système *CRONOS* a été proposé en France ([12]). *CRONOS* s'inscrit dans la même lignée que *OPAC* et *PRODYN*. Ce système se distingue par la possibilité de choix entre une architecture distribuée ou une architecture centralisée. Aussi, il ne considère pas un plan de phase pré-spécifié. Le système est composé de trois module : un module de prévision qui prédit certaines variables du trafic, un module de simulation et un module d'optimisation. La résolution du problème d'optimisation s'effectue à l'aide d'une version modifiée de l'algorithme de Box.

Plus récemment, en 1998, le système nord-américain *RHODES* ([52]) est apparu. Son architecture est composée de trois niveaux hiérarchiques. Le niveau "Réseau" ou "*Network Load Control*" permet d'estimer la charge du trafic sur tous les arcs du réseau se basant sur les changements de la demande à l'entrée du réseau et des choix d'itinéraire des usagers. Ces estimations sont ensuite envoyées vers le "*Network flow Control*" qui détermine les durées approximatives du vert à allouer aux arcs. A partir de ces "contraintes", le niveau le plus bas "*Intersection Control*" choisit les changements de phases adéquats à appliquer au niveau des carrefours.

La complexité des phénomènes physiques du trafic urbain, la multitude de façon de l'influencer, la différence de perception des plans de feux et des phénomènes de congestion ont eu pour conséquence l'apparition de différentes stratégies de régulation. Chacune a ses apports et ses inconvénients. Les systèmes décrits dans les sections précédentes ont la particularité d'avoir été testés directement sur des sites réels. Dans la section suivante, quelques nouvelles stratégies seront décrites.

1.2.2.3 stratégies émergentes

Depuis les systèmes *SCOOT* et *SCATS*, plusieurs travaux sur la régulation du trafic ont émergé. Avec les avancées dans le calcul numérique et les nouvelles méthodes d'optimisation plus efficaces, plusieurs nouvelles stratégies ont été conçues et développées.

Abu-Lebdeh et Benekohal ([3]) ont proposé une stratégie utilisant les algorithmes génétiques pour l'optimisation. Cette stratégie est composée de deux modules : le "*control algorithm*" et la "*disutility function*". Le premier module détermine les durées des phases de vert et les décalages avec l'objectif de maximiser les flux de sortie des artères principales et des arcs adjacents. La "*disutility function*" mesure l'efficacité des réglages déterminés par le "*control algorithm*" selon la gestion voulue. Les auteurs ont proposé quatre variantes de la stratégie suivant le degré de priorité affecté aux arcs adjacents par rapport aux artères principales. Les deux modules s'appuient sur un modèle fin et réaliste avec un haut niveau de détail. Le problème d'optimisation est lourd et de grandes dimensions, il est en plus fortement contraint. D'où l'usage des algorithmes génétiques.

Une récente famille de stratégies est en développement aux États-unis. Elles se basent sur une approximation numérique convergente du modèle *LWR*. Cette approximation a été introduite par Daganzo en 1994 ([21]). Sa formulation en milieu urbain est introduite en 1995 ([22]). L'approximation se base sur la décomposition d'un tronçon de route en des cellules élémentaires dont la longueur dépend de la vitesse du flot et de l'intervalle temporel d'échantillonnage. Cette approximation est connue sous le nom de "*The cell transmission model*".

Nous notons les travaux menés par Lo ([46]), Lo et al. ([47]) et Beard et Ziliaskopoulos ([9]).

Lo ([46]) utilise la programmation mixte en nombre entier pour déterminer, à chaque cycle, les durées des phases du vert qui permettent de minimiser les retards dans le réseau. Les retards dans le réseau sont calculés par "*The cell transmission model*". La stratégie détermine aussi les décalages initiaux qui seront fixes par la suite.

Lo et al. ([47]) ont proposé le système *DISCO* qui se base sur le même modèle que Lo ([46]) mais qui permet de traiter des réseaux plus larges. La programmation mixte en nombre entier a été remplacée par les algorithmes génétiques pour pouvoir appliquer la stratégie en temps réel.

Toujours utilisant "*The cell transmission model*", Beard et Ziliaskopoulos (Beard et Ziliaskopoulos 2006) ont introduit une stratégie combinant l'optimisation des plans de feu et la résolution d'un problème d'affectation dynamique. L'objectif de la stratégie est de minimiser les temps de parcours dans le réseau.

Teorodović ([65]) a mis l'accent sur l'utilisation des méta-heuristiques basées sur les populations dans l'ingénierie du transport et en particulier dans la régulation du trafic. Par exemple, Wen et Wu ([68]) ont utilisé l'algorithme des colonies de fourmis pour trouver les décalages entre carrefours adjacents. Les résultats obtenus avec un réseau de neuf carrefours montrent que l'algorithme converge vers la solution optimale.

Enfin, ils existent une famille de stratégies basées sur les systèmes multi-agents comme les travaux de A. Bazzan [8].

1.3 Les systèmes de régulation des transports en commun

Les véhicules particuliers et les véhicules de transport en commun ont des caractéristiques très différentes. La circulation des voitures peut être assimilée au déplacement de paquets ou de flux. Vu le nombre restreint de véhicules de transport en commun qui circulent sur un tronçon de route, il est difficile de modéliser leur circulation comme un flux. Outre leurs vitesses sont très différentes, les bus doivent s'arrêter dans des stations dédiées. Du point de vue de la régulation, l'objectif des usagers des voitures est de passer le moins de temps possible sur le réseau routier alors que pour les opérateurs de transport en commun, l'objectif est de respecter une table horaire pré-établie ou d'assurer un intervalle de temps régulier entre les véhicules successifs d'une même ligne. Dans la suite, nous utilisons le terme particulier "bus" et le terme général de "véhicule de transport en commun" pour désigner la même chose.

La régulation de la circulation des véhicules de transport en commun comporte deux familles suivant que les actions de régulation se font au niveau des stations ou entre les stations.

1.3.1 Régulation au niveau des stations

Au niveau des stations, trois grandes techniques sont employées :

- Saut de stations ou "Station skipping" : action qui consiste à déterminer les stations où le bus ne s'arrêtera pas.
- Stationnement supplémentaire dans les stations : action qui consiste à déterminer le temps de stationnement supplémentaire d'un bus dans une station. Elle s'emploie quand le bus est en avance sur sa table horaire ou que l'intervalle entre le bus et son prédécesseur est petit.
- Tournée courte : action qui consiste à autoriser une partie de la flotte à faire des tournées plus courtes avec une fréquence plus élevée.

Le stationnement supplémentaire dans les stations a été la stratégie la plus traitée dans la littérature. Ceci peut s'expliquer par la facilité de sa mise en oeuvre. Elle est aussi la première des stratégies au niveau des stations à être traitée analytiquement ([27]). Parmi les premiers travaux, nous pouvons citer ([30]) et ([2]). L'objectif de ces travaux est de minimiser une fonction coût mesurant le temps d'attente des voyageurs. Les variables de décision sont les temps de stationnement supplémentaires ou les intervalles critiques entre véhicules successifs.

Les travaux évoqués ci-dessus n'utilisent pas d'informations en temps réels sur les positions des bus dans le réseau mais les considèrent comme des variables aléatoires avec différentes densités de probabilité. Ceci a eu pour conséquence que les actions à prendre étaient plutôt des spécifications de valeurs seuils et non une seule valeur précise.

Avec l'équipement des bus par des systèmes permettant de suivre en temps réel leurs trajectoires, il est devenu facile d'accéder à des informations sur les intervalles instantanés entre bus ou les temps de parcours. Eberlein en 1995 ([27]) et Zolfaghari et al. ([70]) ont proposé un modèle, qui se base sur ces informations en temps réels, pour contrôler les temps de stationnement de tous les bus sur le réseau. Le modèle considère aussi les cas d'une grande et faible affluences des voyageurs.

Les techniques de "parcours haut le pied" et de "parcours en descente" consistent à réaliser à vide une partie de la trajectoire de la ligne ou à faire vider le véhicule. Pour ces régulations, il y a deux types de stratégies : les stratégies pré-établies ou "off-line" et les stratégies en temps réel ([27]). Il existe peu de travaux pour la première catégorie (voir [35]) et ([6]). Les stratégies en temps réel ont été étudiées entre autres par Li [45]. En 1997, Eberlein et al. [28] ont formulé le problème de "Deadheading" (parcours haut le pied) en temps réel : étant donné un véhicule prêt à partir, les actions de régulation consistent à déterminer si ce véhicule doit sauter des stations et si oui de combien ? L'objectif est de minimiser le temps d'attente totale des voyageurs sur toute la ligne.

Eberlein et al. [29] ont comparé les stratégies de stationnement supplémentaire et les stratégies de saut de stations et ont conclu que les premières sont plus efficaces que les secondes pour assurer la régularité des lignes.

Toutes les stratégies de contrôle dans les stations partagent une même hypothèse. En effet, les temps de parcours des bus entre deux stations successives sont égales à leurs valeurs nominales fixes. Cette hypothèse constitue une limitation majeure puisque ces temps de parcours varient en fonction des conditions de circulation et de l'état des feux dans les carrefours.

1.3.2 Régulation entre les stations

La seconde famille concerne la régulation entre les stations. Trois paramètres influent directement sur la progression d'un bus d'une station à une autre : la vitesse du bus, les congestions dans les tronçons de route traversés et les temps d'attente au niveau des carrefours à feux.

Le contrôle de la vitesse est très rarement utilisé pour la régulation des bus puisque, pour des raisons de sécurité, on ne peut pas demander au conducteur d'accélérer sa cadence. Les premiers travaux ont été justement consacrés aux réseaux ferrés ([15]). Même pour ces réseaux, ces stratégies ne sont pas très efficaces ([27]). Pour les véhicules de transport en commun terrestre, Estrada et al. ([31]) ont proposé une méthode qui agit simultanément sur la vitesse des bus (seulement pour la réduire) et l'extension du feu vert pour les tronçons traversés. L'objectif de leur travail est de minimiser les variations d'intervalles entre les bus successifs. Cependant, ce travail repose sur plusieurs hypothèses difficilement justifiables dans la réalité comme

des distances toujours les mêmes entre les stations et entre les carrefours.

La régulation via les feux de signalisation reste le thème le plus traité dans la littérature. Dans la section suivante, nous passons en revue les systèmes donnant la priorité aux véhicules de transport en commun au niveau des carrefours à feux.

1.4 Les systèmes de priorité aux véhicules de transport en commun

Dans [11], Bhouiri et al. ont proposé une classification principalement axée sur l'étendue des réseaux concernés par la régulation (figure 1.2).

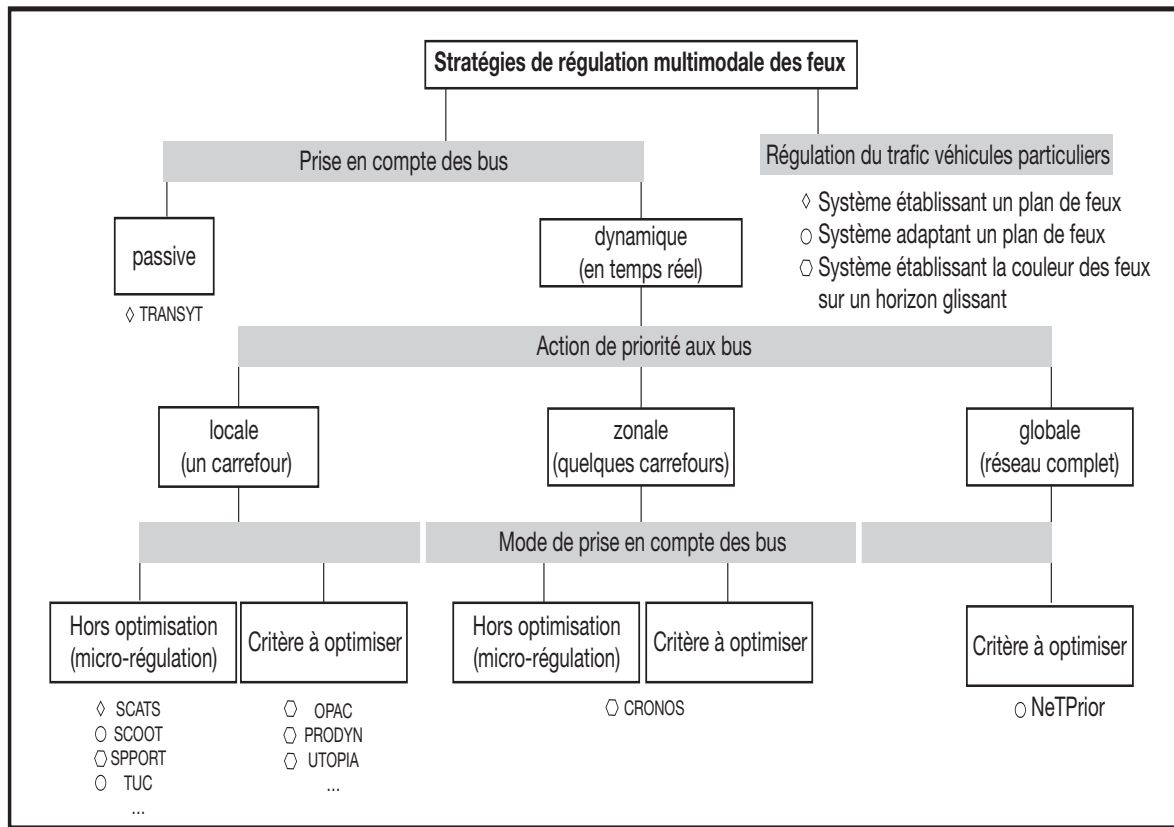


FIGURE 1.2 – Classification selon l'étendue des zones de régulation

A. Shalaby et J. Lee ([63]) ont proposé une classification basée sur la nature de la prise de décision pour la régulation. Ainsi ils distinguent les prises de décision basées sur les règles (simples ou dynamiques) et

les prises de décision basées sur l'optimisation d'un critère (figure 1.3, AVL : localisation automatique de véhicules, APC : Compteur automatique de passagers).

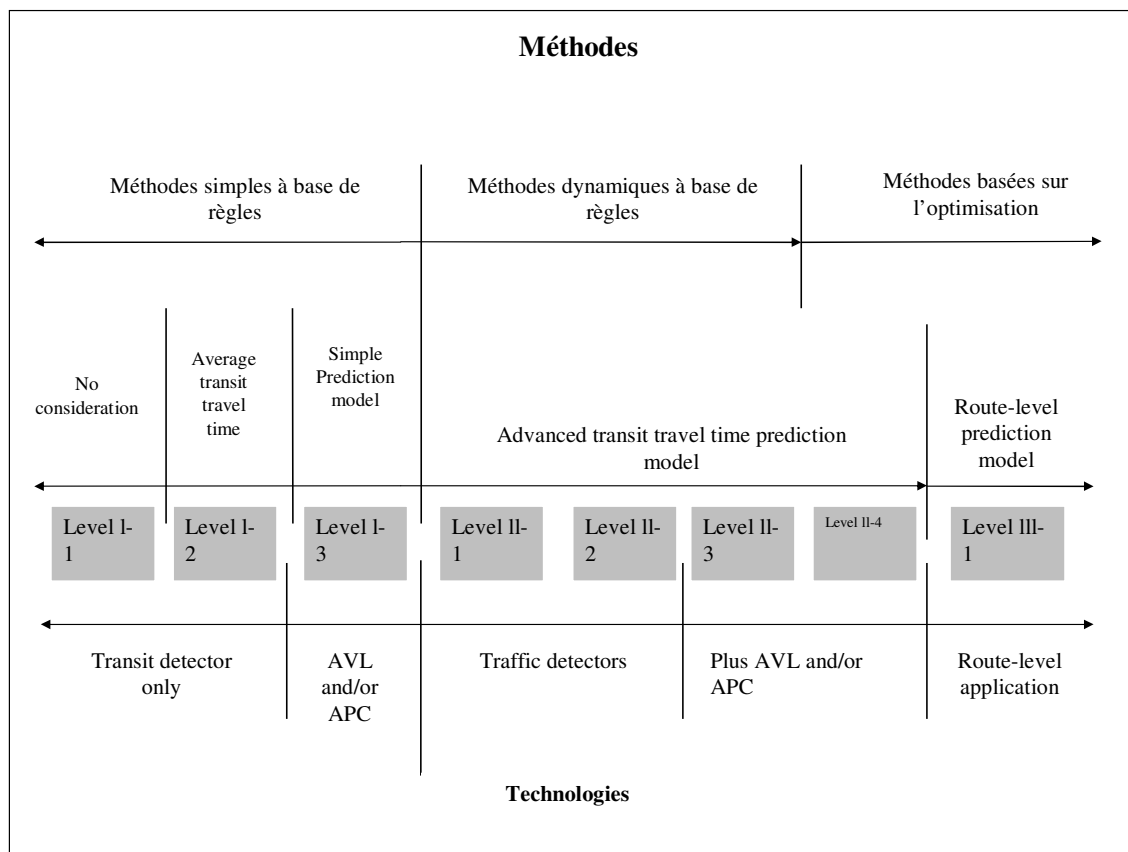


FIGURE 1.3 – Classification selon les actions de régulation

Dans un rapport du projet PRISCILLA de l'"Information Societies Technologies" de la commission européenne ([40]), une autre classification est proposée. Elle se base sur les différentes actions concrètes sur les feux. Ainsi, ils distinguent la priorité passive, la troncature des phases de rouges, l'extension du vert, les stratégies adaptative basées sur le contrôle en temps réels, etc...

Dans cette revue sur la régulation des véhicules de transport en commun via les feux, nous distinguons les systèmes à priorité passive de ceux à priorité active. Pour les systèmes à priorité active, nous nous intéressons plus particulièrement à la façon de prise de décision. Celle-ci peut être à base de règles sont ou bien se basant sur l'optimisation d'un critère.

1.4.1 Priorité passive

Les actions de priorité passive sont opérationnelles continuellement qu'il y ait détection ou pas d'un bus dans le réseau. Pour que ces actions soient plus efficaces, un certain nombre de conditions doivent être réunies ([5]) :

- Les variables du trafic comme les temps de stationnement dans les arrêtes commerciaux ou les temps de parcours dans les tronçons ne doivent pas varier beaucoup.
- La fréquence de circulation des bus doit être importante.
- Le volume du trafic ne doit pas être très important.

Pour parvenir aux objectifs de la régulation passive, nous pouvons appliquer certaines mesures comme ([11]) :

- Ajuster la coordination des feux entre carrefours adjacents à la vitesse des bus au lieu de celles des voitures.
- Réduire la longueur des cycles des feux afin de réduire les attentes des bus devant les lignes de feu.
- Fractionner les phase du vert du tronçon traversé par le bus.
- Concevoir les plans de feu en tenant compte des nombres de passagers plutôt que le nombre de véhicules.

La dernière méthode a été utilisée dans le système *TRANSYT-7F* ([67]). *TRANSYT-7F* est un système pour l’optimisation de la longueur de cycle, des durées des phases du vert et des décalage ([64]). Il reproduit les mouvements des flux et utilise un algorithme itératif qui permet de minimiser le critère suivant :

$$PI = \sum_N^{i=1} W_{D_i} D_i + KW_{S_i} S_i$$

Où N est le nombre d’arcs, W_{D_i} un coefficient pour les retards D_i dans l’arc i et KW_{S_i} un coefficient pour pénaliser le nombres d’arrêt S_i dans l’arc i .

La prise en compte de la priorité des bus dans *TRANSYT-7F* se fait en considérant les trajectoires des bus comme des arcs supplémentaires avec de nouveaux coefficients $WF_{D,S}$. Le choix de ces coefficients dépend de la fréquence de la ligne du bus, des mouvements des flux et des caractéristiques géométriques du réseau ([64]). Skabardonis a effectué une étude numérique sur une artère contrôlée par 10 feux où il a mesuré la sensibilité des retards des bus et des véhicules particuliers par rapport aux coefficients $WF_{D,S}$. Les résultats montrent que les retards subis par les bus diminuent avec les coefficients jusqu’à 20% et que les retards subis par les véhicules particuliers augmentent jusqu’à 40%.

Les plans de feu fixes peuvent être efficaces si la fréquence des lignes de bus est importante et que les prédictions des instants d’arrivée des bus sur les lignes de feux sont fiables. Des variations dans les temps de stationnement aux arrêtes commerciaux, des variations de la charge du trafic dans les arcs traversés par les bus peuvent rendre l’utilisation de ces méthodes obsolètes voire néfastes pour les bus et aussi pour le reste des usagers.

Par opposition à la priorité passive, la priorité active ne change les plans de feu que quand un bus est détecté dans le réseau. Les systèmes de priorité active englobent plusieurs stratégies avec des caractéristiques et des approches bien différentes. Une première tentative de classification repose sur la nature des actions de régulation. Nous pouvons agir sur les feux en imposant des règles qu'il faut respecter. Nous parlons alors de stratégies à bases de règles. Quand les décisions sont prises dans le but de minimiser un ou plusieurs critères, nous parlons de priorité basée sur l'optimisation.

1.4.2 Priorité active à base de règles

Ce sont les stratégies les plus intuitives à imaginer et les plus faciles à mettre en oeuvre. Les règles sont simples et dépendent des objectifs de la régulation. Si par exemple, le but est de minimiser le temps de parcours d'un bus, une règle peut être une priorité absolue quand un bus est détecté près d'un feu. Dans le cas où l'objectif de régulation est le respect d'une table horaire, une règle peut consister à ne donner la priorité au bus que s'il est en retard sur son programme, s'il est en avance rien n'est effectué et le plans de feu de base est gardé.

Les actions sur les plans de feu pour une priorité active à base de règles sont :

- Extension de la phase du vert : quand un bus approchant un feu en phase de vert, la durée de cette phase est étendue pour permettre au bus de passer.
- Troncature de la phase du rouge : la durée de la phase du rouge est tronquée quand un bus arrive sur le carrefour en phase de rouge.
- Insertion de phase : action qui consiste à insérer une phase spéciale pour le passage d'un bus quand il est détecté.
- Rotation de phase : L'ordre des phases du plan de feu de référence est changé afin de permettre au bus détecté de passer sans un temps d'attente.
- Déclenchement de phase : Il s'agit par exemple de rajouter une ligne supplémentaire pour les bus contrôlés par une phase spéciale (figure 1.4).

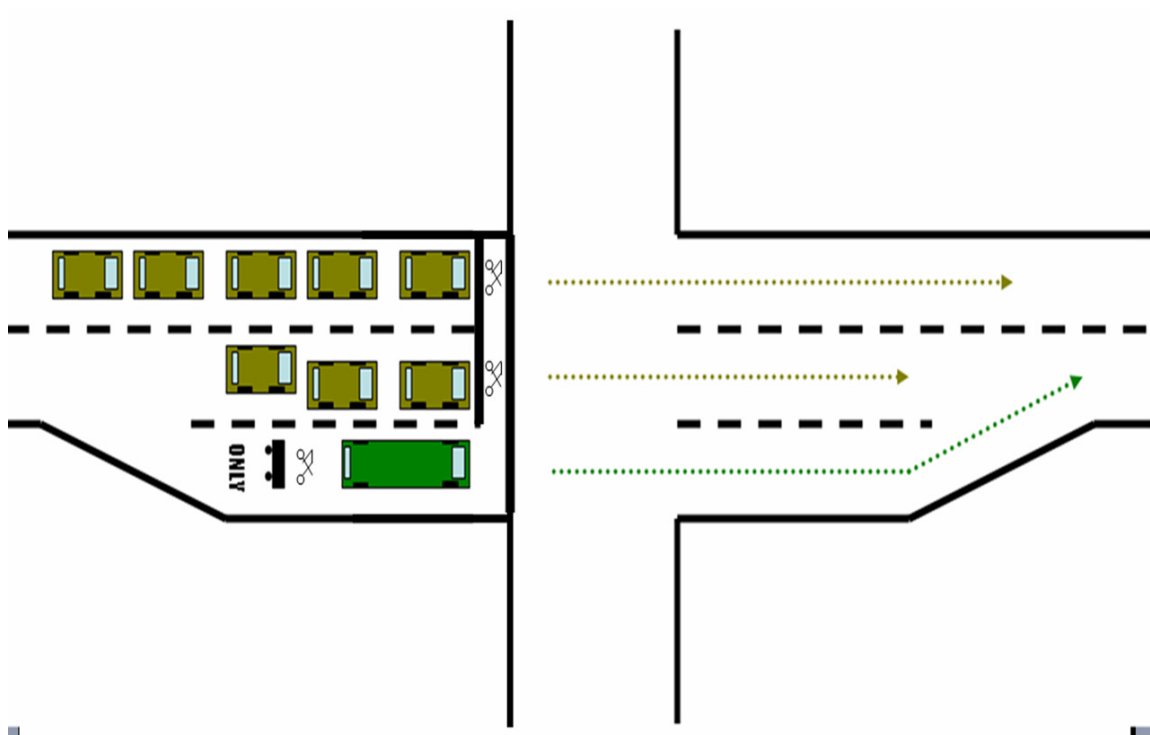


FIGURE 1.4 – Rajout de ligne de saut et phase spéciale

Les cinq stratégies peuvent être utilisées séparément ou ensemble. Les deux premières sont les plus utilisées car elles ne changent pas l'ordre des phases et ne rajoutent pas donc le temps perdu nécessaire à chaque changement de phase.

Afin de structurer et rendre plus claire cette revue sur les systèmes de priorité active à base de règles et les différents travaux qui s'y rapportent, nous divisons les systèmes en deux sous catégories. D'abord nous nous intéressons aux systèmes de priorité appliqués seuls sans une régulation du trafic général. Ensuite, nous présentons les travaux qui intègrent la priorité aux bus dans un système global de régulation du trafic général.

1.4.2.1 Systèmes indépendants du trafic général

Souvent les systèmes combinent les actions d'extension de vert et la troncature de rouge pour l'artère traversé par les lignes de transport en commun. Une compensation est allouée aux autres artères du carrefour quand l'extension de vert pour les bus dépassent un certain seuil. Une règle de ne pas donner la priorité deux fois de suite est aussi envisagée. Vincent et al. ([66]) ont montré par des études en simulation que l'extension du vert seule a des bénéfices très limités sur la progression des bus. En combinant les deux stratégies, les simulations montrent des bénéfices beaucoup plus importants pour les bus mais aussi une dégradation tout aussi importante pour les véhicules particuliers. En recourant au compensation et à la règle de ne pas autoriser une priorité deux fois de suite, la stratégie est moins efficace que la deuxième mais plus efficace que la

première avec une dégradation moins importante pour les autres usagers de la route. Cependant les auteurs ont conclu sur les bénéfices de cette stratégie pour tous les usagers et que la règle d'interdiction des priorités successives n'est pas nécessaire.

Richardson et Ogden ([60]) ont refusé l'idée de Vincent et al. ([66]) selon laquelle la règle de refus de deux priorités successives n'était pas nécessaire. Ils ont développé un système tel que, chaque fois qu'on donne une priorité à un bus, les temps perdus par les autres usagers à cause de cette priorité sont ajoutés à la durée du vert de ces usagers dans le cycle suivant. Quand ce temps cumulé dépasse un seuil, la priorité demandée par un bus est simplement refusée. Les auteurs ont conclu que les pertes causées pour les autres usagers pour des priorités successives dépassent les gains pour les bus.

Courage et al. ([20]) ont proposé l'extension du vert couplée à l'utilisation de sites propres et à la coordination des carrefours adjacents pour les directions surchargées en trafic. Les auteurs ont supposé que le chemin du bus est dans la direction surchargée. Bien évidemment ce n'est pas toujours le cas. Les différentes combinaisons de ses trois techniques ont généré quatre stratégies différentes. La première stratégie qui consiste seulement à l'extension de vert sans modification des durées de vert pour les autres artères. La combinaison des deux premières techniques ont fait augmenter la durée de vert pour les artères non traversées par le bus. La combinaison des techniques 2 et 3 et la combinaison des trois techniques ont fait diminuer les durées de vert de 20%.

Seward et Taube (

Khasnabis et al. ([44]) ont utilisé l'extension du vert et la troncature du rouge avec un modèle analytique. Leurs conclusions confirment celles de [62] que ces stratégies permettent la réduction de la flotte. Ce pendant leur étude n'a pas pris en compte les autres usagers de la route.

Ludwick ([49]) a comparé trois stratégies. Le premier algorithme ne permet une extension du vert que si le temps gagné par les passagers du bus est supérieur au temps perdu par les autres usagers de la route. Le second donne une extension du vert inconditionnelle ainsi qu'une troncature du rouge. Le troisième donne la priorité aux bus sous deux conditions : (i) le temps de priorité ne doit pas dépasser un seuil limite et (ii) attendre un certain temps avant de satisfaire une priorité demandée. Les résultats ont montré que le troisième algorithme est le plus prometteur. En effet, il a permis de gagner 25% sur le temps de parcours du bus et n'a causé qu'une augmentation de 7% sur le temps de parcours des autres usagers. L'autre conclusion de l'auteur est que le système de priorité aux bus est plus bénéfique pour les lignes avec hautes fréquences de passage.

Jacobson et Sheffi ([41]) ont conçu un système de priorité bus qui minimise les temps d'attente de tous les usagers. Ils ont observé que leur système a permis de réduire les temps d'attente pour les passagers du bus et aussi des véhicules particuliers. Ils confirment par la même occasion que ce genre de système est plus efficace avec une fréquence élevée des bus et des lignes chargées en passagers.

Heydecker ([38]) a utilisé l'extension de vert en se fixant des règles sur la durée maximale de l'extension et la non redondance de priorité sur deux cycles successives du même carrefour. D'après l'auteur, ces règles sont nécessaires pour rétablir la capacité du carrefour. Enfin, il suggère de mesurer l'efficacité d'une priorité aux bus en tenant compte de la chute de capacité du carrefour.

Balke et al. ([7]) ont développé et évalué une stratégie de priorité aux bus sans altérer la coordination entre les carrefours adjacents. Pour les auteurs, une stratégie de priorité intelligente doit respecter trois objectifs :

- donner la priorité aux bus sans perturber l'onde verte sur l'artère principal.
- donner la priorité sans altérer significativement la séquence normale des phases et leurs durées dans les artères secondaires.
- ne donner la priorité que pour les bus qui ont en besoin selon un critère fixé par les opérateurs de transport en commun.

La stratégie utilise la localisation, à chaque instant, du véhicule de transport en commun. Le système de localisation permet, dans un premier temps, d'estimer l'instant d'arrivée du véhicule dans une station ou sur la ligne d'arrêt d'un feu de signalisation. Cette estimation est effectuée par le module de prédiction "Arrival time Prediction Module". Ce module transmet ces informations à un module qui détermine si le véhicule concerné a besoin d'une priorité ou pas. Si la réponse est négative, on garde le plan de feu initial inchangé. Dans le cas contraire, on marque le véhicule comme prioritaire et c'est au module "Strategy selection module" de sélectionner la stratégie qu'il faut pour donner la priorité au véhicule concerné. Une fois la stratégie définie, un dernier module s'occupera d'implémenter la stratégie sélectionnée sur un contrôleur. La seule difficulté de cette implémentation est de garantir la coordination sur l'artère principal.

Le "Arrival time Prediction Module" permet d'estimer les instants d'arrivée d'un véhicule dans une station et sur la ligne d'arrêt d'un feu. Pour cela, les auteurs supposent connus les coordonnées des stations, des lignes de feu et du véhicule lui-même. Il est supposé connu aussi les caractéristiques de circulation du véhicule (vitesse, etc...) et l'historique des temps de stationnement du véhicule. A partir de ces informations et du suivi, en temps réel du bus, un algorithme permet d'estimer les variables suivantes :

- Le temps de parcours entre la position actuelle du véhicule et la station.
- Le temps de stationnement du véhicule dans la station.
- Le temps de parcours entre la station et le carrefour à feu.

L'instant d'arrivée du véhicule dans la station déterminera si le véhicule a besoin d'une priorité. L'instant d'arrivée sur la ligne du feu déterminera quelle stratégie choisir. Le "Strategy selection module" choisit une stratégie parmi trois. La première est l'extension du vert. La deuxième est l'insertion d'une phase spéciale. Et enfin la troncature du rouge ou revenir plus tôt à la phase du vert. Pour l'insertion de phase, les auteurs proposent pour l'exemple étudié deux instants possibles d'insertion. Le choix d'une stratégie parmi ces trois se fait de la manière suivante :

- Si le véhicule arrive dans une phase de vert alors rien ne change.
- Si le véhicule arrive dans la fenêtre d’extension alors l’extension du vert est effectuée.
- Si le véhicule ne peut arriver dans la fenêtre d’extension mais dans une phase d’insertion alors on insère une phase spéciale.
- Enfin on procède à une troncature de rouge de la dernière phase.

Les auteurs ont testé leur stratégie sur un réseau de trois carrefours et pour trois valeurs du ratio volume/capacité (v/c) : 0.5, 0.8 et 0.95. Le ratio v/c est le rapport entre le volume réel du trafic (nombre de véhicules sur une ligne) et la capacité maximale de la ligne (nombre de véhicules pouvant passer dans un cycle). Il apparait que le temps d’attente global pour tous les usagers est resté le même pour $v/c = 0.5$ et a augmenté respectivement de 2% et de 11% pour $v/c = 0.8$ et $v/c = 0.95$. Pour les véhicules de transport en commun, les temps des parcours ont été réduits de 25%, 27% et 27% pour les trois valeurs de v/c . Pour les flux incompatibles avec la circulation du bus, les retards causés par le système de priorité bus atteignent 4%, 8% et 26% respectivement pour $v/c = 0.5$, $v/c = 0.8$ et $v/c = 0.95$.

Rakha et Zhang ([58]) ont réalisé une analyse en simulation de l’impact d’une stratégie de priorité pour les bus sur le fonctionnement des carrefours isolés. Les auteurs ont effectué les simulations sur un carrefour avec quatre arcs utilisant le modèle *INTEGRATION* ([4]). La stratégie de priorité accorde une extension du vert si un bus arrive juste après la phase du vert et une troncature du rouge s’il arrive dans la phase du rouge. Enfin, si deux bus en conflit réclament une priorité en même temps, aucune action n’est envisagée.

Dans cette analyse complète, les auteurs ont étudié les effets de certaines variables du trafic sur la progression de bus et les autres véhicules :

- Les effets de la demande du trafic : 80 simulations ont été effectuées. 8 valeurs du temps de départ du bus ont été considérées ainsi que deux plans de phase et 5 niveaux de congestion. Les résultats montrent qu’en augmentant la demande du trafic, les retards, les temps d’arrêt et la consommation de carburants des bus augmentent aussi. Par rapport à une gestion sans priorité, la stratégie de priorité permet de réduire les retards, les temps d’arrêt et la consommation de carburants tout en augmentant le trafic. En effet, les bénéfices d’une stratégie priorité pour les bus augmentent avec l’augmentation du trafic. Concernant les véhicules particuliers, Les résultats montrent que le système de priorité accordé aux bus a des conséquences marginales sur les voitures. Cependant ces conséquences augmentent avec l’accroissement de la demande du trafic.
- Les effets de la demande des véhicules de transport en commun : Les résultats en simulation montrent que les avantages d’un système de priorité aux bus dépendent de la fréquence des bus. En augmentant la fréquence de passage des bus, les voitures sont de plus en plus pénalisées. Leurs temps d’attente et leurs temps d’arrêt augmentent.
- Les effets de la distribution de la demande du trafic à travers les axes et l’influence de l’axe par lequel la

priorité bus est demandée : Les auteurs observent que si les bus arrivent par des arcs congestionnés alors que le trafic dans les arcs en conflit est fluide, un système de priorité bus a des avantages pour tout les usagers. Et inversement, quand un bus arrive par un arc fluide et que les arcs en conflit sont congestionnés, la priorité bus a d'importantes conséquences négatives sur le trafic général.

- Les effets de la séquence des phases dans le plan de feu : Les performances d'une priorité pour les bus dépendent de l'instant dans lequel un bus arrive sur le carrefour et par conséquent de la phase dans laquelle le bus arrive. Les résultats montrent que si le bus arrive au début du cycle c'est à dire dans les premières phases, les effets d'une priorité est marginale pour tout le système. Au contraire, une arrivée dans les dernières phases résulte en d'importantes conséquences négatives sur tout le système.

- Les effets d'un plan de feu non optimale : Les conséquences de la durée d'un cycle et des durées des différentes phases sont analysées. Trois durées de cycles ont été considérées et cinq différentes configurations des durées de phases. Les résultats montrent que les retards et les temps d'arrêt des bus augmentent avec la longueur de cycle mais que la stratégie de priorité devient de plus en plus efficace par rapport à une gestion sans priorité. Pour les voitures, les retards augmentent avec l'augmentation du cycle par contre les temps d'arrêt et la consommation de carburants baissent. La priorité accordée aux bus semblent n'avoir qu'un effet marginale sur les voitures.

Les avantages d'un système de priorité bus semblent ne pas dépendre des durées des différentes phases. Par contre, les conséquences néfastes sur le trafic général dépendent des durées des phases. Les auteurs remarquent, que pour une gestion sous-optimale des durées des phases, un système de priorité bus peut, dans certains cas particuliers, résulter en des bénéfices pour tout les usagers.

- Les effets du temps de stationnement dans les stations proches du carrefour : Cinq valeurs du temps de stationnement ont été testés dans une station près du carrefour. Les auteurs observent, comme il a été prévue, que les avantages d'une priorité aux bus sont insensibles aux variations de ce temps de stationnement. Cependant, l'augmentation du temps de stationnement dans une station proche du carrefour a des effets négatifs sur le reste du trafic. En effet, le temps supplémentaire que passe un bus dans une station est une durée du vert perdue.

Zhou et al. ([69]) ont proposé la combinaison de deux types de stratégies. La première consiste à rajouter un site spéciale aux bus à l'approche du feu "queue jump". La seconde stratégie inclue l'extension du vert, la troncature du rouge, la suppression de phase et l'insertion de phase. Les auteurs ont relevé que les sites propres seuls sont inefficaces pour réduire les temps d'attente des bus. Ils ont relevé aussi que l'insertion de phase est la plus efficace des quatre actions. En effet elle permet la réduction des temps d'attente des bus sans dégrader la circulation générale du carrefour.

Les systèmes ci-dessus cités accordant une priorité active aux véhicules de transport en commun ont un effet positive sur la circulation de ces véhicules. Ils ont permis de réduire les temps d'attente, les temps d'arrêt et les temps de parcours des bus dans les carrefours contrôlés par des feux. Différentes études ont montré les

bénéfices de ces stratégies appliquées séparément ou ensemble. Il ressort néanmoins que l'extension de vert et à un degré moindre la troncature de rouge sont les techniques les plus utilisées pour donner la priorité aux bus. Ces études montrent aussi que dans la majorité des cas, une priorité absolue pour les bus résultent en des retards supplémentaires pour les autres véhicules qui partagent le réseau routier avec les bus. Il a été observé même qu'en accordant une priorité conditionnelle, les véhicules particuliers subissaient des retards et des arrêts supplémentaires. Ces inconvénients rendent, dans beaucoup de cas, l'utilisation et le déploiement de ces systèmes obsolètes et néfaste pour le trafic général. C'est pour cela qu'il est nécessaire avant de donner une priorité aux bus d'étudier l'impact de cette mesure sur les conditions générales du trafic et surtout d'intégrer ces stratégies de priorité dans des systèmes qui régulent les conditions générales du trafic.

1.4.2.2 Systèmes intégrés dans des stratégies de régulation du trafic général

Mirchandani et al. ([53]) ont proposé d'intégrer une stratégie de priorité pour les bus dans le système *RHODES*, une stratégie adaptative pour le contrôle en temps réel du trafic sur un réseau urbain. *RHODES* s'appuie sur des mesures en temps réel fournies par des boucles magnétiques installées au début de chaque tronçon. A partir de ces mesures, un module de prédiction donne des estimations sur les longueurs de files d'attente et des instants d'arrivée des flux sur les lignes de feu du carrefour. Étant donné ces estimations, un module d'optimisation basé sur la programmation dynamique détermine les durées des phases. En réalité, au début *RHODES* ne considère pas la notion de phase mais des instants de commutation. Pour des besoins d'ingénierie, *RHODES* considère maintenant les notions habituelles d'un plan de feu à savoir la longueur de cycle, la séquence de phases, les durées de chaque phase. La priorité pour les bus intégrée dans ce système s'appuie sur une idée fondamentale qu'on retrouve dans d'autres travaux. Cette idée est de considérer un bus comme un véhicule particulier pris en compte dans le module d'optimisation avec un poids de pondération particulier. Le système *RHODES* et la priorité bus intégré ont donné naissance au système *RHODES/BUSBAND*. Les auteurs considèrent deux versions de *RHODES/BUSBAND* selon qu'ils attribuent un poids fixe pour un bus donné ou un poids variable dépendant du nombre de passager et de son retard par rapport à un horaire fixé.

La première version considère un bus avec un poids fixe très important par rapport aux véhicules particuliers. Ceci a la conséquence suivante : chaque fois qu'un bus est détecté dans un tronçon, il est quasiment certain qu'en arrivant sur la ligne de feu, il va bénéficier d'une phase du vert qui lui permettra de passer le carrefour. C'est pour cela que cette version est appelée "Phase-Constrained" *RHODES/BUSBAND*. Les auteurs ont mené des tests sur le logiciel *CORSIM* sur un réseau de la ville de Tucson aux états unis. Le réseau comporte une artère principale sur lequel circule une ligne de bus et plusieurs carrefours. Il est à noter aussi que le logiciel *CORSIM* ([1]) intègre une stratégie de contrôle semi-adaptative du trafic appelée "*SAC*". Les simulations ont été réalisées pour les stratégies *SAC*, *RHODES* et *RHODES/BUSBAND*. En termes de temps de parcours, les véhicules sur l'artère principale, buses et voitures, ont gagné respectivement 8% et 8.5% pour le système *RHODES* et *RHODES/BUSBAND* par rapport à la stratégie *SAC*. Pour les tronçons adjacents, les gains sont de 24.3% et 21.2% toujours par rapport à *SAC*. Pour tous les usagers du réseau, les gains sont respectivement de 17.9 et de 19.1 pour *RHODES* et *RHODES/BUSBAND* pour l'artère principale et sont de 41% et 35.8% pour les tronçons adjacents. En ce qui concerne les bus, les gains

sur les temps de parcours sont de 5.2% et 13% pour *RHODES* et *RHODES/BUSBAND* par rapport à *SAC*.

La deuxième version considère des poids variables qui dépendent du nombre de passagers dans le bus et du retard que le bus possède par rapport à une table horaire pré-établie. Cette version est appelée "weighted-bus". Quand un bus est détecté à l'approche d'un carrefour, le système lui attribue un poids w_i qui est donné par :

$$w_i = n_i \times (1 + f_i)$$

Où n_i est le nombre de passagers qui sont dans le bus et $f_i = 0$ si le retard d_i du bus est négatif (pas de retard ou en avance) et $f_i = K \times d_i$ où K est un coefficient constant s'il y a un retard d_i positif. Les tests en simulation ont été réalisés sur un modèle élaboré pour l'évaluation de *RHODES* dans la ville de Seattle. Le réseau utilisé ne comporte qu'un seul carrefour. Il comporte aussi une station *A* à partir de laquelle les bus sont envoyés et une station *B* qui représente le terminus. L'artère principale est celle qui supporte la ligne de bus et il y a deux tronçons adjacents. L'étude a été réalisée pour les trois stratégies *SAC*, *RHODES* et *RHODES/BUSBAND* et pour deux volumes du trafic sur les tronçons adjacents. Le volume du trafic sur l'artère principale reste fixe à 1074 véhicules par heure. Les bus sont envoyés avec des décalages aléatoires uniformément distribués sur l'intervalle $[-30; 30]$ et un nombre de passager aussi aléatoire uniformément distribué sur $[0; 30]$. Les temps de parcours sont mesurés entre les stations *A* et *B*. Pour un volume de trafic faible sur les tronçons adjacents, le temps de parcours des usagers a été réduit de -0.27% et 2.58% pour *RHODES* et *RHODES/BUSBAND* par rapport à *SAC*. La dispersion autour des valeurs moyennes a été réduite de 13.8% et 18.1% . Pour seulement les bus, les temps de parcours ont été réduits respectivement de 0.39% et 4.6% et les dispersions autour de la moyenne de 9.9% et 14.1% . Pour un important volume du trafic sur les tronçons adjacents, le temps de parcours des usagers a été réduit de 0.43% et 5.2% . La dispersion a été réduite de 21.8% et 28% . Pour les bus, les temps de parcours ont été réduits respectivement de 0.43% et 6.39% et les dispersions de 24.9% et 26.5% .

Le système *RHODES* a permis de réduire les temps de parcours de tous les usagers du réseau routier urbain y compris les véhicules de transport en commun. Le système *RHODES/BUSBAND* a permis de réduire davantage les temps de parcours des bus et surtout de réduire la dispersion des temps d'arrivée ce qui permet d'améliorer la fiabilité des transports en commun. Il est aussi à noter que l'intégration de la priorité aux bus a eu pour conséquence d'augmenter les temps de parcours des véhicules se déplaçant sur les tronçons adjacents par rapport au système sans priorité. Cependant même avec cette priorité aux bus, le trafic sur les tronçons adjacents est largement amélioré par rapport à des plan de feu fixes.

Dion et Rakha ([24]) ont étudié l'intégration d'un système de priorité pour les bus sur un réseau de carrefours contrôlé par une stratégie de régulation adaptative du trafic. Le système de priorité bus comporte deux types d'action : l'extension du vert et la troncature du rouge. Deux versions de la stratégie de régulation adaptative du trafic sont utilisées. Une première stratégie qui détermine seulement les durées des différentes phases du plan du feu. La deuxième permet en plus de déterminer les décalages optimaux entre les carrefours voisins. Les deux stratégies sont implémentées dans le module *INTEGRATION* qui est aussi un logiciel de

simulation microscopique du trafic. Le module de *INTEGRATION* qui permet de déterminer les durées optimales des phases a pratiquement la même philosophie que le système *SCOOT*. Le module qui détermine les décalages optimaux est quant à lui inspiré des systèmes *TRANSYT-7F* et *SCOOT*.

Les tests en simulation effectués sous *INTEGRATION* ont été réalisés sous un réseau comportant une artère principale et 20 carrefours. L'artère s'étend sur une longueur de 6.35 km et traversée par une dizaine de ligne de bus.

Le système de priorité bus s'appuie sur les règles suivantes :

- Si un bus détecté arrive sur le carrefour durant une phase de vert alors aucune action n'est prise.
- Si un bus arrive après la fin de la phase de vert alors cette phase est étendue de n secondes pour que le bus passe le carrefour. La limite de l'extension est contrainte par une durée maximale de la phase du vert. L'extension du vert pour cette phase est déduite des autres phases du cycle à condition que leurs durées minimales du vert ne soient atteintes.
- Si un bus est détecté alors qu'un autre arc a une phase de vert alors cette dernière phase est terminée si elle a dépassé le minimum de vert alloué. La phase prioritaire se termine normalement à l'instant prédéfini.
- Si une requête de priorité a été déjà satisfaite dans le cycle actuel alors aucune nouvelle requête ne sera satisfaite pendant ce cycle.
- La priorité est gérée selon le principe *premier-arrivé-premier-servi*. Dans le cas où deux bus ou plus demandent la priorité simultanément, aucun changement n'est effectué et par conséquent aucune priorité n'est traitée.

Les auteurs ont étudié les bénéfices d'une priorité aux bus sous trois types de gestion des plans de feux : des plans de feux fixes, contrôle adaptatif des durées des phases seul et puis avec les décalages entre les carrefours adjacents. Pour chaque type de gestion des feux, trois types de gestion de la priorité sont considérées. La première consiste simplement à n'accorder aucune priorité. La seconde accorde la priorité qu'aux bus-navette. La troisième accorde la priorité pour tous les types de bus : navettes et lignes régulières.

Pour chaque scénario, les auteurs ont relevé des résultats numériques sur les temps de parcours, les retards, les temps d'arrêt, la consommation de carburants et les différentes émissions de gazs toxiques pour les bus des lignes régulières, les navettes et plus généralement le trafic général.

Ces résultats ont permis de dresser les conclusions suivantes :

- Les bus profitent de la priorité peu importe le type de gestion des plans de feux sous lequel le système de priorité est implémenté.

- Pas de bénéfices certains pour les bus en intégrant la priorité aux bus dans une stratégie de régulation adaptative du trafic.
- Le trafic général est pénalisé par le système de priorité aux bus quelque soit la gestion des plans de feux. Par contre, ces pertes sont réduites grâce aux stratégies de régulation adaptative.
- Les avantages pour les bus dépendent des caractéristiques des flux sur le réseau et aussi sur le type de stratégie de régulation.

Feng et al. ([34]) ont étudié l'intégration d'une priorité aux bus dans le système de régulation du trafic *SCOOT*. L'étude en simulation est effectuée sous l'environnement *VISSIM* ([33]). Le réseau de test est situé dans la ville de Salt Lake City et comporte neuf carrefours sur un axe est-ouest. Le réseau est traversé par sept lignes de bus : cinq sur des axes sud-nord, une sur l'axe est-ouest et une dernière entrant par le nord et sortant par l'ouest. Une étude en simulation est effectuée pour différentes valeurs de fréquences des bus allant de 10 bus par heure à 40 bus par heure.

Les auteurs ont comparé trois stratégies de gestion du trafic. La première est propre à l'environnement *VISSIM* et appelée *OT*. Elle est une stratégie adaptative du trafic. La deuxième est appelée *SWOB* et elle consiste tout simplement au système *SCOOT* sans un système de priorité aux bus. La dernière est *SWB* et elle consiste au système *SCOOT* avec un système de priorité aux bus. La priorité aux bus dans *SCOOT* consiste à l'extension du vert si le bus arrive juste après la fin d'une phase de vert et à un rappel du vert ou troncature du rouge si le bus arrive peu avant le début de sa phase du vert. Enfin, tous les bus sont candidats à la priorité qu'ils soient en retard ou en avance sur le tableau horaire. Par contre les auteurs ne précisent pas comment sont gérés les conflits dans le cas où deux bus sur des arcs conflictuels se présentent au carrefour à peu près en même temps. Ceci laisse penser qu'ils appliquent le principe *premier-arrivé-premier-servi*.

Les premiers résultats montrent que les deux stratégies *SWOB* et *SWB* réduisent les retards de tous les usagers du réseau par rapport à la stratégie *OT*. En comparaison avec *SWOB*, *SWB* permet de réduire les retards des bus de 7.4 s mais d'accroître les retards personnels des autres usagers de 1.5 s. Sur l'axe est-ouest, la priorité aux bus a permis de gagner 5.2 s pour les bus mais de perdre 1.5 s pour chaque usager des autres véhicules. Sur l'axe sud-nord, la priorité aux bus permet de gagner 11.1 s pour les usagers des bus et de perdre 1.5 s pour les autres véhicules. Les résultats montrent ainsi que l'intégration d'un système de priorité aux bus dans un système de régulation de trafic en temps réel profitent aux usagers des bus avec une petite pénalité pour les autres usagers.

En comparant *SWOB* et *SWB* pour les quatre fréquences testées, il apparait que :

- Pour *SWOB*, les retards personnels des passagers des bus sont de 26.1 s à 27.4 s tandis qu'ils sont de 19.0 s à 19.7 s pour *SWB*.

- Entre *SWOB* et *SWB*, la fréquence des bus n’a pas un grand impact sur les retards des bus.
- L’augmentation de la fréquence du bus fait augmenter les retards des usagers des axes nord-sud.
- Par rapport à *SWOB*, *SWB* fait augmenter de 1.9 s, 2.5 s, 5.0 s et 6.5 s les retards des usagers sur les axes nord-sud, pour les quatre valeurs de fréquence.
- Par rapport à *SWOB*, *SWB* fait diminuer les retards des usagers sur les axes est-ouest.

En 2003, un module de gestion de priorité aux bus a été rajouté à la stratégie *TUC*. La priorité n’est accordée que si la charge du trafic de l’arc où se trouve le véhicule et vers lequel se dirige le véhicule est inférieure à un seuil fixé. A l’instant où un véhicule de transport en commun demande une priorité, le module estime l’instant d’arrivée du véhicule sur la ligne de feu en se basant sur un calcul simple de son temps parcours τ_a donné par l’équation suivante :

$$\tau_a = \max(\tau_a^f; \frac{X_a}{S_a})$$

Le temps de parcours est égale au temps de parcours τ_a^f en conditions de trafic fluide. Ce paramètre est calculé grâce à des données historiques. Si le trafic n’est pas fluide, le temps de parcours est égal au temps nécessaire pour que les X_a véhicules particuliers, se trouvant entre le véhicule et la ligne de feu, de se dégager, S_a étant le taux de saturation de l’arc où se trouve le véhicule de transport en commun. Étant donné l’estimation de l’instant d’arrivée du véhicule sur la ligne de feu, le module procède à un prolongement de la phase de vert si le véhicule arrive juste après la fin de celle ci. Dans le cas contraire, le module procède à une anticipation de phase. Le module actuel intégré dans *TUC* ne permet de gérer qu’une demande de priorité par carrefour pour le premier véhicule qui le demande.

Les systèmes de priorité bus ont montré leur efficacité et les gains observés sur les temps de parcours et les temps d’attente au niveau des carrefours plaident pour la généralisation de tels systèmes. Néanmoins les expérimentations en simulation ou sur sites réels ont montré aussi les effets négatifs de ces systèmes sur les autres usagers de la route c’est à dire les voitures. Donner la priorité aux bus au niveau des carrefours a induit des retards importants pour les voitures qui circulent sur des tronçons opposés. Ce constat partagé par toutes les études ont amené à associer les systèmes de priorité aux bus avec des systèmes de régulation du trafic. Si cette association a permis de réduire les effets négatives des systèmes de priorité sur le trafic, il reste cependant plusieurs points sans réponses efficaces :

- La pénalisation du trafic par les systèmes de priorité.
- L’impossibilité de traiter les cas où plus d’un bus demande la priorité sur un même carrefour.
- La non efficacité de ces systèmes dans certains cas de congestion dû au manque d’actions d’anticipation.

- Les actions étant locales, les solutions ne peuvent être globales dans un réseau où tous les carrefours interagissent ensemble.
- Les mesures de priorité sont gérées selon des principes très simples tels que : premier arrivé premier servi ou décision selon que le bus est en retard ou pas et non pas sur la valeur de ce retard.

Afin d’adresser certaines de ces limites, quelques stratégies ont été développées. Elles se basent sur l’optimisation d’un critère de performance pour les véhicules de transport en commun.

1.4.3 Priorité active basée sur l’optimisation

Deux systèmes seront particulièrement décrits dans cette section. Il s’agit des stratégies *PRODYN* et *CRO-NOS*.

PRODYN ([38]) est un système développé à Toulouse en 1980 par le *CERT*. Il permet de commander les feux au niveau des carrefours afin de minimiser les retards des véhicules particuliers. Au départ, *PRODYN* agit seulement au niveau des carrefours isolés pour réguler juste le trafic des véhicules particuliers. Afin de tenir compte des interactions entre les carrefours adjacents, deux nouvelles versions ont été proposées. L’une s’appuie sur une architecture hiérarchisée et l’autre sur une communication entre les carrefours. Étant donné un carrefour isolé, *PRODYN* a pour but de minimiser un critère J sur un horizon K généralement de 75 secondes divisés en des pas de temps de 5 secondes.

$$J = \sum_{k=1}^K \left(\sum_{j=1}^{N_t} X_k^j \right) + F((X_K^j)_{j=1..N_t})$$

N_t est le nombre de tronçons et F une fonction qui évalue le retard occasionné par tous les véhicules présents à la fin de l’horizon K .

Les longueurs de file d’attente sont modélisées en fonction des durées de vert écoulées depuis la dernière commutation, des départs d’un tronçon et des taux de mouvements tournants à gauche. Dans le cas d’une coordination entre les carrefours, une dernière variable est modélisée qui est l’arrivée des véhicules sur une ligne de feu d’un carrefour. Pour résoudre ce problème d’optimisation dynamique, *PRODYN* utilise la programmation dynamique.

Cette version de base ne traite les véhicules de transport en commun que comme des véhicules ordinaires. Deux améliorations ont été introduites afin de proposer des actions pour la priorité aux véhicules de transport en commun. La première idée consiste à représenter un véhicule de transport en commun comme un véhicule particulier avec un poids de pondération plus important. Le processus de programmation dynamique reste le même. La deuxième idée considère les transports publics comme un mode de transport à part avec ses propres caractéristiques. Cette version a été introduite par Henry et Fargès en 1994. Chaque fois qu’un bus est détecté à l’approche d’un carrefour, un nouveau critère est ajouté au critère des véhicules

particuliers ([11])). Le nouveau critère exprime le retard du bus qui dépend de deux variables. La première est le temps nécessaire pour le bus pour atteindre, depuis sa position, la ligne du feu, y compris les stations commerciales. La seconde est la longueur de file d'attente devant le bus.

Pour chaque véhicule de transport en commun, le critère est pondéré par un paramètre qui peut être fixé ou géré de façon dynamique selon que le bus est en retard ou pas, selon l'état général du trafic ou selon un quelconque critère. Pour la régulation d'une zone de carrefours, l'information sur les temps estimés d'arrivée des bus sur les autres carrefours est ajoutée afin de l'intégrer dans l'optimisation sur le prochain pas de temps.

CRONOS est une stratégie développée en 1990 à l'INRETS par Boillot et al. pour la régulation du trafic sur une zone comportant quelques carrefours. Elle a été enrichie en 2002 par un module de priorité pour les bus.

La régulation du trafic général dans *CRONOS* s'appuie sur la même philosophie que les systèmes *PRODYN* ou *OPAC*. L'objectif est de déterminer les instants de commutation entre les phases de façon à minimiser un critère C sur un horizon d'optimisation K . Le critère C s'écrit de la manière suivante :

$$C = \sum_{k=1}^K \sum_{i=1}^{N_c} \left(\sum_{j=1}^{N_{te}^i} l_{k,j}^i + \sum_{h=1}^{N_{int}^i} l_{k,h}^{i,int} \right)$$

où N_c le nombre de carrefours, N_{te}^i le nombre de tronçon d'entrée du carrefour i , N_{int}^i le nombre de zones de stockage internes du carrefour i et $l_{k,j}^i$, $l_{k,h}^{i,int}$ sont respectivement le nombre de véhicules arrêtés dans le tronçon i et à l'intérieur de la zone de stockage h du carrefour i au pas de temps k . Ce problème d'optimisation est résolu par l'algorithme de Box, une heuristique qui donne une solution approchée.

Pour la priorité accordée aux véhicules de transport en commun, l'objectif est de minimiser les temps d'attente des bus. Cet objectif peut être modifié selon que le bus est en avance ou en retard par rapport à un horaire théorique pré-établi. Le système de priorité aux bus est totalement intégré dans le processus d'optimisation et composé de trois modules. Le premier permet de prédire la trajectoire d'un bus à partir de la position dans laquelle il a été détecté. La prédiction de cette trajectoire dépend d'éventuels arrêts commerciaux et des longueurs de file d'attente réajustés à chaque pas de temps. Le deuxième module redéfinit les contraintes sur les feux afin d'assurer le feu vert à l'arrivée du bus sur la ligne. Les nouvelles contraintes ne sont limitées que par les contraintes de sécurité. Le troisième module détermine les instants de commutation qui permettent de minimiser le critère pour le trafic général et celui pour les bus en respectant les contraintes imposées par le deuxième module. Il faut noter que ce deuxième module, dans sa version actuelle, ne peut traiter qu'un seul bus par carrefour.

D'autres stratégies existent dans le même esprit que *CRONOS* ou *PRODYN* telles que *OPAC*, *UTOPIA*, etc...

Ces stratégies décrites ci-dessus considèrent les véhicules de transport en commun comme un mode propre avec ses caractéristiques propres bien différentes de celles des véhicules particuliers. Elles intègrent aussi la priorité aux transports en commun dans le même processus d'optimisation que les véhicules particuliers, chaque stratégie à sa manière. Néanmoins ces avancées tangibles ne peuvent masquer encore quelques limites :

- La vision locale ou semi-locale de la régulation : un seul carrefour pour *PRODYN* et quelques carrefours pour *CRONOS*.
- *CRONOS* dans sa version actuelle ne permet pas de traiter plus d'un bus par carrefour.

1.5 Les systèmes de régulation mixte : trafic général et priorité aux transports en commun

Nous avons présenté dans les sections précédentes deux types d'approches qui gèrent, simultanément, les véhicules de transport en commun et les véhicules particuliers :

- Les systèmes développés initialement pour réguler le trafic général et qui ont été étendus, par la suite, pour gérer la priorité aux véhicules de transport en commun au niveau des carrefours (sous sections 1.4.2.2 et 1.4.3).
- Les systèmes développés uniquement pour donner la priorité aux bus au niveau des carrefours à feux (sous section 1.4.2.1).

La première famille de systèmes ne considère les transports en commun qu'à chaque fois détectés devant une ligne de feu, le système essaye de leur donner la priorité. La deuxième famille est constituée de systèmes plus élaborés pour gérer la priorité des bus mais ne considère pas la régulation du trafic général qui lui-même a des conséquences importantes sur la circulation des véhicules de transport en commun. Enfin les deux approches ne considèrent les véhicules de transport en commun qu'à proximité des carrefours et ne s'intéressent pas aux réseaux plus larges.

Dans cette section, nous allons décrire quelques rares systèmes qui ont pour objectif la régulation globale et simultanée des deux modes de transport : les véhicules particuliers et les transports en commun.

03) ont formulé un problème d'optimisation linéaire qui a pour objectif l'optimisation des durées des phases de vert. La priorité pour les véhicules de transport en commun est gérée de façon indirecte. Quand un véhicule demandant la priorité est détecté sur un arc, ce véhicule est représenté par un coefficient de pondération affecté au poids global de l'arc concerné. Ce poids est utilisé dans le critère d'optimisation. L'originalité de ce travail se vérifie dans l'expression du poids donné aux arcs sur lesquels un véhicule demandant la priorité est détecté. Généralement, ce poids est une constante fixée en fonction des connaissances a priori de la géométrie du réseau et des différentes variables du trafic. Dans ce travail, le poids varie dynamiquement et est fonction de la demande du trafic actuel, de la longueur des files d'attente sur tous les arcs du carrefour et du retard du véhicule sur son plan de marche théorique. Ainsi, pour un arc " m " sur lequel un véhicule est détecté et dont l'arrivée sur la ligne de feu est prévue pour l'instant t pendant la phase p , le poids supplémentaire est donné par :

$$w_m^p(t) = \begin{cases} \frac{X_m^p(t)}{\sum_i X_m^i(t)} + \frac{\lambda_m^p(t)}{\sum_i \lambda_m^i(t)} + b & \text{si le véhicule arrive à } t \\ 0 & \text{sinon} \end{cases}$$

où $X_m^p(t)$ est le nombre de véhicules restés dans l'arc m pour la phase p ,

$\lambda_m^i(t)$ est le flux entrant de l'arc m pour la phase p

et b un facteur compris entre 0 et 1 qui traduit le retard du véhicule et qui est exprimé comme le rapport entre le retard réel du véhicule et un retard maximal fixé par l'opérateur des transports en commun.

L'évolution des flux sortant et entrant et des longueurs des files d'attente de tous les arcs est décrite par des fonctions linéaires. Ainsi, au pas de temps t , les flux entrant et sortant sont déterminés, ce qui permet de calculer les variables $X_m^p(t)$ et par conséquent les poids des arcs traversés par les véhicules de transport en commun. L'algorithme détermine ensuite les arcs critiques pour toutes les phases et détermine les flux sortant optimaux qui permettent alors de déterminer les durées des phases optimales. La dernière étape consiste à comparer les plans de feux ainsi déterminés avec les plans de feux de départ. S'il n'y a pas un appel de priorité, les plans de feu de départ sont gardés. Sinon l'algorithme change les paramètres du plan de feu par les nouveaux paramètres déterminés par optimisation.

Cette méthode, bien qu'ayant pour objectifs l'optimisation du trafic général et la circulation des véhicules de transport en commun, ne permet de changer les plans de feux que lorsque un véhicule réclame une priorité, dans le cas contraire les plans de feux initiaux sont gardés ce qui limite les bénéfices sur le trafic général en dehors des périodes de demande de priorité et aussi sur la circulation des transports en commun puisqu'aucune action d'anticipation n'est prise. Aussi, l'approche de la méthode et les tests de simulation effectués laissent penser que cette stratégie n'est applicable que pour des carrefours isolés.

Une stratégie qui permet de considérer plusieurs carrefours a été développée à l'INRETS à partir de 2002. Il s'agit de *NeTPrior* (Network Transit Priority ([54])). Cette stratégie détermine les durées optimales des phases de vert afin de minimiser un critère qui fait intervenir le trafic général et la circulation des véhicules de transport en commun.

Le modèle utilisé pour décrire le trafic général est le même que celui de la stratégie *TUC*. Ainsi, il décrit linéairement l'évolution du nombre de véhicules sur chaque arc du réseau cycle par cycle en fonction des durées des phases de vert de tous les carrefours du réseau. Pour les véhicules de transport en commun, le modèle est très simple. Connaissant l'itinéraire de chaque ligne, l'équation de progression de la ligne b_i est donnée par l'équation suivante :

$$x_a^{b_i}(k) = x_{a'}^{b_i}(k - \tau_{a',a}^{b_i})$$

où $x_a^{b_i}(k)$ est le nombre de véhicules de la ligne b_i dans l'arc a au cycle k et $\tau_{a',a}^{b_i}$ est le temps de parcours des véhicules pour aller de l'arc a' à l'arc a .

Pour des raisons évidentes sur le pas de temps égal à la durée du cycle C , $\tau_{a',a}^{b_i}$ est donné par

$$\tau_{a',a}^{b_i} = \begin{cases} \times C & \text{si l'arc a contient une station pour la ligne } b_i \\ C & \text{sinon} \end{cases}$$

NeTPrior utilise comme *TUC* la commande linéaire quadratique mais avec un vecteur d'état contenant le nombre de bus de chaque ligne traversant le réseau. Dans *TUC*, le critère quadratique ne mesure que le nombre total de tous les véhicules sur tous les arcs avec des poids donnés par une matrice fixé par l'utilisateur. Dans *NeTPrior*, le critère est un peu différent dans la mesure où il fait intervenir le nombre total des véhicules sur tous les arcs mais avec des poids qui varie dynamiquement en fonction du nombre de véhicules de transport en commun présent sur les arcs. Par exemple un arc où est présent deux bus aura un poids plus important, dans le critère d'optimisation, qu'un arc où est présent un seul bus. La priorité donnée aux véhicules de transport en commun n'est pas donnée directement mais en minimisant les files d'attente sur les arcs où sont les véhicules et au moment où ils sont.

NeTPrior a été testée en simulation sur le logiciel *Dynasim* pour un réseau situé dans le sud de la ville de Paris. Les résultats montrent une amélioration des conditions de trafic en général et plus particulièrement sur les arcs traversés par les bus.

NeTPrior est une stratégie globale qui agit sur de très larges réseaux urbains et qui tient compte de la progression des véhicules de transport en commun au même titre que le trafic général. Cependant, le modèle linéaire utilisé pour les véhicules de transport en commun ne traduit que sommairement le comportement réel des véhicules dans un réseau réel et avec des conditions réelles du trafic.

P.A. Duerr ([26]) a proposé une approche intégrale appelée *DARVIN* pour garantir une priorité pour les véhicules de transport en commun au niveau des carrefours à feux sans pénaliser le trafic général.

Le système *DARVIN* a été développé pour répondre aux objectifs suivants :

- Réduire les temps de parcours des transports en commun.
- Augmenter l'adhérence à la table horaire théorique.
- Minimiser l'impact de la priorité aux transports en commun sur les autres véhicules.

Les deux premiers objectifs sont un peu contradictoires dans le sens où la réduction des temps de parcours peut avoir comme conséquence la réduction de l'adhérence à la table horaire théorique.

Le système *DARVIN* se base sur un modèle événementiel pour la progression des véhicules de transport en commun qui a les caractéristiques suivantes :

- Il représente les véhicules individuellement dans un niveau microscopique.

- Il modélise les interactions entre les véhicules et le reste du trafic.
- Il ne nécessite pas de lourds calculs ce qui rend la simulation rapide.

L'objectif de ce système est la détermination des instants de début et les durées des phases de vert des carrefours du réseau qui minimisent le critère suivant :

$$F(b_{j,k}, e_{j,k}) = \sum_i \sum_j (d_{i,j} + \lambda \cdot s_{i,j}) \cdot \gamma_i$$

où $d_{i,j}$ représente le retard du véhicule i dans le carrefour j et $s_{i,j}$ une variable booléenne égale à 1 s'il y a un arrêt dans la file d'attente et 0 sinon. λ et γ_i sont des coefficients de pondération. Enfin $b_{j,k}$ et $e_{j,k}$ représentent le début et la fin de la phase de vert pour l'arc du carrefour j au cycle k (hypothèse d'un plan de feu à deux phases).

Selon le modèle, le retard $d_{i,j}$ du véhicule i au niveau du carrefour j dépend de l'instant d'arrivée du véhicule sur le carrefour, des paramètres du plan de feu et du nombre de véhicules particuliers en file d'attente dans l'arc traversé par le véhicule de transport en commun.

Pour résoudre ce problème d'optimisation, l'auteur utilise la méthode des algorithmes génétiques. Aucune mention de la manière dont le système *DARVIN* mesure et limite les conséquences du système sur le trafic général qui figure comme un objectif principal.

Le système *DARVIN* est, à notre connaissance, le seul système qui donne la priorité aux transports en commun sur un très large réseau urbain avec une modélisation plus fine de leur progression. Cependant les détails du modèle ne sont pas publiés et selon l'auteur, il s'agit d'un modèle très simplifié pour ne pas alourdir les calculs. Au niveau de la régulation et contrairement aux objectifs initiaux, le système *DARVIN* ne permet pas de maximiser l'adhérence à une table horaire théorique mais plutôt de donner une priorité inconditionnelle aux véhicules de transport en commun. Enfin, aucun critère mathématique n'est mentionné pour mesurer les conséquences sur le trafic général.

1.6 Conclusion

Dans ce chapitre, nous avons passé en revue les méthodes de régulation du trafic urbain multimodale. Nous nous sommes focalisés sur la régulation via les feux de signalisation. Nous avons décrits en premier lieu les stratégies qui ont pour objectif la régulation du seul mode de transport des véhicules particuliers et nous avons distingué les stratégies avec notion de cycle de celles qui ne l'utilisent pas. De toutes ces stratégies, seule le système *TUC* peut effectuer la régulation, en temps réel, sur de très larges réseaux urbains avec une architecture centralisée. Cependant, le système *TUC* n'est efficace que pour des conditions saturées du trafic.

En second lieu, nous avons décrit des systèmes qui donnent la priorité aux véhicules de transport en commun.

Nous avons divisé ces systèmes en deux familles selon la façon de donner la priorité. Ainsi, nous avons décrit en premier les stratégies à base de règles et en second les stratégies basées sur l’optimisation mathématique. Des gains pour les deux types de stratégies ont été observés par plusieurs études en simulation et à travers de déploiement sur sites réels. Certaines stratégies mesurent les conséquences négatives sur le reste du trafic et en tiennent compte dans la prise de décision sur la priorité aux véhicules de transport en commun. Cependant, nous observons dans toutes les stratégies une pénalisation non négligeable et parfois très importante du trafic des véhicules particuliers.

Dans la troisième section, nous avons présenté les stratégies qui avaient pour but la régulation du trafic général et la gestion de priorité pour les véhicules de transport en commun. Nous avons décrit en particulier deux systèmes : *NetPrior* développée à l’INRETS et le système *DARVIN*. Le système *DARVIN* est le système le plus élaboré pour la gestion de la priorité aux bus. Cependant, la régulation du trafic général n’est pas explicite. En outre, l’objectif initial avancé par l’auteur qui est de maximiser l’adhérence est mis en second plan à la faveur de l’objectif simple de donner la priorité inconditionnelle aux bus. La stratégie *NetPrior* est, en terme d’objectifs, la plus élaborée. Mais pour la régulation du trafic général, elle présente le même inconvénient que la stratégie *TUC*. Pour la régulation des véhicules de transport en commun, l’objectif est de donner la priorité inconditionnelle mais le modèle linéaire actuel est trop simpliste pour reproduire finement les phénomènes réelles.

Dans les chapitres suivants, nous allons décrire la stratégie que nous avons conçu depuis les modèles internes jusqu’à l’architecture de commande et les algorithmes d’optimisation.

Chapitre 2

Analyse et modélisation du trafic des véhicules particuliers

2.1 Introduction

Le trafic routier et plus précisément le trafic routier urbain englobent des phénomènes physiques complexes et parfois difficiles à formaliser. Les modèles les plus élaborés se basent sur une analogie entre la circulation routière et l'écoulement des fluides. C'est ainsi que les modèles du premier ordre et de second ordre sont apparus. En milieu urbain, d'autres modèles plus appropriés décrivent l'évolution des files d'attente au niveau des carrefours à feux. La modélisation du trafic nécessite une représentation mathématique de toutes les variables impliquées que ce soit les variables purement propres au trafic comme la vitesse, longueur de file d'attente, densité ou les variables des feux de signalisation comme la durée du cycle, les durées des phases ou encore les variables géométriques comme les localisations des carrefours, le nombre de ligne, la localisation des stations, etc...

Dans ce chapitre, nous nous intéressons à la représentation des variables du trafic en milieu urbain et à la description du modèle des véhicules particuliers que nous avons développé. Dans un premier lieu, nous décrivons les variables du trafic, leurs représentations mathématiques et les relations qui peuvent exister entre elles. Nous nous intéressons en particulier à une représentation simple des carrefours complexes. Dans un deuxième temps, nous présentons le modèle développé pour décrire l'évolution des files d'attente dans un réseau urbain. Ce modèle se base sur le modèle *store-and-forward* que nous décrivons aussi. Nous présentons enfin quelques résultats de simulation sur le modèle développé.

2.2 Représentation mathématique des variables du trafic

2.2.1 réseau routier urbain

Un réseau routier urbain est composé de carrefours et de tronçons de route reliant les carrefours entre eux. Il comprend aussi des entrées de réseau et des sorties. Dans notre cas, le réseau routier urbain est représenté

par un graphe orienté. Les carrefours occupent les sommets et les tronçons occupent les arcs du graphe. Dans le cas général, un tronçon de route peut compter plusieurs voies. Dans le cas où des lignes supportent des véhicules qui vont dans les mêmes directions, elles sont comptées comme un seul arc. Une voie réservée pour un mouvement de tourne à gauche est considérée comme un arc indépendant. Dans la plupart des zones urbaines, certains tronçons de route sont strictement réservés pour les véhicules de transport en commun. Dans ce cas, chaque voie réservée aux bus est considérée comme un arc indépendant de l'arc principal.

Par définition, un carrefour élémentaire est composé de quatre tronçons. Dans la réalité, on rencontre souvent de carrefours géométriquement complexes tels que les rond-points. Dans ce cas, un carrefour réel peut correspondre à plusieurs sommets du graphe associé au réseau urbain. La figure 2.1 donne un exemple de représentation d'un réseau de deux carrefours. Le premier est un carrefour simple comprenant une ligne strictement réservée aux bus et une ligne réservée au tourne à gauche. Le deuxième est un rond-point complexe avec plusieurs feux de signalisation. La partie représentée de ce carrefour est représentée par deux sommets dans le graphe associé.

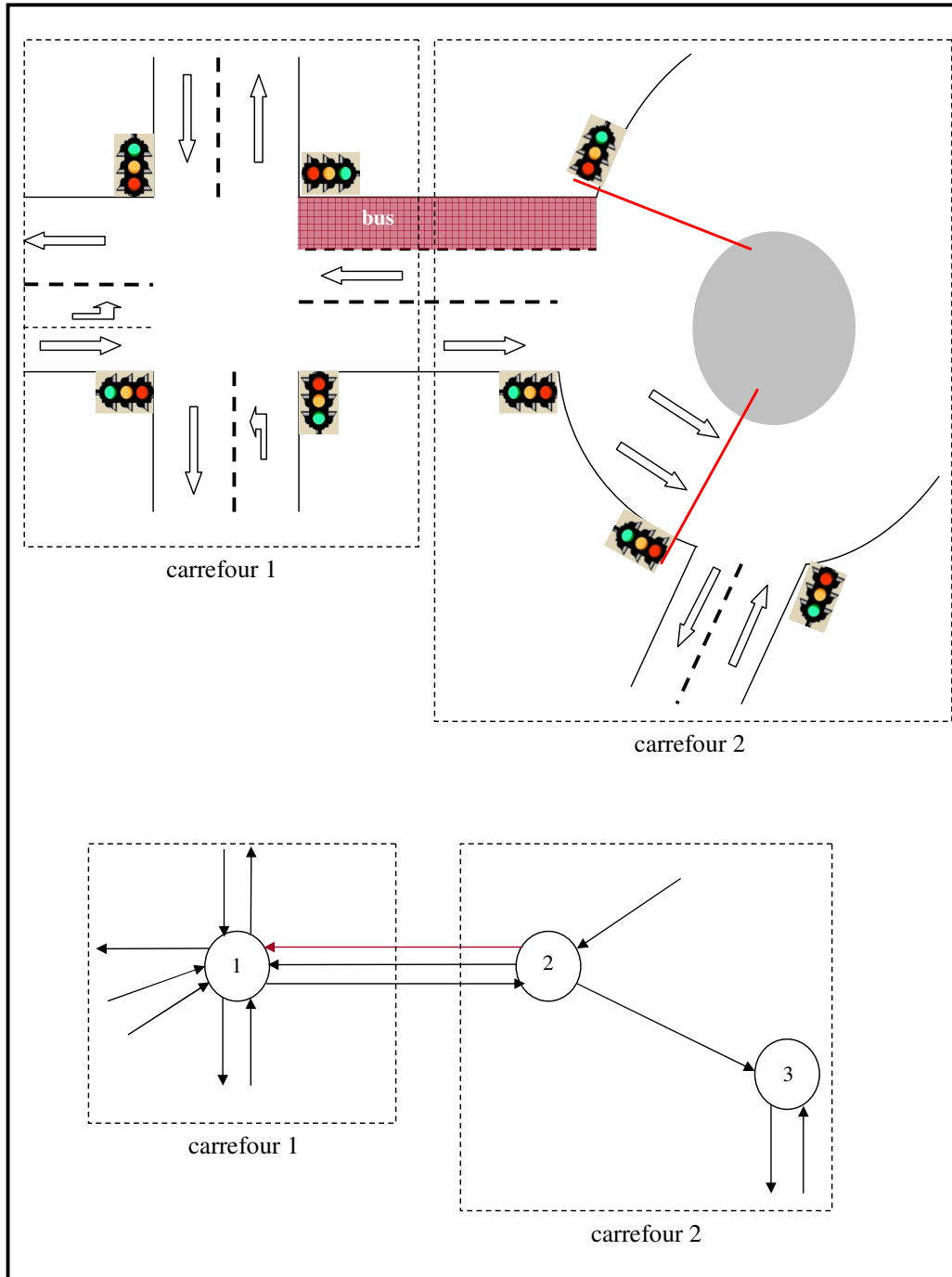


FIGURE 2.1 – Représentation graphique d'un carrefour réel

Les carrefours sont les lieux où un flux de véhicules venant d'une même origine se divise en plusieurs flux

vers plusieurs destinations suivant des coefficients appelés les taux de mouvements tournants. Ils sont définis entre deux arcs d'un carrefour. Étant donné un arc entrant " a " et un arc sortant " b ", le taux de mouvement tournant entre ces deux arcs est défini comme le rapport entre le nombre de véhicules allant de " a " vers " b " et le nombre total des véhicules venant de " a " sur la même période de temps. Les taux de mouvements tournants varient au cours du temps selon les choix d'itinéraire des usagers. Néanmoins, en l'absence de décisions extérieures correctrices importantes sur le trafic, on considère que ces taux varient très peu sur une même période. Ces périodes et les valeurs numériques des taux de mouvements tournants sont déterminées par les données statistiques collectées dans le passé. Pour avoir plus de précision, des algorithmes peuvent être utilisés pour adapter en temps réel ces valeurs en se basant sur des mesures en ligne collectées à chaque pas de temps fixé par les exploitants du réseau urbain. Afin de caractériser complètement les flux de véhicules dans un réseau urbain, il est nécessaire de connaître les flux entrants dans le réseau à chaque entrée. Comme pour les taux de mouvements tournants, les valeurs des flux d'entrée sont déterminées de manière statistique utilisant les données historiques. Les processus d'arrivée de véhicules dans l'ingénierie du trafic sont souvent considérés comme aléatoires et suivent une loi de poisson. Concernant les véhicules de transport en commun, les notions de taux de mouvements tournants et de valeurs de flux n'existent pas. Ces notions sont remplacées par les itinéraires d'une ligne de transport en commun et la fréquence de passage donnée par son tableau de marche. L'itinéraire d'une ligne dans un réseau urbain est défini comme une succession d'arcs traversés par le véhicule. La fréquence de passage d'une ligne peut être définie indifféremment comme le nombre de bus d'une même ligne qui passe par unité de temps ou l'intervalle qui sépare le passage de deux véhicules successifs de la même ligne. Enfin la trajectoire d'un véhicule de transport en commun dans un réseau est définie comme l'évolution de sa position en fonction du temps. La position est référencée par rapport à son entrée du réseau.

Chaque carrefour du réseau est contrôlé par des feux de signalisation. Dans la section suivante, nous nous intéresserons à la représentation mathématique des feux de signalisation des carrefours.

2.2.2 Les feux de signalisation

Au niveau des carrefours, les feux de signalisation ont une grande importance sur le trafic. D'une part, ils ont un rôle de sécurité en empêchant des flux conflictuels de traverser simultanément le carrefour et en garantissant un passage sécurisé pour les piétons. D'autre part, ils ont un rôle de régulation en bloquant l'accès au carrefour quand ce dernier n'est pas dégagé. Pour des raisons évidentes de synchronisation et gestion de conflits entre les différents flux, chaque carrefour est contrôlé par un contrôleur unique qui détermine le plan de feu du carrefour. Un plan de feu est caractérisé par les éléments suivants :

- Diagramme de phase : une phase est une période de temps pendant laquelle des mouvements compatibles ont accès à l'intérieur du carrefour. Un diagramme de phase est une succession de phases qui permet l'accès à tous les mouvements.
- Durées des différentes phases.
- Temps de dégagement appelé également rouge intégral : temps insérés entre deux phases pour des raisons

de sécurité.

- Cycle : la durée totale de toutes les phases et des temps de dégagement.
- Décalage : la durée qui sépare deux événements de référence de deux carrefours successifs.

Dans certaines approches pour la régulation du trafic urbain, la notion de cycle disparaît. Nous parlons alors de stratégies acycliques.

Dans le cadre de cette thèse, nous considérons une durée de cycle fixe notée C commune à tous les carrefours de la zone de régulation concernée. Nous supposons que le diagramme de phase est fixe et connu. Les temps de dégagement sont des constantes qui dépendent de la géométrie du carrefour et des caractéristiques des flux qui le traversent. Pendant les temps de dégagement, tous les feux du carrefour sont au rouge, nous parlons de rouge intégral. Nous supposons qu'elles sont nulles. Cela ne change absolument rien puisque dans le cas de valeurs non nulles, la valeur du cycle sera prise égale à la vraie valeur C à laquelle nous soustrayons la somme des temps de dégagement du carrefour. Les variables concernées par la régulation sont les durées des différentes phases et les décalages. En réalité, comme nous le verrons plus tard, les décalages concernés sont seulement des carrefours traversés par des lignes de transport en commun. Ceci est dû au fait que les décalages n'auront aucun effet, dans le modèle, sur l'état du trafic des véhicules particuliers qui est modélisé de façon macroscopique.

Lors de l'établissement d'un plan de phase pour un carrefour donné, il est admis qu'un flux ne peut avoir deux phases de vert disjointes dans le même cycle. Ainsi la figure 2.1 montre une configuration interdite pour un flux donné.

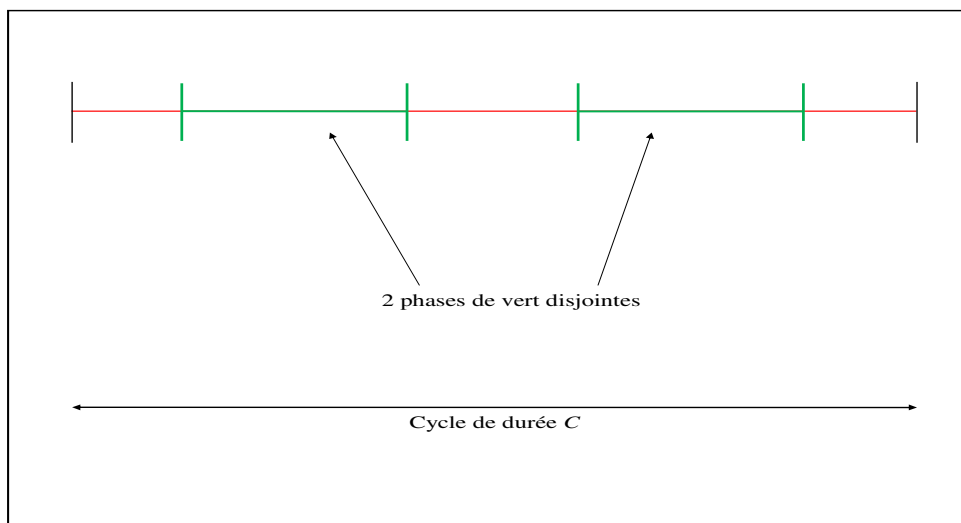


FIGURE 2.2 – Plan de feu interdit : 2 phases de vert disjointes pour le même flux

Par contre, la figure 2.3 montre deux configurations autorisées.

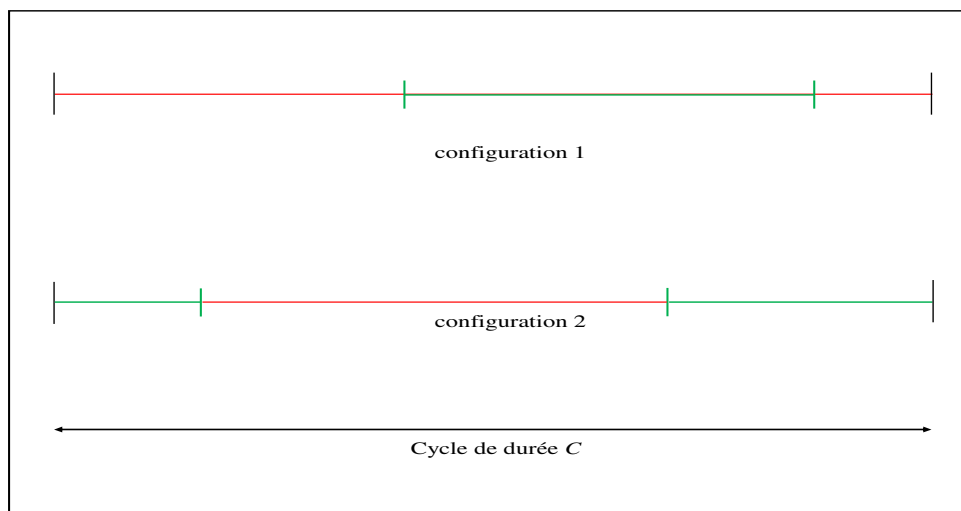


FIGURE 2.3 – Plans de feu autorisés

L'autorisation de la configuration (1) est triviale vu la règle énoncé ci-dessus. La configuration (2) présente deux phases de vert disjointes. La première phase est le reste de la deuxième phase du cycle précédent. La seule vraie phase de ce cycle est la deuxième. Ainsi, la règle des deux phases disjointes est respectée. Une conséquence immédiate est la suivante : "La période de feu vert pour un flux donné est caractérisée par seulement deux variables : l'instant de début du vert et sa durée". Ainsi, un flux noté i est entièrement déterminé, d'un point de vue plan de feu, par Td_i l'instant de début du vert qui lui est alloué et G_i la durée de ce feu vert (voir figure 2.4).

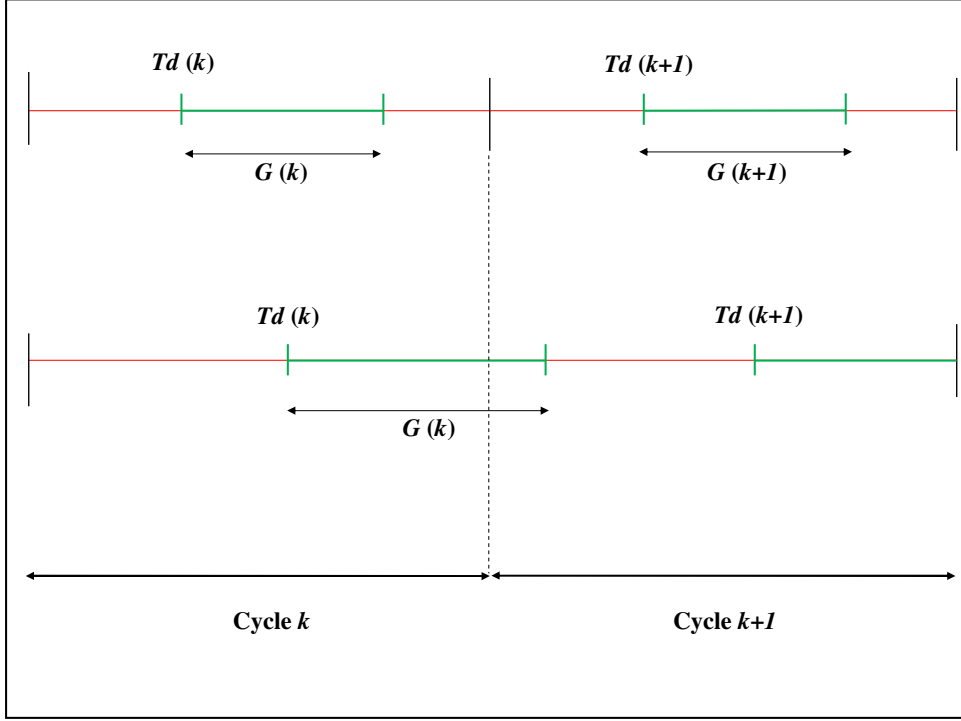


FIGURE 2.4 – Les deux variables des plans de feu pour deux configurations différentes

Dans la configuration (2), il existe une autre manière de voir le "bout de phase" restant du cycle précédent. En effet, si au cycle présent, l'instant de début du vert est Td_i et sa durée est G_i , mathématiquement, la configuration (2) correspond au cas où $Td_i + G_i > C$. C'est à dire que le "bout de phase" qui appartient au cycle précédent peut apparaître comme appartenant au cycle présent. Ceci n'a aucune conséquence sur le trafic réel mais a d'importantes conséquences sur le développement mathématique des modèles décrits ultérieurement. Quand nous nous situons dans un cycle donné, nous ne serons pas obligés de parler systématiquement de variables relatives au cycle précédent mais bien du cycle présent.

Pour récapituler, la phase de vert allouée à un flux i donné est entièrement caractérisée par deux variables qui sont l'instant de début de la phase Td_i et sa durée G_i . Pour la suite, nous parlerons de configuration (1) dans le cas où $Td_i + G_i > C$ et de configuration (2) dans le cas contraire c-à-d quand $Td_i + G_i \leq C$ (figure 2.5).

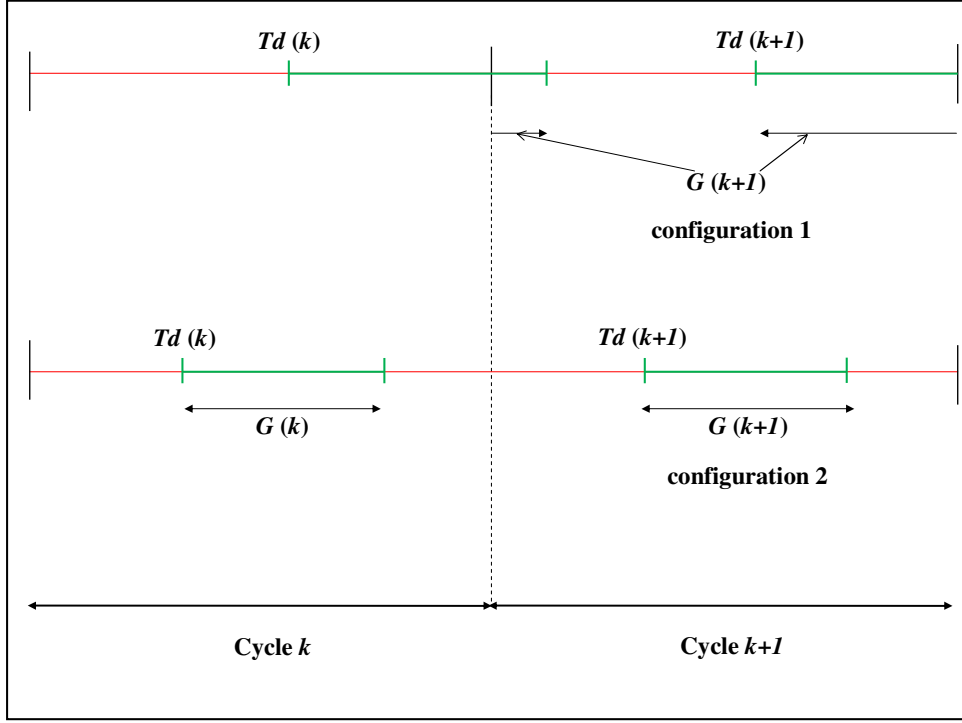


FIGURE 2.5 – Les deux configurations (1) et (2)

La valeur du décalage entre deux carrefours dépend des instants de début des phases de vert des flux concernés par la synchronisation ou l'"onde verte". Dans la suite, nous parlerons d'optimisation des instants de début de phases de vert et de l'optimisation des décalages pour désigner le même processus d'optimisation. Pour un flux i , l'instant de début de sa phase de vert doit être compris entre 0 et la durée du cycle C . Ainsi, la seule contrainte pour Td_i est :

$$0 \leq Td_i \leq C$$

Pour la durée de la phase du flux i , il est évident que G_i doit être comprise entre 0 et C . Pour des raisons de sécurité, G_i doit être supérieure à une valeur minimale G_i^{min} et inférieure à une valeur maximale G_i^{max} . Les deux valeurs G_i^{min} et G_i^{max} sont des constantes communes à tous les flux de tous les carrefours du réseau considéré. Il est fréquent qu'il y ait des contraintes supplémentaires exclusives au flux i qui sont dues essentiellement aux passages piétons. Ainsi nous pouvons avoir la contrainte suivante :

$$G_i^{min} \leq G_i \leq G_i^{max}$$

où $G_i^{min} < G_i^{min}$ et $G_i^{max} < G_i^{max}$. Dans la suite, pour ne pas alourdir les développements futurs, nous omettrons les contraintes G_i^{min} et G_i^{max} et nous considérons que la contrainte

$$G_i^{min} \leq G_i \leq G_i^{max}$$

Les deux contraintes d'inégalité pour les instants de début Td et les durées G des phases de vert, que nous venons de voir, ne sont pas les seules. Pour bien introduire les nouvelles contraintes, nous considérons l'exemple du carrefour élémentaire à 4 branches de la figure 2.6.

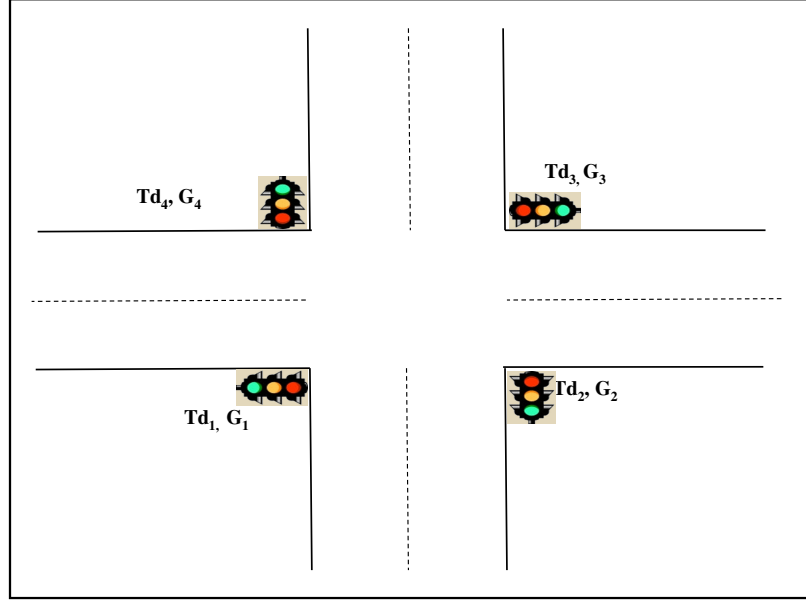


FIGURE 2.6 – Carrefour élémentaire à 4 branches

Partant de l'hypothèse qu'il n'y a pas de phases spéciales pour les tournées à gauche, les flux 1 et 3 sont compatibles et par conséquent ils peuvent traverser le carrefour simultanément. Il est de même pour les flux 2 et 4. Par conséquent, si l'indice i concerne le flux i , nous pouvons déduire que :

$$G_1 = G_3; Td_1 = Td_3$$

$$G_2 = G_4; Td_2 = Td_4$$

En conséquence, il suffit de connaître quatre variables par exemple G_1, Td_1, G_2 et Td_2 pour déduire les quatre autres.

Les flux 1 et 2 sont incompatibles donc ils ne peuvent traverser le carrefour simultanément. Comme les temps de dégagement sont supposés nuls, nous pouvons écrire :

$$G_1 + G_2 = C$$

et donc

$$G_2 = C - G_1$$

Pour l'instant de début de la phase du flux 2, nous pouvons écrire sans difficulté :

$$Td_2 = (Td_1 + G_1) \text{ modulo } C$$

Ainsi, pour caractériser complètement le plan de feu d'un carrefour élémentaire à deux phases, il suffit de connaître deux variables par exemple G_1 et Td_1 . Un résultat très utile et très important pour la suite. Grâce à ces nouvelles contraintes, nous passons d'un système à 8 variables à un autre à seulement 2 variables.

Pour montrer le même processus de réduction des variables sur un carrefour plus complexe, nous considérons le carrefour de la figure 2.7.

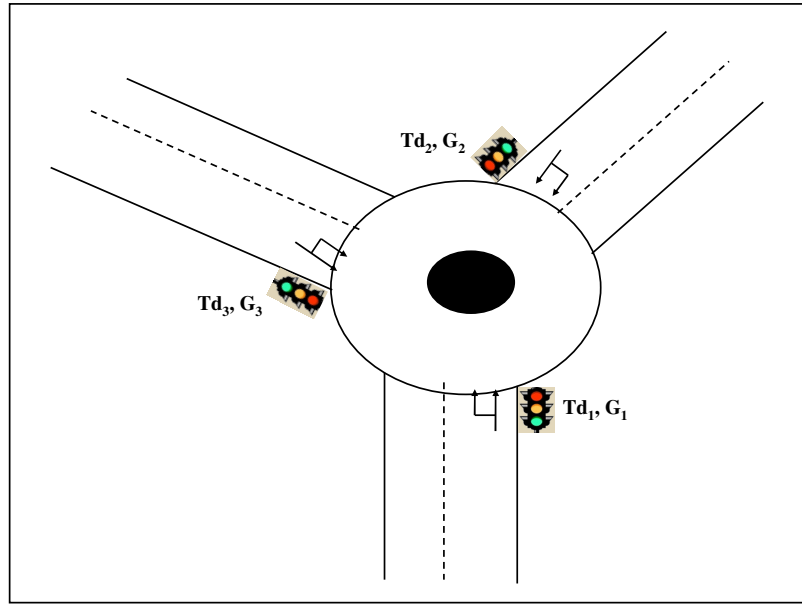


FIGURE 2.7 – Carrefour compliqué à 3 phases

En suivant la même démarche que pour le carrefour élémentaire précédent, nous pouvons écrire les équations suivantes :

$$G_1 + G_2 + G_3 = C; Td_2 = (Td_1 + G_1) \text{ modulo } C$$

$$Td_3 = (Td_2 + G_2) \text{ modulo } C$$

Ainsi, en remplaçant Td_2 par son expression dans la dernière équation, nous obtenons les nouvelles équations suivantes :

$$G_3 = C - G_1 - G_2; Td_2 = (Td_1 + G_1) \text{ modulo } C$$

$$Td_3 = (Td_1 + G_1 + G_2) \text{ modulo } C$$

Il vient, d'après ces dernières équations, que pour caractériser le plan de feu du carrefour de la figure 2.7 à 3 phases, il suffit de connaître 3 variables qui peuvent être dans ce cas : Td_1 , G_1 et G_2 .

Plus généralement, pour un carrefour à n phases, il suffit de connaître n variables pour caractériser totalement le plan de feu du carrefour, la séquence des phases est supposée connue. Il suffit de $n-1$ durées de phase pour $n-1$ phases. La durée de la dernière phase est déduite de la contrainte $\sum_{i=1}^n G_i = C$. Et il est nécessaire aussi de connaître seulement l'instant de début de la première phase Td_1 , les autres instants sont déduits de cet instant et des durées des $n-1$ phases. En effet, l'instant de début de la phase i peut être donné par $Td_i = (Td_{i-1} + G_{i-1}) \text{ modulo } C$. En appliquant la même formule de récurrence pour Td_{i-1} et ainsi de suite, nous obtenons la formule générale de Td_i :

$$Td_i = (Td_1 + \sum_{j=1}^{i-1} G_j) \text{ modulo } C$$

2.2.3 Respect des contraintes

Cette section sera utile pour la chapitre 4 dans la présentation de l'architecture de la stratégie et de l'algorithme d'optimisation.

Dans la section précédente, nous avons montré que, pour un carrefour quelconque et pour caractériser totalement son plan de feu, il suffit de connaître l'instant de début de la première phase et les durées de toutes les phases sauf la dernière qui se déduit facilement des autres. Pour l'instant de début de la première phase, la seule contrainte consiste en ce que sa valeur doit être comprise entre 0 et la durée du cycle C . Pour les durées des différentes phases, les contraintes sont plus compliquées comme nous l'avons déjà vu.

Imaginons un processus (ça sera un algorithme d'optimisation dans notre cas) qui permet de se déplacer dans le domaine de définition d'une variable. Pour un carrefour élémentaire à deux phases, les variables peuvent être l'instant de début d'une phase ou sa durée. Sans tenir compte des contraintes, le domaine de définition initial de ce couple de variables est $\mathbb{R}^2$. Avec la prise en compte des contraintes vues précédemment, le domaine de définition réel du couple (Td, G) devient

$$[0; C] \times [G_{min}; G_{max}]$$

Si un processus permet de se déplacer dans ce domaine avec des mécanismes qui ne nous intéressent pas pour le moment et que nous verrons dans le chapitre 4, il se peut qu'un déplacement fasse sortir l'une et/ou l'autre des deux variables de leur domaine de définition déterminé par les contraintes (voir figure 2.8).

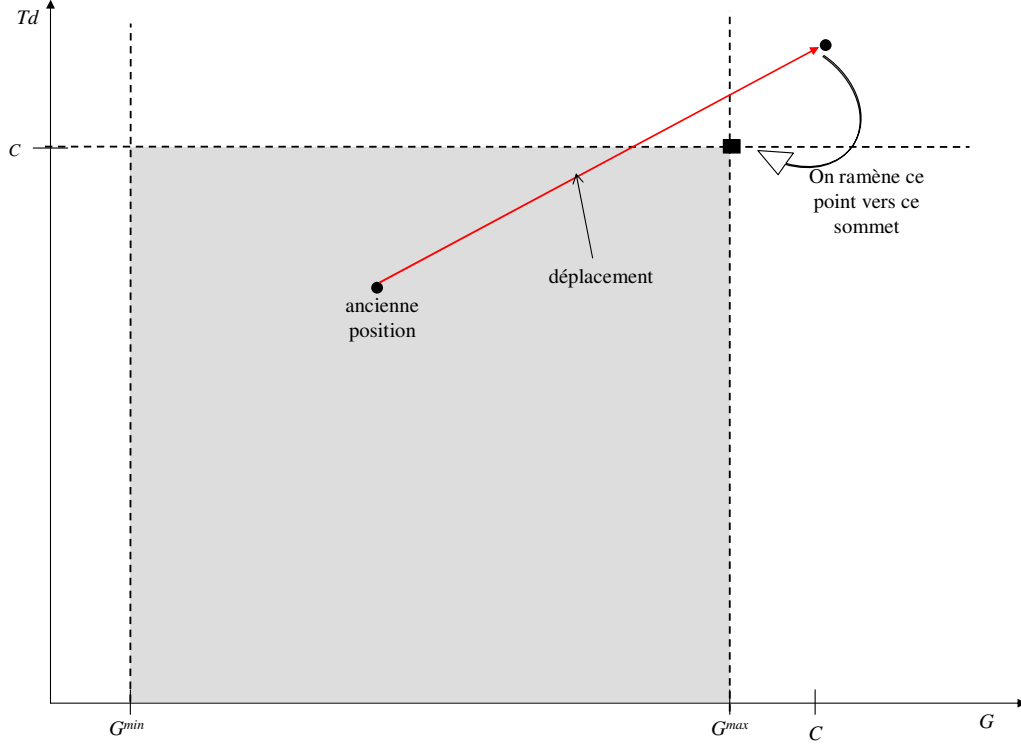


FIGURE 2.8 – Déplacement faisant sortir les variables de leur domaine de définition

Dans l'exemple de la figure 2.8, les deux variables Td et G , du fait d'un déplacement, ont vu leurs valeurs sortir de leur domaine de définition. La solution est très simple et consiste à ramener ce point au point le plus proche du domaine de définition. Ainsi, si Td^n et G^n désignent les nouvelles valeurs de Td et G après le déplacement, nous pouvons écrire :

$$Td = \max(0; \min(Td^n; C))$$

$$G = \max(G^{\min}; \min(G^n; G^{\max}))$$

Nous considérons maintenant l'exemple du carrefour de la figure 2.7 à 3 phases. Nous avons montré qu'il suffit de connaître l'instant de début de la première phase Td_1 et les durées des deux premières phases G_1 et G_2 . La durée de la troisième phase est donnée par l'équation $G_3 = C - G_1 - G_2$. Pour Td_1 , si un déplacement la fait sortir de son domaine de définition, nous n'avons qu'à ramener sa valeur au point le plus proche du domaine de définition. Nous nous intéressons seulement ici aux variables G_1 et G_2 . Le domaine de définition initial du couple (G_1, G_2) est le carré, dans $\mathbb{R}^2$, $[G^{\min}; G^{\max}]^2$.

Cependant, étant donné que $G_3 = C - G_1 - G_2$ et que $G^{\min} \leq G_3 \leq G^{\max}$, il vient alors :

$$C - G^{\max} \leq G_1 + G_2 \leq C - G^{\min}$$

Pour des raisons de symétrie, il est généralement admis que $G^{min} + G^{max} = C$. Par conséquent, la première contrainte $G_1 + G_2 \geq C - G^{max}$ est équivalente à l'inégalité suivante :

$$G_1 + G_2 \geq G^{min}$$

Cette dernière inégalité est toujours vérifiable dans le domaine de définition du couple (G_1, G_2) . Le domaine de définition initial reste alors inchangé.

La deuxième contrainte $G_1 + G_2 \leq C - G^{min}$ est équivalente à $G_2 \leq G^{max} - G_1$. Dans ², cette dernière contrainte coupe l'espace en deux demi-plans, le demi-plan inférieur étant celui qui respecte la contrainte. Ainsi, le domaine de définition final du couple (G_1, G_2) est donné par la figure 2.9.

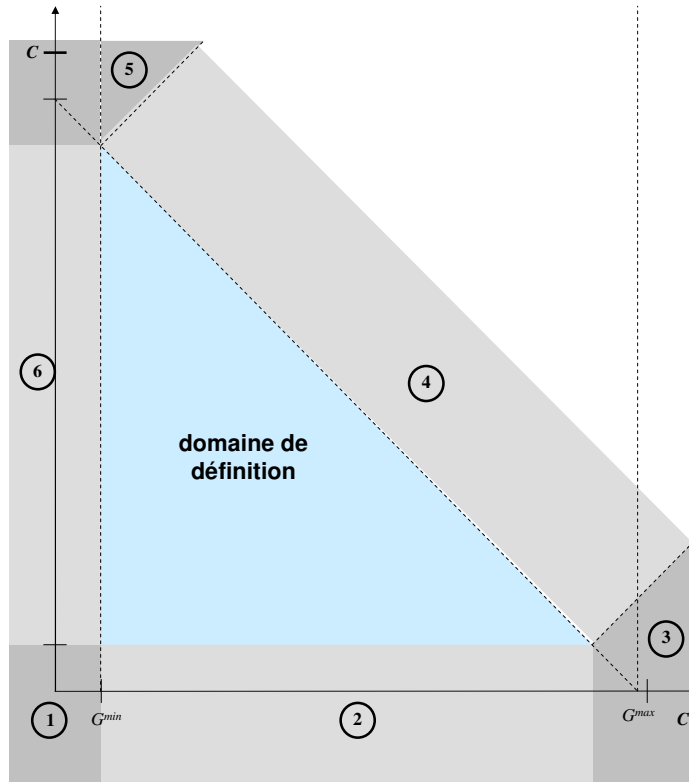


FIGURE 2.9 – Domaine de définition finale du couple (G_1, G_2)

L'espace $\mathbb{R}^2$ est ainsi divisé en 7 régions : l'espace de définition des deux variables et 6 autres régions où le couple (G_1, G_2) ne respecte pas les contraintes. Dans le cas où un déplacement fait sortir ce couple de son

domaine de définition (D_f) vers une autre région, un remplacement de ce point est prévu vers le point le plus proche du domaine de définition. Le remplacement dépend de la zone où le point se retrouve :

- Dans la zone 1, le point est ramené au point A, le point le plus proche du domaine de définition. Ainsi, si $G_1^n < G^{min}$ et $G_2^n < G^{min}$ alors $G_1^n = G_2^n = G^{min}$.
- Dans la zone 2, le point est remplacé vers le point le plus proche du segment $[AB]$. Autrement, si $G^{min} \leq G_1^n \leq G^{max}$ et $G_2^n < G^{min}$ alors nous gardons la valeur de G_1^n et $G_2^n = G^{min}$.
- Dans la zone 3, le point est remplacé au point B. Algébriquement, si $G_1^n > G^{max}$ et $G_2^n \leq G_1^n + 3 \times G^{min} - C$ (contrainte illustrée par la perpendiculaire au segment $[BC]$ en B), alors $G_1^n = C - 2 \times G^{min}$ et $G_2^n = G^{min}$.
- Dans la zone 4, le point est remplacé vers le point le plus proche du segment $[BC]$. Le nouveau point appartient à la zone 4 si $G_2^n \geq G_1^n + 3 \times G^{min} - C$, $G_2^n \leq G_1^n + C - 3 \times G^{min}$ et $G_2^n \geq G^{max} - G_1^n$. Dans ce cas, le remplacement donne les valeurs suivantes :

$$G_1^n = \frac{1}{2} \times (G_1^n - G_2^n + C - G^{min})$$

$$G_2^n = \frac{1}{2} \times (G_2^n - G_1^n + C - G^{min})$$

- Dans la zone 5, le point est remplacé vers le point C. La zone 5 est définie par : $G_2^n > C - 2 \times G^{min}$ et $G_2^n \geq G_1^n + C - 3 \times G^{min}$. Dans ce cas, nous avons les valeurs finales suivantes : $G_1^n = G^{min}$ et $G_2^n = C - 2 \times G^{min}$.
- Dans la zone 6, le point est remplacé vers le point le plus proche du segment $[AC]$. Autrement, si $G_1^n < G^{min}$ et $G^{min} \leq G_2^n \leq G^{max}$ alors nous gardons la valeur de G_2^n et nous avons $G_1^n = G^{min}$.

Plus généralement, nous considérons un carrefour à n phases. Nous avons montré précédemment qu'il suffit de n variables pour caractériser complètement le plan de feu du carrefour : l'instant de début de la première phase et les durées des $n - 1$ premières phases. La durée de la dernière phase se déduit de la contrainte

$$\sum_{i=1}^n G_i = C$$

Pour les $n - 1$ durées des $n - 1$ phases, le domaine de définition initial est l'hypercube, dans l'espace $n-1$, $[G^{min}; G^{max}]^{n-1}$. A partir des contraintes $\sum_{i=1}^n G_i = C$ et $G^{min} \leq G_n \leq G^{max}$, il est très facile de déduire la nouvelle contrainte :

$$C - G^{max} \leq \sum_{i=1}^{n-1} G_i \leq C - G^{min}$$

La contrainte $C - G^{max} \leq \sum_{i=1}^{n-1} G_i$ est toujours respectée puisque $C - G^{max} = G^{min}$ et que pour tout i , $G_i \geq G^{min}$. Compte tenu du fait que $C - G^{min} = G^{max}$, la deuxième contrainte devient $\sum_{i=1}^{n-1} G_i \leq G^{max}$. Géométriquement, cette contrainte signifie que, dans l'hypercube de dimension $n - 1$, les variables doivent

se situer en dessous de l'hyperplan défini par l'équation suivante :

$$\sum_{i=1}^{n-1} G_i = G^{max}$$

Le domaine final de définition est donc la partie inférieure de l'hypercube $[G^{min}; G^{max}]^{n-1}$ délimitée par l'hyperplan dont l'équation a été donnée précédemment.

2.3 Modélisation du trafic urbain

2.3.1 Revue des modèles du trafic

Avant de tenter des actions de régulation du trafic, il est nécessaire de pouvoir décrire mathématiquement l'écoulement du trafic. Les modèles de trafic sont de ce fait une nécessité absolue pour la régulation et le contrôle. Les modèles du trafic sont variés et nous pouvons les classer en 3 grandes familles selon la manière dont est considérée la circulation routière. C'est ainsi que nous distinguons les modèles macroscopiques, les modèles microscopiques et les modèles de file d'attente.

Les modèles macroscopiques découlent par analogie avec l'écoulement des fluides. La circulation des véhicules est assimilée à un écoulement de flux homogènes. Ces modèles se basent sur une équation de conservation du nombre de véhicules sur un tronçon routier. Deux types de modèles sont particulièrement décrits dans la littérature. Il s'agit des modèles du premier ordre et ceux du second ordre. Pour les premiers, le modèle LWR figure comme une référence et est fondé sur l'hypothèse que le trafic est en équilibre dans le sens où il suppose que la vitesse n'est fonction que de la concentration des véhicules. La résolution de ce modèle peut se faire de manière analytique mais dans des cas complexes, il est très difficile d'évaluer exactement les solutions. Dans ce cas, la résolution numérique par discrétisation spatio-temporelle est préférée. Le modèle LWR est utilisé par exemple pour le contrôle d'accès coordonné. L'inconvénient majeur de ces modèles sont la non reproduction des phénomènes hors équilibre comme l'arrêt et le démarrage. Ceci limite l'intérêt de ces modèles pour le milieu urbain où les arrêts et les démarrages sont fréquents. Les modèles du second ordre permettent de reproduire les régimes hors équilibre en rajoutant une seconde équation exprimant l'accélération des flux. La résolution des modèles du second ordre reste difficile et compliquée ce qui limite leur utilité dans les problèmes de contrôle et de régulation en temps réel du trafic.

En parallèle des modèles macroscopiques, une deuxième famille de modèles est développée. Il s'agit des modèles microscopiques. Contrairement aux modèles macroscopiques, les véhicules dans les modèles microscopiques sont considérés individuellement. Plusieurs modèles microscopiques existent comme les modèles d'inter-distance de sécurité, les modèles de poursuite, les modèles à automates cellulaires, les modèles multi-agents, etc... Le haut degré de détail des modèles microscopiques a été à l'origine de leur succès dans la simulation et l'évaluation des stratégies de régulation du trafic urbain. Par contre ceci a rendu ces modèles très peu utilisables dans la conception de ces stratégies de régulation globale et pour de larges réseaux urbains composés de plusieurs carrefours.

Pour bien représenter le trafic urbain dans la conception des stratégies de régulation, une autre approche de modélisation s'est développée. Le trafic n'est plus représenté comme des flux dans les modèles macroscopiques ou des véhicules individuels interagissant ensemble dans les modèles microscopiques mais comme des files d'attente devant des lignes de feu de signalisation au niveau des carrefours. Les modèles de file d'attente en trafic reposent sur les équations de conservation des véhicules et sur des distributions stochastiques reproduisant les phénomènes d'arrivée et de départ. Ces modèles ont été utilisés dans la conception de stratégies de régulation comme *TUC*, *PRODYN*, etc... Dans le trafic urbain, les processus d'arrivée et de départ ne sont pas de nature stochastique et dépendent des états des feux de signalisation qui contrôlent la circulation. Un modèle particulièrement pertinent pour décrire les sorties de véhicules des files d'attente est le modèle *store-and-forward*.

2.3.2 Modèle *store-and-forward*

Le modèle *store-and-forward* a été développé par Gazis et Potts en 1963 ([?]). Il a été à l'origine de plusieurs stratégies et de méthodes pour la régulation et le contrôle du trafic routier urbain. Il permet de décrire de façon simple, les flux dans un réseau urbain composé de plusieurs carrefours contrôlés par des feux de signalisation. L'hypothèse principale concerne l'expression du flux sortant d'un arc. Selon le modèle, le flux de sortie est égal au flux de saturation S_i de l'arc tant que le feu est vert et égal à 0 sinon. Pour un arc " i " contrôlé par un feu de signalisation avec une durée de cycle C et une durée du vert G_i , le flux sortant u_i est donné sur la figure 2.10.

$$u_i = S_i \times \frac{G_i}{C}$$

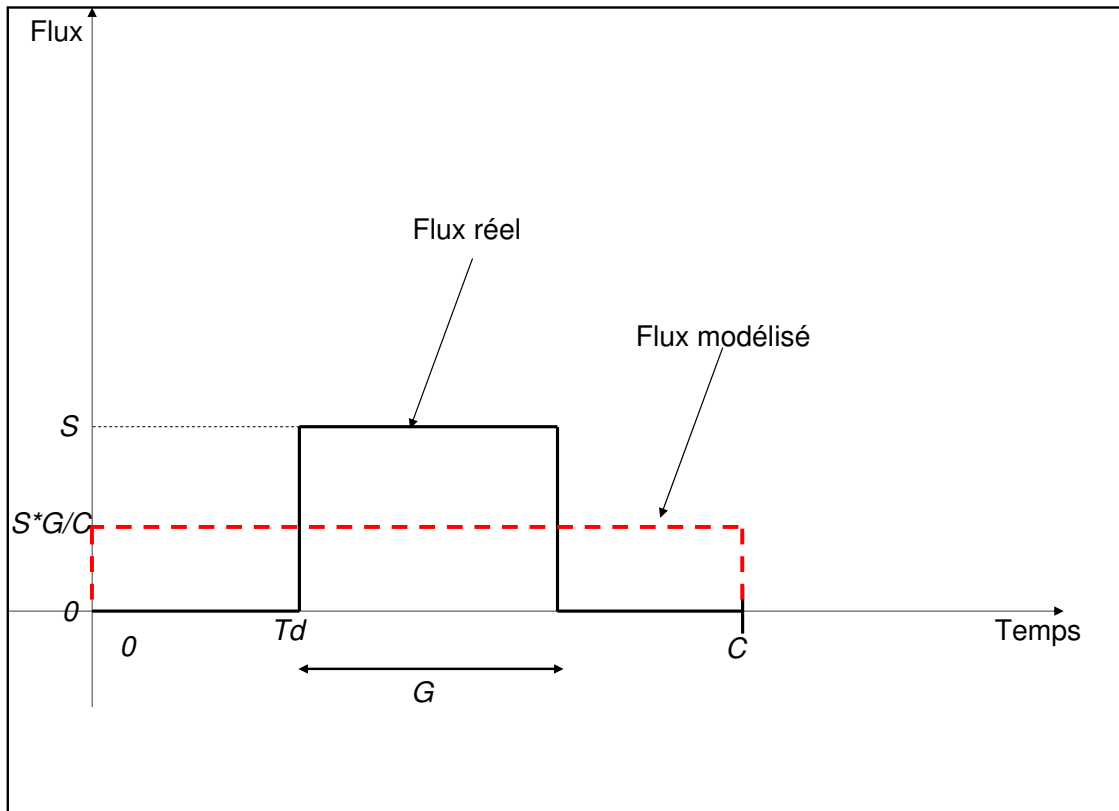


FIGURE 2.10 – flux sortant durant un cycle

Une première conséquence de cette hypothèse réside dans l'obligation de choisir un pas de temps au moins égal à la durée du cycle C . Le modèle *store-and-forward* permet de modéliser les flux de façon très simple et de manière linéaire ce qui permet l'utilisation des méthodes de programmation linéaire ou de commande linéaire quadratique pour l'optimisation et le contrôle du trafic. Cependant, le modèle ne permet pas de modéliser finement les périodes de transition lors des commutations entre les phases de vert et de rouge. En plus, il suppose qu'il y a toujours assez de véhicules pour que le flux soit égal au flux de saturation durant toute la durée de la phase de vert. Ceci est particulièrement vrai quand les carrefours sont sursaturés. Il l'est moins quand le trafic est fluide. Dans la suite nous présentons l'amélioration que nous avons introduit sur ce modèle pour le développement du modèle de trafic des véhicules particuliers.

2.4 Modèle d'écoulement du trafic

Le modèle a pour objectif de décrire l'évolution temporelle du nombre de véhicules dans tous les arcs du réseau en fonction des durées des phases de vert de tous les feux de signalisation. Le pas de temps discret est la durée du cycle C . Ce choix est une conséquence du modèle *store-and-forward*.

2.4.1 Équation de conservation

Soit un arc " i " reliant le carrefour " M " au carrefour " N " (figure 2.11).

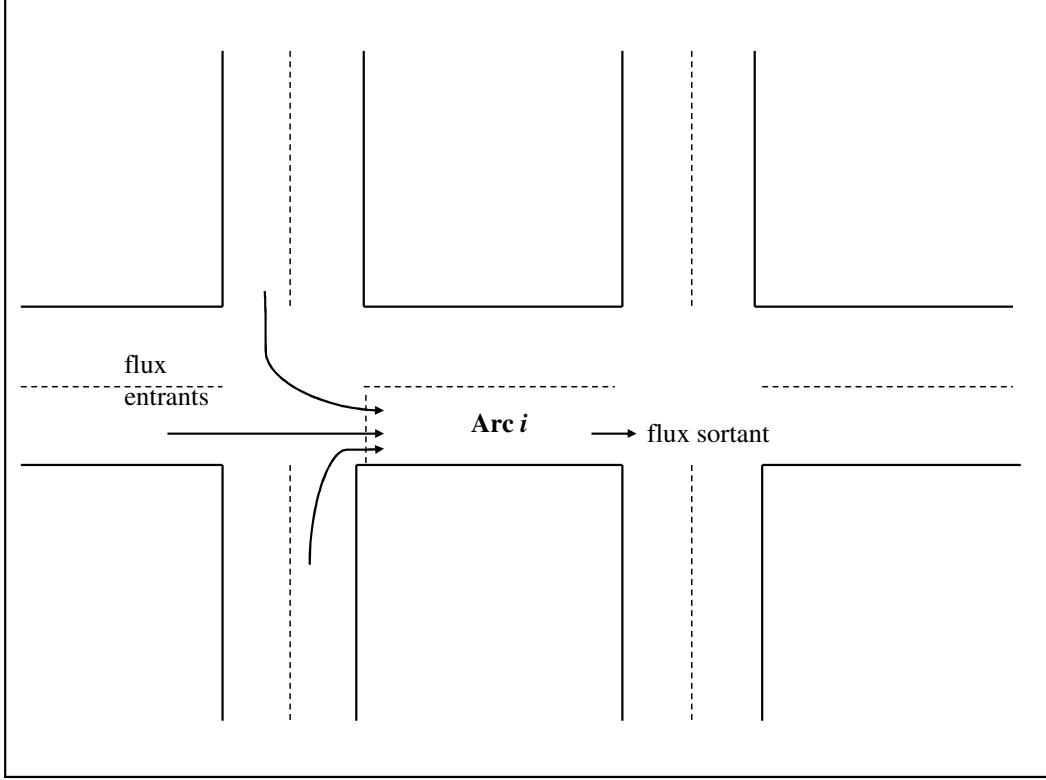


FIGURE 2.11 – nombre de véhicule dans un arc

Si $X_i(k)$ désigne le nombre de véhicules présent dans l'arc " i " au cycle k , alors l'équation de conservation du nombre de véhicules dans l'arc " i " peut s'écrire ainsi :

$$X_i(k+1) = X_i(k) + C \times (u_i(k) - q_i(k))$$

où $u_i(k)$ et $q_i(k)$ sont respectivement les débits entrant et sortant de l'arc " i " durant le cycle k .

Si l'arc " i " est un arc d'entrée du réseau alors le flux entrant dans cet arc s'écrit comme suit :

$$u_i(k) = E_j(k)$$

où $E_j(k)$ est le flux entrant à l'entrée " j " du réseau au cycle k . Les valeurs de E_j sont supposées connues.

Si l'arc est un arc intérieur, le flux entrant est formé par les proportions des flux sortant des arcs en amont.

Ainsi, si In_i désigne l'ensemble des arcs en amont de l'arc " i " alors $u_i(k)$ peut être donné par

$$u_i(k) = \sum_{j \in In_i} \tau_{j,i} \times q_j(k)$$

où $\tau_{j,i}$ est le taux de mouvement tournant de l'arc " j " vers l'arc " i ". Les valeurs des taux de mouvement tournant sont supposées connues et sont données par la matrice *origine-destination* du réseau.

Recombinant les équations précédentes, l'évolution du nombre de véhicules dans un arc " i " peut être exprimée uniquement en fonction des flux sortant de l'arc et des arcs en amont. D'où l'équation d'évolution suivante :

$$X_i(k+1) = X_i(k) + C \times \left(\sum_{j \in I_{n_i}} q_j(k) - q_i(k) \right)$$

L'hypothèse implicite que nous avons considérée est que les flux sortant des arcs en amont de l'arc " i " peuvent tous entrer dans l'arc ce qui suppose que soit l'arc " i " est d'une capacité infinie soit que les conditions du trafic sont telles que les arcs ne sont jamais complètement saturés.

Pour décrire complètement la dynamique du nombre de véhicules sur l'arc " i ", il suffit alors de déterminer les expressions des flux sortants. Pour cela, nous utilisons le modèle *store-and-forward* qui permet d'exprimer facilement et linéairement un flux sortant, pendant un cycle de feu, en fonction de la durée de la phase de vert. Ainsi, pour un arc " j ", le flux sortant est donné par

$$u_j = S_j \times \frac{G_j}{C}$$

où S_j et G_j désignent respectivement les flux de saturation et la durée de la phase de vert pour l'arc " j ".

Le modèle *store-and-forward*, comme nous l'avons exposé précédemment, est plus adapté à des conditions du trafic où les carrefours sont saturés c'est à dire qu'il y a suffisamment de véhicules pour que le flux soit constamment égal au flux de saturation durant toute la phase de vert. Afin de pouvoir tenir compte de conditions de non saturation des carrefours, nous introduisons une amélioration qui permet de modéliser les flux sortant pour des carrefours non saturés. En effet, si la file d'attente devant un feu de signalisation est suffisamment longue, le flux sortant est modélisé par le modèle *store-and-forward*. Dans le cas contraire, le flux sortant sera égal au rapport entre le nombre de véhicules et la durée du cycle C . Ainsi, l'expression finale du flux sortant d'un arc " j " est donnée par

$$u_j = \min\left(\frac{X_j(k)}{C}; S_j \times \frac{G_j}{C}\right)$$

Étant donnée la dernière expression des flux sortants, l'évolution du nombre de véhicules est donnée par l'équation suivante :

$$X_i(k+1) = X_i(k) + C \cdot \left[\sum_{j \in I_i} \tau_{j,i} \cdot \min\left(S_j \cdot \frac{G_j(k)}{C}, \frac{X_j(k)}{C}\right) \right] - \min(S_i \cdot G_i(k), X_i(k))$$

En réécrivant cette équation avec l'introduction de variables binaires, nous obtenons :

$$X_i(k+1) = X_i(k) + C \cdot \left[\sum_{j \in I_i} \tau_{j,i} \cdot (\xi_j \cdot \left(S_j \cdot \frac{G_j(k)}{C} + (1 - \xi_j) \cdot \frac{X_j(k)}{C}\right)) \right] - \xi_i \cdot S_i \cdot G_i(k) - (1 - \xi_i) \cdot X_i(k)$$

où

$$\begin{cases} \xi_j = 1 & \text{si } S_j \cdot G_j \geq X_j(k) \\ \xi_j = 0 & \text{sinon} \end{cases}$$

Cette équation peut se mettre encore sous cette forme

$$X_i(k+1) = \xi_i \cdot X_i(k) - \sum_{j \in I_i} (1 - \xi_j) \cdot \tau_{j,i} \cdot X_j(k) - \xi_i \cdot S_i \cdot G_i(k) + \sum_{j \in I_i} \xi_j \cdot \tau_{j,i} \cdot S_j \cdot G_j(k)$$

Sous une forme matricielle, cela donne

$$X_i(k+1) = A_i \times X(k) + B_i \times G(k)$$

$$\text{où } X(k) = \begin{bmatrix} 1(k) \\ \cdot \\ \cdot \\ X_i(k) \\ \cdot \\ \cdot \\ X_{N_a}(k) \end{bmatrix} \quad G(k) = \begin{bmatrix} 1(k) \\ \cdot \\ \cdot \\ G_i(k) \\ \cdot \\ \cdot \\ G_{N_a}(k) \end{bmatrix}$$

Les matrices A_i et B_i sont des vecteurs ligne de dimension N_a où

$$\begin{cases} A_i(i) = \xi_i \text{ et } B_i(i) = -\xi_i \cdot S_i \\ A_i(j) = \tau_{j,i} \cdot (1 - \xi_j) \text{ et } B_i(j) = \xi_j \cdot \tau_{j,i} \cdot S_j \text{ si } j \in I_i \\ A_i(j) = 0 \text{ et } B_i(j) = 0 \text{ sinon} \end{cases}$$

Si l'arc " i " est un arc d'entrée nous avons alors

$$X_i(k+1) = A_i^{int} \times X(k) + B_i^{int} \times G(k) + E_i(k)$$

où

$$\begin{cases} A_i^{int}(i) = \xi_i \text{ et } B_i^{int}(i) = -\xi_i \cdot S_i \\ A_i^{int}(j) = 0 \text{ et } B_i^{int}(j) = 0 \quad \forall j \neq i \end{cases}$$

En généralisant l'équation de l'arc " i " pour tous les arcs y compris les arcs d'entrée du réseau, nous obtenons l'équation générale suivante :

$$X_i(k+1) = A \times X(k) + B \times G(k) + E(k)$$

le vecteur $E(k)$ est de dimension N_a et défini par $E_i(k) = 0$ si l'arc " i " est un arc d'entrée du réseau sinon $E_i(k)$ est une valeur déterminée par des données statistiques historiques ou mesurées en temps réel par des capteurs.

Les matrices A et B sont de dimensions $N_a \times N_a$ et définies ainsi :

$$\begin{cases} \text{ligne } i \text{ de } A = A_i \text{ et ligne } i \text{ de } B = B_i & \text{si } i \text{ est un arc intérieur} \\ \text{ligne } i \text{ de } A = A_i^{int} \text{ et ligne } i \text{ de } B = B_i^{int} & \text{si } i \text{ est un arc d'entrée} \end{cases}$$

Contrairement au modèle utilisé dans *TUC* par exemple, ce modèle est non linéaire mais permet de modéliser plus finement toutes les conditions du trafic.

2.4.2 Étude en simulation

2.4.2.1 Description du réseau routier et des caractéristiques des flux

L'objectif de cette section est d'étudier numériquement et en simulation le modèle des véhicules particuliers sur un réseau virtuel afin de vérifier sa cohérence. Il est aussi comparé au modèle linéaire simplifié.

Le réseau comporte quatre carrefours et six entrées. Il est constitué de 13 arcs reliant les carrefours (figure 2.12).

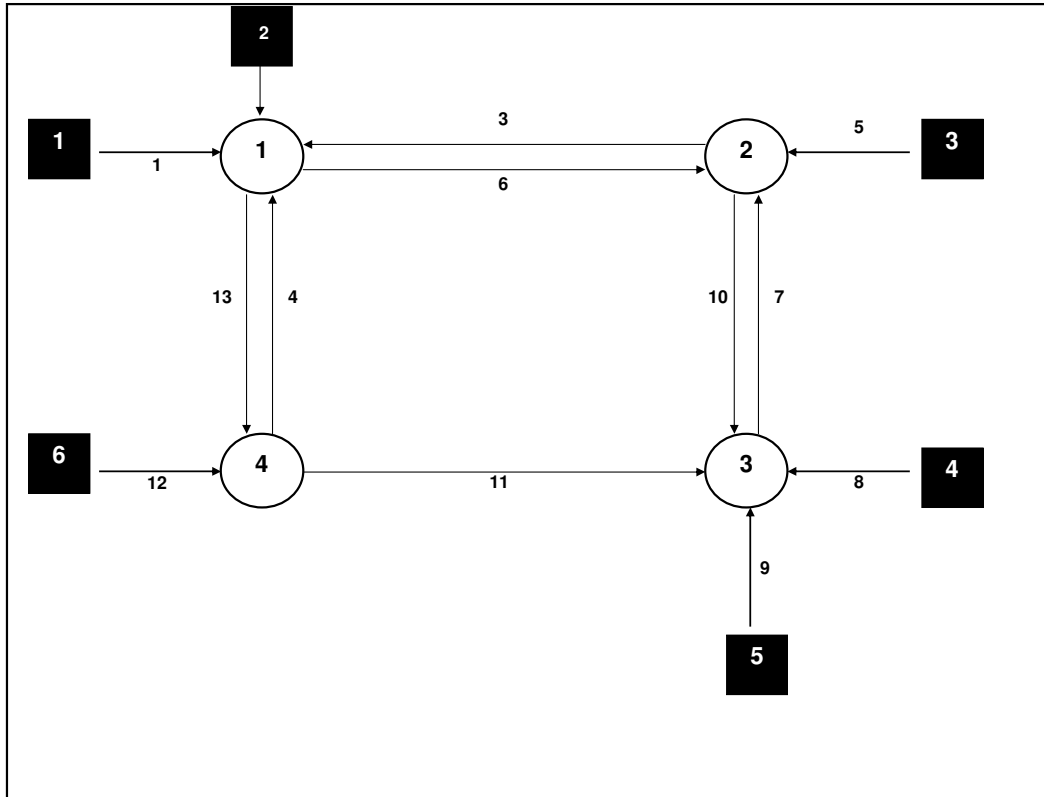


FIGURE 2.12 – réseau virtuel de test

La circulation des flux entre les carrefours est définie par les taux des mouvements tournants qui sont donnés

par la matrice $Taux$. $Taux(i, j)$ représente le taux du mouvement tournant de l'arc " i " vers l'arc " j ". Tous les termes de cette matrice sont nuls sauf pour les termes suivants :

Taux(1,6) =0.5 ; Taux(1,13)=0.3
Taux(2,6) =0.4 ; Taux(2,13)=0.5
Taux(3,13)=0.3
Taux(5,3) =0.6 ; Taux(5,10)=0.2
Taux(7,3) =0.3
Taux(8,7) =0.6
Taux(9,7) =0.7
Taux(11,7)=0.2
Taux(12,4)=0.2 ; Taux(12,11)=0.6
Taux(13,11)=0.3

TABLE 2.1 – Taux des mouvements tournants

Les simulations sont effectuées pour 40 cycles de feu, un cycle dure 80 secondes. Le réseau est supposé initialement vide de véhicules. Le temps des simulations est divisé en 4 périodes chacune composée de 10 cycles de feu. Pendant la première période, les flux entrants par les entrées du réseau sont les valeurs nominales que nous avons introduites. Pour les trois autres périodes, des perturbations sont introduites sur les flux d'entrée. Le tableau suivant résume les caractéristiques des flux pour les quatre périodes :

Première période	pas de perturbations
Deuxième période	augmentation sur les entrées 4 et 6
Troisième période	augmentation sur les entrées 1 et 5 et diminution sur 2 et 3
Quatrième période	diminution sur les entrées 1, 4, 5 et 6

TABLE 2.2 – Valeurs des flux sur la période de simulation

Les valeurs nominales des flux d'entrée pour les 10 premiers cycles sont données sur le tableau suivant :

Pour les feux de signalisation, nous avons considéré un cycle fixe et commun à tous les carrefours de durée 80 secondes. Afin de ne pas mettre des valeurs arbitraires pour les durées des phases de vert, nous avons considéré une approche pour partager les phases de vert de manière équitable entre les arcs en conflit. Pour un carrefour élémentaire à 4 branches avec deux phases déjà représenté sur figure 2.6, les arcs 1 et 3 d'une part et les arcs 2 et 4 d'autre part ont la même phase de vert. Si X_i désigne le nombre de véhicules dans l'arc " i " alors les durées des phases de vert pour l'arc 1 et 3 sont données par l'équation suivante :

$$G_1 = G_3 = \frac{\max(X_1; X_3)}{\max(X_1; X_2) + \max(X_3; X_4)} \times C$$

$$G_2 = G_4 = \frac{\max(X_2; X_4)}{\max(X_1; X_2) + \max(X_3; X_4)} \times C$$

Entrées	1	2	3	4	5	6	7	8	9	10	Valeurs moyennes
Entrée 1	19	16	18	27	9	26	13	11	17	10	21
Entrée 2	32	22	17	9	32	27	29	14	16	26	25
Entrée 3	10	16	15	18	25	27	8	27	25	4	22
Entrée 4	17	10	22	5	0	23	11	10	17	16	18
Entrée 5	10	17	23	23	14	0	23	23	12	14	18
Entrée 6	27	16	10	27	5	21	21	19	21	3	21

TABLE 2.3 – Valeurs des flux sans perturbations sur les 10 premiers cycles

Ces équations suggèrent implicitement que si le nombre de véhicules dans les arcs varie d'un cycle de feu à un autre, les durées des phases de vert varieraient elles aussi. Or il s'agit ici d'un plan de feu fixe. Pour déterminer ces plans de feux fixes, nous considérons les valeurs nominales des flux d'entrée dans le réseau. Nous supposons que tous les arcs ne sont plus contrôlés par les feux et donc que tous les accès sont libres. Nous observons après une courte période qu'un régime d'équilibre s'installe dans le réseau. Ce régime est caractérisé par un nombre constant de véhicules dans chaque arc qui ne varie plus d'un cycle à un autre. Nous utilisons ces valeurs constantes pour déterminer les durées fixes des phases de vert pour tous les carrefours. Le tableau suivant récapitule le régime d'équilibre pour chaque arc et la durée du vert qui lui est allouée.

arcs	1	2	3	4	5	6	7	8	9	10	11	12	13
nombre de VP en équilibre	21	25	21	4	22	21	27	18	18	4	20	21	25
durée de vert	37	43	37	43	36	36	44	42	38	38	42	37	43

TABLE 2.4 – Régime d'équilibre et durées fixes des phases de vert

La charge du trafic que nous avons considéré permet d'avoir des conditions de circulation dans les limites de la saturation. Nous considérons une autre charge du trafic qui correspond à un trafic très chargé et des conditions sur-saturées.

Dans les stratégies de régulation *TUC* et *NetPrior*, les flux sortant d'un arc donné sont donnés par le modèle *store-and-forward* déjà exposé. Dans le modèle que nous avons développé, nous avons introduit une amélioration qui permet de tenir compte des conditions de non saturation du trafic. Dans la suite, nous présentons une étude comparative et les résultats numériques pour les deux charges du trafic et pour les deux modèles suivants :

- Le modèle *store-and-forward* sans amélioration.
- Le modèle développé avec amélioration.

Les résultats de deux scénarii sont comparés avec le cas où les arcs ne sont pas contrôlés par les feux c'est à dire quand l'accès est libre. Ce dernier cas servira de référence pour juger de la pertinence des résultats des deux scénarii.

2.4.2.2 Résultats numériques

Nous considérons en premier lieu une charge de trafic normale et nous nous intéressons à l'évolution du nombre de véhicules par cycle dans l'arc numéro 1 qui est un arc d'entrée. Cet arc bénéficie chaque cycle d'une phase de vert de durée 37 secondes permettant d'écouler 18 véhicules compte tenu de la valeur du taux de saturation $S = 0.5$ véhicule/seconde. Les valeurs des flux d'entrée dans cet arc sont nominales pour les deux premières périodes puis augmentent dans la troisième pour diminuer dans la quatrième et dernière période. La figure 2.13 donne l'évolution du nombre de véhicules dans l'arc pour le modèle sans amélioration, le modèle amélioré et enfin pour le modèle amélioré avec accès libre (pas de feux de signalisation).

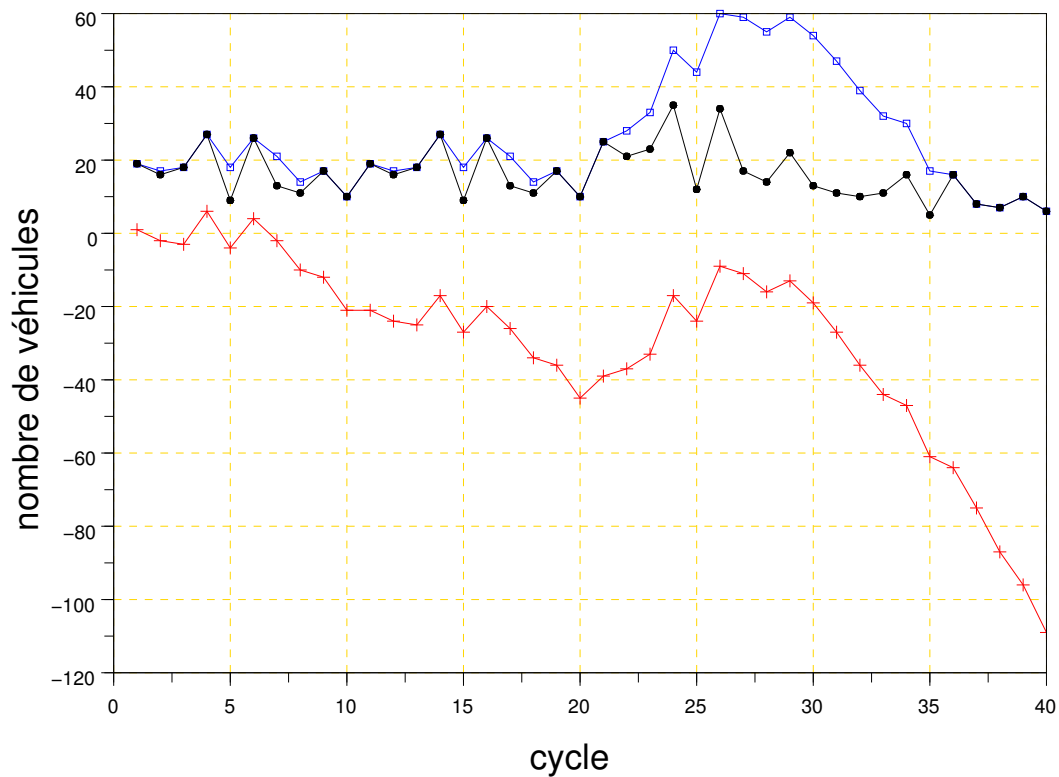


FIGURE 2.13 – Évolution du nombre de véhicules dans l'arc 1

Tout d'abord, le plus surprenant ce sont les valeurs négatives pour le nombre de véhicules donné par le modèle sans amélioration. Pendant les deux première périodes, le nombre de véhicules décroît et cela est tout à fait normal. D'après le modèle sans amélioration, la ligne de feu de l'arc 1 peut faire écouler, à chaque cycle, 18 véhicules qu'il y a réellement ou pas ce nombre de véhicules. Or dans les deux première

périodes qui s'étalent jusqu'au 20_{me} cycle, le nombre moyen de véhicules qui entrent dans l'arc à chaque cycle est de l'ordre de 16 véhicules. Ce qui explique les valeurs négatives et la décroissance du nombre de véhicules jusqu'à atteindre le seuil de -43 véhicules. Parallèlement, le modèle amélioré montre une évolution cohérente avec les valeurs des flux d'entrée. Les deux courbes associées au modèle amélioré pour un accès contrôlé par les feux et celui libre montrent des différences négligeables jusqu'au 20_{me} . Le trafic, durant cette période, est non saturé ce qui explique que malgré la présence des feux de signalisation, ceux-ci n'exercent pas une fonction de blocage puisque les valeurs des variables de plan de feu permettent de faire écouler le trafic comme si les accès étaient complètement libres. Pour la 3_{me} période (du cycle 21 au cycle 30), les flux d'entrée sur l'arc 1 sont augmentés de 30%. Nous observons alors une différence entre un accès libre et un accès contrôlé. Pour l'accès libre, rien ne change et le nombre de véhicule continue à osciller autour de 19 véhicules. Pour l'accès contrôlé, les paramètres des plans de feu ne permettent plus de faire écouler le trafic puisque ce dernier devient chargé. Pour le modèle non amélioré, nous observons une croissance dans les mêmes proportions que celui du modèle amélioré. Ce modèle se comporte alors de manière plus fidèle à la réalité. Cependant, les fortes valeurs négatives des 20 premiers cycles laissent le nombre de véhicules négatif même avec une surcharge du trafic. Dans la dernière période (les 10 derniers cycles), le trafic est diminué et nous retrouvons le comportement symptomatique du modèle non amélioré et une concordance quasi-totale entre un accès libre et un accès contrôlé.

Nous nous intéressons maintenant à deux autres arcs : l'arc d'entrée 2 et l'arc intérieur 7. L'évolution du nombre de véhicules dans ses arcs pour les trois simulations est donnée dans la figure 2.14.

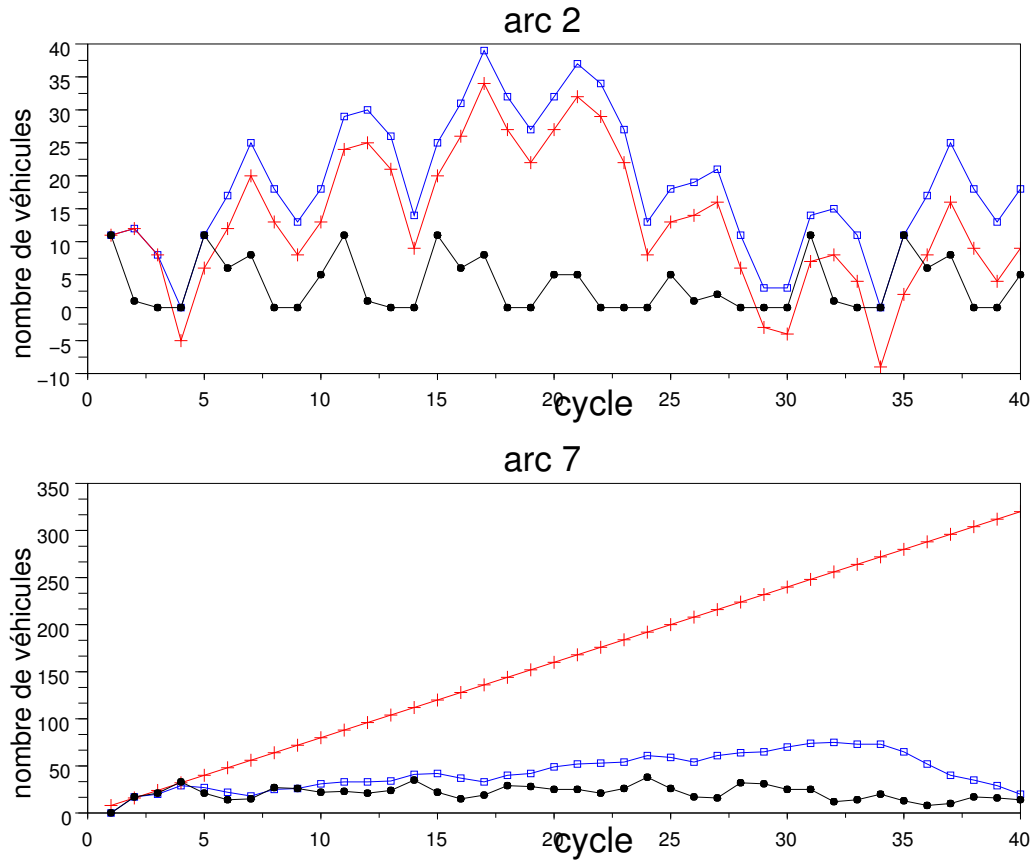


FIGURE 2.14 – Évolution du nombre de véhicules dans les arcs 2 et 7

Comme pour l'arc 1, nous observons une légère différence entre un accès libre et un accès contrôlé pour les 10 premiers cycles. Ceci s'explique par le fait que le feu de signalisation de cet arc permet d'écouler le trafic. Pour les 20 cycles suivants, l'écart augmente puisque les flux augmentent sur les entrées 4 et 5 qui alimentent, à travers les arcs 8 et 9, l'arc 7. Concernant le modèle sans amélioration, le nombre de véhicules augmente avec une pente constante de 8 véhicules/cycle. Selon le modèle non amélioré, la dynamique d'un arc intérieur ne dépend pas de la charge du trafic mais seulement les durées des phases de l'arc lui-même et des arcs qui l'alimentent. L'arc 7 est alimenté par les arcs 8,9 et 11 qui l'alimentent, à chaque cycle, de respectivement 13, 13 et 4 véhicules (en tenant compte des taux de mouvement tournant). L'arc 7 est contrôlé par un feu qui lui accorde une phase de vert de durée 44 secondes qui permet de faire passer 22 véhicules (compte tenu du taux de saturation 0.5 véhicule/seconde). En faisant la différence entre la somme de 13, 13 et 4 et 22, nous retrouvons bien 8 véhicules qui sont, à chaque cycle, rajoutés au nombre total déjà cumulé dans l'arc.

L'arc 2 est un arc d'entrée qui est traversé par des flux importants. Le trafic sur cet arc étant chargé, nous observons une différence importante entre l'accès libre qui permet d'écouler le trafic et l'accès contrôlé qui retient les véhicules. Nous observons par ailleurs que le modèle sans amélioration reproduit bien le comportement du modèle amélioré. Dans des conditions de trafic dans les limites de la surcharge, les véhicules retenus dans les files d'attente continuent de passer durant toute la durée de la phase de vert. Ceci explique

bien la petite différence entre les deux modèles.

Afin de vérifier le comportement des deux modèles, les différences entre les deux et la cohérence du modèle amélioré, nous augmentons la charge du trafic du double sur toutes les entrées du réseau. Sur la figure 2.15 est représentée l'évolution du nombre de véhicules sur les trois arcs 1, 2 et 7 pour le modèle non amélioré, le modèle amélioré et le cas du réseau en accès libre sur toutes les lignes du feu.

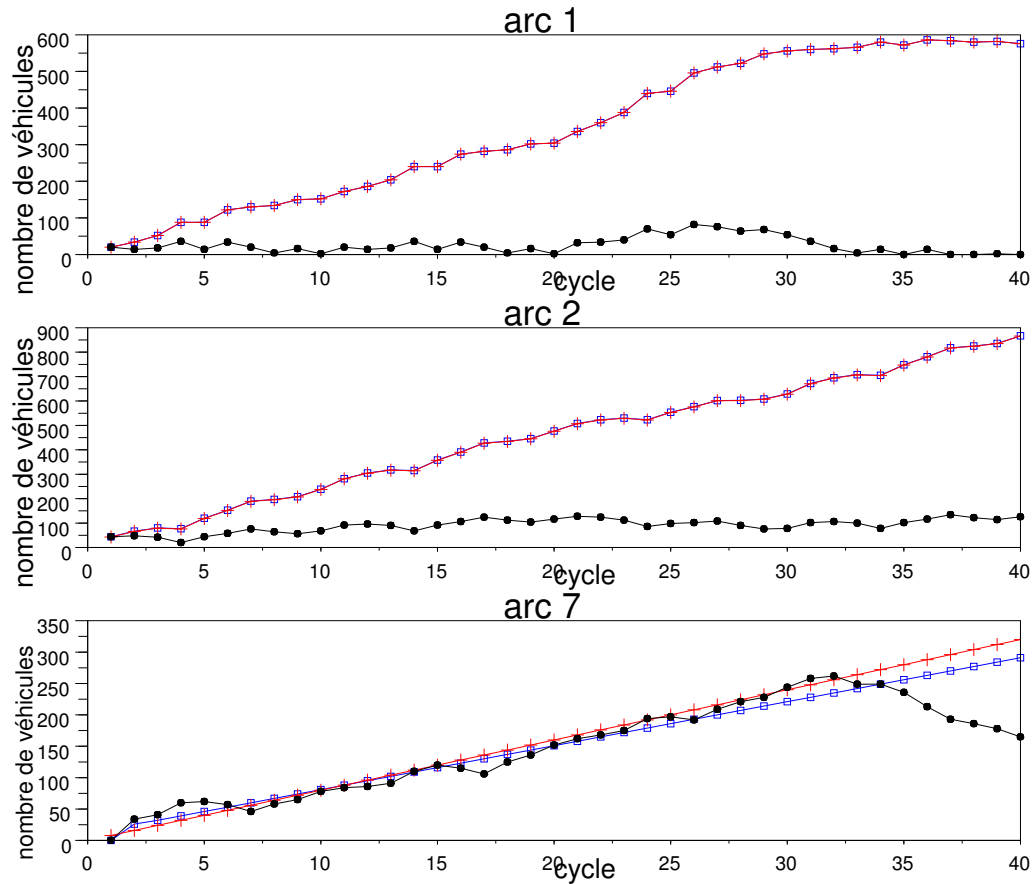


FIGURE 2.15 – Évolution du nombre de véhicules dans les arcs 1, 2 et 7

Pour les arcs d'entrée 1 et 2, les modèles amélioré et non amélioré donnent exactement les mêmes résultats. Dans des conditions de trafic sur-chargé, il n'y a pas de différence entre les deux modèles pour les arcs d'entrée puisque l'évolution du nombre de véhicules ne dépend que des flux d'entrée supposés connus et des durées des phases de vert de ces arcs. Pour l'arc intérieur 7, il y a une légère différence entre les deux modèles. Ceci revient à ce qu'un arc intérieur dépend aussi des variables relatives aux arcs en amont et donc pour le modèle non amélioré de leurs paramètres de feu de signalisation qui sont indépendants de la charge du trafic. Dans un réseau en accès libre, il est naturel que le nombre de véhicules dans les arcs d'entrée soient très inférieur à celui du réseau contrôlé. Pour l'arc intérieur 7, la différence n'est pas aussi importante puisque cet arc reçoit des flux importants des arcs 8, 9 et 11.

2.5 Conclusion

La modélisation du trafic urbain nécessite une description mathématique de toutes les variables et des paramètres qui interviennent et qui influent sur le trafic. Nous avons présenté dans ce chapitre les formalismes mathématiques des réseaux routiers, des paramètres des plans de feu et certaines opérations mathématiques importantes pour les développements du chapitre 4. Nous avons décrit aussi le modèle du trafic des véhicules particuliers. Ce modèle décrit l'évolution du nombre de véhicules sur chaque arc du réseau en fonction des paramètres et des variables des plans de feu de tous les carrefours. Le modèle se base sur une équation de conservation du nombre de véhicules et sur l'approximation du modèle *store-and-forward* pour l'expression des flux sortants. Cette approximation linéaire ne permet de reproduire que des conditions saturées du trafic. Nous avons alors introduit une amélioration pour tenir compte aussi des conditions de sous saturation. Ceci a transformé le modèle initial linéaire à un modèle non linéaire.

L'étude en simulation a montré la cohérence et la pertinence du modèle amélioré ainsi que les différences et similitudes entre les deux modèles pour différentes conditions du trafic. Dans le chapitre suivant, nous nous intéressons au développement du modèle des véhicules de transport en commun.

Chapitre 3

Modèle de progression des véhicules de transport en commun de surface

3.1 Introduction

Jusqu'à un passé très proche, toute l'attention des exploitants et des chercheurs dans le domaine de la régulation du trafic était concentrée sur les véhicules particuliers et ce pour deux raisons principales. La première est l'importance du trafic des véhicules particuliers par rapport à celui des véhicules de transport en commun. La seconde raison réside dans l'indisponibilité de modèles pour représenter la circulation des véhicules de transport en commun tandis que le trafic général a été largement étudié dans la littérature.

Au cours des dernières années, les transports en commun ont connu une importante croissance. La nécessité de disposer d'outils mathématiques pour comprendre, analyser et agir sur le trafic de ces véhicules est devenue une urgence. Cependant, malgré cette prise de conscience, aucune avancée majeure n'a été observée dans la littérature. Les quelques travaux sur les véhicules de transport en commun considèrent ces derniers plutôt comme des obstacles pour les autres véhicules. D'autres travaux sur la régulation du trafic ne proposent que des modèles statistiques portant sur les temps d'arrivée des véhicules sur les lignes des feux de signalisation.

L'objectif de ce chapitre est de décrire un modèle pour les transports en commun. Il s'agit d'un modèle à base d'événements qui décrit la progression d'un véhicule de transport en commun à travers des carrefours contrôlés par des feux de signalisation et de stations dans lesquelles le véhicule doit s'arrêter. L'originalité de ce modèle réside dans sa capacité à modéliser finement et simplement le trafic de ces véhicules. Il permet aussi de tenir compte de tous les aspects du trafic urbain comme les interactions entre les transports en commun et les autres véhicules. Deux modèles sont présentés. Le premier permet de tenir compte que des durées des feux verts des carrefours traversés par les véhicules de transport en commun. Les instants de début des phases de vert sont supposés inconnus et traités comme des variables aléatoires. Le deuxième modèle, plus complet mais aussi plus complexe, tient compte des instants de début des phases de vert. Une étude en

simulation est présentée ainsi qu’une comparaison entre les deux modèles.

3.2 Modèle simple

L’objectif du modèle est de décrire l’évolution dans le temps de la position d’un véhicule de transport en commun. Le temps est discrétisé selon la durée, commune à tous les carrefours, d’un cycle de feu. L’itinéraire de la ligne de transport en commun est supposé connu et est constitué d’une succession de carrefours et de stations. Les temps d’arrêt aux stations sont supposés connus. Dans la suite, nous parlerons de véhicule de transport en commun ou bus pour désigner la même chose.

En toute rigueur, la position future d’un bus dépend de :

- sa position actuelle,
- l’état général du trafic en particulier les files d’attente des véhicules particuliers,
- l’état des feux de signalisation des carrefours à traverser à savoir les instants de début et les durées des phases de vert durant lesquelles le bus peut passer le carrefour,
- les temps d’arrêt éventuels dans les stations.

Ce premier modèle tient compte de tous ces paramètres sauf des instants de début des phases de vert qui sont considérés inconnus. Ceci a l’avantage de rendre le modèle plus simple étant donné qu’il sert comme un estimateur dans une stratégie de régulation en boucle fermée. La philosophie du modèle consiste à percevoir les carrefours successifs et les stations comme des obstacles à passer. Autrement dit, le passage des carrefours et des stations sont des événements après lesquels une ré-initialisation de certaines variables est nécessaire. Après le passage d’un événement, le bus doit immédiatement passer à un autre jusqu’à ce que le cycle actuel se termine ou qu’il sort de la zone contrôlée du réseau. Pour suivre l’évolution du temps au cours du cycle actuel, nous introduisons une nouvelle variable qui mesure le temps restant avant la fin du cycle. Définie de la sorte, cette variable sera initialisée à la valeur de la durée C du cycle et quand le cycle se termine, elle prend la valeur 0. Deux types d’événements peuvent se produire : le passage d’un carrefour à feux et le stationnement dans une station. Dans le cas d’un carrefour, la position du bus et le temps restant sont mis à jour par l’algorithme « g_feu ». Pour une station, nous utilisons l’algorithme « $g_station$ ». La position du bus dans le cycle k est notée $P(k)$. L’objectif du modèle est l’estimation de $P(k+1)$. Avant le passage d’un événement, la position du bus et le temps restant avant la fin du cycle sont désignées par P_p et tr_p . Après le passage, ces deux variables sont indexées par la lettre n . Enfin, nous présentons les variables suivantes :

- Td : instant de début de la phase de vert durant laquelle le bus peut passer.
- G : durée de la phase de vert durant laquelle le bus peut passer.
- Nb : nombre de véhicules particuliers intercalés entre le bus et la ligne de feu.

- T_{st} : temps de stationnement du bus dans une station.

Compte tenu de toutes ces notations, le pseudo-code du modèle est le suivant :

Fonction générale du modèle

$$P_p = P(k)$$

$$tr_p = C$$

tant que ($tr_p > 0$) faire

si (événement=station) alors

$(P_n, tr_n) = g_station(P_p, tr_p, T_{st})$

sinon

$(P_n, tr_n) = g_feu(P_p, tr_p, G, Nb)$

fin si

$$P_p = P_n$$

$$tr_p = tr_n$$

fin tant que

$$P(k+1) = P_n$$

Dans la section suivante, nous décrirons en détail cet algorithme puis nous nous intéresserons à l'algorithme plus complexe « g_feu ».

3.2.1 Algorithme pour le passage d'une station

Nous admettons, pour ce modèle, qu'aucune file d'attente ne remonte d'un carrefour jusqu'à une station proche placée en amont. Ceci peut se réaliser dans les deux situations suivantes :

- La station est placée suffisamment loin du carrefour en aval.
- Les files d'attente ne sont pas aussi importantes pour remonter jusqu'à la station.

Les exploitants des transport publics et du réseau routier font de sorte qu'on se place le plus souvent dans ces deux situations.

Le prochain obstacle à traverser est une station (située à la distance L_s) dont le temps de stationnement estimé T_{st} est supposé connu. Une variation de ce paramètre peut engendrer d'importantes conséquences dans la régulation du trafic. Par exemple, une régulation qui permet de garantir « une onde verte » aux bus est très dépendante de ces paramètres. En général, les temps de stationnement sont calculés statistiquement à partir de mesures effectuées dans le passé et peuvent varier en fonction de la période dans la journée. Le bus est situé à la position P_p et il reste tr_p secondes avant la fin de l'actuel cycle de feu (figure 3.1).

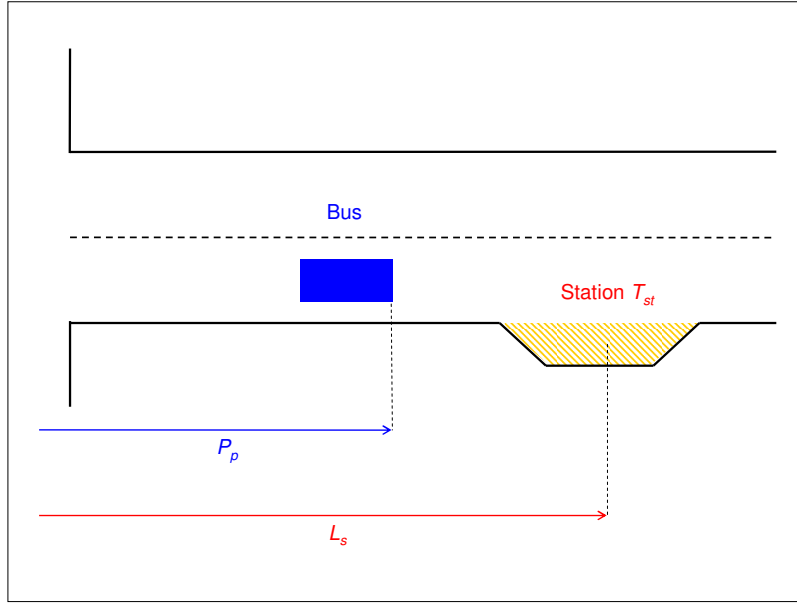


FIGURE 3.1 – Bus au passage d’une station

Le bus, quand il n’est pas dans une file d’attente, est supposé rouler avec une vitesse libre V_b connue. Cette vitesse peut dépendre des tronçons de routes et des caractéristiques géométriques du réseau routier. En toute rigueur, V_b dépend de l’indice i d’un arc. Dans la suite, nous omettrons cet indice pour plus de simplicité. Depuis sa position actuelle P_p , le bus roule avec la vitesse V_b jusqu’au prochain événement. Deux événements peuvent se produire : l’arrivée du bus dans la station et la fin du cycle actuel de durée C . Le bus arrive dans la station avant la fin du cycle si $tr_p \times V_b \geq L_s - P_p$. Dans ce cas, la nouvelle position du bus est donnée par :

$$P_n = L_s$$

A la sortie de la station, le nouveau temps restant tr_n avant la fin du cycle est égal à l’ancien temps restant auquel on soustrait le temps nécessaire pour parcourir la distance $L_s - P_p$ et le temps de stationnement T_{st} . L’expression de tr_n est donc donnée par l’équation suivante :

$$tr_n = tr_p - \frac{L_s - P_p}{V_b} - T_{st}$$

Le pseudo-code de l’algorithme « g_station » est le suivant :

Fonction g_station
si ($tr_p \times V_b < L_s - P_p$) alors
$P_n = P_p + tr_p \times V_b$
$tr_n = 0$
sinon
$P_n = L_s$
$tr_n = tr_p - T_{st} - \frac{L_s - P_p}{V_b}$
fin si

La variable tr , qui compte le temps restant avant la fin du cycle actuel, est initialisée à C . A la fin du cycle, elle prend la valeur 0. En réalité ce n'est pas toujours le cas. Quand un cycle prend fin et qu'un bus est toujours stationné dans une station, tr prend une valeur négative. Pour cela il suffit de revoir l'expression de $tr = tr - \frac{L_s - P_p}{V_b} - T_{st}$ dans ce cas particulier. Dans cette situation, tr est négative et la nouvelle valeur d'initialisation de tr pour le cycle suivant sera égale à $C + tr$, une valeur inférieure à C puisque la valeur finale de tr , pour le cycle précédent, est négative.

3.2.2 Algorithme pour le passage d'un carrefour à feux

Cet algorithme permet de mettre à jour la position du bus ainsi que le temps restant avant la fin du cycle actuel dans le cas où l'obstacle à passer est un carrefour à feux. Initialement, le bus est dans la position P_p dans un arc d'indice i et il reste tr_p secondes avant la fin du cycle de durée C . La ligne de feu est située à une distance L_f par rapport à une même référence que la position du bus. Enfin entre le bus et la ligne de feu, il y a Nb véhicules particuliers avec ont une longueur moyenne a généralement prise égale à 8 m (figure 3.2). Le feu contrôlant l'arc i a une durée de feu vert $G_i(k)$ où k désigne le k^{me} cycle de feu. Dans la suite, nous omettrons l'indice k pour plus de simplicité puisque nous nous plaçons, dans toute cette section, dans le seul cycle k .

Avant de détailler le modèle et rentrer dans la discussion sur les différents cas qui se présentent, nous présentons le calcul de deux variables qui sont nécessaires pour le développement du modèle.

3.2.2.1 Calcul de l'instant où un bus rejoint une file d'attente

L'objectif est de calculer l'instant Y où le bus peut rejoindre la file d'attente. Le bus avant d'atteindre la file d'attente a une vitesse V_b . Nous notons V_f la vitesse de progression de la file d'attente (figure 3;3).

Il est évident alors que Y doit respecter l'équation suivante :

$$P_p + V_b \cdot Y = L_f - a \cdot Nb^i + V_f \cdot Y$$

Ceci donne une première expression de Y :

$$Y = \frac{L_f - P_p - a \cdot Nb^i}{V_b - V_f}$$

La vitesse V_f est la vitesse de progression de la file d'attente ou de disparition des véhicules de la file d'attente. D'après le modèle *store-and-forward*, le nombre de véhicules qui sortent d'un tronçon durant un cycle de

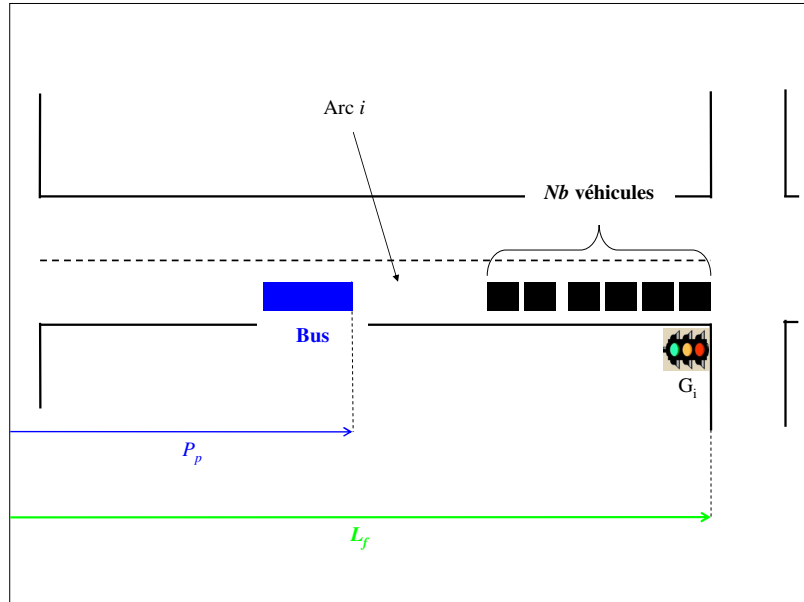


FIGURE 3.2 – Bus au passage d'un carrefour à feux

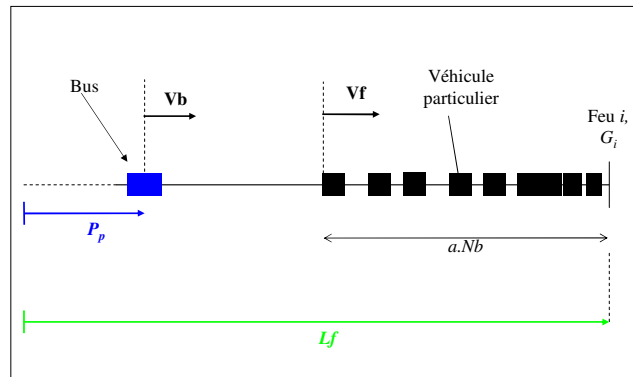


FIGURE 3.3 – Un bus derrière une file d'attente

durée C contrôlé par un feu de durée de vert G_i est $S_i \times G_i$. Étant donné que la longueur moyenne d'un véhicule particulier est a , la distance qui disparaît de la file d'attente, pendant un cycle, est $aS_i \times G_i$. D'où l'expression de V_f

$$V_f = \frac{aS_i \times G_i}{C}$$

En remplaçant la nouvelle expression de V_f dans l'expression de Y , on obtient :

$$Y = C \times \frac{Lf - P_p - a \times Nb^i}{C \times V_b - aS_i \times G_i}$$

Il ne reste qu'à vérifier que $Y > 0$. Ceci est le cas si $V_b > V_f$. Or $V_f = \frac{aS_i \times G_i}{C}$ et $G_i < C$ donc nous avons $V_f < a \times S_i$. Généralement les valeurs de a et S_i sont respectivement autour de 8 m et 0.5 veh/s ce qui donne $V_f < 4$. De l'autre côté, la vitesse libre V_b d'un bus est largement supérieure à 4 m/s équivalente à 14.4 km/h. D'où $V_b > V_f$ et donc $Y > 0$. Théoriquement nous avons montré qu'un bus peut rattraper une file d'attente à condition qu'elle soit suffisamment longue.

3.2.2.2 Temps d'attente moyen d'un bus dans une file d'attente

Nous considérons que la situation est la suivante : un bus est sur une ligne de feu et il reste tr secondes avant la fin du cycle de feu actuel. L'objectif est d'estimer son temps d'attente moyen.

Une phase d'un cycle de feu est caractérisée par deux paramètres qui sont son instant de début que nous noterons Td et sa durée que nous noterons G . Pour des raisons expliquées précédemment, le modèle ne tient compte que des durées des phases. Les instants de début des phases sont supposés des variables aléatoires uniformément distribuées sur le cycle de durée C .

La phase du vert pour le tronçon traversé par le bus a une durée G dans un cycle de durée C . Comme son instant de début Td est aléatoire et uniformément distribué, un calcul élémentaire permet de déterminer la durée moyenne de la phase du vert disponible durant la période de tr secondes. Ce calcul donne la valeur moyenne de $\frac{tr}{C} \times G$. Par exemple, si un cycle a une durée de 80 secondes et qu'une phase de vert est de durée 40 secondes alors dans une période de 30 secondes, la durée du vert sera $\frac{30}{80} \times 40 = 15$.

Nous nous plaçons maintenant dans la période de tr secondes avec une durée de vert de $\frac{tr}{C} \times G$. L'instant de début de vert est noté Td_r et il est considéré aussi comme aléatoire et uniformément distribué sur l'intervalle $[0; tr]$. Td_r doit vérifier la condition suivante $0 \leq Td_r \leq tr \times (1 - \frac{G}{C})$. Le temps d'attente du bus est noté t_{stop} . Puisqu'il n'y a pas de véhicules particuliers entre le bus et la ligne du feu, il est évident que $t_{stop} = Td_r$. Dans le cas le plus favorable, le vert commence quand le bus arrive sur la ligne du feu et donc $t_{stop} = 0$. Dans le cas le moins favorable, le temps d'attente du bus sera $t_{stop} = tr \times (1 - \frac{G}{C})$. Le temps d'attente moyen t_{stop_moy} est calculé grâce à l'équation suivante

$$t_{stop_moy} = \frac{1}{tr \times (1 - \frac{G}{C})} \times \int_0^{tr \times (1 - \frac{G}{C})} x \cdot dx = \frac{1}{2} \times tr \times (1 - \frac{G}{C})$$

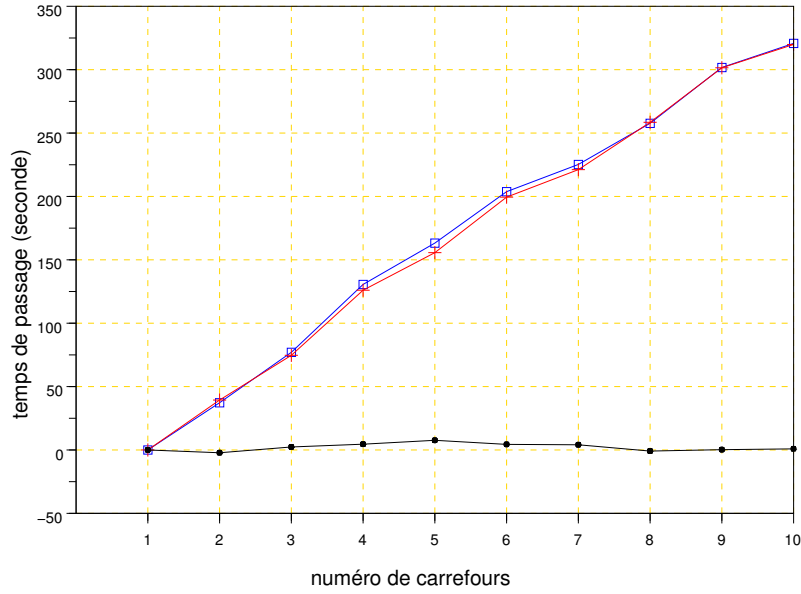


FIGURE 3.4 – temps de passage moyen du bus sur les 10 carrefours pour 1000 essais

Il est clair que cette expression analytique n'est qu'une approximation des temps d'attente d'un bus devant une ligne de feu. Néanmoins, elle permet de modéliser ces variables de façon précise et simple. Afin d'avoir un aperçu numérique, nous considérons l'exemple élémentaire suivant. Un bus part d'une position initiale au temps $t = 0$ et il doit traverser 10 carrefours à feu. La distance moyenne entre ces carrefours est de 200 m. Les 10 carrefours fonctionnent avec la même durée de cycle $C = 80$ secondes. Les phases pour les tronçons traversés par le bus ont des durées fixes et connues de moyenne 40 secondes. Les instants de début des phases de vert sont générés de façon aléatoire avec une distribution uniforme sur l'intervalle $[0; C]$. Afin de considérer la tendance moyenne, nous avons généré 1000 tests et nous nous intéressons aux instants de franchissement du bus pour les 10 carrefours. L'approximation est comparée au cas où les instants de début de phase sont connus et fixes. La figure 3.4 donne les instants moyens de franchissement des 10 carrefours pour l'approximation et pour le cas où les instants de début des phases sont connus ainsi que la différence entre les deux. En trait rouge avec des signes « + » sont représentés les résultats pour l'approximation. Les temps de passage calculés avec un modèle tenant compte des instants de début des phases sont tracés en trait bleu avec des carrés. Enfin, la différence est tracée en trait noir avec des points.

Les courbes de la figure 3.4 montrent que l'approximation reproduit bien, en moyenne, le comportement des bus en tenant compte des instants de début de phase. Sur un seul essai, la figure 3.5 montre des écarts plus importants mais l'approximation reproduit assez bien, malgré ses écarts, le comportement général.

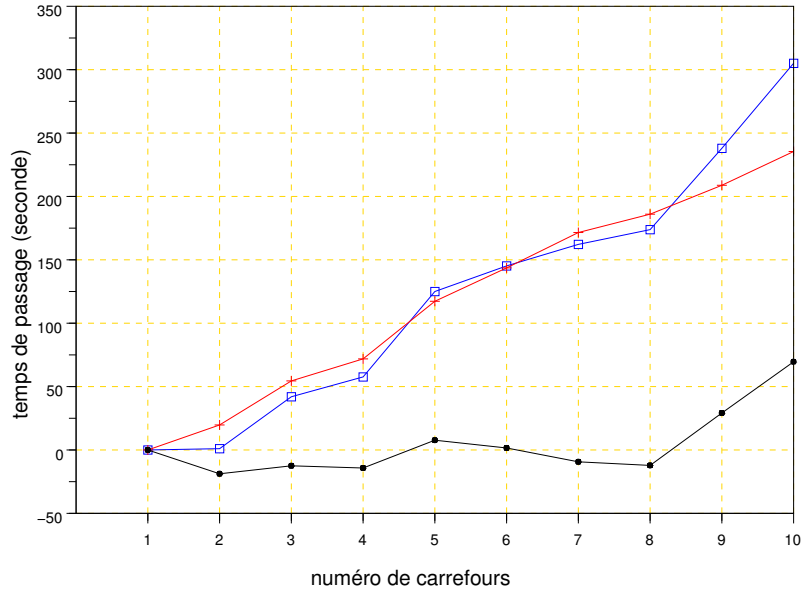


FIGURE 3.5 – temps de passage du bus sur les 10 carrefours pour un seul essai

3.2.2.3 L'algorithme « g_feu »

La situation est illustrée par la figure 3.2. D'après les sections précédentes, le bus peut rejoindre la file d'attente après un temps Y donné par

$$Y = C \times \frac{Lf - P_p - a \times Nb^i}{C \times V_b - aS_i \times G_i}$$

L'objectif est de déterminer la nouvelle position du bus P_n et le nouveau temps restant tr_n après le passage du carrefour ou à la fin du cycle de feu actuel. Pour cela plusieurs cas sont à discerner selon l'occurrence des événements les uns par rapport aux autres.

– cas 1 :

$$Lf_{m,i} - P_p - a \cdot Nb^i \geq tr_p \cdot V_b$$

Dans cette situation, le bus ne pourra pas rejoindre la file d'attente de longueur $a \times Nb^i$. Les nouvelles valeurs P_n et tr_n sont données par :

$$\begin{cases} P_n = P_p + tr_p \cdot V_b \\ tr_n = 0 \end{cases}$$

– cas 2 :

$$Lf_{m,i} - P_p - a \cdot Nb^i \leq tr_p \cdot V_b$$

Il s'agit de la situation où le bus peut rejoindre la file d'attente avant la fin du cycle actuel. Ceci se réalise après un temps Y donné par

$$Y = C \cdot \frac{Lf_{m,i} - P_p - a \cdot Nb^i}{C \cdot V_b - a \cdot S_i \cdot G_i(k)}$$

Dans cette situation, il faut distinguer deux cas : celui où le bus rejoint la file d'attente avant la ligne de feu et celui où il la rejoint après.

1. sous-cas 2.1 : Il correspond à l'inégalité $Y \cdot V_b \geq Lf_{m,i} - P_p$ où le bus n'a pas la possibilité de rejoindre la file d'attente avant la ligne du feu. Il roule avec sa vitesse libre V_b jusqu'à la ligne et il l'atteint à l'instant $tr_p - \frac{Lf_{m,i} - P_p}{V_b}$ avant la fin du cycle. Ici aussi, on doit considérer deux possibilités :

– sous-cas 2.1.1 : $tr_p \cdot V_b \leq Lf_{m,i} - P_p$. Les nouvelles valeurs P_n et tr_n sont données par :

$$\left| \begin{array}{l} P_n = P_p + tr_p \cdot V_b \\ tr_n = 0 \end{array} \right.$$

– sous-cas 2.1.2 : $tr_p \cdot V_b \geq Lf_{m,i} - P_p$. Le bus atteint la ligne du feu à l'instant $C - tr_p + \frac{Lf_{m,i} - P_p}{V_b}$. Avant la fin du cycle actuel, le temps restant est $tr_p - \frac{Lf_{m,i} - P_p}{V_b}$. Suivant l'approximation de la section 2.2.2, le bus est arrêté pendant un temps $\frac{1}{2} \times (tr_p - \frac{Lf_{m,i} - P_p}{V_b}) \times (1 - \frac{G_i(k)}{C})$. En quittant l'arc, le temps restant avant la fin du cycle est $\frac{1}{2} \times (tr_p - \frac{Lf_{m,i} - P_p}{V_b}) \times (1 + \frac{G_i(k)}{C})$. Finalement, nous avons :

$$\left| \begin{array}{l} P_n = Lf_{m,i} \\ tr_n = \frac{1}{2} \cdot (tr_p - \frac{Lf_{m,i} - P_p}{V_b}) \cdot (1 + \frac{G_i(k)}{C}) \end{array} \right.$$

2. sous-cas 2.2 : Il correspond à l'inégalité $Y \cdot V_b \leq Lf_{m,i} - P_p$ où le bus rejoint la file d'attente avant la ligne du feu. Le nombre de véhicules particuliers devant le bus est $Nb^i - \frac{S_i \cdot Y \cdot G_i(k)}{C}$. Le temps restant avant la fin du cycle est $tr_p - Y$ dans lequel $(tr_p - Y) \cdot \frac{G_i(k)}{C}$ la durée du feu vert restante. Ici encore, deux possibilités doivent être distinguées :

– sous-cas 2.2.1 : $Nb^i \geq \frac{S_i \cdot G_i(k) \cdot tr_p}{C}$. La file d'attente ne peut pas disparaître complètement pendant ce cycle. Il existe deux situations selon que « $tr_p \cdot V_b$ est supérieur ou inférieur à $Lf_{m,i} - P_p - a \cdot (Nb^i - \frac{S_i \cdot G_i(k) \cdot tr_p}{C})$ ». Pour le premier cas, nous avons :

$$\left| \begin{array}{l} P_n = P_p + tr_p \cdot V_b \\ tr_n = 0 \end{array} \right.$$

Et pour le second, nous avons :

$$\left| \begin{array}{l} P_n = Lf_{m,i} - a \cdot (Nb^i - \frac{S_i \cdot G_i(k) \cdot tr_p}{C}) \\ tr_n = 0 \end{array} \right.$$

– sous-cas 2.2.2 : $Nb^i \leq \frac{S_i \cdot G_i(k) \cdot tr_p}{C}$. La file d'attente disparaît complètement durant ce cycle et dans ce cas P_n et tr_n sont données par :

$$\left| \begin{array}{l} P_n = Lf_{m,i} \\ tr_n = \frac{1}{2} \cdot (tr_p - Y) \cdot \left(1 + \frac{G_i(k)}{C}\right) - \frac{Nb^i}{S_i} + \frac{Y \cdot G_i(k)}{C} \end{array} \right.$$

Nous pouvons récapituler les différents cas et sous-cas précédemment traités comme suit :

Les conditions
A : $Lf_{m,i} - P_p - a \cdot Nb^i \geq tr_p \cdot V_b$
B : $Nb^i \geq \frac{S_i \cdot G_i(k) \cdot tr_p}{C}$
C : $tr_p \cdot V_b \leq Lf_{m,i} - P_p - a \cdot \left(Nb^i - \frac{S_i \cdot G_i(k) \cdot tr_p}{C}\right)$
D : $G_i(k) \geq \frac{C \cdot Nb^i \cdot V_b}{S_i \cdot (Lf_{m,i} - P_p)}$
E : $tr_p \cdot V_b \leq Lf_{m,i} - P_p$
Les équations
EQ 1 : $P_n = P_p + tr_p \cdot V_b$ et $tr_n = 0$
EQ 2 : $P_n = Lf_{m,i} - a \cdot \left(Nb^i - \frac{S_i \cdot G_i(k) \cdot tr_p}{C}\right)$ et $tr_n = 0$
EQ 3 : $P_n = Lf_{m,i}$ et $tr_n = \frac{1}{2} \cdot (tr_p - Y) \cdot \left(1 + \frac{G_i(k)}{C}\right) - \frac{Nb^i}{S_i} + \frac{Y \cdot G_i(k)}{C}$
EQ 4 : $P_n = Lf_{m,i}$ et $tr_n = \frac{1}{2} \cdot \left(tr_p - \frac{Lf_{m,i} - P_p}{V_b}\right) \cdot \left(1 + \frac{G_i(k)}{C}\right)$

Avec ces dernière notations, l'algorithme de la fonction « g_feu » peut s'énoncer ainsi :

Fonction g_feu
Si $(A \mid (\overline{A} \times D \times E) \mid (\overline{A} \times B \times C \times \overline{D}))$ alors EQ1 ;
Si $(\overline{A} \times B \times \overline{C} \times \overline{D})$ alors EQ2 ;
Si $(\overline{A} \times \overline{B} \times \overline{D})$ alors EQ3 ;
Si $(\overline{A} \times D \times \overline{E})$ alors EQ4 ;

où « $\mid$ » désigne « ou » logique, $\times$ le « et » logique et $\overline{A}$ « la négation de A ».

Pour vérifier numériquement les résultats de ce modèle, nous avons réalisé un test en simulation illustré par la figure 3.6. Un bus est initialement à la position 0 au début d'un cycle de durée $C = 80$ secondes. Il doit traverser un carrefour à feu distant de 320 m et dont la durée du vert est G . Entre le bus et la ligne de feu, il y a nb véhicules.

Les simulations sont effectuées en faisant varier G de 0 (pas de vert ou rouge intégral) à 80 (pas de rouge ou vert intégral) et en faisant varier nb de 0 à 40 véhicules (le maximum que peut contenir le tronçon de 320 mètres). Les résultats du modèle sont comparés à un modèle plus réaliste qui tient compte des instants de début de la phase du vert. Pour ce dernier modèle, plusieurs tests sont réalisés afin de considérer la moyenne par rapport aux instants de début de vert générés de manière aléatoire. Le résultat est la position du bus à la fin du cycle qui est représentée sur la figure 3.7

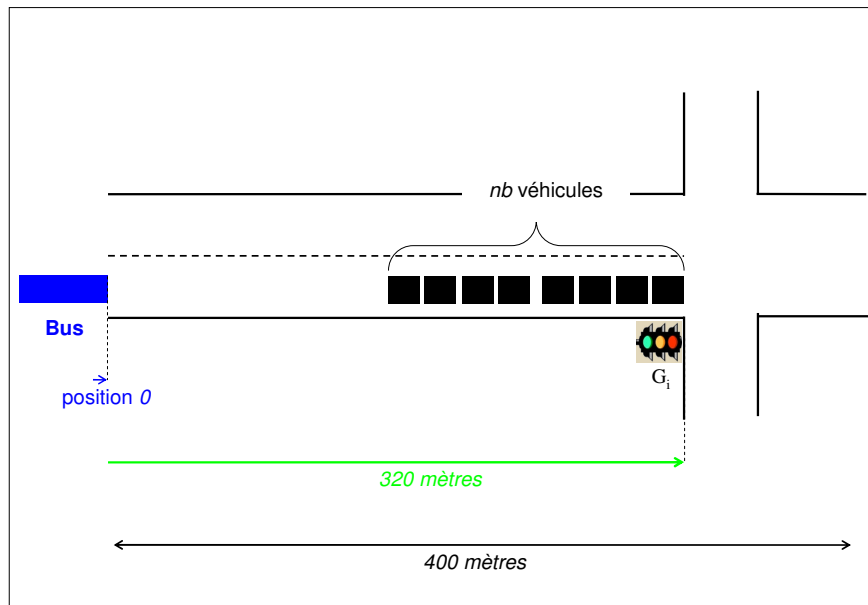


FIGURE 3.6 – Scénario de test

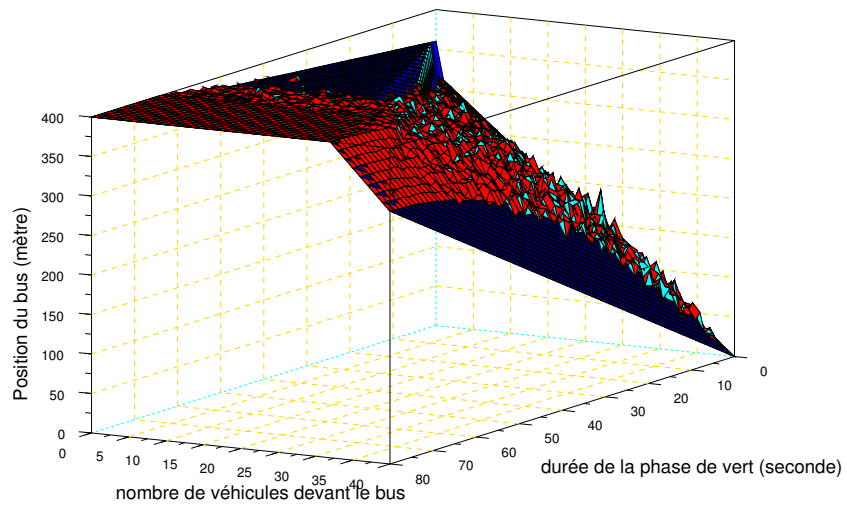


FIGURE 3.7 – Position du bus à la fin du cycle

Les courbes de cette figure montrent clairement deux zones symétriques séparées par la droite d'équation $G - 2 \times nb = 0$. Pour $G > \frac{nb}{S}$, où S est le taux de saturation de l'arc, le modèle est très proche du modèle réel où l'erreur est de l'ordre de 7%. Par contre pour $G \leq \frac{nb}{S}$, le modèle est un peu plus loin du modèle réel où l'erreur peut croître jusqu'à 20%. Le rapport $\frac{nb}{S}$ représente le temps nécessaire pour faire disparaître toute la file d'attente. Si la durée G de la phase du vert est supérieure $\frac{nb}{S}$ alors la file d'attente disparaît durant ce cycle et le modèle reproduit parfaitement le comportement du modèle le plus réaliste. Dans le cas contraire, la file d'attente ne disparaît pas durant ce cycle, le bus ne passe pas le carrefour et le modèle est moins précis. Cette erreur n'a pas de lourdes conséquences et ce pour les raisons suivantes :

- Le modèle est un modèle de commande avec une boucle fermée qui permettra de corriger les erreurs de modélisation.
- La stratégie qui se base sur ce modèle a pour objectif d'améliorer la progression des bus. Or l'imprécision provient du fait que le modèle sous-estime la progression du bus. Un modèle qui aurait sur-estimé la progression du bus aurait de plus graves conséquences.
- L'erreur se situe dans le cas où la file d'attente ne disparaît pas durant ce cycle ce qui correspond à une situation dégradée de la circulation. Or le but du travail est justement d'éviter les congestions sous l'action d'une régulation.

Ce premier modèle simple décrit la progression d'un bus dans un réseau routier urbain composé de carrefours et de stations. La dynamique d'un bus est représentée par sa position. L'évolution temporelle de cette dynamique dépend des durées des phases de vert des carrefours traversés par le bus, de leurs temps de stationnement dans les stations et aussi des longueurs de files d'attente dans les arcs traversés par le bus. Le modèle s'appuie sur des hypothèses qui ne sont pas toujours vérifiées dans la réalité. En outre, il ne tient pas compte des instants de début des phases qui déterminent les décalages entre carrefours. Dans la section suivante, nous décrivons un modèle plus complet et plus réaliste qui permet de s'affranchir de ces hypothèses et de considérer les décalages entre carrefours.

3.3 Modèle complet

Ce nouveau modèle se distingue de l'ancien, déjà présenté dans les sections précédentes, sur les deux points suivants :

- Il permet de tenir compte des instants de début des phases de vert pour les carrefours traversés par les lignes de transport en commun. Ceci revient à tenir compte des décalages entre les carrefours successifs.
- Il permet de tenir compte des cas où des files d'attente au niveau des carrefours remontent jusqu'aux stations des bus placées en amont.

Ces deux points vont compliquer davantage le modèle mais la philosophie reste la même. Au niveau de l'algorithme général de ce modèle, il diffère en ces trois points suivants :

- Les paramètres des plans de feu ne sont plus seulement les durées des phases mais aussi les instants de début des phases.
- La fonction « $g_station$ » ne dépend plus seulement des paramètres de la station mais aussi des plans du feu en aval.
- Le calcul de la file d'attente devant le bus ne se base plus simplement sur le modèle *store-and-forward* mais sur une hypothèse plus élaborée pour tenir compte des nouveaux paramètres des plans de feu.

L'algorithme général de ce modèle est le suivant :

Fonction générale du modèle

$P_p = P(k)$

$tr_p = C$

tant que ($tr_p > 0$) faire

 si (événement=station) alors

$(P_n, tr_n) = g_station(P_p, tr_p, G, Td, Nb, T_{st})$

 sinon

$(P_n, tr_n) = g_feu(P_p, tr_p, G, Td, Nb)$

 fin si

$P_p = P_n$

$tr_p = tr_n$

fin tant que

$P(k+1) = P_n$

Avant d'exposer la discussion générale sur le modèle, nous commençons par décrire le calcul de la file d'attente intercalée entre le bus et la ligne de feu.

3.3.1 Calcul de la longueur de la file d'attente devant le bus

Une des originalités de ce travail réside dans la capacité du modèle à reproduire simplement les interactions entre les véhicules de transport en commun qui sont les bus et les véhicules particuliers. Ces interactions sont perçues comme une interaction entre un bus et une file d'attente. Dans le chapitre précédent, la modélisation des files d'attente sur tout un arc a été décrite. Dans cette section, nous nous intéressons au calcul des files d'attente intercalées entre un bus et une ligne de feu.

Le bus est à la position P_p dans un arc " i " dont le début commence à la position Lf_1 et la fin à la position Lf_2 . Le cycle actuel est le cycle k et il reste tr secondes avant sa fin (figure 3.8).

Pour calculer le nombre de véhicules particuliers entre la ligne de feu et le bus dans la position P_p , nous

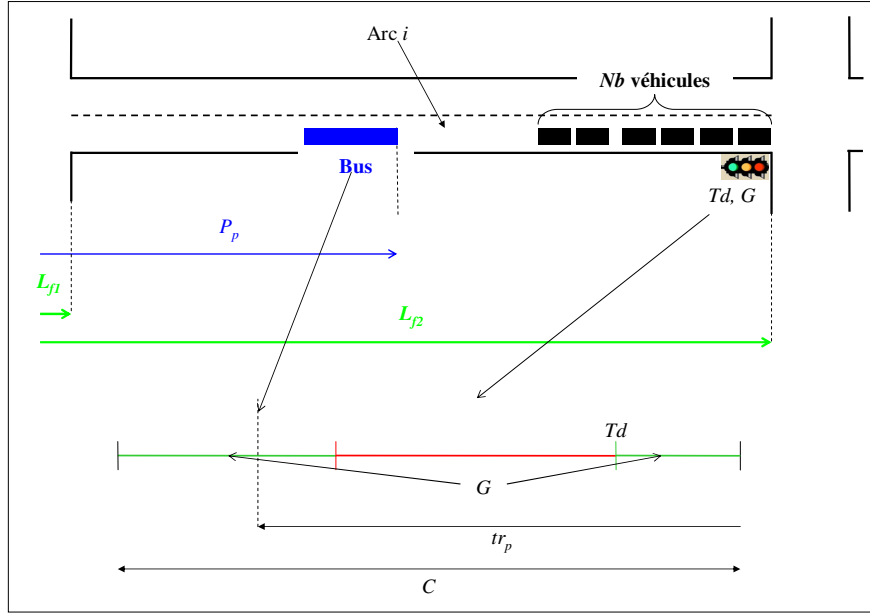


FIGURE 3.8 – Situation spatiale et temporelle du bus

considérons l'hypothèse suivante :

« Entre l'instant actuel et le début du cycle précédent (l'instant $(k-2) \times C$), le flux entrant dans l'arc se répartit uniformément sur toute sa longueur »

Nous pourrions aussi supposer, sans grande difficulté, que le bus est entré dans l'arc après le début du cycle précédent. En effet, dans le pire des cas où $tr = 0$, nous supposons que le bus n'a pas passé plus d'un cycle dans l'arc. Une hypothèse bien réaliste étant donné les longueurs moyenne entre deux feux successives, les vitesses des bus et la durée des cycles de feu. Partant de ces hypothèses, nous pouvons écrire :

$$nb = X_i((k-1) \cdot C) + E((k-2) \cdot C, k \cdot C - tr) \times \frac{Lf_2 - P_p}{Lf_2 - Lf_1} - S((k-2) \cdot C, k \cdot C - tr)$$

où

- $X_i((k-1) \cdot C)$ désigne le nombre de véhicules particuliers dans l'arc "i" à l'instant $(k-1) \times C$
- $E((k-2) \cdot C, k \cdot C - tr)$ et $S((k-2) \cdot C, k \cdot C - tr)$ sont respectivement le nombre de véhicules qui entrent dans l'arc et qui sortent de l'arc entre les instants $(k-2) \times C$ et $k \times C - tr$.

Le calcul des deux dernières variables se fait de la même manière que pour le modèle de file d'attente dans le chapitre 2 en se basant sur le modèle *store-and-forward*. Ceci donne l'équation finale suivante :

$$nb = X_i(k-1) + \sum_{j \in In_i} S_j \cdot [G_j(k-1) + g_j] \cdot \frac{Lf_2 - P_p}{Lf_2 - Lf_1} - S_i \cdot [G_i(k-1) + g_i]$$

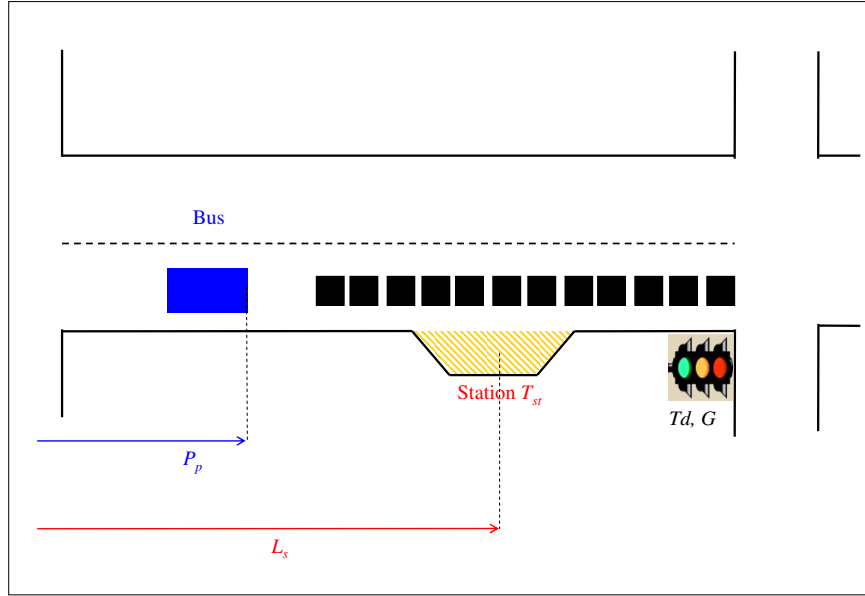


FIGURE 3.9 – Bus devant une station

Dans le cas où le bus est à l'entrée de l'arc " i ", ce calcul est parfaitement exact en considérant le modèle *store-and-forward*. Dans le cas où le bus est détecté au milieu de l'arc " i ", cette approximation peut introduire un biais surtout quand le bus passe plus d'un cycle dans le même arc ce qui se passe généralement dans le cas d'importantes congestions. Dans ce cas, le nombre de véhicules dans la file d'attente ne doit pas dépasser la valeur de $\frac{Lf_2 - P_p}{a}$. Ainsi, nous pouvons écrire l'expression finale du nombre de véhicules devant un bus :

$$nb = \min\left(\frac{Lf_2 - P_p}{a}; X_i(k-1) + \sum_{j \in In_i} S_j \cdot [G_j(k-1) + g_j] \cdot \frac{Lf_2 - P_p}{Lf_2 - Lf_1} - S_i \cdot [G_i(k-1) + g_i]\right)$$

3.3.2 Algorithme pour le passage d'une station

Par rapport à l'ancien algorithme du modèle simplifié, nous ne considérons plus l'hypothèse qu'aucune file d'attente ne remonte jusqu'à une station. Pour tenir compte de cette nouvelle possibilité, nous devons reproduire la dynamique des files d'attente qui dépend de l'état du feu du carrefour en aval de la station. La figure 3.9 illustre un cas général où un bus est devant une station suivi d'un carrefour à feu et où la file d'attente du feu remonte jusqu'à la station.

Le cas où la file d'attente ne remonte pas jusqu'à la station a été traité par l'ancien modèle simplifié et nous donnons le nom de « g_st1 » pour cet algorithme. Ce cas est réalisé quand $a \times nb \leq L_f - L_s$. Dans le cas contraire, c'est l'algorithme g_st2 qui mettra à jour la position du bus P_p et le temps restant tr_p . Il est clair que cette mise à jour dépend de l'évolution de la file d'attente qui elle-même dépend de sa longueur initiale et de l'état du feu à l'origine de celle-ci. Nous détaillons donc cet algorithme.

La première interrogation est de savoir si le bus aura la possibilité d'atteindre la file d'attente avant la fin du cycle actuel. Ce n'est pas le cas si $P_p + tr_p \times Vb < L_f - a \times nb$. Par conséquent, la nouvelle position P_n

et le temps restant tr_n sont donnés par l'équation simple suivante :

$$\left| \begin{array}{l} P_n = P_p + tr_p \cdot V_b \\ tr_n = 0 \end{array} \right.$$

Nous nous intéressons maintenant au cas contraire c'est à dire quand le bus aura la possibilité d'atteindre la file d'attente. Dans le premier chapitre, nous avons distingué deux types de configurations des plans de feu. Dans cette description, nous nous focalisons que sur la configuration 1. La même philosophie est applicable pour la configuration 2. Nous appelons respectivement g_st2_cfg1 et g_st2_cfg2 les algorithmes qui traitent respectivement le cas de la configuration 1 et la configuration 2 et nous nous intéressons seulement à l'algorithme g_st2_cfg1 .

Le bus est à la position P_p , la file d'attente remonte jusqu'à la station et le bus peut théoriquement atteindre la file d'attente pendant la période restante de durée tr_p . Le feu est supposé être de la première configuration c'est à dire tel que $T_d + G > C$. La première discussion consiste à savoir si à l'instant $C - tr_p$ l'arc bénéficie d'une période de vert ou pas. Pour cela, nous distinguons trois cas.

3.3.2.1 Cas 1 : $tr_p \leq C - T_d$

L'arc bénéficie d'une phase de vert et la durée du vert restante est tr_p . Le bus pourrait rejoindre la file d'attente après une période Y que nous avons calculé précédemment. D'autres événements peuvent se produire comme la fin du cycle actuel ou l'arrivée du bus dans la station et il est nécessaire, de ce fait, de discuter de leur antériorité ou de leur postériorité. Nous commençons tout d'abord par discuter quand le bus pourrait ou pas rejoindre la file d'attente avant la fin du cycle.

3.3.2.1.a Sous-cas 1.1 : $Y \leq tr_p$

Cette condition traduit le fait que le bus pourrait rejoindre la file d'attente avant la fin du cycle. À l'instant initial, la file d'attente remonte jusqu'à la station. Étant donné que l'arc bénéficie d'une phase de vert, la longueur de la file d'attente décroît. Deux possibilités peuvent survenir :

- Sous-cas 1.1.1 : la file d'attente est encore assez longue après Y secondes pour qu'elle dépasse la station, auquel cas, le bus la rejoint avant la station.
- Sous-cas 1.1.2 : La file d'attente décroît suffisamment pour qu'elle ne remonte plus jusqu'à la station après la période Y auquel cas le bus s'arrête dans la station avant de pouvoir rejoindre la file d'attente.

Le Sous-cas 1.1.2 se vérifie quand la condition suivante est réalisée $P_p + Y \times V_b \geq L_s$. Dans ce cas, le bus arrive sur la station sans rencontrer la file d'attente après $\frac{L_s - P_p}{V_b}$ secondes. Étant donné que le bus stationnera pour un temps T_{st} , à la sortie de la station la nouvelle position et le nouveau temps restant sont donnés par l'équation suivante :

$$\left| \begin{array}{l} P_n = L_s \\ tr_n = tr_p - \frac{L_s - P_p}{V_b} - T_{st} \end{array} \right.$$

Dans le Sous-cas 1.1.1, la longueur de la file d'attente dépasse encore la station et nous avons l'inégalité $P_p + Y \times V_b < L_s$. A l'instant où le bus rejoint la file d'attente, il est dans la position $P_p + Y \times V_b$ et il reste un temps $tr_p - Y$ secondes avant la fin du cycle. Le bus, étant dans la file d'attente, avancera avec la vitesse d'écoulement de celle-ci. A partir de cet instant et de cette position, deux événements sont à envisager : le bus atteint la station ou la fin de l'actuel cycle de feu. Ainsi, deux possibilités sont à distinguer selon que le bus rejoint la station avant ou après la fin du cycle.

Pour que la première situation se réalise, il faut que le nombre de véhicules quittant l'arc durant le temps restant $tr_p - Y$ soit suffisamment grand pour que la file d'attente à la fin du cycle ne remonte pas jusqu'à la station. La longueur de file d'attente qui disparaît jusqu'à la fin du cycle est $a \times S \times (tr_p - Y)$ (jusqu'à la fin du cycle, le feu est vert). La distance entre le bus quand il atteint la file d'attente et la station est $L_s - P_p - Y \times V_b$. Par conséquent, nous avons :

- Sous-cas 1.1.1.1 : la première situation se réalise quand la condition $a \times S \times (tr_p - Y) > L_s - P_p - Y \times V_b$ est satisfaite. Le bus atteint la station avant la fin du cycle avec la vitesse de la file d'attente $a \times S$ pour une distance $L_s - P_p - Y \times V_b$. Comme il stationne pour un temps T_{st} , Les nouvelles valeurs de la position et du temps restant sont données par :

$$\left| \begin{array}{l} P_n = L_s \\ tr_n = tr_p - Y - \frac{L_s - P_p - Y_b}{a \cdot S} - T_{st} \end{array} \right.$$

- Sous-cas 1.1.1.2 : le bus ne peut pas rejoindre la station avant la fin du cycle. Ceci se réalise quand la condition $a \times S \times (tr_p - Y) \leq L_s - P_p - Y \times V_b$ est satisfaite. Dans ce cas, le cycle se termine et le bus est encore dans la file d'attente. La nouvelle position et le nouveau temps restant sont donc donnés par :

$$\left| \begin{array}{l} P_n = P_p + Y \cdot V_b + a \cdot S \cdot (tr_p - Y) \\ tr_n = 0 \end{array} \right.$$

3.3.2.1.b Sous-cas 1.2 : $Y > tr_p$

Dans ce cas, le bus ne peut pas rejoindre la file d'attente avant la fin du cycle actuel. Initialement, la file d'attente comprend nb véhicules. Comme le feu est vert jusqu'à la fin du cycle, la file d'attente compte à la fin $nb - S \times tr_p$ véhicules.

- Sous-cas 1.2.1 : la nouvelle longueur de file d'attente est supérieure à la distance qui sépare la ligne du feu et la station alors la file d'attente remonte toujours vers la station. Mathématiquement ceci revient à $a \times (nb - S \times tr_p) > L_f - L_s$. Comme le bus ne rejoindra pas la file d'attente, ceci est synonyme que le bus roulera avec sa vitesse libre V_b jusqu'à la fin du cycle. Et donc la mise à jour est donnée par :

$$\left| \begin{array}{l} P_n = P_p + tr_p \cdot V_b \\ tr_n = 0 \end{array} \right.$$

– Sous-cas 1.2.2 : la file d'attente à la fin du cycle ne remonte pas jusqu'à la station. Deux événements peuvent se produire : le cycle se termine ou le bus arrive sur la station avant la fin du cycle. D'où les deux sous-cas suivants :

– Sous-cas 1.2.2.1 : le premier sous-cas se réalise si $P_p + tr_p \times V_b > L_s$ et nous avons alors :

$$\left| \begin{array}{l} P_n = P_p + tr_p \cdot V_b \\ tr_n = 0 \end{array} \right.$$

– Sous-cas 1.2.2.2 : ce sous-cas contraire se réalise quand $P_p + tr_p \times V_b \leq L_s$. Le bus arrive sur la station après $\frac{L_s - P_p}{V_b}$ et il stationne pendant une période T_{st} . Les nouvelles valeurs des variables qui nous intéressent sont données par :

$$\left| \begin{array}{l} P_n = L_s \\ tr_n = tr_p - \frac{L_s - P_p}{V_b} - T_{st} \end{array} \right.$$

Nous appelons $G_st_conf1_cas1$ l'algorithme qui traite le cas 1. Pour résumer, nous considérons les conditions et les équations suivantes vues au cours de la discussion précédente :

Les conditions	
A_1 :	« $(Vb - a \cdot S) \cdot tr_p + P_p + a \cdot Nb \geq L_f$ »
C_1 :	« $a \cdot Nb - a \cdot S \cdot tr_p > L_f - L_s$ »
D_1 :	« $p_p + tr_p \cdot Vb < L_s$ »
Les équations :	
EQ 1 :	$\left \begin{array}{l} P_n = L_s \\ tr_n = tr_p - \frac{L_s - P_p}{V_b} - T_{st} \end{array} \right.$
EQ 2 :	$\left \begin{array}{l} P_n = L_f + a \cdot S \cdot tr_p - a \cdot Nb \\ tr_n = 0 \end{array} \right.$
EQ 3 :	$\left \begin{array}{l} P_n = L_s \\ tr_n = tr_p - \frac{L_s - L_f + a \cdot Nb}{a \cdot S} - T_{st} \end{array} \right.$
EQ 4 :	$\left \begin{array}{l} P_n = P_p + tr_p \cdot Vb \\ tr_n = 0 \end{array} \right.$

Le pseudo-code de l'algorithme « $G_st_conf1_cas1$ » est le suivant :

Fonction $G_st_conf1_cas1$
Si $((A_1 \times B_1) \mid (\overline{A_1} \times \overline{C_1} \times \overline{D_1}))$ alors EQ1 ;
Si $(A_1 \times \overline{B_1} \times C_1)$ alors EQ2 ;
Si $(A_1 \times \overline{B_1} \times \overline{C_1})$ alors EQ3 ;
Si $(\overline{A_1} \times C_1) \mid (\overline{A_1} \times \overline{C_1} \times D_1)$ alors EQ4 ;

où $\mid$ signifie "ou", $\times$ "et" et $\overline{A}$ la négation de A .

Afin de vérifier la cohérence de l'algorithme, il est nécessaire de tester si toutes les possibilités ont été prises en compte. Pour cela il suffit de vérifier que $A_1 \times B_1 + \overline{A_1} \times \overline{C_1} \times \overline{D_1} + A_1 \times \overline{B_1} \times C_1 + A_1 \times \overline{B_1} \times \overline{C_1} + \overline{A_1} \times C_1 \mid (\overline{A_1} \times \overline{C_1} \times D_1 = 1$, ce qui est le cas.

3.3.2.2 Cas 2 : $C - T_d < tr_p \leq 2 \times C - T_d - G$

Le feu est rouge jusqu'à l'instant T_d . La file d'attente initiale reste immobile et garde la même longueur. Le bus roule donc de sa position P_p avec la vitesse V_b . Deux événements peuvent se produire : la phase du rouge se termine et ça sera à l'instant T_d ou le bus rejoint la file d'attente avant T_d . La distance que peut parcourir le bus jusqu'à T_d est $(T_d - C + tr_p) \times V_b$. Le bus rejoint la file d'attente avant T_d si $P_p + (T_d - C + tr_p) \times V_b > L_f - a \times nb$. Dans ce cas, le bus restera la position $L_f - a \times nb$ jusqu'à T_d puisque le feu est rouge et la file d'attente ne progresse pas. Dans le cas contraire, à l'instant T_d , le bus sera à la position $P_p + (T_d - C + tr_p) \times V_b$. En conclusion, à l'instant T_d , nous pouvons écrire les nouvelles valeurs intermédiaires de la position du bus et du temps restant avant la fin du cycle comme suit :

$$\left| \begin{array}{l} P_{p1} = \min(P_p + (T_d - C + tr_p) \times V_b; L_f - a \times nb) \\ tr_{p1} = C - T_d \end{array} \right.$$

La nouvelle situation est exactement la même que celle traitée par l'algorithme du premier cas « $G_st_conf1_cas1$ » avec la position initiale P_{p1} au lieu de P_p et le temps restant tr_{p1} au lieu de tr_p . Nous appelons « $G_st_conf1_cas2$ » l'algorithme qui traite ce deuxième cas et il est donné par :

$$(P_n, tr_n) = G_st_conf1_cas2(P_p, tr_p, nb)$$

$$\left| \begin{array}{l} P_{p1} = \min(P_p + (T_d - C + tr_p) \times V_b; L_f - a \times nb) \\ tr_{p1} = C - T_d \end{array} \right.$$

$$(P_n, tr_n) = G_st_conf1_cas1(P_{p1}, tr_{p1}, nb)$$

3.3.2.3 Cas 3 : $tr_p > 2 \times C - T_d - G$

Le feu est vert jusqu'à l'instant $2 \times C - T_d - G$. Le bus pourrait rejoindre la file d'attente après un temps $Y =$

3.3.2.3.a Sous-cas 3.1 : $Y \leq T_d + G + tr_p - 2 \times C$

Le bus pourrait rejoindre la file d'attente avant la fin de l'actuelle phase de vert. Il faut distinguer encore deux nouveaux sous-cas. Dans la perspective que le bus rejoigne la file d'attente, il est nécessaire de vérifier si cet événement prendra lieu avant ou après la station.

- Sous-cas 3.1.1 : pour que le bus rejoigne la file d'attente après la station il suffit que sa distance parcourue durant Y secondes soit supérieure à la distance qui sépare le bus et la station. Mathématiquement ceci se traduit par l'inégalité $Y \times V_b \geq L_s - P_p$ qui est équivalente à $P_p + Y \times V_b \geq L_s$. Dans ce cas, à la sortie de la station, la nouvelle position du bus P_n est celle de la station. Le nouveau temps restant est l'ancien temps restant auquel est soustrait la somme du temps nécessaire pour parcourir la distance $L_s - P_p$ et du temps de stationnement T_{st} . Les nouvelles valeurs des deux variables sont données par :

$$\left| \begin{array}{l} P_n = L_s \\ tr_n = tr_p - \frac{L_s - P_p}{V_b} - T_{st} \end{array} \right.$$

- Sous-cas 3.1.2 : il correspond à la situation où le bus rejoint la file d'attente avant la fin de l'actuelle phase de vert. À l'instant où le bus rejoint la file d'attente, il reste $tr_p - Y$ secondes avant la fin de cette phase et $tr_p + T_d + G - 2 \times C - Y$ avant la fin du cycle, le bus étant à la position $P_p + Y \times V_b$. Cette position coïncide avec la queue de la file d'attente. Par conséquent, la vitesse du bus devient celle de la file d'attente $V_f = a \times S$ où S est le taux de saturation de l'arc et ce jusqu'à la fin de la phase actuelle. L'événement à venir est soit la fin de cette phase du vert soit l'entrée du bus dans la station. Pour savoir lequel de ces événements se produira le premier, il suffit de comparer $a \times S \times (tr_p + T_d + G - 2 \times C)$ et $L_s - P_p - Y \times V_b$. Le terme $a \times S \times (tr_p + T_d + G - 2 \times C)$ est la distance que parcourt le bus avec la vitesse $a \times S$ jusqu'à la fin de la phase actuelle de vert. Le terme $L_s - P_p - Y \times V_b$ représente la distance qui sépare la station et le bus à l'instant où il rejoint la file d'attente.

- Sous-cas 3.1.2.1 : c'est le cas où le premier terme est supérieur au deuxième, le bus atteindra la station avant la fin de l'actuelle phase et par conséquent, à la sortie de la station, nous pouvons écrire les nouvelles valeurs de la position et du temps restant :

$$\left| \begin{array}{l} P_n = L_s \\ tr_n = tr_p - Y - \frac{L_s - P_p - Y \cdot V_b}{a \cdot S} - T_{st} \end{array} \right.$$

- Sous-cas 3.1.2.2 : le bus n'atteindra pas la station avant la fin de la phase actuelle. A cet instant, le bus est dans la position $P_{p1} = P_p + Y \times V_b + a \times S \times (tr_p + T_d + G - 2 \times C)$ dans une file d'attente qui compte un nombre nb_1 véhicules donné par

$$nb_1 = nb - S \times (tr_p + T_d + G - 2 \times C)$$

Le bus se retrouve alors dans la situation traitée par l'algorithme « G_st_conf1_cas2 » du deuxième cas avec comme position initiale P_{p1} , temps restant initial $tr_{p1} = 2 \times C - T_d - G$ et une file d'attente initiale contenant nb_1 véhicules. Par conséquent, nous pouvons écrire :

$$\left| \begin{array}{l} P_{p1} = P_p + Y \times V_b + a \times S \times (tr_p + T_d + G - 2 \times C) \\ tr_{p1} = 2 \times C - T_d - G \\ nb_1 = nb - S \times (tr_p + T_d + G - 2 \times C) \\ (P_n, tr_n) = G_st_conf1_cas2(P_{p1}, tr_{p1}, nb_1) \end{array} \right.$$

3.3.2.3.b Sous-cas 3.2 : $Y > T_d + G + tr_p - 2 \times C$

Dans ce cas, le bus ne peut pas atteindre la file d'attente avant la fin de l'actuelle phase de vert. De cet instant initial et jusqu'à la fin de cette phase, la longueur de la file d'attente continue de décroître puisque des véhicules continuent de quitter l'arc. A la fin de la phase du vert, le nombre de véhicules dans la file d'attente est :

$$nb_1 = nb - S \times (T_d + G + tr_p - 2 \times C)$$

Il est ici nécessaire de vérifier si cette nouvelle file d'attente remonte toujours jusqu'à la station. Ceci est le cas quand $a \times nb_1 > L_f - L_s$ et nous avons alors les deux nouveaux sous-cas :

- Sous-cas 3.2.1 : à la fin de l'actuelle phase de vert, la position du bus, le temps restant avant la fin du cycle et le nombre de véhicules dans la file d'attente sont donnés par :

$$\left| \begin{array}{l} P_{p1} = P_p + V_b \times (tr_p + T_d + G - 2 \times C) \\ tr_{p1} = 2 \times C - T_d - G \\ nb_1 = nb - S \times (T_d + G + tr_p - 2 \times C) \end{array} \right.$$

La fin de la phase de vert courante correspond au début de la phase du rouge. La situation du bus à cet instant est exactement la même que celle traitée dans le deuxième cas avec comme variables initiales P_{p1}, tr_{p1} et nb_1 . Par conséquent, pour obtenir les valeurs finales de la position du bus et du temps restant à la sortie de la station, il suffit d'appliquer l'algorithme « G_st_conf1_cas2 » aux variables P_{p1}, tr_{p1} et nb_1 . D'où les équations suivantes :

$$\left| \begin{array}{l} P_{p1} = P_p + V_b \times (tr_p + T_d + G - 2 \times C) \\ tr_{p1} = 2 \times C - T_d - G \\ nb_1 = nb - S \times (T_d + G + tr_p - 2 \times C) \\ (P_n, tr_n) = G_st_conf1_cas2(P_{p1}, tr_{p1}, nb_1) \end{array} \right.$$

- Sous-cas 3.2.2 : c'est le cas où la nouvelle longueur de la file d'attente a suffisamment diminué pour ne plus remonter jusqu'à la station, deux nouvelles situations doivent être distinguées selon que le bus pourrait atteindre ou pas, durant tr_p , la station.

- Sous-cas 3.2.2.1 : le bus serait en mesure de d'atteindre la station durant tr_p si $P_p + tr_p \times V_b \geq L_s$ et nous avons alors :

$$\left| \begin{array}{l} P_n = L_s \\ tr_n = tr_p - \frac{L_s - P_p}{V_b} - T_{st} \end{array} \right.$$

- Sous-cas 3.2.2.2 : le bus ne pourra pas atteindre la file d'attente avant la fin du cycle. Par conséquent, la nouvelle position et le nouveau temps restant sont donnés par :

$$\left| \begin{array}{l} P_n = P_p + tr_p \cdot V_b \\ tr_n = 0 \end{array} \right.$$

Pour résumer la discussion sur ce dernier cas, nous considérons les conditions et les équation suivantes :

Les conditions	
A_3 :	$\ll P_p + a \cdot Nb + (Vb - a \cdot S) \cdot tr_p \geq L_f + (2 \cdot C - T_d - G - tr_p)(Vb - a \cdot S) \gg$
B_3 :	$\ll a \cdot S \cdot P_p + a \cdot Vb \cdot Nb \leq Vb \cdot L_f - (Vb - a \cdot S) \cdot L_s \gg$
C_3 :	$\ll a \cdot S \cdot P_p + a \cdot Vb \cdot Nb + (L_s + a \cdot S \cdot (2 \cdot C - T_d - G - tr_p) \cdot (Vb - a \cdot S) \geq Vb \cdot L_f \gg$
D_3 :	$\ll a \cdot Nb - a \cdot S \cdot tr_p > L_f - L_s - a \cdot S(2 \cdot C - T_d - G) \gg$
E_3 :	$\ll P_p + tr_p \cdot Vb < L_s \gg$
Les équations	
EQ 1 :	$P_n = L_s$ $tr_n = tr_p - \frac{L_s - P_p}{V_b} - T_{st}$ $P_{p1} = P_p + Y \cdot Vb + a \cdot S \cdot (tr_p + G + T_d - 2 \cdot C)$
EQ 2 :	$tr_{p1} = 2 \cdot C - T_d - G$ $Nb_1 = Nb - S \cdot (tr_p + G + T_d - 2 \cdot C)$ $(P_n, tr_n) = G_st_conf1_cas2(P_{p1}, tr_{p1}, G, T_d, Nb_1, T_{st})$
EQ 3 :	$P_n = L_s$ $tr_n = tr_p - \frac{L_s - L_f + a \cdot Nb}{a \cdot S} - T_{st}$ $P_{p1} = P_p + (tr_p + G + T_d - 2 \cdot C) \cdot Vb$
EQ 4 :	$tr_{p1} = 2 \cdot C - T_d - G$ $Nb_1 = Nb - S \cdot (tr_p + G + T_d - 2 \cdot C)$ $(P_n, tr_n) = G_st_conf1_cas2(P_{p1}, tr_{p1}, G, T_d, Nb_1, T_{st})$
EQ 5 :	$P_n = P_p + tr_p \cdot Vb$ $tr_n = 0$

Ainsi, l'algorithme « G_st_conf1_cas3 » est donné comme suit :

Fonction « G_st_conf1_cas3 »

Si $((A_3 \times B_3) \mid (\overline{A_3} \times \overline{D_3} \times \overline{E_3}))$ alors EQ1 ;
 Si $(A_3 \times \overline{B_3} \times C_3)$ alors EQ2 ;
 $(A_3 \times \overline{B_3} \times \overline{C_3})$ alors EQ3 ;
 Si $(\overline{A_3} \times D_3)$ alors EQ4 ;
 Si $(\overline{A_3} \times \overline{D_3} \times E_3)$ alors EQ5 ;

Les 3 cas possibles sont présentés, l'algorithme de mise à jour de la position du bus et du temps restant pour la configuration 1 des feux est le suivant :

Fonction « G_st_conf1 »

si $(tr_p \leq C - T_d)$ alors
 $(P_n, tr_n) = G_st_conf1_cas1(P_p, tr_p, nb)$
 sinon
 si $(tr_p \leq 2 \times C - T_d - G)$ alors
 $(P_n, tr_n) = G_st_conf1_cas2(P_p, tr_p, nb)$
 sinon
 $(P_n, tr_n) = G_st_conf1_cas3(P_p, tr_p, nb)$
 fin si
 fin si

3.3.2.4 Algorithme pour la configuration 2 des feux

Nous avons traité, dans la section précédente, le cas des plans de feux dont la configuration est de type 1. Pour le cas d'un feu de configuration 2 (figure 3.10), la même approche est appliquée.

Comme dans le cas de la configuration 1, trois situations sont à distinguer selon la localisation de tr_p dans le cycle de feu de durée C .

3.3.2.4.a Cas 1 : $tr_p < C - T_d - G$

La phase actuelle des feux est rouge. Puisque la file d'attente dépasse la station et que sa longueur restera la même jusqu'à la fin du cycle, le bus pourra au mieux rejoindre la queue de cette file avant la fin du cycle. Dans le cas contraire, il parcourt la distance $tr_p \times V_b$. Dans les deux cas, le nouveau temps restant est 0 puisqu'il est certain que le bus n'atteindra pas la station avant la fin du cycle. L'algorithme $G_st_conf2_cas1$ qui traite ce premier cas est le suivant :

Fonction « G_st_conf2_cas1 »

$$\left| \begin{array}{l} P_n = \min(L_f - a \times nb; P_p + tr_p \cdot V_b) \\ tr_n = 0 \end{array} \right.$$

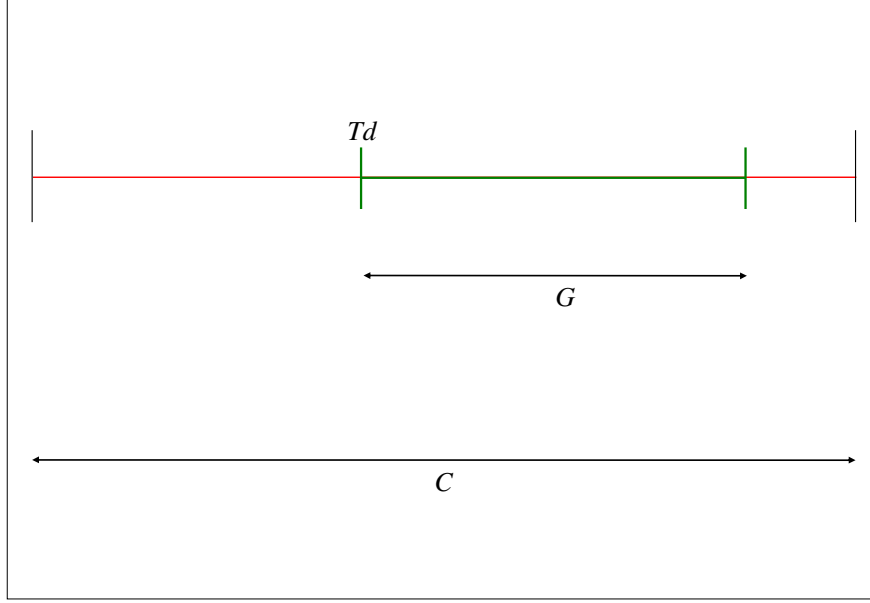


FIGURE 3.10 – Configuration 2 des plans de feu

3.3.2.4.b Cas 2 : $C - T_d - G \leq tr_p \leq C - T_d$

La phase courante est le vert. La discussion est similaire aux discussions des algorithmes « G_st_conf1_cas1 » et « G_st_conf1_cas3 » et de ce fait, nous ne présenterons pas les détails. Comme pour les deux algorithmes cités, cet algorithme *G_st_conf2_cas2* comprend 5 situations différentes et donc 5 équations. Afin de récapituler, nous présentons la synthèse suivante :

Les conditions

$$A_5 : \langle P_p + a \cdot Nb + (Vb - a \cdot S) \cdot tr_p \geq L_f + (C - T_d - G)(Vb - a \cdot S) \rangle$$

$$B_5 : \langle a \cdot S \cdot P_p - Vb \cdot nb \geq (Vb - a \cdot S) \cdot L_s - Vb \cdot L_f \rangle$$

$$C_5 : \langle a \cdot nb - a \cdot S \cdot tr_p > L_f - L_s - a \cdot S \cdot (C - T_d - G) \rangle$$

$$D_5 : \langle a \cdot nb - a \cdot S \cdot (tr_p + T_d + G - C) > L_f - L_s \rangle$$

$$E_5 : \langle P_p + tr_p \cdot Vb < L_s \rangle$$

Les équations

EQ1 :	$P_n = L_s \cdot V_b$ $tr_n = tr_p - T_{st} - \frac{L_s - P_p}{V_b}$
EQ2 :	$P_n = L_f - a \cdot nb + a \cdot S \cdot (tr_p + T_d + G - C)$ $tr_n = 0$
EQ3 :	$P_n = L_s \cdot V_b$ $tr_n = tr_p - T_{st} - \frac{L_s - L_f + a \cdot nb}{a \cdot S}$
EQ4 :	$P_n = \min(P_p + tr_p \cdot V_b; L_f - a \cdot nb + a \cdot S \cdot (tr_p + T_d + G - C))$ $tr_n = 0$
EQ5 :	$P_{p1} = P_p + V_b \cdot (tr_p + T_d + G - C)$ $tr_{p1} = C - T_d - G$ $nb_1 = nb - S \cdot (tr_p + T_d + G - C)$ $(P_n, tr_n) = G_st_conf2_cas1(P_{p1}, tr_{p1}, nb_1)$

Avec cette dernière synthèse, l'algorithme « G_st_conf2_cas2 » s'énonce de la manière suivante :

Fonction « G_st_conf2_cas2 »

Si $((A_5 \times B_5) \mid (\overline{A_5} \times \overline{D_5} \times \overline{E_5}))$ alors EQ1 ;
Si $(A_5 \times \overline{B_5} \times C_5)$ alors EQ2 ;
Si $(A_5 \times \overline{B_5} \times \overline{C_5})$ alors EQ3 ;
Si $(\overline{A_5} \times D_5)$ alors EQ4 ;
Si $(\overline{A_5} \times \overline{D_5} \times E_5)$ alors EQ5 ;

3.3.2.4.c Cas 3 : $tr_p > C - T_d$

La phase actuelle est rouge jusqu'à l'instant T_d . De ce fait, la longueur de la file d'attente ne change pas et dépasse toujours la station. À la fin de l'actuelle phase de rouge, le bus atteint la queue de la file d'attente si la distance qu'il peut parcourir en $tr_p + T_d - C$ secondes le permet. Dans le cas contraire sa position est $P_p + V_b \times tr_p + T_d - C$. A la fin de cette phase, la position et le temps restant sont donnés par les équations suivantes :

$$\left| \begin{array}{l} P_{p1} = \min(L_f - a \cdot nb; P_p + V_b \cdot (tr_p + T_d - C)) \\ tr_{p1} = C - T_d \end{array} \right.$$

À cet instant, la situation est exactement la même que celle traitée par le précédent algorithme « G_st_conf2_cas2 » mais avec comme variables initiales la position P_{p1} , le temps restant tr_{p1} et le nombre de véhicules dans la file d'attente nb qui ne change pas. L'algorithme « G_st_conf2_cas3 » est donné par

Fonction « G_st_conf2_cas3 »

$$\left| \begin{array}{l} P_{p1} = \min(L_f - a \cdot nb; P_p + V_b \cdot (tr_p + T_d - C)) \\ tr_{p1} = C - T_d \\ (P_n, tr_n) = G_st_conf2_cas2(P_{p1}, tr_{p1}, nb) \end{array} \right|$$

3.3.3 Algorithme pour le passage d'un carrefour à feux

La modélisation du passage d'une ligne de feu reprend les mêmes bases que celle du passage d'une station. La figure 3.8 illustre la situation du bus devant un carrefour à feux. Le feu de signalisation peut avoir deux configurations distinctes. Les algorithmes relatifs à la configuration 1 et 2 sont respectivement « G_feu_conf1 » et « G_feu_conf2 ». L'algorithme général de la modélisation du passage d'un carrefour à feux peut être énoncé comme suit :

Fonction « G_feu »

si $(T_d + G > C)$ alors

$$(P_n, tr_n) = G_feu_conf1(P_p, tr_p, nb)$$

sinon

$$(P_n, tr_n) = G_feu_conf2(P_p, tr_p, nb)$$

fin si

Nous nous traiterons ici que la configuration 1. Le traitement de la configuration 2 est immédiat et repose sur les mêmes fondements que la configuration 1. Trois cas sont distingués selon que l'instant actuel se situe dans l'une des trois phases du cycle de feu.

3.3.3.1 Cas 1 : $tr_p \leq C - T_d$

Le feu est vert jusqu'à la fin du cycle. Quatre situations sont envisagées et qui résultent en quatre équations de mise à jour de la position du bus et du temps restant.

- Situation 1 : le bus ne quitte pas l'arc avant la fin du cycle et ne rejoint pas la file d'attente. Par conséquent, la nouvelle position du bus et le nouveau temps restant sont donnés par l'équation *EQ1* :

$$\left| \begin{array}{l} P_n = P_p + tr_p \cdot V_b \\ tr_n = 0 \end{array} \right|$$

- Situation 2 : le bus ne quitte pas l'arc avant la fin du cycle mais rejoint la file d'attente. La mise à jour est donnée par l'équation *EQ2* :

$$\left| \begin{array}{l} P_n = L_f - a \cdot (nb - S \cdot tr_p) \\ tr_n = 0 \end{array} \right|$$

- Situation 3 : le bus quitte l'arc en ne rejoignant pas la file d'attente. L'équation *EQ3* donne les nouvelles valeurs de la position et du temps restant :

$$\left| \begin{array}{l} P_n = L_f \\ tr_n = tr_p - \frac{L_f - P_p}{V_b} \end{array} \right.$$

- Situation 4 : le bus quitte l'arc après avoir rejoint la file d'attente. La mise à jour des deux variables est donnée par l'équation *EQ4* suivante :

$$\left| \begin{array}{l} P_n = L_f \\ tr_n = tr_p - \frac{nb}{S} \end{array} \right.$$

La mise à jour parmi les quatre situations dépend des conditions qui dépendent de tous les paramètres comme les variables dynamiques P_p , tr_p et nb et les variables du plan de feu T_d et G . Les autres paramètres sont les données géométriques. Ces conditions sont récapitulées comme suit :

Condition *A* : $V_b \cdot tr_p + P_p + a \cdot nb \leq L_f$

Condition *B* : $\frac{V_b \cdot nb}{S} + P_p < L_f$

Condition *C* : $tr_p \cdot V_b + P_p \leq L_f$

Condition *D* : $nb > S \cdot tr_p$

Condition *E* : $(V_b - a \cdot S) \cdot tr_p + P_p + a \cdot nb \leq L_f$

L'algorithme « G_feu_conf1_cas1 » peut s'énoncer comme suit :

Fonction « G_feu_conf1_cas1 »

si $((A_1 \times B_1) \mid (\overline{A_1} \times B_1 \times C_1) \mid (\overline{A_1} \times \overline{B_1} \times D_1 \times E_1))$ alors EQ1

si $(A_1 \times \overline{B_1} \times D_1 \times \overline{E_1})$ alors EQ2

si $(\overline{A_1} \times B_1 \times \overline{C_1})$ alors EQ3

si $(\overline{A_1} \times \overline{B_1} \times \overline{D_1})$ alors EQ4

3.3.3.2 Cas 2 : $C - T_d < tr_p \leq 2 \cdot C - T_d - G$

Dans ce cas, le feu est rouge jusqu'à l'instant T_d . Trois situations existent et sont :

- Jusqu'à l'instant T_d , le bus roule avec sa vitesse libre V_b sans rejoindre la file d'attente. A l'instant T_d , sa position est P_{p1} et le temps restant est $tr_{p1} = C - T_d$. Cette nouvelle situation peut être traitée par l'algorithme du premier cas avec comme paramètres d'entrée P_{p1} , tr_{p1} et nb qui ne change pas. D'où l'équation suivante :

$$\left| \begin{array}{l} P_{p1} = P_p + V_b \cdot (tr_p + T_d - C) \\ tr_{p1} = C - T_d \\ (P_n, tr_n) = G_feu_conf1_cas1(P_{p1}, tr_{p1}, nb) \end{array} \right.$$

- Le bus ne pourra pas quitter l’arc durant ce cycle et restera bloqué dans la file d’attente. Les équations de mise à jour de sa position et du temps restant sont données par :

$$\left| \begin{array}{l} P_n = L_f - a \cdot (nb - S \cdot (C - T_d)) \\ tr_n = 0 \end{array} \right.$$

- Le bus quitte l’arc et nous avons les équations suivantes :

$$\left| \begin{array}{l} P_n = L_f \\ tr_n = tr_p - T_d - \frac{nb}{S} \end{array} \right.$$

Les conditions qui permettent de se situer dans une des trois situations précédentes sont les suivantes :

- Condition A : $V_b \times tr_p + P_p + a \times nb \leq L_f + V_b \times (C - T_d)$
- Condition B : $nb > S \times (C - T_d)$

Avec les équations et les conditions ainsi présentées, l’algorithme « G_feu_conf1_cas2 » est le suivant :

Fonction « G_feu_conf1_cas2 »
Si (A_2) alors EQ1
Si $(\overline{A_2} \times B_2)$ alors EQ2
Si $(\overline{A_2} \times \overline{B_2})$ alors EQ3

3.3.3.3 Cas 3 : $2 \cdot C - T_d - G < tr_p$

Le feu est vert jusqu’à l’instant $T_d + G - C$. Quatre situations sont à distinguer :

- Le bus roule avec sa vitesse libre sans rejoindre la file d’attente jusqu’à la fin de l’actuelle phase de vert. Durant cette période, la longueur de file d’attente décroît et à sa fin, elle ne contient que nb_1 véhicules. A la fin de la phase du vert, la situation du bus est similaire à sa situation traitée par l’algorithme précédent « G_feu_conf1_cas2 ». Les nouvelles valeurs de la position et du temps restant sont données par l’équation EQ1 suivante :

$$\left| \begin{array}{l} P_{p1} = P_p + V_b \cdot (tr_p + T_d + G - 2 \cdot C) \\ tr_{p1} = 2 \cdot C - T_d - G \\ nb_1 = nb - \min(nb; S \cdot (tr_p + T_d + G - 2 \cdot C)) \\ (P_n, tr_n) = G_feu_conf1_cas2(P_{p1}, tr_{p1}, nb_1) \end{array} \right.$$

- Le bus roule jusqu'à la ligne de feu avec sa vitesse libre et quitte l'arc. Nous avons alors l'équation *EQ2* suivante :

$$\left| \begin{array}{l} P_n = L_f \\ tr_n = tr_p - \frac{L_f - P_p}{V_b} \end{array} \right.$$

- Le bus quitte l'arc mais après avoir rejoint la file d'attente. L'équation *EQ3* donne la mise à jour des deux variables :

$$\left| \begin{array}{l} P_n = L_f \\ tr_n = tr_p - \frac{nb}{S} \end{array} \right.$$

- Le bus rejoint la file d'attente avant la fin de l'actuelle phase de vert mais ne quitte pas l'arc durant cette période. La file d'attente décroît et à la fin de cette période, elle ne contient que nb_1 véhicules. A la fin de cette phase, le bus est dans la même situation que celle du deuxième cas traitée par l'algorithme « G_feu_conf1_cas2 ». L'équation *EQ4* met à jour la position du bus et le temps restant :

$$\left| \begin{array}{l} P_{p1} = L_f - a \cdot nb + a \cdot S \cdot (tr_p + T_d + G - 2 \cdot C) \\ tr_{p1} = 2 \cdot C - T_d - G \\ nb_1 = nb - \min(nb; S \cdot (tr_p + T_d + G - 2 \cdot C)) \\ (P_n, tr_n) = G_feu_conf1_cas2(P_{p1}, tr_{p1}, nb_1) \end{array} \right.$$

Les conditions qui permettent de se situer dans l'une des quatre situations sont les suivantes :

$$\begin{array}{l} A : V_b \cdot tr_p + P_p + a \cdot nb \leq L_f + V_b \cdot (2 \cdot C - T_d - G) \\ B : (V_b - a \cdot S) \cdot (T_d + G + tr_p - 2 \cdot C) + P_p + a \cdot nb < L_f \\ C : V_b \cdot (tr_p + T_d + G - 2 \cdot C) + P_p \leq L_f \\ D : \frac{V_b \cdot nb}{S} + P_p < L_f \\ E : T_d + G + tr_p - 2 \cdot C \geq \frac{nb}{S} \end{array}$$

L'algorithme « G_feu_conf1_cas3 » peut s'écrire de la manière suivante :

Fonction G_feu_conf1_cas3

Si $((A_3) \mid (\overline{A_3} \times B_3 \times C_3))$ alors EQ1

Si $((\overline{A_3} \times B_3 \times \overline{C_3}) \mid (\overline{A_3} \times \overline{B_3} \times \overline{C_3} \times D_3))$ alors EQ2

Si $(\overline{A_1} \times \overline{B_3} \times \overline{D_3} \times E_3)$ alors EQ3

Si $(\overline{A_1} \times \overline{B_3} \times \overline{D_3} \times \overline{E_3})$ alors EQ4

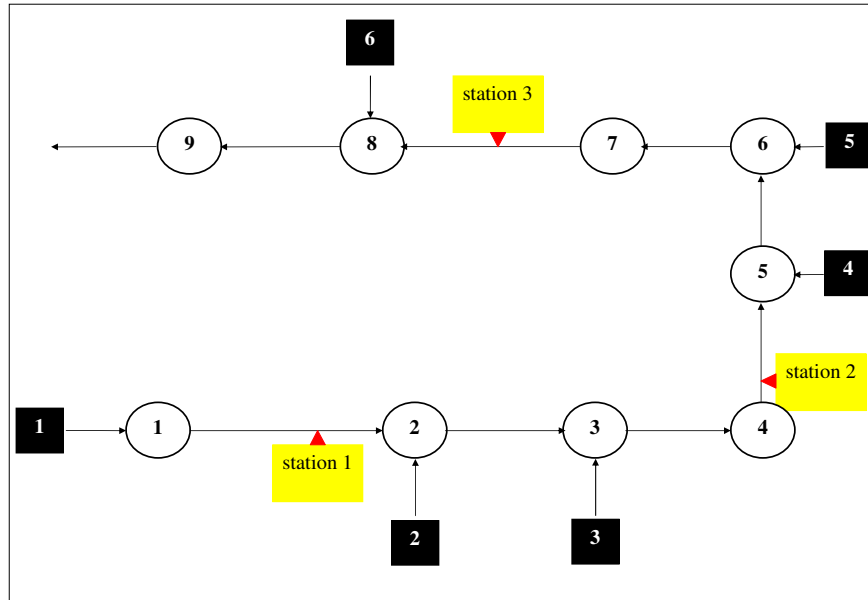


FIGURE 3.11 – Réseau de test virtuel

3.4 Étude en simulation

3.4.1 Description du réseau et des flux

Les deux modèles présentés précédemment décrivent la progression des véhicules de transport en commun dans un réseau urbain traversé par des flux de véhicules et contrôlé par des feux de signalisation. Le premier ne considère que les durées des phases de vert comme seules variables influençant la progression des véhicules de transport en commun. Les valeurs des décalages entre les carrefours sont supposées inconnues. Le deuxième modèle est plus fin et plus complet et intègre les décalages entre les carrefours comme variables supplémentaires. Afin de vérifier la cohérence, la pertinence des modèles et analyser les différences entre les deux, nous considérons des tests en simulation réalisés sur un réseau virtuel simple.

Le réseau est constitué de 9 carrefours inter-connectés avec 6 entrées (figure 3.11). Il comprend 14 arcs et est traversé par une ligne de bus qui parcourt tout le réseau de l'entrée 1 jusqu'à la sortie du carrefour 9.

La ligne de bus traverse dans l'ordre tous les carrefours du 1 au 9 et stationne dans 3 stations. L'entrée de la ligne est prise comme référence spatiale. Sur le tableau 3.1 sont données les distances des lignes de feu consécutifs et des stations par rapport à la référence spatiale ainsi que les durées de stationnement dans les stations.

Les simulations sont effectuées sur un horizon temporel de 40 cycles avec un cycle de durée 80 secondes. Pour les 10 premiers cycles, la charge du trafic est modérée. Pour les 30 cycles restants, nous introduisons des perturbations comme dans les simulations du chapitre 2. Les perturbations sont introduites grâce à 4

carrefours et stations	entrée 1	1	station 1	2	3	4	station 2	5	6	7	station 3	8	9
positions (m)	0	200	450	500	750	1200	1260	1450	1660	1730	1880	2030	2350
temps de stationnement (s)	—	—	20	—	—	—	15	—	—	—	17	—	—

TABLE 3.1 – Positions des carrefours et des stations

Entrées	1	2	3	4	5	6
flux moyens (nombre de VP/cycle)	21	25	22	18	18	21

TABLE 3.2 – Flux moyens

coefficients. La charge normale du trafic est donnée par les valeurs moyennes des flux entrants suivants :

Les perturbations sont données par le tableau suivant :

Nous avons considéré trois situations de trafic : trafic léger, trafic normal et trafic chargé. Le passage de l'une à l'autre se fait grâce aux 4 coefficients a_1 , a_2 , b_1 et b_2 . Pour les trois situations de trafic, les valeurs des coefficients sont données comme suit :

- trafic léger : $a_1 = 0.8$, $a_2 = 0.8$, $b_1 = 0.6$ et $b_2 = 0.5$
- trafic normal : $a_1 = 1.2$, $a_2 = 1.3$, $b_1 = 0.8$ et $b_2 = 0.6$
- trafic chargé : $a_1 = 2.1$, $a_2 = 2.4$, $b_1 = 1.65$ et $b_2 = 1.5$

Le réseau comporte 9 carrefours simples à deux phases. Pour caractériser le plan de feu d'un carrefour, il suffit de spécifier la durée d'une phase et son instant de début. Le premier modèle ne nécessite que la connaissance de la durée de la phase. Nous avons choisi de diviser le vert équitablement entre les deux phases. Ainsi, toutes les phases ont une durée de 40 secondes. Pour les instants de début des phases, elles sont déterminées de façon à garantir une "*onde verte*" pour les bus compte tenu de leur vitesse libre et des distances qui séparent les carrefours successifs. Le tableau 3.4 récapitule les instants de début des phases de vert pour les bus de l'ensemble des les carrefours.

périodes	perturbations
cycles 11-20	entrées 4 et 6 multipliées par a_1
cycles 21-30	entrées 1 et 5 multipliées par a_2 et entrées 2 et 3 par b_1
cycles 31-40	entrées 1, 4, 5 et 6 multipliées par b_2

TABLE 3.3 – Perturbations des flux

carrefours	1	2	3	4	5	6	7	8	9
instant de début (s)	25	2	33	9	55	1	10	65	25

TABLE 3.4 – Plans de feux des carrefours

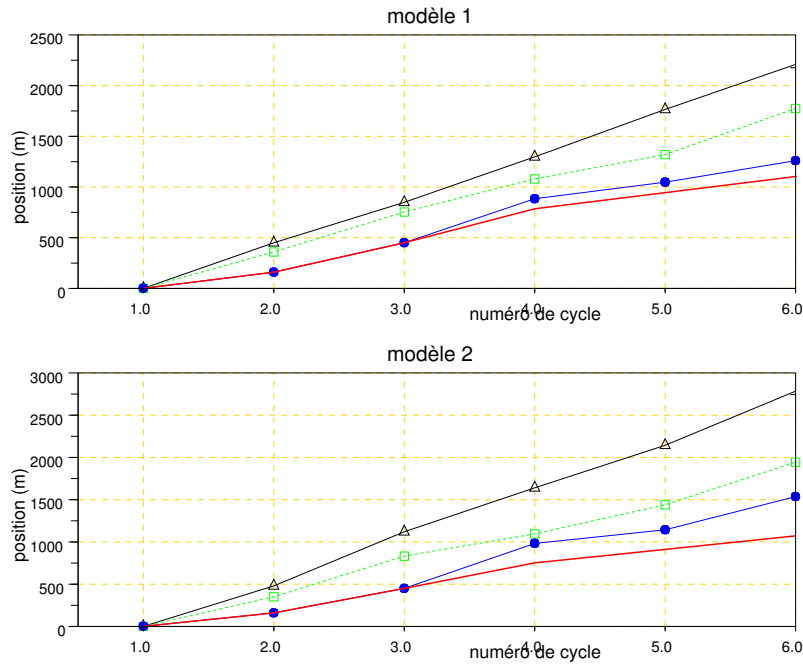


FIGURE 3.12 – Influence de la charge du trafic

3.4.2 Influence de la charge du trafic

La progression des véhicules de transport en commun dans un réseau urbain dépend de la charge du trafic sur les régions traversées par ceux-ci. Plus le trafic est dense plus la progression est difficile. Afin de vérifier le comportement des deux modèles en fonction de la charge du trafic, nous avons réalisé les tests pour trois scénarii sur le trafic comme expliqués dans la section précédente. Sur la figure 3.12, nous avons tracé l'évolution de la position d'un bus sur 7 cycles pour les 3 scénarii et en comparaison avec un scénario où le trafic est nul et les carrefours sont en accès libre (le cas le plus favorable pour la progression des bus). La première figure donne les résultats pour le premier modèle simple et la deuxième pour le modèle plus complet.

Les résultats pour les deux modèles confirment la dépendance de la progression des véhicules de transport en commun envers la charge du trafic. Cette dépendance semble légèrement plus accentuée pour le deuxième que le premier. Sur la figure 3.13, nous traçons la différence entre les deux modèles pour les 4 scénarii.

Les courbes de la figure 3.13 montrent clairement que plus le trafic est chargé moins la différence entre les deux modèles est importante. Ceci est dû principalement à l'hypothèse sur les temps d'attente des bus au niveau des lignes de feux et qui concerne le premier modèle. Quand le trafic est nul ou très léger, les bus se retrouvent très souvent seuls, sans véhicules intercalés, devant les lignes de feu ce qui nécessite l'utilisation de l'hypothèse pour le modèle 1. Quand le trafic est chargé, le bus se retrouvent plutôt dans des files d'attente

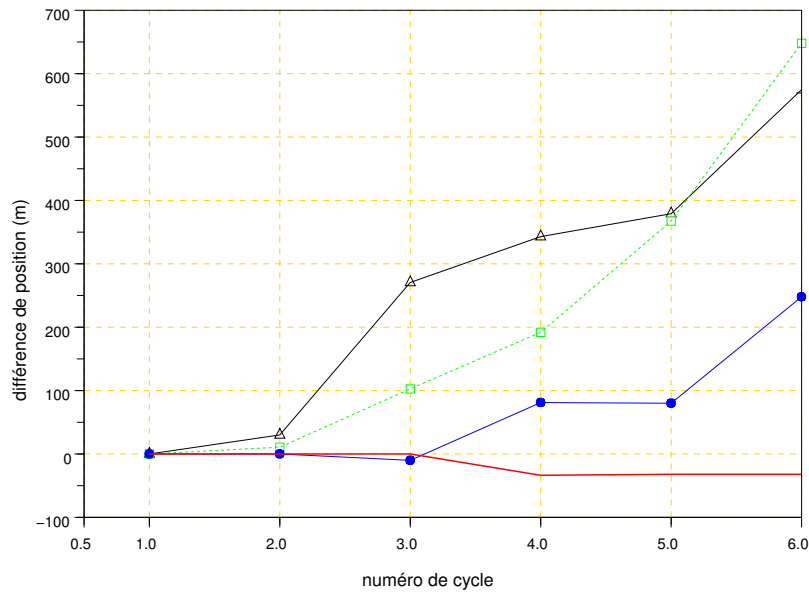


FIGURE 3.13 – Différence entre les deux modèles

ce qui limite l'utilisation de l'hypothèse.

3.4.3 Influence des paramètres des plans de feu

Pour le modèle 1, l'évolution de la position des bus ne dépend que des durées des phases de verts pendant lesquelles les bus traversent les carrefours. Le modèle 2 intègre en plus les instants de début de ces phases. L'objectif de cette section est d'étudier la sensibilité des deux modèles par rapport aux durées des phases de vert et aussi du deuxième par rapport à l'instant de début de phase pour le carrefour 3.

Par rapport aux durées des phases de vert, trois scénarii sont envisagés :

- Plans de feux standards : durées de 40 secondes pour toutes les phases.
- Plans de feux favorables aux bus : durées de 60 secondes pour toutes les phases concernées par les bus.
- Plans de feux défavorables aux bus : durées de 20 secondes pour toutes les phases concernées par les bus.

Les trois scénarii sont comparés avec le cas d'un trafic nul et carrefours en accès libre (cas le plus favorable). La figure 3.14 rapporte l'évolution de la position d'un bus pour les 4 scénarii.

Les deux modèles reproduisent bien la dynamique réelle de la progression des véhicules de transport en commun en variation en fonction des durées des phases. Afin d'observer maintenant la différence entre les deux modèles, nous traçons sur la figure 3.15, la différence de positions entre les deux modèles pour les quatre cas étudiés.

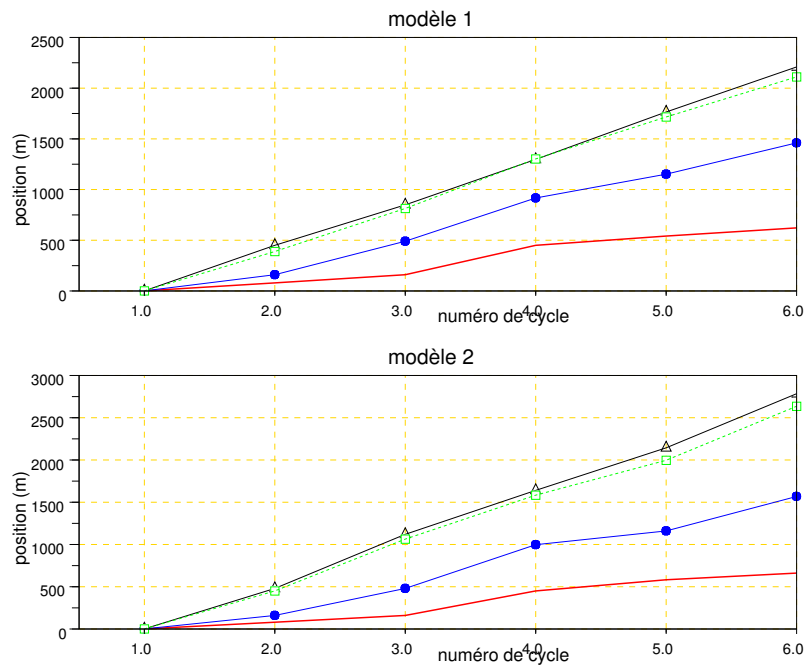


FIGURE 3.14 – Influence des durées des phases de vert

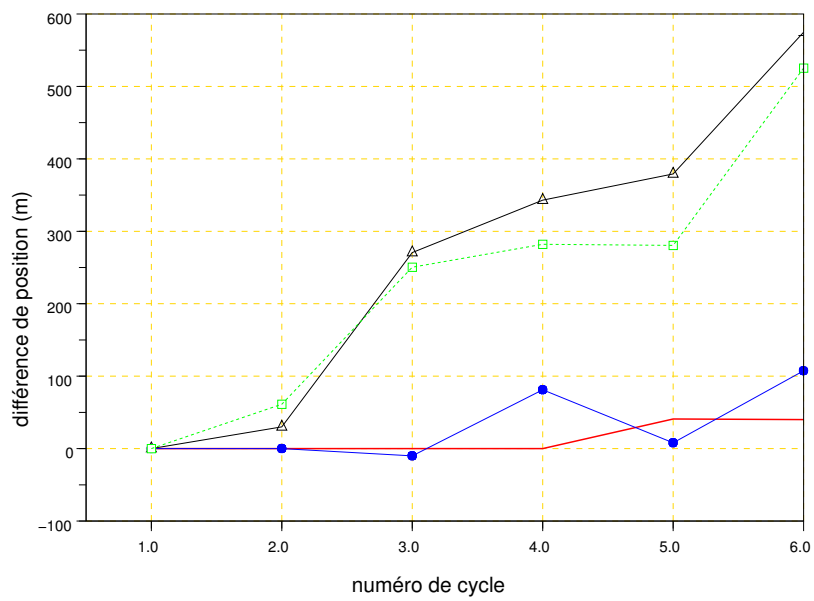


FIGURE 3.15 – Différence entre les deux modèles

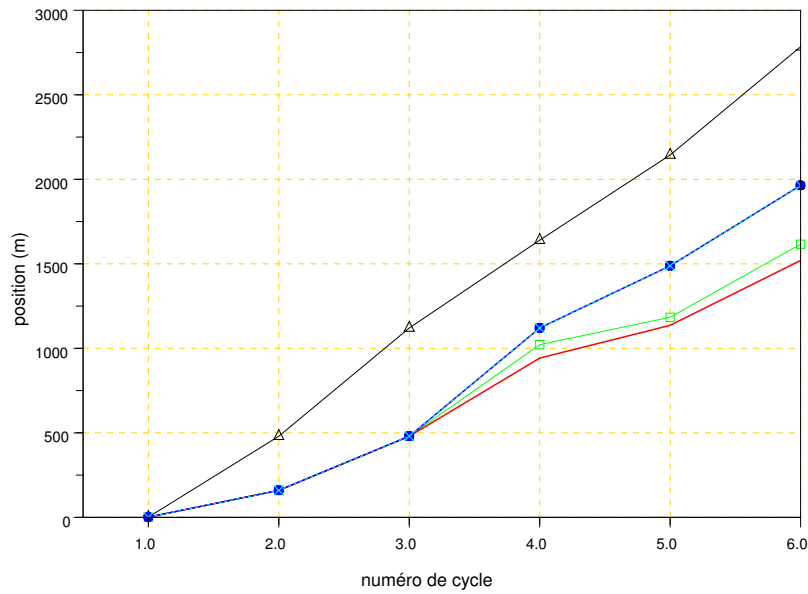


FIGURE 3.16 – Influence des instants de début des phases de vert

L'écart entre les deux modèles est d'autant plus important que les durées des phases de vert allouées aux bus sont importantes. Quand le feu vert alloué aux bus est important, les files d'attente au niveau des lignes de feux sont rares ce qui augmente le nombre de fois où on fait appel à l'hypothèse pour le modèle 1. Dans le cas contraire, les files d'attente sont beaucoup plus nombreuses et par conséquent l'appel à l'hypothèse est moins fréquent.

Pour vérifier l'influence des instants de début des phases sur la progression des bus, nous avons fait varier l'instant de début de phase du carrefour 3. Sur la figure 3.16 sont tracées les courbes d'évolution de la position du bus pour plusieurs valeurs de l'instant de début de phase.

Les valeurs 0 et 70 comme instants de début de phase donnent des courbes confondues. Ces valeurs permettent aux bus de passer le minimum de temps possible sur la ligne de feu du carrefour 3. La différence entre les 4 courbes est localisée au cycle 4 : le bus traverse le carrefour 3 entre les cycles 3 et 4. L'écart au cycle 4 peut s'accroître dans les cycles suivants comme le montre les courbes pour les valeurs 0 et 70.

3.5 Conclusion

La complexité de la modélisation de la progression des véhicules de transport en commun dans un milieu urbain réside dans la reproduction des phénomènes d'arrêt et de démarrage au niveau des carrefours et des stations et aussi au niveau des files d'attente des véhicules particuliers. Ceci a pour conséquence l'absence de modèles permettant de reproduire finement et simplement la circulation des véhicules de transport en commun. Nous avons présenté dans ce chapitre deux modèles qui décrivent l'évolution des positions des véhicules. Le premier ne permet de considérer que les durées de phases de vert comme variables des plans de

feu. Les décalages entre les carrefours successifs sont considérés inconnus et représentés comme des variables aléatoires. Le deuxième modèle est plus réaliste et permet de tenir compte des décalages comme variables supplémentaires. Les deux modèles sont fondés sur des événements qui sont de deux types : passage d'un feu et passage d'une station. Ils permettent de tenir compte dans la progression des véhicules de transport en commun du trafic des véhicules particuliers.

L'étude en simulation a montré la pertinence et la cohérence des deux modèles. Elle a permis aussi de vérifier que le premier modèle suit le deuxième quand les conditions du trafic sont chargées. Quand le trafic est sous-saturé, le premier modèle s'écarte du deuxième à cause des hypothèses considérées pour le premier.

Les deux modèles et surtout le deuxième permettent de représenter finement et simplement la circulation des véhicules de transport en commun. La simplicité des modèles est de la plus grande importance pour la suite. Ils seront utilisés dans l'étape de prédiction de la stratégie de régulation en temps réel du trafic.

Chapitre 4

La régulation du trafic multimodal

4.1 Introduction

Dans ce chapitre, nous proposons une architecture pour la régulation du trafic. La régulation des véhicules particuliers est fondée sur la minimisation du nombre total des véhicules sur tout le réseau. Pour les véhicules de transport en commun, l'objectif de régulation est de suivre une trajectoire de référence fixée à l'avance. La régulation des deux modes est formulée comme un problème d'optimisation multi-objectif.

Nous commençons, dans un premier temps, par présenter la formulation générale standard d'un problème d'optimisation mono et multi-objectif. Nous nous intéressons ensuite à l'algorithme d'optimisation par essais particuliers, une méta-heuristique fondée sur la coopération entre individus d'un essaim. Nous présentons l'algorithme original ainsi que l'algorithme modifié et l'adaptation des deux au cas de l'optimisation multi-objectif. Dans un second temps, nous présentons l'architecture principale de la stratégie de régulation du trafic fondée sur la commande prédictive. Puis nous continuons par la stratégie secondaire qui utilise la commande linéaire quadratique pour initialiser les variables de la commande prédictive.

4.2 Formulation générale d'un problème d'optimisation

Un problème d'optimisation est défini comme la recherche du minimum ou du maximum (optimum) d'une ou plusieurs fonctions. Quand il s'agit d'une seule fonction, nous parlons d'optimisation mono-objectif. Dans le cas contraire, nous parlons d'optimisation multi-objectif.

4.2.1 Problème standard de l'optimisation mono-objectif

Résoudre un tel problème revient à trouver un optimum pour une seule fonction appelée fonction objectif. Elle est appelée aussi fonction coût et notée f . Nous parlerons d'optimisation de la fonction f seulement dans le sens de la minimisation puisque la maximisation de f revient à minimiser la fonction $-f$. Dans ce qui suit, nous donnons quelques définitions utiles pour la suite.

Définition 1 : Variables de décision

Ce sont les paramètres dont dépend la fonction objectif f .

Ces variables sont regroupées dans le vecteur $\vec{x}$. Par exemple, la fonction $f(x, y, z, a) = 3 \times x^2 - a \times y + a^2 \times e^{-(z^2-1)}$ dépend des quatre paramètres a, x, y et z . Si pour une raison quelconque, a est fixé, les variables de décision sont x, y et z et sont regroupées dans le vecteur $\vec{x} = [x \ y \ z]^T$.

Définition 2 : minimum global

$\vec{x}^*$ est un minimum global de la fonction f si et seulement si pour tout $\vec{x} \neq \vec{x}^*$, $f(\vec{x}^*) \leq f(\vec{x})$.

Définition 3 : minimum local

$\vec{x}^*$ est un minimum local de la fonction f si et seulement si pour tout $\vec{x} \in V(\vec{x}^*)$ tel que $\vec{x} \neq \vec{x}^*$, $f(\vec{x}^*) \leq f(\vec{x})$.
 $V(\vec{x}^*)$ est un voisinage de $\vec{x}^*$.

La figure 4.1 permet d'illustrer la différence entre minimum global et local.

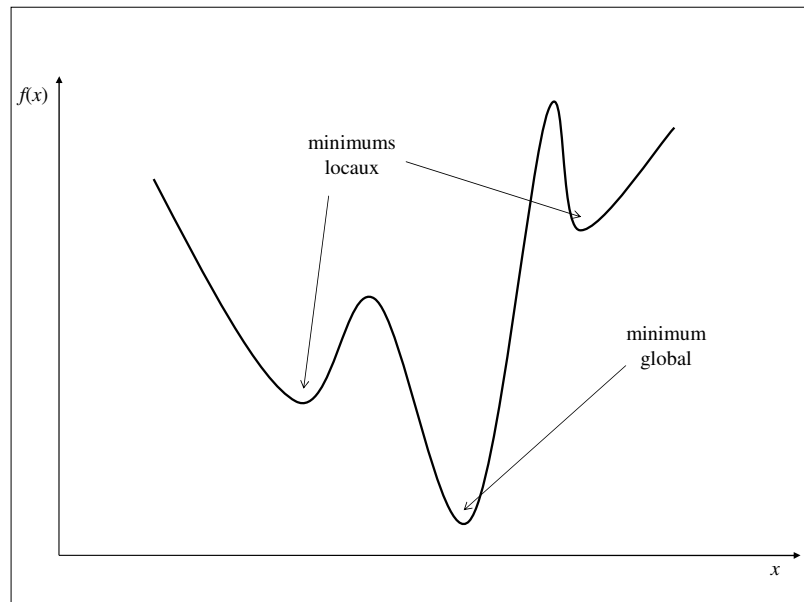


FIGURE 4.1 – Minimum global et local d'une fonction

Souvent, les variables de décision doivent respecter certaines contraintes. Dans ce cas il s'agit d'un problème d'optimisation sous contraintes. Deux types de contraintes existent : contraintes d'égalité et contraintes d'inégalité. Ces contraintes redéfinissent l'espace de recherche des variables de décision en le limitant à l'espace des solutions réalisables.

Mathématiquement, un problème d'optimisation mono-objectif avec contraintes se présente sous la forme standard suivante :

$$\begin{aligned} & \text{Minimiser} && f(\vec{x}) \\ & \text{avec} : && g_i(\vec{x}) = 0; \ i = 1, \dots, m \\ & && h_j(\vec{x}) \leq 0; \ j = 1, \dots, p \\ & \text{et} && \vec{x} \in \mathbf{R}^n \end{aligned}$$

où n est le nombre de variables de décision, m le nombre de contraintes d'égalité et p le nombre de contraintes d'inégalité. f est la fonction objectif et les fonctions g_i et h_j expriment les contraintes d'égalité et d'inégalité.

4.2.2 Problème standard de l'optimisation multi-objectif

Dans plusieurs situations réelles, l'objectif est de minimiser plusieurs critères ou fonctions "coût" : c'est l'optimisation multi-objectif. Par rapport à un problème d'optimisation mono-objectif, le cas multi-objectif ne diffère que par le nombre de fonctions objectif. Mathématiquement, un problème d'optimisation multi-objectif prend la forme suivante :

$$\begin{aligned} & \text{Minimiser} && f_1(\vec{x}), \dots, f_k(\vec{x}) \\ & \text{avec} : && g_i(\vec{x}) = 0; \ i = 1, \dots, m \\ & && h_j(\vec{x}) \leq 0; \ j = 1, \dots, p \\ & \text{et} && \vec{x} \in \mathbf{R}^n \end{aligned}$$

k désigne le nombre d'objectifs à minimiser.

Dans un problème mono-objectif, la solution est unique. Dans le cas multi-objectif, le résultat de la résolution peut être un ensemble de solutions. Nous introduisons quelques définitions utiles pour la résolution d'un problème d'optimisation multi-objectif.

Définition 4 : Relation de dominance

Pour un problème d'optimisation multi-objectif, le vecteur $\vec{x}_1$ domine le vecteur $\vec{x}_2$ si :

- pour tout $i \in \{1, \dots, k\}$, $f_i(\vec{x}_1) \leq f_i(\vec{x}_2)$ et
- il y a au moins un indice i tel que $f_i(\vec{x}_1) < f_i(\vec{x}_2)$

Définition 5 : Front de Pareto

Le front de Pareto est l'ensemble des solutions optimales non dominées du problème.

La figure 4.2 illustre la relation de dominance et le front de Pareto pour un problème d'optimisation à deux objectifs.

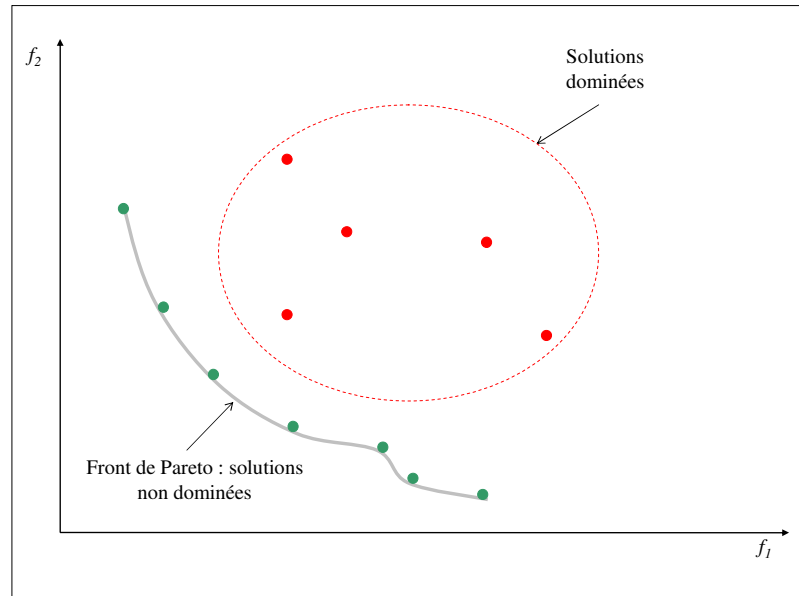


FIGURE 4.2 – Relation de dominance et front de Pareto

4.3 Optimisation par essais particuliers

L'algorithme d'optimisation par essais particuliers (*PSO*) est une méthode méta-heuristique relativement récente puisque les premiers articles la concernant datent de 1995 ([42]). L'algorithme *OEP* fait appel à une population d'agents ou particules. Contrairement à d'autres méthodes de la famille des méta-heuristiques, l'efficacité de l'OEP est due à la collaboration entre agents plutôt que la compétition. Comme toutes les méthodes méta-heuristiques, l'OEP explore l'espace de recherche de manière efficace et intelligente. Pour explorer un espace de recherche, les chances de satisfaire l'objectif de la recherche avec un seul agent peu intelligent sont quasi-inexistantes. Par contre avec plusieurs agents, nous pouvons imaginer plusieurs tactiques pour que ces agents "imbéciles" se comportent collectivement de manière astucieuse.

L'OEP est née presque par hasard. Au départ, les inventeurs souhaitaient simuler des interactions sociales, essentiellement en s'appuyant sur les algorithmes culturels adaptatifs. Ces algorithmes sont souvent décrits comme une sorte d'algorithmes évolutionnaires, avec une population d'individus (agents), dans laquelle, à chaque pas de temps, les "meilleurs" (selon un critère prédéfini) sont plus ou moins imités par les autres. Un aspect essentiel, qui la différencie, par exemple, des algorithmes génétiques classiques, est l'existence d'une mémoire, à laquelle ne contribuent que les meilleurs éléments. Le modèle est censé expliquer et reproduire certains comportements sociaux mais s'est révélé un peu simpliste.

Par contre, convenablement modifié, il peut résoudre des problèmes d'optimisation, car le processus fait apparaître des individus de plus en plus performants : c'est ainsi que l'OEP est née grâce à la collaboration de deux personnes : Russ Eberhart et Jim Kennedy.

Pour illustrer ce que coopération signifie en *PSO*, voici un exemple très simple. D'autres exemples plus

classiques existent comme l'essaim d'abeilles recherchant la nourriture. Comme d'habitude, le gros poisson est au plus profond de l'étang. A chaque pas de temps, chaque pêcheur indique à son collègue l'endroit le plus profond qu'il a déjà trouvé. Au début, pas de différence, chacun suit sa propre route indépendamment. Ensuite, le pêcheur B trouve un meilleur emplacement, alors le pêcheur A tend à aller vers lui, assez vite. Ce faisant, il détériore sa position. Dès lors, il a un compromis à faire entre la meilleure position qu'il a trouvée jusque-là et celle trouvée par B. Il continue d'aller vers B, mais plus lentement. Le point important est que, ce faisant, il échappe au minimum local. Dans cet exemple, nous trouvons les notions essentielles : position, vitesse (ou, plus précisément, un opérateur modificateur de position), échange d'informations, mémoire (très limitée), capacité rudimentaire à combiner les informations pour prendre une décision. L'idée est que tout cela doit rester simple. Dans les sections suivantes, nous présentons l'algorithme original et une version modifiée.

4.3.1 Algorithme original de l'optimisation par essaims particuliers

Nous considérons, dans l'espace de recherche, un essaim de particules. Chaque particule est en train de se déplacer, c'est-à-dire qu'à chaque instant, elle a une position et une vitesse instantanée. Également, chaque particule a une petite mémoire, lui permettant de se souvenir de sa meilleure performance, en position et en valeur. Enfin, chaque particule dispose d'un groupe d'informatrices qui forment son voisinage.

A partir des quelques informations dont elle dispose, une particule doit décider de son prochain mouvement, c'est-à-dire décider de sa nouvelle vitesse ou déplacement. Pour cela, elle combine linéairement trois informations :

- sa tendance actuelle (vitesse actuelle).
- sa meilleure performance.
- la meilleure performance de ses voisines (ses informatrices).

À Chaque terme de la combinaison linéaire est affecté un coefficient parfois appelé coefficient de confiance, qui pondère trois tendances :

- tendance à suivre sa propre voie.
- tendance conservatrice (revenir sur ses pas).
- tendance "panurgienne" (suivre la meilleur voisine).

Mathématiquement, une particule " i " est représentée, à l'instant t , par sa position $X_i(t)$ et sa vitesse $V_i(t)$. L'algorithme original utilise les équations suivantes pour déterminer la position future de la particule i :

$$\begin{cases} V_i(t+1) = \omega \cdot V_i(t) + c_1 \cdot r_{1,i}(t) \cdot [Xbest_i - X_i(t)] + c_2 \cdot r_{2,i}(t) \cdot [Xgbest_i - X_i(t)] \\ X_i(t+1) = X_i(t) + V_i(t+1) \end{cases}$$

$Xbest_i$ et $Xgbest_i$ désignent respectivement la meilleure position de la particule i et la meilleure position de son voisinage. ω , c_1 et c_2 sont les coefficients de confiance fixés avant le démarrage de l'algorithme. La

convergence de l'algorithme dépend essentiellement des valeurs de ses paramètres. $r_{1,i}(t)$ et $r_{2,i}(t)$ sont deux valeurs aléatoires comprises entre 0 et 1 qui changent à chaque itération. L'efficacité de l'algorithme tient en particulier au caractère aléatoire des deux derniers coefficients.

La première étape consiste à initialiser l'essaim de particules dans l'espace de recherche. Nous pouvons le faire soit de manière aléatoire (ce qui n'est pas forcément si simple selon la forme de l'espace de recherche), soit de manière régulière, en particulier sur la frontière. En général, le plus efficace est une combinaison des deux. La deuxième étape consiste à initialiser également les vitesses, de manière aléatoire. On suppose que l'essaim est de taille constante. La taille représente également un paramètre de l'algorithme, à fixer. Ensuite, pour chaque particule, on définit son voisinage. Il y a deux façons de le définir : soit un voisinage géographique, qui doit être recalculé à chaque pas de temps, et qui suppose l'existence d'une distance dans l'espace de recherche, soit un voisinage "social" défini une fois pour toutes. Généralement c'est le voisinage social qui est le plus utilisé et ce pour plusieurs raisons :

- il est plus simple à programmer,
- il est moins coûteux en temps calcul,
- en cas de convergence, un voisinage social tend à devenir un voisinage géographique.

Pour le voisinage social, il y a aussi plusieurs façons de le définir. La figure 4.3 illustre les architectures de voisinage les plus utilisées.

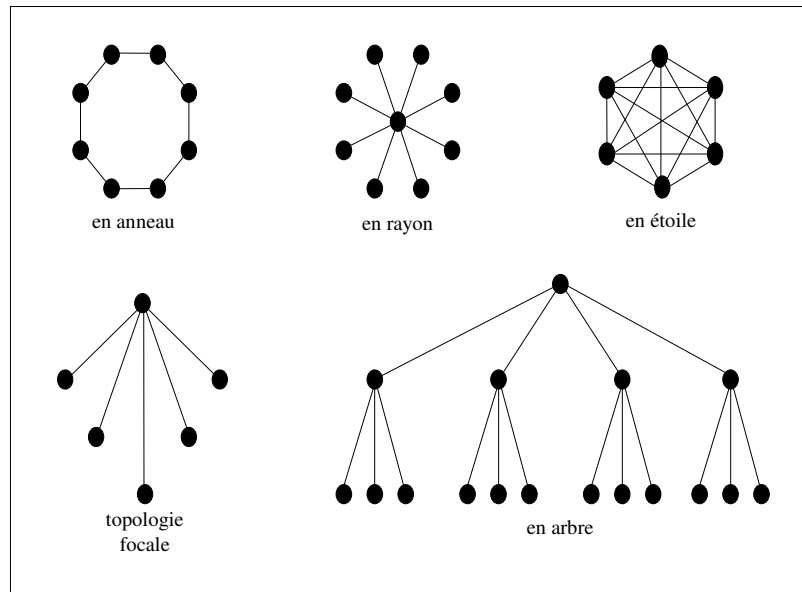


FIGURE 4.3 – Différentes façons pour le voisinage d'une particule

Le nombre de particules de l'essaim est un paramètre très important. Il dépend des dimensions de l'espace

de recherche. Il est bien évident qu'un nombre élevé de particules a l'avantage d'offrir plus de chance pour converger vers un optimum global. Cependant il a aussi l'inconvénient de nécessiter beaucoup de calcul et par conséquent beaucoup de temps pour converger. Les études empiriques ont montré que pour des problèmes à 30 variables de décision avec un espace de recherche hypercubique de l'ordre $[-30; 30]^{30}$, le nombre de particules nécessaires pour une convergence rapide est de l'ordre de 15 à 30 particules.

Le choix des coefficients de confiance ω , c_1 et c_2 a une importance de premier plan pour la convergence de l'algorithme. Bien que l'algorithme est de nature stochastique, certains travaux se sont intéressés à l'analyse mathématique rigoureuse des conditions de convergence de l'algorithme. M. Clerc et J. Kennedy en 2002 ([19]) et F. Van Den Bergh ([10]), la même année, ont établi la condition générale pour la convergence de l'algorithme (voir annexe pour la démonstration) :

$$\omega > \frac{1}{2} \times (c_1 + c_2) - 1$$

La condition générale de convergence est obtenue sous des hypothèses restrictives comme remplacer les variables aléatoires $r_{1,i}(t)$ et $r_{2,i}(t)$ par les valeurs de leurs espérances égales toutes les deux à 0.5. Néanmoins les études empiriques effectuées par les mêmes auteurs confirment statistiquement les résultats de l'étude mathématique. La condition générale de convergence ne permet pas de fixer les valeurs des paramètres et il n'y a pas de règle générale pour les fixer. Un jeu de paramètres empirique a été établi et semble garantir la convergence de l'algorithme dans plusieurs problèmes d'optimisation réels. Dans la suite, nous utilisons cette combinaison pour ne pas transformer le problème d'optimisation initial en un nouveau problème d'optimisation sur les paramètres de l'algorithme *OEP*. Cette combinaison est donnée par les équations suivantes :

$$\begin{cases} \omega = \frac{1}{2 \cdot \ln(2)} \\ c_1 = c_2 = 0.5 + \ln(2) \end{cases}$$

Dans certaines versions, la vitesse des particules est limitée à une valeur V_{max} . L'efficacité de ce nouveau paramètre n'est pas démontrée. Dans la version que nous utilisons, aucune limite de vitesse n'est imposée.

Un des avantages de l'algorithme *OEP* est la prise en compte intuitive des contraintes qui définissent le domaine des solutions réalisables. D'une itération à une autre, une particule, dans son déplacement, peut sortir du domaine des solutions réalisables. Rien ne garantit que, si la position (solution potentielle) $P_i(t)$ à l'itération courante t appartient au domaine des solutions réalisables, la position future à l'instant $t + 1$ appartiendrait elle aussi au domaine des solutions réalisables. Deux solutions existent :

- La première consiste à remplacer la nouvelle position par la position la plus proche de l'espace des solutions réalisables.
- La deuxième solution consiste à considérer la frontière comme un mur qui renvoie la particule qui arrive dessus.

La figure 4.4 permet une illustration graphique des deux solutions pour un problème d'optimisation à deux

dimensions.

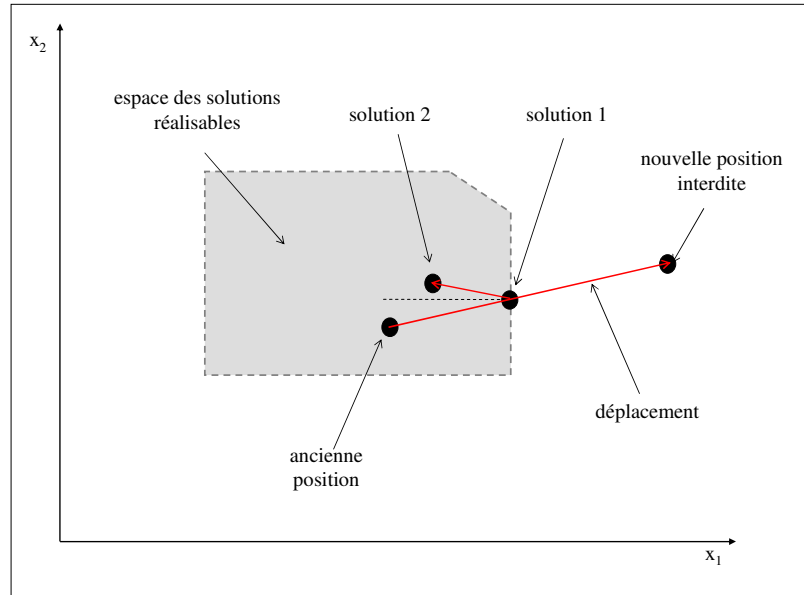


FIGURE 4.4 – Prise en compte des contraintes dans l’algorithme *PSO*

Par la suite, nous considérons la première solution qui a les avantages de nécessiter moins de calcul et de privilégier une exploration plus complète de l’espace de recherche. Les détails sur les calculs ont été donnés dans la section 2.3 du chapitre 2.

Finalement, l’algorithme *PSO* peut s’énoncer comme suit :

Algorithme *PSO* original

```

tant que ( $i < nb\_max$  ou  $min(f) > \epsilon$ ) faire
  pour chaque particule faire
    déterminer meilleure particule du voisinage
    mettre à jour la vitesse et la position
    mettre à jour la meilleure position
  fin pour
  déterminer  $min(f)$ 
   $i++$ 
fin tant que

```

Nous avons réalisé des tests sur l’algorithme pour une série de fonctions de Benchmark. Les caractéristiques des fonctions testées sont sur la figure 4.5.

Nom	Formule	Dimension n	Domaine $[x_{min}, x_{max}]$	Optimum de f
Sphère	$f(x) = \sum_{i=1}^n x_i^2$	30	$[-30, 30]^n$	0
Rosenbrock	$f(x) = \sum_{i=1}^{n-1} (100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2)$	30	$[-30, 30]^n$	0
Rastrigin	$f(x) = \sum_{i=1}^n (x_i^2 - 10\cos(2\pi x_i) + 10)$	30	$[-30, 30]^n$	0
Griewank	$f(x) = \frac{1}{4000} \sum_{i=1}^n x_i^2 - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}}) + 1$	30	$[-30, 30]^n$	-1
Schaffer	$f(x) = 0.5 - \frac{(\sin \sqrt{x_1^2 + x_2^2})^2 - 0.5}{(1 + 0.001(x_1^2 + x_2^2))^2}$	2	$[-30, 30]^2$	0

FIGURE 4.5 – fonctions de test

Sur la figure 4.6 est tracée l'évolution des minimums de ces fonctions sur 80 itérations de l'algorithme.

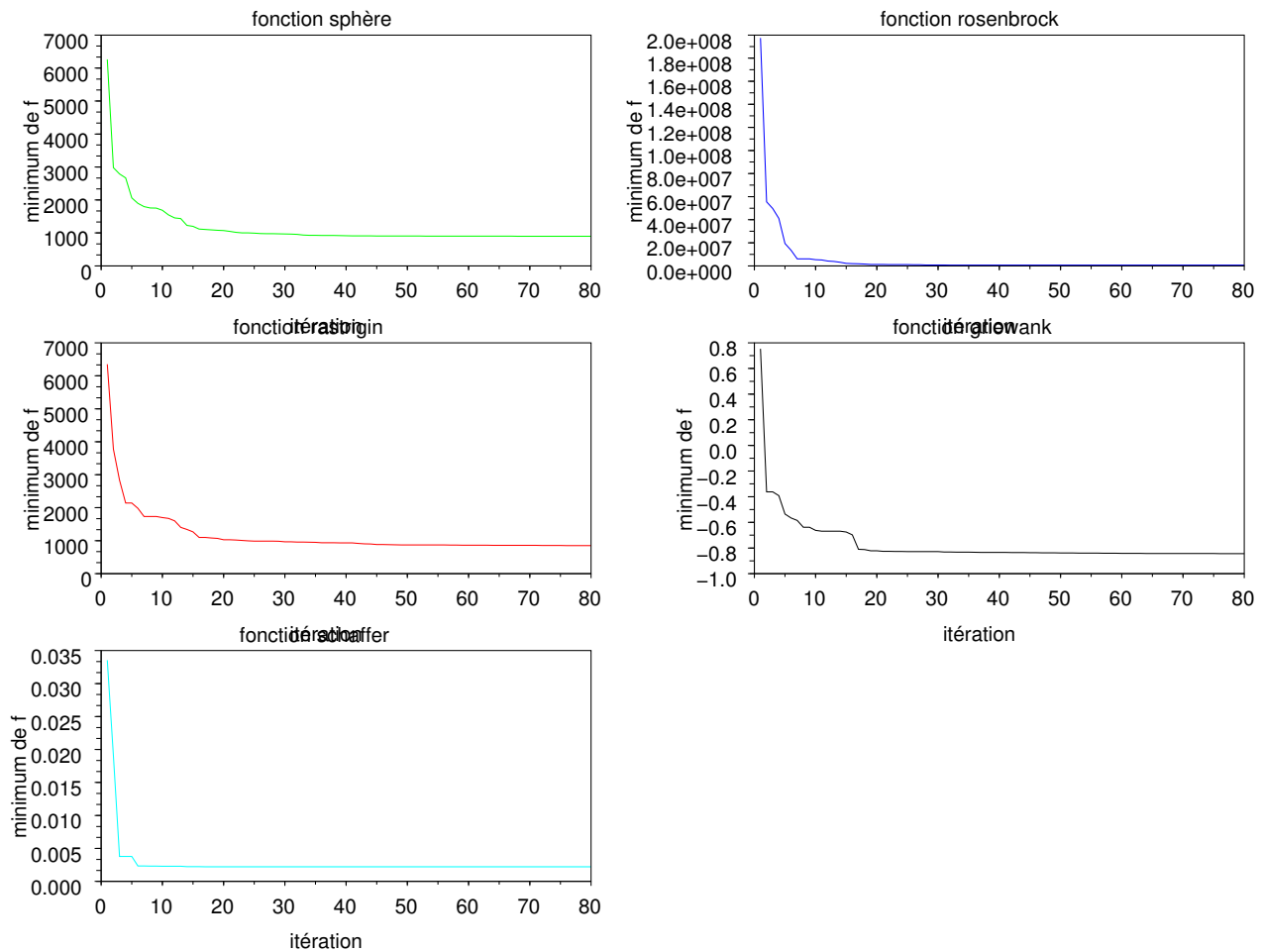


FIGURE 4.6 – Minimum des fonctions de test pour l'algorithme original

Pour les fonctions *Rosenbrock*, *Griewank* et *Schaffer*, l'algorithme converge très bien vers l'optimum au bout de 30 à 40 itérations. Pour les fonctions *Sphère* et *Rastrigin*, l'algorithme converge mais pas vers l'optimum global. La figure 4.7 montre la convergence de la première composante x_1 du vecteur de décision x . Nous observons des amplitudes d'oscillation de moins en moins importantes jusqu'à s'annuler.

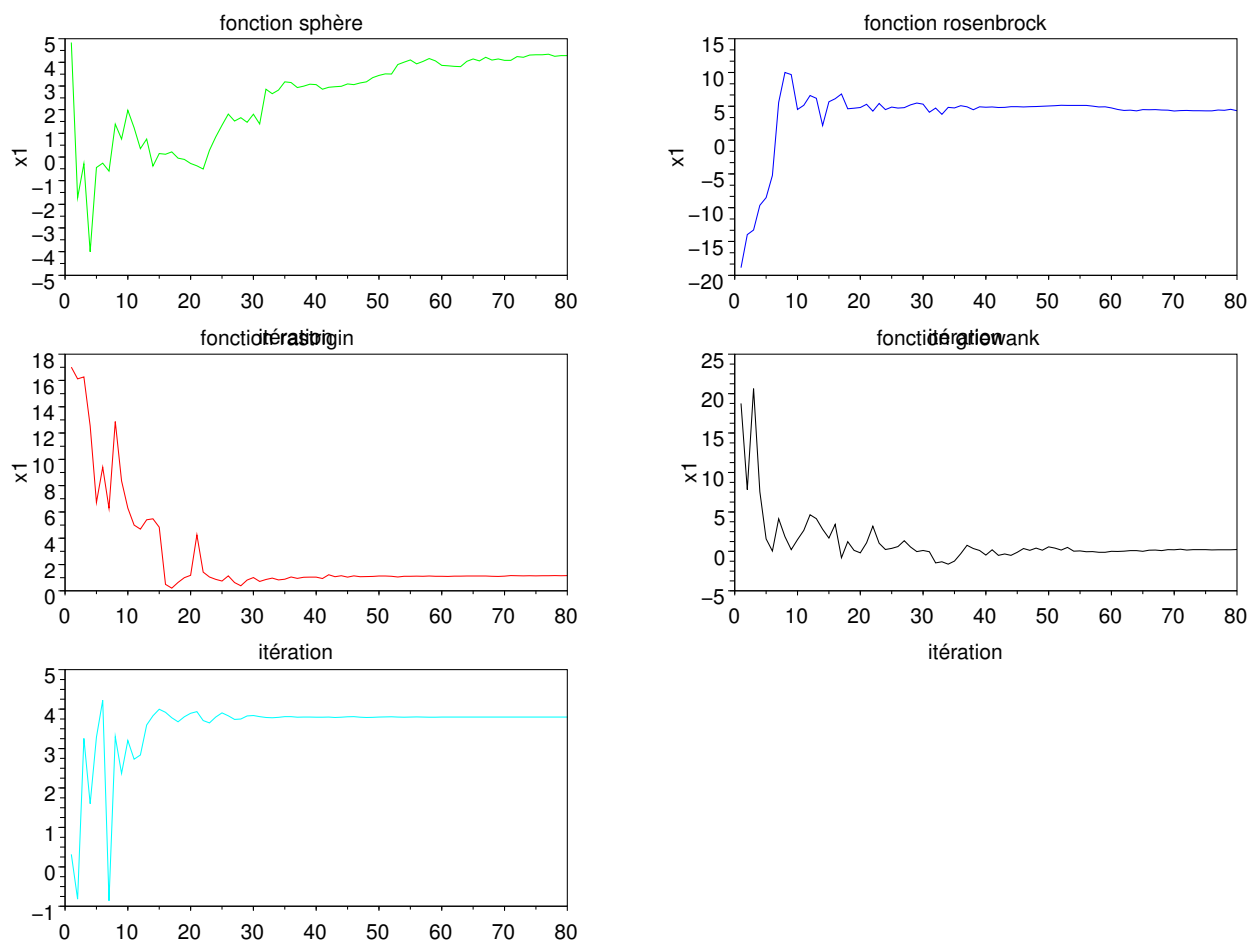


FIGURE 4.7 – Convergence de la 1^{re} composante du vecteur x pour l'algorithme PSO

Les résultats sur ces fonctions montrent que l'algorithme original ne converge pas tout le temps vers l'optimum global. Pour garantir une convergence globale, nous introduisons un algorithme modifié.

4.3.2 Algorithme modifié de l'optimisation par essais particuliers

F. Van Der Bergh a démontré dans ses travaux sur l'optimisation par essais particuliers que l'algorithme original ne permet pas dans tous les cas une convergence vers un optimum global. Il a été même démontré que, parfois, l'algorithme original converge prématurément vers une solution qui n'est même pas un optimum local. Dans ce cas, toutes les particules convergent vers la position de la meilleure particule de l'essaim. Cependant, il est nécessaire de relativiser ses résultats mathématiques puisque les démonstrations sont établies en considérant des hypothèses sur la nature aléatoire sur certains paramètres de l'algorithme. Nous avons vu dans la section précédente que, pour certaines fonctions "coût", l'algorithme original a bien convergé vers des optimaux globaux ou locaux.

Pour corriger la tendance de l'algorithme actuel à converger prématurément, une nouvelle version a été proposée et appelée algorithme modifié. L'idée part de l'observation suivante : Au fur et à mesure que les

itérations avancent, les particules se rapprochent vers la meilleure particule qui elle a tendance de s'arrêter. Ce qui fait que parfois, il y a une convergence prématurée soit vers un optimum local ou tout simplement vers la position de la meilleure particule. L'idée est d'éviter que la meilleure particule ne stagne dans une région. Pour cela, si τ désigne l'indice de la meilleure particule de l'essaim, la mise à jour de la vitesse de cette particule est donnée par l'équation suivante :

$$V_{\tau}(t+1) = -X_{\tau}(t) + X_{best_{\tau}} + \omega \times V_{\tau}(t) + \rho(t) \times (1 - 2 \times r_2(t))$$

Se basant sur cette nouvelle équation des vitesses, la position de la particule est donnée par :

$$X_{\tau}(t+1) = X_{best_{\tau}}(t) + \omega \times V_{\tau}(t) + \rho(t) \times (1 - 2 \times r_2(t))$$

où $r_2(t)$ est une variable aléatoire uniformément distribuée sur l'intervalle $[0;1]$ et $2 \times \rho(t)$ est la longueur du côté de l'hypercube à explorer.

Dans la version originale de l'algorithme, les vitesses tendent à s'annuler. Pour éviter cette tendance, on oblige la meilleure particule à explorer encore un hypercube de côté $2 \times \rho(t)$ autour de sa position actuelle. La valeur de $\rho(t)$ peut être fixée dès le début de l'algorithme. Elle peut aussi varier dynamiquement suivant que l'optimum est de meilleure qualité d'une itération à une autre ou pas. Dans ce dernier cas, la valeur de $\rho(t)$ augmente si pour un certain nombre d'itérations successives, l'optimum est amélioré. Si au contraire, la qualité de l'optimum n'est pas améliorée, la valeur de $\rho(t)$ décroît.

Nous avons testé ce nouvel algorithme sur les fonctions de la figure 4.5. La figure 4.8 montre le minimum de la fonction "coût" sur 200 itérations pour deux versions de l'algorithme modifié :

- Version 1 : en trait bleu continu et correspond à un rayon fixe égale à 1.
- Version 2 : en trait rouge discontinu et correspond à un rayon variable.

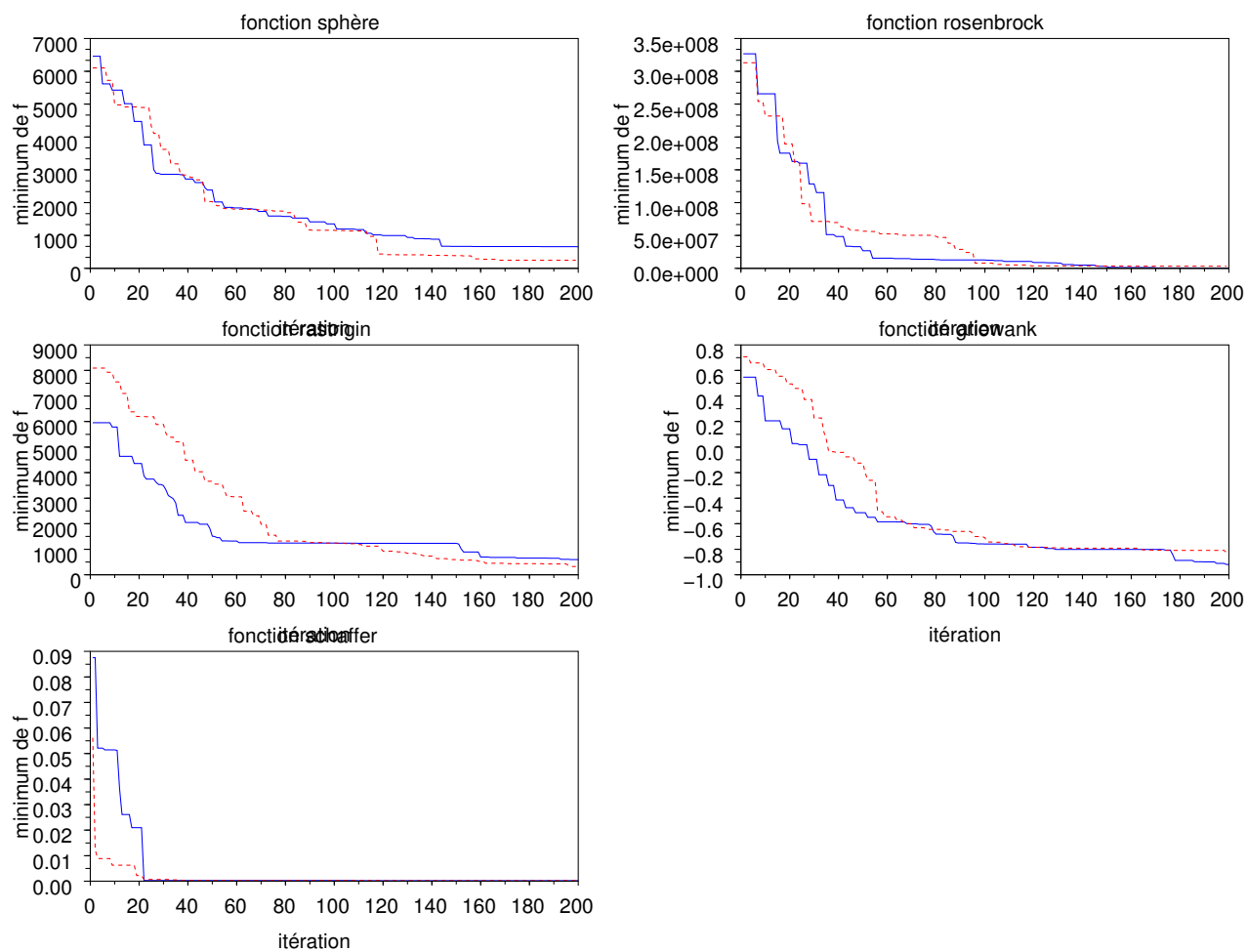


FIGURE 4.8 – Minimum des fonctions de test pour l'algorithme modifié

Sur la figure 4.9 sont tracées le déplacement de la première composante du vecteur x .

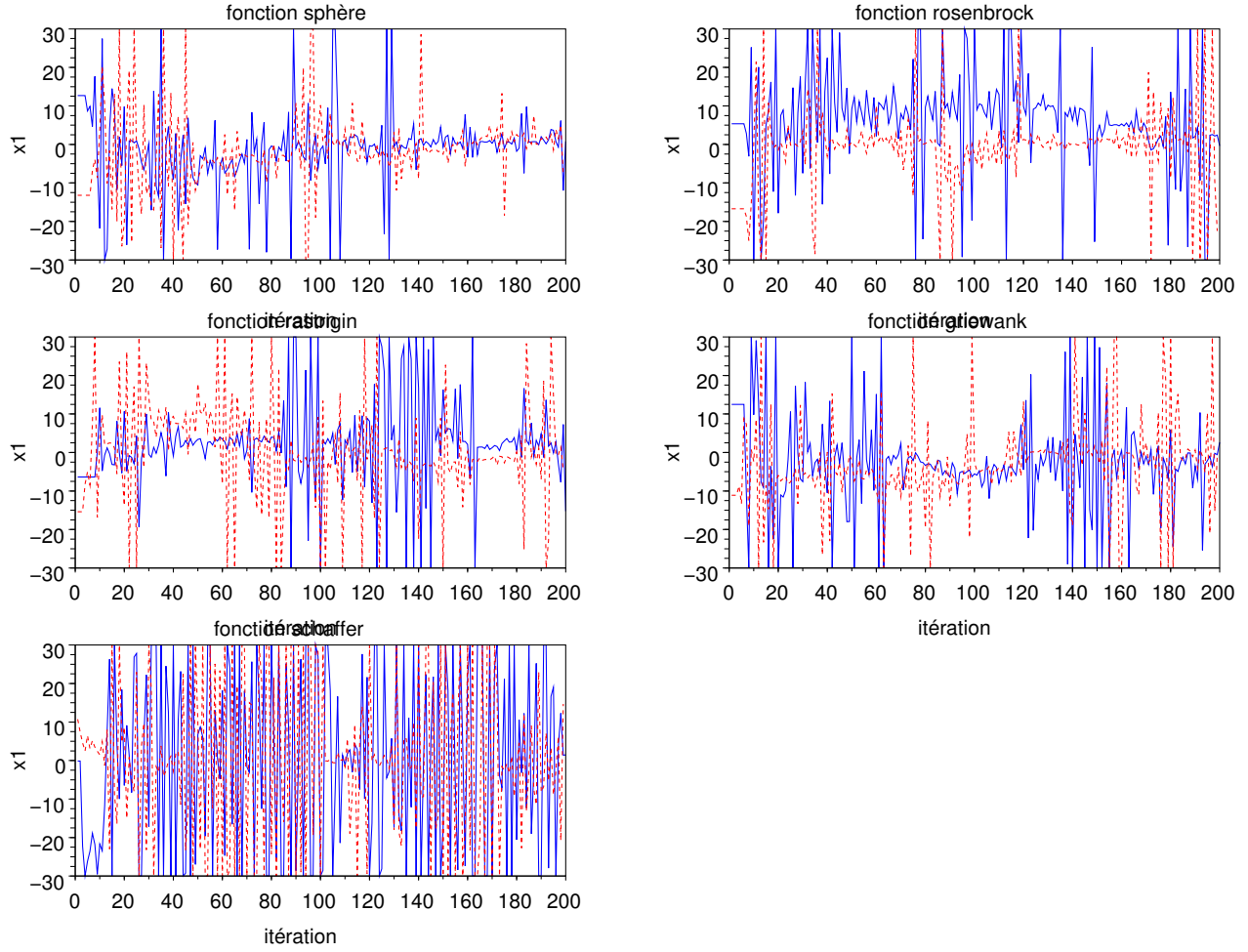


FIGURE 4.9 – Convergence de la 1^{re} composante du vecteur x pour l'algorithme GCPSO

Les deux versions modifiées de l'algorithme sont plus performantes que la version originale. La version modifiée avec variation dynamique du rayon est plus performante que celle où le rayon est fixe sauf pour la fonction *Griewank*. Enfin, les figure 4.7 et 4.9 montrent une différence fondamentale entre le comportement des deux versions modifiées de l'algorithme et de la version originale. Dans la version originale, la particule finit par stagner au bout de 30-40 itérations. Dans les deux versions modifiées, la particule continue d'explorer tout l'espace et ne stagne jamais dans une région même au bout de 200 itérations.

4.3.3 Les essais particuliers pour l'optimisation multi-objectif

Pour un problème d'optimisation multi-objectif standard, la résolution a trois objectifs :

- Augmenter le nombre de solutions dans le front de Pareto.
- Réduire l'écart entre le front de Pareto trouvé et le front de Pareto réel.
- Assurer une répartition uniforme des solutions sur le front de Pareto.

Dans le cas particulier de l'algorithme *PSO*, plusieurs autres points doivent être posés :

- la manière de mise à jour de la meilleure position d'une particule si la nouvelle et l'ancienne position sont non dominées.
- la sélection de la meilleure particule voisine s'il y a plusieurs particules non dominées.

La réponse à tous ces points peuvent être résumer en ces trois mécanismes :

- Sélection d'une solution non dominée parmi plusieurs.
- Conservation et répartition uniforme des solutions non dominées.
- Amélioration de la diversité des solutions dans la création de nouvelles solutions.

Dans la suite, nous présentons, pour chaque mécanisme, différentes manières pour le réaliser rencontrées dans la littérature.

4.3.3.1 Sélection d'une solution non dominée parmi plusieurs

La question de la sélection d'une solution parmi d'autres est d'une première importance pour l'algorithme *PSO* multi-objectif. L'idée la plus élémentaire consiste à choisir aléatoirement la solution. Ceci, du fait de son caractère aléatoire, ne permet pas de contrôler la convergence et la diversité des solutions dans le front de Pareto. La plupart des algorithmes *PSO* multi-objectif font appel à un autre critère pour évaluer la qualité des solutions.

Le critère le plus évident est une fonction d'agrégation sous la forme d'une combinaison linéaire de toutes les fonctions objectifs. Ce critère est bien adapté quand le nombre de fonctions objectifs n'est pas élevé. C'est ce critère que nous allons implanter pour notre cas. Pour des problèmes à plusieurs objectifs, le critère de qualité le plus utilisé est une mesure de la densité. Ce critère permet aussi d'améliorer la diversité des solutions. Nous présentons deux manières de mesurer la densité :

- Estimateur du plus proche voisin : Cet estimateur fournit le périmètre de l'hypercube défini par les solutions les plus proches d'une solution donnée (figure 4.10a). La particule qui a le plus large hypercube est sélectionnée.
- Estimateur du noyau : Pour toutes les solutions non dominées, on définit un rayon σ . Pour une solution donnée, l'estimateur mesure le nombre de solutions comprises dans l'hypersphère de rayon σ centré sur la solution (figure 4.10b). La solution qui a le voisinage le moins dense est sélectionnée.

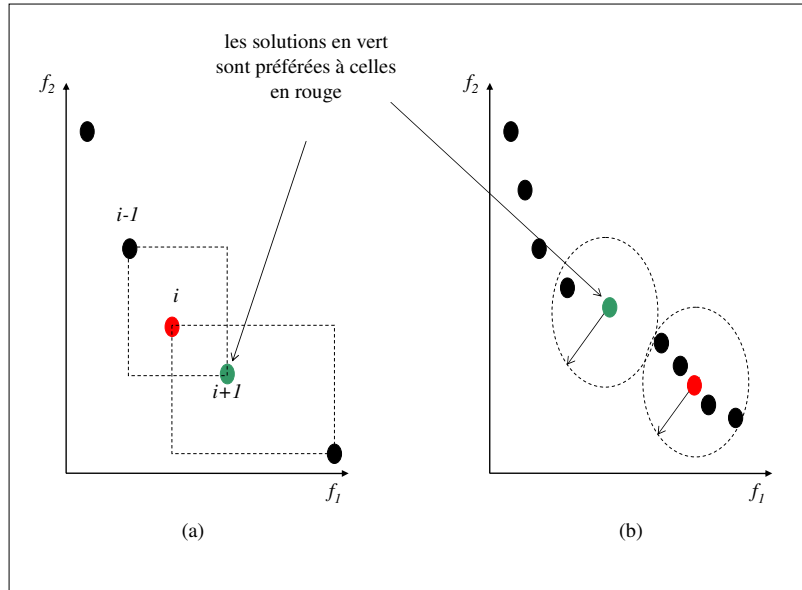


FIGURE 4.10 – Estimateur de la densité des solutions

Ces estimateurs ont l'inconvénient de nécessiter beaucoup de calcul ce qui ralentit la convergence de l'algorithme. Dans notre cas, nous avons choisi comme critère une fonction d'agrégation des deux fonctions objectifs. Nous avons testé aussi, le choix aléatoire et la sélection suivant une seule fonction objectif selon que l'objectif est de privilégier telle ou telle fonction de coût.

4.3.3.2 Conservation et répartition uniforme des solutions non dominées

Dans l'évolution de l'essaim dans l'espace de recherche, plusieurs solutions potentielles non dominées peuvent être découvertes. Conserver toutes ces solutions a un intérêt pratique évident mais aussi théorique. La solution naturelle est l'utilisation d'archives externes. Chaque fois qu'une solution potentielle est découverte, on vérifie si elle domine toutes les solutions conservées dans l'archive auquel cas ces solutions sont effacées et la nouvelle solution dominante demeure seule. Si la nouvelle solution est dominée par au moins une des solutions de l'archive alors l'archive reste comme il est. Dans le cas où la solution est non dominée et non dominante par et pour les solutions de l'archive, on rajoute la nouvelle solution à l'archive. Cependant il est facile de constater qu'avec seulement ces principes simples, il est fréquent que la capacité de l'archive explose. Ce qui a pour conséquence directe l'explosion des temps de calcul. Dans la pratique, la taille des archives est limitée et dépend de la nature du problème et des contraintes calculatoires imposées. Cette nouvelle règle pose un nouveau problème. En effet, si l'archive arrive à saturation, comment décider quelles solutions non dominées doivent sortir et lesquelles doivent entrer? Nous avons répondu, en partie, à cette question dans la section précédente. Nous pouvons utiliser un nouveau critère agrégé des fonctions objectif initiales et choisir les solutions qui minimisent ce nouveau critère. D'autres méthodes se focalisent sur la préservation de la richesse et de la répartition régulière et uniforme des solutions dans le front de Pareto.

Ces méthodes bien qu'elles offrent une meilleure richesse du front de Pareto, ont l'inconvénient de nécessiter des estimateurs de densité, comme nous l'avons vu dans la section précédente, ce qui a pour conséquence des calculs supplémentaires. D'autres techniques sont utilisées comme le principe de filtrage des solutions de l'archive par le concept de ϵ -dominance. Nous avons choisi d'utiliser un critère agrégé ce qui limite les calculs et permet d'aller rapidement vers les solutions qui offrent un équilibre entre les deux fonctions objectif.

En toute rigueur, l'algorithme *PSO* pour le cas multi-objectif nécessite trois types d'archives externes. Un premier pour chaque particule pour stocker les solutions non dominées parcourues par la particule elle-même. Un deuxième pour stocker les solutions des particules voisines. Et le dernier pour stocker les meilleures solutions non dominées de tout l'essaim. Dans la pratique rares sont les travaux qui utilisent les trois archives puisque cela résulte en une explosion des temps de calcul. Généralement un seul archive est utilisé celui pour tout l'essaim. C'est ce dernier que nous utilisons et fixons sa taille à 100 solutions. Quand cette capacité est dépassée, nous utilisons un nouveau critère agrégé pour filtrer les solutions non dominées dans l'archive. Le nouveau critère est le même que dans la section précédente et consiste à une combinaison linéaire non pondérée des deux critères (les critères sont déjà équilibrés par un coefficient de pondération).

4.3.3.3 Amélioration de la diversité des solutions dans la création de nouvelles solutions

La diversification des solutions a deux avantages : éviter une convergence prématurée de l'algorithme et disposer à la fin de plusieurs solutions non dominées offrant un choix large d'une solution unique. Diversifier les solutions peut s'effectuer par deux mécanismes : à travers les mises à jour des positions des particules ou à l'aide de divers manières de mutation et de turbulences.

Pour le premier mécanisme, nous pouvons agir sur la diversification des solutions par le biais du voisinage et de la manière de sélection de la meilleure particule voisine. En privilégiant la sélection selon un estimateur de densité ou en considérant des voisinages locaux, nous favorisons la diversité. Aussi, dans l'équation de mise à jour des positions, le coefficient d'inertie ω a une importance capitale dans la convergence de l'algorithme. Un coefficient trop important favorise l'exploration globale de l'espace de recherche et par conséquent une meilleure diversité de l'algorithme. Par contre, cela a l'inconvénient d'aboutir à une convergence lente. Dans notre cas, nous utilisons un coefficient d'inertie qui concilie la rapidité de convergence et la "bonne" exploration de l'espace.

Le deuxième mécanisme consiste à introduire une mutation ou turbulences dans les trajectoires des particules. Dans l'algorithme original, les particules se déplacent selon une logique bien déterminée. Elles finissent ensuite par converger ou stagner autour d'un optimum local ou global. Une solution qui permet d'éviter la stagnation serait d'introduire une turbulence dans la trajectoire d'une ou plusieurs particules de l'essaim. Lors de la stagnation, les vitesses des particules tendent à s'annuler. Dans notre travail, nous utilisons un algorithme modifié qui permet d'introduire une turbulence maîtrisée dans la vitesse de la meilleure particule de l'essaim. Cette solution permet d'éviter une stagnation de l'algorithme et ainsi éviter une convergence prématurée sans pour autant rajouter de la complexité dans l'algorithme.

Dans la suite, nous décrivons la mise en oeuvre de l'algorithme d'optimisation dans la stratégie de régulation du trafic.

4.4 Stratégie primaire

La stratégie a deux objectifs : d'une part réguler la circulation des véhicules de transport en commun et d'autre part réduire les congestions de la circulation générale et rendre le trafic plus fluide. La progression des véhicules de transport en commun et les véhicules particuliers dépend fortement des états des feux de signalisation contrôlant les carrefours. A chaque début de cycle, des mesures sur l'état du trafic sont collectées depuis des capteurs installés sur l'infrastructure routière et des systèmes de localisation de type GPS/Galilée pour les véhicules de transport en commun. A partir de ces données, la stratégie doit déterminer les états des feux qui permettent de respecter les objectifs de régulation des deux modes de transport. Les objectifs de régulation sont traduits en termes de deux critères mathématiques qui dépendent des états des feux de signalisation de tous les carrefours concernées par la régulation. Respecter les objectifs de régulation revient à minimiser les deux critères par rapport aux états des feux. Ce qui résulte en un problème d'optimisation multi-objectif.

4.4.1 La régulation un problème d'optimisation

Toutes les stratégies de régulation de la circulation sont développées dans l'objectif de minimiser les temps de parcours des véhicules dans le réseau. Certaines approches peuvent estimer ces variables et permettent par la suite de les minimiser en déterminant les plans de feux optimaux à appliquer au trafic. Cependant estimer et minimiser les temps de parcours pour tous les couples "origine-destination" est une opération lourde et difficile. Une manière simple d'atteindre cet objectif consiste à minimiser les files d'attente devant les feux de signalisation ou plus généralement le nombre de véhicules dans les tronçons de route. Cette approche est beaucoup plus facile à mettre en oeuvre puisque nous disposons de plusieurs modèles de représentation des files d'attente et le nombre de véhicules dans les arcs.

De notre point de vue, réguler le trafic dans un réseau routier urbain contrôlé par des feux de signalisation revient à minimiser le nombre de véhicules sur tous les arcs du réseau. Mathématiquement, étant donné un réseau urbain constitué de N_a arcs, et si nous désignons par $X_i(k)$ et $G_i(k)$ respectivement le nombre de véhicules présents dans l'arc " i " et la durée de la phase du vert dans au cycle " k ", réguler le trafic général pour le cycle k revient à résoudre le problème suivant :

$$\begin{aligned} & \textit{Minimiser} \\ & C_VP(G_1(k), \dots, G_{N_a}(k)) = \sum_{i=1}^{N_a} X_i^2(k) \end{aligned}$$

Le second objectif de cette stratégie est de réguler la circulation des véhicules de transport en commun qui se décline en trois possibilités :

- Minimiser les temps de parcours des véhicules.
- Respecter une table horaire pré-établie pour les passages aux stations.
- Respecter des intervalles réguliers entre les véhicules successifs de la même ligne.

Le premier objectif est utilisé dans le cas des véhicules express qui n'ont pas de stations intermédiaires entre l'origine et la destination et aussi pour les véhicules d'urgence. Le second objectif est considéré pour les véhicules de transport en commun urbains dont la fréquence de passage est faible. Le troisième et le dernier objectif est le plus utilisé pour les lignes à haute fréquence de passage. Ces trois objectifs peuvent se ramener à un seul objectif qui consiste à respecter, pour chaque début de cycle, des consignes spatiales pour chaque véhicule. Les consignes temporelles dans un espace spatial sont transformées en des consignes spatiales dans un espace temporel.

Étant donné un véhicule de transport en commun " j " situé à la position $P_j(k)$ au début du cycle " k ", l'objectif de régulation pour ce véhicule peut s'écrire comme suit :

$$\text{Minimiser } (P_j(k) - P_j^s(k))^2$$

où

- $P_j^s(k)$ représente la position consigne à respecter pour le véhicule " j " au début du cycle " k ".
- $Td_{j_p}(k)$ représente l'instant de début de la phase du vert du feu " p " pour le bus " j " au cycle " k ".

Si le réseau est traversé par plusieurs lignes numérotées de 1 jusqu'à n_l et qu'au début du cycle " k ", n_m véhicules de la ligne m circulent sur le réseau, le critère à minimiser devient :

$$C_TC(Td_1(k), \dots, Td_{N_{cb}}(k), G_1(k), \dots, G_{N_a}(k)) = \sum_{m=1}^{n_l} \sum_{n=1}^{n_m} (P_{m,n}(k) - P_{m,n}^s(k))^2$$

Ainsi, réguler le trafic général, réduire les congestions et réguler la circulation des véhicules de transport en commun dans un réseau routier urbain contrôlé par des feux de signalisation revient à résoudre un problème d'optimisation bi-objectifs de très grandes dimensions. Ce problème s'écrit sous la forme suivante :

$$\text{Minimiser} \begin{cases} C_VP(G) = \sum_{i=1}^{N_a} X_i(j)^2 \\ C_TC(T_d, G) = \sum_{m=1}^{n_l} \sum_{n=1}^{n_m} [P_{m,n}(j) - P_{m,n}^s(j)]^2 \end{cases}$$

où

- G est un vecteur de dimension N_a qui contient les durées des phases du vert de tous les arcs.
- T_d est un vecteur dont la dimension est le nombre d'arcs traversés par les véhicules de transport en commun. Il contient les débuts des phases de vert pour les arcs concernés.

Il est nécessaire que les critères soient comparables quantitativement. Le premier critère concernant les véhicules particuliers est un critère quadratique du nombre de véhicules. Le second concernant les véhicules de transport en commun est quadratique d'une différence de position exprimée en mètre. Les deux critères sont très hétérogènes. Le critère C_VP dépend du nombre d'arcs dans le réseau, du nombre de voie et de la charge du trafic. Le critère C_TC dépend du nombre de véhicules de transport en commun présent dans le réseau qui varient au cours du temps, de l'état du trafic (à travers les positions des véhicules) et aussi des positions consignes pour chaque véhicule. En conséquence, les valeurs numériques des deux critères varient en fonction de la géométrie du réseau urbain considéré, des flux de véhicules, de la fréquence de passage des véhicules de transport en commun et des consignes de régulation. Ceci rend le problème de calibrage entre les deux critères très dépendant de l'exemple traité. Une solution est de considérer le réseau concerné dans une capacité proche de la capacité maximale en gardant des conditions fluides du trafic. Les deux critères sont ensuite calculés dans ces conditions et le coefficient de pondération est pris égal au rapport entre les deux critères. Si nous considérons le coefficient de pondération K_p pour le critère C_VP alors étant donné les critères dans les conditions d'équilibre maximal C_VP_{eq} et C_TC_{eq} , nous pouvons écrire :

$$K_p = \frac{C_TC_{eq}}{C_VP_{eq}}$$

Dans ces conditions, le problème d'optimisation devient :

$$\begin{aligned} & \textit{Minimiser} \\ & \left\{ \begin{array}{l} C_VP(G) = K_p \times \sum_{i=1}^{N_a} X_i(j)^2 \\ C_TC(T_d, G) = \sum_{m=1}^{n_l} \sum_{n=1}^{n_m} [P_{m,n}(j) - P_{m,n}^s(j)]^2 \end{array} \right. \end{aligned}$$

Pour un carrefour élémentaire simple constitué de quatre branches et partant du principe que les plans de feu ne comportent que deux phases, nous avons démontré au chapitre 1 qu'il suffit de deux variables qui sont l'instant de début de vert d'une phase et sa durée pour caractériser le plan de feu du carrefour. Plus généralement, nous avons démontré comment réduire le nombre de variables pour caractériser les plans de feu d'un carrefour quelconque. Pour des raisons de sécurité et par la définition d'un cycle de feu (voir chapitre 1), nous pouvons écrire les contraintes suivantes :

$$\begin{aligned} G_i^{min} & \leq G \leq G_i^{max} \\ 0 & \leq Td_i \leq C \end{aligned} \tag{4.1}$$

Le problème de régulation du trafic urbain multimodal se résume à un problème d'optimisation multi-objectif avec contraintes. Dans la section suivante, nous nous intéressons à la résolution de ce problème et son intégration dans l'architecture globale de la stratégie.

4.4.2 Architecture

Le problème de la régulation comme il est posé est un problème de commande d'un système dynamique. A partir des mesures sur l'état à un instant présent, l'objectif est de déterminer les commandes pour l'instant futur de telle façon que les états futurs du système suivent une trajectoire consigne pré-définie. Il s'agit d'un problème de commande prédictive généralisée.

4.4.2.1 La commande prédictive

La commande prédictive est une méthode avancée de contrôle automatique des systèmes dynamiques. Le principe fondamental de cette commande est l'utilisation d'un modèle interne qui prédit l'état du système dans le futur. La première formalisation théorique remonte à 1978 avec les travaux de J. Richalet ([59]). Elle est ensuite généralisée par D.W. Clarke en 1987 ([17],[18]).

La commande prédictive est surtout utilisée dans l'industrie pour contrôler et commander des systèmes complexes. Initialement, cette commande a été appliquée aux systèmes dynamiques lents à cause de la lourdeur des calculs qu'elle nécessite. Avec le développement de la puissance calculatoire des machines, la commande prédictive est devenue une puissante méthode de contrôle de tous systèmes lents et rapides. La commande prédictive peut se résumer à "étant donné l'état actuel du système, comment prendre une décision (sur les variables de commande) pour minimiser un critère donné?". Nous retrouvons alors un vocabulaire propre au domaine de l'optimisation sauf qu'ici, il s'agit de systèmes dynamiques. La commande prédictive est généralement constituée de deux modules : un module de prédiction (modèle interne) et un module d'optimisation.

D'un point de vue mathématique, nous considérons un système dynamique à temps discret régi par l'équation d'état généralisée suivante :

$$x(k+1) = f(x(k), u(k))$$

où $x(k)$ et $u(k)$ désignent respectivement l'état du système et la commande à l'instant k et f une fonction quelconque. Nous nous plaçons à l'instant k , le problème général de la commande prédictive est de trouver les valeurs des commandes à appliquer sur un horizon temporel de N_c pas de temps, pour minimiser, sur un horizon temporel de $N_h > N_c$, le critère général suivant :

$$J = \sum_{i=1}^{N_h} h(x(k+i)) + \sum_{i=1}^{N_c} g(u(k+i))$$

La fonction g exprime le critère correspondant aux valeurs de commande du système. Elle est utilisée pour limiter les amplitudes de la commande. La fonction h exprime le critère principal à minimiser. Elle est généralement quadratique et exprime la différence entre l'état réel du système et l'état de référence à atteindre.

A partir de l'état du système au pas de temps k , le module d'optimisation détermine les valeurs de la commande u pour les N_c pas de temps futurs qui permettent de minimiser le critère J . L'état du système le long de l'horizon temporel N_h est estimé par le module de prédiction constitué du modèle du système. Le principe de la commande prédictive consiste à n'appliquer effectivement au système que la commande pour le premier pas de temps. La même procédure est appliquée au début de chaque pas de temps.

Un paramètre de première importance pour la mise en oeuvre d'une commande prédictive est le choix de l'horizon de contrôle N_c . D'un point de vue strictement théorique, l'horizon doit être infini pour les deux raisons suivantes :

- tenir compte des effets de la commande sur l'état du système et le critère à minimiser.
- garantir la stabilité du système dans le cas où les contraintes ne peuvent garantir cette stabilité.

Dans certaines classes de systèmes dynamiques, il est possible de montrer la stabilité du système avec un horizon de contrôle fini. Dans le cadre de ce travail, nous n'avons pas besoin de nous occuper de la stabilité du système. Les contraintes imposées à la commande permettent de garantir, dans un état de trafic d'équilibre, la stabilité du système dans le sens où les variables d'état restent bornées. D'un point de vue pratique, il faut limiter la longueur de l'horizon de contrôle pour deux raisons :

- Plus l'horizon de contrôle est long plus les calculs sont longs et complexes ce qui représente pour un problème de commande, en temps réel, un obstacle difficile à surmonter.
- Dans le cas où le système a des entrées exogènes qui ne sont pas estimables de façon précise, résoudre le problème d'optimisation avec un long horizon de contrôle peut s'avérer inefficace. Les variables de commande déterminées par l'algorithme d'optimisation sont valables pour des estimations futures des variables exogènes qui peuvent s'avérer être très loin des valeurs réelles.

La commande prédictive est généralement constituée de deux modules : un module de prédiction pour estimer les valeurs futures des variables d'état et un module d'optimisation pour déterminer les valeurs des variables de commande à appliquer aux systèmes. Le schéma général de la commande prédictive est donnée sur la figure 4.11.

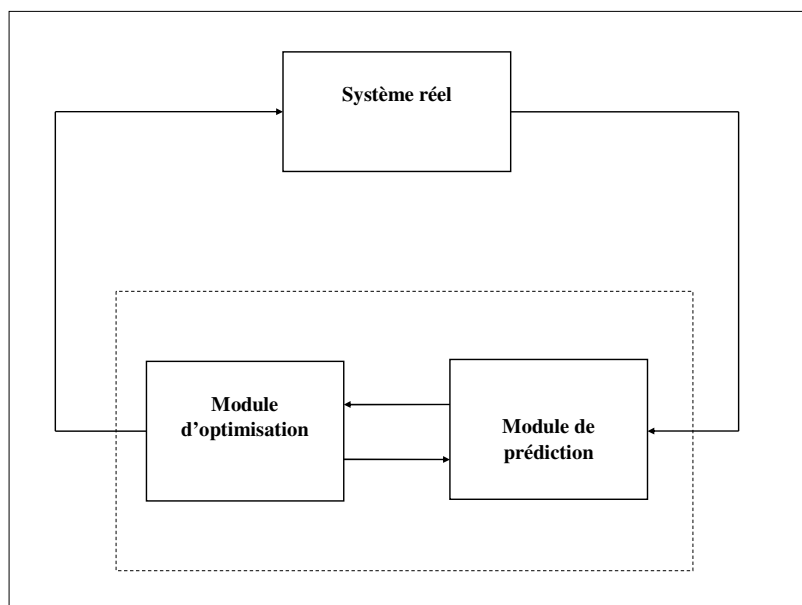


FIGURE 4.11 – Schéma de principe de la commande prédictive

Le module de prédiction est constitué des deux modèles décrits dans les chapitres 2 et 3. Le module d'optimisation sera décrit dans les sections suivantes.

4.4.2.2 Module de transformation des objectifs de régulation

Les objectifs de régulation des exploitants des transports en commun sont de trois types : minimiser les temps de parcours des véhicules quelque soit les conditions du trafic et la demande des utilisateurs, respecter des horaires de passage fixés dans les stations et enfin respecter un intervalle régulier entre les véhicules successifs d'une même ligne. Ceci a pour conséquence l'obligation de modéliser la progression des véhicules de transport en commun par des temps de passage dans les stations ou par le retard d'un véhicule par rapport à son prédécesseur de la même ligne. Le modèle développé décrit dans le chapitre 3 ne permet de modéliser les trajectoires des véhicules de transport en commun qu'en termes de positions relevés en des pas de temps réguliers égaux au cycle d'un feu de durée C . La régulation s'effectue de telle façon que les positions réelles des véhicules suivent des positions-consignes. L'exploitant ne donne que des consignes temporelles en des points précis du réseau, il est obligatoire alors de transformer ces consignes temporelles en des consignes spatiales. Nous distinguons par la suite trois mécanismes de transformation selon les consignes temporelles.

Premier cas : Objectif "minimiser les temps de parcours"

C'est le cas le plus simple à traiter. Minimiser le temps de parcours d'un véhicule de transport en commun revient, rigoureusement, à maximiser la distance parcourue de ce véhicule pendant une période temporelle fixe. Si le temps est discrétisé avec un pas de temps égal à la durée d'un cycle de feu C , minimiser le temps de parcours d'un véhicule revient à ce qu'il effectue une distance maximale pendant la période de durée C .

La distance théorique maximale que peut parcourir un véhicule durant C est

$$d_{max} = C \times V_b$$

où V_b est la vitesse libre du véhicule.

Ainsi, pour un véhicule de transport en commun dont la trajectoire est représentée par sa position P , minimiser son temps de parcours revient à minimiser pour chaque pas de temps k (cycle de durée C) le critère suivant

$$P(k) - (k - 1) \times C \times V_b$$

Deuxième cas : Objectif "respecter un horaire de passage pré-fixé dans les stations"

Cet objectif est surtout utilisé pour les lignes de transport en commun à faible fréquence de passage. La trajectoire d'un véhicule d'une ligne est représentée par les temps de passage de ce véhicule aux stations dans lesquelles il doit s'arrêter. D'un point de vue mathématique, soit un véhicule i au début d'un cycle k dans la position P_i . Depuis cette position, l'objectif de régulation est qu'il arrive à la prochaine station, dont la position est L_s , à l'instant $t_{s,i}^c$. Le temps d'arrivée réel estimé du véhicule à cette station est $t_{s,i}$. Mathématiquement, respecter une table horaire revient à minimiser l'écart

$$t_{s,i} - t_{s,i}^c$$

Pour transformer ce critère, nous illustrons tout d'abord la situation par la figure 4.12.

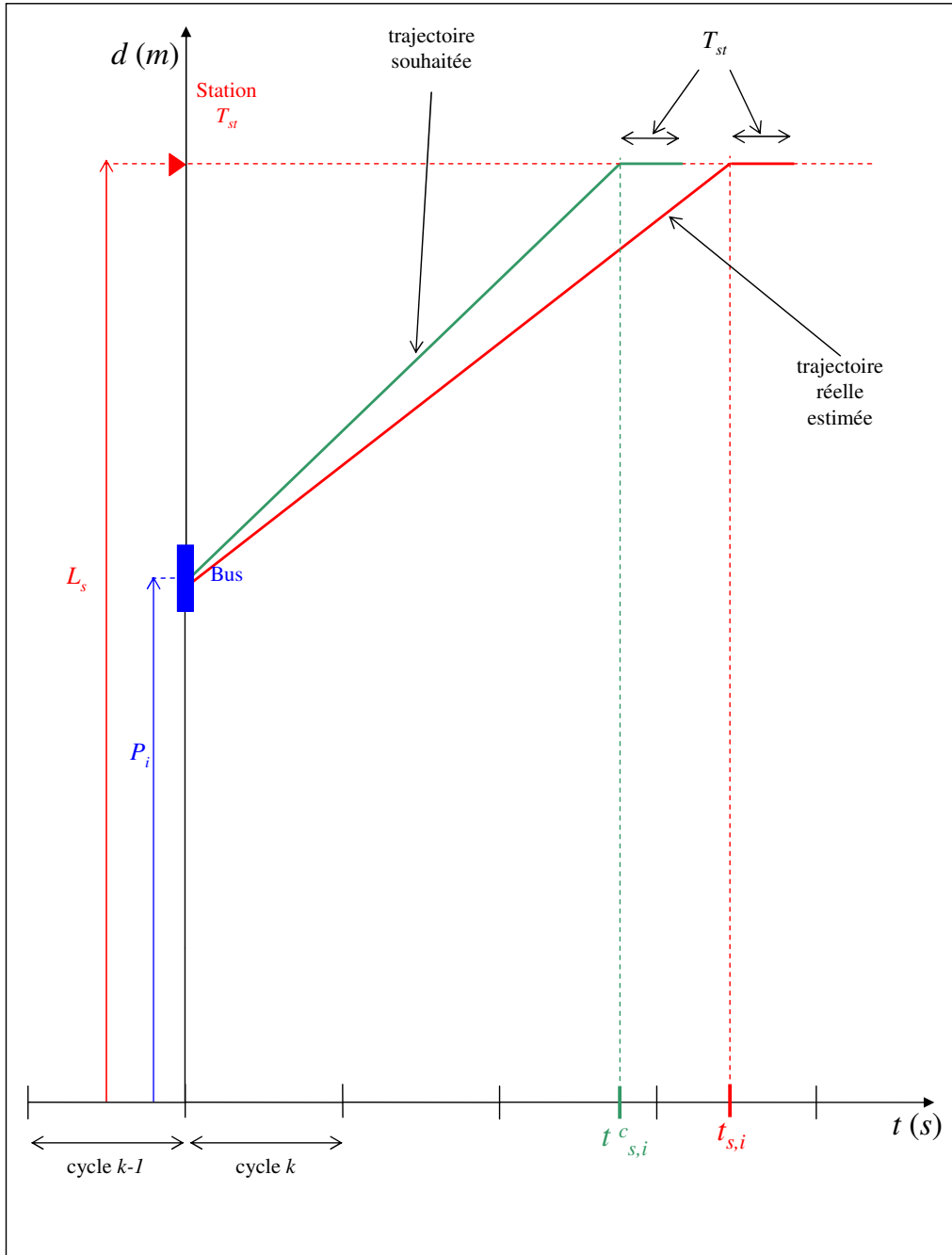


FIGURE 4.12 – Trajectoire souhaitée et trajectoire réelle estimée

Nous effectuons la division euclidienne de $t_{s,i}$ par C et nous pouvons donc écrire

$$t_{s,i} = n \times C + t_{res}$$

L'objectif de régulation devient alors "arriver dans le cycle $n + 1$ après t_{res} secondes à la position L_s où le véhicule devra s'arrêter pour une durée estimée à T_{st} ".

Si le véhicule doit quitter la station avant la fin du cycle $n + 1$, ce qui est équivalent à $C - t_{res} - T_{st} > 0$, alors l'objectif peut être formulé comme suit

$$\text{"arriver à la fin du cycle } n + 1 \text{ à la position } L_s + (C - t_{res} - T_{st}) \times V_b \text{"}$$

Rigoureusement, les deux objectifs ne sont équivalents du point de vue équivalence mathématique que dans le cas où nous supposons que le véhicule roulera à la vitesse V_b jusqu'à la fin du cycle. Même dans des cas où ceci n'est pas rigoureusement vérifié, il n'est pas très difficile de remarquer que l'écart entre les deux objectifs n'est pas trop important pour pouvoir poser un sérieux problème de régularité de la ligne. Ainsi, l'objectif final consiste à minimiser

$$P_i(n + 1) - (L_s + (C - t_{res} - T_{st}) \times V_b)$$

Dans le cas où le temps de stationnement prévu fait que le véhicule ne pourra quitter la station qu'au cycle suivant c'est à dire le cycle $n + 2$ alors l'objectif devient

$$\text{"arriver à la fin du cycle } n + 2 \text{ à la position } L_s + (2 \times C - t_{res} - T_{st}) \times V_b \text{"}$$

Le critère final se résume à minimiser

$$P_i(n + 2) - (L_s + (2 \times C - t_{res} - T_{st}) \times V_b)$$

Troisième cas : Objectif "respecter un intervalle régulier entre deux véhicules successifs"

C'est l'objectif de régulation pour les lignes de transport en commun à forte fréquence de passage. Théoriquement, respecter un intervalle régulier entre un véhicule de tête et un véhicule suiveur revient à ce que ce dernier respecte un horaire de passage pour la prochaine station. Soient un véhicule " i " suivi d'un véhicule " $i + 1$ " qui va s'arrêter dans une station " s ". L'objectif de régulation est de respecter un intervalle de passage entre les deux véhicules égal à h . Si le temps de passage du véhicule " i " est estimé ou est égal à $t_{s,i}$, alors le temps de passage du véhicule " $i + 1$ " dans la station doit être égal à

$$t_{s,i+1} = t_{s,i} + h$$

Ainsi l'objectif de régularité de l'intervalle est transformé en un objectif de respect d'un horaire de passage. Dans le paragraphe précédent, nous avons expliqué comment transformer cet objectif.

4.4.2.3 Module d'optimisation et architecture globale

Le problème d'optimisation formulé dans la section 4.1 consiste à déterminer les durées et les instants de début des phases de vert du cycle présent qui permettent de minimiser deux critères relatifs au trafic des véhicules particuliers et aux véhicules de transport en commun. Les critères sont exprimés uniquement pour le cycle présent.

Nous considérons un réseau régulier du type présenté au **chapitre 1** contenant N_c carrefours simples à 4 branches où seulement N_{cb} sont traversés par des lignes de transport en commun. Le réseau est traversé par n_l lignes de transport en commun. Au début du cycle k , n_m véhicules de transport en commun de la ligne m sont présents sur le réseau. Le problème d'optimisation à résoudre se présente sous la forme suivante :

Minimiser par rapport à $Td_1(k), \dots, Td_{N_{cb}}(k), G_1(k), \dots, G_{N_c}(k)$

$$\begin{cases} C_{-VP}(G) = K_p \times \sum_{i=1}^{N_a} X_i(j)^2 \\ C_{-TC}(T_d, G) = \sum_{m=1}^{n_l} \sum_{n=1}^{n_m} [P_{m,n}(j) - P_{m,n}^s(j)]^2 \end{cases}$$

Le problème ainsi posé a pour objectif de minimiser les deux critères pour le cycle k présent. Les commandes optimales déterminées ne permettent d'atteindre les objectifs de régulation que pour le cycle actuel. Pour tenir compte des conséquences de telles commandes sur l'état du système dans le futur, Il est nécessaire de poser le problème d'optimisation sur un horizon plus long. L'horizon de simulation sur lequel les critères sont calculés est noté H_s . L'horizon de contrôle sur lequel les commandes sont calculées est noté H_c et nous avons la relation $H_s \geq H_c$. Les variable de commande du cycle $k + H_c$ jusqu'au cycle $k + H_s$ sont les variables nominales qui sont fixes. Le nombre de variables de commande est alors $H_c \times (N_c + N_{cb})$ au lieu de $N_c + N_{cb}$ pour un problème d'optimisation formulé sur un seul cycle. Le nouveau problème d'optimisation est le suivant :

$$\text{Minimiser}_{U(k), \dots, U(k+H_c)} \begin{cases} C_{-VP}(G) = \sum_{j=k}^{k+H_s} K_p \times \sum_{i=1}^{4 \times N_c} X_i(j)^2 \\ C_{-TC}(T_d, G) = \sum_{j=k}^{k+H_s} \sum_{m=1}^{n_l} \sum_{n=1}^{n_m} [P_{m,n}(j) - P_{m,n}^s(j)]^2 \end{cases}$$

où

$$U(j) = [Td_1(j) \dots Td_{N_{cb}}(j), G_1(j) \dots G_{N_c}(j)]^T$$

Le problème ainsi posé est de très grandes dimensions. Si nous considérons un réseau de 16 carrefours simples à 4 branches dont 11 sont traversés par des lignes de transport en commun et que nous posons le problème de régulation sur un horizon de 6 cycles, le nombre de variables d'optimisation s'élève à $6 \times (16 + 11)$ c'est à dire 162!

Le critère de régulation pour les véhicules particuliers peut s'exprimer analytiquement puisque le modèle de prédiction du trafic des véhicules est analytique. Cependant, il reste fortement non linéaire. En ce qui concerne le critère pour les véhicules de transport en commun, le modèle de prédiction est fondé sur les événements et sur une vision très réaliste de la progression d'un véhicule de transport en commun. Il s'agit donc d'un modèle très complet mais aussi complexe. Pour résoudre ce problème, en temps réel, de très grande dimension et fortement non linéaire, nous utilisons les deux versions de l'algorithme d'optimisation par essaims particulières déjà introduits dans la section 3.

Les deux versions de l'algorithme des essais particuliers partagent un certain nombre de caractéristiques. L'opération fondamentale de cet algorithme est la mise à jour de la position et de la vitesse d'une particule donnée par l'équation. La trajectoire d'une particule dépend de quatre coefficients discutés dans les sections précédentes. Ce choix a une influence directe sur les trajectoires des particules dans l'espace de recherche et par conséquent sur la vitesse et la convergence de l'algorithme. Il a été établi par (Clerc et Kennedy) une relation qui permet d'assurer la convergence de l'algorithme. Il a été établi aussi l'existence d'une combinaison de coefficients permettant une convergence rapide. Cette combinaison est établie empiriquement et nous l'utilisons pour résoudre le problème d'optimisation multi-objectif. La combinaison est donnée par :

$$\begin{cases} \omega = \frac{1}{2 \cdot \ln(2)} \\ c_1 = c_2 = 0.5 + \ln(2) \end{cases}$$

Un autre paramètre très important pour l'algorithme est le nombre de particules de l'essaim qui dépend des dimensions de l'espace de recherche. Il est évident que plus il y a de particules, plus il y a de chances que l'algorithme converge. Il est aussi évident que plus il y a de particules, plus le temps d'exécution de l'algorithme est long. Dans un environnement de commande en temps réel, ce dernier point a une importance primordiale. Un algorithme qui n'est pas capable de converger dans la limite des contraintes temporelles n'a tout simplement aucune utilité. Des études empiriques ont montré que le nombre minimal de particules nécessaires pour trouver l'optimum d'un critère donné sur un espace de recherche $D_r = [-30; 30]^{30}$ est 10. Cependant pour réconcilier rapidité de convergence et qualité de l'optimum trouvé, le nombre de particules généralement utilisé est de 15 à 20 particules. Dans notre cas, nous utilisons 20 particules pour explorer l'espace de recherche défini comme suit :

$$[0; C]^{H_s \cdot N_{cb}} \times [G_{min}; G_{max}]^{H_s \cdot N_c}$$

Dans l'algorithme des essais particuliers, la mise à jour de la position et de la vitesse d'une particule se fait selon la meilleure position de la particule et selon la meilleure position des particules voisines qui dépend du type de voisinage choisie. Dans le cas mono-objectif, les meilleures positions sont choisies par rapport à la valeur du critère. Dans le cas multi-objectif, la sélection des meilleures particules et la mise à jour de la meilleure position d'une particule donnée est beaucoup plus complexe. Dans les sections précédentes, nous avons présenté les différents mécanismes qui permettent de mettre à jour la meilleure position d'une particule et de sélectionner les meilleures particules de l'essaim. Dans cette thèse, nous avons fait le choix de considérer les mécanismes suivants :

- La mise à jour de la meilleure position de la particule i : Soit $X_i(t+1)$ la nouvelle position (solution potentielle) de la particule et $X_i^*(t)$ la meilleure position jusqu'à l'itération t . Si $X_i(t+1)$ domine $X_i^*(t)$ alors $X_i^*(t+1) = X_i(t+1)$. Dans le cas où c'est $X_i^*(t)$ qui domine $X_i(t+1)$, alors $X_i^*(t+1) = X_i^*(t)$. Le cas le plus problématique concerne l'éventualité qu'aucune des deux solutions ne domine l'autre. Nous avons choisi de gérer cette situation de deux façons différentes. La première consiste à choisir aléatoirement entre les deux solutions avec des probabilités égales à 0.5 chacune. La deuxième consiste à considérer un

nouveau critère qui sera l'agrégation entre les deux critères considérés dans le problème d'optimisation. Ainsi, la solution qui permet de minimiser ce nouveau critère est choisie comme meilleure solution. Ainsi, étant donné les deux critères d'optimisation C_VP et C_TC , nous pouvons résumer le deuxième type de mise à jour de la meilleure particule comme suit :

```

si ( $C\_VP(X_i(t+1)) \leq C\_VP(X_i^*(t))$ ) et ( $C\_TC(X_i(t+1)) \leq C\_TC(X_i^*(t))$ ) alors
     $X_i^*(t+1) = X_i(t+1)$ 
sinon
    si ( $C\_VP(X_i(t+1)) \geq C\_VP(X_i^*(t))$ ) et ( $C\_TC(X_i(t+1)) \geq C\_TC(X_i^*(t))$ ) alors
         $X_i^*(t+1) = X_i^*(t)$ 
    sinon
         $X_i^*(t+1) = \operatorname{argmin}_{X \in \{X_i^*(t), X_i(t+1)\}} (C\_VP(X) + C\_TC(X))$ 
    fin si
fin si

```

- La sélection de la meilleure particule du voisinage : Le principe de sélection est le même que pour mettre à jour la meilleure position d'une particule. Soit une particule i , nous avons choisi de considérer comme voisinage, toutes les autres particules dont les indices sont regroupés dans l'ensemble $V_i = j \in [1; nb_prt], j \neq i$, $nb_prt = 20$ étant le nombre de particules de l'essaim. Chaque particule du voisinage a sa propre meilleure position dont la mise à jour a été évoquée ci dessus. Le principe de sélection de la meilleure particule du voisinage est simple. S'il y a une particule qui domine toutes les autres, alors la meilleure particule est celle là. Dans le cas où il y a plusieurs particules non dominées, nous envisageons deux façons de sélection. La première consiste à choisir aléatoirement entre les particules non dominées avec des probabilités égales. La seconde façon consiste à définir un nouveau critère comme celui défini pour la mise à jour de la meilleure position d'une particule. La sélection de la meilleure particule du voisinage est alors effectuée selon la valeur du nouveau critère. Ainsi, pour résumer, nous pouvons écrire :

```

si  $\exists j \in V_i$  tel que  $\forall p \in V_i$  et  $p \neq j$   $C\_VP(X_j^*(t)) \leq C\_VP(X_p^*(t))$  et  $C\_TC(X_j^*(t)) \leq C\_TC(X_p^*(t))$  alors
     $X_i^{vsq}(t) = X_j^*(t)$ 
sinon
     $X_i^{vsq}(t) = \min_{j \in V_i} (C\_VP(X_j^*) + C\_TC(X_j^*))$ 
fin si

```

- Mise à jour de l'archive externe : L'archive externe sert à sauvegarder toutes les solutions non dominées trouvées par l'algorithme au cours des itérations. La mise à jour de l'archive est effectuée après chaque itération. A l'itération t , la première itération consiste à déterminer la meilleure particule de l'essaim ou à défaut toutes les solutions non dominées de l'essaim. La deuxième étape consiste à sélectionner parmi ces solutions celles qui peuvent entrer dans l'archive. Si ces solutions dominent toutes celles de l'archive établies au fur et au mesure des itérations précédentes alors l'archive est initialisé à l'aide de ces solutions. Dans le cas contraire, les solutions non dominées de l'itération t sont rajoutées à l'archive. Dans ce cas, il faut supprimer toutes les solutions de l'ancien archive qui deviennent dominées par les nouvelles solutions. Une limite de capacité de l'archive est imposée afin d'éviter une explosion de la taille de l'archive qui aura

pour conséquence l'explosion des temps de calcul. Nous fixons la taille de l'archive à 100 solutions. La règle quand le nombre de solutions dépasse la taille limite consiste à donner la priorité aux solutions qui minimisent un nouveau critère défini comme la somme des deux critères : C_{VP} et C_{TC} .

Afin de récapituler, l'algorithme d'optimisation, au cycle k , a pour entrées :

- Les variables d'état mesurées au début du cycle k : il s'agit du nombre de véhicules dans chaque arc et des positions des véhicules de transport en commun présents dans le réseau.
- Des prédictions mesurées (statistiquement) sur les changements des volumes du trafic et des instants d'entrée des véhicules de transport en commun sur le réseau concerné. Ces prédictions sont nécessaires puisque l'optimisation est effectuée sur un horizon temporel dans le futur.
- Les objectifs de régulation : ce sont les trajectoires consignes que les variables d'état doivent approcher. Pour le trafic général, il s'agit d'un nombre de véhicules par arc généralement pris égal à 0. Pour les véhicules de transport en commun, il s'agit de positions à respecter au début de chaque cycle de feu.

L'algorithme permet de :

1. déterminer les durées et les instants de début des phases de vert pour les H_c futurs cycles qui composent l'horizon de commande.
2. seules les variables optimales pour le premier cycle k sont effectivement appliquées au système réel.

Le schéma en bloc du module de régulation est donné par le diagramme de la figure 4.13.

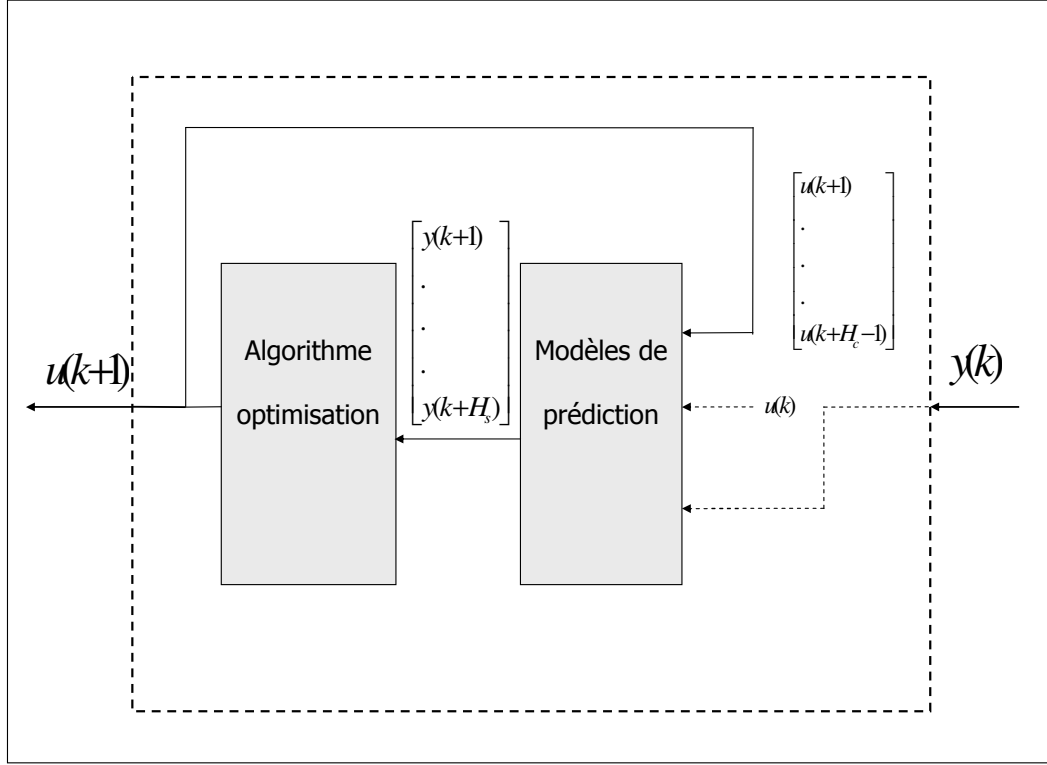


FIGURE 4.13 – Architecture du module de régulation

Le vecteur $y(k)$ est le vecteur des variables d'état du système au cycle k . Il est formé du nombre de véhicules pour chaque arc du réseau et de la position de chaque véhicules de transport en commun présent sur le réseau. Ainsi, si N_a et $n_b (= \sum_{i=1}^{n_l} p_i)$ désignent respectivement le nombre d'arcs et le nombre de véhicules de transport en commun présents sur l'arc, alors $y(k)$ peut être donné par :

$$y(k) = [X_1(k), \dots, X_{N_a}(k), P_1(k), \dots, P_{n_b}(k)]^T$$

Le vecteur $u(k)$ désigne le vecteur des commandes pour le cycle k et il est formé des instants de débuts des phases de vert pour les carrefours uniquement traversés par les lignes de transport en commun (N_{cb} carrefours) et des durées des phases de tous les carrefours (le nombre total est N_c). Le vecteur $u(k)$ s'écrit de la manière suivante :

$$u(k) = [Td_1(k), \dots, Td_{N_{cb}}(k), G_1(k), \dots, G_{N_c}(k)]^T$$

La stratégie est fondée sur la commande prédictive généralisée. Son architecture correspond à l'architecture classique de la commande prédictive avec un module complémentaire qui traduit les objectifs de régulation en des objectifs de respect d'une trajectoire consigne pour les véhicules de transport en commun. La figure suivante donne le diagramme de cette stratégie :

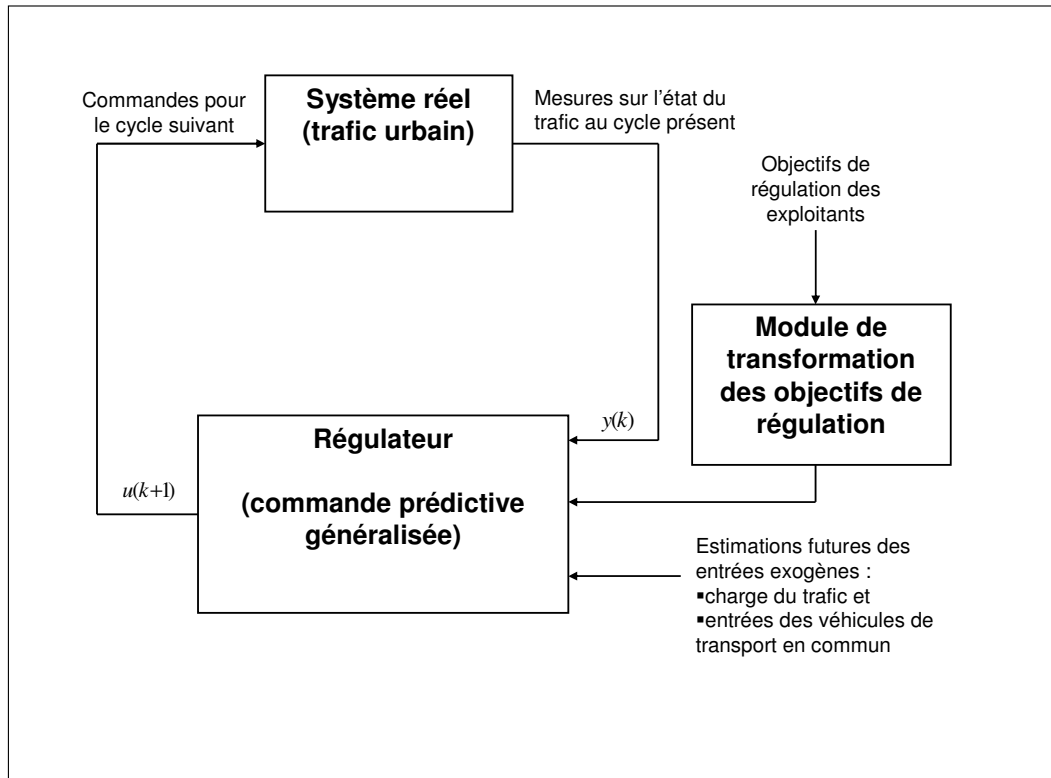


FIGURE 4.14 – Architecture complète de la stratégie de régulation

4.5 Stratégie secondaire

L'algorithme d'optimisation par essais particuliers est une méthode méta-heuristique. Les contraintes temporelles de temps réel du problème nécessitent que l'algorithme soit rapide en convergence. La solution la plus simple consiste à fixer un nombre d'itération maximal que l'algorithme ne peut dépasser. Cependant, cette solution peut avoir comme conséquence la convergence vers un optimum local ou tout simplement la non convergence. Aller rapidement pour trouver une mauvaise solution est bien inutile. Une nouvelle architecture fondée est proposée afin que l'algorithme converge avec une meilleure vitesse de convergence. Cette architecture reprend l'architecture de la stratégie principale avec un module supplémentaire fondé sur la commande linéaire quadratique.

4.5.1 Commande linéaire quadratique

L'objectif de cette commande est de déterminer les durées des phases de vert de tous les carrefours du réseau afin de minimiser le nombre total des véhicules sur le réseau.

L'évolution du nombre de véhicules sur chaque arc du réseau est régie par une équation de conservation déjà décrite dans le chapitre 2. Étant donné un arc " i ", le nombre de véhicules présents dans cet arc au cycle $k + 1$ peut s'écrire comme suit

$$X_i(k+1) = X_i(k) + C \times (u_i(k) - q_i(k))$$

où C est la durée du cycle, $u_i(k)$ et $q_i(k)$ sont respectivement les débits entrant et sortant de l'arc i durant le cycle k .

La modélisation des débits entrant et sortant s'appuie sur le modèle *store-and-forward*. Les détails de la modélisation ont été déjà présentés dans le chapitre 2. Si nous désignons par $G_i(k)$ la durée totale du vert allouée à l'arc i durant le cycle k , la dynamique de l'arc i peut s'écrire comme suit

$$X_i(k+1) = X_i(k) + \sum_{w \in In_{j1}} (\tau_{w,i} \cdot S_w \cdot G_w(k)) - S_i \cdot G_i(k)$$

Nous assumons l'existence d'un état de non saturation du trafic c'est à dire l'existence de valeurs nominales G^N des durées des phases de vert tel que :

$$0 = 0 + \sum_{w \in In_{j1}} (\tau_{w,i} \cdot S_w \cdot G_w^N(k)) - S_i \cdot G_i^N(k)$$

Ces valeurs sont généralement les valeurs fixes que les exploitants du trafic déterminent sous les conditions nominales du trafic.

En effectuant la différence entre les deux équations précédentes, nous pouvons écrire :

$$X_i(k+1) = X_i(k) + \sum_{w \in In_{j1}} (\tau_{w,i} \cdot S_w \cdot \Delta G_w(k)) - S_i \cdot \Delta G_i(k)$$

où

$$\Delta G_i(k) = G_i(k) - G_i^N(k)$$

En généralisant cette équation pour tous les arcs du réseau, nous pouvons écrire

$$X(k+1) = A \times X(k) + B \times \Delta G(k)$$

où

$$\begin{aligned} X(k) &= [X_1(k), \dots, X_i(k), \dots, X_{N_a}(k)]^T \\ \Delta G(k) &= [\Delta G_1(k), \dots, \Delta G_i(k), \dots, \Delta G_{N_a}(k)]^T \end{aligned}$$

Les matrices A et B sont définies par :

$$\begin{cases} A(i, i) = 1 \quad \forall i \in \{1, \dots, N_a\} \\ A(i, j) = 0 \quad \forall j \neq i \end{cases}$$

$$\begin{cases} B(i, j) = \tau_{i,j} \text{ si l'arc } i \text{ est en amont l'arc } j \\ B(i, j) = 0 \text{ sinon} \end{cases}$$

Le système défini par les équations ci-dessus est un système linéaire. Le problème d'optimisation est formulé comme suit

$$\text{minimiser}_{\Delta G_1(k), \dots, \Delta G_{N_a}(k)} \sum_{i=1}^{N_a} [\alpha_i \times X_i^2(k+1) + \beta_i \times \Delta G_i(k+1)^2]$$

En réalité, l'objectif est de minimiser le nombre total des véhicules sur le réseau. Le terme qui fait appel aux commandes ΔG est nécessaire pour limiter l'écart entre les valeurs à appliquer au système et les valeurs nominales G^N .

Le critère peut être réécrit sous la forme matricielle suivante

$$X^t(k) \times Q \times X^t(k) + G^N(k)^t \times R \times G^N(k)$$

où les matrices de pondération R et Q sont des matrices diagonales définies par :

$$Q(i, i) = \alpha_i, \quad i \in \{1, \dots, N_a\}$$

$$R(i, i) = \beta_i, \quad i \in \{1, \dots, N_a\}$$

Le problème ainsi formulé a pour objectif de minimiser le nombre total des véhicules dans le réseau seulement sur le cycle prochain. Cependant comme il a été expliqué dans d'autres sections, il est nécessaire de minimiser ce critère sur un horizon plus long. Dans ce cas précis d'un système linéaire avec un critère quadratique, la résolution de ce problème d'optimisation est beaucoup plus aisée et beaucoup plus simple quand l'horizon d'optimisation est infini. La solution est donnée par la résolution de l'équation de Riccati. Si le vecteur de commande recherché $\Delta G(k)$ s'écrit sous la forme $\Delta G(k+1) = -K_c \times X(k)$, la matrice K_c appelée matrice de gain est donnée par :

$$K_c = R^{-1} \times B^t \times P_c$$

P_c est la matrice solution de l'équation de Riccati

$$P_c \times A + A^t \times P_c - P_c \times B \times R^{-1} \times B^t \times P_c + Q = 0$$

L'intérêt de cette architecture et la commande LQ réside dans le fait que la détermination des commandes optimales à appliquer au système ne nécessite qu'une simple multiplication matricielle. Étant donné les mesures sur l'état du trafic $X(k)$ au cycle k , les commandes optimales pour le cycle suivant $k+1$ sont données par $\Delta G(k+1) = -K_c \times X(k)$. La matrice de gain K_c est déterminée hors ligne puisque elle ne dépend que des matrices A et B du modèle et des matrices R et Q du critère d'optimisation. Ces quatre matrices sont constantes. Partant de l'expression de $\Delta G(k+1) = G(k+1) - G^N(k+1)$, les commandes sont données par :

$$G(k+1) = G^N(k+1) - K_c \times X(k)$$

Cette commande constitue le noyau de la stratégie *TUC*, une composante de la stratégie intégrale *IN-TUC*. Elle est utilisée aussi dans la commande *NeTPrior* qui a pour objectif la régulation simultanée du trafic

général et des véhicules de transport en commun. La stratégie *TUC* a été déployée dans les villes de Glasgow en Écosse et Chania en Grèce et été testée sous le logiciel *METACOR* sur des réseaux des deux villes. Le réseau de Glasgow comporte 13 carrefours et 61 arcs. Celui de Chania est constitué de 17 carrefours et 79 arcs. La stratégie a été comparée à une stratégie de plan de feu fixe :

- Pour Glasgow, le temps de parcours total des véhicules a été réduit de 19 – 34% et le temps total passé dans le réseau de 20 – 54%.
- Pour Chania, le temps de parcours total a été réduit de 19 – 78%, le temps total passé dans le réseau de 32 – 91% et la consommation de carburant de 23 – 86%.

Pour les deux villes, les résultats montrent que la stratégie a évité la formation de congestion au niveau des carrefours permettant ainsi d'améliorer les conditions du trafic.

Dans le cadre de l'évaluation de la stratégie *NeTPrior*, nous avons réalisé des tests en simulation en comparaison avec des plans de feu fixes. Les simulations sont effectuées sous le logiciel *Dynasim* et pour un réseau de 5 carrefours dans le sud de la ville de Paris (figure 4.15).

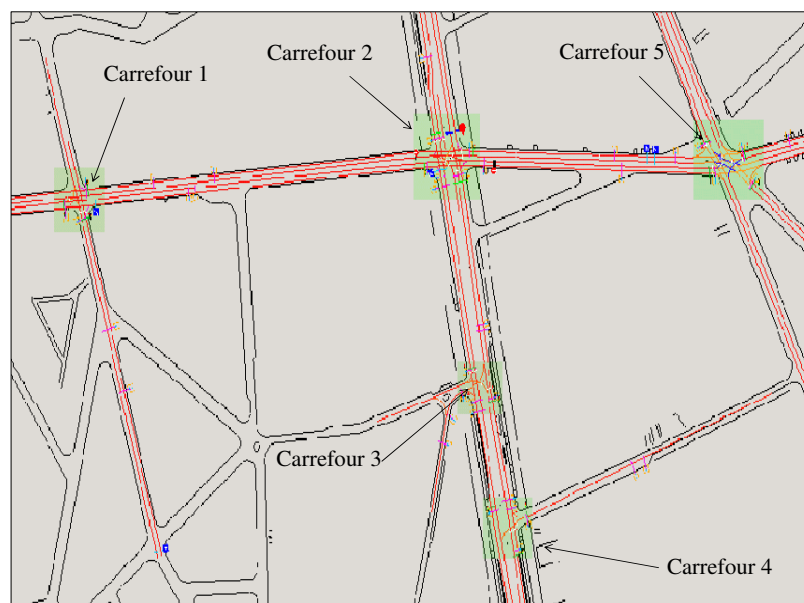


FIGURE 4.15 – Réseau de la ville de Paris modélisé sous Dynasim

Les simulations sont effectuées sur une période de deux heures et 15 minutes avec une charge nominale de trafic correspondant à une période de pointe. Afin d'examiner les avantages de la stratégie par rapport à une stratégie de plans de feu fixes, nous avons introduit des perturbations sur le volume du trafic suivant le scénario du tableau 4.1 :

Heure de simulation	Perturbations
7h45 - 8h00	pas de perturbations
8h00 - 8h30	+25% sur l'axe avenue d'Italie
8h30 - 9h00	+25% sur l'axe rue de Tolbiac
9h00 - 9h30	+25% sur l'axe avenue d'Italie
9h30 - 10h00	+25% sur l'axe rue de Tolbiac

TABLE 4.1 – Perturbations sur les volumes de trafic

Le réseau considéré est traversé par deux lignes de bus sur les deux axes principaux avenue d'Italie et rue de Tolbiac. L'objectif de cette stratégie est la régulation simultanée du trafic général et de la circulation des véhicules de transport en commun. Cependant, nous nous intéressons ici seulement au trafic général. Sur la figure 4.16, nous traçons l'évolution du nombre de véhicules sur deux arcs du carrefour 4 ainsi que les valeurs des commandes (durées de vert) en comparaison avec une stratégie de plans de feux fixes.

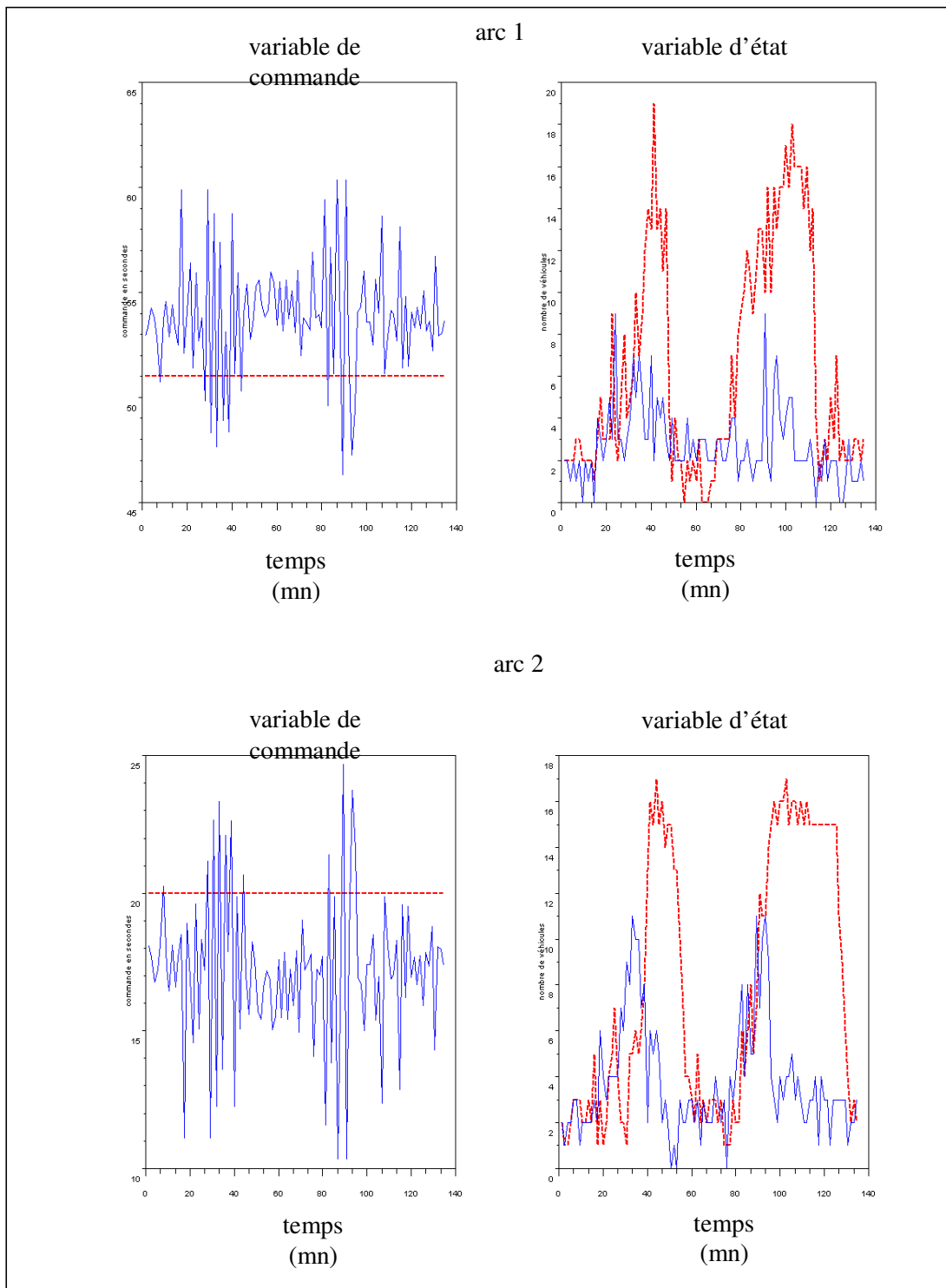


FIGURE 4.16 – Nombre de véhicules et valeurs des durées des feux verts pour le carrefour 4

Des résultats plus complets montrent que cette stratégie est très bien adaptée pour des conditions saturées et qu'elle est moins efficace quand le trafic est moins saturé.

La commande linéaire quadratique est une commande simple à mettre en oeuvre et permet de respecter les contraintes de temps réel. Cependant, elle présente deux inconvénients :

- La matrice de gain permettant de calculer les commandes optimales est déterminée hors ligne et elle ne permet pas de prendre en compte des changements des taux des mouvements tournants dans les carrefours.
- Le modèle est basé sur l'hypothèse *store-and-forward*. Une hypothèse qui se vérifie facilement pour des conditions saturées du trafic mais pas pour des conditions de trafic moins saturé.

La commande LQ sera utilisée pour initialiser les durées des phases de vert pour l'algorithme d'optimisation par essais particuliers.

4.5.2 Architecture

L'intérêt de cette nouvelle architecture est d'accélérer la convergence de l'algorithme d'optimisation. La rapidité de convergence de l'algorithme PSO dépend fortement des dimensions de l'espace de recherche. Bien que l'algorithme PSO procède à une exploration efficace et économique de l'espace de recherche, il est évident qu'au cours de l'exploration l'algorithme passera par de nombreuses régions non prometteuses. Ceci a pour conséquences une recherche inutile et donc un gaspillage et surtout un risque d'une convergence prématurée dans une région qui contient un optimum local. Disposer de connaissances à priori sur l'espace de recherche et les régions prometteuses seraient une excellente solution pour orienter l'algorithme vers ces régions.

Dans le cas de la régulation multimodale du trafic sur de larges réseaux urbains, les connaissances à priori sur l'espace de recherche et les valeurs des critères dans certaines régions de l'espace sont inexistantes. Une solution que nous avons imaginée consiste à trouver des régions prometteuses en utilisant un autre algorithme plus simple, très rapide qui orientera l'algorithme PSO vers des régions bien particulières.

Pour les durées des phases de vert, la commande LQ permet de déterminer les valeurs optimales des phases de vert. Ces valeurs sont optimales par rapport au modèle linéaire utilisé qui n'est pas aussi précis que le modèle que nous utilisons. Aussi, elles ne sont optimales qu'au sens de la minimisation du seul critère relatif aux véhicules particuliers tandis que le problème d'optimisation est multi-objectif et concerne la minimisation d'un critère pour les véhicules particuliers et un autre pour les véhicules de transport en commun.

Concernant les instants de début des phases de vert pour les arcs (ou les carrefours) traversés par des lignes de transport en commun, une solution simple serait d'orienter l'algorithme PSO vers des régions autour des valeurs fixes utilisées par les exploitants des infrastructures urbaines. Cependant ces valeurs sont généralement déterminées afin de donner une "*onde verte*" pour les véhicules particuliers. Une première solution consiste à adapter ces valeurs pour les véhicules de transport en commun. Cependant ces valeurs sont déterminées hors ligne suivant des données statistiques. Une simple variation des conditions du trafic,

des temps de stationnement des véhicules de transport en commun dans les stations ou de la vitesse des véhicules peuvent rendre ces valeurs obsolètes et aucune "onde verte" n'est plus garantie. Par conséquent, l'algorithme explorera, pour les instants de début de phase, tout l'espace de recherche.

Il suffit d'appliquer la commande LQ pour les mesures collectées sur le terrain. Les valeurs des variables de commande ainsi déterminées orienteront l'algorithme PSO à trouver les optimums recherchés. Pour cela, il est suffisant de rechercher dans un périmètre restreint autour de ces valeurs. Ainsi, le domaine de recherche initial est réduit à un domaine plus restreint. Ceci permettra à l'algorithme PSO d'être plus efficace et plus rapide en convergence.

Dans le cas d'un réseau de N_c carrefours simples, le domaine de recherche initial est $[0; C]^{N_{cb}} \times [G_{min}; G_{max}]^{N_c}$. La commande LQ détermine le vecteur de commande U^{init} donné par :

$$U^{init} = \begin{bmatrix} 1 \\ \cdot \\ \cdot \\ G_{N_c} \end{bmatrix}$$

Si nous désignons par P_G le périmètre de la nouvelle recherche pour les durées des phases de vert, le nouvel espace de recherche est donné par :

$$\left[\max(G_{min}, G_1 - \frac{P_G}{2}); \min(G_{max}, G_1 + \frac{P_G}{2}) \right] \times \dots \times \left[\max(G_{min}, G_{N_c} - \frac{P_G}{2}); \min(G_{max}, G_{N_c} + \frac{P_G}{2}) \right]$$

Il n'existe pas de règles pour déterminer les valeurs du diamètre P_G . Dans notre cas, nous fixons sa valeur à 20.

L'architecture complète de la nouvelle stratégie est illustrée par la figure 4.17.

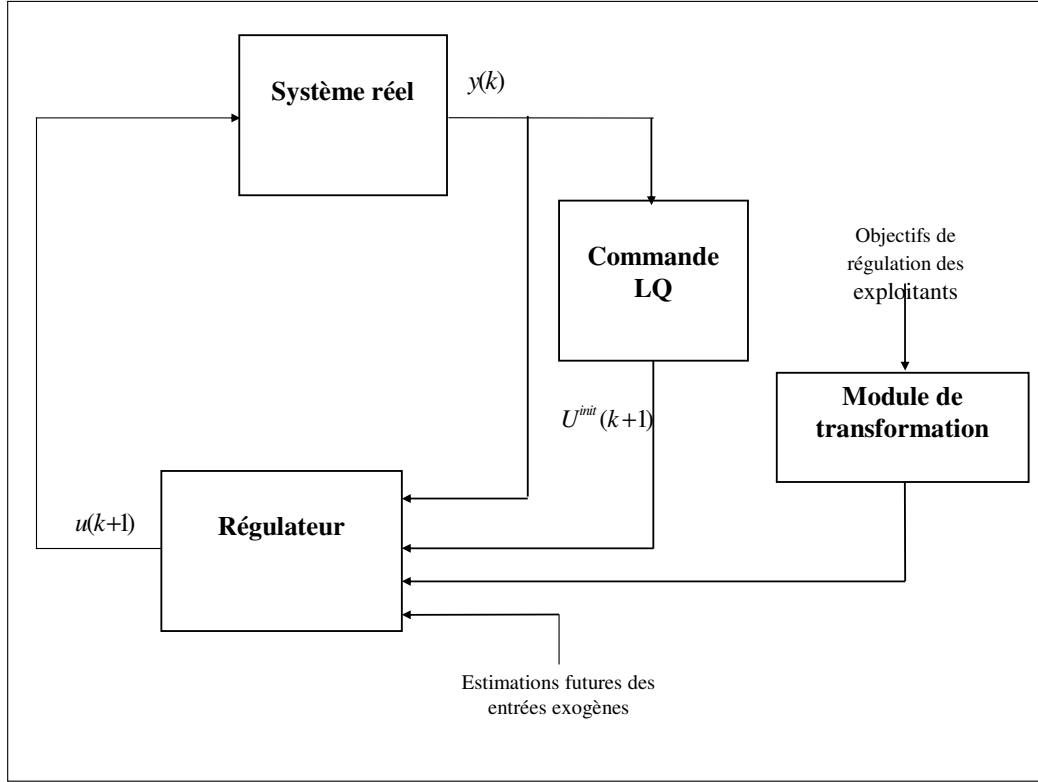


FIGURE 4.17 – Architecture complète de la stratégie secondaire

Le cycle présent étant le cycle k , les différentes étapes du nouvel algorithme sont :

1. Collecte des mesures sur l'état du trafic : le nombre de véhicules sur tous les arcs formant le vecteur $X(k)$ et les positions de tous les véhicules de transport en commun formant le vecteur $P(k)$.
2. Détermination des valeurs initiales, pour le cycle $k + 1$, pour les durées des phases de vert formant le vecteur $U^{init}(k + 1)$.
3. Détermination du nouvel espace de recherche D_R défini comme un hypercube centré sur le point de coordonnées $U^{init}(k + 1)$.
4. Application de la commande principale avec le nouvel espace de recherche D_R pour l'algorithme *PSO*.
5. Détermination des valeurs optimales des variables de commande pour tous les cycles de l'horizon de commande.
6. Envoi vers le système réel les valeurs de commande uniquement pour le prochain cycle $k + 1$.

4.6 Conclusion

Dans ce chapitre, nous avons présenté l'architecture de la régulation du trafic urbain. La régulation s'intéresse aux deux modes de transport : les véhicules particuliers et les véhicules de transport en commun et est posée pour de larges réseaux urbains. Ce problème de régulation pose deux sérieuses difficultés.

D'une part, la formulation du problème sous forme d'un problème d'optimisation bi-objectif présente la difficulté de la manière dont les deux critères doivent rendre compte des objectifs réels des exploitants du trafic urbain. Pour les véhicules particuliers, le critère à minimiser correspond au nombre total des véhicules présent sur le réseau. Ce critère permet de minimiser le temps de parcours moyen des usagers des véhicules particuliers. Pour les véhicules de transport en commun, le but des exploitants est la régularité des lignes. Le critère que nous avons construit permet de mesurer exactement la régularité de la ligne et par conséquent de la maximiser en minimisant l'écart entre la trajectoire réelle des véhicules et la trajectoire souhaitée.

D'autre part, les contraintes de temps réel de la régulation nécessite que le temps d'exécution pour la résolution du problème d'optimisation bi-objectif ne soit pas trop important. Pour respecter cette contrainte, nous utilisons la méthode d'optimisation par essais particuliers (*PSO*). L'algorithme *PSO* est une méta-heuristique très efficace pour résoudre les problèmes difficiles à grandes dimensions. Pour aider davantage l'algorithme à converger rapidement, nous utilisons une deuxième architecture fondée sur la commande linéaire quadratique (*LQ*) pour initialiser les variables de décision.

L'application de la régulation sur un réseau urbain est effectuée sur un réseau virtuel. Les résultats numériques et l'étude en simulation seront exposés dans le chapitre suivant.

Chapitre 5

Étude en simulation de la stratégie de régulation

5.1 Introduction

Dans le chapitre précédent, nous avons présenté l'algorithme d'optimisation, les critères construits pour mesurer le degré de régulation et l'architecture de la stratégie. Dans les chapitres 2 et 3, nous avons présenté les modèles et des résultats de simulation sur de petits réseaux routiers simples.

Dans ce chapitre, nous nous proposons d'étudier en simulation la stratégie développée. Dans un premier temps, nous présentons le réseau sur lequel les tests seront réalisés. Nous décrivons ses caractéristiques géométriques ainsi que les flux de véhicules particuliers et des transports en commun qui le traversent. Dans un second temps, une étude est effectuée sur l'influence de quelques paramètres de l'algorithme d'optimisation. Ensuite nous nous intéressons à l'étude de l'influence de l'horizon de simulation et de l'horizon de contrôle sur les résultats de la stratégie. Une fois les paramètres de l'algorithme d'optimisation et la longueur des horizons fixés, nous étudions l'influence des valeurs mathématiques des objectifs de régulation. Enfin la dernière partie sera dédiée à l'analyse des résultats pour les véhicules particuliers et les véhicules de transport en commun.

5.2 Le réseau routier urbain

Afin de vérifier les performances de la stratégie développée, nous réalisons une série de tests en simulation sur un réseau routier urbain. Le réseau est virtuel et possède beaucoup de caractéristiques des réseaux réels. Il peut servir comme un réseau de test pour comparer différentes stratégies entre elles.

5.2.1 Caractéristiques géométriques

Une des originalités de ce travail de thèse réside dans la capacité de la stratégie développée à réguler le trafic sur de larges réseaux urbains en utilisant une architecture centralisée. Pour vérifier les performances

de la stratégie, nous considérons un réseau virtuel composé de 16 carrefours numérotés de 1 jusqu'à 16. Il comporte 11 entrées, les sorties du réseau sont ignorées étant donné que les arcs de sortie ne sont pas pris en compte dans la stratégie. La figure 5.1 donne un aperçu du réseau sous la forme d'un graphe direct.

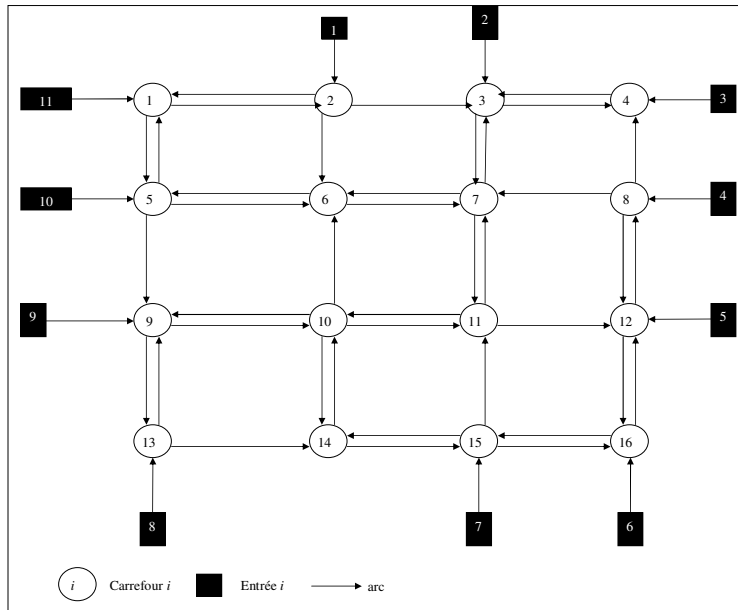


FIGURE 5.1 – Réseau virtuel de test

Le réseau ainsi construit est composé de 51 arcs qui relient les carrefours entre eux. La stratégie part de l'hypothèse que quel que soit l'état du trafic, un flux de véhicules a toujours la possibilité d'entrer dans un arc. Ceci peut être garanti si les arcs sont supposés de capacités infinies ou si elles sont très supérieures à la charge du trafic sur les différents arcs. Du point de vue des véhicules particuliers, il n'est donc pas nécessaire de spécifier les longueurs des arcs (équivalentes aux capacités). Nous verrons par la suite que du point de vue des transports en commun, il est nécessaire de spécifier les longueurs des arcs seulement traversés par ces véhicules.

Nous avons expliqué dans le chapitre 2 la manière de prise en compte des arcs contenant plusieurs lignes. Les arcs considérés ici n'en contiennent qu'une seule. Les véhicules tournant à gauche et à droite partagent la même ligne que les autres véhicules. Les véhicules de transport en commun ne disposent pas de sites propres comme ça peut être le cas dans certaines agglomérations. Toutes ces considérations nous placent dans le cas où tous les flux de véhicules particuliers et de transports en commun partagent une seule même ligne.

5.2.2 Caractéristiques des flux des véhicules particuliers

Tous les carrefours fonctionnent avec le même cycle C de durée 80 secondes. Les simulations sont effectuées sur 40 cycles de feux ce qui correspond à 54 minutes. Nous supposons, qu'initialement, le réseau est totalement vide. Le réseau comporte 11 entrées par lesquelles les flux de véhicules entrent. Les valeurs des flux

Entrées	1	2	3	4	5	6	7	8	9	10	11
Flux moyens	23	20	16	19	23	24	14	25	21	16	19

TABLE 5.1 – Flux entrants moyens

Cycles	1	2	3	4	5	6	7	8	9	10
Entrée 1	24	18	23	9	3	2	19	25	24	19
Entrée 2	17	5	10	12	25	22	18	13	14	15
Entrée 3	14	0	17	17	17	23	23	22	3	15
Entrée 4	13	25	9	27	0	16	27	26	14	25
Entrée 5	23	23	12	21	23	23	23	19	2	22
Entrée 6	17	32	20	8	24	26	18	17	28	27
Entrée 7	18	8	16	0	18	11	18	18	18	13
Entrée 8	5	31	21	14	23	32	32	16	26	22
Entrée 9	30	13	25	28	13	18	18	17	30	25
Entrée 10	10	23	13	3	20	12	0	8	20	23
Entrée 11	9	13	11	27	13	23	17	19	5	23

TABLE 5.2 – Flux entrants pour les 10 premiers cycles

entrants vont déterminer le degré de saturation du réseau. Dans ce travail, nous définissons les flux comme le nombre de véhicules entrants par cycle de feu. Le tableau 5.1 donne les valeurs moyennes des flux entrants pour les 11 entrées du réseau :

La valeur moyenne des flux entrants moyens est égale à 22 véhicules/cycle. Dans l'hypothèse où chaque arc dispose, en moyenne, d'une période de vert de 40 secondes ($= \frac{C}{2}$) et où le taux de saturation est de 0.5 véhicule/cycle, ces valeurs de flux permettent d'avoir une circulation à la limite de la saturation.

Pour les flux entrants, nous avons considéré un générateur à distribution de poisson avec les valeurs moyennes données dans le tableau 5.1. Le générateur a été appliqué pour les 10 premiers cycles, les valeurs réelles des flux entrants pour les 11 entrées sont données dans le tableau 5.2.

Le trafic sur les 10 premiers cycles permet d'avoir des conditions à la limite de la saturation mais fluide. Il permet en outre d'avoir un trafic équilibré entre les différentes zones du réseau. Afin d'étudier la stratégie sur des situations de trafic plus riches et plus variées, nous introduisons des perturbations sur les flux entrants pour les 30 cycles restants. Les perturbations sont introduites grâce aux 6 paramètres suivants :

- Les paramètres d'augmentation des flux : $a_1 = 1.2$, $a_2 = 1.3$ et $a_3 = 1.4$.
- Les paramètres de réduction des flux : $b_1 = 0.7$, $b_2 = 0.8$ et $b_3 = 0.6$.

Les 40 cycles de simulation sont divisés en 4 périodes chacune composée de 10 cycles. La première période

Périodes (en 10 cycles)	Période 1 (1-10)	Période 2 (11-20)	Période 3 (21-30)	Période 4 (31-40)
Entrée 1	trafic nominal	trafic nominal	multiplication par b_2	trafic nominal
Entrée 2	trafic nominal	trafic nominal	multiplication par b_2	trafic nominal
Entrée 3	trafic nominal	multiplication par a_1	trafic nominal	multiplication par b_3
Entrée 4	trafic nominal	multiplication par a_1	trafic nominal	multiplication par b_3
Entrée 5	trafic nominal	multiplication par a_1	trafic nominal	multiplication par b_3
Entrée 6	trafic nominal	multiplication par a_1	trafic nominal	multiplication par b_3
Entrée 7	trafic nominal	multiplication par a_1	trafic nominal	multiplication par b_3
Entrée 8	trafic nominal	multiplication par a_1	trafic nominal	multiplication par b_3
Entrée 9	trafic nominal	trafic nominal	multiplication par a_2	multiplication par b_3
Entrée 10	trafic nominal	trafic nominal	multiplication par a_2	multiplication par b_3
Entrée 11	trafic nominal	trafic nominal	multiplication par a_2	multiplication par b_3

TABLE 5.3 – Perturbations des flux entrants

correspond à un trafic nominal donné par les flux du tableau 5.2. Les trois dernières périodes sont les seules concernées par la modification des flux. Chaque paramètre a_i ou b_i correspond à la modification des flux pour la période $i \in \{1, 2, 3\}$. Le tableau 5.3 résume les perturbations des flux pour les trois dernières périodes :

Chaque tronçon de route est caractérisé par son taux de flux de saturation qui est défini comme le nombre maximal de véhicules qui peuvent traverser une section en une unité de temps. Dans le cas d'un tronçon urbain avec une seule ligne, le taux de saturation varie très légèrement autour de 0.5 véhicule/seconde. C'est cette valeur que nous considérons pour tous les arcs du réseau. La circulation des flux dans le réseau est déterminée par les taux des mouvements tournants. Ainsi entre un arc " i " et un arc " j ", $\tau_{i,j}$ désigne le pourcentage des véhicules allant de " i " vers " j " par rapport au nombre total de véhicules présents sur l'arc " i ". Il est bien évident que les taux des mouvements directs sont supérieurs aux taux des mouvements de tourne à droite qui sont quant eux supérieurs aux taux des mouvements de tourne à gauche. Tous les taux sont regroupés dans une matrice carrée de dimension 51, le nombre d'arcs du réseau.

5.2.3 Caractéristiques des lignes de transport en commun

L'un des deux objectifs principaux de la stratégie est la régulation de la circulation des véhicules de transport en commun de surface sur de larges réseaux urbains et pour différentes situations de trafic. Pour bien étudier les performances de la stratégie pour les objectifs fixés, nous considérons quatre lignes dont les trajectoires sont représentées sur la figure 5.2.

Trajectoire	E3	C4	S1	C3	C7	C11	C10	S2	C9	C13
Temps de stationnement (s)	—	—	20	—	—	—	—	15	—	—
Positions (mètres)	0	190	250	350	580	690	940	1140	1200	1430

TABLE 5.4 – Trajectoires et distances pour la ligne 1

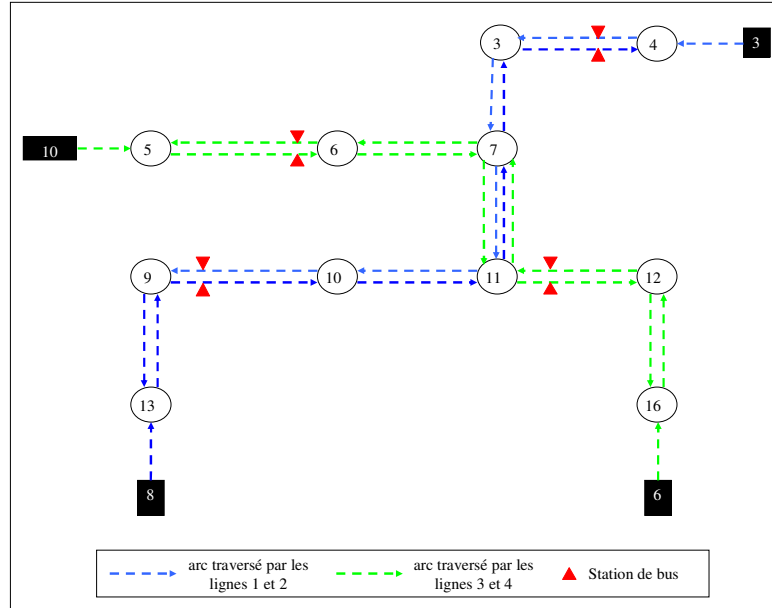


FIGURE 5.2 – Réseau virtuel de test

Les lignes ont les caractéristiques suivantes :

- Les lignes traversent le réseau du nord au sud et d’est en ouest.
- Certaines lignes partagent les mêmes arcs.
- Certaines lignes sont en conflit sur plusieurs carrefours.
- Chaque ligne possède deux stations : l’une placée loin et l’autre près des carrefours à feux.

Chaque entrée d’une ligne est prise comme référence spatiale pour les véhicules. Ainsi, à l’entrée, chaque véhicule a la position 0 m. Les tableaux 5.4 jusqu’à 5.7 résume les positions des carrefours à traverser et des stations pour chaque ligne (les positions sont référencées par rapport à l’entrée de chaque ligne). E désigne une entrée du réseau, C un carrefour et S une station.

Les lignes 1 et 2 traversent le réseau en sens contraire tout comme les lignes 3 et 4. Pour les deux premières lignes, nous avons fixé une fréquence de passage égale à un véhicule tous les quatre cycles ce qui correspond

Trajectoire	E8	C13	C9	S1	C10	C11	C7	C3	S2	C4
Temps de stationnement (s)	–	–	–	15	–	–	–	–	20	–
Positions (mètres)	0	210	440	500	700	950	1060	1290	1390	1450

TABLE 5.5 – Trajectoires et distances pour la ligne 2

Trajectoire	E1	C5	S1	C6	C7	C11	S2	C12	C16
Temps de stationnement (s)	–	–	15	–	–	–	20	–	–
Positions (mètres)	0	320	460	500	730	840	910	1130	1350

TABLE 5.6 – Trajectoires et distances pour la ligne 3

à 1 véhicule toutes les 5 minutes et 20 secondes. Pour les lignes 3 et 4, la fréquence de passage est égale à un véhicule tous les 7 cycles ce qui correspond à un véhicules toutes les 9 minutes et 20 secondes. Les lignes sont à très haute fréquence pour les deux premières et à fréquence moyenne pour les deux dernières.

5.2.4 La stratégie des plans de feux fixes

Nous comparons la stratégie que nous avons développée avec une stratégie de plans de feux fixes. Dans le cas d'un réseau réel, les plans de feux fixes sont déterminés par les exploitants du réseau routier. Ils peuvent varier selon les périodes de la journée. Ces plans sont déterminés par exemple pour maximiser les flux ou minimiser les temps d'attente des véhicules. Dans le cas du réseau virtuel sur lequel nous effectuons les simulations, il est alors nécessaire de déterminer les plans de feux fixes. Un plan de feux fixes déterminé de manière aléatoire peut avoir des conséquences très néfastes sur le trafic ce qui peut faire montrer des performances exceptionnelles de notre stratégie alors qu'en réalité elles peuvent être pas aussi exceptionnelles voire médiocres. Afin d'avoir un aperçu sur les réelles performances de la stratégie, nous la comparons avec la stratégie de plans de feux fixes la plus performante selon les critères que la stratégie développée essaie d'atteindre. La stratégie de plans de feux fixes a les deux objectifs suivants :

- Minimiser le nombre de véhicules particuliers sur les arcs en fixant les durées des phases de vert.
- Minimiser les temps d'attente des véhicules de transport en commun en garantissant une "onde verte" (en fixant les instants de début des phases de vert).

Trajectoire	E6	C16	C12	S1	C11	C7	C6	S2	C5
Temps de stationnement (s)	0	0	0	20	0	0	0	15	0
Positions (mètres)	0	260	480	700	770	880	1110	1150	1290

TABLE 5.7 – Trajectoires et distances pour la ligne 4

Tous les carrefours du réseau fonctionnent seulement avec deux phases. Comme nous l'avons expliqué dans le chapitre 2, il suffit alors de spécifier l'instant de début de vert et la durée d'une seule phase pour définir complètement le plan de feu d'un carrefour. Afin d'expliquer comment déterminer ces deux variables, nous considérons l'exemple du carrefour de la figure 5.3.

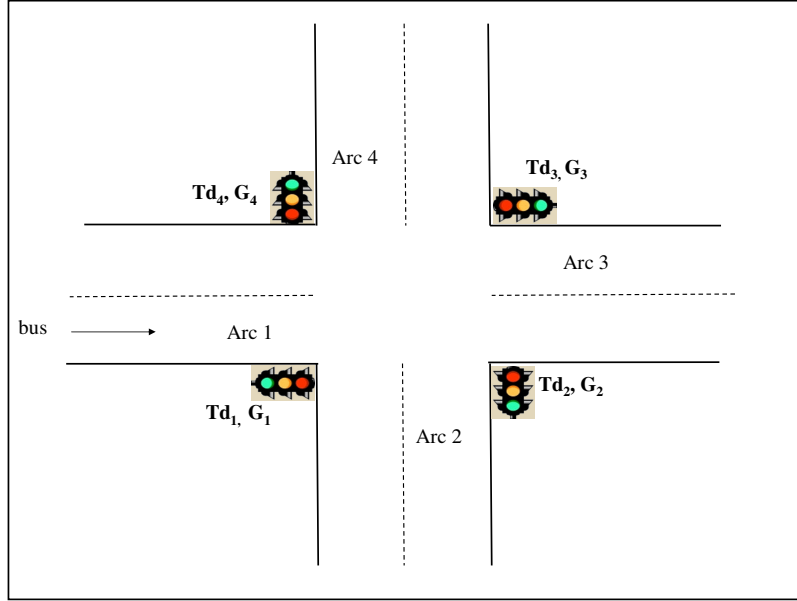


FIGURE 5.3 – Carrefour simple à deux phases

Il suffit pour décrire complètement ce carrefour de spécifier Td_1 et G_1 puisque nous pouvons écrire les équations suivantes :

$$\begin{cases} Td_3 = Td_1 \\ G_3 = G_1 \end{cases}$$

$$\begin{cases} Td_2 = Td_4 = (Td_1 + G_1) \text{ modulo } C \\ G_2 = G_4 = C - G_1 \end{cases}$$

Pour déterminer une valeur fixe pour G_1 , le principe consiste à partager la durée du cycle C entre deux flux incompatibles : d'une part les flux des arcs 1 et 3, d'autre part ceux des arcs 2 et 4. Ainsi, la durée de phase de vert G_1 allouée aux arcs 1 et 3 est donnée par

$$G_1 = C \times \frac{\max(X_1^{eq}; X_3^{eq})}{\max(X_1^{eq}; X_3^{eq}) + \max(X_2^{eq}; X_4^{eq})}$$

où X_i^{eq} désigne le nombre de véhicules particuliers dans l'arc " i ", présents dans un cycle, dans un réseau en équilibre. Pour déterminer X_i^{eq} pour chaque arc " i ", nous utilisons la procédure suivante :

1. Aux entrées du réseau, les flux d'entrée sont fixés aux valeurs nominales estimées.

2. Nous supposons qu'aucun feu de signalisation ne contrôle les carrefours : l'accès est totalement libre pour tous les carrefours.
3. Nous lançons les simulations (selon le modèle des véhicules particuliers du chapitre 2).
4. Un régime d'équilibre s'installe au bout de 4 ou 5 cycles (le nombre de véhicules ne varie plus) et on relève les valeurs X_i^{eq} pour chaque arc " i ".

Pour les instants de début des phases de vert, ils sont fixés seulement pour les carrefours traversés par les lignes de transport en commun. Le principe est simple, pour une ligne " j " qui entre dans le réseau au début d'un cycle et pour un carrefour " n ", nous calculons la distance d_n entre ce carrefour et l'entrée. Compte tenu que le véhicule est supposé rouler avec sa vitesse libre V_b , le véhicule est censé arriver au carrefour à l'instant t_n après le début d'un cycle donné. Cet instant t_n est donné par

$$t_n = \frac{d_n}{V_b} \text{ modulo } C$$

Dans notre cas, les arcs qui serviront de références pour les paramètres des carrefours sont les arcs horizontaux. Si le véhicule de transport en commun arrive au carrefour n par un arc horizontal, alors l'instant de début de la phase de vert sera pris égal à

$$t_n$$

Dans le cas contraire, il sera pris égal à

$$t_n + G_n \text{ modulo } C$$

Pour ce réseau, le tableau 5.8 résume les valeurs des instants de début (pour les arcs traversés par plusieurs lignes, la ligne 1 est prioritaire) et des durées des phases pour chaque carrefour (les arcs pris comme référence pour chaque carrefour sont les arcs horizontaux, figure 5.1).

La géométrie du réseau et ses dimensions, les caractéristiques des flux de véhicules particuliers et les caractéristiques des lignes de transport en commun ont été imaginées afin de reproduire un réseau réel de grandes dimensions. La stratégie des plans de feux fixes générée va permettre de confronter la stratégie développée avec les plans de feux fixes les plus performants. Ceci permettra de bien juger les performances de la stratégie et des gains souhaités.

5.3 Étude de sensibilité par rapport aux paramètres de l'algorithme d'optimisation

L'algorithme d'optimisation par essais particuliers est une méthode méta-heuristique stochastique. L'évolution de la trajectoire de la population dépend de plusieurs paramètres propres aux mécanismes de l'algorithme. La version originale de l'algorithme fait intervenir deux paramètres qui sont le coefficient de constriction ω et les coefficients $c_1 = c_2$. Dans la version modifiée *GCPSO*, un troisième paramètre intervient qui est le rayon r . Nous avons expliqué dans le chapitre 4 qu'une inéquation permettait de relier

Carrefour	Durée (s)	Instant de début (s)
1	48	–
2	31	–
3	42	63
4	36	23
5	63	40
6	39	77
7	44	48
8	42	–
9	45	25
10	58	57
11	38	68
12	44	16
13	10	43
14	53	–
15	39	–
16	33	10

TABLE 5.8 – Instants de début et durées des phases de vert de référence pour la stratégie à plans de feux fixes

ces paramètres pour garantir la convergence de l'algorithme et qu'en dehors de cette inégalité, l'algorithme peut avoir un comportement quelconque. En outre, une combinaison empirique ($\omega = \frac{1}{2 \times \ln(2)} \approx 0.75$ et $c_1 = c_2 = 0.5 + \ln(2) \approx 1.19$) a été établie qui permet d'assurer les meilleures performances de l'algorithme dans la résolution de plusieurs problèmes. Cependant, dans d'autres cas, les valeurs optimales de ces paramètres peuvent varier avec la nature du problème.

D'un autre côté, afin de pouvoir arrêter l'algorithme après un certain temps, nous utilisons un critère d'arrêt qui est le nombre maximale d'évaluations des fonctions objectif. Afin de garantir une convergence dans un temps raisonnable (≈ 50 secondes), ce paramètre est fixé à 4000 évaluations. Ce paramètre est défini comme le produit du nombre de particules de l'essaim par le nombre d'itérations. Ainsi pour obtenir 4000 évaluations, nous pouvons prendre un essaim de 40 particules et un nombre maximal d'itérations de 100 ou prendre un essaim de 5 particules avec 800 itérations. La qualité des résultats obtenus peut ne pas être la même.

Étant donné tous ces éléments, nous considérons deux types d'étude de sensibilité :

1. Étude par rapport aux paramètres ω et c pour la version originale et r pour la version modifiée.
2. Étude par rapport au nombre de particules et nombre maximal d'itérations autorisé. Leur produit est

fixé à 4000 (nombre d'évaluations des fonctions objectif).

5.3.1 Sensibilité par rapport aux paramètres

Dans un premier temps, nous nous intéressons à l'algorithme *PSO* original et ses deux paramètres : ω et $c = c_1 = c_2$. Dans un second temps, nous nous focalisons sur l'algorithme modifié *GCP SO* et le paramètre supplémentaire r en plus de ω et c .

5.3.1.1 Algorithme *PSO* original

Dans la version originale de l'algorithme, la trajectoire des individus de l'essaim dépend de deux paramètres qui sont :

- Le coefficient de constriction ω : des grandes valeurs favorisent l'exploration globale de l'espace de recherche au dépend de l'exploitation locale autour de solutions potentielles.
- Le coefficient c : il permet de jouer sur la tendance de la particule à revenir sur ces pas et à suivre la meilleure particule de l'essaim. Une valeur importante permet d'accélérer la convergence mais ne garantit pas la convergence vers un optimum global.

5.3.1.1.a Sensibilité par rapport à ω

Empiriquement, il a été établi que les valeurs de $\omega = \frac{1}{2 \times \ln(2)} \approx 0.75$ et $c = 0.5 + \ln(2) \approx 1.19$ étaient les plus optimales pour traiter une large gamme de problèmes. Dans cette étude, nous fixons la valeur de c à $0.5 + \ln(2) \approx 1.19$ et nous faisons varier la valeur de ω de 0.25 à 1.25 avec un pas de 0.05.

Les simulations sont effectuées sur 40 cycles pour chaque valeur de ω . Pour chaque valeur ω_i de ω et pour chaque cycle j , nous nous intéressons aux critères suivants :

- $C_VP(\omega_i, j) = \sum_{i=1}^{N_a} X_i^2(j)$ où $X_i(j)$ désigne le nombre de véhicules particuliers présents à l'arc i durant le cycle j et N_a désigne le nombre d'arcs de tout le réseau.
- $C_TC(\omega_i, j) = \sum_{i=1}^{n_b(j)} (P_i(j) - P_i^{ref}(j))^2$ où $P_i(j)$ et $P_i^{ref}(j)$ désignent respectivement la position réelle et la position référence du bus i à la fin du cycle j et $n_b(j)$ le nombre de bus présents sur le réseau au cycle j .

Les figures 5.4 et 5.5 montrent les valeurs des deux critères pour les 21 valeurs de ω et pour les 40 cycles.

[H]

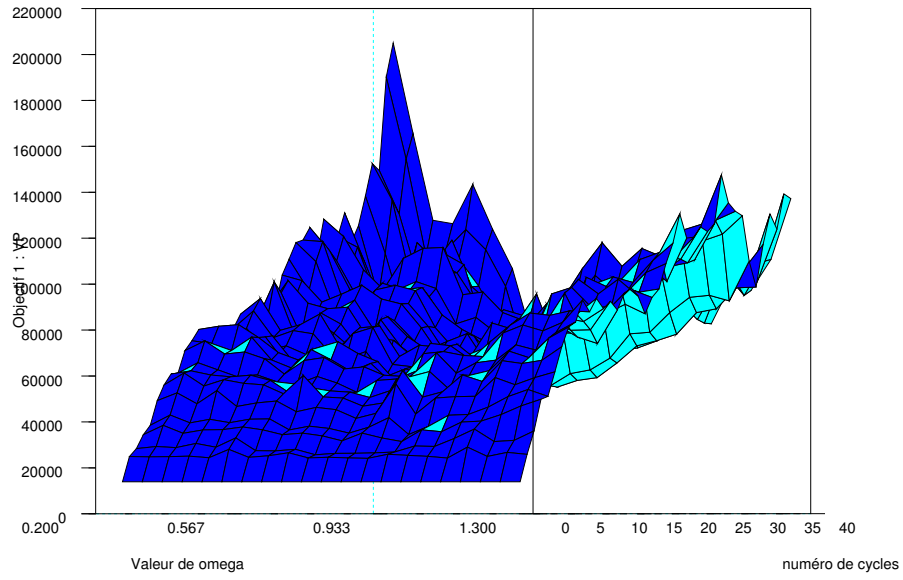


FIGURE 5.4 – Critère C_{VP}

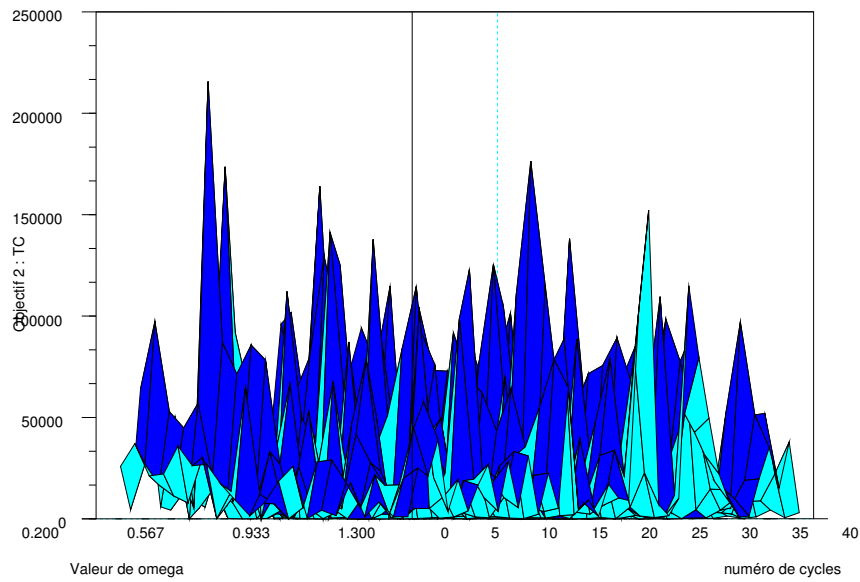


FIGURE 5.5 – Critère C_{TC}

Sur la figure 5.4, nous remarquons un creux sur la courbe pour les valeurs 0.75 et 0.80 de ω . Ce même creux n'est pas perceptible sur la figure 5.5 pour le critère des transports en commun. Pour cela, au lieu

de visualiser les deux courbes en 3 dimensions, nous traçons les résultats en 2 dimensions en effectuant une moyenne sur les 40 cycles. Ces résultats sont visualisés sur la figure 5.6.

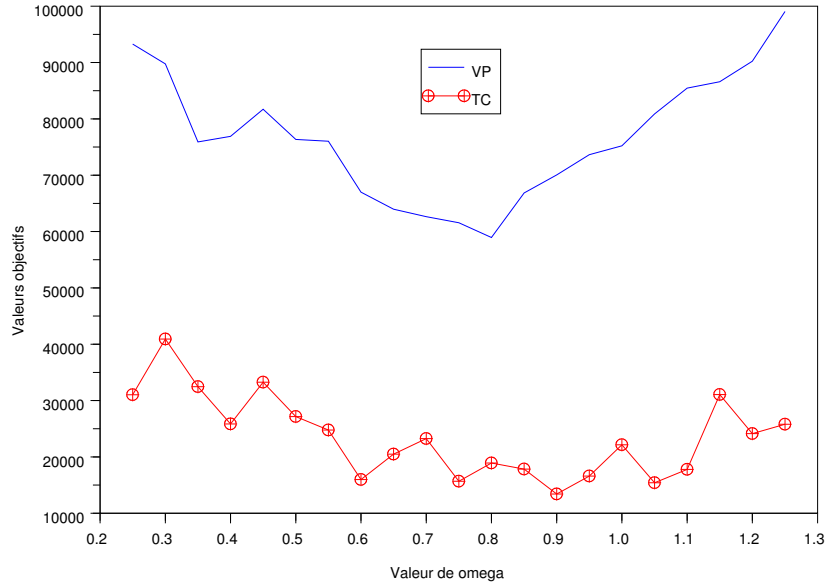


FIGURE 5.6 – Moyenne des critères C_{VP} et C_{TC}

Pour les véhicules particuliers, la plage de ω qui s'étend de 0.6 à 0.85 permet de minimiser le critère considéré. Pour les transports en commun, les valeurs minimales sont obtenues pour la plage allant de 0.75 à 0.95. Ainsi afin d'avoir un compromis entre les deux critères, la valeur de ω peut être choisie dans la plage $[0.75, 0.85] = [0.75, 0.95] \cap [0.60, 0.85]$. En regardant, les courbes de la figure 5.6, la valeur 0.75 semble offrir un meilleur compromis entre les deux objectifs.

5.3.1.1.b Sensibilité par rapport à c

L'étude par rapport au paramètre ω a confirmé que la valeur $\omega = 0.75$ est la plus appropriée pour notre cas d'étude. Empiriquement, la valeur de c qui permet d'obtenir les meilleures performances dans plusieurs problèmes d'optimisation se situe autour de 1.2. Pour vérifier ce fait, nous effectuons des simulations avec c prenant les valeurs de l'ensemble $\{0.0, 0.1, 0.2, 0.4, 0.8, 0.9, 1.0, 1.1, 1.2, 1.3, 1.5\}$. Les simulations sont effectuées sur 24 cycles et nous nous intéressons aux mêmes critères. Les figures 5.7 et 5.8 représentent respectivement, en 3 dimensions, les critères VP et TC en fonction de l'évolution des cycles et des différentes valeurs de c .

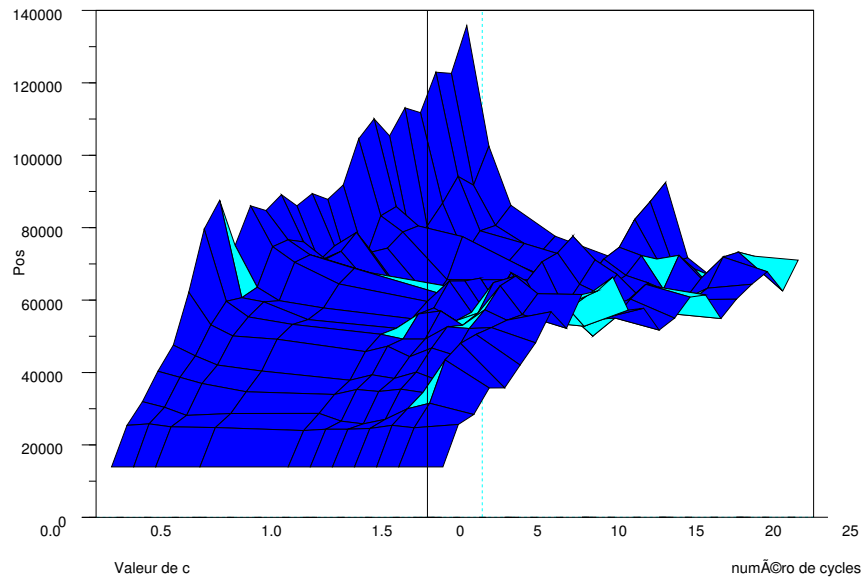


FIGURE 5.7 – Evolution du critères C_{VP}

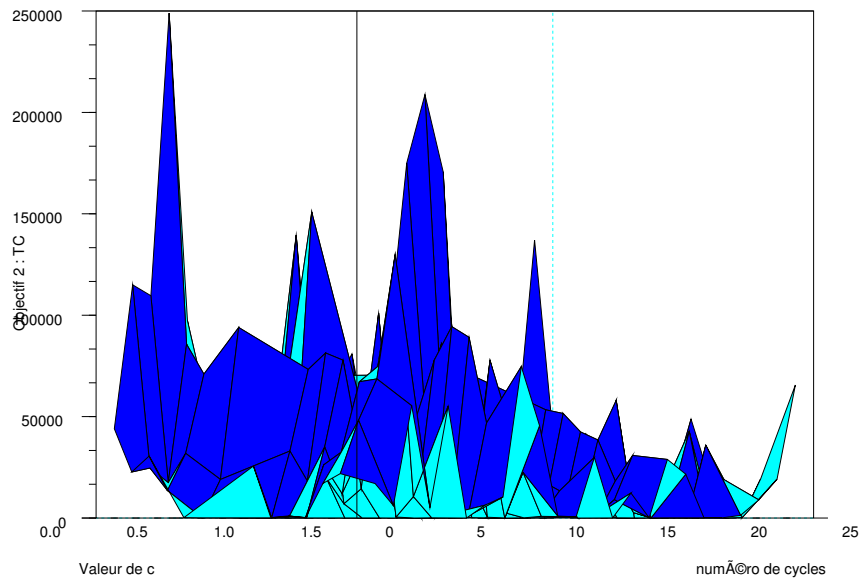


FIGURE 5.8 – Evolution du critères C_{TC}

Dans les deux courbes, nous observons que pour $c = 0$, les performances de la stratégie sont médiocres. Ceci est prévisible, puisque pour $c = 0$, l'algorithme d'optimisation perd toutes ces caractéristiques de convergence et se résume à une exploration totalement aléatoire de l'espace de recherche. Sur la courbe de la figure 5.7,

le critère décroît avec les valeurs croissantes de c jusqu'à la valeur $c = 1.2$. Le même constat se vérifie sur le critère des transports en commun représenté sur la figure 5.8. Schématiquement, ce constat se vérifie plus clairement en effectuant une moyenne, sur les 24 cycles, des deux critères (figure 5.9).

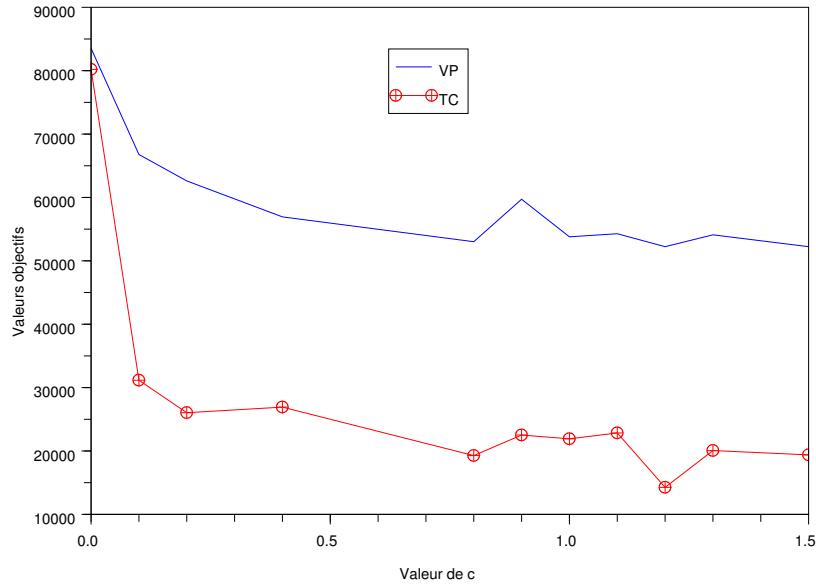


FIGURE 5.9 – Moyenne des critères C_{VP} et C_{TC}

Les courbes de la figure 5.9 montrent que les deux critères considérés atteignent leurs minimums pour $c = 1.2$ et que pour $c = 0$, les performances de l'algorithme sont très mauvaises.

Les deux études sur les deux paramètres ω et c montrent que les meilleures performances de la stratégie sont obtenues pour $\omega = 0.75$ et $c = 1.2$ confirmant ainsi les valeurs empiriques déterminées pour une large classe de problèmes. Dans la section suivante, nous nous intéressons aux paramètres de la version modifiée *GCPSO* de l'algorithme.

5.3.1.2 Algorithme *GCPSO*

L'algorithme modifié *GCPSO* fait intervenir trois paramètres : les deux paramètres ω et c et le rayon r . Le rayon r définit le périmètre autour duquel une turbulence sur la position de la meilleure particule devra se situer. Le fait d'introduire une turbulence permet d'éviter une convergence prématurée de la version originale de l'algorithme.

5.3.1.2.a Sensibilité par rapport à ω

Comme pour l'algorithme *PSO*, nous effectuons une étude de sensibilité par rapport au paramètre ω qui prend des valeurs entre 0.25 et 1.25. La figure 5.10 montre les mêmes critères que dans la section précédente.

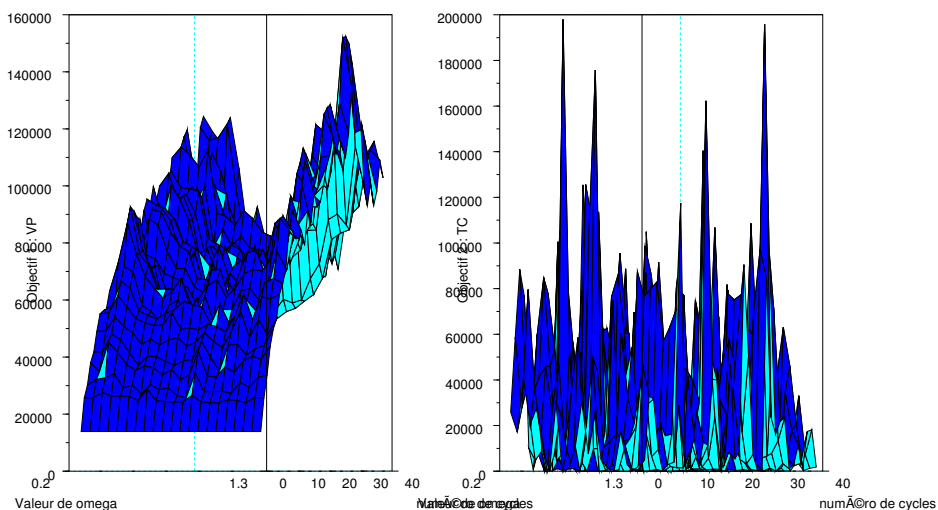


FIGURE 5.10 – Evolution du critères C_{VP} et C_{TC}

Sur la première courbe, un creux est bien perceptible et se situe entre les valeurs 0.5 et 1.0 de ω . Ce creux n'est pas très clair sur la deuxième courbe relative au critère des transports en commun. Pour bien apercevoir l'influence de ω , nous traçons les mêmes critères, en deux dimensions, et en moyennant sur les cycles. La figure 5.11 montre ces résultats.

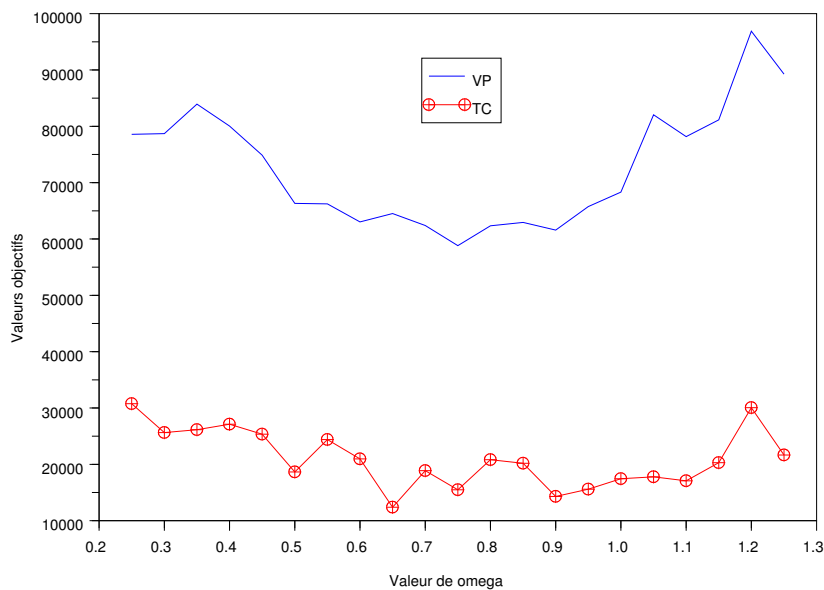


FIGURE 5.11 – Moyenne des critères C_{VP} et C_{TC}

Avec ces nouvelles courbes, il est très clair que la stratégie a de meilleures performances pour ω variant entre 0.6 et 0.9. Le critère VP est minimal pour $\omega = 0.75$ et le critère TC pour $\omega = 0.65$. Cependant la valeur $\omega = 0.75$ offre un meilleur compromis.

5.3.1.2.b Sensibilité par rapport à c

Sur la figure 5.12 sont illustrés les deux critères VP et TC. Sur la figure 5.13 sont illustrées les moyennes sur les cycles des deux critères. Comme pour la version *PSO* originale, les critères décroissent pour c allant de 0 jusqu'à 1.1. La valeur $c = 1.1$ permet d'obtenir les meilleurs performances.

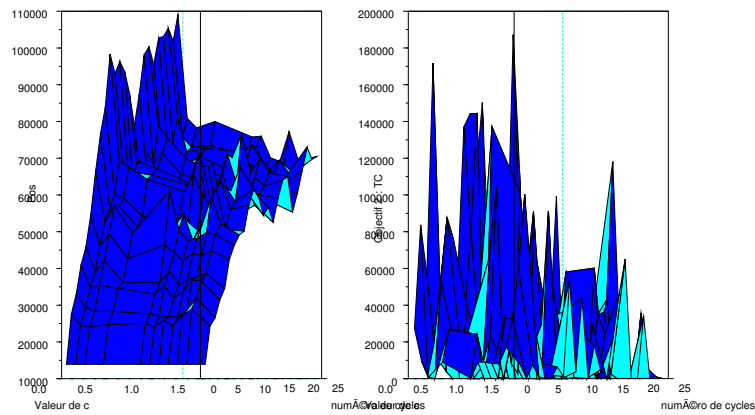


FIGURE 5.12 – Evolution du critères C_{VP} et C_{TC}

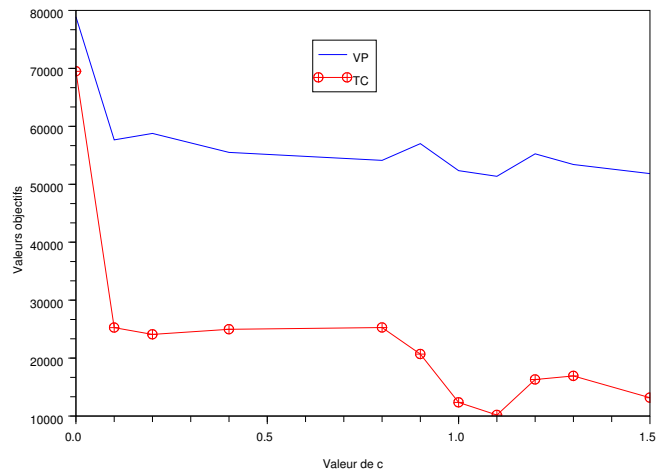


FIGURE 5.13 – Moyenne des critères C_{VP} et C_{TC}

5.3.1.2.c Sensibilité par rapport à r

Le rayon r est un paramètre important dans la convergence de l'algorithme *GCPSO*. Il permet de définir le périmètre autorisé pour l'introduction d'une turbulence à la position de la meilleure particule de l'essaim. Comme pour les paramètres ω et c , la même procédure est appliquée. Les simulations sont effectuées sur 40 cycles et pour r prenant les valeurs de l'ensemble $\{0.5, 1, 2, 6, 10, 15, 20\}$.

La figure 5.14 montre l'évolution des deux critères pour les différentes valeurs de r et pour les 40 cycles. La figure 5.15 permet de montrer la moyenne des deux critères effectuée sur les cycles. Pour $r = 6$, les performances de l'algorithme sont les meilleures pour les deux modes de transport. Cependant, les courbes ne permettent pas de dégager une tendance générale de l'algorithme par rapport au paramètre r . La valeur de $r = 6$ sera utilisée pour les prochaines simulations.

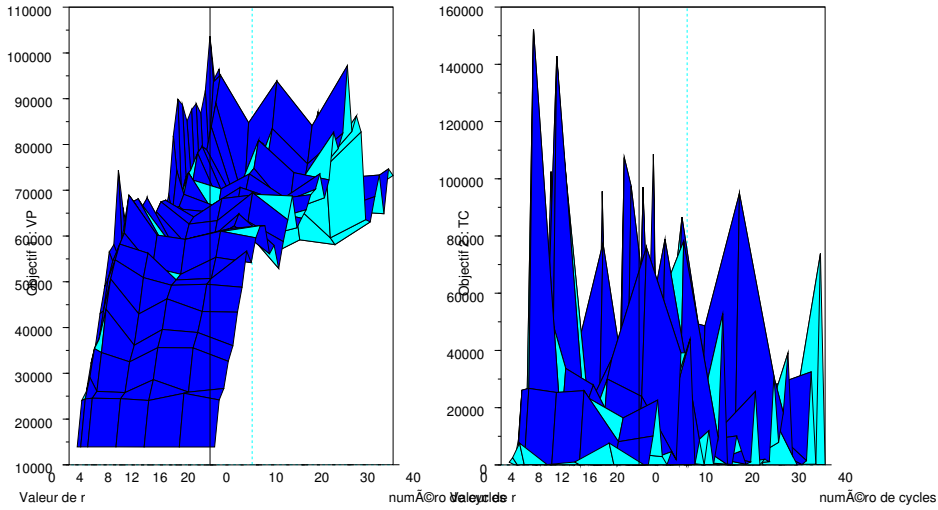


FIGURE 5.14 – Evolution du critères C_{VP} et C_{TC}

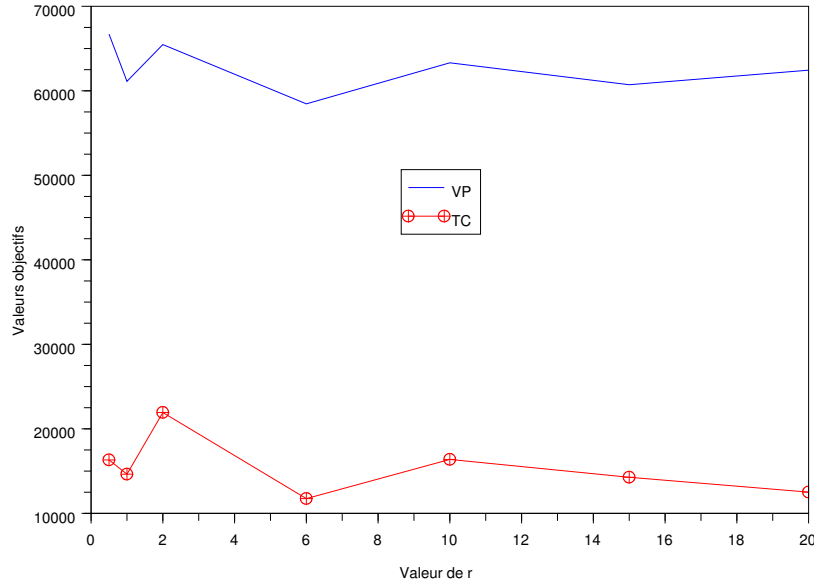


FIGURE 5.15 – Moyenne des critères C_{VP} et C_{TC}

5.3.2 Sensibilité par rapport au nombre de particules et nombre maximal d'itérations

Le nombre de particules de l'essaim jouent un rôle très important dans la convergence de l'algorithme. Il détermine aussi la rapidité de convergence. Il n'existe pas de règle générale pour le choix du nombre de particules mais un nombre important permet de mieux couvrir l'espace de recherche mais résulte en l'explosion des temps de calcul. Un compromis est donc nécessaire pour une meilleure convergence avec des temps de calcul raisonnable. Dans la plupart des études sur l'optimisation par méta-heuristiques, l'algorithme est arrêté quand on aura atteint un nombre maximum d'évaluations des fonctions objectifs. Dans cette section, nous limiterons le nombre maximale des évaluations à 4000. Pour un essaim contenant n_p particules, le nombre d'itérations maximal autorisé est égal à la partie entière de $\frac{4000}{n_p}$. Pour les deux versions de l'algorithme, le nombre de particules n_p prend ses valeurs dans l'ensemble $\{2, 5, 10, 20, 30, 40, 80, 100, 200\}$.

5.3.2.1 Algorithme *PSO* originale

Comme pour les sections précédentes, les simulations sont effectuées sur 40 cycles. Les critères considérés pour mesurer les performances de la stratégie sont aussi les mêmes. La figure 5.16 montre, en 3 dimensions, les critères VP et TC, pour les différentes valeurs de n_p .

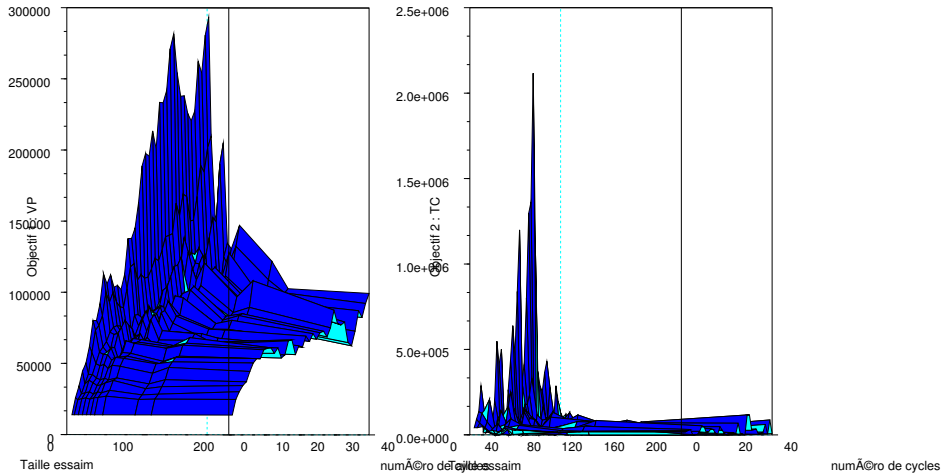


FIGURE 5.16 – Evolution du critères C_{VP} et C_{TC}

La figure 5.17 montre la moyenne, effectuée sur les cycles, des deux critères ainsi que les temps d'exécution de la stratégie pour 40 cycles de simulation.

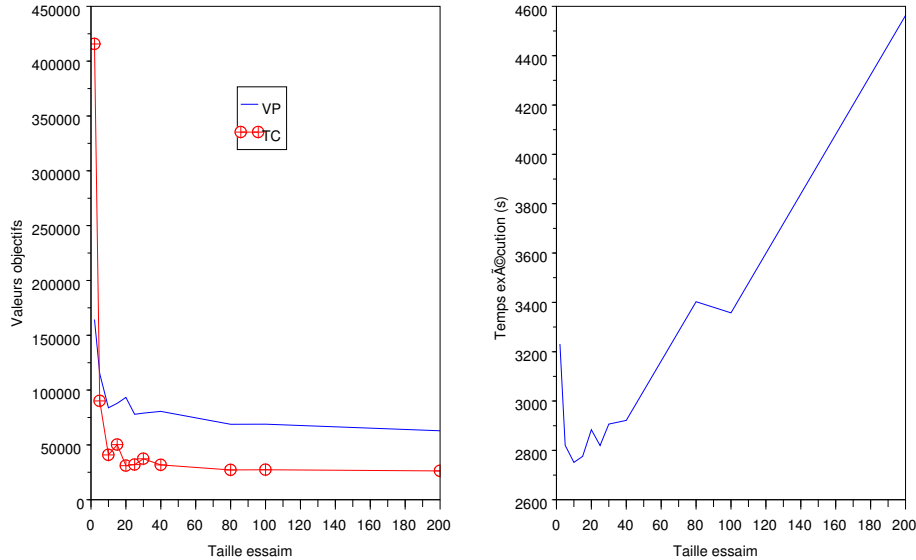


FIGURE 5.17 – Moyenne du critères C_{VP} et C_{TC} et temps d'exécution

Les courbes montrent clairement que les valeurs des critères décroissent avec le nombre de particules (pour un nombre d'évaluations constant). A partir de $n_p = 20$, le rythme de décroissance ralentit beaucoup. A partir de 80 particules, les critères ne bougent pratiquement plus. A l'inverse, le temps d'exécution croît avec n_p . Les temps de calcul peuvent croître même de 160% pour $n_p = 200$ par rapport à $n_p = 10, 15$ ou 20.

Un nombre croissant de particules évite à la version *PSO* originale de converger prématurément vers un optimum de mauvaise qualité. Pour la version originale, le nombre de particules est plus important que le nombre d'itérations étant donné qu'avec un faible nombre de particules, l'algorithme converge au bout de quelques dizaines d'itérations rendant inutile d'augmenter le nombre d'itérations. Par contre, augmenter le nombre de particules résulte en des temps de calcul plus importants. La valeur de 20 ou 30 particules offre un compromis entre convergence et rapidité.

5.3.2.2 Algorithme *GCPSO* modifié

L'algorithme *GCPSO* modifié a pour objectif d'éviter à la version originale de converger prématurément au bout de quelques dizaines d'itérations. Il est donc prévu que l'algorithme *GCPSO* soit moins sensible au nombre de particules de l'essaim.

La figure 5.18 montre l'évolution des deux critères *VP* et *TC* sur 40 cycles pour les différentes valeurs de n_p . La figure 5.19 montre les moyennes des deux critères et les temps d'exécution pour chaque valeur de n_p .

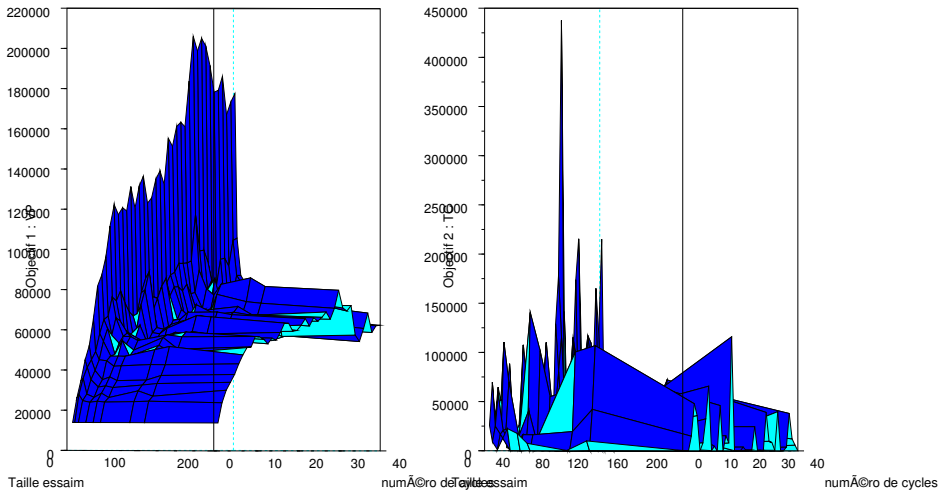


FIGURE 5.18 – Evolution du critères C_{VP} et C_{TC}

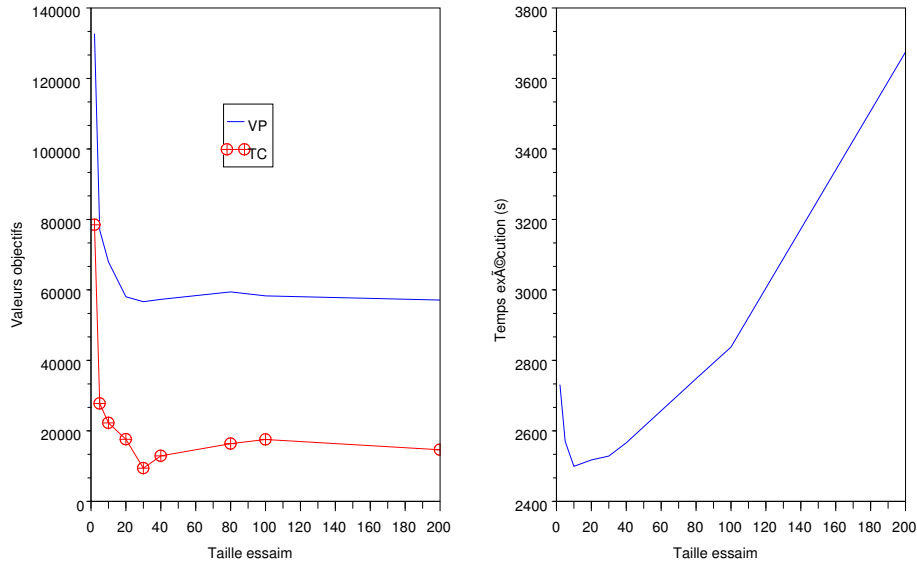


FIGURE 5.19 – Moyenne du critères C_{VP} et C_{TC} et temps d'exécution

Comme l'algorithme *PSO*, un petit nombre de particules n'est pas efficace. Par contre, pour l'algorithme *GCPSO*, les critères ont un minimum pour $n_p = 30$. A partir de 30 particules, les critères varient peu. Les temps de calcul croissent avec le nombre de particules comme l'algorithme *PSO*. Cependant, les temps d'exécution pour *GCPSO* sont inférieurs à ceux enregistrés pour l'algorithme *PSO*. La figure 5.20 illustre les temps d'exécution des deux algorithmes pour différentes valeurs de n_p .

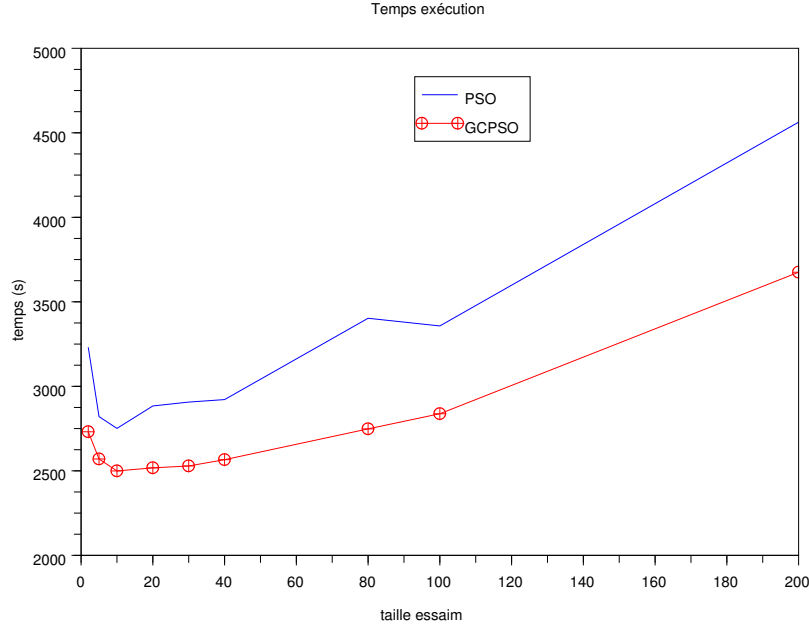


FIGURE 5.20 – Temps d'exécution pour *PSO* et *GCPSO* en fonction de la taille de l'essaim

Il est très clair que l'algorithme *GCPSO* est beaucoup plus rapide que l'algorithme *PSO*. Ceci s'explique par le fait que l'algorithme *GCPSO* fait une recherche plus efficace dans l'espace de recherche permettant ainsi de minimiser le nombre de solutions non dominées dans l'archive externe.

5.4 Sensibilité par rapport aux horizons de simulation et de contrôle

Le principe de la commande prédictive repose sur l'optimisation des critères sur un horizon temporel plus long qu'un seul cycle. Cet horizon est l'horizon de simulation. Ceci permet de tenir compte des répercussions des commandes, pour le seul cycle suivant, sur les cycles futurs. La commande prédictive repose aussi sur un autre type d'horizon appelé horizon de contrôle. Au lieu de déterminer les commandes seulement pour le cycle prochain, on détermine les commandes pour plusieurs cycles futurs. Ceci autorisera davantage de souplesse et permettra d'atténuer les répercussions éventuelles de la commande sur les cycles futurs. Dans cette section, nous nous intéressons aux effets des longueurs de l'horizon de simulation et de contrôle sur les performances de la stratégie pour les algorithmes *PSO* et *GCPSO*.

Par définition, la longueur de l'horizon de simulation H_s est supérieure à la longueur de l'horizon de contrôle H_c . Les longueurs sont mesurées en nombre de cycles. Pour cette étude, nous considérons les valeurs suivantes :

- L'horizon de contrôle H_c prend les valeurs 1, 2 et 4 ($H_c \in \{1, 2, 4\}$).
- Pour chaque valeur H_c , trois valeurs de l'horizon de simulation H_s sont considérées : H_c , $H_c + 2$ et $H_c + 4$.

Par conséquent, nous avons 9 scénarii. Le premier avec $H_s = H_c = 1$ et le dernier avec $H_c = 4$ et $H_s = H_c + 4 = 8$. Comme dans les sections précédentes, nous considérons deux critères : le premier pour les véhicules particuliers (VP) et le second pour les véhicules de transport en commun (TC). Le critère VP reste le même et se présente sous la forme de $\sum_{i=1}^{N_a} X_i^2(k)$ pour le cycle k . Ce critère est tracé pour chaque cycle, la simulation est effectuée sur 24 cycles (≈ 32 minutes car le cycle dure 80 secondes). Pour les véhicules TC, nous considérons un nouveau critère plus simple à mettre en oeuvre pour cette étude. A chaque cycle et pour chaque véhicule TC, la stratégie a pour objectif de faire respecter au véhicules une position consigne fixé à l'avance. Vu la géométrie du réseau, de sa longueur et des caractéristiques de vitesse des véhicules, chaque bus ne doit, en théorie, passer plus de 4 cycles dans le réseau. Pour ces 4 cycles, les consignes sont les mêmes pour toutes les lignes et sont : être à 350 mètres à la fin du premier cycle après l'entrée au réseau, à 700 au second cycle, à 1050 au troisième et à 1400 au quatrième. Si P_i (respectivement P_i^{ref}) désigne la position réelle (respectivement position consigne) du véhicule i , le critère TC considéré ici s'écrit sous la forme

$$\sum_{i=1}^{n_b} (P_i(k_i) - P_i^{ref}(k_i))^2$$

n_b désigne le nombre de véhicules TC qui ont traversé le réseau pendant la période de simulation (24 cycles) et k_i désigne le numéro de cycle après l'entrée au réseau du véhicule i ($k \in \{1, 2, 3, 4\}$).

Nous distinguons deux études : l'une se focalise sur l'horizon de simulation pour un horizon de contrôle fixé et l'autre sur l'horizon de contrôle.

5.4.0.3 Sensibilité par rapport à l'horizon de simulation

Pour chaque longueur de l'horizon de contrôle $H_c = 1, 2, 4$, trois longueurs de l'horizon de simulation sont considérées : $H_s = H_c, H_c + 2, H_c + 4$. Les figures 5.21 et 5.22 tracent les valeurs des objectifs pour les différentes valeurs de H_c et de H_s pour respectivement l'algorithme original *PSO* et l'algorithme modifié *GCPSO*.

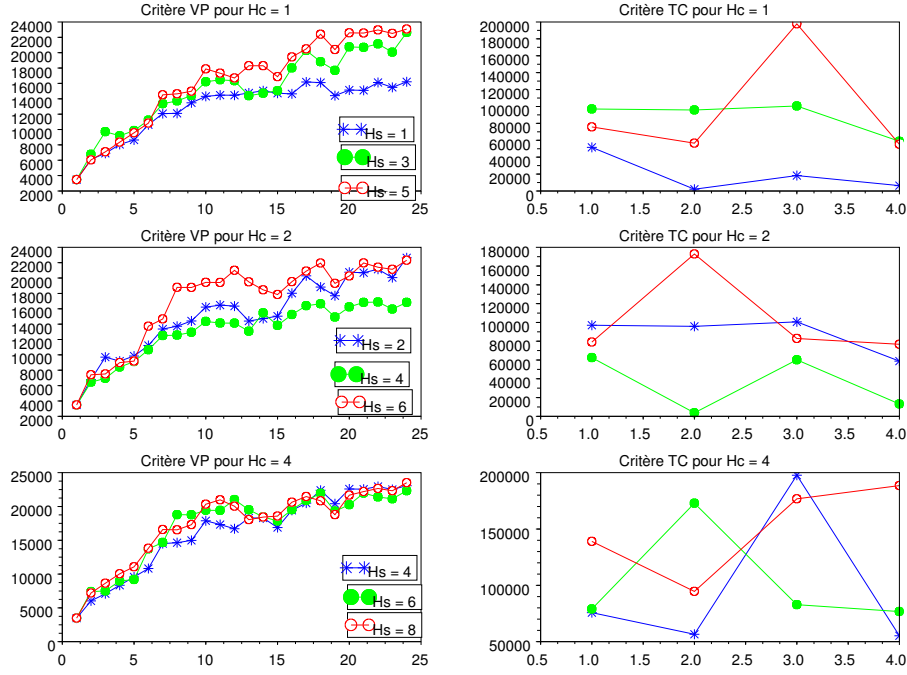


FIGURE 5.21 – Critères VP et TC pour différents horizons de simulation pour l’algorithme PSO

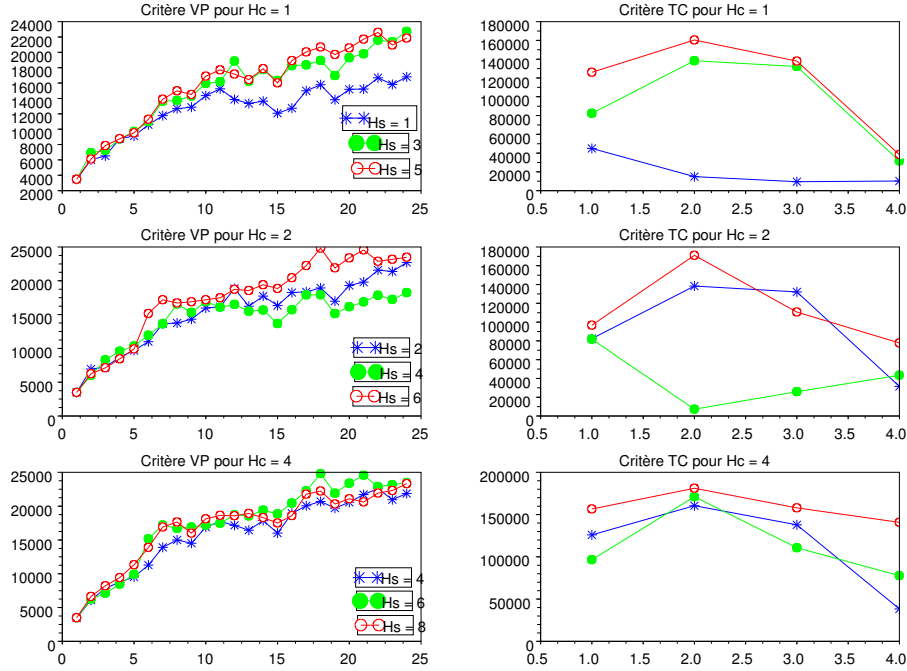


FIGURE 5.22 – Critères VP et TC pour différents horizons de simulation pour l’algorithme GCP SO

Pour $H_c = 1$ et pour les deux algorithmes *PSO* et *GCP SO*, les performances de la stratégie se dégradent avec la longueur des horizons de simulation. Pour $H_c = 2$, ce n’est pas la même tendance. Les performances pour les deux critères VP et TC sont meilleures pour $H_s = 4$. Viennent ensuite les performances pour $H_s = 2$ et

$H_s = 6$. Pour $H_c = 4$, les trois performances pour les trois valeurs de H_s sont sensiblement les mêmes.

Pour chaque valeur de H_c fixé, nous avons considéré trois valeurs de H_s qui sont : $H_s = H_c$, $H_s = H_c + 2$ et $H_s = H_c + 4$. Pour pouvoir calculer les critères sur un horizon $H_s \neq 1$, il est nécessaire de disposer d'estimations pour les entrées exogènes (valeurs des flux de véhicules aux entrée du réseau). Or ces estimations ne sont jamais fiables surtout pour les phénomènes de trafic. Elles sont donc biaisées et le biais est d'autant plus grand que l'horizon de simulation est plus long. L'intérêt de considérer un horizon de simulation plus long que l'horizon de contrôle se vérifie seulement pour le cas $H_c = 2$. Cependant les meilleures performances pour tous H_c et tous H_s sont pour $H_s = H_c = 1$ où aucune estimation n'est nécessaire.

5.4.0.4 Sensibilité par rapport à l'horizon de contrôle

Ce sont exactement les mêmes résultats que dans la sous section précédente. Mais nous les présentons de manière différente afin de mettre l'accent sur l'influence du paramètre H_c pour H_s fixé. Les figures 5.23 et 5.24 montrent les résultats pour les algorithmes *PSO* et *GCPSO*.

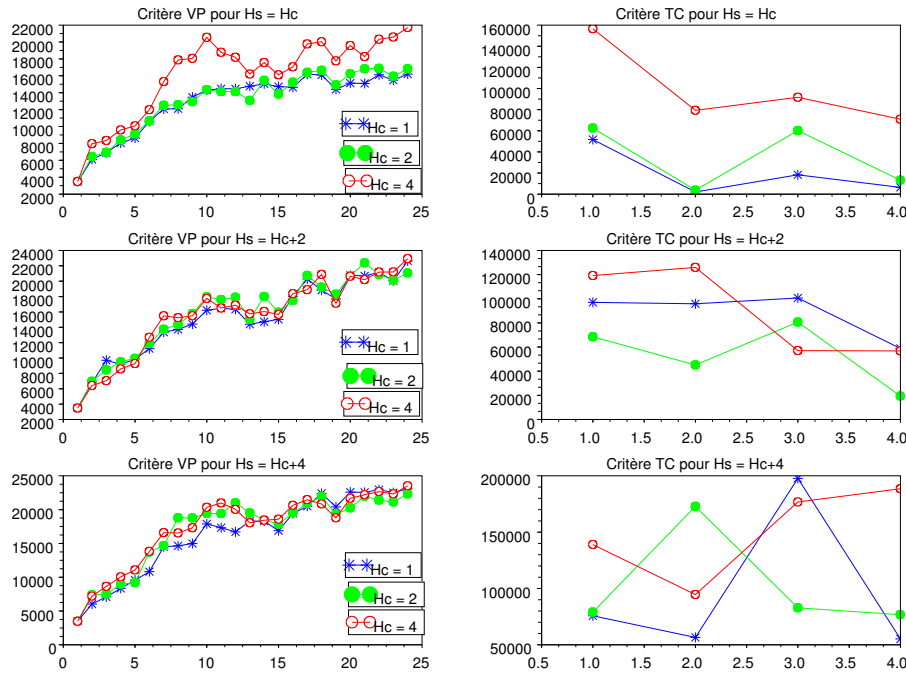


FIGURE 5.23 – Critères VP et TC pour différents horizons de contrôle pour l'algorithme PSO

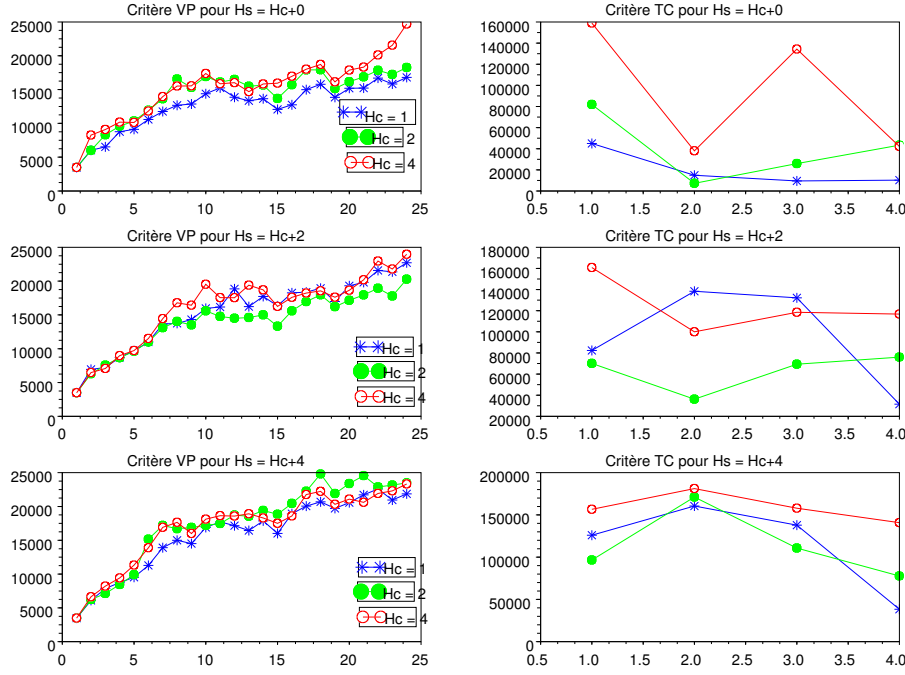


FIGURE 5.24 – Critères VP et TC pour différents horizons de contrôle pour l’algorithme GCPSO

Quand $H_s = H_c$, les performances de la stratégie pour les deux critères se dégradent avec H_c . Pour $H_s = H_c + 2$ et $H_s = H_c + 4$, les performances sont très proches pour les trois valeurs de H_c même si pour l’algorithme *GCPSO* et pour $H_s = H_c + 2$ et $H_c = 2$, la performance de la stratégie est légèrement meilleure que les autres. Comme pour la discussion précédente, les meilleures performances sont obtenues pour $H_s = H_c = 1$.

Nous observons que quand l’horizon de contrôle croit, généralement les performances sont moins bonnes. En plus des estimations biaisées sur les entrées exogènes, ceci s’explique par le fait que quand on considère un horizon de contrôle $H_c > 1$, le nombre des variables de décision du problème d’optimisation est multiplié par H_c . Pour les algorithmes méta-heuristique *PSO* et *GCPSO*, ceci engendre des calculs beaucoup plus importants et une difficulté supplémentaire pour la convergence avec un nombre d’itérations limité. Ce qui résulte en une convergence prématurée.

5.4.0.5 Conclusion

Les figures 5.25 et 5.26 résument les résultats de cette sous section pour les algorithmes *PSO* et *GCPSO*. En abscisse, sont représentés les scénarii par des entiers allant de 1 à 9. Le tableau 5.9 donne la signification de ces entiers.

Pour les critères, nous considérons une moyenne sur les cycles. Ainsi pour le critère VP, une moyenne est effectué sur les 24 cycles de simulation. Pour le critère TC, la moyenne est effectué sur 4 cycles.

Entier	1	2	3	4	5	6	7	8	9
H_c	1			2			4		
H_s	H_c	$H_c + 2$	$H_c + 4$	H_c	$H_c + 2$	$H_c + 4$	H_c	$H_c + 2$	$H_c + 4$

TABLE 5.9 – Signification des entiers des abscisses

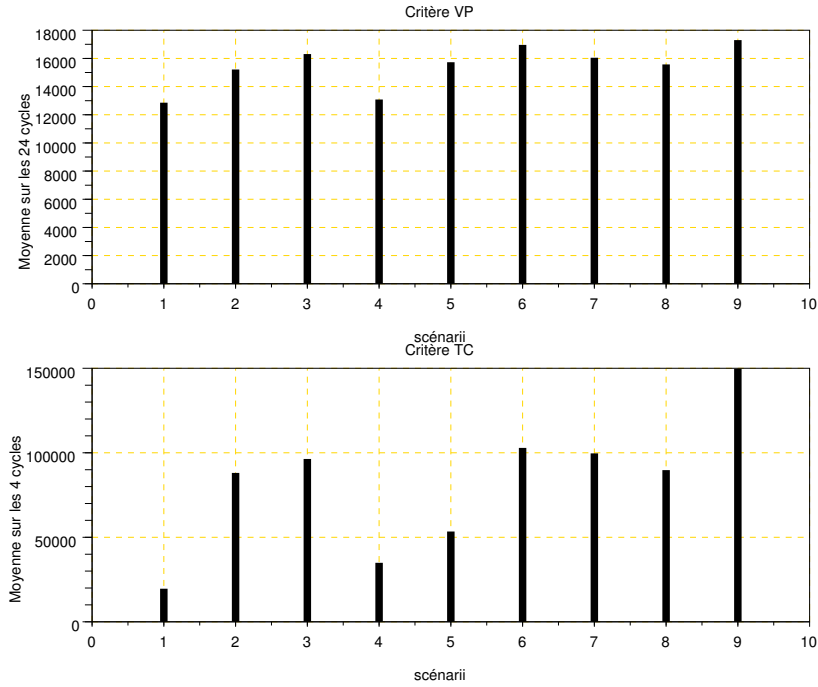


FIGURE 5.25 – Critères VP et TC pour différents horizons de contrôle pour l'algorithme PSO

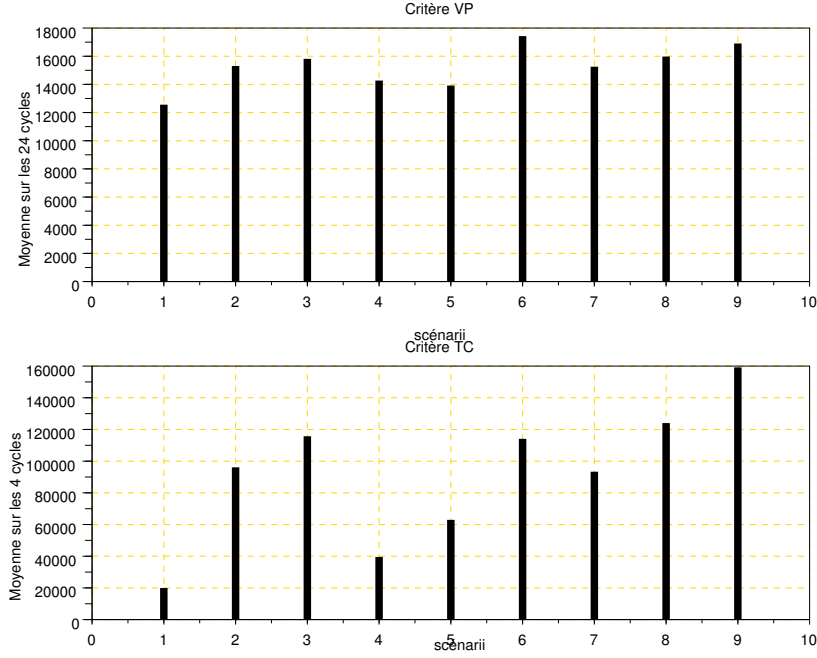


FIGURE 5.26 – Critères VP et TC pour différents horizons de contrôle pour l’algorithme GCPSO

Il est très clair que les meilleures performances sont obtenues pour $H_s = H_c = 1$ (ni horizon de contrôle ni de simulation) et ce contrairement aux principes généraux de la commande prédictive. Dans le même sens, les pires performances sont obtenues pour $H_c = 4$ et $H_s = H_c + 4 = 8$. Ceci s’explique par les deux points suivants :

1. D’abord, pour considérer des horizons de contrôle et de simulation, il est nécessaire de disposer d’estimations des entrées exogènes non contrôlables. Ces estimations sont basées sur des moyennes calculées historiquement. Or pour le trafic, les valeurs réelles s’écartent des moyennes et les perturbations sont nombreuses. Ceci résulte en des calculs tenant compte de valeurs fausses.
2. L’augmentation de l’horizon de contrôle résulte en l’augmentation du nombre des variables de décision du problème d’optimisation. Or les méthodes méta-heuristiques comme les algorithmes *PSO* et *GCPSO* sont très sensibles au nombre de variables de décision et leurs performances décroissent avec ce nombre.

5.5 Influence des objectifs de régulation

La stratégie est traduite sous la forme d’un problème d’optimisation bi-objectif. Le critère pour les véhicules particuliers est la somme quadratique du nombre de véhicules sur tous les arcs du réseau que la stratégie tend à minimiser. Le minimum théorique est facile et consiste à un nombre nul de véhicules sur tous les arcs. Cependant, il est parfois difficile d’atteindre cet objectif vu les conditions de trafic et les flux constants de véhicules. Il est donc nécessaire de reformuler le critère des véhicules particuliers (critère VP) afin de proposer

un objectif plus facile à atteindre. Pour cela, l'objectif n'est plus de minimiser le nombre de véhicules sur chaque arc mais de minimiser un écart quadratique entre le nombre réelle et un nombre fixe. L'ancien critère est formulé de la manière suivante

$$\sum_{i=1}^{N_a} X_i^2(k)$$

Le nouveau critère s'écrit de la manière suivante

$$\sum_{i=1}^{N_a} (X_i(k) - X_i^{ref})^2$$

Nous observons que l'ancien critère s'obtient à partir du nouveau en prenant $X_i^{ref} = 0$. Dans la sous section 5.2.4, nous avons évoqué un régime d'équilibre qui s'installe dans l'hypothèse où tous les accès contrôlés par le feux sont libres. Pour chaque arc i , le nombre de véhicules d'équilibre est noté X_i^{eq} . Le nouveau critère s'écrit comme suit

$$\sum_{i=1}^{N_a} (X_i(k) - p \times X_i^{eq})^2$$

p désigne un coefficient que nous allons le modifier pour vérifier quelle valeur permet d'optimiser les performances de la stratégie. Le coefficient p prend les valeurs dans l'ensemble $\{0, 0.25, 0.5, 0.75, 1\}$. Les critères de performance sont les mêmes que pour la section 5.3.

La figure 5.27 donne l'évolution des critères VP et TC pour les différentes valeurs de p pour l'algorithme *PSO*. La figure 5.28 donne une moyenne des mêmes critères.

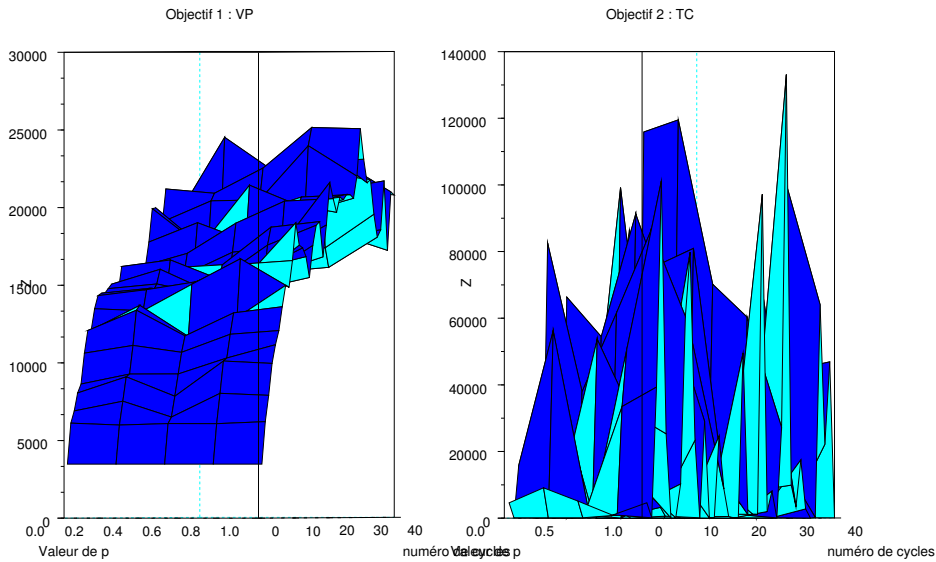


FIGURE 5.27 – Critères VP et TC pour différentes valeurs de p pour l'algorithme *PSO*

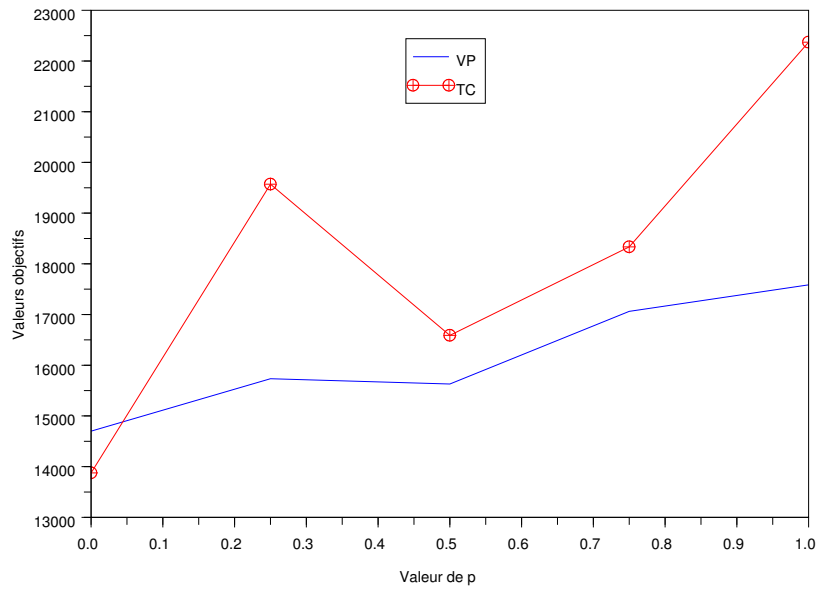


FIGURE 5.28 – Moyennes des critères VP et TC pour différentes valeurs de p pour l'algorithme *PSO*

Les résultats montrent que, pour l'algorithme original *PSO*, les meilleures performances sont obtenues pour $p = 0$ c'est à dire quand on considère le critère VP initial. Le critère TC est aussi minimal pour $p = 0$, ceci s'explique par le fait que la minimisation de la charge du trafic VP profite aussi aux véhicules de transport en commun. Les moins bonnes performances sont obtenues pour $p = 1$ c'est à dire quand les objectifs de la régulation VP sont les moins enthousiastes.

Les mêmes résultats sont donnés, pour l'algorithme *GCPSO*, sur les figures 5.29 et 5.30.

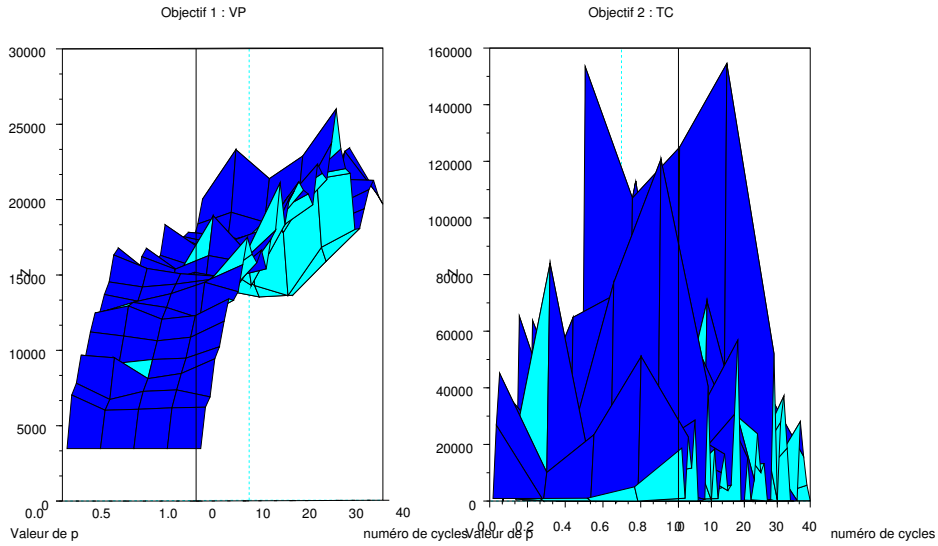


FIGURE 5.29 – Critères VP et TC pour différentes valeurs de p pour l'algorithme *GCPSO*

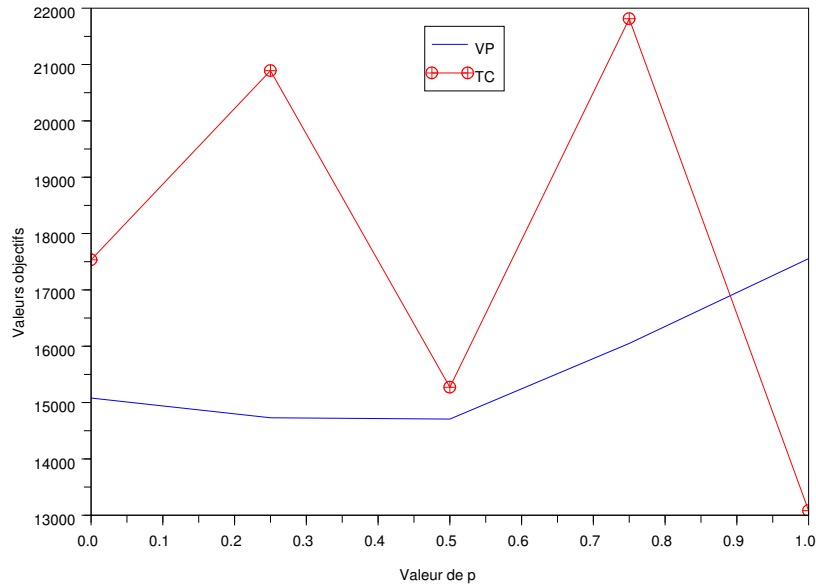


FIGURE 5.30 – Moyennes des critères VP et TC pour différentes valeurs de p pour l'algorithme *GCPSO*

Contrairement à l'algorithme *PSO*, la version modifiée *GCPSO* ne présente pas les meilleures performances pour $p = 0$. Le critère VP est minimal pour $p = 0.25$ et $p = 0.5$. Le critère TC est minimal pour $p = 1$. Les objectifs de régulation des véhicules TC sont mieux respectés quand les objectifs de régulation VP sont moins enthousiastes. Bien que la régulation VP profite dans certains cas aux véhicules TC, le fait d'être moins enthousiaste pour la régulation VP, permet au critère TC d'avoir plus de poids dans le processus

d'optimisation. Ceci ne s'est pas vérifié pour l'algorithme original *PSO* contrairement à l'algorithme *GCPSO* parce que le premier est moins efficace dans la recherche d'optimums globaux.

5.6 Analyse des performances de la stratégie

L'objectif de cette section est l'analyse et l'interprétation des résultats du point de vue du trafic. Pour les véhicules particuliers, nous nous intéressons au nombre de véhicules sur les arcs. Pour les transports en commun, nous nous intéressons à la progression des véhicules suivant les consignes de régularité fixées. Dans un premier temps, une étude comparative est effectuée selon les consignes données pour la régularité des lignes de transports en commun. Dans un second temps, nous comparons trois configurations de la stratégie entre eux et par rapport aux plans de feux fixes. Enfin, nous montrons les performances de la stratégie et son comportement sous une petite perturbation de l'état du trafic.

5.6.1 Analyse selon les consignes de régularité données aux transports en commun

L'objectif de cette étude est l'analyse du comportement de la stratégie par rapport aux consignes de régularité données aux véhicules de transport en commun. Nous rappelons que les consignes sont des positions pour chaque fin de cycle et pour chaque véhicule présent sur le réseau. Avec une longueur de cycle $C = 80$ secondes et une vitesse libre des véhicules fixée à 8 m/s , un véhicule peut parcourir au maximum la distance de $8 \times 80 = 640 \text{ m}$ en un seul cycle. Pour cette étude, nous appliquons la stratégie pour sept valeurs de la consigne. Chaque valeur correspond à une configuration, par exemple la configuration 1 correspond à une consigne de 100 m par cycle, ce qui signifie que l'objectif est qu'un véhicule doit parcourir exactement 100 m par cycle. Le tableau suivant résume les 7 configurations :

Configuration	1	2	3	4	5	6	7
Consigne	100	200	300	400	500	600	650

TABLE 5.10 – Les différentes configurations pour les consignes TC

Chaque configuration de la stratégie est comparée avec la stratégie des plans de feux fixes. Les figures 5.31 jusqu'à la figure 5.37 donne la progression moyenne des quatre lignes pour chaque configuration.

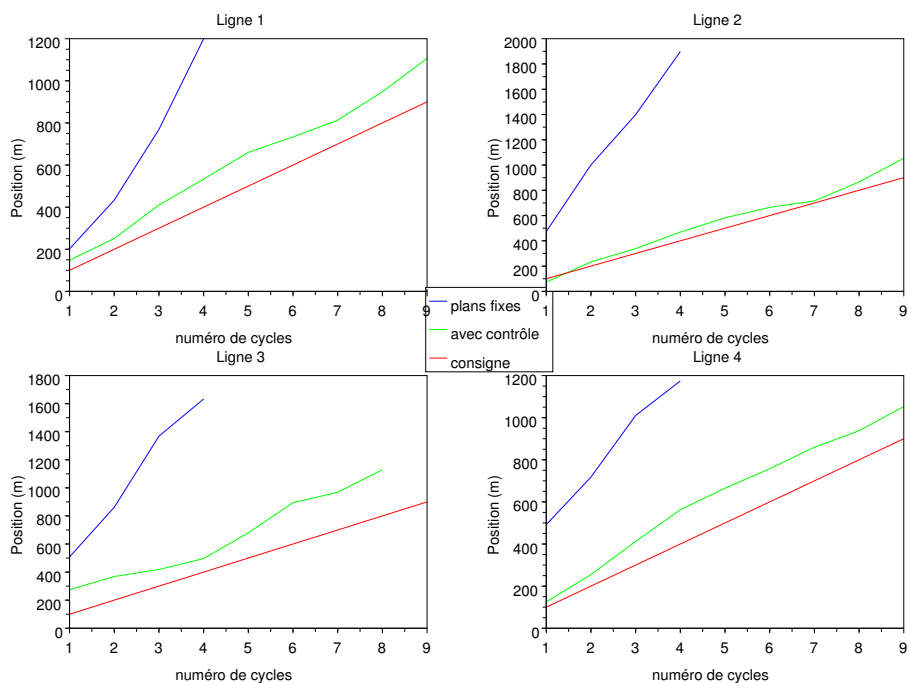


FIGURE 5.31 – Progression moyenne des 4 lignes de bus pour une consigne de 100 m

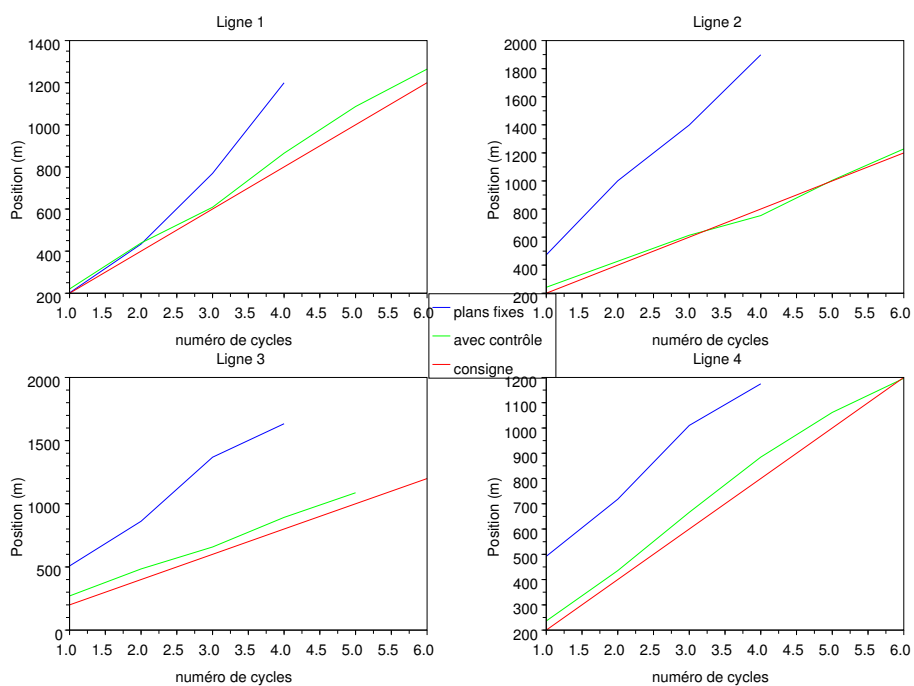


FIGURE 5.32 – Progression moyenne des 4 lignes de bus pour une consigne de 200 m

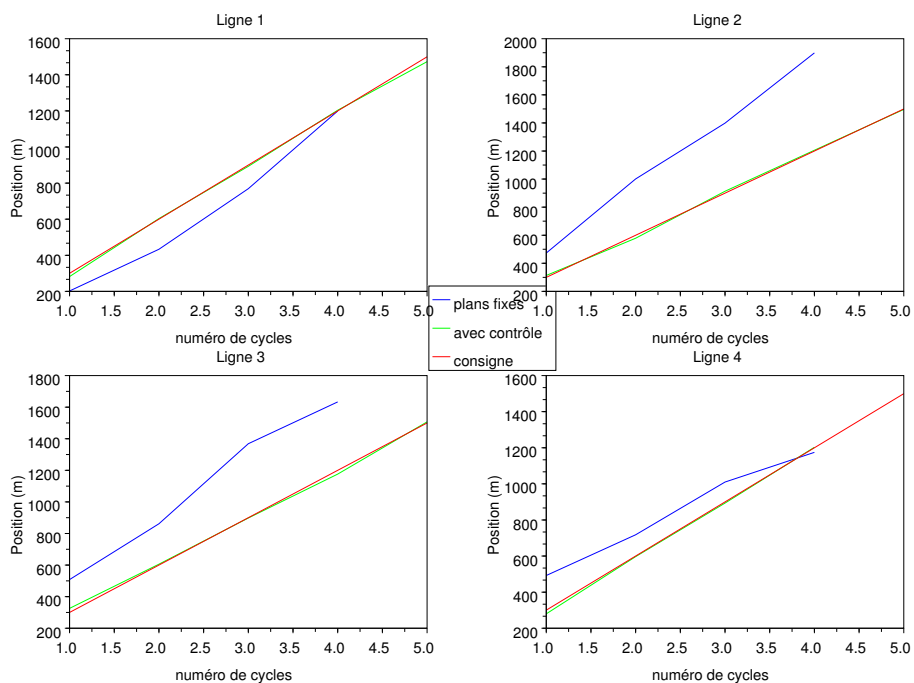


FIGURE 5.33 – Progression moyenne des 4 lignes de bus pour une consigne de 300 m

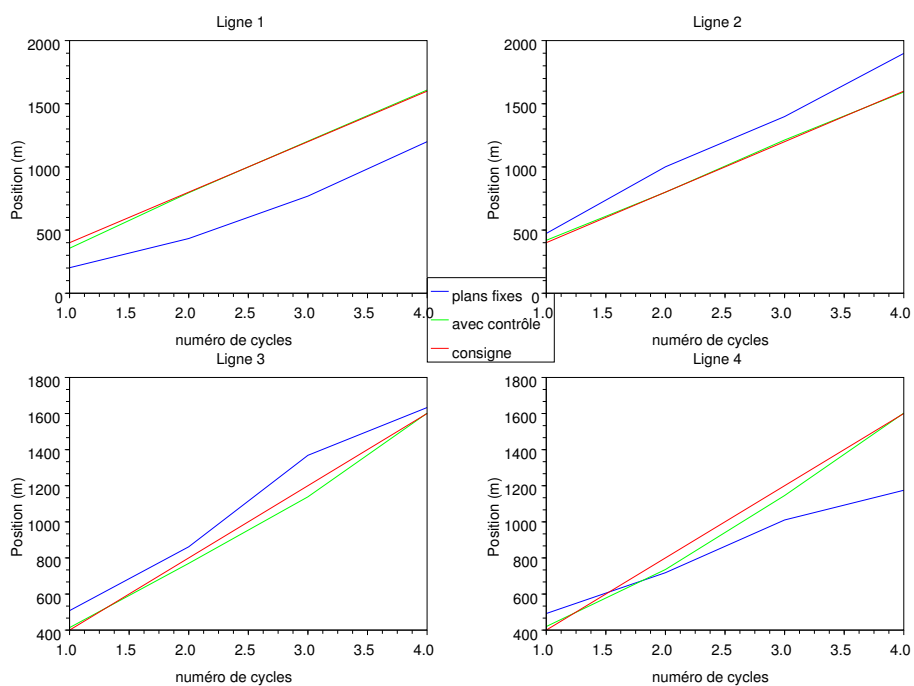


FIGURE 5.34 – Progression moyenne des 4 lignes de bus pour une consigne de 400 m

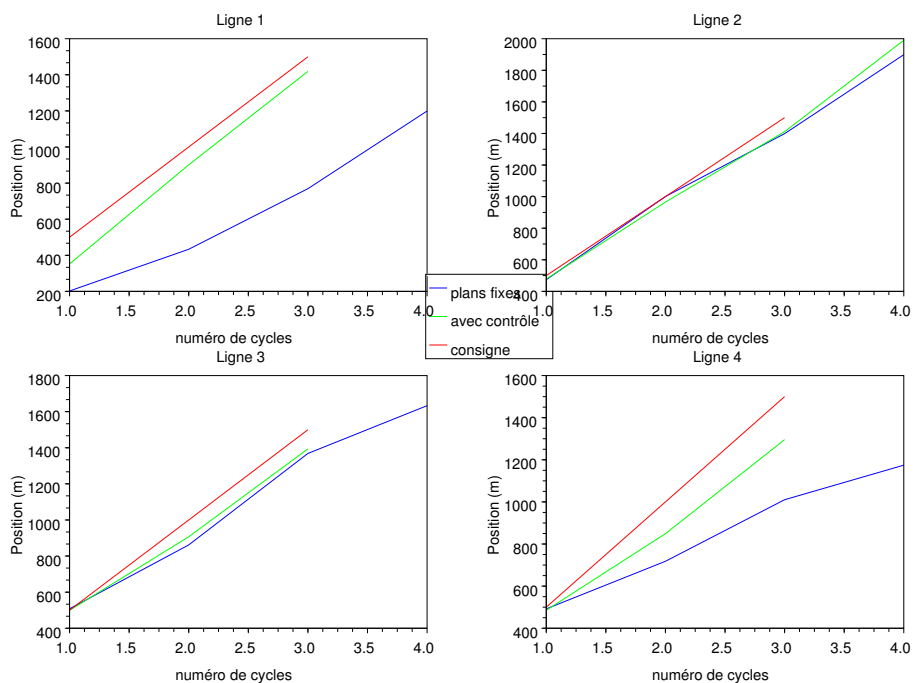


FIGURE 5.35 – Progression moyenne des 4 lignes de bus pour une consigne de 500 m

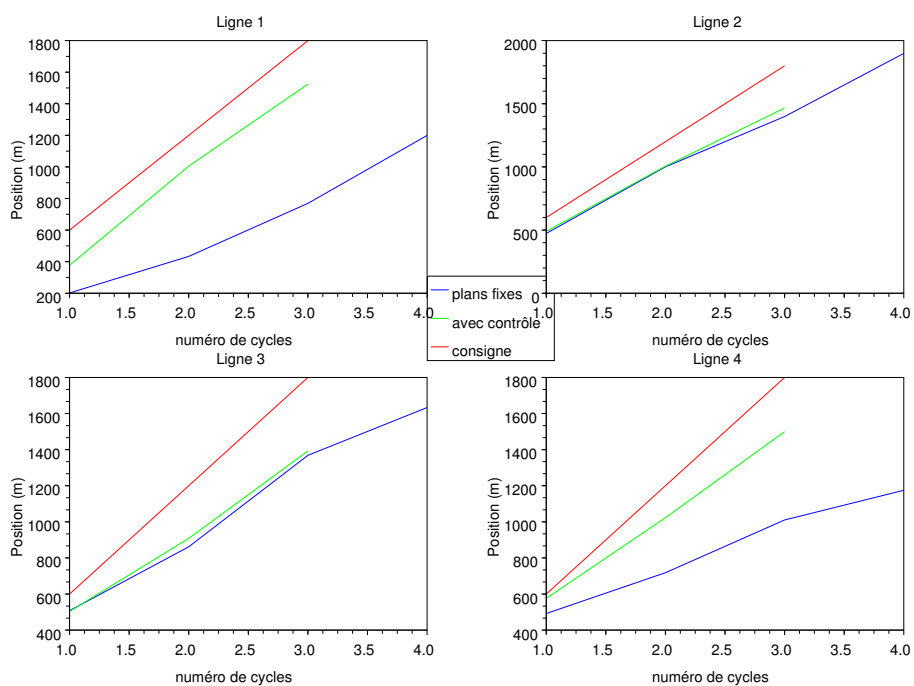


FIGURE 5.36 – Progression moyenne des 4 lignes de bus pour une consigne de 600 m

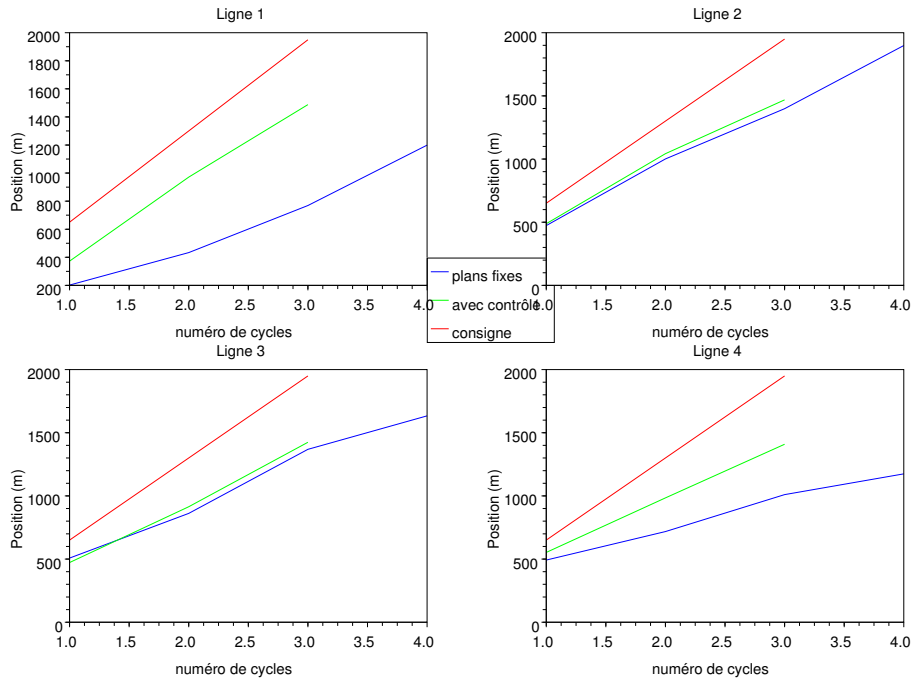


FIGURE 5.37 – Progression moyenne des 4 lignes de bus pour une consigne de 650 m

Pour les différentes valeurs de la consigne, nous observons très clairement que la stratégie essaye de faire en sorte que la progression réelle des bus suive la consigne donnée. Même si dans certains cas extrêmes comme pour les consignes de 100 m et 650 m, la stratégie n'arrive pas à coller aux consignes, celle-ci présente de meilleures performances que les plans de feux fixes.

Afin d'analyser un peu plus en détail le comportement de la stratégie pour les différentes consignes, nous distinguons les arcs du réseau traversés par les lignes de bus de ceux non traversés. Puis nous considérons la somme du nombre de véhicules particuliers de chaque groupe pour les 40 cycles de simulation. Nous considérons aussi la durée moyenne de vert des arcs de chaque groupe. La figure 5.38 donne les valeurs des deux variables pour les deux groupes.

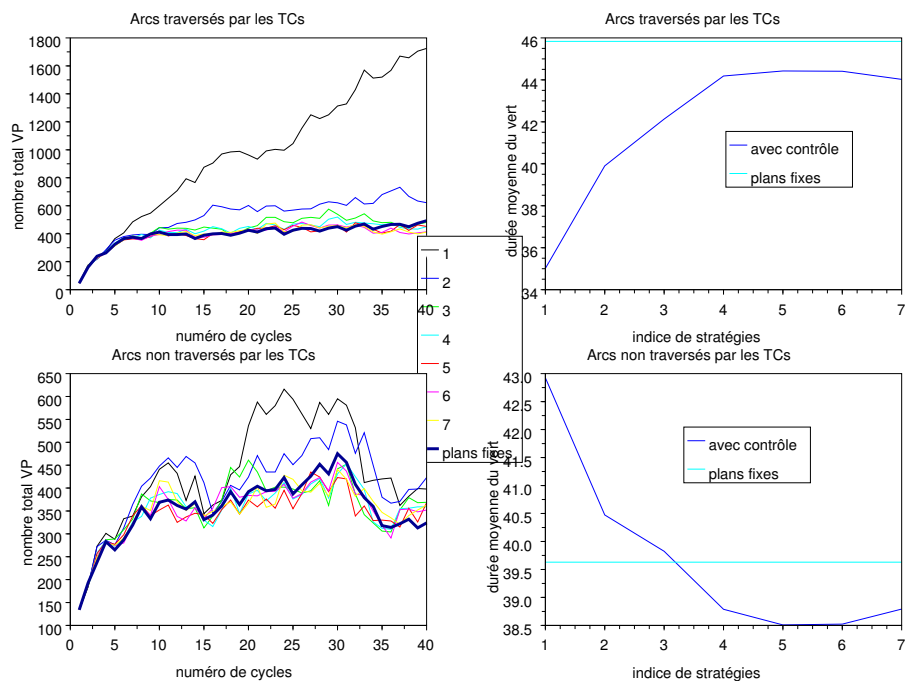


FIGURE 5.38 – Nombre de VP et durée moyenne de vert pour les deux groupes d’arcs

Pour les arcs traversés par les bus, la durée moyenne de vert allouée augmente avec les valeurs des consignes. Pour respecter des valeurs de consignes croissantes, il est tout à fait logique d’allouer plus de temps de vert pour les arcs pour faire attendre les bus le moins de temps possible et pour rendre le trafic général plus fluide. Le nombre de véhicules particuliers sur les arcs traversés par les bus décroît logiquement avec les valeurs des consignes. Pour les arcs non traversés par les bus, la durée moyenne de vert décroît avec les valeurs des consignes. Le nombre de véhicules particuliers sur ces arcs décroît mais pas dans les mêmes proportions que pour les arcs traversés par les bus.

Pour avoir une idée plus claire, nous considérons le carrefour 16 et les arcs 49, traversé par la ligne 4 de bus, et l’arc 51, non traversé par des lignes de bus. Sur la figure 5.39 sont représentés le nombre de VP et la durée moyenne de vert pour les 7 consignes et pour les deux arcs.

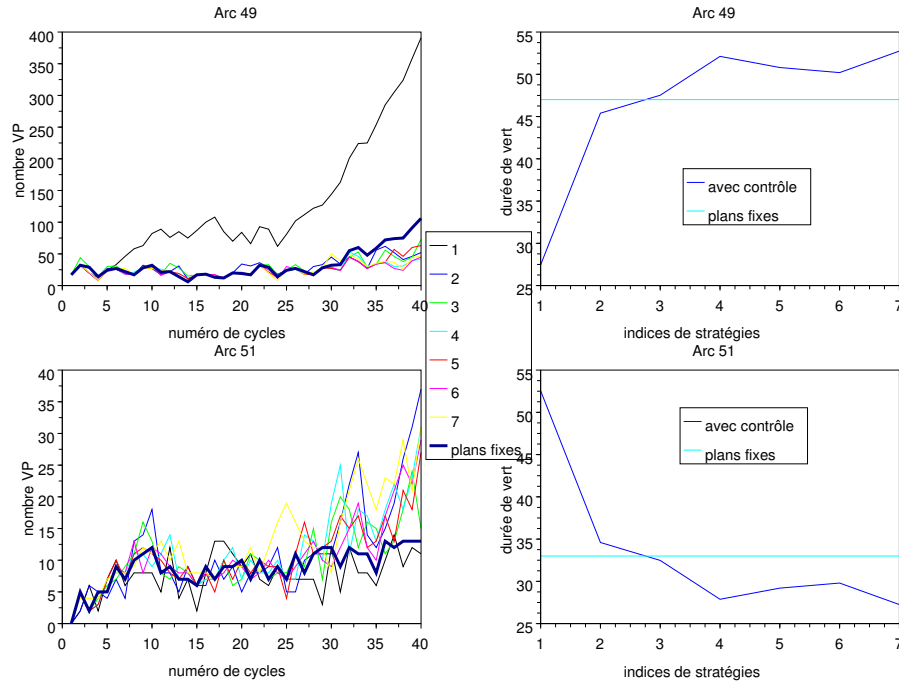


FIGURE 5.39 – Nombre de VP et durée moyenne de vert pour les arcs 49 et 51

Pour l'arc 49, la durée moyenne de vert augmente avec les consignes et le nombre de véhicules particuliers inversement. Pour l'arc 51, c'est l'inverse. La stratégie donne plus de vert à l'arc 49 ce qui résulte en une diminution du trafic général sur l'arc 49 et une légère augmentation sur l'arc 51.

5.6.2 Analyse selon la priorité : véhicules particuliers ou transport en commun

L'objectif de cette sous section est d'analyser le comportement de la stratégie selon les objectifs de la régulation globale que nous lui fixons. Ainsi, nous définissons trois scénarii de régulation :

- Scénario 1 : priorité donnée équitablement entre les véhicules particuliers et les véhicules de transport en commun.
- Scénario 2 : priorité absolue donnée aux véhicules particuliers. Les consignes pour les véhicules de transport en commun ne sont traitées pas dans l'algorithme d'optimisation.
- Scénario 3 : priorité absolue donnée aux véhicules de transport en commun. Aucune régulation n'est définie pour les véhicules particuliers.

Pour les véhicules de transport en commun et quand une priorité est définie (scénarii 1 et 3), cette consigne est fixée à 200 m par cycle. Ceci correspond à une forte consigne difficile à atteindre et qui nécessite le ralentissement du trafic. Les figure 5.40, 5.41 et 5.42 donnent la progression moyenne des quatre ligne de bus pour les 3 scénarii définis.

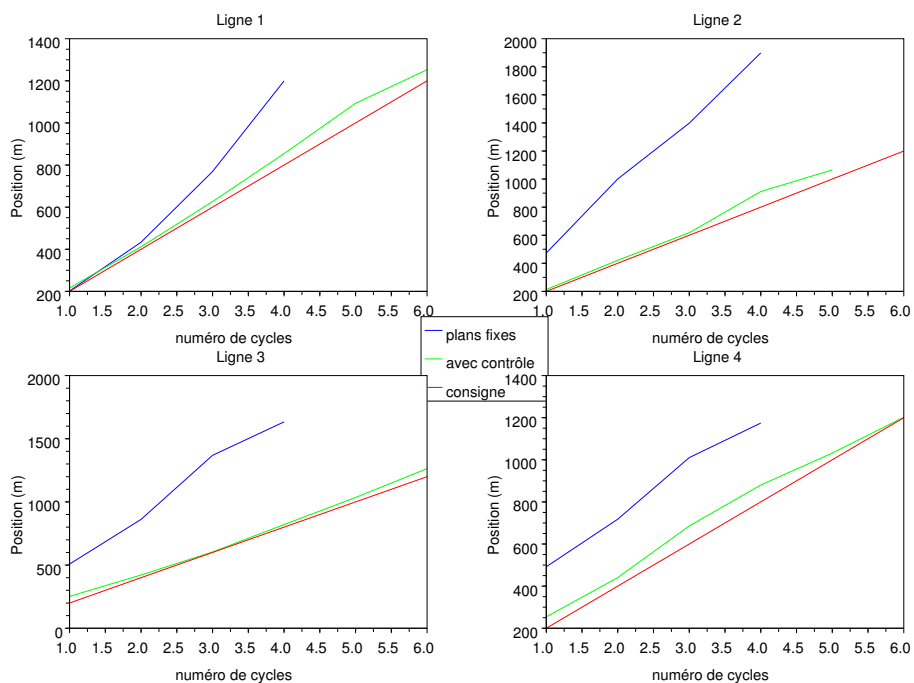


FIGURE 5.40 – Progression moyenne des 4 lignes de bus pour le scénario 1 : VP et TC

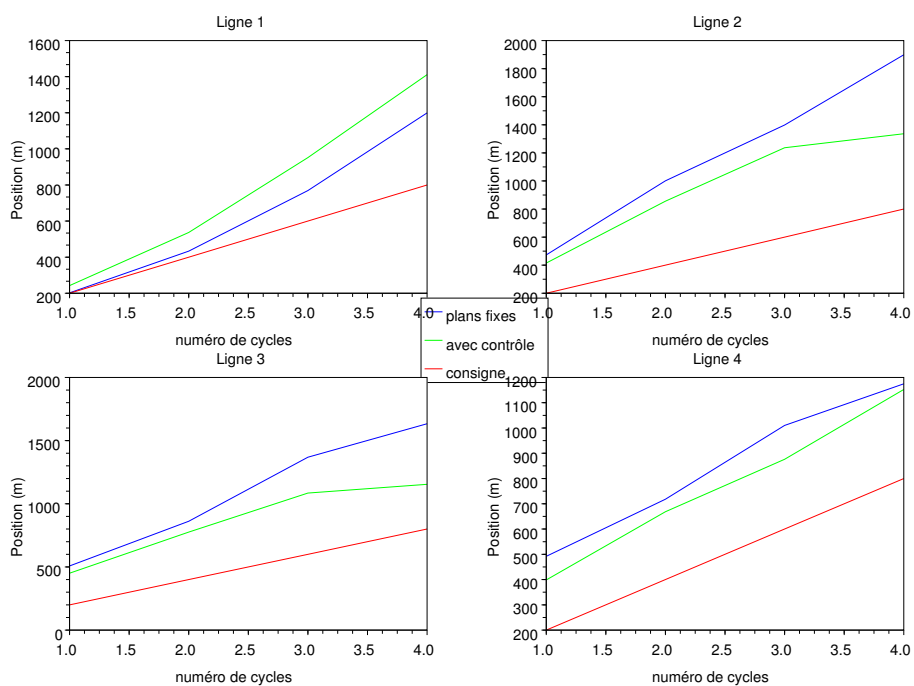


FIGURE 5.41 – Progression moyenne des 4 lignes de bus pour le scénario 2 : VP

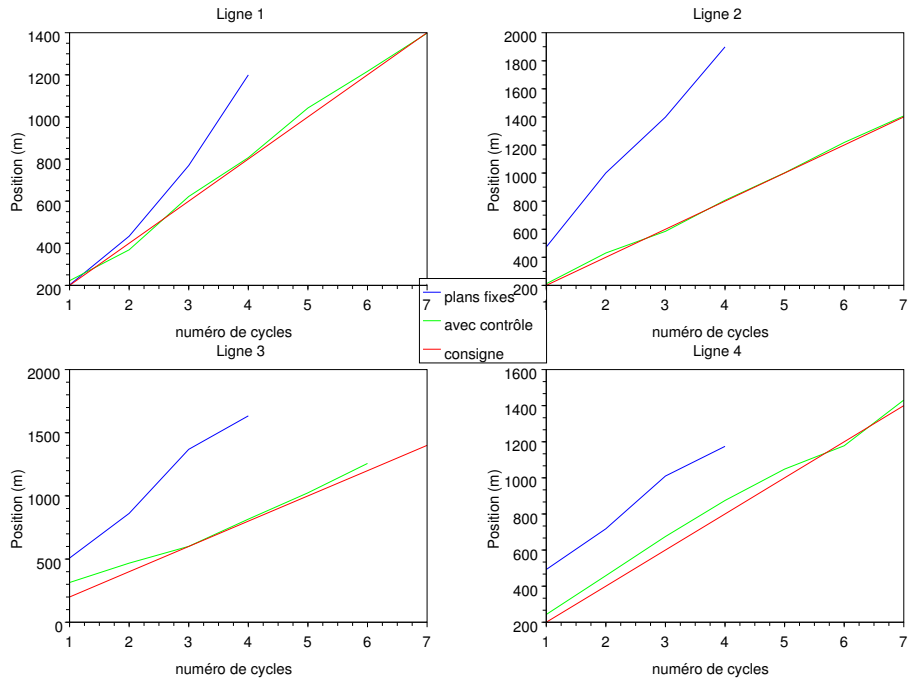


FIGURE 5.42 – Progression moyenne des 4 lignes de bus pour le scénario 3 : TC

Pour le scénario 2 (figure 5.41), la progression moyenne des quatre lignes est loin des consignes fixées. Pour le scénario 1 (figure 5.40), les bus essayent de suivre les consignes. Dans le dernier scénario (figure 5.42), les consignes sont les mieux respectées ce qui est tout à fait logique compte tenu que seule la régularité des bus est prise en compte.

Parallèlement, nous nous intéressons aux véhicules particuliers. L'évolution du nombre total de véhicules sur le réseau pour les 40 cycles de simulation et pour les 3 scénarii est donnée sur la figure 5.43.

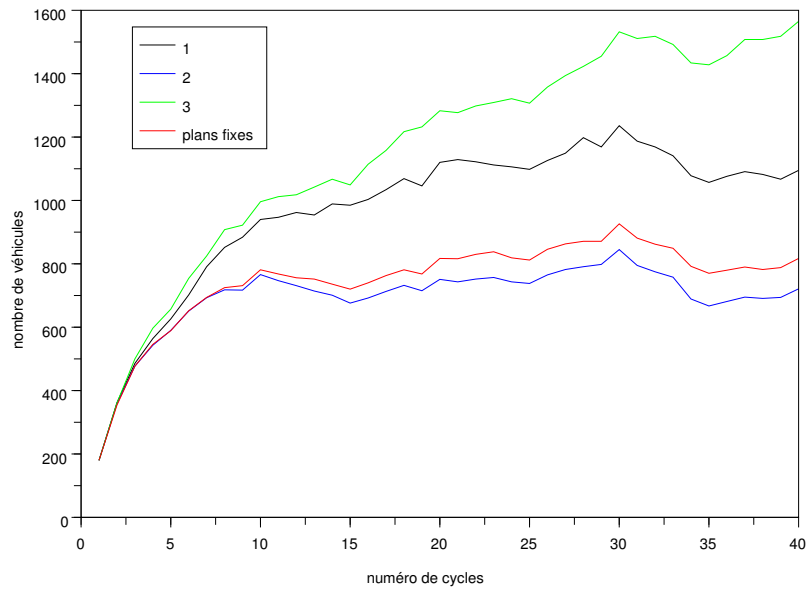


FIGURE 5.43 – Évolution du nombre total de véhicules pour chaque scénario

Nous remarquons sans difficulté que les meilleurs résultats pour les véhicules particuliers sont obtenus quand on donne une priorité absolue à ces véhicules. Pour une priorité absolue aux transports en commun, la stratégie a de plus mauvais résultats que pour des plans de feux fixes. Ceci est logique dans le sens où aucune régulation n'est demandée. Par contre, même pour une priorité mixte, les résultats pour les véhicules particuliers sont plus mauvais que les plans de feux fixes. Ceci s'explique simplement par les consignes données aux transports en commun qui sont très difficile à respecter et qui nécessitent un ralentissement du trafic général pour freiner les bus. La stratégie retient plus de véhicules sur les arcs que pour les plans de feux fixes. Pour des consignes plus facile à atteindre, la figure 5.44 montre l'évolution du nombre total de véhicules pour une priorité mixte comparée à des plans de feux fixes.

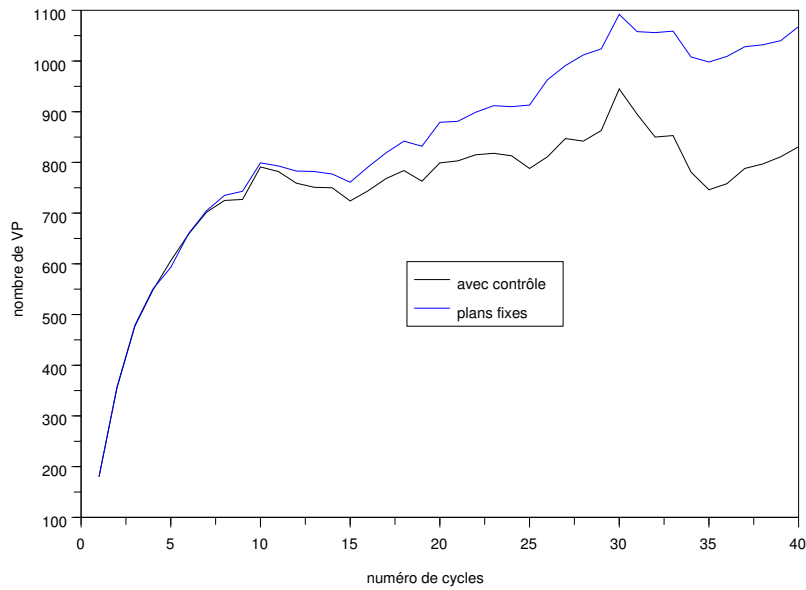


FIGURE 5.44 – Évolution du nombre total de véhicules avec une priorité mixte et consigne de 350 m

5.6.3 Comportement de la stratégie sous l'effet d'une perturbation simple

L'objectif de cette sous section n'est pas d'étudier les performances de la stratégie sous l'effet de plusieurs perturbations mais de montrer son comportement pour un exemple simple de perturbations. Nous considérons les arcs 18 et 31 qui relient respectivement le carrefours 7 à 6 et 11 à 10. Le premier est traversé par les bus de la ligne 4 et le second par les bus de la ligne 1. Avec une probabilité de 0.1 (respectivement de 0.2), nous introduisons une perturbation pour l'arc 18 (respectivement 31) qui consiste à multiplier par deux le nombre de véhicules particuliers. Moyennement, sur les 40 cycles de simulation, il y aura 4 cycles où une perturbation sera constatée sur l'arc 18 et 8 cycles sur l'arc 31.

La figure 5.45 montre la progression moyenne des bus pour les quatre lignes. Malgré les perturbations, les bus respectent toujours les consignes données (500 mètres par cycle). La stratégie, même dans un environnement perturbé, arrive à bien réguler la progression des bus.

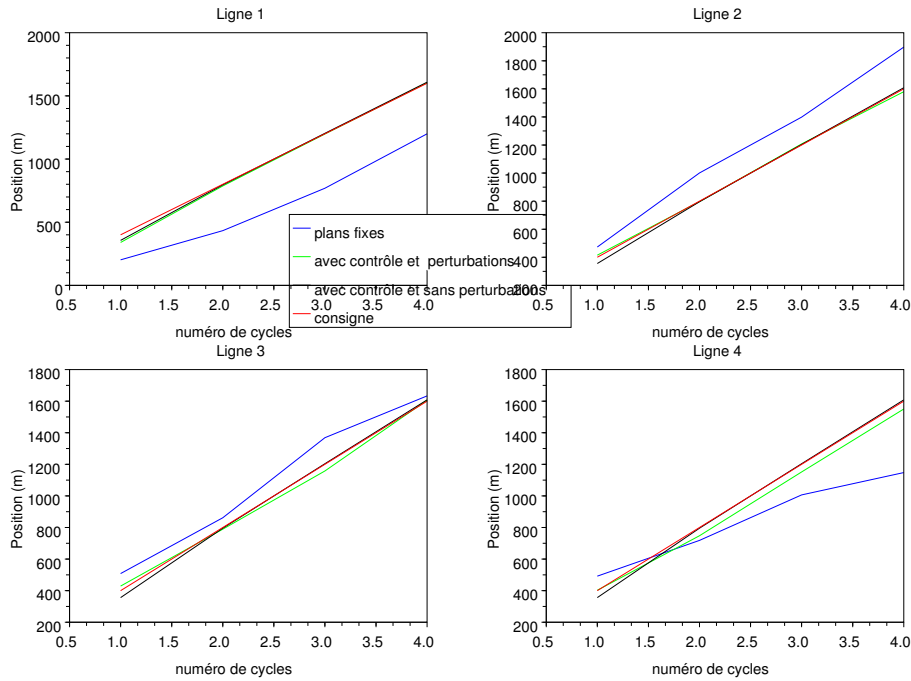


FIGURE 5.45 – Progression moyenne des 4 lignes de bus avec un trafic perturbé

Cette régulation par rapport aux perturbations est essentiellement due à la faculté de la stratégie à réguler le trafic général qui figure comme le second objectif. Ceci est effectué en augmentant la durée de vert pour les arcs perturbés. La figure 5.46 montre l'évolution du nombre de véhicules sur les arcs 18 et 31 et la durée moyenne de vert allouée à ces arcs pour la stratégie en mode normal et en mode perturbé.

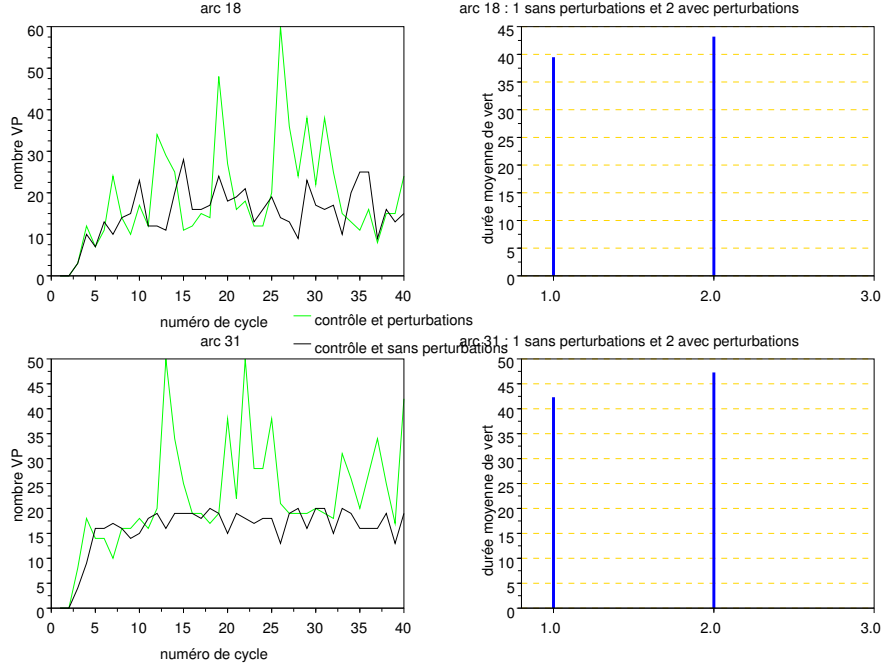


FIGURE 5.46 – Évolution du nombre de VP et durée moyenne de vert pour les arcs 18 et 31

La stratégie pour respecter ses deux objectifs fait augmenter la durée moyenne de vert pour les deux arcs puisque les perturbations consistent à une augmentation du trafic.

5.7 Conclusion

La mise en oeuvre de la stratégie de régulation et sa validation sur un réseau réel ou virtuel posent deux difficultés. D'abord sur le réseau urbain de test. La difficulté concerne le choix de la géométrie du réseau, des flux qui le traversent et des caractéristiques des plans de feux fixes. Pour cela, nous avons imaginé un réseau de grandes dimensions contenant 16 carrefours et 51 arcs. Il est traversé par 4 lignes de transport en commun. Les flux imaginés permettent d'avoir toutes les conditions du trafic. Les plans de feux fixes synthétisés servent comme références pour juger les résultats de la stratégie développée. Ils sont déterminés de façon optimale par rapport aux flux nominaux estimés. La seconde difficulté réside dans le calibrage des paramètres de l'algorithme d'optimisation et des différents paramètres de la stratégie pour avoir de meilleurs résultats. C'est pour cela que nous avons lancé la stratégie pour plusieurs valeurs des paramètres. Les résultats obtenus confirment les valeurs optimales rapportées dans la littérature spécialisée. L'étude de sensibilité des résultats par rapport aux horizons de simulation et de contrôle montre des résultats surprenants pour une architecture fondée sur la commande prédictive. Ceci peut ouvrir un champ de recherche intéressant sur la commande prédictive basée sur des algorithmes d'optimisation méta-heuristiques et sur les performances de cette commande pour des systèmes ayant des entrées exogènes non contrôlables.

La validation et les différents scénarii de test de la stratégie sur le réseau et pour différents objectifs de régulation montrent des gains considérables par rapport aux plans de feux fixes. Pour les véhicules particuliers, la

stratégie permet de diminuer le nombre total des véhicules sur tout le réseau. Pour les véhicules de transport en commun, la régularité des lignes est fortement améliorée pour différents objectifs et pour tous les états du trafic.

Conclusion et perspectives

L'objectif de cette thèse est le développement d'une stratégie de régulation du trafic urbain multimodale sur de larges réseaux urbains via les feux de signalisation. Les deux modes concernés sont les véhicules particuliers et les transports en commun de surface. Pour le premier mode, l'objectif de régulation est la minimisation des temps d'attente dans les carrefours et plus généralement des temps de parcours sur le réseau. Pour le second mode, l'objectif est de garantir la régularité des lignes traversant le réseau.

Dans la littérature, plusieurs travaux se sont intéressés à la régulation du trafic des véhicules particuliers via les feux de signalisation. Ceux qui régulent le trafic en temps réel utilisent un haut niveau de détail qui rend une utilisation centralisée impossible sur de larges réseaux. La stratégie *TUC* a permis de pallier cet inconvénient grâce à la commande linéaire quadratique qui permet de déterminer les commandes facilement et rapidement pour de très larges réseaux. Cependant, le modèle utilisé à la base de *TUC* a l'inconvénient de ne pas reproduire des conditions de trafic non saturé.

En ce qui concerne la régulation des transports en commun, la littérature rapporte dans la majorité des cas des systèmes de micro-régulation uniquement au niveau des carrefours isolés. De plus, nombre de ces systèmes se contentent de donner la priorité aux véhicules quel que soit le niveau de régularité de la ligne, des conditions du trafic en amont et en aval et des conséquences de telles actions sur les autres usagers.

La stratégie que nous avons proposé a l'avantage de formuler globalement le problème de régulation pour les deux modes sur de larges réseaux urbains. La stratégie fonctionne en boucle fermée en utilisant le concept intuitif de la commande prédictive généralisée. La première difficulté concerne la traduction des objectifs de régulation en des critères mathématiques. Nous avons construit deux critères chacun exprimant l'objectif de régulation pour un mode. Pour les véhicules particuliers, la régulation consiste à minimiser les temps de parcours qui dépendent essentiellement des longueurs de file d'attente au niveau des carrefours qui dépendent à leurs tours du nombre total des véhicules sur les arcs. Ainsi le premier critère est la somme de tous les véhicules sur chaque arc du réseau. Pour les transports en commun, la régulation consiste soit à maximiser la régularité des lignes ou plus simplement à faire de façon que les véhicules passent le moins de temps sur le réseau. La régularité d'une ligne correspond soit au respect d'une table horaire pré-établie soit au respect d'un intervalle régulier entre les véhicules successifs. Pour tenir compte de tous ces cas de figure, nous avons construit un critère original qui exprime l'écart quadratique entre les positions réelles des véhicules de transport en commun et des positions références à atteindre. Les positions références dépendent

du type de la régulation envisagée.

Le principe de la commande prédictive repose sur l'estimation des variables d'état dans le futur. Pour notre système, les variables sont d'une part le nombre de véhicules particuliers dans chaque arc du réseau et d'autre part les positions de chaque véhicule de transport en commun. Les estimations de ces variables sont effectuées par deux modèles que nous avons développés. Pour les véhicules particuliers, nous reprenons le même modèle utilisé dans les stratégies *TUC* et *NeTPrior* en introduisant une modification pour tenir compte de conditions de trafic sous saturé. La modification, simple sur le plan théorique, a des conséquences très importantes. D'abord, elle permet au modèle d'être plus précis ce qui rend la régulation beaucoup plus efficace. Ensuite, elle a supprimé le caractère linéaire du modèle initial.

Pour les transports en commun, nous avons développé deux modèles originaux. Le premier, plus simple, ne tient compte que des durées des phases. Les instants de début sont considérés inconnus et représentés comme des variables aléatoires. Le deuxième permet de tenir compte de ces dernières variables et permet de s'affranchir de quelques restrictions. La difficulté pour le développement de ces deux modèles réside dans la réalisation d'un compromis entre simplicité et précision. Un modèle très précis nécessiterait beaucoup de calcul ce qui le rend inutilisable dans un environnement temps réel. Un modèle trop simpliste peut rendre la régulation inefficace puisqu'il reproduirait, dans ce cas, des situations loin de la réalité. Les deux modèles sont fondés sur les événements et permettent de reproduire efficacement les interactions entre les transports en commun et les autres véhicules.

Après l'estimation des variables d'état données par les modèles, un algorithme d'optimisation permet de déterminer les variables de contrôle optimales à appliquer au trafic réel compte tenu des deux critères construits. Une méthode qui a particulièrement fait ses preuves est la méthode d'optimisation par essais particuliers. Il s'agit d'une méta-heuristique dont la simplicité de mise en oeuvre et l'efficacité de convergence en font une méthode très prisée dans la littérature et très prometteuse. Nous avons implanté deux versions de cette méthode et nous avons testé des mécanismes pour adapter la méthode, initialement dédiée à l'optimisation mono-objectif, au cas de l'optimisation multi-objectif.

Nous avons proposé deux architectures de commande. La première est sous la forme classique de la commande prédictive. La deuxième utilise la commande linéaire quadratique (comme *TUC* ou *NeTPrior*) pour orienter l'algorithme d'optimisation vers les régions de l'espace de recherche les plus prometteuses. Cette architecture fait économiser à l'algorithme d'optimisation des recherches inutiles.

La stratégie, à travers les résultats de simulation, a réussi à satisfaire les objectifs de départ à savoir améliorer significativement le trafic général et la régularité des lignes de transport en commun. Dans certaines situations, les résultats sont exceptionnels ce qui démontre l'efficacité de la stratégie et la nécessité d'approfondir encore plus les simulations et aussi certains aspects théoriques et pratiques.

Du point de vue de la simulation, il sera intéressant et nécessaire de tester la stratégie et de la valider sur des logiciels de simulation du trafic permettant de tenir compte des spécificités de la stratégie et des variables

du trafic qui y interviennent.

Concernant la modélisation, il peut être envisagé d'améliorer les deux modèles des transports en commun en poussant la réflexion sur le calcul de la longueur de file d'attente devant les véhicules de transport en commun.

Pour l'algorithme d'optimisation, les suites de ce travail peuvent se situer sur deux axes :

- Améliorer la convergence de l'algorithme en explorant de nouveaux mécanismes propres à l'algorithme et en proposant de nouvelles architectures comme la parallélisation de l'algorithme.
- Proposer de nouveaux critères en décomposant les deux critères de base. Ceci nécessite alors d'adapter l'algorithme d'optimisation pour tenir compte de plusieurs critères.

Enfin, pour l'architecture générale de la commande, il sera intéressant de pousser la réflexion sur l'utilité de la commande prédictive et des longueurs des horizons de contrôle et de simulation dans le cas de systèmes possédant des entrées exogènes non contrôlables. D'une manière plus générale, il sera intéressant aussi d'investiguer plus en profondeur l'apport des méthode méta-heuristiques pour la commande prédictive.

Bibliographie

- [1] Corsim user's guide. *U.S. Department of Transportation, Federal Highway Administration*, 1999.
- [2] M. Abkowitz and I. Engelstein. Optimal control of headway variation on transit routes. *Journal of Advanced Transportation*, pages 73–88, 1986.
- [3] G. Abu-Lebdeh and R. F. Benekohal. Genetic algorithms for traffic signal control and queue management of oversaturated two-way arterials. *Transportation Research Board, No. 1727*, pages 61–67, 2001.
- [4] M. Van Aerde, B. Hellinga, M. Baker, and H. Rakha. Integration : An overview of traffic simulation features. *the 1996 Transportation Research Board Annual Meeting*, 1996.
- [5] ITS America. An overview of transit signal priority, prepared by the advanced traffic management systems committee and advanced public transportation system committee. 2002.
- [6] S. Araya, K. Abe, and K. Fukumori. An optimal rescheduling for online train traffic control in disturbed situations. *IEEE*, pages 489–494, 1983.
- [7] K.N. Balke, C. L. Dudek, and T. Urbanik. Development and evaluation of an intelligent bus priority concept. *79th Annual Meeting of the Transportation Research Board*, 2000.
- [8] A. Bazan. A distributed approach for coordination of traffic signal agents. *Autonomous Agents and Multi-Agent Systems*, pages 131–164, 2005.
- [9] C.W. Beard and A.K. Ziliaskopoulos. System optimal signal optimization formulation. *Transportation Research Board*, 2006.
- [10] F. Van Den Bergh. *An Analysis of Particle Swarm Optimizers*. PhD thesis, Department of Computer Science, University of Pretoria, 2002.
- [11] N. Bhouri, F. Boillot, and P. Vinant. Régulation multimodale du trafic routier et des transports en commun de surface. une classification des méthodes. *Recherche Transports Sécurité Vol. 25/98*, pages 53–72, 2008.
- [12] F. Boillot, J.M. Blosseville, J.B. Lesort, V. Motyka, M. Papageorgiou, and S. Sellam. Optimal signal control of urban traffic networks. *Proceedings of the 6th IEE International Conference on Road Traffic Monitoring and Control*, pages 75–79, 1992.
- [13] A. Bolleli, V. Mauro, and E. Perono. Models and strategies for dynamic route guidance - part b : A decentralize, fully dynamic, infrastructure supported route guidance. *Proceedings of the DRIVE Conference*, pages 99–105, 1991.
- [14] C. Braban and F. Boillot. *Les systèmes temps réel de commande de feux en milieu urbain*. Les collections de l'Inrets, 2003.

- [15] V. Van Breusegem and G. Campion. Traffic modelling and control for metro circle lines. *IFAC/IFIP/IFORS Symposium*, pages 125–132, 1989.
- [16] C. Charbonnier, J.L. Farges, and J-J. Henry. Models and strategies for dynamic route guidance - part c :optimal control approach. *Proceedings of the DRIVE Conference*, pages 106–112, 1991.
- [17] D.W. Clarke, C. Mohtadit, and P.S. Tuffs. Generalized predictive control part i. the basic algorithm. *Automatica*, Vol. 23, pages 137–148, 1987.
- [18] D.W. Clarke, C. Mohtadit, and P.S. Tuffs. Generalized predictive control part ii. extensions and interpretations. *Automatica*, Vol. 23, pages 149–160, 1987.
- [19] M. Clerc and J. Kennedy. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, pages 58–73, 2002.
- [20] K.C. Courage, C. E. Wallace, and J. A. Wattleworth. Effect of bus priority system operation on performance of traffic signal control equipment on nw 7th avenue. *U.S. Department of Transportation, Report No. UTMA FL-06-0006*, 1978.
- [21] C.F. Daganzo. The cell-transmission model : A dynamic representation of highway traffic consistent with the hydrodynamic theory. *Transportation Research 28B4*, pages 269–288, 1994.
- [22] C.F. Daganzo. The cell transmission model, part ii : Network traffic. *Transportation Research 29B2*, pages 79–93, 1995.
- [23] C. Diakaki. *Integrated control of traffic flow in corridor networks*. PhD thesis, Department of Production Engineering and Management, Technical University of Crete, 2002.
- [24] F. Dion and H. Rakha. Integrating transit signal priority within adaptive traffic signal control systems. *Transportation Research Record*, 2004.
- [25] M. Dotoli, M.P. Fanti, and C. Meloni. A signal timing plan formulation for urban traffic control. *Control Engineering Practice*, pages 1297–1311, 2005.
- [26] P.A. Duerr. Dynamic right-of-way for transit vehicles : Integrated modeling approach for optimizing signal control on mixed traffic arterials. *Transportation Research Board*, pages 31–39, 2007.
- [27] X.J. Eberlein. *Real-Time Control Strategies in Transit Operations : Models and Analysis*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, 1995.
- [28] X.J. Eberlein, N. H. M. Wilson, C. Barnhart, and D. Bernstein. The real-time deadheading problem in transit operations control. *Transportation Research B*, Vol. 32, pages 77–100, 1998.
- [29] X.J. Eberlein, N. H. M. Wilson, and D. Bernstein. Modeling real-time control strategies in public transit operations. *Lecture note in Economics and Mathematical Systems : Computer Aided Transit Scheduling*, pages 325–346, 1999.
- [30] I. Engelstein. *Methods for maintaining bus service regularity*. PhD thesis, 1983.
- [31] M. Estrada, L. Thorson, M. Campos, and F. Robusté. Bus headway variations : Improving bus reliability and passenger travel time in dynamic systems. 2007.
- [32] J.-L. Farges, J.-J. Henry, and J. Tufal. The prodyn real-time traffic algorithm. *Proceedings of the 4th IFAC Symposium on Transportation Systems*, pages 307–312, 1983.

- [33] M. Fellendor. A microscopic simulation tool to evaluate actuated signal control including bus priority. *64th ITE annual meeting, session 32*, 1994.
- [34] Y. Feng, J. Perrin, and P.T. Martin. Bus priority of scoot evaluated in a vissim simulation environment. *82th Annual Meeting of the Transportation Research Board*, 2003.
- [35] P.G. Furth. Alternating deadheading in bus route operations. *Transportation Science Vol. 19*, pages 13–28, 1985.
- [36] N.H. Gartner. Opac : Ademand-responsive strategy for traffic signal control. *Transportation Research Record 906*, pages 75–84, 1983.
- [37] Y. Hawas and H. Mahmassani. A decentralized scheme for real-time route guidance in vehicular traffic networks. *Proceedings of the 2nd World Congress of Intelligent Transport Systems*, pages 1956–1963, 1995.
- [38] J.J. Henry, J.L. Farges, and J. Tufal. The prodyn real time traffic algorithm. *IFAC Symposium*, 1983.
- [39] P.B. Hunt, D.L. Robertson, R.D. Bretherton, and M.C. Royle. The scoot on-line traffic signal optimisation technique. *Traffic Engineering & Control 23*, pages 190–199, 1982.
- [40] ITS. Public transport priority : State of the art review. *Rapport Projet PRISCILLA*, 2002.
- [41] J. Jacobson and Y. Sheffi. Analytical model of traffic delays under bus signal preemption : theory and application. *Transportation Research, 15B(2)*, pages 127–138, 1981.
- [42] J. Kennedy and R. Eberhart. Particle swarm optimization. *Proceedings of IEEE International Conference on Neural Networks*, 1995.
- [43] A. Kessaci, J.-L. Farges, and J.-J. Henry. Upper level for real time urban traffic control systems. *11th IFAC World Congress, Vol. 10*, pages 226–229, 1990.
- [44] S.G. Khasnabis, V. Reddy, and B.B. Chaudry. Signal preemption as a priority treatment tool for transit demand management. *Vehicle Navigation and Information System Conference Proceedings*, 1991.
- [45] Y. Li. *Real-Time Scheduling on a Transit Bus Route*. PhD thesis, 1994.
- [46] H.K. Lo. A novel traffic signal control formulation. *Transportation Research A 33*, pages 433–448, 1995.
- [47] H.K. Lo, E. Chang, and Y.C. Chan. Dynamic network traffic control. *Transportation Research A 35*, pages 721–744, 2001.
- [48] P.R. Lowrie. Scats : The sydney co-ordinated adaptive traffic system - principles, methodology, algorithms. *Proceedings of the IEE International Conference on Road Traffic Signalling*, pages 67–70, 1982.
- [49] J.S. Ludwick. Bus priority system : Simulation and analysis, final report. *U.S. Department of Transportation*, 1976.
- [50] H.S. Mahmassani and S. Peeta. Network performance under system optimal and user equilibrium dynamic assignments : Implications for advanced traveller information systems. *Transportation Research Record 1408*, pages 83–93, 1994.
- [51] V. Mauro and C. Di Taranto. Utopia. *Proceedings of the 6th IFAC/IFIP/IFORS Symposium on Control, Computers, Communication on Transportation*, pages 245–252, 1989.

- [52] P. Mirchandani and L. Head. Rhodes : A real-time traffic signal control system : architecture, algorithms and analysis. *TRISTAN*, 1998.
- [53] P. Mirchandani, L. Head, A. Knayazyan, and W. Wu. An approach towards the integration of bus priority, traffic adaptive signal ccontrol, and bus information/scheduling system. 2000.
- [54] N.Bhouri and P. Lotito. An intermodal traffic control strategy for private vehicles and public transport. *Euro Working Group on Transportation*, pages 423–428, 2005.
- [55] M. Papageorgiou. Dynamic modeling, assignment and route guidance in traffic networks. *Transportation Research 24B*, pages 471–495, 1990.
- [56] M. Papageorgiou. *Handbook of transportation engineering :Traffic control*. Kluwer Academic Publishers, 1999.
- [57] M. Papageorgiou. Méthodes de contrôle automatique dans les transports et la gestion du trafic. *Génie Logiciel, numéro 75*, pages 13–25, 2006.
- [58] H. Rakha and Y. Zhang. Sensitivity analysis of transit signal priority impacts on operation of a signalized intersection. *Transportation Engineering, 17B(5)*, pages 796–804, 2003.
- [59] J. Richalet, A. Rault, J.L. Testud, and J. Papon. Model predictive heuristic control : Applications to industrial processes. *Automatica*, pages 413–428, 1978.
- [60] A.J. Richardson and K. W. Ogden. Evaluation of active bus-priority signals. *Transportation Research Record 718*, pages 5–12, 1979.
- [61] D.I. Robertson. Transyt method of area traffic control. *Traffic engineering & Control 10*, pages 276–281, 1969.
- [62] S.R. Seward and R. N. Taube. Methodology for evaluating bus-actuated, signal-preemption systems. *Transportation Research Record 630*, pages 11–17, 1977.
- [63] A. Shalaby, J. Lee, J. Greenough, S. Hung, and M.D. Bowie. Development, evaluation, and selection of advanced transit signal priority concept directions. *Journal of Public Transportation, Vol. 9, No. 5*, 2008.
- [64] A. Skabardonis. Control strategies for transit priority. *Transportation Research Record 1727*, 2000.
- [65] D. Teodorovic. Swarm intelligence systems for transportation engineering : Principles and applications. *Transportation Research Part C*, pages 651–667, 2008.
- [66] R.A. Vincent, B. R. Cooper, and K. Wood. Bus-actuated signal control at isolated intersections-simulation studies of bus priority. *Transport and Road Research Laboratory Report 814*, 1978.
- [67] C.E. Wallace and K. Courage. *TRANSYT-7F User’s Manual*. McTransCenter, University of Florida, 1992.
- [68] Y. Wen and T. Wu. Regional signal coordinated control system based on an ant algorithm. *Fifth World Congress on Intelligent Control and Automation*, pages 5222–5226, 2004.
- [69] G. Zhou1, A.Gan1, and X. Zhu. Determination of optimal detector location for transit signal priority with queue jumper lanes. *Transportation Research Board*, pages 123–129, 2005.
- [70] S. Zolfaghari, N. Azizi, and M.Y. Jaber. A model for holding strategy in public transit systems with real-time information. *International Journal of Transport Management, Vol. 2*, pages 99–110, 2005.