

TP MOTUS 2014

Calage de deux modèles de poursuite sur des données de trajectoires

BACH Victor CLAIRAIS Aurélien

14/03/2014

Sommaire

Introduction.....	2
Partie 1 : Observation des trajectoires.....	3
Partie 2 : Simulation de paires de véhicules se suivant continument pendant toute la traversée de la section	6
Modèle de Newell	6
Modèle de General Motors	7
Critique des paramètres	9
Partie 3 : Calage des deux modèles de poursuite	11
Conclusion	16
ANNEXES.....	17

Table des figures

Figure 1 : Emplacement de la portion d'autoroute étudiée. Source : Google Maps	2
Figure 2 : Trajectoires (x,y) de 120 véhicules	3
Figure 3 : Vitesses moyennes des véhicules dans la portion étudiée	4
Figure 4 : Trajectoires (x,t) et zoom (à droite).....	5
Figure 5 : Représentation de la trajectoire du suiveur comme une translation de celle du leader dans le cadre du modèle (Newell, 2002) d'après (Chiabaut et al., 2010)	6
Figure 6 : Illustration de la méthode de Newell	7
Figure 7: Simulation GM avec C=8.....	8
Figure 8: Simulation GM avec C=30.....	8
Figure 9: Zoom sur les problèmes de la simulation.....	9
Figure 10 : Tracés pour différentes paires de véhicules.....	10
Figure 11 : Fonction de coût scénario "Newell/RMSE"	11
Figure 12 : Fonction de coûts scénario "GM / RMSE"	12
Figure 13: Fonction de coûts scénario "Newell/MAE"	12
Figure 14 : Fonction de coûts scénario "GM / MAE"	13
Figure 15 : Tableau des paramètres optimaux pour les scénarios choisis	14
Figure 16 : Trajectoires (x,t) pour la paire 3 avec le GoF RMSE	14
Figure 17 : Trajectoires (x,t) pour la paire 3 avec le GoF MAE	15

Introduction

Le TP de MOTUS 2014 propose d'étudier les trajectoires individuelles de véhicules ayant emprunté l'autoroute américaine I80 au niveau d'Emeryville en Californie. Ces trajectoires ont été recueillies via le programme NGSim. La situation géographique exacte de la portion d'autoroute étudiée est explicitée en Figure 1.

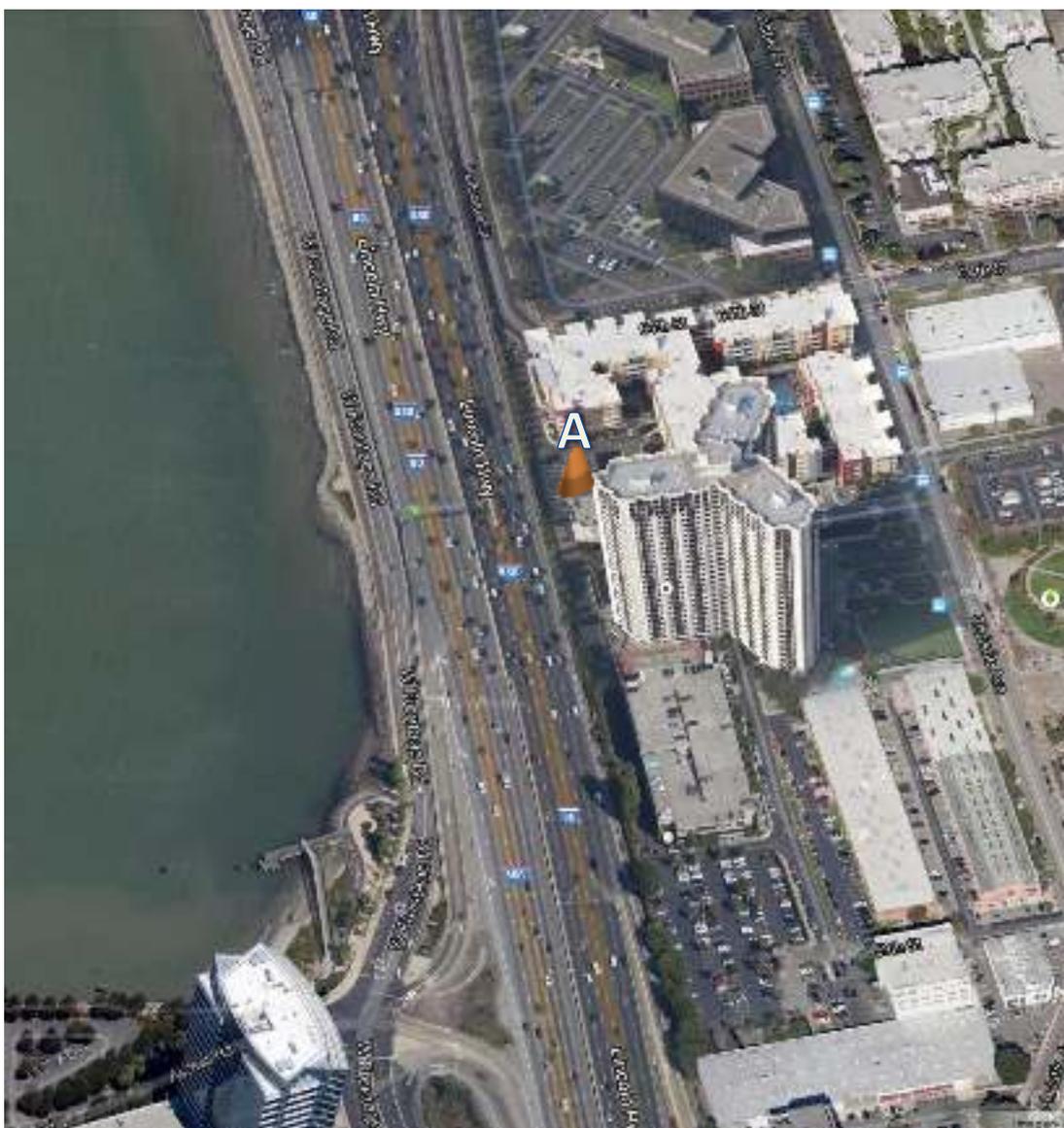


Figure 1 : Emplacement de la portion d'autoroute étudiée. Source : Google Maps

La méthode de capture des trajectoires est la méthode du programme NGSim, un programme de suivi de véhicule par caméras. Ce n'est pas moins de sept caméras placés au point A (cf. Figure 1) qui ont permis de capturer les trajectoires de plus de 2500 véhicules sur une portion totale de 500m. Le but de cette étude est d'étudier la cohérence et de caler deux modèles connus de suivi du trafic. C'est ainsi que nous étudierons, en amont, les trajectoires des véhicules, puis les deux modèles proposés et enfin, nous calerons et critiquerons ces modèles.

Partie 1 : Observation des trajectoires

Les variables auxquelles nous avons accès sont :

- Le numéro du véhicule (attribué dans l'ordre d'apparition)
- L'instant de la prise d'image en sec (tous les dixièmes de seconde)
- La position longitudinale en mètre
- La position latérale en mètre
- La voie dans laquelle le véhicule circulait lors de sa détection

Ces variables sont donc regroupés dans la variable *Data.mat* dont les dimensions sont 1262678x5. Par des calculs simples, on a donc accès à :

- La vitesse instantanée des véhicules
- L'accélération des véhicules
- Les espaces inter-véhiculaires
- D'autres variables plus ou moins utiles dans la suite de l'étude

L'infrastructure est du type autoroute à 6 voies sur la portion considérée. Les trajectoires de 120 véhicules sont présentées en Figure 2.

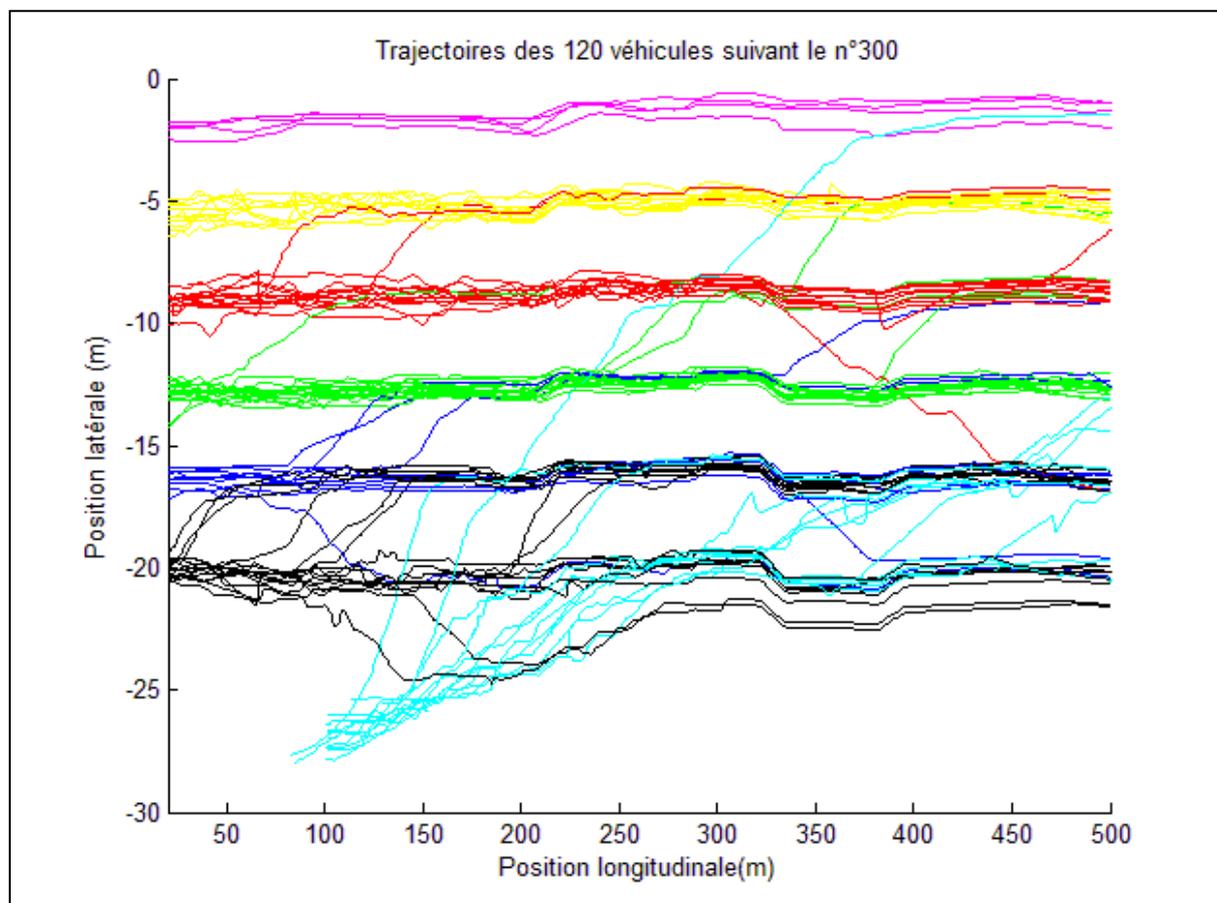


Figure 2 : Trajectoires (x,y) de 120 véhicules

On remarque en fait 7 couleurs différentes. Les couleurs rose, jaune, rouge, verte, bleue, noire représentent respectivement des véhicules ayant fait leur entrée sur le tronçon étudié sur la voie 1, 2, 3, 4, 5, ou 6. Ce qui interpelle est la présence d'une septième couleur pour seulement 6 voies. En fait, au vue des trajectoires représentées, on peut affirmer que la septième voie est une bretelle d'insertion car elle disparaît après avoir rejoint la voie 6. Sur la Figure 2, on peut observer les différents changements de voie des véhicules. Si pour les voies 2 à 6, les changements sont nombreux et laisse présager du fait que les utilisateurs dépassent le véhicule les précédent, il n'y a aucun changement de voie pour la voie 1. Le fait est que la voie 1 possède le symbole \diamond signifie que la voie est réservée au covoiturage. Cela explique que en période de congestion, les véhicules des voies 2 à 6 roulent à une allure lente ($\approx 7\text{m/s}$) alors que les véhicules empruntant la voie de covoiturage roulent à une allure plus soutenue ($\approx 24\text{m/s}$) comme le montre la Figure 3 mais aussi l'annexe 1 représentant les moyennes des vitesses instantanées des véhicules présents sur chaque voie, à chaque instant.

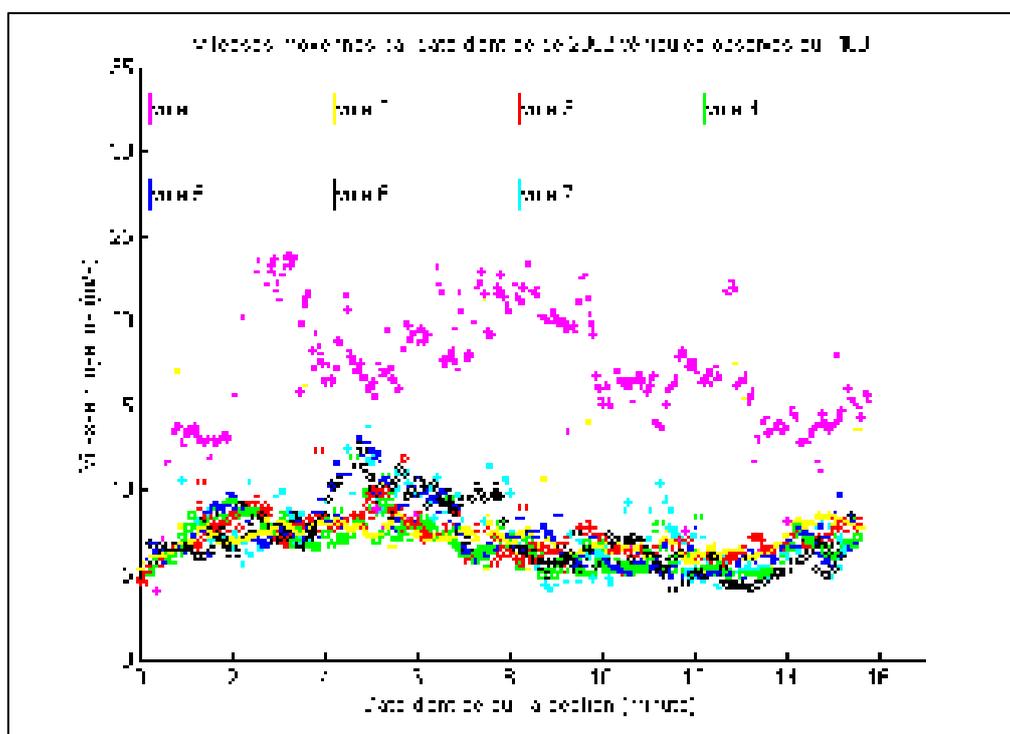


Figure 3 : Vitesses moyennes des véhicules dans la portion étudiée

Le graphe précédent représente déjà un indice du fait que l'étude s'est déroulée en période de congestion. Afin de mieux comprendre pourquoi, il nous faut représenter les trajectoires (x,t) . Les graphiques sont présents en Figure 4 (page suivante). On peut faire les mêmes remarques que précédemment. Les véhicules sur la voie de covoiturage ont une vitesse élevée (pente très inclinée) et constante. Cela corrobore bien la thèse de la voie rapide réservée à ce type de voyageurs. Sur les autres voies, les pentes sont plus faibles et non constantes. Les périodes de ralentissement (ou même d'arrêt) sont caractérisés par des baisses de l'inclinaison des courbes, visibles sur le zoom. Cette observation confirme le fait que l'étude a été menée en période de congestion où les véhicules s'arrêtent puis redémarrent constamment.

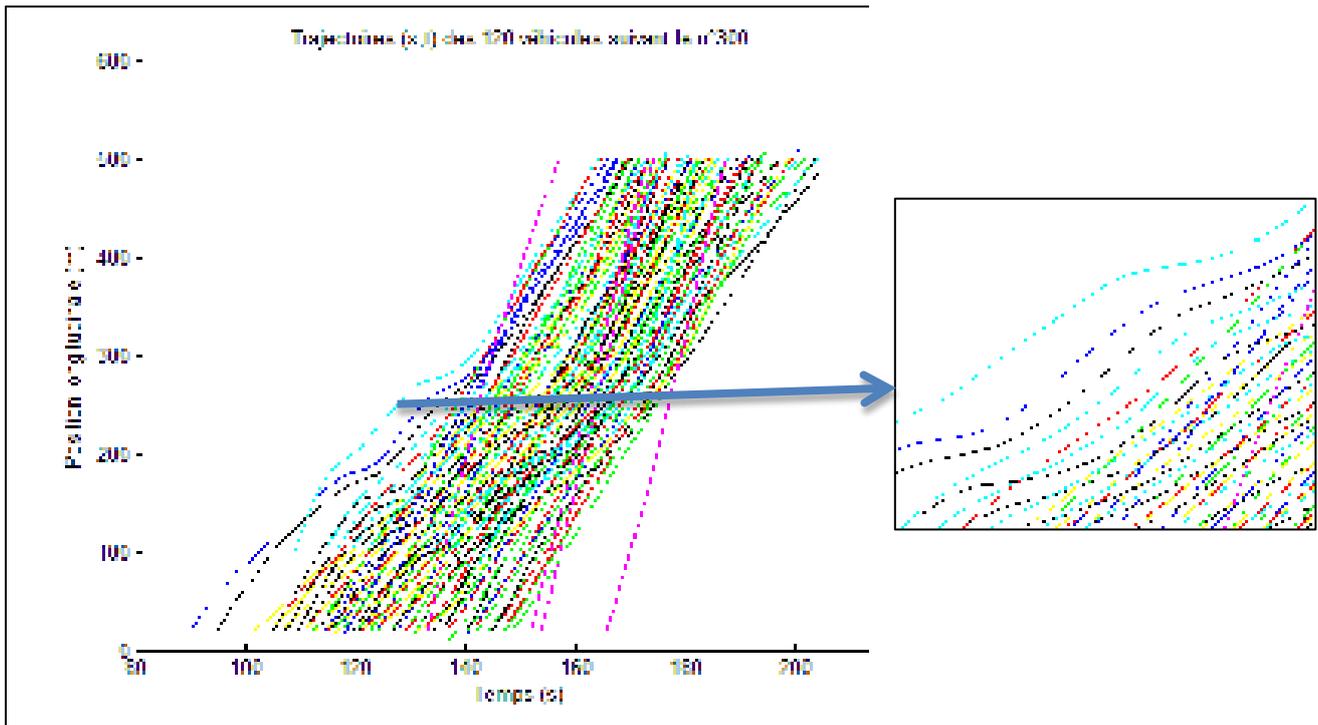


Figure 4 : Trajectoires (x,t) et zoom (à droite)

En résumé de cette partie, l'étude a été menée sur une portion d'autoroute à 6 voies + 1 voie d'insertion dont une voie est réservée au covoiturage. L'analyse des vitesses a permis de vérifier que l'étude s'est faite lors d'une période de congestion. (Heure de pointe ?) Nous devons maintenant simuler des trajectoires de paires de véhicules se suivant sur toute la longueur du tronçon d'étude.

Partie 2 : Simulation de paires de véhicules se suivant continuellement pendant toute la traversée de la section

Dans un modèle de poursuite simple, on considère que le suiveur adapte sa vitesse directement en fonction de celle de son leader et qu'il ne considère pas plus d'un leader. C'est ainsi que certaines caractéristiques sont passées sous silence. On peut en citer certaines :

- Les différentes caractéristiques des véhicules ne sont pas prises en compte. En effet, en réalité la capacité d'accélération d'un véhicule dépend d'un nombre important de variables. (masse, motorisation...)
- Des études ont montré qu'en réalité, le suiveur n'adapte pas son allure qu'en fonction d'un leader mais d'au moins deux leaders. (Le véhicule précédant le précédent...)
- En réalité, au volant, nous adaptons notre vitesse en fonction du leader mais aussi en fonction de ce qu'il se passe sur les voies autour. (dépassement, rabattement...)

Nous allons maintenant analyser les différents modèles proposés.

Modèle de Newell

Le modèle de Newell est vise à considérer la trajectoire du suiveur comme une translatée de la trajectoire du leader, comme l'atteste la figure suivante.

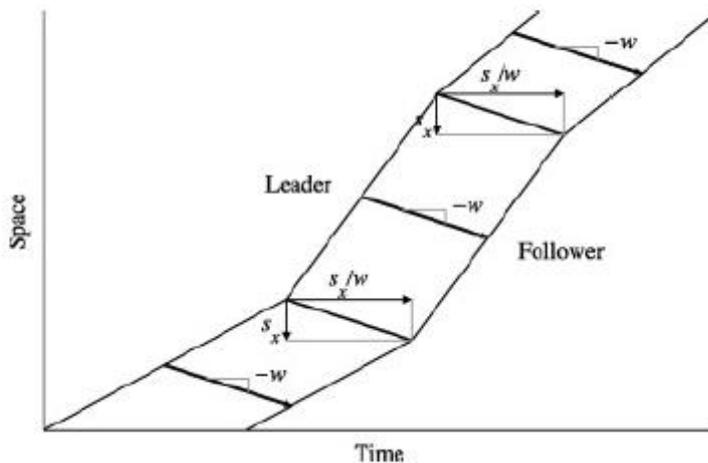


Figure 5 : Représentation de la trajectoire du suiveur comme une translation de celle du leader dans le cadre du modèle (Newell, 2002) d'après (Chiabaut et al., 2010)

La translation dépend de la vitesse individuelle de remontée de la congestion w (pour la direction) et de la distance à l'arrêt entre le leader et le suiveur S_x (pour la longueur). En conséquence, la méthode pour calculer la trajectoire du suiveur en fonction du leader est la suivante. Les considérations suivantes descendent directement de l'analyse de la fonction **Newell.m** fourni.

1. On donne la trajectoire (x,t) du leader dans un repère (X,T) défini par une rotation du repère initial autour de l'origine et d'un angle $\theta = \arctan\left(\frac{1}{-w}\right)$ qui correspond à l'angle orienté entre l'axe $(O; x)$ et les droites de pentes $-w = -\frac{S_x}{\tau}$.
2. On translate la trajectoire du leader d'une longueur valant dans le repère intermédiaire : $\frac{S_x}{\cos(w)}$.
3. On donne la trajectoire du suiveur dans le repère initial par la rotation d'angle $-\theta$.

Les différentes étapes sont données dans la figure suivante :

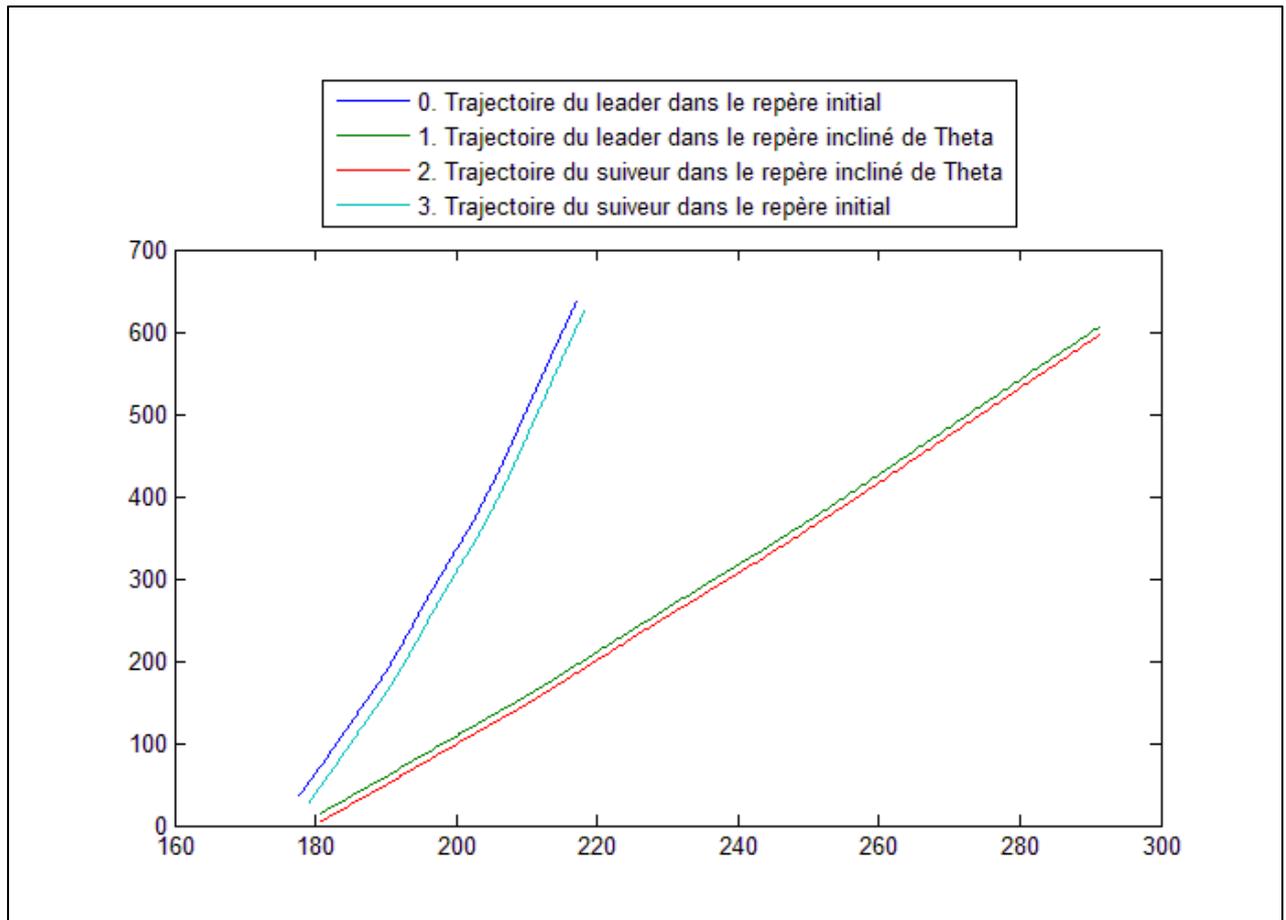


Figure 6 : Illustration de la méthode de Newell

Modèle de General Motors

A la différence du modèle de Newell, le modèle de General Motors programmé ne possède pas d'autre considération que celle de l'équation fournie dans l'énoncé du TP :

$$\ddot{x}_n(t + T^n) = C^n \times \frac{(x_{n-1}^{\dot{}} - x_n^{\dot{}})}{(x_{n-1} - x_n)}$$

Ici, l'accélération du suiveur est fonction de la différence de position entre lui-même et son leader et de la différence de vitesse entre lui-même et leader. En fait, dans le modèle de Newell, plus la distance entre lui et son leader est faible, plus le suiveur va adapter sa vitesse de manière brutale. Lorsqu'il est loin, le suiveur régule sa vitesse plus tranquillement. Les paramètres sont ici T et C . T est le temps de réaction du suiveur, et C est défini comme la sensibilité du suiveur ratio de la différence de vitesse par la différence de position entre lui-même

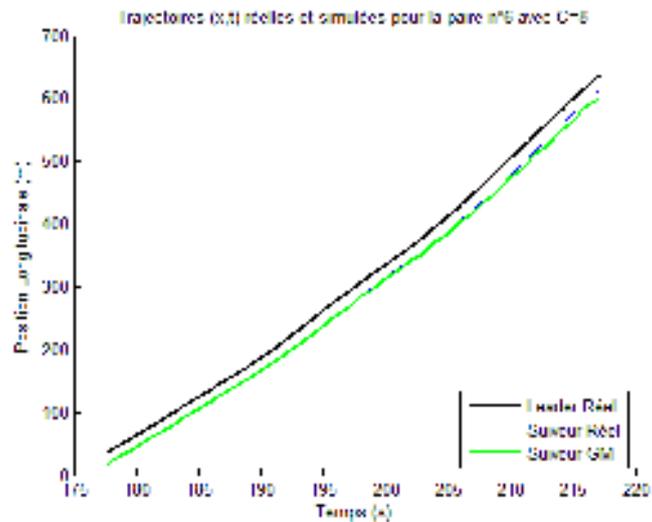


Figure 7: Simulation GM avec C=8

et son leader. Finalement ce qu'on sait sur ce paramètre, c'est que c'est une grandeur homogène à une vitesse (analyse dimensionnelle) qui permet de faire varier la sensibilité du suiveur aux différences de conduites de son leader. MatLab n'est pas un logiciel de calcul formel. En conséquence, tout calcul de dérivé ou d'intégration doit être discrétisé (calcul de la pente d'une corde \approx calcul de la pente de la tangente). Finalement, plus le pas de discrétisation est important, plus les erreurs sont majorées. A cela, on peut rajouter l'influence du paramètre C qui, en augmentant, augmente par la même occasion les erreurs de discrétisation. Ces dernières considérations (directement descendues de l'analyse de la fonction **GMMModel.m**) permettent d'expliquer les trajectoires simulées par le modèle de General Motors. En effet, ces courbes présentent des sauts dont l'amplitude est en relation avec le pas de discrétisation (qu'on ne peut modifier) et le paramètre C .

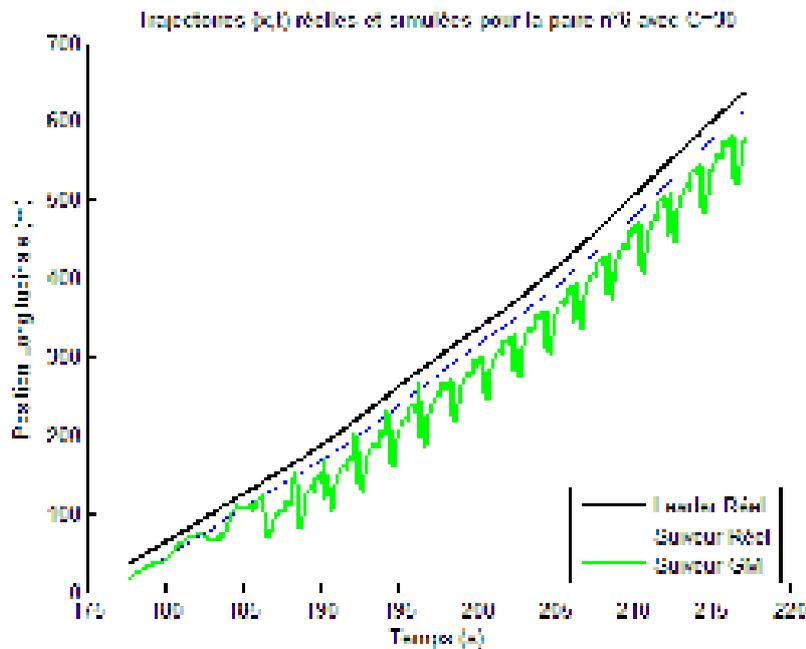


Figure 8: Simulation GM avec C=30

Critique des paramètres

Maintenant que les deux modèles ont été décortiqués, nous allons analyser et critiquer les paramètres par défaut.

Les paramètres utilisés sont :

- S_x est la distance à l'arrêt entre deux centres de véhicules se suivant. Par défaut, elle est fixée à 10m.
- τ est le temps de réaction d'un conducteur. Par défaut, il est de 1,2s ce qui est cohérent par rapport à la moyenne pour des conditions normales de conduite (sans alcool,...)
- C est la sensibilité du conducteur au ratio entre la différence de vitesse entre lui et son leader par la distance le séparant de son leader. Par défaut elle est fixée à 8m/s.
- T est le temps de réaction pour le modèle de General Motors. Par défaut, il est fixé pour ce modèle à 2s. Cette valeur est bien au-dessus des valeurs moyennes dans les conditions normales de conduite.

La figure en page suivante représente les trajectoires (x,t) réelles et simulées des leader (réelle) et suiveur (réelle et simulées). On peut y voir les différents problèmes ci-dessous.

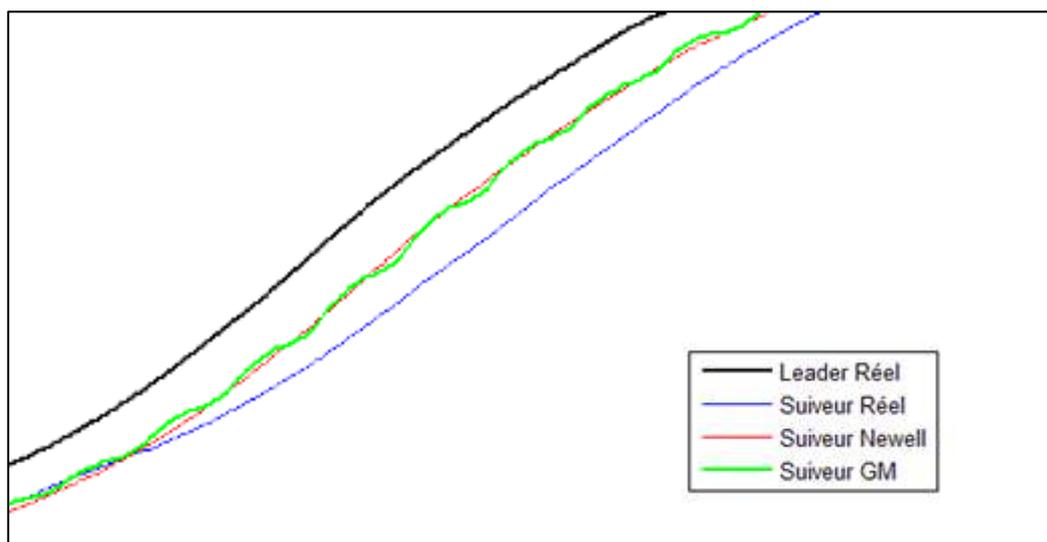


Figure 9: Zoom sur les problèmes de la simulation

On observe une différence entre le suiveur réel et les simulateurs. Ceci est caractéristique d'un temps de réaction sous-estimé, pour cet individu la tout au moins. On observe bien entendu les sauts dans la simulation de General Motors. La raison est expliquée auparavant.

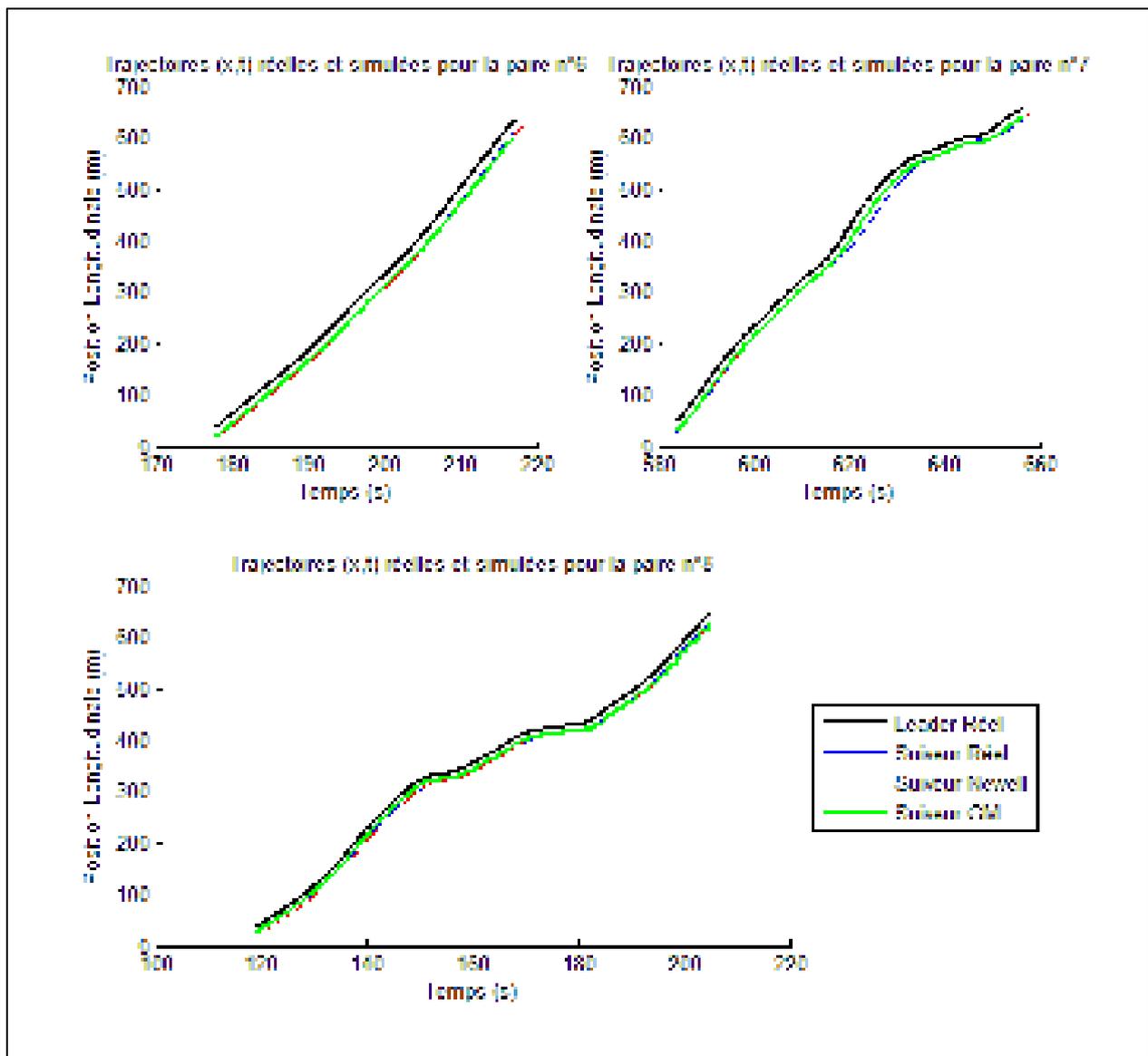


Figure 10 : Tracés pour différentes paires de véhicules

Au vue de toutes ces considérations, on peut affirmer que les modèles ne sont pas calés de manière optimale. Dans la troisième partie, nous allons essayer de caler au mieux les différents paramètres de simulation.

Partie 3 : Calage des deux modèles de poursuite

La première étape pour effectuer le calage des modèles proposé est de choisir une variable à caler. La seule variable directement accessible est la position. On propose ainsi de caler le modèle en fonction de la position. Une autre raison est que les calculs de dérivé et d'intégration nécessitent des approximations par discrétisation. Ceci est très souvent source d'erreur, on rejette donc l'utilisation de la vitesse et de l'accélération.

Le choix de GoF se fait en fonction des scripts fournis. En conséquence, on utilise le GoF « *Root mean square error* » (*RMSE*) qui permet de pénaliser les grandes erreurs et le « *Mean absolute error* » (*MAE*) qui est un bon indicateur mais qui ne permet pas de déterminer le signe des erreurs. Il faut donc ensuite caler les différents modèles, c'est-à-dire que nous devons déterminer les paramètres permettant un écart minimum dans chaque cas. La combinaison « *Newell / RMSE* » est codée dans *Partie3a.m* et la combinaison « *Newell / MAE* » est codée dans *Partie3abis.m*. Nous proposons de coder la combinaison « *GM / RMSE* » dans *Partie3a.m* et la combinaison « *GM / MAE* » dans *Partie3abis.m* afin d'avoir un comparatif direct et graphique pour les deux modèles.

Le script *Partie3a.m* est en annexe ***. Il donne les résultats suivant pour la paire n°3 choisie pour la clarté de sa représentation.

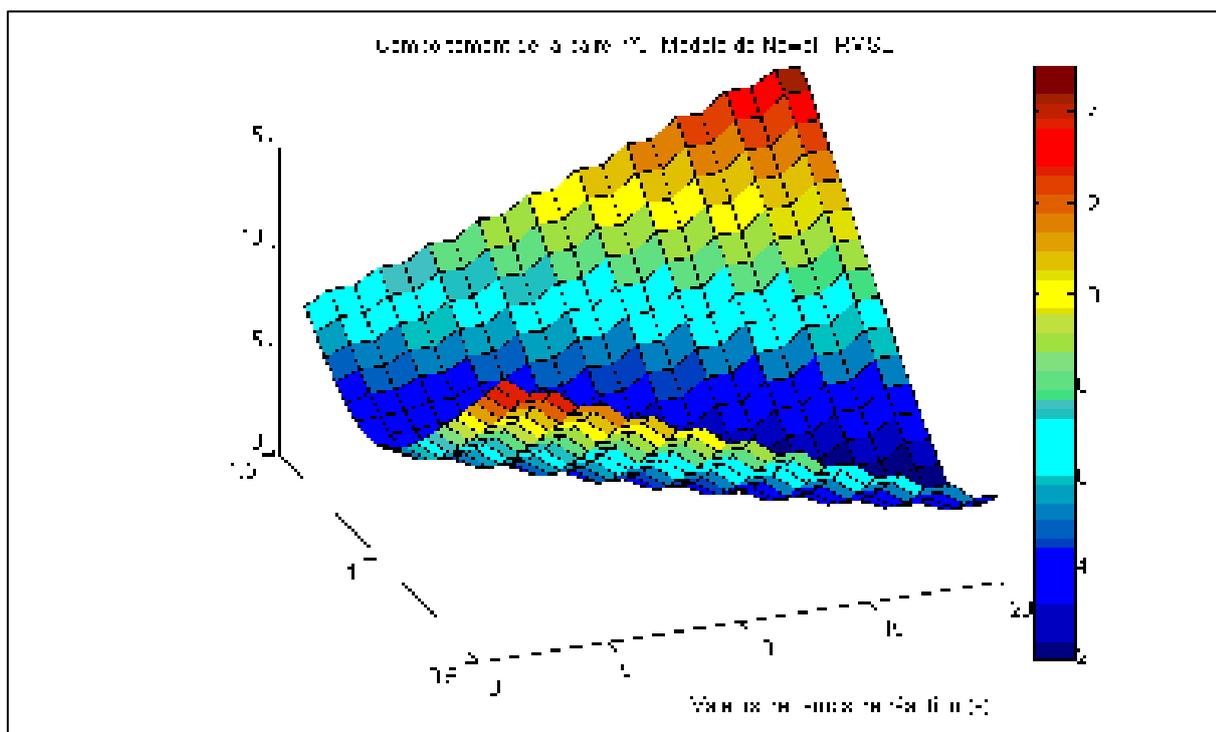


Figure 11 : Fonction de coût scénario "Newell/RMSE"

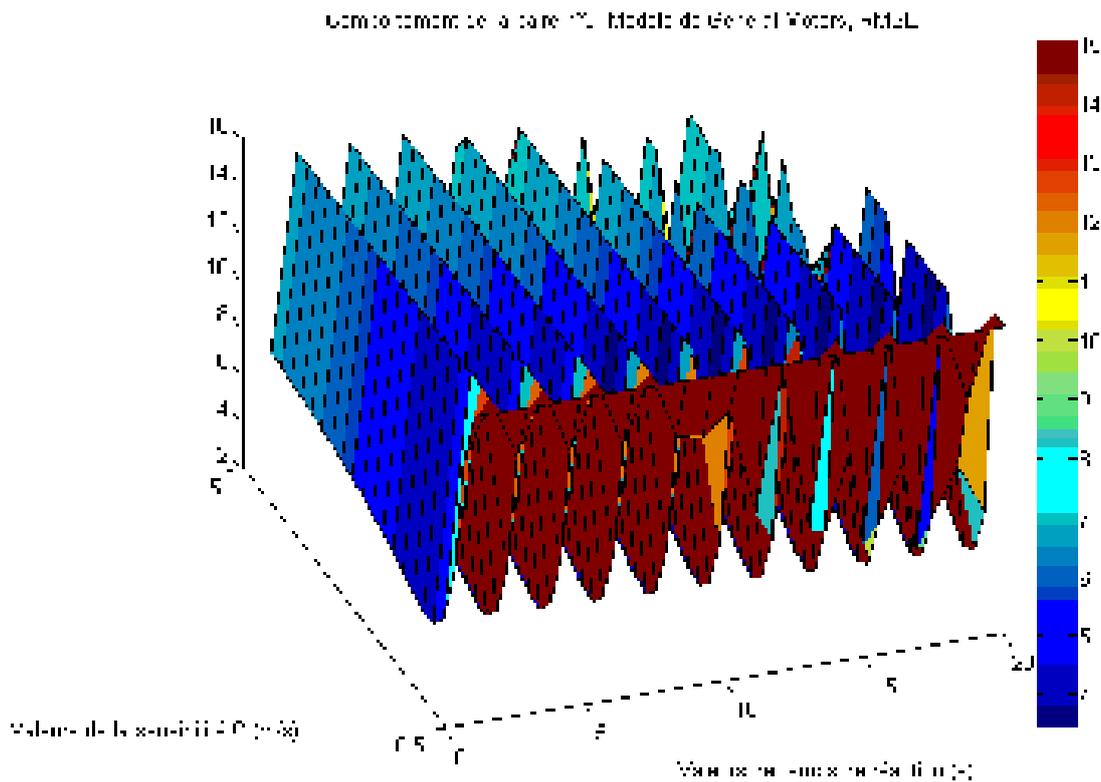


Figure 12 : Fonction de coûts scénario "GM / RMSE"

Le script *Partie3abis.m* sensiblement le même que le script *Partie3a.m* à la différence près que l'on appelle *mae_traj.m* plutôt que *rmse_traj.m*.

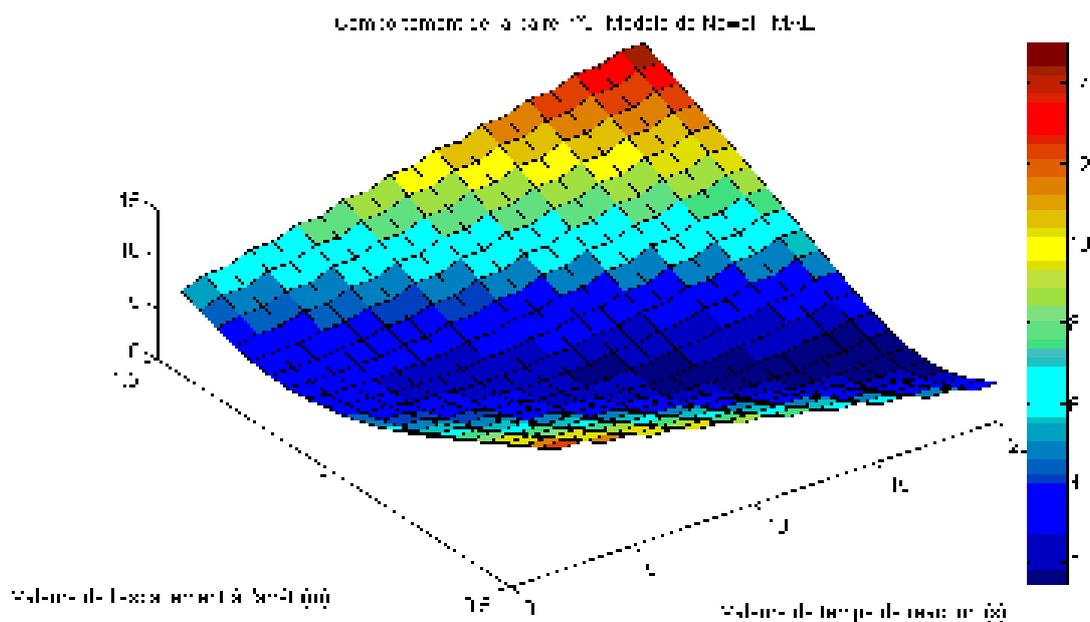


Figure 13: Fonction de coûts scénario "Newell/MAE"

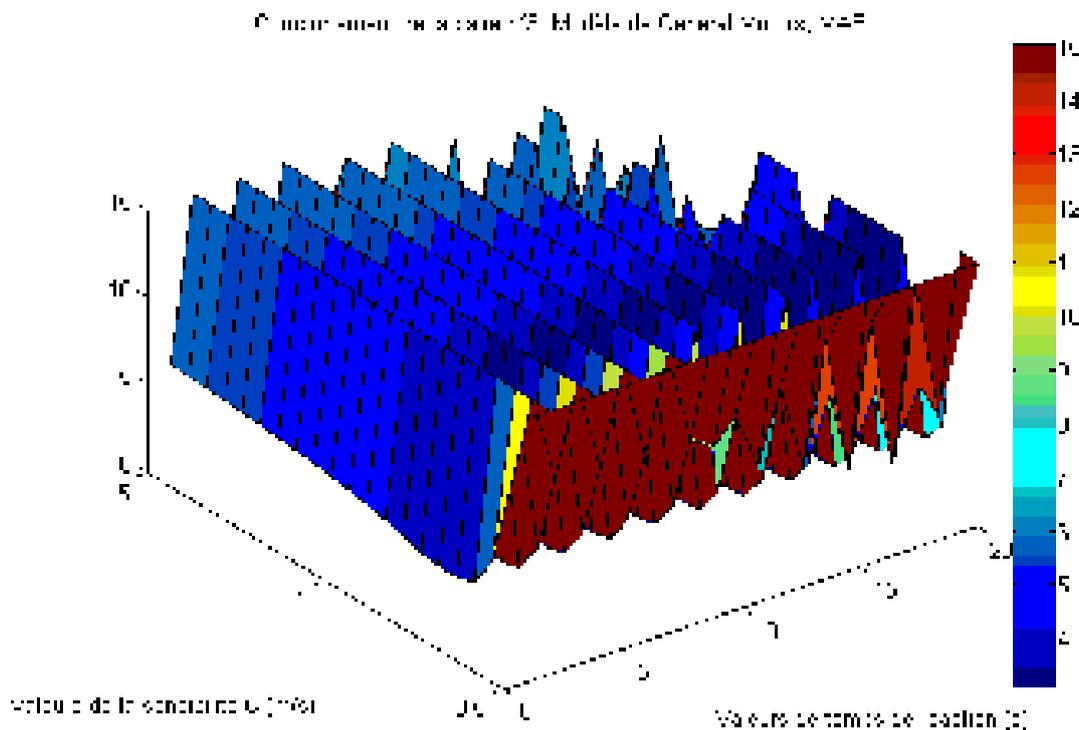


Figure 14 : Fonction de coûts scénario "GM / MAE"

Pour un même modèle, les formes obtenues sont sensiblement les mêmes, on remarque cependant que les amplitudes des valeurs atteintes changent. Le modèle de General Motors donne des résultats à l'allure beaucoup plus chaotique que le modèle de Newell. Une des raisons possibles est que le modèle de General Motors est beaucoup plus sensible aux variations de ses paramètres comparé au modèle de Newell. Cela peut être expliqué par le fait que l'on fait intervenir des dérivées calculées par discrétisation, et donc imprécises. De plus, l'influence du paramètre C était difficile à analyser, comme vu dans la partie 2. Cependant, il semblerait que les variations du paramètre de temps de réaction dans le modèle de Newell aient une grande influence sur la fonction de coûts.

En annexe 3, il y a la version améliorée du script *Partie3b.m* permettant en plus d'obtenir les paramètres optimaux pour le scénario « Newell / RMSE », l'ensemble des paramètres optimaux pour tous les scénarios « Modèle / GoF » choisis. Ces paramètres optimaux sont représentés par les points bleus. Le tableau précédant présente cette synthèse mais sous forme de tableau.

	Paire	1	2	3	4	5	6	7	8	9	10	11	12	13
Newell	S optimal RMSE	6.7	2.0	5.8	15.3	2.9	3.9	10.5	9.6	6.7	20.0	5.8	10.5	7.7
	S optimal MAE	8.6	2.0	6.7	14.3	2.0	3.9	7.7	8.6	6.7	20.0	6.7	8.6	10.5
	Tau optimal RMSE	1.0	1.2	1.4	0.8	1.5	1.2	1.5	0.6	0.8	0.9	1.5	0.9	0.9
	Tau optimal MAE	0.9	0.6	0.8	1.2	0.6	0.7	0.9	0.9	0.8	1.5	0.8	0.9	1.0
GM	C optimal RMSE	12.4	11.5	4.8	14.3	7.7	15.3	3.9	20.0	11.5	3.9	6.7	5.8	14.2
	C optimal MAE	12.4	12.4	4.8	15.3	5.8	13.4	5.8	15.3	11.5	3.9	6.7	5.8	14.2
	T optimal RMSE	1.2	0.6	1.4	0.5	1.4	0.5	1.5	1.2	0.6	1.3	1.3	1.3	0.5
	T optimal MAE	1.2	0.5	1.4	0.5	1.4	0.9	1.5	1.4	0.6	1.4	1.3	1.3	0.5

Figure 15 : Tableau des paramètres optimaux pour les scénarios choisis

Comme visible dans le tableau et sur les graphiques en annexe 3, dans certains cas, les valeurs des paramètres optimaux correspondent aux bornes des intervalles des paramètres choisis. Dans ces cas-là, le calage nécessite des intervalles de plus grande envergure afin de fonctionner. Afin d'illustrer le calage, nous avons tracé les trajectoires (x,t) pour la paire n°3, représentative d'un calage réussi.

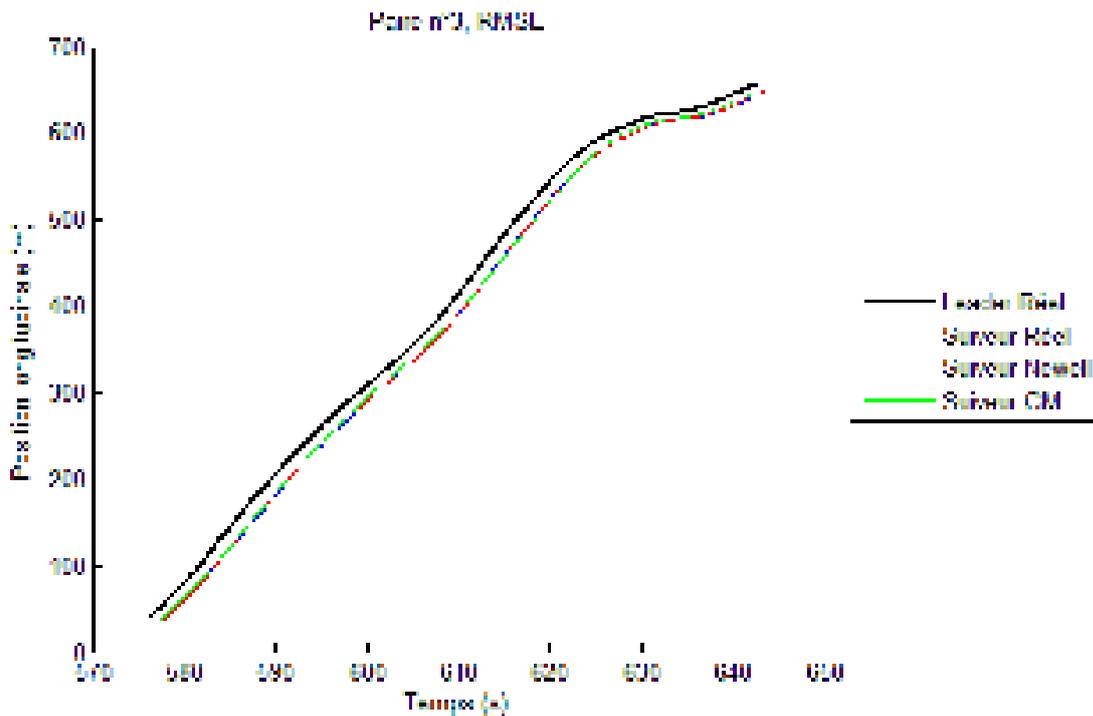


Figure 16 : Trajectoires (x,t) pour la paire 3 avec le GoF RMSE

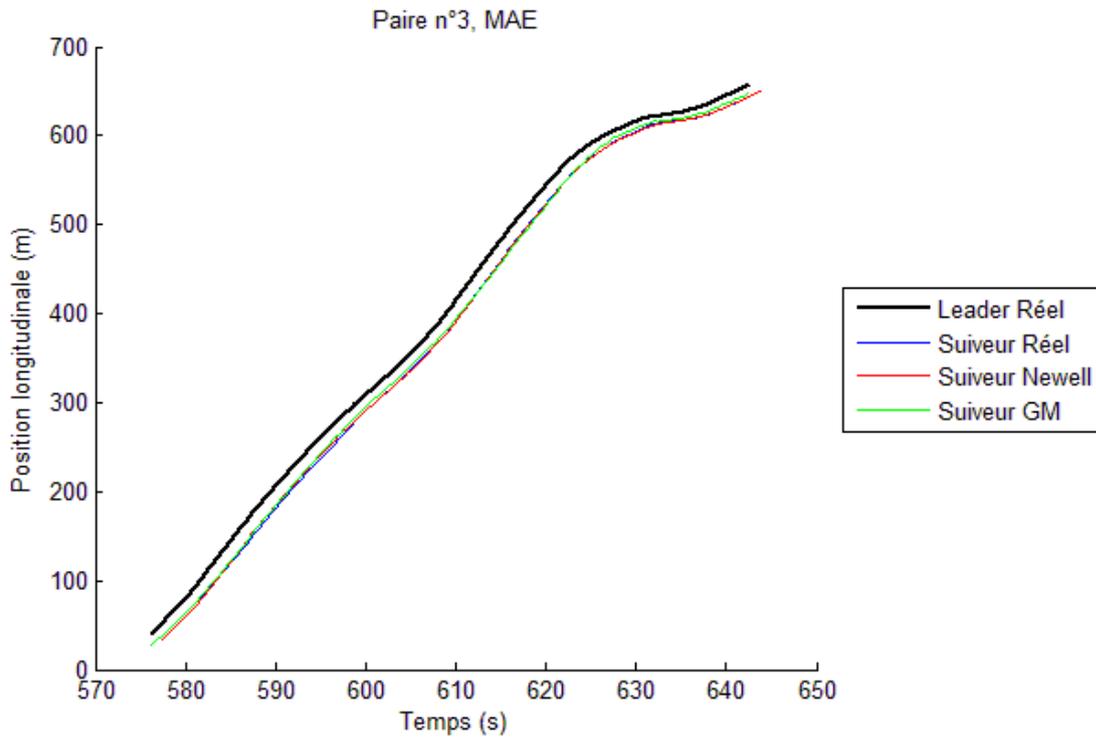


Figure 17 : Trajectoires (x,t) pour la paire 3 avec le GoF MAE

Conclusion

Ce TP de MOTUS est bien différent du TP de MOTUS sur la génération des déplacements. En effet, ce TP représente plus une approche mathématique analytique qu'une approche probabiliste. De plus, ce TP s'attache à la modélisation du trafic et non aux comportements des mobilités. Le cours de MOTUS permet donc d'avoir plusieurs points de vue dans la modélisation des transports.

Pour revenir plus particulièrement sur ce TP, il nous a permis de mieux comprendre le fonctionnement de certains modèles de trafic et de nous familiariser encore un peu plus avec l'outil informatique qu'est MatLab tout en restant sur du concret et sur une expérience réelle.

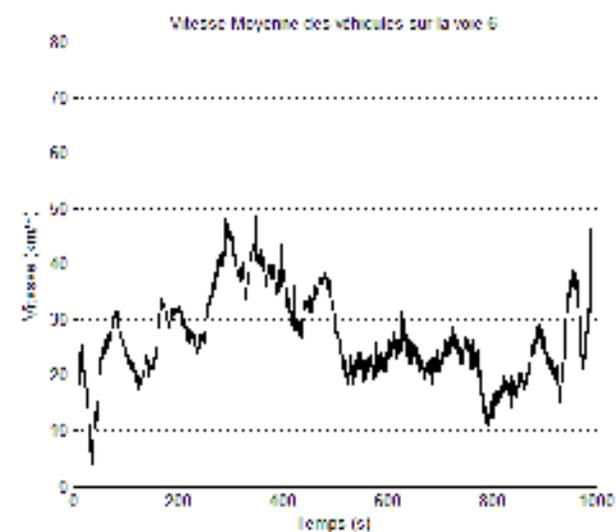
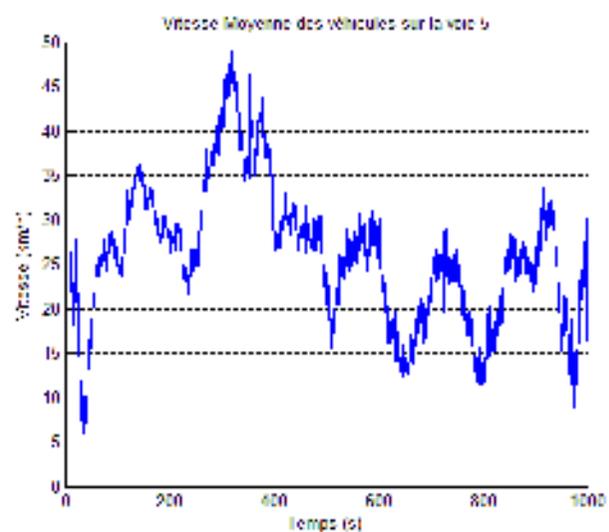
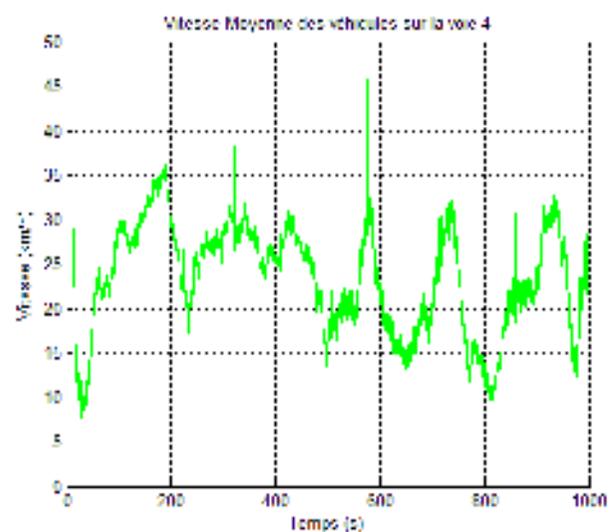
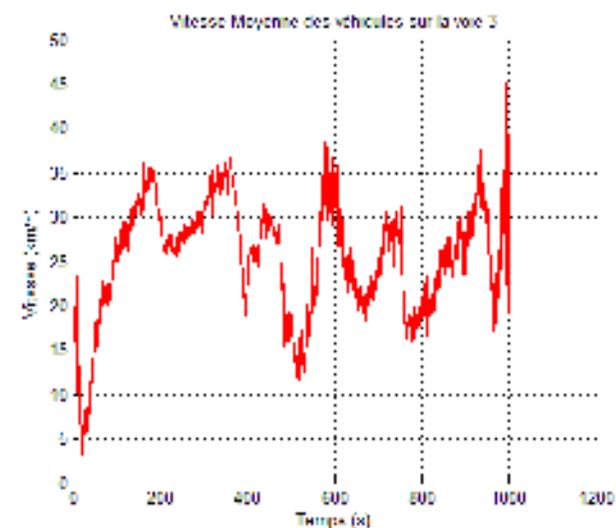
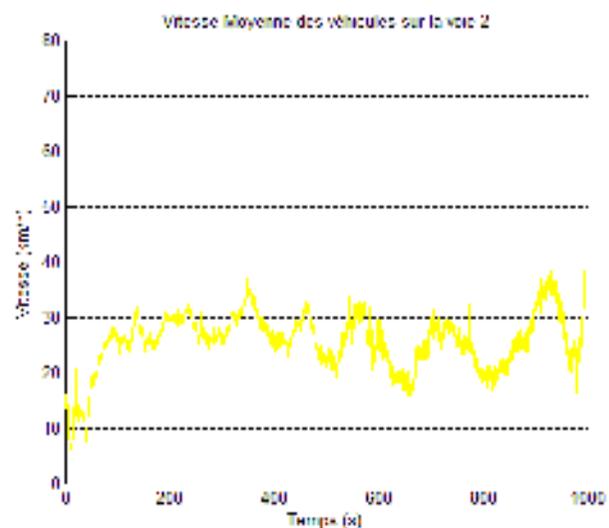
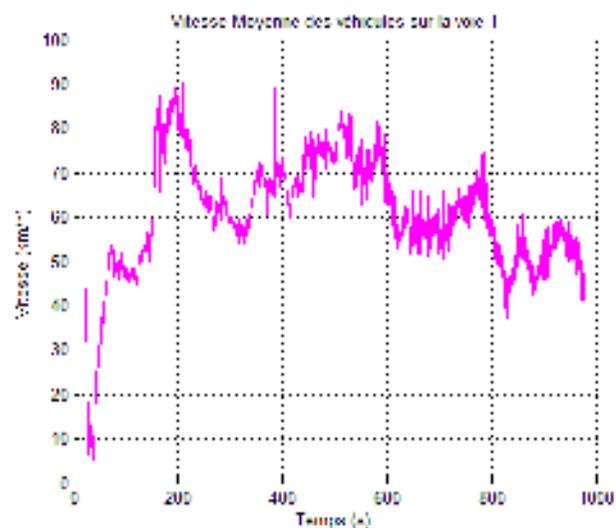
Passons maintenant à l'analyse de la méthode utilisée. A la première lecture du sujet, on pense que l'on va déterminer des paramètres permettant de caler les modèles dans toutes les situations. Cependant, la troisième partie nous affirme le contraire. En effet, le fait que les paramètres optimaux soit différents pour chaque véhicules démontre le fait que chaque individu est différent lorsqu'il conduit. Ceci est vrai tant au niveau psychomoteur par la différence des résultats en terme de temps de réaction, qu'au niveau comportemental par la différence en terme de distance de sécurité adoptée.

N'existe-t-il vraiment pas de modèle, non découverts, permettant de décrire le trafic en toute situation ?

En Annexe 4, se trouve un code permettant d'afficher image par image le trafic et la moyenne des vitesses instantanées des véhicules sur la voie 1 (réservée) et la voie 5 (banale).

ANNEXES

ANNEXE 1 : Vitesses moyennes des véhicules sur chaque voie en fonction du temps et code permettant d'obtenir le rendu



CODE :

```
%-----  
% Script d'affichage de la moyenne des vitesses instantannées sur chaque  
% voie  
% Aurélien CLAIRAIS - mars 2014  
%-----  
  
clear all  
close all  
  
%Récupération des données  
load Data.mat ;  
  
%Mise en forme debug (des problèmes en laissant comme initialement  
Data(:,2)=floor(Data(:,2)*10);  
  
%Palette de couleur  
Color(1)='m';  
Color(2)='y';  
Color(3)='r';  
Color(4)='g';  
Color(5)='b';  
Color(6)='k';  
Color(7)='c';  
  
%MISE EN PLACE DE LA FENETRE GRAPHIQUE  
fig = figure;  
%Plein écran  
set(fig, 'Units', 'Normalized', 'Position', [0 0 1 1]);  
%Fenêtre des voies  
for i=1:6  
    subplot(2,3,i);  
    hold on;  
    grid on;  
    TITRE=sprintf('Vitesse Moyenne des véhicules sur la voie %d',i)  
    title(TITRE)  
    xlabel('Temps (s)')  
    ylabel('Vitesse (km/h)')  
end  
  
% INITIALISATION DES VARIABLES  
for i=1:6  
  
    DONNEES(i).Vehicules=[];  
    DONNEES(i).Vehicules2=[];  
    DONNEES(i).VitesseMoyenne=[];  
    DONNEES(i).Vt =[];  
  
End  
  
% La grosse partie Calcul...  
for k=min(Data(:,2)):max(Data(:,2))  
    %Sélection des données utiles  
    k/100  
    Lignes=find(Data(:,2)==k);  
    Donnees_Considerees=Data(Lignes,:);  
  
    %Récupération des vitesses moyenne des véhicules en présence sur chaque  
    %voie (sauf la voie d'insertion)
```

```

for i=1:6
    Vitesses=[];

DONNEES(i).Vehicules2=Donnees_Considerees(find(Donnees_Considerees(:,5)==i),[1,4]);
% A la première itération, la voie est vide au départ...
if isempty(DONNEES(i).Vehicules)
else
    for j=1:length(DONNEES(i).Vehicules)
        % SI les véhicules présents à l'étape d'avant ne sont plus
        % là...
        if isempty(find(DONNEES(i).Vehicules2(:,1)==DONNEES(i).Vehicules(j)))
        else
            %Sinon

Vitesses=[Vitesses,(DONNEES(i).Vehicules2(find(DONNEES(i).Vehicules2(:,1)==DONNEES(i).V
ehicules(j,1)),2)-DONNEES(i).Vehicules(j,2))*10];
                end
            end
        end

        % Si on a pas calculé de vitesses pour les véhicules , faute de
        % véhicules, on ne calcule pas de vitesse moyenne et l'instant est
        % abandonné
        if isempty(Vitesses)
        else

DONNEES(i).VitesseMoyenne=[DONNEES(i).VitesseMoyenne,sum(Vitesses)/length(Vitesses)*3.6
];
            DONNEES(i).Vt=[DONNEES(i).Vt,k/10];
        end

        %Préparation de l'instant suivant...

DONNEES(i).Vehicules=Donnees_Considerees(find(Donnees_Considerees(:,5)==i),[1,4]);

    end

end
% On affiche sur chaque graphique les vitesses instantannées moyennes
% calculées
for i=1:6
    subplot(2,3,i);
    plot(DONNEES(i).Vt,DONNEES(i).VitesseMoyenne,'Color',Color(i))
    AXES=axis;
    axis([0,AXES(2),0,AXES(4)]);
end

```

ANNEXE 2 : *Partie3a.m* modifiée

```
%% Partie 3 : calage des deux modèles de poursuite

close all
load Vos13Traj.mat
%% On a sélectionné pour vous 13 paires de véhicules qui se suivent
% dans toute la traversée de la section (sans changement de voie ni
% ni du leader ni du suiveur).

% choix de la paire de vehicules parmi les 13
for i = 1:13;
    leader_exp = a(i).leader_exp;
    suiveur_exp = a(i).suiveur_exp;

    % bornes d'évolution des paramètres de Newell
    sxmin = 1; % Valeur minimale de la distance entre deux avants
    % successifs de voiture (en mètres !)
    sxmax = 20; % valeur maximale
    taumin = 0.5; % temps de réaction minimal (en secondes)
    taumax = 1.5; % temps de réaction maximal (idem)

    % bornes d'évolution des paramètres de General Motors
    Cmin = 1; %Valeur minimale du coefficient de sensibilité (en m/s)
    Cmax = 20; %Valeur maximale (en m/s)
    Tmin = 0.5; %temps de réaction minimal (en secondes)
    Tmax = 1.5; %temps de réaction maxiaml (idem)

    NbPas = 20;
    SVect=[sxmin:(sxmax-sxmin)/NbPas:sxmax]; % creation d'un vecteur entre
    % la valer min et la valeur max
    TauVect=[taumin:(taumax-taumin)/NbPas:taumax]; % idem
    CVect=[Cmin:(Cmax-Cmin)/NbPas:Cmax]; %idem
    TVect=[Tmin:(Tmax-Tmin)/NbPas:Tmax]; %idem

    % faisons une exploration sur 21 cases sur S et 21 cases sur T pour trouver
    % les valeurs des écarts (GoF) entre le MoP (ici X) simulé et observé.
    for s=1:length(SVect)
        for t=1:length(TauVect)
            S=SVect(s); % Dans le tableau on prend la valeur du paramètre qui
            T=TauVect(t);
            W=S/T;
            suiveur_Mod = Newell(leader_exp(1:2,:),S,W);
            suiveur_simu = suiveur_Mod.suiveur_mod;
            Z1(s,t)=rmse_traj(suiveur_exp,suiveur_simu); % comme GoF on prend le RMSE
        end;
    end;

    % faisons une exploration sur 21 cases sur C et 21 cases sur T pour trouver
    % les valeurs des écarts (GoF) entre le MoP (ici X) simulé et observé.
    for c=1:length(CVect)
        for t=1:length(TVect)
            C=CVect(c); % Dans le tableau on prend la valeur du paramètre qui
            T=TVect(t);
            suiveur_Mod = GMMModel(leader_exp(1:3,:),suiveur_exp(1:3,:),C,T);
            suiveur_simu = suiveur_Mod;
            Z2(c,t)=rmse_traj(suiveur_exp,suiveur_simu); % comme GoF on prend le RMSE
        end;
    end;

    %% Si Z1 est saturé par des valeurs trop élevées, il faut supprimer tout
    % ce qui supérieur à un seuil.
    Seuil1 = 15;
    Z1=min(Seuil1.*ones(length(SVect),length(TauVect)),Z1);
```

```

%% Si Z2 est saturé par des valeurs trop élevées, il faut supprimer tout
% ce qui supérieur à un seuil.
Seuil2 = 15;
Z2=min(Seuil2.*ones(length(CVect),length(TVect)),Z2);
%%

% Quelles sont les valeurs de paramètres optimales ?
MinMin1=min(min(Z1));
[a1(i),b1(i)]=find(Z1==MinMin1);

MinMin2=min(min(Z2));
[a2(i),b2(i)]=find(Z2==MinMin2);

%Tracé de la fonction de coût pour le modèle de Newell
figure;hold on
title(['Comportement de la paire n°', num2str(i),', Modèle de Newell, RMSE'])
if(1)
    surf(SVect,TauVect,Z1) % ensuite on peut utiliser surf, mais certains préfèrent
mesh.
else
    mesh(SVect,TauVect,Z1)
end
colorbar
ylabel('Valeurs de l''espacement à l''arrêt (m)');
xlabel('Valeurs de temps de réaction (s)');

%Tracé de la fonction de coût pour le modèle de General Motors
figure;hold on
title(['Comportement de la paire n°', num2str(i),', Modèle de General Motors,
RMSE'])
if(1)
    surf(CVect,TVect,Z2) % ensuite on peut utiliser surf, mais certains préfèrent
mesh.
else
    mesh(CVect,TVect,Z2)
end
colorbar
ylabel('Valeurs de la sensibilité C (m/s)');
xlabel('Valeurs de temps de réaction (s)');

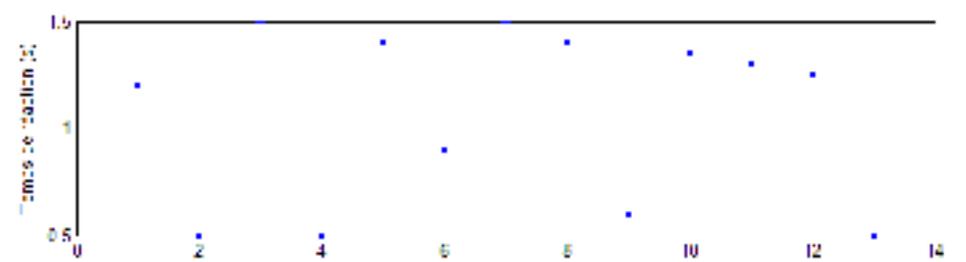
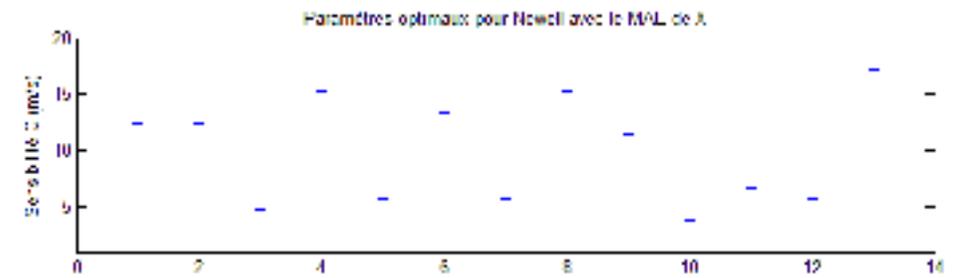
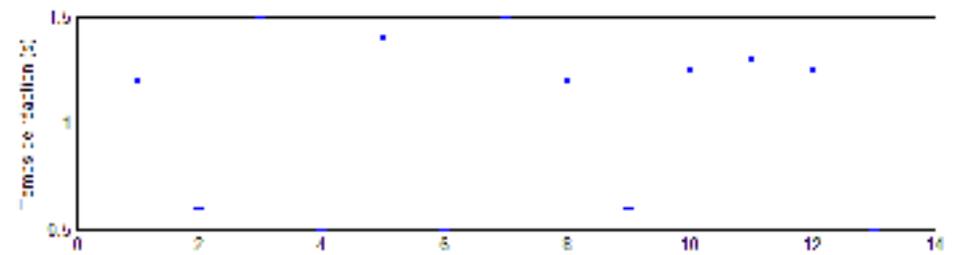
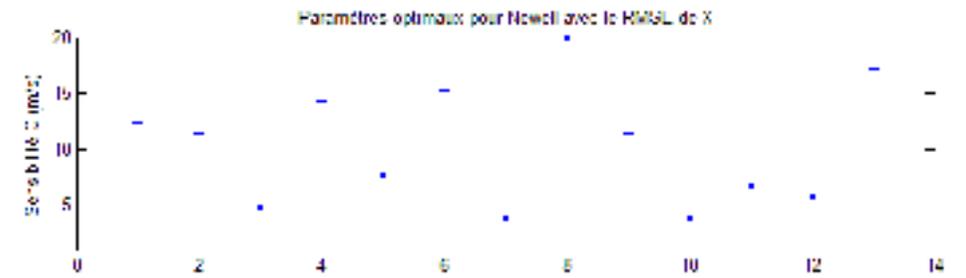
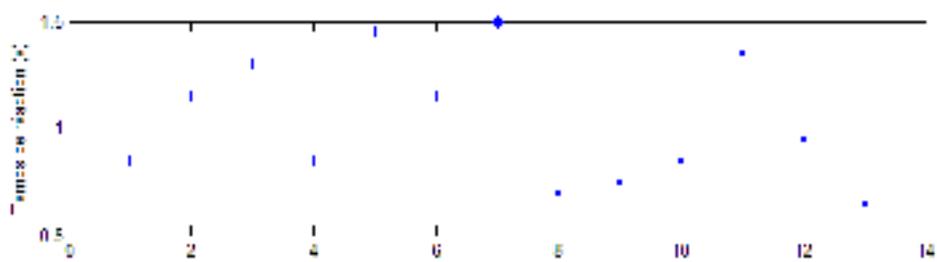
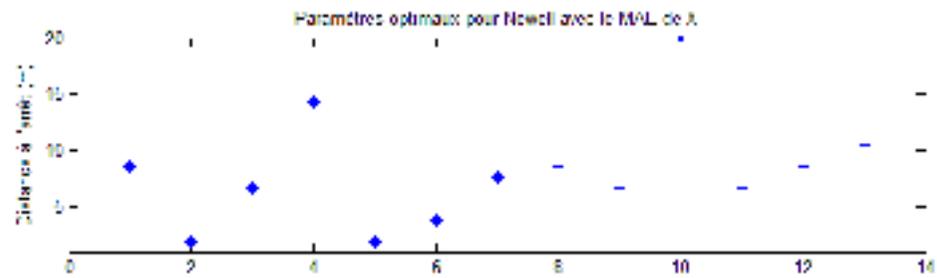
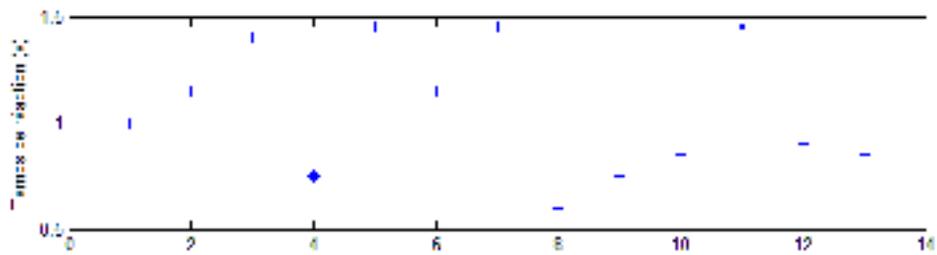
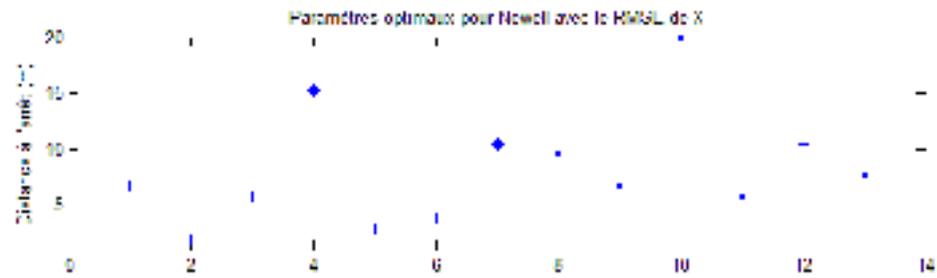
%% Dessin des trajectoires du suiveurs avec les paramètres optimaux.
figure;
hold on;
leader_exp = a(i).leader_exp;
suiveur_exp = a(i).suiveur_exp;
plot(leader_exp(1,:),leader_exp(2:,:), 'k', 'LineWidth',2);
plot(suiveur_exp(1,:),suiveur_exp(2:,:), 'b');
title(['Paire n°', num2str(i),', RMSE'])

b=Newell(leader_exp(1:2,:),SVect(a1(i)),SVect(a1(i))./TauVect(b1(i)))
gm=GMMModel(leader_exp(1:3,:),suiveur_exp(1:3,:),CVect(a2(i)),TVect(b2(i)))
suiveur_Ne(1:2,1:length(b.suiveur_mod(1,:))) = b.suiveur_mod;
plot(suiveur_Ne(1,:),suiveur_Ne(2:,:), 'r');
plot(gm(1,:),gm(2:,:), 'g')

legend('Leader Réel','Suiveur Réel','Suiveur Newell','Suiveur
GM','Location','EastOutside');
xlabel('Temps (s)')
ylabel('Position longitudinale (m)')
end

```

ANNEXE 3 : Bilan des paramètres par paires et *Partie3b.m* modifié permettant d'obtenir ce rendu



CODE :

```
%%% Exo de calibrage sur les deux modèles
load Vos13Traj.mat
%% On a sélectionné pour vous 13 paires de véhicules qui se suivent
% dans toute la traversée de la section (sans changement de voie ni
% ni du leader ni du suiveur).

%% Partie 3 : calage des deux modèles de poursuite
% on exécute pour toutes les 13 paires de vehicules
for i=1:13
    leader_exp = a(i).leader_exp;
    suiveur_exp = a(i).suiveur_exp;

    % bornes d'évolution des paramètres de Newell
    sxmin = 1; % Valeur minimale de la distance entre deux avants
    % successifs de voiture (en mètres !)
    sxmax = 20; % valeur maximale
    taumin = 0.5; % temps de réaction minimal (en secondes)
    taumax = 1.5; % temps de réaction maximal (idem)

    % bornes d'évolution des paramètres de General Motors
    Cmin = 1; %Valeur minimale du coefficient de sensibilité (en m/s)
    Cmax = 20; %Valeur maximale (en m/s)
    Tmin = 0.5; %temps de réaction minimal (en secondes)
    Tmax = 1.5; %temps de réaction maxiaml (idem)

    NbPas = 20;
    SVect=[sxmin:(sxmax-sxmin)/NbPas:sxmax]; % creation d'un vecteur entre
    % la valer min et la valeur max
    TauVect=[taumin:(taumax-taumin)/NbPas:taumax]; % idem
    CVect=[Cmin:(Cmax-Cmin)/NbPas:Cmax]; %idem
    TVect=[Tmin:(Tmax-Tmin)/NbPas:Tmax]; %idem

    % faisons une exploration sur 21 cases sur S et 21 cases sur T pour trouver
    % les valeurs des écarts (GoF) entre le MoP (ici X) simulé et observé.
    for s=1:length(SVect)
        for t=1:length(TauVect)
            S=SVect(s); % Dans le tableau on prend la valeur du paramètre qui
            T=TauVect(t);
            W=S/T;
            suiveur_Mod = Newell(leader_exp(1:2,:),S,W);
            suiveur_simu = suiveur_Mod.suiveur_mod;
            Z1(s,t)=rmse_traj(suiveur_exp,suiveur_simu); % comme GoF on prend le RMSE
            Z12(s,t)=mae_traj(suiveur_exp,suiveur_simu); % comme GoF on prend le MAE
        end;
    end;

    % faisons une exploration sur 21 cases sur C et 21 cases sur T pour trouver
    % les valeurs des écarts (GoF) entre le MoP (ici X) simulé et observé.
    for c=1:length(CVect)
        for t=1:length(TVect)
            C=CVect(c); % Dans le tableau on prend la valeur du paramètre qui
            T=TVect(t);
            suiveur_Mod = GMMModel(leader_exp(1:3,:),suiveur_exp(1:3,:),C,T);
            suiveur_simu = suiveur_Mod;
            Z2(c,t)=rmse_traj(suiveur_exp,suiveur_simu); % comme GoF on prend le RMSE
            Z22(c,t)=mae_traj(suiveur_exp,suiveur_simu); % comme GoF on prend le MAE
        end;
    end;

    % Quelles sont les valeurs de paramètres optimales ?
    MinMin1=min(min(Z1));
    [a1(i),b1(i)]=find(Z1==MinMin1);
    MinMin12=min(min(Z12));
```

```

[c1(i),d1(i)]=find(Z12==MinMin12);
MinMin2=min(min(Z2));
[a2(i),b2(i)]=find(Z2==MinMin2);
MinMin22=min(min(Z22));
[c2(i),d2(i)]=find(Z22==MinMin22);

clear b leader_exp suiveur_exp suiveur_Ne
% clear suiveur_GM suiveur_GMV2
end;

% Dans les tableaux a1 et b1, il y a l'ensemble des indices dans les
% tableaux SVect et TauVect. Pour avoir les valeurs numériques des
% paramètres correspondants, il faut se reporter à la valeur dans le
% tableau.
fig = figure;
%PLEIN ECRAN
set(fig, 'Units', 'Normalized', 'Position', [0 0 1 1]);

%Affichage Newell RMSE
subplot(4,2,1)
plot(SVect(a1(:)), '*');
axis([ 0 14 SVect(1) SVect(end)]);
title('Paramètres optimaux pour Newell avec le RMSE de X')
ylabel('Distance à l''arrêt (m)');
subplot(4,2,3)
plot(TauVect(b1(:)), '*');
axis([0 14 TauVect(1) TauVect(end)]);
ylabel('Temps de réaction (s)');

%Affichage Newell MAE
subplot(4,2,5)
plot(SVect(c1(:)), '*');
axis([ 0 14 SVect(1) SVect(end)]);
title('Paramètres optimaux pour Newell avec le MAE de X')
ylabel('Distance à l''arrêt (m)');
subplot(4,2,7)
plot(TauVect(d1(:)), '*');
axis([0 14 TauVect(1) TauVect(end)]);
ylabel('Temps de réaction (s)');

%Affichage GM RMSE
subplot(4,2,2)
plot(CVect(a2(:)), '*');
axis([ 0 14 CVect(1) CVect(end)]);
title('Paramètres optimaux pour Newell avec le RMSE de X')
ylabel('Sensibilité C (m/s)');
subplot(4,2,4)
plot(TVect(b2(:)), '*');
axis([0 14 TVect(1) TVect(end)]);
ylabel('Temps de réaction (s)');

%Affichage GM MAE
subplot(4,2,6)
plot(CVect(c2(:)), '*');
axis([ 0 14 CVect(1) CVect(end)]);
title('Paramètres optimaux pour Newell avec le MAE de X')
ylabel('Sensibilité C (m/s)');
subplot(4,2,8)
plot(TVect(d2(:)), '*');
axis([0 14 TVect(1) TVect(end)]);
ylabel('Temps de réaction (s)');

```

ANNEXE 4 : Script permettant d'afficher le trafic de manière dynamique et rendu graphique (pouvant être obtenu par copier/coller du script)

```
%-----  
% Script d'affichage dynamique du trafic  
% Aurélien CLAIRAIS - mars 2014  
% Ce script nécessite la variable Data.mat  
%-----  
clear all  
close all  
%Récupération des données  
load Data.mat ;  
  
%Mise en forme debug  
Data(:,2)=floor(Data(:,2)*10);  
  
%Palette de couleur  
Color(1)='m';  
Color(2)='y';  
Color(3)='r';  
Color(4)='g';  
Color(5)='b';  
Color(6)='k';  
Color(7)='c';  
  
%MISE EN PLACE DE LA FENETRE GRAPHIQUE  
fig = figure  
annotation('textbox','Units','normalize','FontUnits','normalized',...  
          'FontSize',0.012,'Position',[0.002 0.002 0.1 0.1],...  
          'String','Aurélien CLAIRAIS mars 2014','LineStyle','none')  
%Plein écran  
set(fig, 'Units', 'Normalized', 'Position', [0 0 1 1]);  
%Fenêtre du trafic  
subplot(2,3,[1,2,4,5]);  
hold on;  
axis([20 500 -30 0]);  
  
%Axes de la voie 1  
subplot(2,3,3);  
grid on  
title('V Moyenne des véhicules sur la voie 1')  
xlabel('Temps (s)')  
ylabel('V Moy (m/s)')  
%Axes de la voie 5  
subplot(2,3,6);  
grid on  
title('V Moyenne des véhicules sur la voie 5')  
xlabel('Temps (s)')  
ylabel('V Moy (m/s)')  
  
% INITIALISATION DES VARIABLES  
VIDES=[];  
Vehicules1=[];  
Vehicules5=[];  
VitesseMoyenne1=[];  
VitesseMoyenne5=[];  
Vt1 =[];  
Vt5=[];  
image=[];  
M=[];
```

```

for i=min(Data(:,2)):max(Data(:,2))
    %Sélection des données utiles
    Lignes=find(Data(:,2)==i);
    %(DEBUGGING) n'est plus utile
    if isempty(Lignes)
        VIDES=[VIDES,i];
    end
    Donnees_Considerees=Data(Lignes,:);

    %Affichage du temps
    subplot(2,3,[1,2,4,5]);
    Minutes=floor(i/600);
    Secondes=floor(i/10-Minutes*60);
    Centiemes=(i/10-Minutes*60-Secondes)*100;
    Temps=sprintf('Trafic à Temps = %5.0f' '%3.0f' '%3.0f',...
        Minutes,Secondes,Centiemes);
    title(Temps);

    %Affichage du trafic
    subplot(2,3,[1,2,4,5]);
    cla;
    %Seul les véhicules s'une même voie ont la même couleur
    for v=1:7
        VoieX=Donnees_Considerees(find(Donnees_Considerees(:,5)==v),4);
        VoieY=floor(-Donnees_Considerees(find(Donnees_Considerees(:,5)==v),3));
        plot(Donnees_Considerees(find(Donnees_Considerees(:,5)==v),4),...
            -Donnees_Considerees(find(Donnees_Considerees(:,5)==v),3),...
            'o','MarkerFaceColor',Color(v));
    end

    %Affichage de la vitesse moyenne sur la voie 1
    Vitesses=[];
    Vehicules12=Donnees_Considerees(find(Donnees_Considerees(:,5)==1),[1,4]);
    % Au début, il y avait aucun véhicules présents avant : pas de calcul
    % de vitesses
    if isempty(Vehicules1)
    else
        for j=1:length(Vehicules1)
            %Si le véhicule correspondant est sortit du champs de vision,
            %on ne le considère plus
            if isempty(find(Vehicules12(:,1)==Vehicules1(j)))
            else
                Vitesses=[Vitesses,(Vehicules12(find(Vehicules12(:,1)==Vehicules1(j,1)),2)-
                    Vehicules1(j,2))*10];
            end
        end
    end

    if isempty(Vitesses)
    else
        VitesseMoyenne1=[VitesseMoyenne1,sum(Vitesses)/length(Vitesses)];
        Vt1=[Vt1,i/10];
        subplot(2,3,3);
        cla;
        plot(Vt1,VitesseMoyenne1,'m');
        grid on
        title('V Moyenne des véhicules sur la voie 1')
        xlabel('Temps (s)')
        ylabel('V Moy (m/s)')
    end
end

```

```

        AXES=axis;
        axis([0 AXES(2) 0 AXES(4)]);
end

%Etape suivante
Vehicules1=Donnees_Considerees(find(Donnees_Considerees(:,5)==1),[1,4]);
%     pause(0.005)%0.1/0.85);

%Affichage de la vitesse moyenne sur la voie 5
Vitesses=[];
Vehicules52=Donnees_Considerees(find(Donnees_Considerees(:,5)==5),[1,4]);
% Au début, il y avait aucun véhicules présents avant : pas de calcul
% de vitesses
if isempty(Vehicules5)
else
    for j=1:length(Vehicules5)
        %Si le véhicule correspondant est sortit du champs de vision,
        %on ne le considère plus
        if isempty(find(Vehicules52(:,1)==Vehicules5(j)))
            else
Vitesses=[Vitesses,(Vehicules52(find(Vehicules52(:,1)==Vehicules5(j,1)),2)-
Vehicules5(j,2))*10];
                end
            end
        end

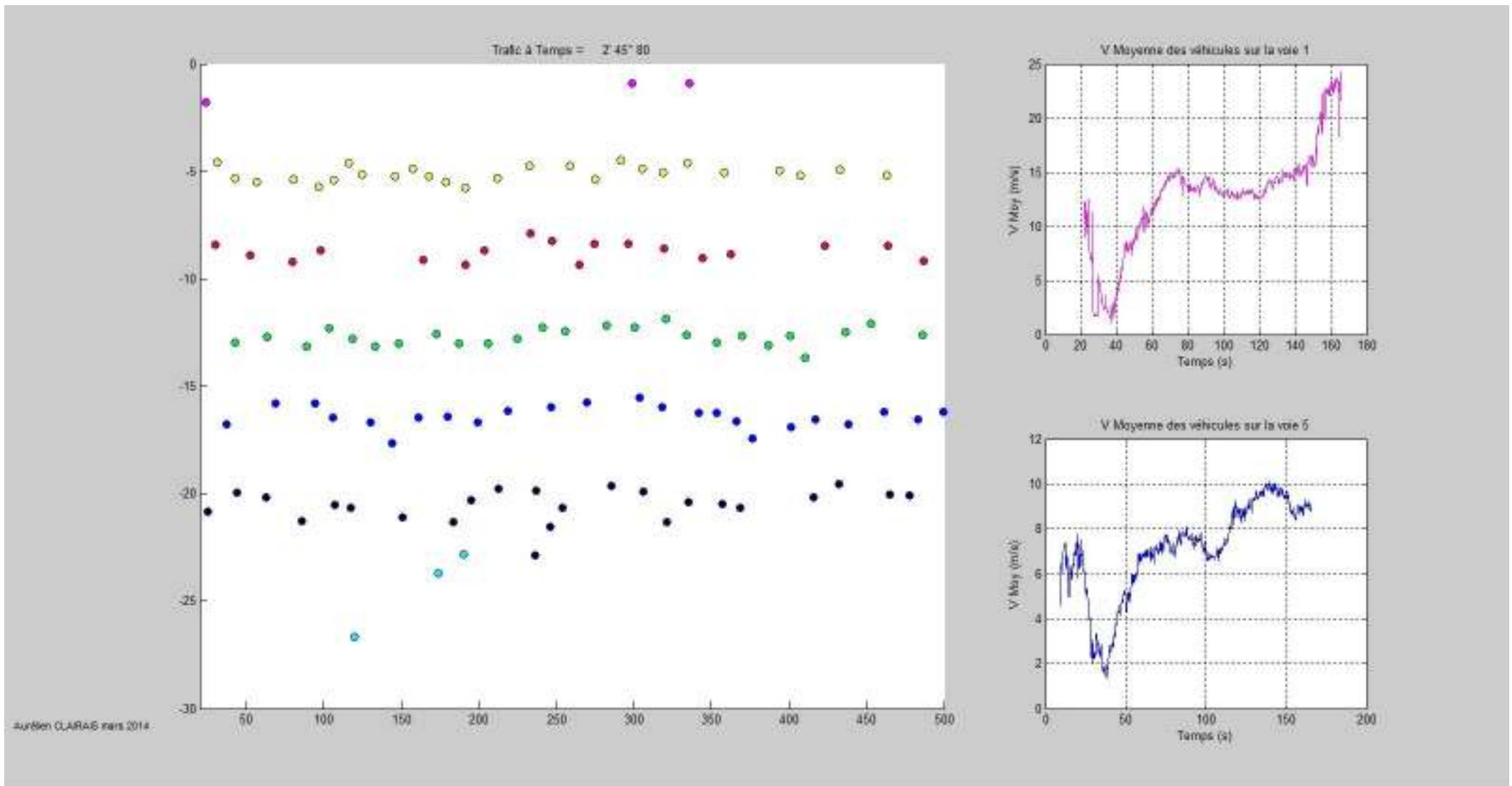
if isempty(Vitesses)
else
    VitesseMoyenne5=[VitesseMoyenne5,sum(Vitesses)/length(Vitesses)];
    Vt5=[Vt5,i/10];
    subplot(2,3,6);
    cla;
    plot(Vt5,VitesseMoyenne5,'b');
    grid on;
    title('V Moyenne des véhicules sur la voie 5')
    xlabel('Temps (s)')
    ylabel('V Moy (m/s)')
    AXES=axis;
    axis([0 AXES(2) 0 AXES(4)]);
end

%Préparation de l'étape suivante
Vehicules5=Donnees_Considerees(find(Donnees_Considerees(:,5)==5),[1,4]);

%AFICHAGE DE L'ANIMATION ?
pause(0.01)%0.1/0.85);

%Dans le cas où on veut enregistrer image par image les figures
%     image=getframe(fig);
%     nom_image=sprintf('MOTUS_VIDEO\\Trafic_%d.jpg',i);
%     imwrite(image.cdata,nom_image,'jpg','Quality',100);
end

```



Je recherche toujours le moyen d'assembler mes 10 000 images pour faire une vidéo « temps réel ».