

Modélisation du trafic routier

Camara Saran

Fofana Nango

Mathilde Thollet

Université Versailles Saint Quentin-en-Yvelines

Mai 2019



Table des matières

| | | |
|------------|----------------------------------------------------------|-----------|
| I | Introduction | 3 |
| II | Modèles Macroscopiques | 4 |
| 1 | Le modèle Lighthill Whitham et Richards (LWR) | 4 |
| 2 | Résolution Analytique de l'équation du modèle LWR | 5 |
| 3 | Résolution Numérique de l'équation du modèle LWR | 7 |
| 3.1 | Feu tricolore | 7 |
| 3.2 | Ralentissement | 9 |
| 3.3 | Feu tricolore et travaux | 10 |
| 3.4 | Comparaison avec la solution exacte | 13 |
| III | Modèles Microscopiques | 14 |
| 1 | Modèles à vitesses optimales | 14 |
| 1.1 | Modèle de Newell | 14 |
| 1.2 | Modèle de Bando | 14 |
| 1.3 | Modèle de Chandler | 14 |
| 2 | Modèles à distance de sécurité | 15 |
| 2.1 | Les modèles simples de Pipse | 15 |
| 2.2 | Les modèles de collision (AC) | 15 |
| 2.3 | Les Travaux de Gibbs | 15 |
| 3 | Modèles des automates cellulaires | 16 |
| 4 | Exemple d'analyse d'un modèle microscopique | 17 |
| 4.1 | Présentation d'un modèle | 17 |
| 4.2 | Passage du modèle de chandler au modèle de LWR | 18 |
| IV | Conclusion | 19 |

Première partie

Introduction

Depuis la naissance des premiers véhicules, le trafic routier s'est énormément développé. Pour le gérer, il a fallu comprendre et modéliser ses phénomènes. Aujourd'hui, il fait entièrement parti du quotidien et il est utile de comprendre son fonctionnement pour trouver des solutions afin de le rendre plus optimal.

Notre projet portera donc sur l'étude de la modélisation du trafic routier en utilisant différentes méthodes : le point de vue macroscopique et le point de vue microscopique.

Les modèles macroscopiques sont des modèles qui s'intéressent au problème en considérant le trafic comme un flux de voiture. Ils se basent donc sur des équations de mécanique des fluides qui sont des équations différentielles partielles. Ici nous nous intéresserons au modèle LWR que nous allons résoudre analytiquement et numériquement afin de modéliser et d'observer plusieurs situations du trafic.

Les modèles microscopiques étudient le problème en s'intéressant à l'évolution de chaque véhicule. Ceux-ci se basent sur des équations différentielles ordinaires et utilisent plusieurs paramètres pour modéliser les situations. L'objectif est d'arriver à une modélisation microscopique fidèle de l'évolution de ce système afin d'en déduire et de comprendre les paramètres qui influent sur son comportement microscopique.

Ainsi le but est de comprendre comment les bouchons se forment. Quelles conséquences peut-on avoir sur le trafic lorsqu'il arrive des perturbations ?



Deuxième partie

Modèles Macroscopiques

On cherche à comprendre le comportement de la densité des véhicules en partant de plusieurs situations initiales. Dans cette partie, on va donc, à partir du modèle LWR, chercher la solution exacte et ensuite modéliser une solution approchée pour mieux comprendre les phénomènes en fonction des différentes situations sur une route de longueur que l'on définira.

On se placera donc dans la situation où :

- il n'y a pas de dépassement possible
- il n'y pas d'intersection le long de la route
- la vitesse dépend seulement de la densité

On définit la vitesse comme une fonction linéaire en fonction de la densité. Elle est maximale et vaut v_{max} lorsque la densité est nulle, c'est-à-dire dans la situation où la route est libre. La vitesse est nulle lorsque la densité est maximale c'est-à-dire $\rho = \rho_{max}$ ce qui correspond à la situation où la route est saturée.

On peut donc définir la fonction vitesse par :

$$v(\rho) = -\frac{v_{max}}{\rho_{max}}\rho + v_{max}$$

On définit aussi le flux par un polynôme en fonction de la densité, qui peut s'écrire : $q(\rho) =$

$$\rho v(\rho) = \rho v_{max} - \frac{v_{max}}{\rho_{max}}\rho^2$$

1 Le modèle Lighthill Whitham et Richards (LWR)

Le modèle Lighthill Whitham et Richards, en abrégé modèle LWR, est un modèle macroscopique dit du premier ordre qui décrit l'évolution du trafic sur une route. Ce modèle repose sur le fait que le même nombre de véhicules qui entrent dans un intervalle de temps en un point de l'espace ressort par un autre point de l'espace dans le même intervalle de temps. Ce modèle fait intervenir 3 variables qui sont la vitesse v qui désigne la vitesse des véhicules, la densité ρ qui désigne le nombre de véhicule, et le flux q qui correspond aux nombres de véhicule passant en un point dans un intervalle de temps.

En effet la densité ρ , la vitesse v et le flux q sont liées par les lois de la mécanique des fluides. Cette liaison est définie par l'équation suivante :

$$\begin{cases} \partial_t \rho(x, t) + \partial_x q(x, t) = 0 \\ q(x, t) = \rho(x, t)v(x, t) \end{cases} \quad (1)$$

Cette équation est une équation de conservation. En tenant compte des états fondamentaux d'équilibre du trafic, on peut faire l'hypothèse que la vitesse et le flux ne sont qu'en fonction de la densité :

$$v(x, t) = v(\rho(x, t)) \text{ et } q(x, t) = q(\rho(x, t)). \quad (2)$$

Grace à la relation fondamentale d'équilibre du flux $q = v\rho$, de la vitesse de (2) et de la loi des mécaniques des fluides concernant les écoulements de (1), on obtient ainsi notre équation du modèle LWR définie par :

$$\partial_t \rho(x, t) + \partial_x q(\rho(x, t)) = 0. \quad (3)$$

Ce qui donne nous :

$$\partial_t \rho(x, t) + q'(\rho(x, t)) \partial_x \rho(x, t) = 0. \quad (4)$$

2 Résolution Analytique de l'équation du modèle LWR

Soit l'équation du modèle LWR définie par :

$$\partial_t \rho(x, t) + q'(\rho(x, t)) \partial_x \rho(x, t) = 0. \quad (5)$$

avec

$$q(\rho) = \rho v(\rho) = \rho v_m \left(1 - \frac{\rho}{\rho_m}\right)$$

où v_m désigne la vitesse maximale.

Soit la donnée initiale définie par :

$$\rho_0(x) = \begin{cases} a & \text{si } x < 0 \\ b & \text{si } x > 0 \end{cases} \quad \text{avec } a > b$$

On remarque que :

$$q'(\rho) = v_m \left(1 - \frac{2\rho}{\rho_m}\right) \quad \text{donc } a > b \Leftrightarrow q'(a) < q'(b).$$

Résolvons notre équation par la méthode des caractéristiques pour obtenir la solution exacte.

On définit les courbes caractéristiques par l'équation différentielle ordinaire (EDO) suivante :

$$\begin{cases} X'(t) = q'(\rho(X(t), t)) \\ X(t_*) = x_* \end{cases}$$

Etant donné que la solution est constante le long des caractéristiques on a aussi :

$$\begin{cases} X'(t) = q'(\rho_0(X(0))) \\ X(t_*) = x_* \end{cases} \quad (6)$$

Après résolution de l'équation (6), on a :

$$X(t) = x_* + (t - t_*) q'(\rho_0(X(0)))$$

On a

$$\frac{d}{dt} \rho(X(t), t) = \partial_t \rho(x, t) + X'(t) \partial_x \rho(x, t) = \partial_t \rho(x, t) + q'(\rho(X(t), t)) \partial_x \rho(x, t) = 0$$

donc on a :

$$\frac{d}{dt} \rho(X(t), t) = 0 \quad (7)$$

On remarque alors que la fonction densité définie par : $\rho(X(t), t)$ est constante.

En intégrant (7) par rapport au temps entre 0 et t_* on a alors : ,

$$\int_0^{t_*} \frac{d}{dt} \rho(X(t), t) dt = 0 \quad (8)$$

Ce qui équivaut à :

$$\rho(X(t_*), t_*) - \rho_0(X(0)) = 0$$

Par conséquent on a :

$$\rho(X(t_*), t_*) = \rho_0(X(0)). \quad (9)$$

En remplaçant $X(t_*)$ par sa valeur dans (9) on a :

$$\rho(x_*, t_*) = \rho(X(t_*), t_*) = \rho_0(X(0)) \text{ avec } X(0) = x_* - t_* q'(\rho_0(X(0))) = 0$$

D'où :

$$\rho(x_*, t_*) = \rho_0(x_* - t_* q'(\rho(X(0))))$$

Où encore :

$$\rho(x, t) = \rho_0(x - t q'(\rho(X(0)))).$$

Cherchons la valeur de $\rho(x, t)$ dans les cas suivants :

Si $X(0) > 0$ c'est à dire si $x - t q'(a) > 0$ alors $\frac{x}{t} < q'(a)$

D'où $\rho(x, t) = a$.

De même, si $X(0) < 0$ c'est à dire si $x - t q'(a) < 0$ alors $\frac{x}{t} > q'(b)$.

Et dans ce cas $\rho(x, t) = b$.

On remarque que la méthode des caractéristiques nous donne seulement :

$$\rho(x, t) = \begin{cases} a & \text{si } \frac{x}{t} < q'(a) \\ ? & \text{si } q'(a) < \frac{x}{t} < q'(b) \\ b & \text{si } \frac{x}{t} > q'(b) \end{cases}$$

Cherchons la valeur de la solution $\rho(x; t)$ dans le cas où $q'(a) < \frac{x}{t} < q'(b)$.

Pour trouver la valeur de la solution $\rho(x, t)$ lorsque $q'(a) < \frac{x}{t} < q'(b)$, on fait l'hypothèse que la solution $\rho(x, t)$ ne dépend que de la variable $\frac{x}{t}$.

Posons $\frac{x}{t} = \epsilon$ et $\rho(x, t) = \rho_*(\epsilon)$ avec $q'(a) < \epsilon < q'(b)$.

On a :

$$\partial_t \rho + q'(\rho) \partial_x \rho = 0 \Leftrightarrow \frac{-x}{t^2} \rho'_*(\epsilon) + q'(\rho_*(\epsilon)) \frac{1}{t} \rho'_*(\epsilon) = 0 \Leftrightarrow \frac{\rho'_*(\epsilon)}{t} (q'(\rho_*(\epsilon)) - \epsilon) = 0$$

En supposant que $\rho'_*(\epsilon) \neq 0$

On a :

$$q'(\rho_*(\epsilon)) - \epsilon = 0 \Leftrightarrow q'(\rho_*(\epsilon)) = \epsilon \Leftrightarrow v_m \left(1 - \frac{2\rho_*(\epsilon)}{\rho_m}\right) = \epsilon \Leftrightarrow \rho_*(\epsilon) = \frac{\rho_m}{2v_m} (v_m - \epsilon)$$

Ce qui nous donne :

$$\rho(x, t) = \frac{\rho_m}{2v_m} \left(v_m - \frac{x}{t}\right)$$

Finalement la solution exacte est donnée par la formule suivante :

$$\rho(x, t) = \begin{cases} a & \text{si } \frac{x}{t} < q'(a) \\ \frac{\rho_m}{2v_m} \left(v_m - \frac{x}{t}\right) & \text{si } q'(a) < \frac{x}{t} < q'(b) \\ b & \text{si } \frac{x}{t} > q'(b) \end{cases} \quad (10)$$

avec $q'(a) = v_m \left(1 - \frac{2a}{\rho_m}\right)$ et $q'(b) = v_m \left(1 - \frac{2b}{\rho_m}\right)$

3 Résolution Numérique de l'équation du modèle LWR

On applique ensuite la méthode des volumes finis à notre équation de conservation. On cherche à construire une approximation de la solution exacte.

Pour cela, on introduit un maillage sur un intervalle $[a, b]$ formé de cellules $C_j = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$ où $x_{j+\frac{1}{2}} = (j + \frac{1}{2})\Delta x$ et Δx est le pas d'espace choisi. On introduit aussi un pas de temps Δt et à chaque instant $t^n = n\Delta t$, on calcule une approximation moyenne ρ_j^n de la solution sur chaque cellule.

On pose donc :

$$\rho_j^n \approx \frac{1}{\Delta x} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \rho(x, t^n) dx$$

Ensuite, à l'aide d'un schéma, on utilise une formule qui permet de trouver les ρ_j^{n+1} en fonction des ρ_j^n pour connaître toutes les valeurs de l'approximation de ρ , à chaque instant et en chaque point.

Shéma de Godunov

Pour modéliser l'approximation de la solution ρ , nous allons utiliser le schéma de Godunov qui est celui-ci :

$$\rho_j^{n+1} = \rho_j^n - \frac{\Delta t}{\Delta x} (g(\rho_j^n, \rho_{j+1}^n) - g(\rho_{j-1}^n, \rho_j^n)) \text{ où } g(\rho_j^n, \rho_{j+1}^n) = \begin{cases} \min(q(\rho_j^n), q(\rho_{j+1}^n)) & \text{si } \rho_j^n \leq \rho_{j+1}^n \\ q(\rho_{j+1}^n) & \text{si } \frac{\rho_{max}}{2} \leq \rho_{j+1}^n < \rho_j^n \\ q(\frac{\rho_{max}}{2}) & \text{si } \rho_{j+1}^n < \frac{\rho_{max}}{2} < \rho_j^n \\ q(\rho_j^n) & \text{si } \rho_{j+1}^n < \rho_j^n \leq \frac{\rho_{max}}{2} \end{cases}$$

3.1 Feu tricolore

On cherche à modéliser un feu rouge, situé au point $x = \frac{a+b}{2}$: la densité avant le feu rouge est importante, alors que la route est libre après le feu rouge, la densité est donc nulle.

On introduit donc la condition initiale :

$$\rho_0(x) = \begin{cases} \frac{1}{4}\rho_{max} & \text{si } x \leq \frac{(a+b)}{2} \\ 0 & \text{sinon} \end{cases}$$

On va donc modéliser la densité de voiture lorsque le feu passe au vert à l'aide du schéma de Godunov. Nous avons donc implémenter ce programme :

On se place sur un intervalle $[a, b]$, on définit un nombre de points N pour le maillage en espace et T un temps d'arrêt de calcul. On définit le pas en espace Δx par $\frac{b-a}{N}$ et on fixe la condition $\Delta t \leq \frac{\Delta x}{2max|q'(\rho)|}$. Or la fonction q' est une droite définie sur \mathbb{R}^+ et de pente négative.

Donc elle atteint son maximum lorsque $\rho = 0$. La condition devient donc : $\Delta t \leq \frac{\Delta x}{2q'(0)}$.

On fixe ensuite une vitesse maximale et une densité maximale. On définit les fonctions vitesse, flux et dérivée du flux.

Puis on définit le vecteur $x(j) = (a + \frac{\Delta x}{2}) + (j - 1)\Delta x$ pour tout $j = 1, \dots, N$ où chaque coordonnée va être un point auquel on va évaluer l'approximation de ρ .

On fournit la fonction g du schéma de Godunov et la donnée initiale donnée ci-dessus où la donnée à gauche est notée u_g et la donnée à droite est notée u_d .

On introduit une variable temps que l'on initialise à 0.

On définit ensuite deux vecteurs : le vecteur G de taille $N + 1$ qui nous permettra de faciliter l'écriture du schéma lorsque la fonction g interviendra et le vecteur U de taille N qui correspond à la densité à l'instant t en chaque point.

Au temps $t = 0$, on connaît la densité puisqu'elle est égale à la donnée initiale ρ_0 , on peut donc définir le vecteur U à cet instant.

On crée ensuite une boucle qui s'arrêtera lorsque $t = T$. Dans cette boucle, on incrémente donc de Δt le temps t à chaque passage.

On impose des conditions de bord : le vecteur G aura donc en première coordonnée (pour $j = 1$), la donnée $q(u_g)$ et la dernière coordonnée (pour $j = N + 1$), la donnée $q(u_d)$, fixées pour tout temps. Puis on définit le reste des coordonnées de G par la différence : $G(k) = g(U(k-1), U(k))$ pour tout $K = 2, \dots, N$ qui représente donc $g(\rho_{j-1}^n, \rho_j^n)$.

Pour finir, on définit $U(j)$, au temps $n+1$, pour tout $j = 1, \dots, N$, à l'aide de la formule du schéma de Godunov et de la donnée au temps précédent, $U(j)$ devient donc $U(j) - \frac{\Delta t}{\Delta x}(G(j+1) - G(j))$ au temps suivant. Et ainsi de suite jusqu'à ce que $t = T$.

On affiche ensuite la solution approchée au temps $t = T$ et on obtient le graphe suivant :

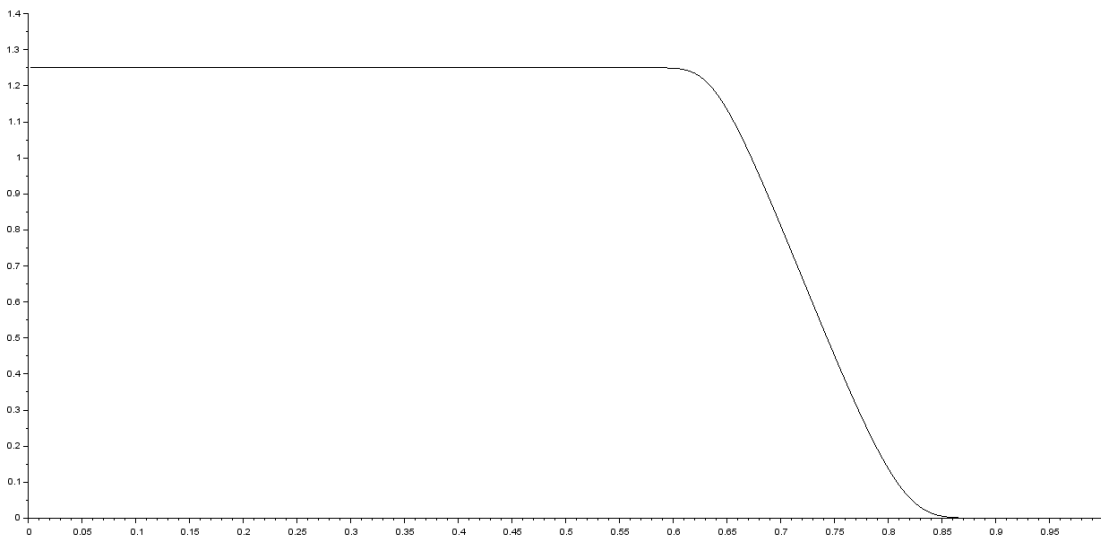


FIGURE 1 – Courbe de la solution approchée dans le cas d'un feu tricolore

Interprétations des résultats :

On remarque sur la figure 1 qu'une onde de détente se propage au cours du temps. Les voitures vont se répartir sur la route et la densité à droite du feu va augmenter. On peut observer sur le graphique que lorsque la densité est nulle, c'est que les voitures n'ont encore pas atteint ce point. De même la portion de courbe qui est égale à la densité maximale va se déplacer sur la droite car il faut un certain temps avant que les voitures se dispersent le long de la route.

3.2 Ralentissement

Petit ralentissement

On cherche à modéliser un petit ralentissement : la densité avant le feu rouge est faible, alors que la densité à partir du milieu de l'intervalle est forte.

On introduit donc la condition initiale :

$$\rho_0(x) = \begin{cases} \frac{1}{8}\rho_{max} & \text{si } x \leq \frac{(a+b)}{2} \\ \frac{1}{2}\rho_{max} & \text{sinon} \end{cases}$$

On va donc modéliser la densité de voiture lorsque l'embouteillage se crée à l'aide du schéma de Godunov. Nous avons donc implémenté un programme à l'aide du précédent, en changeant seulement la donnée initiale.

Nous obtenons donc un graphe pour la solution approchée du problème au temps $t = T$:

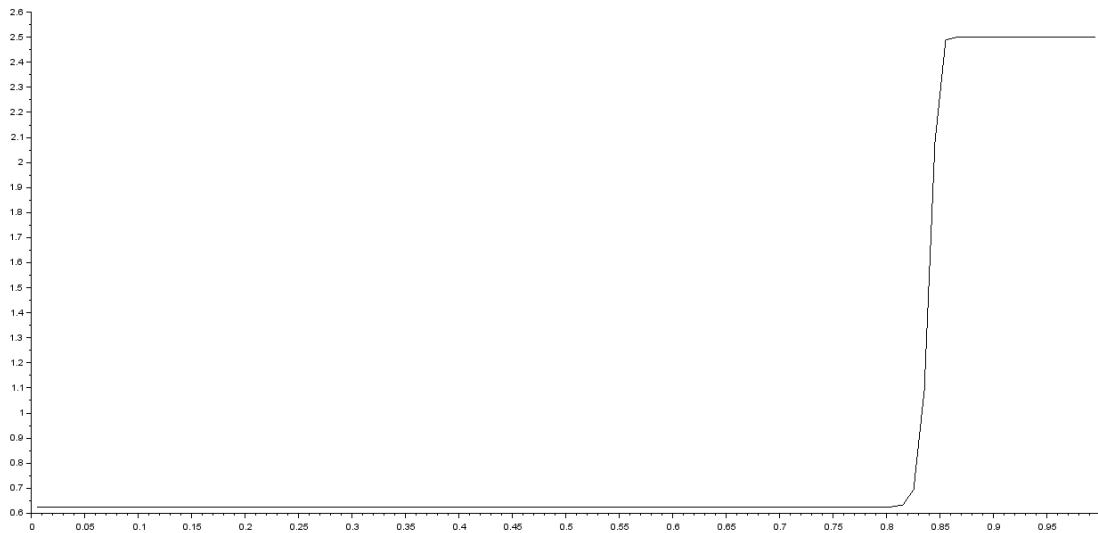


FIGURE 2 – Courbe de la solution approchée dans le cas d'un petit ralentissement

Interprétations des résultats :

On remarque avec la figure 2 que la solution se propage vers la droite au cours du temps. Cela s'explique par le fait que les voitures vont ralentir au cours du temps mais ne vont pas s'arrêter car la densité à droite est de $\frac{1}{2}\rho_{max}$. L'onde de choc va donc se déplacer en même temps que les voitures qui avancent donc le long de la route.

Important ralentissement

On cherche à modéliser un important ralentissement : la densité avant le feu rouge est faible, alors que la densité à partir du milieu de l'intervalle est maximale.

On introduit donc la condition initiale :

$$\rho_0(x) = \begin{cases} \frac{1}{8}\rho_{max} & \text{si } x \leq \frac{(a+b)}{2} \\ \rho_{max} & \text{sinon} \end{cases}$$

On va donc modéliser la densité de voiture lorsque l'embouteillage se crée à l'aide du schéma de Godunov. Nous avons donc implémenté un programme à l'aide du précédent, en changeant seulement la donnée initiale.

Nous obtenons donc un graphe pour la solution approchée du problème à l'instant $t = T$:

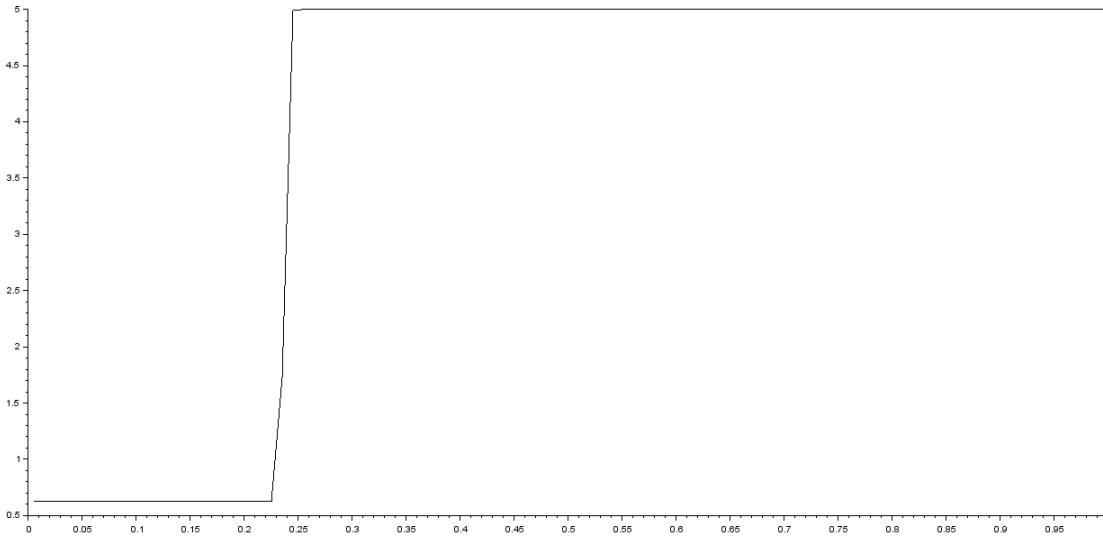


FIGURE 3 – Courbe de la solution approchée dans le cas d'un fort ralentissement

Interprétations des résultats :

Puisqu'ici le ralentissement est fort, c'est-à-dire que la densité est maximale à droite, les voitures vont s'arrêter d'un coup et l'onde de choc va donc se propager vers la gauche, en direction des voitures qui ne sont pas encore arrêtées. C'est bien ce qu'on remarque sur la figure 3.

3.3 Feu tricolore et travaux

On cherche maintenant à modéliser un feu rouge qui passe au vert, situé au point $x = \frac{a+b}{4}$ mais cette fois ci, on modélise aussi des travaux lorsque $j = \frac{3}{4}N$.

On introduit donc la condition initiale :

$$\rho_0(x) = \begin{cases} \frac{1}{5}\rho_{max} & \text{si } x \leq \frac{(a+b)}{4} \\ 0 & \text{sinon} \end{cases}$$

On va modéliser la densité lorsque le feu passe au vert suivi de travaux à un instant qui sont modélisés par une condition sur le flux à l'aide du schéma de Godunov. Nous avons donc implémenté un programme à l'aide du précédent :

Nous changeons donc la donnée initiale car le feu est maintenant placé au niveau du premier quart de la route. Il faut aussi créer une condition de plus, qui oblige, lorsque $j = \frac{3}{4}N$, à prendre le minimum entre le flux et une donnée fixée du flux : ici on choisi $F = 10$. Il va donc se créer une perturbation au moment $j = \frac{3}{4}N$, qui va avoir des conséquences en tout point.

Nous obtenons donc la figure 4 pour la solution approchée du problème à l'instant $T = 0.01$:

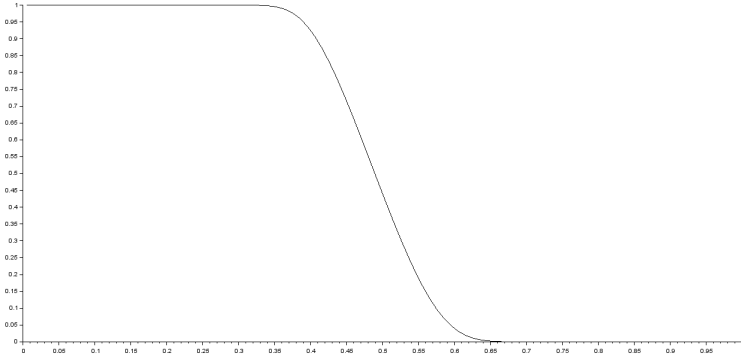


FIGURE 4 – Courbe de la solution approchée dans le cas d'un feu tricolore suivi de travaux avec $T=0.01$

Nous obtenons donc la figure 5 pour la solution approchée du problème à l'instant $T = 0.02$:

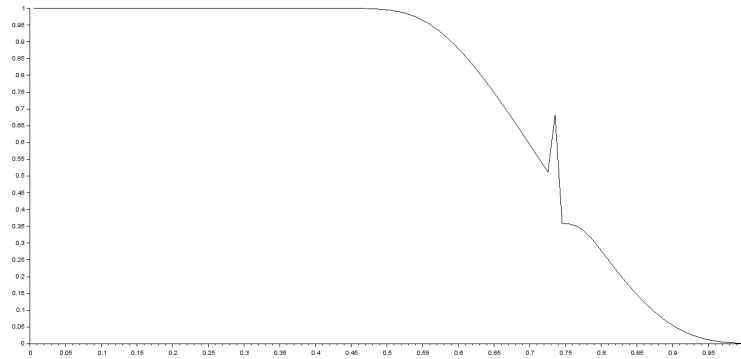


FIGURE 5 – Courbe de la solution approchée dans le cas d'un feu tricolore suivi de travaux avec $T=0.02$

Nous obtenons donc la figure 6 pour la solution approchée du problème à l'instant $T = 0.04$ et la figure 7 pour la solution approchée à l'instant $T = 0.08$:

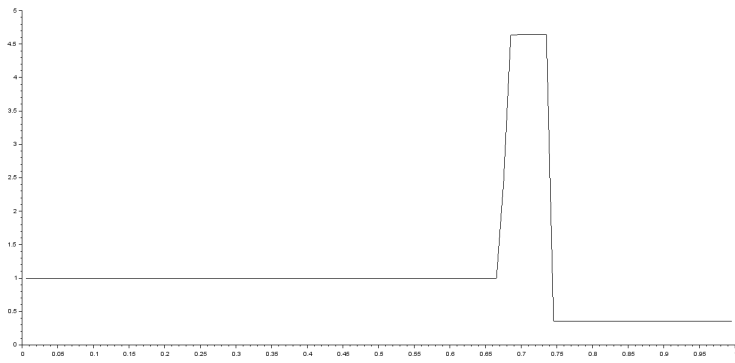


FIGURE 6 – Courbe de la solution approchée dans le cas d'un feu tricolore suivi de travaux avec $T=0.04$

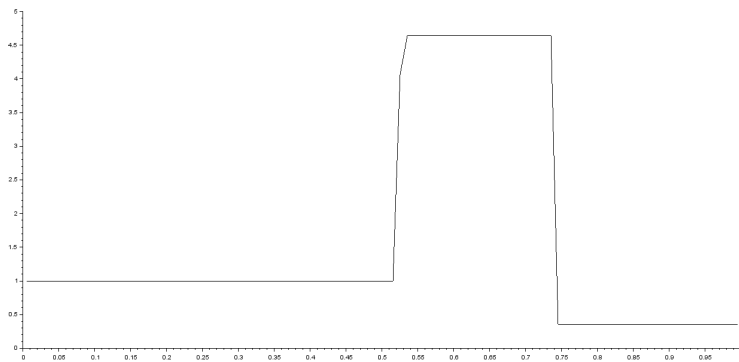


FIGURE 7 – Courbe de la solution approchée dans le cas d'un feu tricolore suivi de travaux avec $T=0.08$

Interprétations des résultats

On utilise le programme pour plusieurs T afin d'observer la situation au cours du temps. On remarque que la courbe de la figure 4 se comporte d'abord comme lors d'un feu tricolore c'est-à-dire comme une onde de détente. Les voitures n'ont pas encore atteint la perturbation. Dans la figure 5 lorsque certaines voitures ont commencé à atteindre les travaux, on observe un pique c'est-à-dire une augmentation rapide de la densité. Un petit embouteillage s'est alors formé.

Dans la figure 6, l'embouteillage s'est alors intensifié. Mais l'on observe quand même que la densité redevient normale ensuite. Ce qui signifie que cette perturbation crée un embouteillage mais les voitures ne sont pas à l'arrêt et passent quand même. Dans la figure 7, on peut alors observer que l'embouteillage se déplace vers la gauche, la densité augmente donc en avant de la route comme dans la situation d'un fort ralentissement. Nous pouvons donc conclure que les travaux ont une influence importante sur la densité car elle impose une condition maximale sur le flux de voitures. Et nous observons que, comme seulement quelques voitures peuvent passer au niveau des travaux, la densité va de plus en plus augmenter vers la gauche et créer un fort embouteillage.

3.4 Comparaison avec la solution exacte

Dans la partie II. (1), nous avons trouvé une écriture de la solution exacte. Dans un programme, nous comparons donc la solution exacte et la solution approchée par le modèle de Godunov dans le cas d'une onde de détente c'est-à-dire dans le cas d'un feu tricolore qui passe au vert.

Nous obtenons donc ce graphe, où est tracé la solution approchée et la solution exacte (en rouge) :

Lorsque $N=100$:

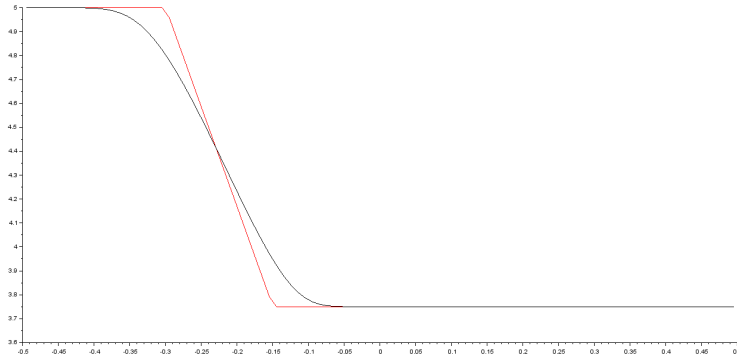


FIGURE 8 – Courbe de la solution exacte et de la solution approchée lorsque $N=100$

Lorsque $N=600$:

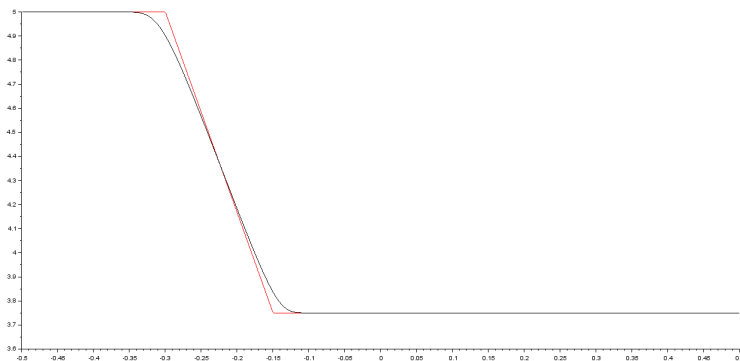


FIGURE 9 – Courbe de la solution exacte et de la solution approchée lorsque $N=600$

Interprétations des résultats

Sur la figure 8, on fixe le nombre de points du maillage à $N = 100$. On observe déjà que la solution approchée coïncide avec la solution exacte. Lorsque l'on augmente le nombre de points et qu'on le fixe à $N = 600$ comme dans la figure 9, on voit clairement que la courbe de la solution approchée se rapproche encore plus. Ce qui est normal car plus le maillage est fin, mieux on approche la solution.

Troisième partie

Modèles Microscopiques

Les modèles microscopiques ont pour but d'étudier le trafic en se focalisant sur chaque véhicule. Nous allons présenter des modèles basés sur le comportement en poursuite correspondant au comportement d'un conducteur qui réagit aux actions d'un véhicule qui le précède.

1 Modèles à vitesses optimales

La vitesse optimale est la vitesse jugée satisfaisante par le conducteur, elle dépend de son interdistance par rapport au véhicule meneur.

1.1 Modèle de Newell

Le premier modèle à vitesse optimale est le modèle de Newell apparu en 1961, il est basé sur cette équation :

$$\dot{x}_i(t + \Delta t) = v_0(x_{i-1}(t) - x_i(t))$$

où v_0 est la vitesse optimale et Δt est le temps nécessaire au changement de vitesse.

1.2 Modèle de Bando

En 1990, la notion de vitesse désirée, qui dépend encore de l'interdistance, est proposé par Bando. Puis en 1995, Bando et Hasebe publient un livre qui propose le modèle OVM, issu du développement limité au 1er ordre du modèle de Newell :

$$\ddot{x}_i(t + \Delta t) = \alpha(V_0(x_{i-1}(t) - x_i(t)) - \dot{x}_i(t))$$

$$\text{où } V_0(x_{i-1}(t) - x_i(t)) = \frac{V_{max}}{2}(\tanh(x_{i-1}(t) - x_i(t) - S_c) + \tanh(S_c))$$

et S_c est l'interdistance du comportement de poursuite, paramètre à fixer et α est la sensibilité du conducteur.

1.3 Modèle de Chandler

Dans le modèle de Chandler, tous les couples véhicule/conducteur sont identiques c'est à dire qu'on suppose qu'il y a le même temps de réaction et la même distance d'arrêt. De plus, l'accélération est proportionnelle au différentiel de vitesse entre le véhicule et celui de devant.

Il est basé sur cette équation :

$$\ddot{x}_i(t + \Delta t) = \alpha(\dot{x}_i(t) - \dot{x}_{i+1}(t)) \text{ avec } \alpha \text{ est la sensibilité du conducteur}$$

2 Modèles à distance de sécurité

Dans ces modèles à distance de sécurité nous cherchons à décrire simplement la dynamique du seul véhicule en relation avec son prédécesseur de sorte à respecter une certaines distances de sécurité.

2.1 Les modèles simples de Pipes

En 1953 le modèles simples de Pipes devient l'un des premier modèles à avoir développé cette idée. Ce modèle supposait la règle d'interdistance minimale. Cette règle d'interdistance minimale definie par :

$$S_i(t) = x_{i-1}(t) - x_i(t)$$

Où $x_i(t)$ designant la position du vehicule i au temps t et $x_{i-1}(t)$ désignant la position du vehicule i-1 au temps t. Une bonne règle pour suivre un autre véhicule à une distance sûre est de laisser l'équivalence d'une longueur de son propre véhicule soit 16,1 km/h. Ce qui se traduit par la relation suivante :

$$S_{i,sécurité}(t) = L_i \left(1 + \frac{v_i(t)}{16.1} \right)$$

Où v_i est la vitesse du véhicule du vehicule i au temps t noté aussi \dot{x}_i , L_i est la longueur du vehicule i, $S_{i,sécurité}$ est la distance minimale de sécurité qui augmente lineairement dans ce modèle avec la vitesse du véhicule $v_i(t)$

2.2 Les modèles de collision (AC)

Les modèles dit de collision avoidance en abrégé AC soit litteralement, modèles de prévention des accidents. Le premier modèle de ce type a été développé en 1959 par Kometani et Sasaki. Le modèle vise plutôt à retranscrire la trajectoire d'un véhicule en fonction d'une distance de sécurité minimale. Le developpement de ce type de modèle s'appuie sur une utilisation des systèmes d'équations newtoniennes de mouvement qui s'expriment par :

$$S_{i,sécurité}(t) = \alpha v_{i-1}^2(t) + \beta v_i^2(t + T) + \gamma v_i(t + T) + \delta$$

Où α et β représentent respectivement, l'inverse de la capacité maximale de décélération du véhicule leader et du véhicule suiveur. δ est homogène à une distance, γ est homogène à l'inverse d'un temps et T est le temps de relaxation.

2.3 Les Travaux de Gibbs

Les travaux de GIPPS ont pour objectif de compléter l'approche initiale en incorporant une vitesse maximale désirée $V_{i,désirée}$ ainsi q'un taux de freinage maximal b_i^{max} s'apparentant aux termes cinétiques de l'équation précédente. Ainsi la vitesse du véhicule est definie par la valeur minimale entre $v_i^a(t + T_r)$, la vitesse qu'il peut réellement atteindre d'un point de vue dynamique, et $v_i^b(t + T_r)$ la vitesse qu'il est possible d'adopter en connaissant les contraintes de sécurité liées à la présence d'un véhicule leader. Cela s'exprime comme suit :

$$v_i(t + T_r) = \min(v_i^a(t + T_r), v_i^b(t + T_r))$$

Où S_i est une interdistance minimale prenant en compte la longueur du véhicule leader ainsi qu'une distance minimale dans laquelle le véhicule suiveur ne pénètre jamais même à l'arrêt. La décélération maximale du véhicule leader est estimée visuellement par le conducteur du véhicule suivant. Elle est exprimée par b_{i-1}^{max} .

3 Modèles des automates cellulaires

Les automates cellulaires peuvent être vu comme l'intermédiaire entre les approches macroscopiques et microscopiques dans lesquelles l'espace et le temps y sont discrétisés ainsi que la vitesse des véhicules.

Nous avons plusieurs types de modèles d'automates cellulaires.

Le Modèle de Nagel-Schreckenberg (Nasch)

Nous ne pouvons pas commencer la description des modèles d'automates cellulaires sans rappeler les règles du modèle pionnier proposées par Nasch qui précise que chaque véhicule à une position (X_n) et une vitesse entière

$$V_n = 0, 1, 2, \dots, V_{\max}.$$

La distance entre les véhicules est notée

$$d_n = X_n + V_n.$$

La dynamique est effectuée en parallèle pour tous les véhicules.

Etape 1 : Accélération

Si

$$V_n < V_{\max}$$

La vitesse du $n^{\text{ième}}$ véhicule est augmentée d'une unité mais V_n demeure inchangée si $V_n = V_{\max}$.

Mais ce cas :

$$V_n \longrightarrow \min(V_n + 1, V_{\max})$$

Etape 2 : Décélération (due aux autres véhicules)

Si $d_n < V_n$ alors la vitesse du $n^{\text{ième}}$ véhicule est réduite à $d_n - 1$

$$V_n \longrightarrow \min(V_n, d_n - 1)$$

Etape 3 : Facteur aléatoire

Si $V_n > 0$ alors la vitesse du $n^{\text{ième}}$ véhicule est réduite d'une unité avec une probabilité p mais demeure inchangée si $V_n = 0$.

$$V_n \longrightarrow \max(V_n - 1, 0)$$

Etape 4 : Mouvement des véhicules

Dans le modèle de Nasch chaque véhicule avance selon :

$$V_n \longrightarrow V_n + d_n$$

4 Exemple d'analyse d'un modèle microscopique

4.1 Présentation d'un modèle

Considerons le modèle microscopique du second ordre de Chandler et Alfred :

$$M \frac{\partial U_i(t)}{\partial t} = \lambda(U_{i-1}(t - \Gamma) - U_i(t - \Gamma)), \text{ pour tout } t > \Gamma, \text{ avec } i = 2, \dots, N \quad (1)$$

Où $(U_i)_i$ est l'ensemble des N vitesses des véhicules considérés, M est la masse supposée être identique pour tous les véhicules, λ est un coefficient réel donné et $\Gamma \geq 0$ est un temps de retard.

Supposons que les véhicules sont ordonnés de sorte que le véhicule $i - 1$ précède le véhicule i . Considérons maintenant $(x_i)_i$ l'ensemble des positions des véhicules définies telles que :

$$x_i(t) = \int_0^t U_i(s) \cdot \partial s, \text{ pour tout } t \geq 0, \text{ avec } i = 1, \dots, N \quad (2)$$

On remarque donc que $\dot{x}_i(t) = U_i(t)$. On pose $\rho = \frac{\lambda}{M}$.
Et on peut écrire l'équation (1) de la forme :

$$\ddot{x}_i(t) = \rho(\dot{x}_{i-1}(t - \Gamma) - \dot{x}_i(t - \Gamma)), \text{ pour tout } t \geq \Gamma, \text{ avec } i = 2, \dots, N \quad (3)$$

Remarquons aussi que ce modèle peut s'écrire de façon plus générique sous la forme :

$$\ddot{x}_i(t) = \Lambda(\dot{x}_{i+1}(t - \Gamma) - \dot{x}_i(t - \Gamma)), \text{ pour tout } t \geq \Gamma, \text{ avec } i = 2, \dots, N \quad (4)$$

Où $\Lambda : v \mapsto \Lambda(v)$ est une fonction continue, monotone et strictement croissante.

Rappelons que $x_{i-1}(t) \geq x_i(t)$ à $t = 0$ et pour tout $i = 2, \dots, N$.

Pour que le problème soit bien posé mathématiquement, nous considérons une condition initiale, c'est à dire à $t = 0$, qui donne les positions initiales des véhicules.

Lorsque $\Gamma > 0$, il est aussi nécessaire de se donner une conditions sur les vitesses des véhicules sur l'intervalle $[0, \Gamma]$.

Lorsque $N < \infty$, nous avons un système fini et il est nécessaire de se donner une condition au bord qui donne la trajectoire des premier véhicule .

Ces hypothèses suffisent ainsi pour montrer l'existence et l'unicité de l'équation (3).

Il est intéressant de noter que l'équation (3) peut être intégrée en temps de telle sorte que l'on obtienne un modèle de premier ordre qui s'écrit sous la forme :

$$\dot{x}_i(t) = \rho(x_{i-1}(t - \Gamma) - x_i(t - \Gamma)), \text{ pour tout } t \geq \Gamma, \text{ avec } i = 2, \dots, N \quad (5)$$

Dans ce cas, il est possible de réécrire l'équation (5) sous sa forme générique :

$$\dot{x}_i(t) = \nu(x_{i-1}(t - \Gamma) - x_i(t - \Gamma)), \text{ pour tout } t \geq \Gamma, \text{ avec } i = 2, \dots, N \quad (6)$$

Où $\nu \mapsto \nu(d)$ une fonction continue, monotone, strictement croissante et lipschitzienne.

4.2 Passage du modèle de chandler au modèle de LWR

L'idée de relier les modèles microscopiques et les modèles macroscopiques est assez générique lorsque l'on modélise un phénomène physique. En modélisation du trafic routier, l'idée est similaire bien que les techniques diffèrent. On peut formellement montrer que le modèle de Chandler et Albert tend vers le modèle de Lighthill-Whitham-Richards (LWR).

Considérons cette équation :

$$\dot{x}_i(t) = \nu(x_{i-1}(t - \Gamma) - x_i(t - \Gamma)), \text{ pour tout } t \geq \Gamma, \text{ avec } i \in \mathbb{Z}$$

Montrons qu'elle converge par passage à la limite vers la solution de l'équation aux dérivées partielles (équation d'Hamilton - Jacobi) suivante :

$$\partial_t X^0(t, n) = \nu(-\partial_n X^0(t, n)), \text{ pour tout } (t, n) \in [0, +\infty[\times \mathbb{R} \quad (7)$$

On définit la fonction X^ϵ telle que :

$$X^\epsilon = \epsilon x_{\lfloor \frac{n}{\epsilon} \rfloor}(\frac{t}{\epsilon}), \text{ pour tout } (t, n) \in [0, +\infty[\times \mathbb{R}$$

Où $\lfloor \cdot \rfloor$ est la partie entière, $\epsilon > 0$ un facteur d'échelle que l'on va tendre vers 0, le temps et n l'indice des véhicules. En remplaçant dans l'équation (6) les $(x_i)_i$, on obtient :

$$X^\epsilon = \frac{1}{\epsilon} X^\epsilon(\epsilon t, \epsilon i), \text{ pour tout } t \in [0, +\infty[\text{ et } i \in \mathbb{Z}$$

Pour démontrer la convergence, utilisons ce Théorème :

Théorème :

On suppose que X vérifie l'équation d'Hamilton - Jacobi, si la densité ρ est définie par :

$$\rho(x, t) = \frac{1}{\partial_n X(n, t)} \text{ avec } x = X(n, t) \quad (8)$$

alors ρ vérifie la loi de conservation c'est à dire l'équation du modèle LWR .

Démonstration :

Soit t fixé, X est inversible par rapport à sa première variable alors on peut donc poser $n = X(x, t)$ avec $Y(., t) = X^{-1}(., t)$

La densité ρ est alors égale à :

$$\rho(x, t) = \frac{1}{\partial_n X(n, t)} = \partial_x Y(x, t)$$

Puisque X et Y sont des facteurs inversibles alors on dérive par rapport au temps :

$$\partial_t \rho(x, t) = \partial_t \partial_x Y(x, t) = \partial_x \partial_t Y(x, t)$$

D'autre part pour $n = Y(x, t)$ on peut écrire alors :

$$x = X(n, t) = X(Y(x, t), t)$$

Ainsi en différenciant x par rapport à t , on obtient directement :

$$\partial_t X + \partial_n X \partial_t Y = 0$$

D'où d'après l'équation (6) et (7)

$$\partial_t Y = -\frac{\partial_t X}{\partial_n X}$$

Donc $\partial_t Y = -\rho(x, t)\nu(\partial_n X(n, t))$

Et finalement :

$$\partial_t \rho(x, t) + \partial_x [\rho(x, t)\nu(\partial_n X(n, t))] = 0$$

Or $\partial_n X(n, t) = \frac{1}{\rho(x, t)}$

Par conséquent :

$$\partial_t \rho(x, t) + \partial_x [\rho(x, t)\nu(\frac{1}{\rho(x, t)})]$$

En posant $Q(\rho(x, t)) = [\rho(x, t)\nu(\frac{1}{\rho(x, t)})]$, on a :

$$\partial_t \rho(x, t) + Q(\rho(x, t)) = 0 \text{ (Equation du modèle LWR)}$$

Quatrième partie

Conclusion

Pour conclure, il est intéressant d'observer le trafic routier à l'aide des modèles macroscopiques pour pouvoir comprendre la situation d'un point de vue plus général. Mais aussi les modèles microscopiques nous permettent d'avoir plus d'hypothèses sur chaque véhicule. Pour modéliser le système routier, il est donc utile d'utiliser les deux méthodes afin d'obtenir le plus d'informations. Cela peut ensuite servir à trouver de nouvelles solutions pour rendre le trafic fluide dans différentes situations.

Pour nous, ce projet a permis de travailler en groupe, d'en apprendre plus sur les équations différentielles et de les modéliser. Cela nous a donc permis d'appliquer nos compétences acquises en cours. On a pu observer comment les mathématiques pouvaient s'appliquer dans la vie active.

Programmes en scilab

Programme qui modélise un feu tricolore :

```
1 a=0
2 b=1 // intervalle [a,b]
3 N=300 // nombre de points sur le maillage en espace
4 T=0.01 // temps d'arret de calcul
5 dx=(b-a)/N // pas en espace
6 Vmax=30 // vitesse max
7 pmax=5 // densite max
8 deff('y=V(p)', 'y=(-Vmax/pmax)*p+Vmax'); // fonction vitesse
9 deff('y=q(p)', 'y=p*Vmax*(1-p/pmax)'); // fonction flux
10 deff('y=dq(p)', 'y=Vmax-2*(Vmax/pmax)*p'); //derivee fonction flux
11 dt=dx/(2*dq(0)) // pas de temps
12 x=zeros(N)
13 for i=1:N
14     x(i)=(a+dx/2)+(i-1)*dx // vecteur x
15 end
16
17 ug=(1/4)*pmax
18 ud=0
19
20 function z=g(x,y) // fonction g definie par morceaux
21     if x<=y then
22         z=min(q(x),q(y))
23     end
24     if ((pmax/2)<=y)&(y<x) then
25         z=q(y)
26     end
27     if (y<pmax/2)&((pmax/2)<x) then
28         z=q(pmax/2)
29     end
30     if (y<x)&(x<=(pmax/2)) then
31         z=q(x)
32     end
33 endfunction
34
35 function y=px0(x) // donnee initiale : p(x,t=0) : simule un feu rouge
36     if x<=((a+b)/2) then
37         y=ug // ug=(1/4)*pmax car la densite est importante avant le feu car il
                 est rouge
38     else
39         y=ud // ud=0 car la densite est nulle apres le feu
40     end
41 endfunction
```

```

42
43 t=0
44 G=zeros(N+1)
45 U=zeros(N)
46
47 for j=1:N
48     U(j)=px0(x(j)) // au temps t=0, U(x,t)=p(x,t=0)
49 end
50
51 while(t<=T)
52     t=t+dt // on incremente le temps
53     G(1)=g(ug,ug) // condition de bord
54     G(N+1)=g(ud,ud) // condition de bord
55     for k=2:N
56         G(k)=g(U(k-1),U(k))
57     end
58     for j=1:N
59         U(j)=U(j)-dt/dx*(G(j+1)-G(j)) // modification du vecteur U au temps n+1
60     end
61 end
62
63 plot2d(x,U)

```

Programme qui modélise un ralentissement :

Petit Ralentissement :

```

1 a=0
2 b=1 // intervalle [a,b]
3 N=100 // nombre de points sur le maillage en espace
4 T=0.03 // temps d'arret de calcul
5 dx=(b-a)/N // pas en espace
6 Vmax=30 // vitesse max
7 pmax=5 // densite max
8 deff('y=V(p)', 'y=(-Vmax/pmax)*p+Vmax'); // fonction vitesse
9 deff('y=q(p)', 'y=p*Vmax*(1-p/pmax)'); // fonction flux
10 deff('y=dq(p)', 'y=Vmax-2*(Vmax/pmax)*p'); //derivee fonction flux
11 dt=dx/(2*dq(0)) // pas de temps
12 x=zeros(N)
13 for i=1:N
14     x(i)=(a+dx/2)+(i-1)*dx // vecteur x
15 end
16
17 ug=(1/8)*pmax
18 ud=(1/2)*pmax

```

```

19
20 function z=g(x,y) // fonction g definie par morceaux
21     if x<=y then
22         z=min(q(x),q(y))
23     end
24     if ((pmax/2)<=y)&(y<x) then
25         z=q(y)
26     end
27     if (y<pmax/2)&((pmax/2)<x) then
28         z=q(pmax/2)
29     end
30     if (y<x)&(x<=(pmax/2)) then
31         z=q(x)
32     end
33 endfunction
34
35 function y=px0(x) // donnee initiale : p(x,t=0) : simule un petit
    ralentissement
36     if x<=((a+b)/2) then
37         y=ug // la densite est faible
38     else
39         y=ud // la densite est plus forte
40     end
41 endfunction
42
43 temps=0
44 G=zeros(N+1)
45 U=zeros(N)
46
47 for j=1:N
48     U(j)=px0(x(j)) // au temps t=0, U(x,t)=p(x,t=0)
49 end
50
51 while(t<=T)
52     t=t+dt // on incremente le temps
53     G(1)=g(ug,ug) // condition de bord
54     G(N+1)=g(ud,ud) // condition de bord
55     for k=2:N
56         G(k)=g(U(k-1),U(k))
57     end
58     for j=1:N
59         U(j)=U(j)-dt/dx*(G(j+1)-G(j)) // modification du vecteur U au temps n+1
60     end
61 end
62
63 plot2d(x,U)

```


Fort Ralentissement :

```
1  a=0
2  b=1 // intervalle [a,b]
3  N=100 // nombre de points sur le maillage en espace
4  T=0.07 // temps d'arret de calcul
5  dx=(b-a)/N // pas en espace
6  Vmax=30 // vitesse max
7  pmax=5 // densite max
8  deff('y=V(p)', 'y=(-Vmax/pmax)*p+Vmax'); // fonction vitesse
9  deff('y=q(p)', 'y=p*Vmax*(1-p/pmax)'); // fonction flux
10 deff('y=dq(p)', 'y=Vmax-2*(Vmax/pmax)*p'); //derivee fonction flux
11 dt=dx/(2*dq(0)) // pas de temps
12 x=zeros(N)
13 for i=1:N
14     x(i)=(a+dx/2)+(i-1)*dx // vecteur x
15 end
16
17 ug=(1/8)*pmax
18 ud=pmax
19
20 function z=g(x,y) // fonction g definie par morceaux
21     if x<=y then
22         z=min(q(x),q(y))
23     end
24     if ((pmax/2)<=y)&(y<x) then
25         z=q(y)
26     end
27     if (y<pmax/2)&((pmax/2)<x) then
28         z=q(pmax/2)
29     end
30     if (y<x)&(x<=(pmax/2)) then
31         z=q(x)
32     end
33 endfunction
34
35 function y=px0(x) // donnee initiale : p(x,t=0)
36     if x<=((a+b)/2) then
37         y=ug // la densite est faible
38     else
39         y=ud // la densite est maximale
40     end
41 endfunction
42
43 temps=0
44 G=zeros(N+1)
```

```

45 U=zeros(N)
46
47 for j=1:N
48     U(j)=px0(x(j)) // au temps t=0, U(x,t)=p(x,t=0)
49 end
50
51 while(t<=T)
52     t=t+dt // on incremente le temps
53     G(1)=g(ug,ug) // condition de bord
54     G(N+1)=g(ud,ud) // condition de bord
55     for k=2:N
56         G(k)=g(U(k-1),U(k))
57     end
58     for j=1:N
59         U(j)=U(j)-dt/dx*(G(j+1)-G(j)) // modification du vecteur U au temps n+1
60     end
61 end
62
63 plot2d(x,U)

```

Programme qui modélise un feu tricolore suivi de travaux :

```

1
2 a=0
3 b=1 // intervalle [a,b]
4 N=100 // nombre de points sur le maillage en espace
5 T=0.08 // temps d'arret de calcul
6 dx=(b-a)/N // pas en espace
7 Vmax=30 // vitesse max
8 pmax=5 // densite max
9 deff('y=V(p)', 'y=(-Vmax/pmax)*p+Vmax'); // fonction vitesse
10 deff('y=q(p)', 'y=p*Vmax*(1-p/pmax)'); // fonction flux
11 deff('y=dq(p)', 'y=Vmax-2*(Vmax/pmax)*p'); //derivee fonction flux
12 dt=dx/(2*dq(0)) // pas de temps
13 x=zeros(N)
14 for i=1:N
15     x(i)=(a+dx/2)+(i-1)*dx // vecteur x
16 end
17
18 ug=pmax/5
19 ud=0
20
21 F = 10 // condition sur le flux
22
23 function z=g(x,y) // fonction g definie par morceaux
24     if x<=y then

```

```

25         z=min(q(x),q(y))
26     end
27     if ((pmax/2)<=y)&(y<x) then
28         z=q(y)
29     end
30     if (y<pmax/2)&((pmax/2)<x) then
31         z=q(pmax/2)
32     end
33     if (y<x)&(x<=(pmax/2)) then
34         z=q(x)
35     end
36 endfunction
37
38 function y=px0(x) // donnee initiale : p(x,t=0) : simule un feu rouge
39     if x<=((a+b)/4) then
40         y=ug // ug=pmax la densite est maximale avant le feu car il est rouge
41     else
42         y=ud // ud=0 car la densite est nulle
43     end
44 endfunction
45
46 temps=0
47 G=zeros(N+1)
48 U=zeros(N)
49
50 for j=1:N
51     U(j)=px0(x(j)) // au temps t=0, U(x,t)=p(x,t=0)
52 end
53
54 while(temps<=T)
55     temps=temps+dt // on incremente le temps
56     G(1)=g(U(1),U(1)) // condition de bord
57     G(N+1)=g(U(N),U(N)) // condition de bord
58     for k=2:N
59         if k==N*3/4
60             G(k)=min(g(U(k-1),U(k)),F)
61         else
62             G(k)=g(U(k-1),U(k))
63         end
64     end
65     for j=1:N
66         U(j)=U(j)-dt/dx*(G(j+1)-G(j)) // modification du vecteur U au temps n+1
67     end
68 end
69
70 plot2d(x,U)

```

Programme qui modélise un feu tricolore et trace la solution exacte :

```
1  a=-0.5
2  b=0.5 // intervalle [a,b]
3  N=100 // nombre de points sur le maillage en espace
4  T=0.01 // temps d'arrêt de calcul
5  dx=(b-a)/N // pas en espace
6  vmax=30 // vitesse max
7  pmax=5 // densite max
8  deff('y=v(p)', 'y=(-vmax/pmax)*p+vmax'); // fonction vitesse
9  deff('y=q(p)', 'y=p*vmax*(1-p/pmax)'); // fonction flux
10 deff('y=dq(p)', 'y=vmax-2*(vmax/pmax)*p'); //derivee fonction flux
11 dt=dx/(2*dq(0)) // pas de temps
12 x=zeros(N)
13 for i=1:N
14     x(i)=(a+dx/2)+(i-1)*dx // vecteur x
15 end
16
17 ug=pmax
18 ud=(3/4)*pmax
19
20 function z=g(x,y) // fonction g definie par morceaux
21     if x<=y then
22         z=min(q(x),q(y))
23     end
24     if ((pmax/2)<=y)&(y<x) then
25         z=q(y)
26     end
27     if (y<pmax/2)&((pmax/2)<x) then
28         z=q(pmax/2)
29     end
30     if (y<x)&(x<=(pmax/2)) then
31         z=q(x)
32     end
33 endfunction
34
35 function y=px0(x) // donnee initiale : p(x,t=0) : modelise un feu rouge
36     if x<=((a+b)/2) then
37         y=ug // la densite est maximale
38     else
39         y=ud // la densite est faible
40     end
41 endfunction
42
43 t=0
44 G=zeros(N+1)
```

```

45 U=zeros(N)
46 V=zeros(N)
47 for j=1:N
48     U(j)=px0(x(j)) // au temps t=0, U(x,t)=p(x,t=0)
49 end
50
51 dfa=vmax*(1-(2*ug)/pmax)
52 dfb=vmax*(1-(2*ud)/pmax)
53
54 while(t<=T)
55     t=t+dt // on incremente le temps
56     G(1)=g(ug,ug) // condition de bord
57     G(N+1)=g(ud,ud) // condition de bord
58     for k=2:N
59         G(k)=g(U(k-1),U(k))
60     end
61     for j=1:N
62         U(j)=U(j)-dt/dx*(G(j+1)-G(j)) // modification du vecteur U au temps n+1
63     end
64 end
65
66 function z=p(x,y) // solution exacte
67     if (x/y)<dfa then
68         z=ug
69     elseif dfb<(x/y) then
70         z=ud
71     elseif ((x/y)>dfa) & ((x/y)<dfb) then
72         z=(pmax/(2*vmax))*(vmax-(x/y))
73     end
74 endfunction
75
76 for j=1:N
77     V(j)=p(x(j),T) // creation du vecteur V : solution exacte appliquee en x
78                     // et au temps T
79 end
80 plot2d(x,V,style=2)
81 plot2d(x,U,style=1)

```

Bibliographie

Références

- [1] Guillaume Costeseque, *TIPE : modélisation mathématique du trac routier* Octobre 2018
- [2] Guillaume Costeseque, *Modélisation du trafic routier : passage du microscopique au macroscopique* Avril 2012
- [3] Guilhem Dupuis, *Modélisation du trac routier Le modèle de Lighthill-Whitham-Richards* Juin 2016
- [4] Thomas Richez, Charles Bordet *Un modèle de trac routier* Mars 2011
- [5] Emmanuel Bourrel *Modélisation dynamique de l'écoulement du trafic routier : du macroscopique au microscopique* Décembre 2013
- [6] Cyrille Bertelle et Damien Olivier *Modélisation du trafic routier*
- [7] Mika Uzunova *Commande non entière des systèmes de commande, développement et application pour les modèles de flux de trafic routier* Décembre 2009