

# La Modélisation du Flux Routier : Application au choix des jonctions routières

TAHA OULHAZZAN  
SCEI 15800

Épreuve de TIPE

Session 2023

# Plan de l'exposé

- 1 Introduction
- 2 Etude théorique
  - Modèle LWR et relations de GREENSHIELDS
  - Comparaison entre deux Modes de Jonctions
- 3 Etude Expérimentale
- 4 Simulation Informatique
- 5 Conclusion et Discussion

# Introduction et Motivations



# Introduction et Motivations

Pourquoi prêter attention choix du mode de jonctions routières adéquat ?

- Les intersections souvent à l'origine d'embouteillages.
- Engagement financier et humain.
- Difficilement modifiables/réversibles.

# Etude théorique

## Modèle LWR et relations de GREENSHIELDS

- Le flux  $q$ , la densité  $k$  et la vitesse  $v$  sont liés par La relation fondamentale :

$$q = k v$$

- Les mesures expérimentales dûes à GREENSHIELDS en 1934 permettent d'ajouter :

$$v = v_f \left(1 - \frac{k}{k_m}\right) \quad \text{donc} \quad q(k) = v_f \left(k - \frac{k^2}{k_m}\right)$$

où  $\begin{cases} v_f : \text{la vitesse maximale autorisée sur la route} \\ k_m : \text{la densité maximale avant congestion} \end{cases}$

# Etude théorique

## Modèle LWR et relations de GREENSHIELDS

- Le flux  $q$ , la densité  $k$  et la vitesse  $v$  sont liés par La relation fondamentale :

$$q = k v$$

- Les mesures expérimentales dûes à GREENSHIELDS en 1934 permettent d'ajouter :

$$v = v_f \left(1 - \frac{k}{k_m}\right) \quad \text{donc :} \quad q(k) = v_f \left(k - \frac{k^2}{k_m}\right)$$

où  $\begin{cases} v_f : \text{la vitesse maximale autorisée sur la route} \\ k_m : \text{la densité maximale avant congestion} \end{cases}$

# Etude théorique

## Modèle LWR et relations de GREENSHIELDS

- Le flux  $q$ , la densité  $k$  et la vitesse  $v$  sont liés par La relation fondamentale :

$$q = k v$$

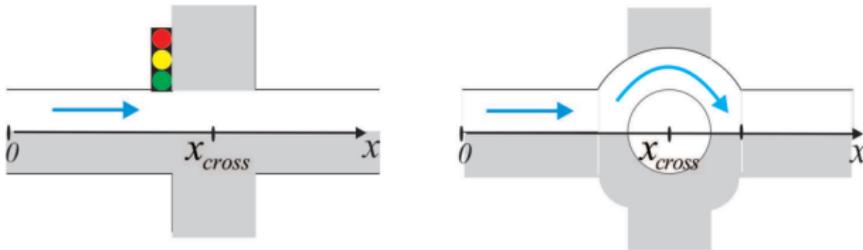
- Les mesures expérimentales dûes à GREENSHIELDS en 1934 permettent d'ajouter :

$$v = v_f \left(1 - \frac{k}{k_m}\right) \quad \text{donc :} \quad q(k) = v_f \left(k - \frac{k^2}{k_m}\right)$$

où  $\begin{cases} v_f : \text{la vitesse maximale autorisée sur la route} \\ k_m : \text{la densité maximale avant congestion} \end{cases}$

# Comparaison entre Rond-Points et Feux Tricolores

Modélisation :



Hypothèses :

- Une portion de route unidimensionnelle, de caractéristiques  $v_f$  et  $k_m$ , avec une intersection située à  $x_{cross}$ .
- On définit  $t_{clear}$  le temps nécessaire pour qu'un nombre donné de véhicules traverse l'intersection.

# Comparaison entre Rond-Points et Feux Tricolores

Modélisation :

## FEUX TRICOLORES :

On introduit un paramètre  $\alpha$  représentant la proportion de feu vert pendant un cycle :

$$\alpha = \frac{T_{vert}}{T_{cycle}} \quad , \quad v = \begin{cases} v_f & \text{si le feu est vert} \\ 0 & \text{si le feu est rouge} \end{cases}$$

Donc le flux moyen à travers l'intersection :  $q(k) = \alpha \cdot Q_{max}$

# Comparaison entre Rond-Points et Feux Tricolores

Modélisation :

## ROND-POINTS :

A l'approche d'un rond-point, les véhicules ralentissent généralement.

Les observations expérimentales indiquent que leur nouvelle vitesse est réduite d'un facteur  $\beta \in [0, 1]$

$$v = \beta.v_f$$

Donc le flux à travers le rond-point :  $q(k) = \beta.Q_{max}$

# Comparaison entre Rond-Points et Feux Tricolores

## Résultats théoriques :

On démontre alors que :

Pour les feux tricolores : 
$$t_{clear} = \frac{1}{\alpha} \cdot \frac{4x_{cross}}{v_f}$$

Et pour un rond-point : 
$$t_{clear} = \frac{1}{\beta} \cdot \frac{4x_{cross}}{v_f}$$

### Conclusion :

L'étude comparative par le modèle LWR macroscopique est dans notre cas insuffisante pour juger.

# Comparaison entre Rond-Points et Feux Tricolores

## Résultats théoriques :

On démontre alors que :

Pour les feux tricolores : 
$$t_{clear} = \frac{1}{\alpha} \cdot \frac{4x_{cross}}{v_f}$$

Et pour un rond-point : 
$$t_{clear} = \frac{1}{\beta} \cdot \frac{4x_{cross}}{v_f}$$

### Conclusion :

L'étude comparative par le modèle LWR macroscopique est dans notre cas insuffisante pour juger.

# Etude Expérimentale

Cadre de travail :

## Défi :

Vu les flux importants que connaissent les intersections en milieu urbain (*plusieurs milliers de véhicules par heure*), on est dans l'impossibilité d'en effectuer un suivi manuel avec rigueur.

Nous optons pour exploiter des mesures réelles effectués dans le ville de *Dublin, Etat de l'Ohio, Etats-Unis* par **M. Vincent SPAHR**, dans le cadre d'une thèse de recherche à l'*Université de Dayton*.

Toutes les mesures ont été effectuées avec des caméras spécialisées de trafic *MioVision*, durant l'été de 2014.

# Etude Expérimentale

Cadre de travail :

## Défi :

Vu les flux importants que connaissent les intersections en milieu urbain (*plusieurs milliers de véhicules par heure*), on est dans l'impossibilité d'en effectuer un suivi manuel avec rigueur.

Nous optons pour exploiter des mesures réelles effectués dans le ville de *Dublin, Etat de l'Ohio, Etats-Unis* par **M. Vincent SPAHR**, dans le cadre d'une thèse de recherche à l'*Université de Dayton*.

Toutes les mesures ont été effectuées avec des caméras spécialisées de trafic *MioVision*, durant l'été de 2014.

# Etude Expérimentale

Cadre de travail :



Figure – Avery-Muirfield Drive, Dublin, OH, USA



# Etude Expérimentale :

Mesures :

*Matinée :*

|            | Flux              | Retard        | Queue               | Saturation |
|------------|-------------------|---------------|---------------------|------------|
| Rond-point | 2157 <i>veh/h</i> | 19 <i>s</i>   | 12 <i>vehicules</i> | 83 %       |
| Feux       | 2979 <i>veh/h</i> | 54.6 <i>s</i> | 61 <i>vehicules</i> | 101 %      |

*Après-midi (heure de pointe) :*

|            | Flux              | Retard        | Queue               | Saturation |
|------------|-------------------|---------------|---------------------|------------|
| Rond-point | 3108 <i>veh/h</i> | 39.6 <i>s</i> | 33 <i>vehicules</i> | 103 %      |
| Feux       | 3420 <i>veh/h</i> | 44.9 <i>s</i> | 36 <i>vehicules</i> | 87 %       |

# Etude Expérimentale :

## Résultats :

On observe que :

- Le rond-point performe largement mieux sous trafic relativement léger.
- Le rond-point chute en performance lorsqu'on excède sa capacité recommandée.
- Le feu-rouge présente des temps de retard importants dans les deux cas, avec légèrement moins de saturation en régime congestionné.

# Simulation Informatique

Modèle :

Sous *Python*, nous modelisons dans ce qui suit une portion de route par une liste de Booléen *True* ou *False* telle que :

$\left\{ \begin{array}{l} \textit{True} : \text{correspond à une case occupée par un véhicule.} \\ \textit{False} : \text{correspond à une case vide.} \end{array} \right.$

Exemple :



$L=[\textit{True}, \textit{False}, \textit{True}, \textit{True}, \textit{False}, \textit{False}, \textit{False}, \textit{False}, \textit{False}, \textit{False}, \textit{True}]$

# Simulation Informatique

Modèle :

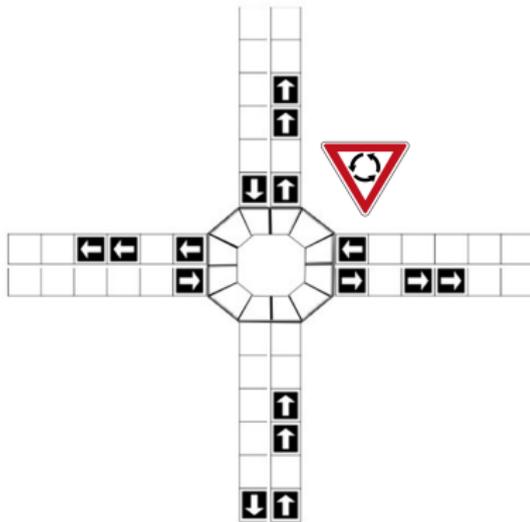


Figure – Modèle Rond-point



Figure – Modèle Feux tricolores

# Simulation Informatique

Principe :

Principe : Afin d'obtenir une estimation de  $t_{clear}$  :

- On génère une queue : portion de route contenant un nombre donné “*reserv*” de véhicules, distribués selon une densité que l'on choisit.
- On estime le nombre d'iterations requis pour évacuer cette reserve de véhicules à travers chacunes des deux jonctions.

# Simulation Informatique

Résultats :

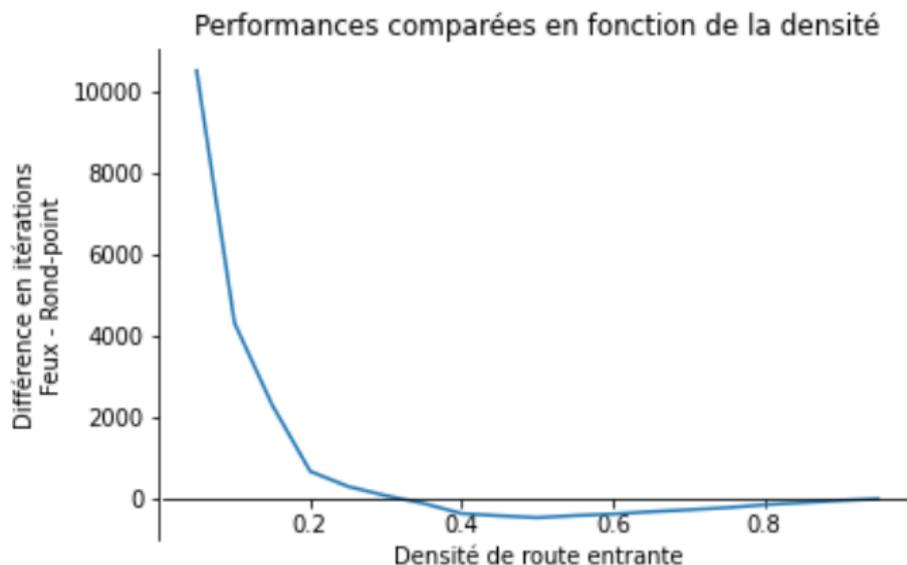
Résultats des compilations : (*reserv* = 500 véhicules)

|                       |            |      |      |      |
|-----------------------|------------|------|------|------|
|                       | Densités : | 0.25 | 0.33 | 0.5  |
| Itérations requises : | Rond-point | 1788 | 1702 | 1529 |
|                       | Feux       | 2080 | 1576 | 1054 |

# Simulation Informatique

Résultats :

## Résultats des compilations :



# Conclusion et Discussion

Performances :

- A trafic **léger**, les rond-points sont largement plus optimaux.
- Pour un trafic plus **important**, les feux tricolores sont légèrement plus performants.

# Conclusion et Discussion

Performances :

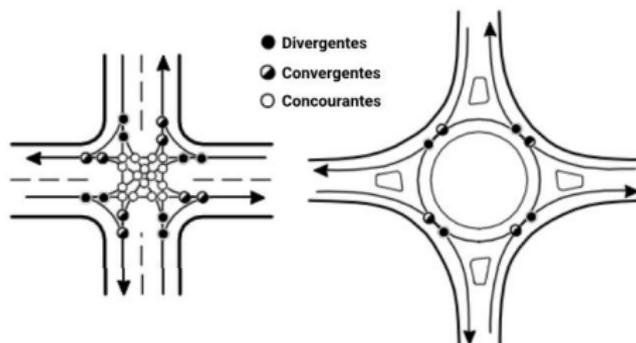
- A trafic **léger**, les rond-points sont largement plus optimaux.
- Pour un trafic plus **important**, les feux tricolores sont légèrement plus performants.

# Conclusion et Discussion

Sécurité :

Table – Données enregistrées à *Dublin, OH, USA* entre 2009 et 2013

|             | Accidents | Blessures | Sans | Pourcentage |
|-------------|-----------|-----------|------|-------------|
| Rond-points | 107       | 19        | 78   | 17.7 %      |
| Feux        | 97        | 29        | 68   | 29.8 %      |



## Annexe

Démonstration de l'expression de  $t_{clear}$  :

On a d'après GREENSHIELDS :  $q(k) = v_f(k - \frac{k^2}{k_m})$

$$\text{Donc : } \frac{dq}{dk} = 0 \implies k = \frac{k_m}{2}$$

$$\text{D'où : } Q_{max} = \frac{v_f \cdot k_m}{4}$$

$$\text{Et : } t_{clear} = \frac{n_{vehicules}}{q} = \frac{x_{cross} \cdot k_m}{\gamma \cdot Q_{max}}$$

$$\text{Ainsi : } t_{clear} = \frac{1}{\gamma} \cdot \frac{4x_{cross}}{v_f} \quad \text{où } \gamma = \alpha, \beta$$

# Annexe

```
1      """
2      TIPE 2023
3
4      L=liste de booleen (True,false) representant une portion de route
5      Toutes les Listes ci dessous sont prises de meme longueur
6
7      @author: TAHA OULHAZZAN
8      SCEI : 15800
9      """
10
11     def avancer(L,p):
12         return [p]+L[:-1]
13
14     def obstacle(L,ob,p):
15         i=ob
16         while L[i-1]: i+=-1
17         if i!=0 : return avancer(L[:i],p)+L[i+1:ob]+[False]+L[ob:-1]
18         else : return -1
19
20     def impasse(L,p):
21         j=len(L)
22         while j>0 and L[j-1]:
23             j-=1
24         if j!=0 : return avancer(L[:j],p)+L[j:]
25         elif p : return -1
26         else : return L
```

## Annexe

```
28 def inter(L1,p1,L2,p2,ob):
29     if L1[ob-1] : return avancer(L1,p1),obstacle(L2,ob,p2)
30     else : return avancer(L1,p1), avancer(L2,p2)
31
32 def ele_rondpoint(LL,Lp,rp):
33     outL=[]
34     rp=avancer(rp,rp[-1])
35     for i in range(len(LL)):
36         if rp[(3*i+6)%12]==i+1 : rp[(3*i+6)%12]=0
37         if rp[3*i]!=0 :
38             outL.append(impasse(LL[i],Lp[i]))
39         else :
40             outL.append(avancer(LL[i],Lp[i]))
41             if LL[i][-1]: rp[3*i]=i+1
42     return outL,rp
43
44 def densite(n,k):
45     import random as rnd
46     L=n*[False]
47     S=rnd.sample(range(n),k)
48     for i in S: L[i]=True
49     return L
50
```

## Annexe

```
51 def rondpoint4(Lk,n,itr):
52     LL=[densite(n,k) for k in Lk]
53     rp=12*[0]
54     for i in range(0,itr):
55         r=i%n
56         if r==0 : queues=[densite(n,k) for k in Lk]
57         Lp=[x[-(r+1)] for x in queues]
58         LL,rp=ele_rondpoint(LL,Lp,rp)
59         if -1 in LL : return "Saturée à",i+1,"itérations"
60     return LL,rp
61
62 def ele_feux(L,p,signal):
63     if signal : return avancer(L,p)
64     else : return impasse(L,p)
65
66 def queue(reserv, k, n):
67     queue=[]
68     for i in range(reserv//k):
69         queue+=densite(n,k)
70     queue=densite(n*(reserv%k)//k,reserv%k)+queue
71     return queue
72
```

## Annexe

```
72
73 def feux(alpha, reserv, n, k): #k<reserv
74     Lsignal=int(n*alpha)*[True]+(n-int(n*alpha))*[False]
75     L=queue(reserv,k,n)
76     compt=0
77     while L!=len(L)*[False] :
78         compt+=1
79         L=ele_feux(L,False,Lsignal[compt%n])
80     return compt
81
82 def rondpoint(beta, reserv, n, k):
83     rp=12*[0]
84     LL=2*[queue(a,b,n)for(a,b)in[(reserv,k),(int((1-beta)*reserv*n/k),int((1-beta)*n))]]
85     Lp=4*[False]
86     compt=0
87     while LL[0]!=len(LL[0])*[False] or 1 in rp :
88         compt+=1
89         LL,rp=ele_rondpoint(LL,Lp,rp)
90     return compt
91
92 def differ(x):
93     return feux(x,500,25,12)-rondpoint(x,500,25,12)
94
```

## Annexe

```
95 import matplotlib.pyplot as plt
96 n=0
97 pas=0.05
98 Lx=[]
99 Ly=[]
100 while n+pas<0.96:
101     n+=pas
102     Lx.append(n)
103     Ly.append(differ(n))
104 fig = plt.figure()
105 ax = fig.add_subplot(1, 1, 1)
106 ax.spines['left'].set_position(('data', 0.0))
107 ax.spines['bottom'].set_position(('data', 0.0))
108 ax.spines['right'].set_color('none')
109 ax.spines['top'].set_color('none')
110 plt.plot(Lx, Ly)
111 plt.xlabel('Densité de route entrante')
112 plt.ylabel('Différence en itérations\nFeux - Rond-point')
113 plt.title('Performances comparées en fonction de la densité')
114 plt.show()
115
```