



# Modélisation, Simulation multi-niveau pour l'optimisation des politiques de vaccination

**Etudiante** : TRAN Thi Cam Giang

**Directeurs** : Yann CHEVALEYRE, Marc CHOISY

VU Dinh Thiem, Jean-Daniel ZUCKER

**Ecole doctoral** : EDITE 130, Université Pierre et Marie CURIE

**Date de début** : 01/10/2012

Bourse en cotutelle : UPMC/NIHE

**Financement**: PDI MSI, UPMC/IRD

# Contexte et Objectif

- Nombreuses maladies infectieuses comme la rougeole, la dengue en Asie, dans la plupart des pays d'Asie du sud-est. (REF)
- Vaccination : la politique *"de masse"* (*vacciner le maximum d'enfants avant un certain âge*) la plus ancienne (*débuts dans les années cinquante pour les pays riches*) et encore aujourd'hui la plus utilisée.
- Problèmes : trop **chère**, inefficace et strictement impossible à mettre en œuvre dans nombres de pays pauvres, notamment en Afrique comme à la fois problèmes financières et logistiques. (e.g. le projet de l'OMS d'extinction de la rougeole au Vietnam avant 2012 est un échec).

**OPTIMISATION des politiques de vaccination en Intelligence Artificielle qui soient plus efficaces, moins chères en prenant en compte la dimension spatiale.**

# Outil ?

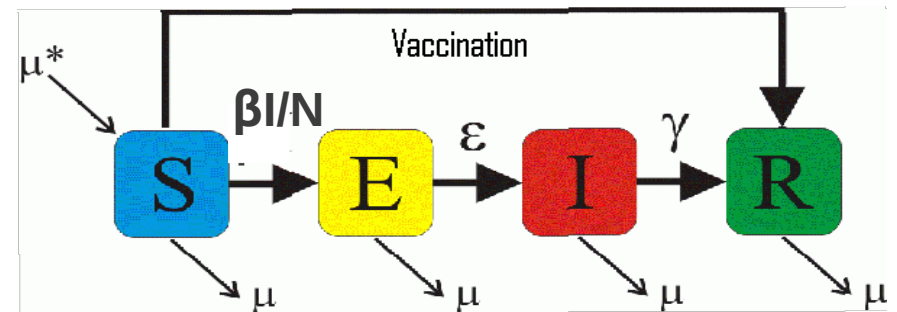


- Il existe le package R : très lent !
- Utiliser C++ pour faire la simulation stochastique avec  $N = 10M \rightarrow$  excellent !
- Développer un package d'intégration R/C++.

# Méthode

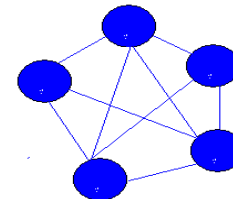
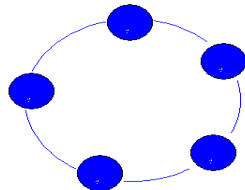
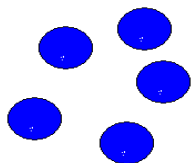
## (1) Modélisation épidémiologique stochastique

- Modèle SEIR de population :



- Algorithme stochastique de Gillespie (1977)
  - Motivation : pas d'**extinction** dans modèle déterministe
  - Approche: population-based time-to-event model, 2 étapes :
    - Chercher le temps du prochain événement
    - Chercher la nature du prochain événement

## (2) Modélisation épidémiologique spatiale



# Méthode

## (3) Optimisation des politiques de vaccination : apprentissage par renforcement

Pour une structure de population donnée, où et quand il faut vacciner afin de diminuer au maximum l'incidence globale ou augmenter au maximum la probabilité d'éradication globale.

- SARSA : Etat - Action - Récompense – Etat – Action
- Un état au moment  $t$  : ( $\in \mathbb{N}^4$ )

$$S = \left( (s_1, e_1, i_1, r_1), (s_2, e_2, i_2, r_2), \dots, (s_n, e_n, i_n, r_n) \right)$$

- Ensemble d'états :  $\mathbb{N}^{4 \times \text{nbVilles}}$
- Action au moment  $t$ , vaccination ou non vaccination



- Somme de récompense d'une politique :  $J : S \rightarrow \mathbb{R}$

$$\sum_{t=0}^{\infty} \gamma^t r_t = r_0 + \gamma r_1 + \gamma^2 r_2 + \gamma^3 r_3 + \dots$$

# Résultats attendus



- Les résultats de cette thèse sont : un **algorithme** efficace pour l'optimisation de **politiques** de vaccinations évaluées par **simulations stochastiques et spatiales**.
- Un **outil** informatique d'aide à la décision des politiques de vaccination utilisable par les professionnels de santé disponible sous la forme d'un **package R** (Interface R/C++)

# Objectifs Sept-Dec 2012

- Trouver la formule de la force d'infection pour un contexte spatial.

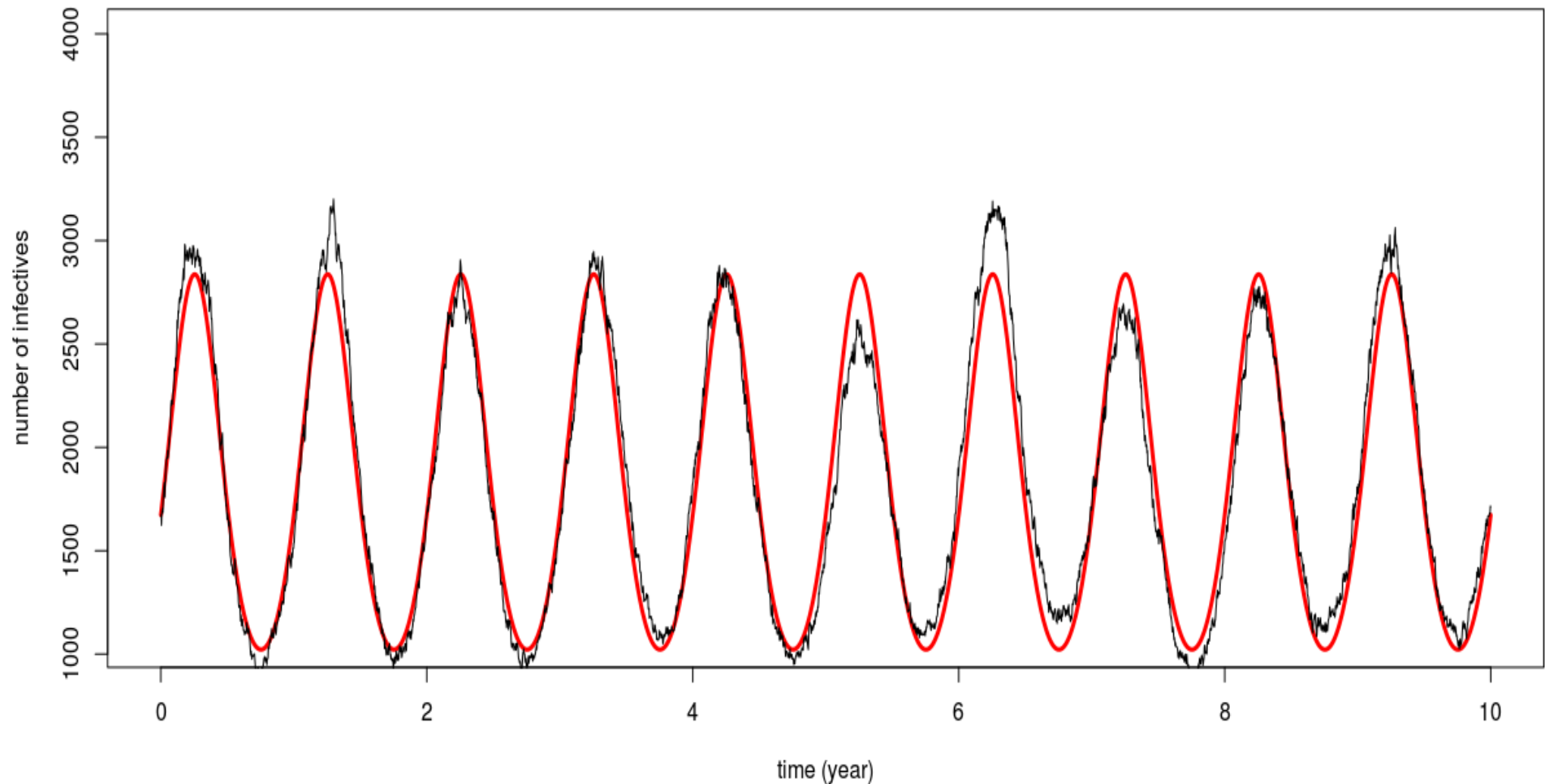
$$\lambda_i = \left( 1 - \sum_{\substack{k=1 \\ k \neq i}}^n \rho_{ik} \right) \beta_i \frac{I_i}{N_i} + \sum_{\substack{k=1 \\ k \neq i}}^n \rho_{ik} \frac{(1 - \varepsilon_{ik})\beta_i N_k + \varepsilon_{ik}\beta_k N_i}{N_i N_k} I_k$$

- Modélisation et simulation du modèle SEIR déterministe et stochastique sous les langages de programmation.

# Prototype développé en C++ : Exemple

## Nombre d'infectés pour 1 ville de 10M d'habitants

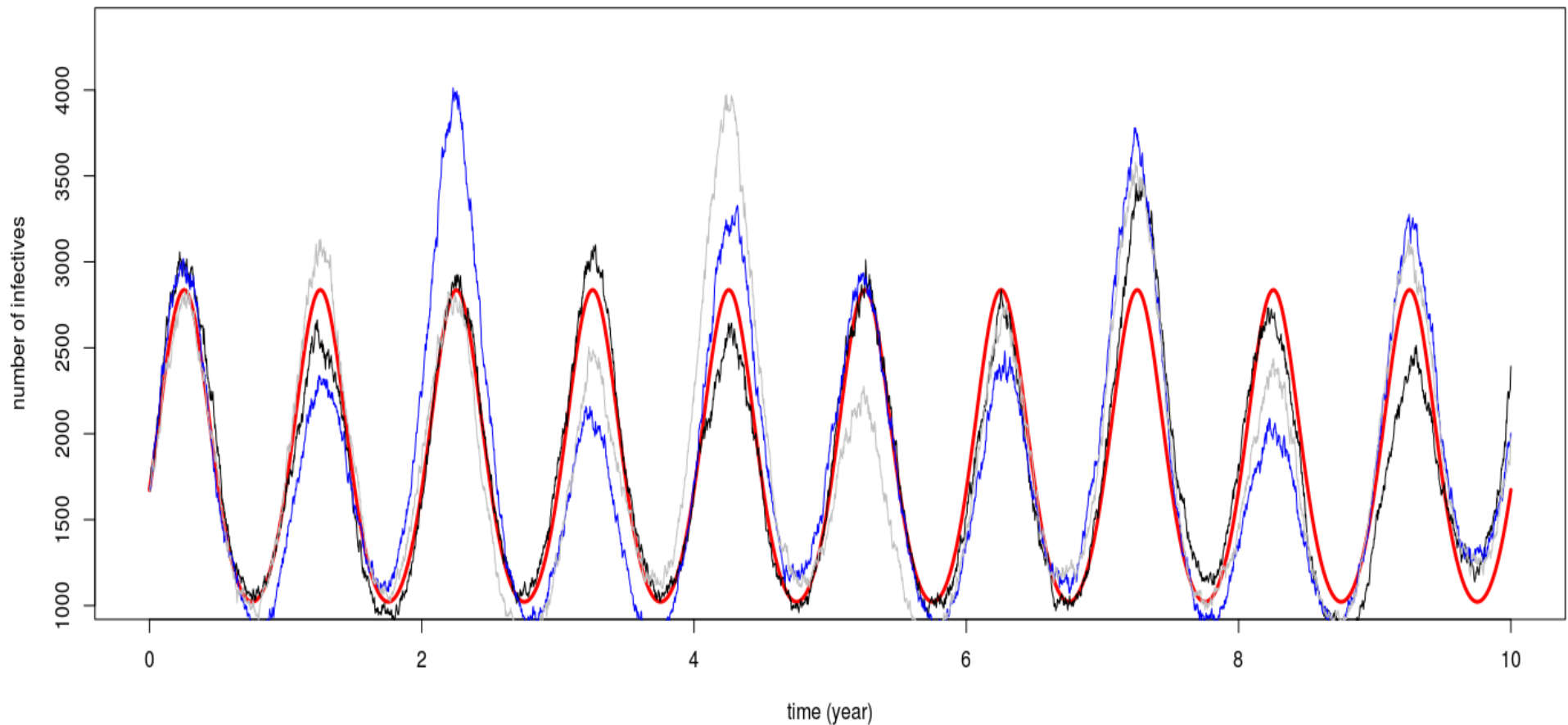
```
SIMULATION: -nbVilles 1 -tmax 3650 -sigma 0.125 -gamma 0.2  
-mu 3.91389432485323e-05 -epsilon 0 -topology 0 -rho 0 -unitTemps 1 -graine 1355734407.61301  
-S0 741559.447874164 -E0 2794.62443191756 -I0 1673.21742231949 -R0 9253972.7102716 -N0 1e+07 -beta0 2.73972602739726 -beta1 0.1  
-phiMIN 0 -phiMAX 0 -nbSimu 1 -typeFonc 0
```





# Exemple 2 : Nombre d'infectés en fonction du temps pour 3 villes de 10Millions d'habitants

SIMULATION: -nbVilles 3 -tmax 3650 -sigma 0.125 -gamma 0.2  
-mu 3.91389432485323e-05 -epsilon 0 -topology 0 -rho 0 -unitTemps 1 -graine 1355734476.85985  
-S0 741559.447874164 -E0 2794.62443191756 -I0 1673.21742231949 -R0 9253972.7102716 -N0 1e+07 -beta0 2.73972602739726 -beta1 0.1  
-phiMIN 0 -phiMAX 0 -nbSimu 1 -typeFonc 0



# Question 1 : persistance dans le modèle simulé



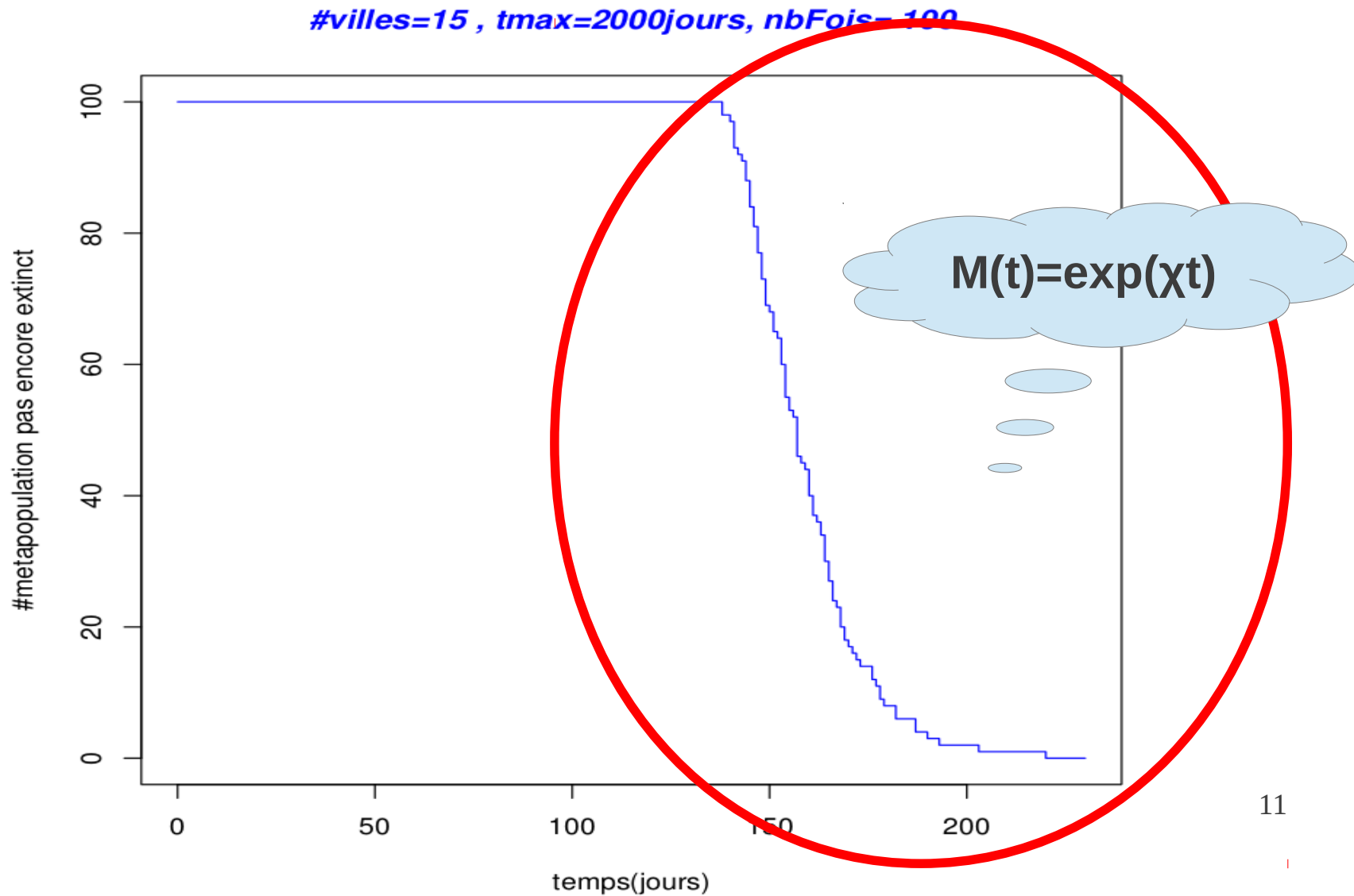
- Trouver la caractérisation de la persistance globale. C'est la courbe de survie qui est en forme de

$$M(t) = \exp(-\chi t)$$

où  $M(t)$  le nombre de méta-populations qui ne sont pas éteintes au moment  $t$ .

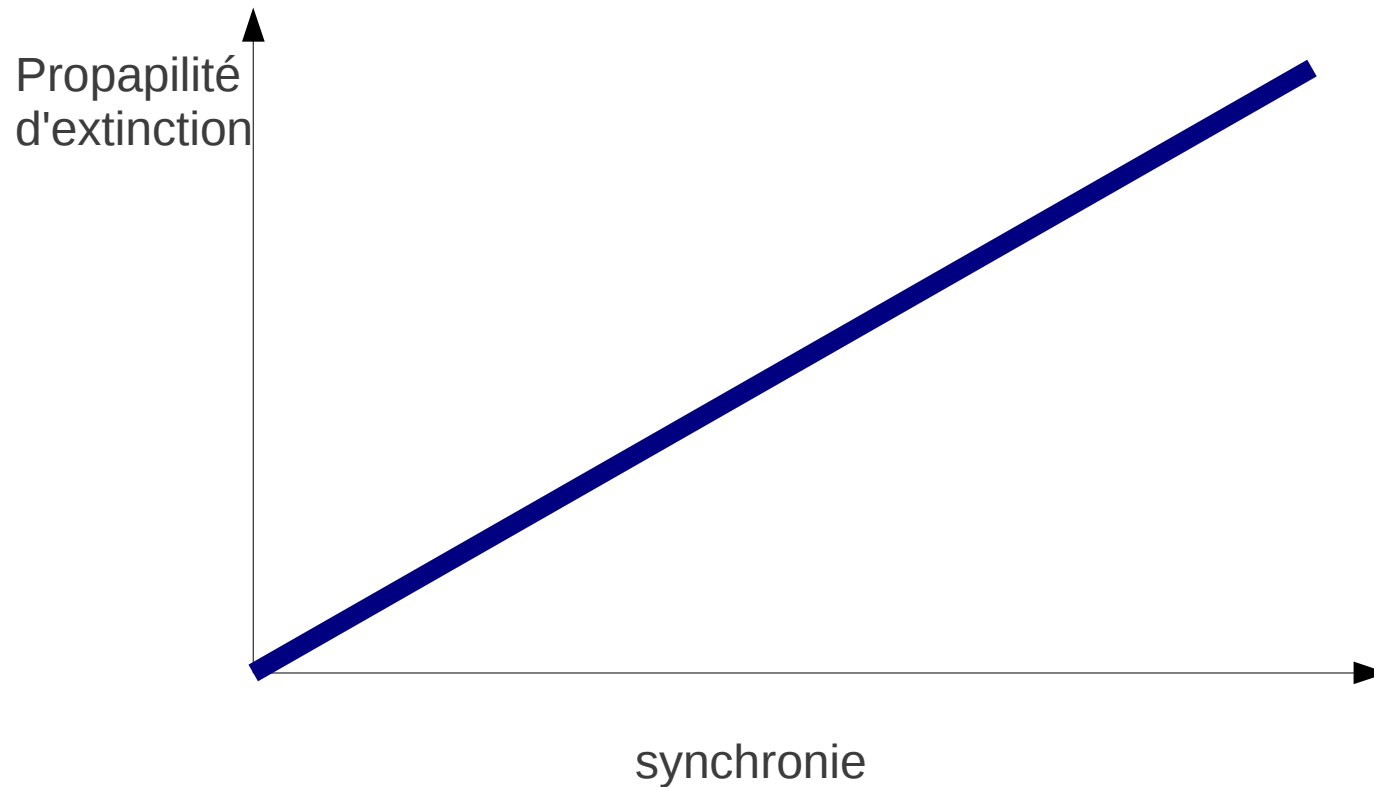
# Persistance dans le modèle simulé

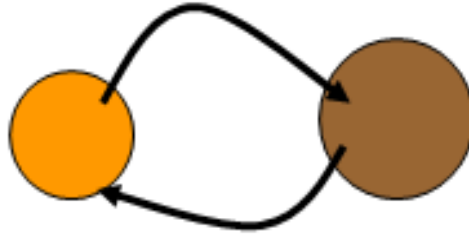
## Courbe de survie Kaplan-Meier



# Question 2

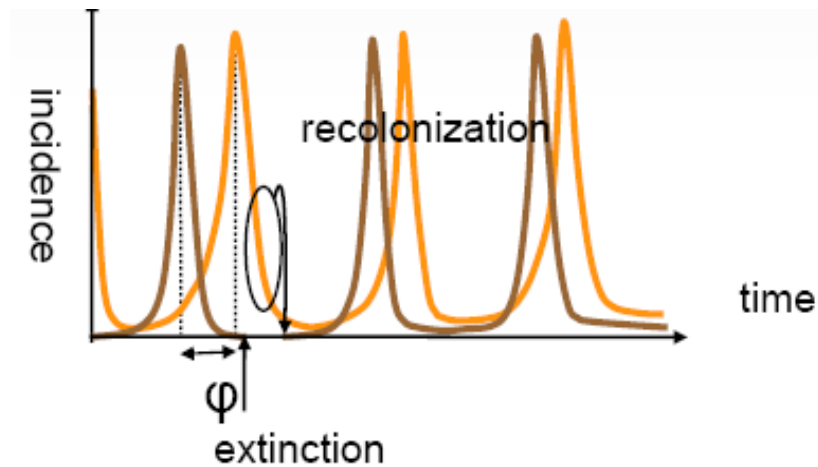
- Trouver la synchronie d'une méta-population contenant deux populations.





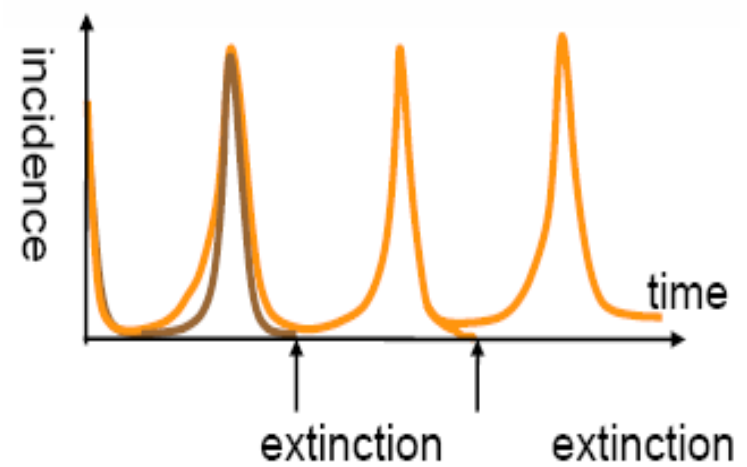
## Asynchronie spatiale

no extinction at larger scale



## Synchronie spatiale

Extinction at larger scale



# Planification pour cette première année

Temps	Travail
01/10/2012 → 30/12/2012	<ul style="list-style-type: none"><li>- Trouver la formule générale de la force d'infection pour le contexte spatial.</li><li>- Faire la modélisation et la simulation sous les langages de programmations.</li><li>- Faire les tests.</li></ul>
01/01/2013 → 30/02/2013	<ul style="list-style-type: none"><li>- Faire l'estimation de la persistance et de la synchronie des maladies infectieuses dans un contexte spatial.</li><li>- Faire les tests.</li></ul>
01/03/2012 → 30/04/2013	<ul style="list-style-type: none"><li>- Chercher les politiques de vaccinations actuelles.</li><li>- Donner le modèle de vaccination.</li></ul>
01/05/2013 → 30/06/2013	<ul style="list-style-type: none"><li>- Faire la modélisation et la simulation du modèle de vaccination.</li><li>- Faire les tests.</li></ul>
01/07/2013 → 30/09/2013	<ul style="list-style-type: none"><li>- Faire la modélisation et la simulation des politiques de vaccinations actuelles.</li><li>- Faire les tests.</li></ul>

# Référence



## REFERENCES

- 1- Earn, D. J.; Rohani, P. & Grenfell, B. T. Persistence, chaos and synchrony in ecology and epidemiology.  
*Proceedings of the Royal Society of London B*, **1998**, 265, 7-10
- 2- Grenfell, B. T.; Bjørnstad, O. N. & Kappey, J. Travelling waves and spatial hierarchies in measles epidemics.  
*Nature*, **2001**, 414, 716-723
- 3- Nokes, D. J. & Swinton, J. Vaccination in pulses: a strategy for global eradication of measles and polio?

# Objectifs Jan-Mai 2013

## package **dizzys**



- Construire le package **dizzys** :
  - Permettre d'intégrer C++ dans R.
  - Installer automatiquement des packages dépendants.
  - Faire des simulations déterministe/stochastique pour les modèles épidémiques SEIR et SIR en utilisant des équations déterministes, l'algorithme stochastique de Gillespie et l'algorithme adaptivetau.
  - Faire les simulations pour n sous-populations dans une méta-population.
  - Interface affiché sous R en 2D, 3D.

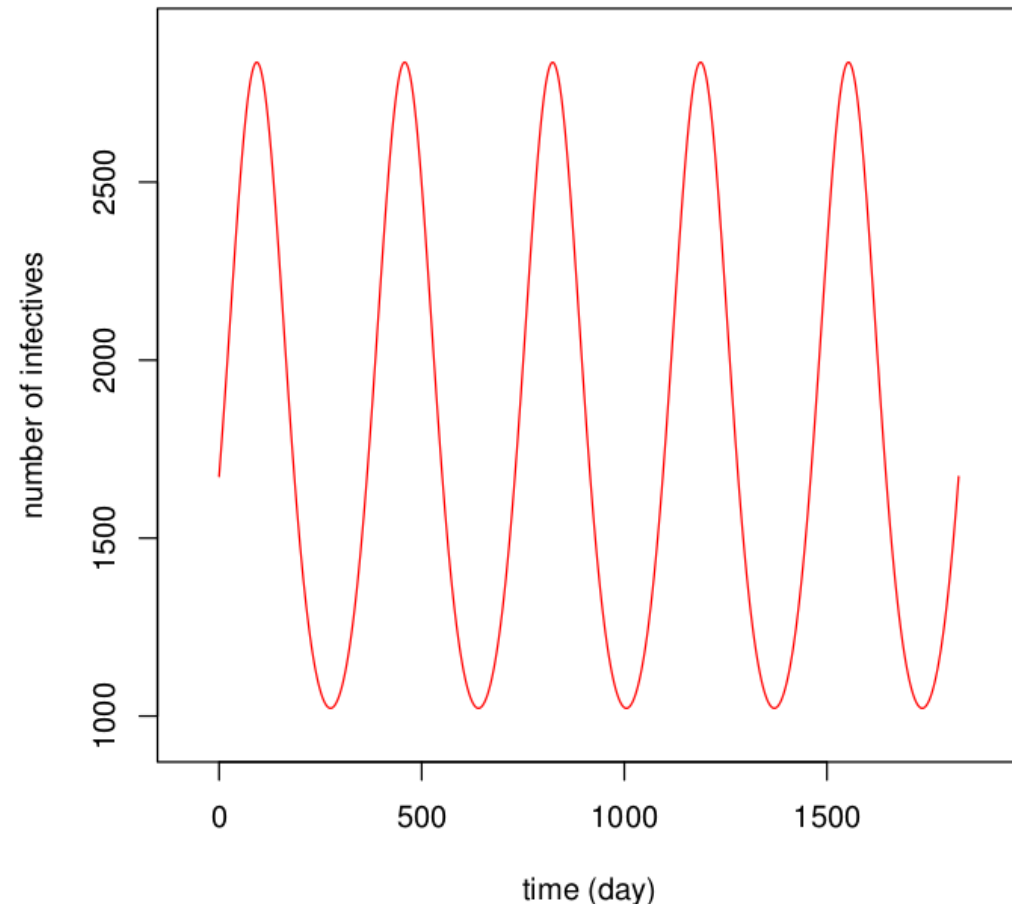


# Objectifs Jan-Mai 2013

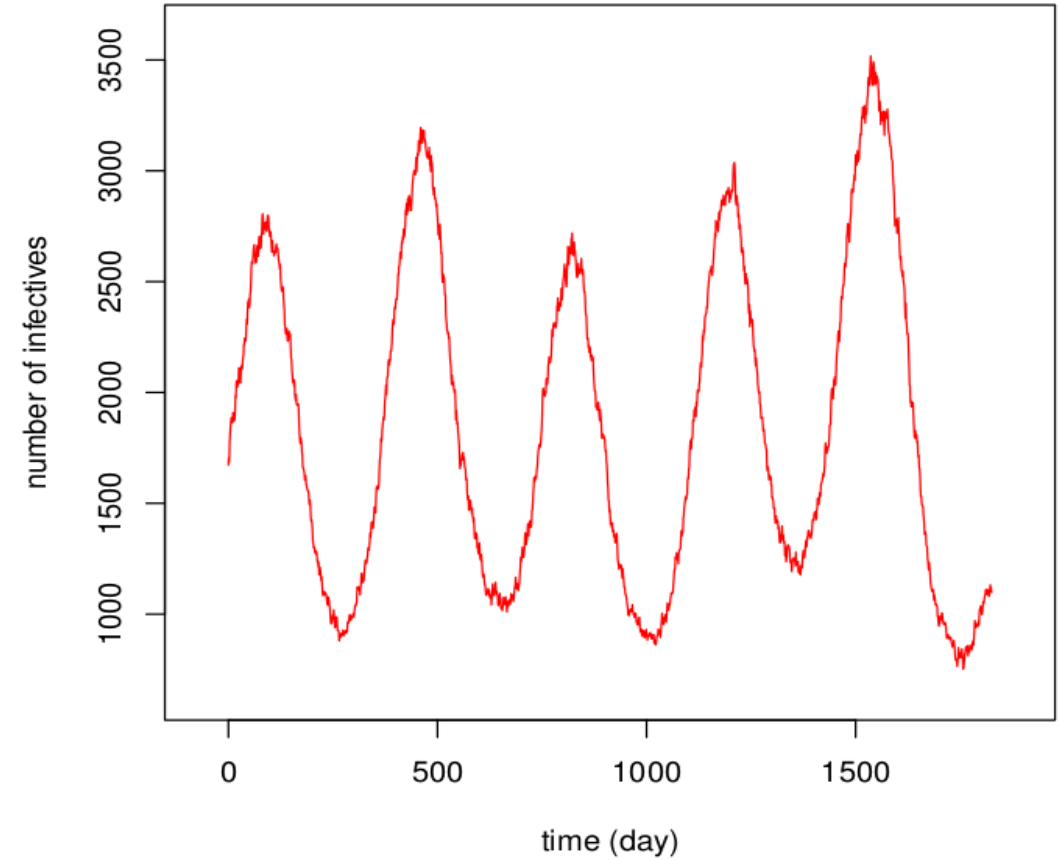
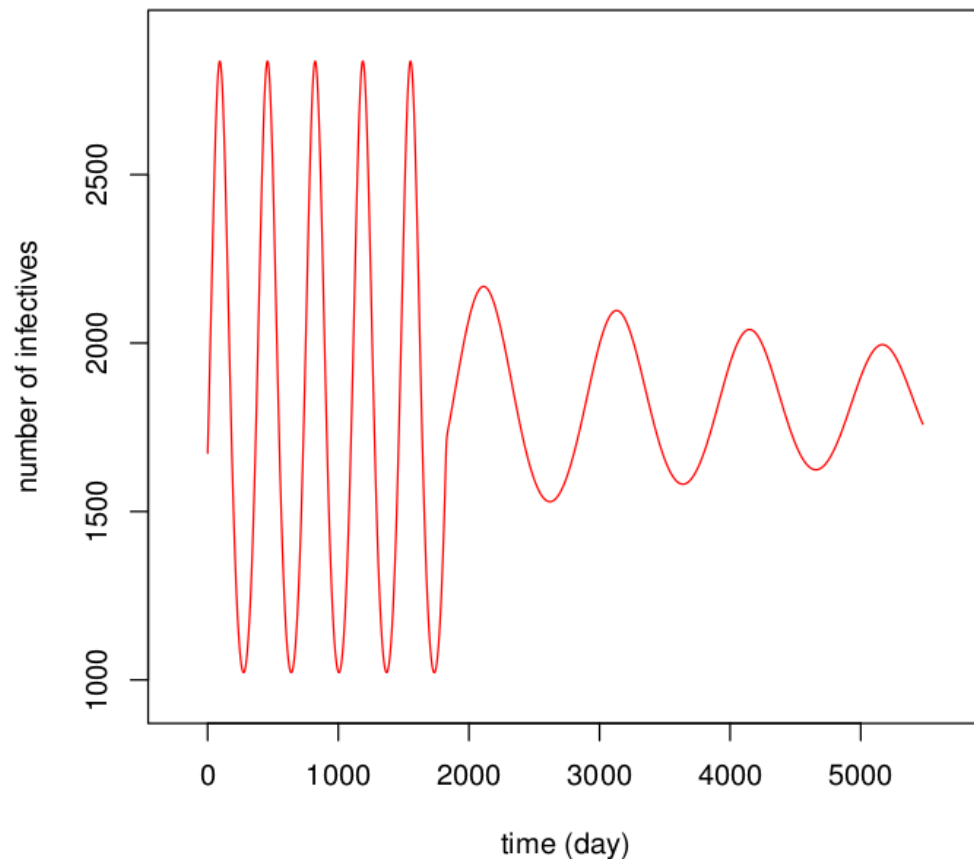
```
> library(dizzys)
> # We have the values of parameters and of variables.
> # Here, we have  $S=E=I=R=NULL$  and  $N=1e7$ .
> # It means that we use  $N=1e7$  to calculate the
> obj<- seir(type="deterministic",duration=5*365,
+ beta1=.1,sigma=1/8,gamma=1/5,
+ T=365,phi=0,nbVilles=1,epsilon=0.0,rho=0.0,S=
```

S.S	E.E	I.P	R.S	N
741559	2795	1673	9253973	10000000

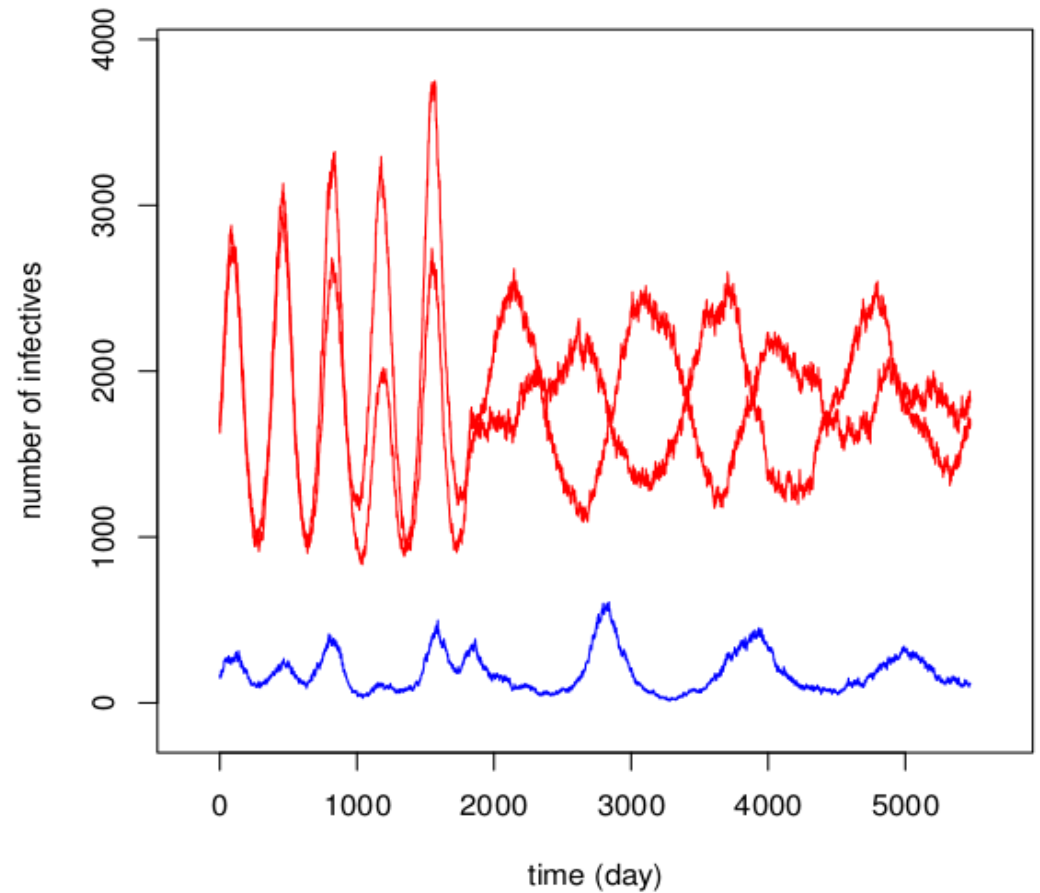
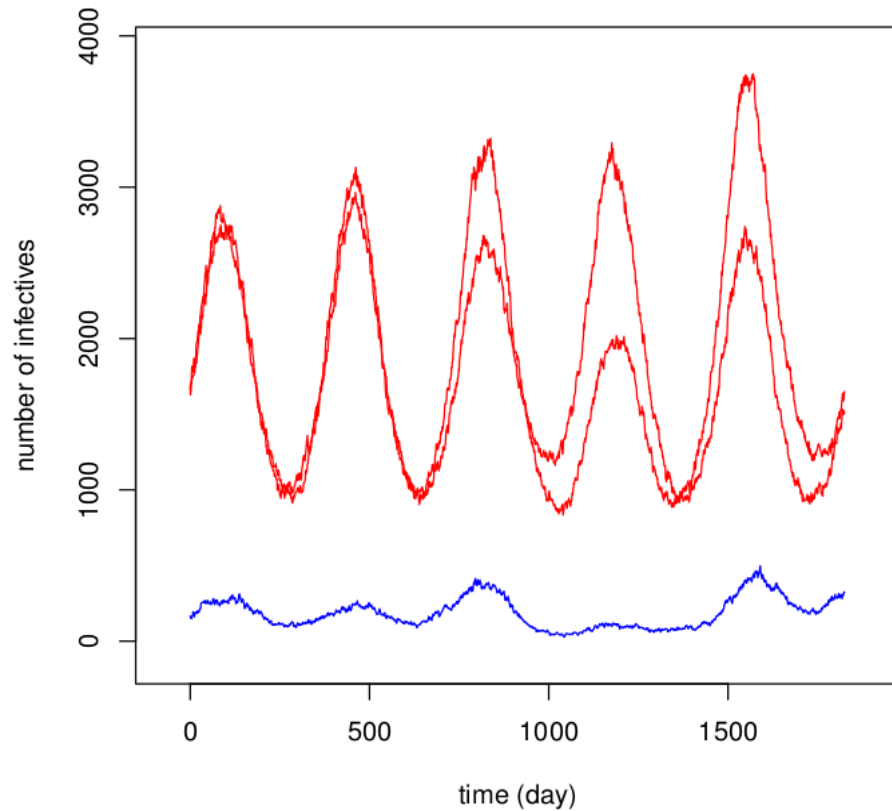
```
> # Use the plot function of the seir class
> plot(obj,col="red",ylab="number of infectives",
```



# Objectifs Jan-Mai 2013



# Objectifs Jan-Mai 2013



# Objectifs Jan-Mai 2013



- **dizzys** maintient des temps d'exécution rapide et des résultats exacts sur les modèles SIR / SEIR avec nombre de sous-populations différents et temps de simulation différents.
- **dizzys** implémente avec succès une solution à la fois exacte et une solution approchée.
- Avec hybride C++/R, simulation est plus rapide que cela mise en œuvre sous R dans le package GillespieSSA.

# Objectifs Juin-Août 2013

## synchronie dans méta-population



- **Mesurer la persistance globale avec le temps**
  - Faire  $N$  simulations identiques (sauf la graine bien entendu...) à  $n$  populations chacune.
  - Regarder les temps d'extinction globale de chaque simulation.
    - utiliser ceci pour tracer une courbe de Kaplan-Meier.

# Objectifs Juin-Août 2013

## synchronie dans méta-population

- Mesurer la persistance globale et le temps

- Test :

`nbsimul=10,nbVilles=2,N=1e6,`

`duration=2*3650,`

`seed=sample(0:10000,10,replace=F),`

`rho=0.001,`

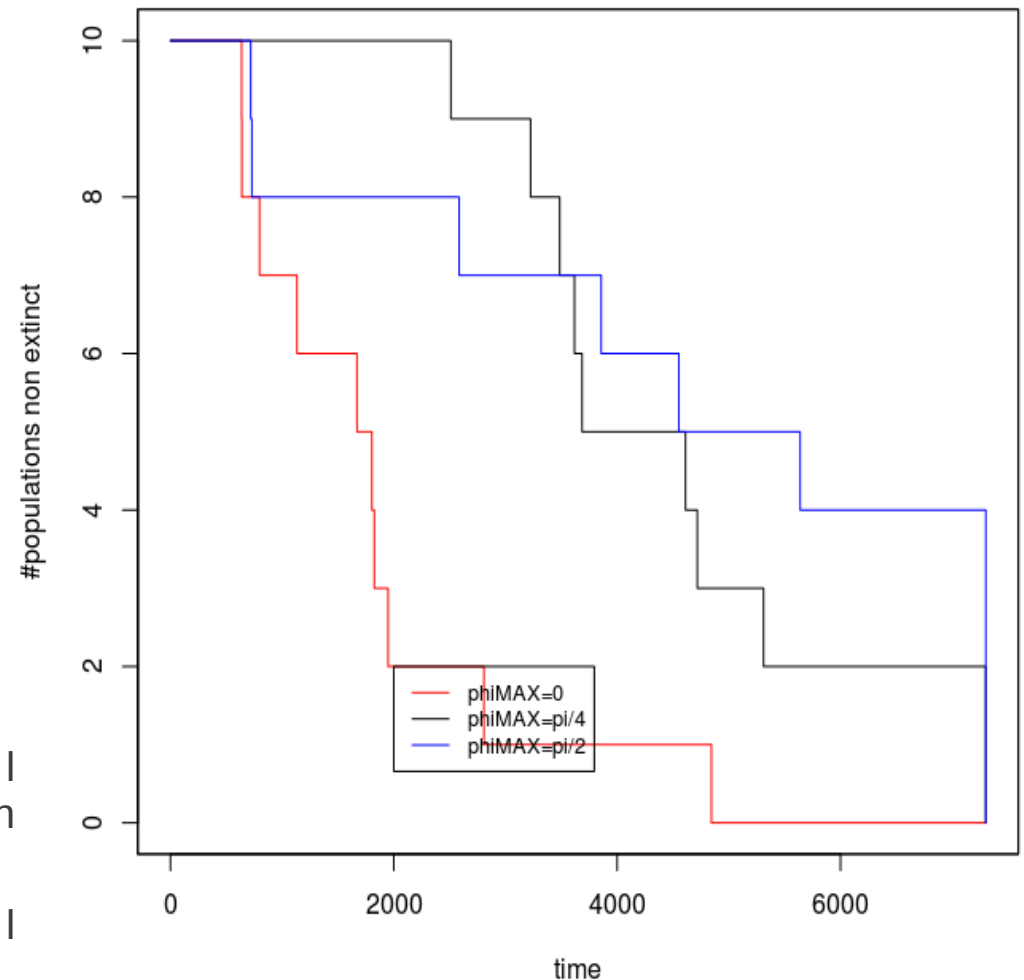
`mu=1/(70*365),beta0=1000/365,beta1=.1,`

`sigma=1/8,gamma=1/5, epsilon=0.0`

`phi_MAX = { 0.0, pi/4, pi/2}.`

- Conclusion :

- Si la valeur de `phi_Max` augmente, alors la vitesse pour que le nombre de population non-extinct arrive à zéro diminue.
    - Si la valeur de `phi_Max` augmente, alors la persistance globale augmente.



# Objectifs Juin-Août 2013

## synchronie dans méta-population

---

- **Mesurer la relation entre le taux de persistance globale et  $\phi_{MAX}$ .**
  - chaque  $\phi_{MAX}$ , faire les répétitions.
    - Après, créer une courbe de survie.
    - Calculer le taux hasard moyen.
    - Obtenir le taux de persistance global pour cette courbe de survie en utilisant la fonction de hasard cumulée  $h(t)$ , le taux de hasard moyen et la régression de COX (les pages 48 et 64 du livre "Survival Analysis").

# Objectifs Juin-Août 2013

## synchronie dans méta-population

- Mesurer la persistance globale et  $\phi_{MAX}$ .

- Test :

```
nbsimul=40,nbVilles=2,N=1e6,  
duration=2*3650,  
seed=sample(0:20000,40,replace=F),  
rho=0.001,  
mu=1/(70*365),beta0=1000/365,beta1=.1,  
sigma=1/8,gamma=1/5, epsilon=0.0  
 $\phi_{MAX} = \{ 0.0, \pi/8, 2*\pi/8, 3*\pi/8, 4*\pi/8 \}.$ 
```

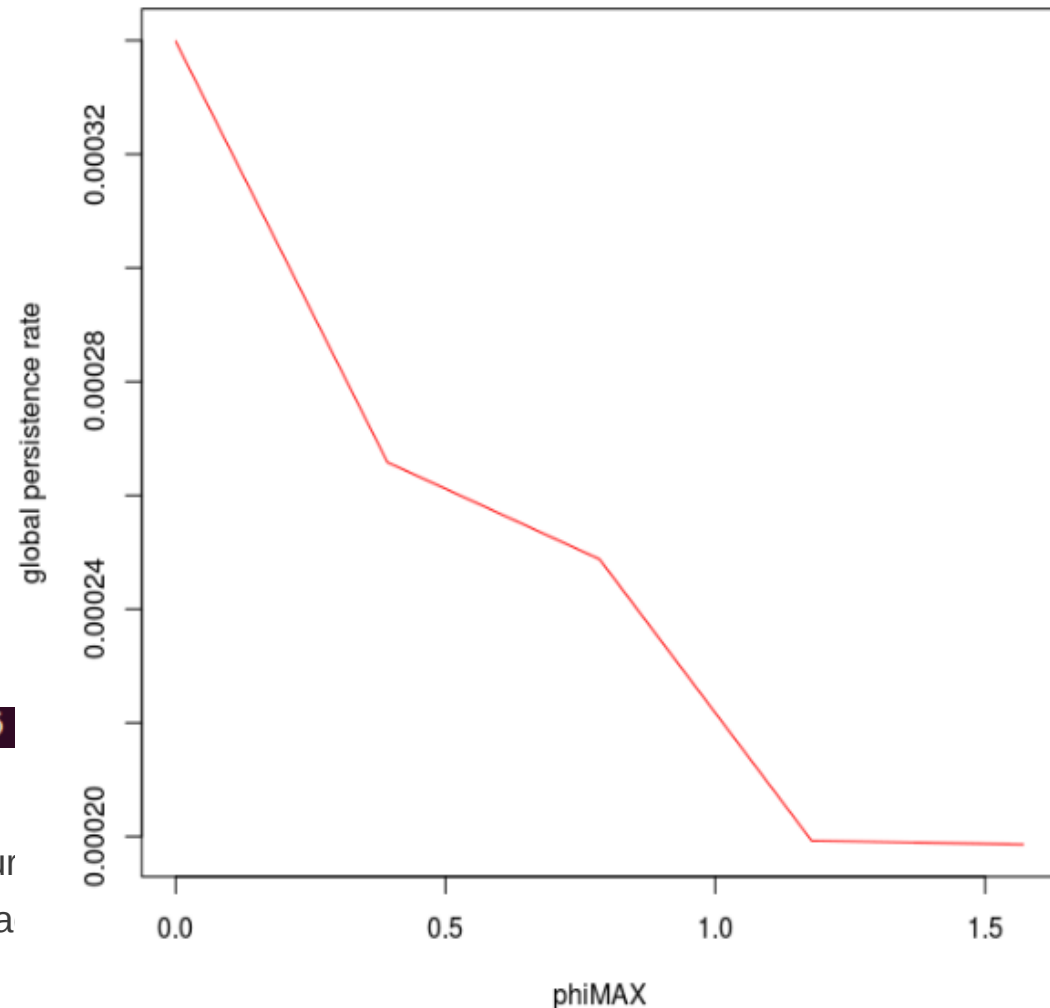
- Résultat :

taux de persistance global =

0.0003398817 0.0002657949 0.0002487686 0.0001992826

- Conclusion :

- Si le taux hasard augmente, alors le temps de sur
- si le nombre de simulations augmente, alors l'exa augmente aussi.





# Objectifs Juin-Août 2013

## synchronie dans méta-population

- Mesurer la relation entre l'intervalle de confiance de taux d'extinction et  $\phi_{MAX}$ .
  - Pour trouver l'intervalle de confiance :
    - Avec chaque valeur de  $\phi_{MAX}$  :
      - avoir la séquence du nombre de répétitions (`seqnbrepetition=seq(5,100,length=20)`).
      - chaque nombre de répétition, obtenir une valeur de taux d'extinction, alors, avec une séquence du nombre de répétitions, on obtient aussi une séquence des valeur de taux d'extinction  $X=\{x_1, x_2, x_3, \dots, x_n\}$ .
    - Après, utiliser les formules statistiques pour trouver la valeur moyenne **xmean**, la variance et l'écart-type **s**.
    - Voilà, l'intervalle de confiance à 95% est:  
$$[xmean - 1,96 * s / \sqrt{n}; xmean + 1,96 * s / \sqrt{n}]$$

# Objectifs Juin-Août 2013

## synchronie dans méta-population

- Mesurer la persistance globale

et  $\phi_{MAX}$ .

- Test :

`seqnbrepetition= seq(5,100,length=20)`

`nbVilles=2,N=1e6,`

`duration=2*3650,`

`seed=sample(0:20000,40,replace=F),`

`rho=0.001,`

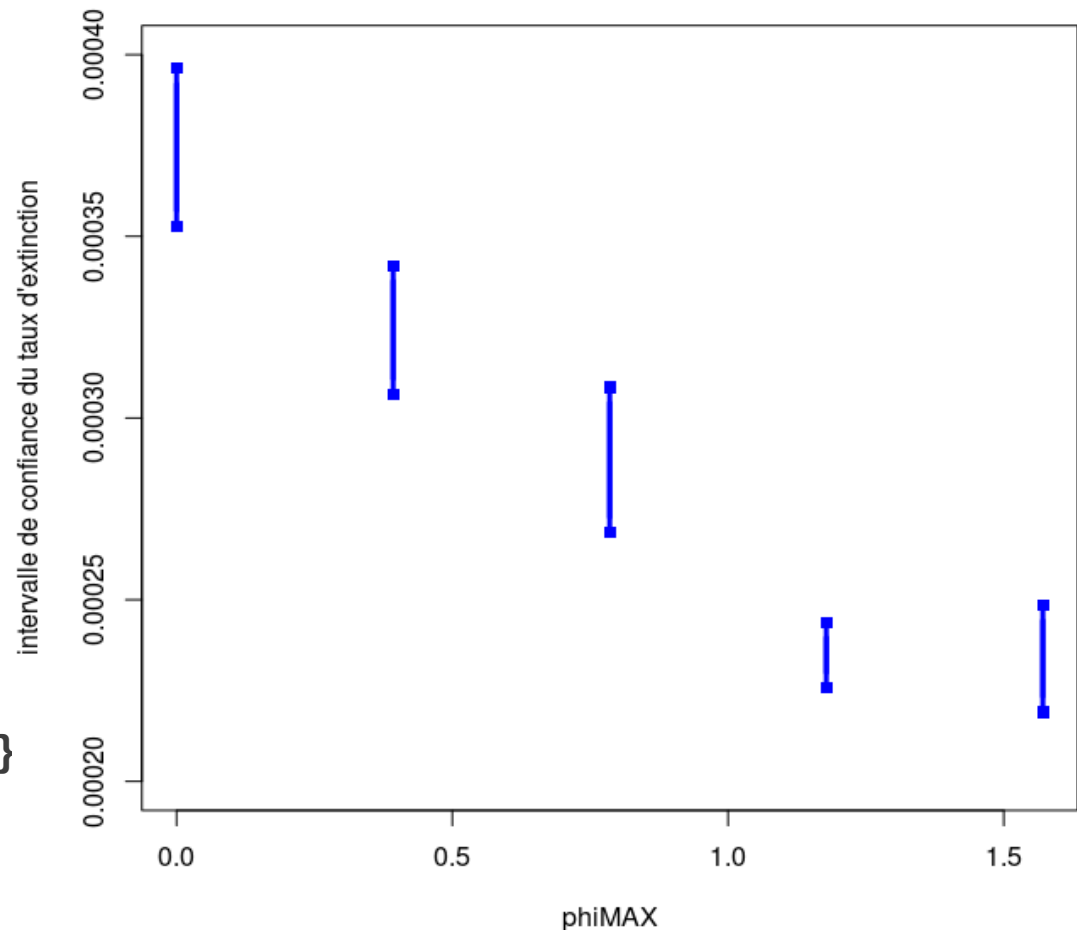
`mu=1/(70*365),beta0=1000/365,beta1=.1,`

`sigma=1/8,gamma=1/5, epsilon=0.0`

`$\phi_{MAX} = \{ 0.0, \pi/8, 2*\pi/8, 3*\pi/8, 4*\pi/8 \}$`

- Conclusion :

- si le nombre de simulations augmente, alors l'exactitude augmente aussi.



# Recherches (1)

- Déjà prouver que la formule est vraie.

Consider that in the metapopulation the phases  $\varphi_i$  of the contact rates in the  $n$  subpopulations are evenly distributed between 0 and  $\varphi_{\max}$  ( $0 \leq \varphi_{\max} \leq \pi$ ). We can express the mean of the pairwise phase differences  $\delta_{ij} = \delta_{ji}$  as

$$\langle \delta_{ij} \rangle = \langle \delta_{ji} \rangle = 2\varphi_{\max} \sum_{k=1}^{n-1} \frac{(n-k)k}{(n-1)n^2} = \frac{n+1}{3n} \varphi_{\max} \quad (12)$$

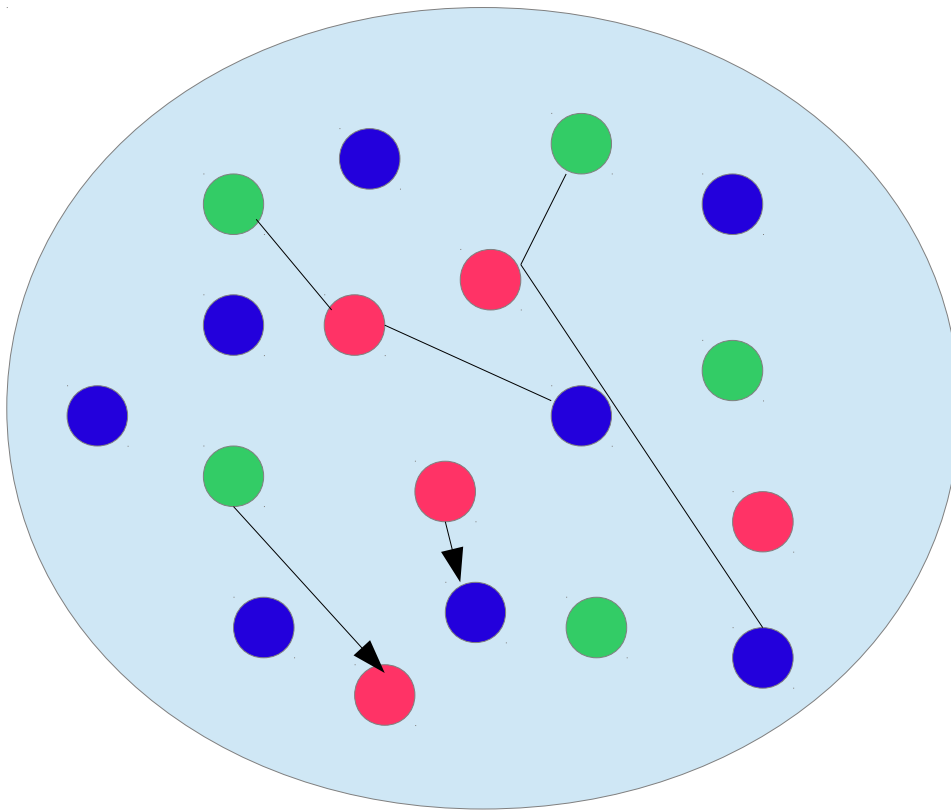
# Recherches (2)



- Hladish2012 « EpiFire : An open source C++ library and application for contact network epidemiology »
  - Source:  
<http://dash.harvard.edu/bitstream/handle/1/10579231/3496>
  - C'est une application sur l'interface de programmation EpiFire qui nous permet de générer et analyser des réseaux en forme de populations de maladies infectieuses en utilisant le modèle SIR.

# Recherches (2)

- Un réseau EpiFire = une population = un modèle épidémique en individu-centré.



- + ) une population = un réseau d'individuels
- + ) 1 individu = un nœud
- + ) les contacts à travers des arêtes entre des nœuds.
- + ) 1 degré de sommet = le nombre de contact que cet individu a.
- + ) Transmission épidémique utilise deux approches différentes "the Reed-Frost chain-binomial model" et "the percolation model".

# Recherches (2)

- Idée proposée :
  - Utiliser les deux approches ci-dessus "the Reed-Frost chain-binomial model" et "the percolation model", pour le problème de relation entre des subpopulations dans une metapopulation.
  - une méta-population = une grande population
  - des sous-populations = des sommets
  - 01 sommet a un nombre entier qui est le nombre de contacts que cette sous-population a.
  - Selon "the percolation model" :
    - **rho** = le taux de coupling entre le sommet i et son voisin j.
    - pour chaque voisin de sommet, on génère un nombre aléatoire uniforme **rand** entre 0 et 1, si **rand** < **rho** et les sommets i et j se connectent, la capacité d'infection entre i et j par le coupling se produit.

# Recherches (3)

- Problème :
  - Selon Marc, la méthode de Gillespie n'est exacte que pour les systèmes sans forçage. Dans le cas d'un forçage, l'utilisation de l'algorithme de Gillespie ne donnera qu'une simulation approchée.
  - Selon Rozhnova2011, on peut utiliser « **master equation** », « **algo de Gillespie** » et « **system-size expansion of van Kampen** ».
  - Rohani1999, déjà utilisé l'algo de Gillespie pour une metapopulations.
- **Anderson2007** « A modified next reaction method for simulating chemical systems with time dependent propensities and delays »
  - Source : <http://www.math.wisc.edu/~anderson/papers/AndNRM.pdf>
  - Contexte :
    - Algo de Gillespie : deux nombres aléatoires demandés pour chaque étape. Si  $t_{max}$  est grand, cet algorithme génère trop de nombres aléatoires. A cause de cette raison, les générateurs de nombres pseudo-aléatoires fait le cycle avec quelques régularités, utilisant trop beaucoup de nombres aléatoires épuiserait vite les capacités du générateur, et devrait être évité avec préjudice extrême.
  - « Modified next reaction method » :
    - n'utilise que un seul nombre aléatoire par étape. Alors, le temps de calcul est moins.
    - n'est pas beaucoup différent avec l'algo de Gillespie.

# Recherches



- Simuler des systèmes chimiques basés sur des conditions de processus de Markov.
  - **Monte-Carlo Simulation Methods**
    - Stochastic Simulation Algorithm : Gillespie1977, GibsonBruck(2000)
      - First Reaction Method (Gillespie '77)
      - Next Reaction Method (Gibson and Bruck '00)
      - Direct Method (Gillespie '77)
    - Tau-leaping : Gillespie2001, Gillespie2003, Cao2005
    - System Partitioning Methods
      - Fast--Slow Partitions : Rao2003, Cao2005,
      - Continuous--Discrete Partitions : Haseltine2002, Salis2005.
  - **Density Computations using the Finite State Projection**
    - Revoir : Chemical master equation
    - The Finite State Projection Algorithm : Munsky2006



# Recherches (4)

- Munsky2006 « The finite state projection (FSP) algorithm for the solution of the chemical master equation »

## **Avantages:**

- Selon cet article, cette méthode est plus efficace que les méthodes de Monte Carlo parce qu'elle fournit une solution directe ou approximation de la CME sans générer des simulations comme Monte Carlo. De plus, le FSP propose une garantie quant à son exactitude.
- Pour déterministe.
  - Chaque simulation de la FSP donne le même résultat.
  - Permet des comparaisons plus faciles de différents systèmes.
- Ne dépend pas des conditions initiales.

## **Désavantages:**

- A cause de l'utilisation de CME, pour un grand système,
- Raideur numérique peut conduire à l'inefficacité de calcul.
- Les systèmes peuvent devenir très volumineux comme des distributions couvrent de vastes régions de l'espace de configuration.
  - Distributions compactes peuvent dériver au fil du temps.
  - Distributions diluées peuvent se propager sur de grandes régions.

# Recherches (4)

- Munsky2006 « The finite state projection algorithm for the solution of the chemical master equation »

## **Avantages:**

- Selon cet article, cette méthode est plus efficace que les méthodes de Monte Carlo parce qu'elle fournit une solution directe ou approximation de la CME sans générer des simulations comme Monte Carlo. De plus, le FSP propose une garantie quant à son exactitude.
- Pour déterministe.
  - Chaque simulation de la FSP donne le même résultat.
  - Permet des comparaisons plus faciles de différents systèmes.
- Ne dépend pas des conditions initiales.

## **Désavantages:**


- A cause de l'utilisation de CME, pour un grand système,
- Raideur numérique peut conduire à l'inefficacité de calcul.
- Les systèmes peuvent devenir très volumineux comme des distributions couvrent de vastes régions de l'espace de configuration.
  - Distributions compactes peuvent dériver au fil du temps.
  - Distributions diluées peuvent se propager sur de grandes régions.

# Recherches (5)



- Cao2006 « Efficient step size selection for the tau-leaping simulation method »
  - Donner l'algorithme "tau-leaping" de la classe Monte Carlo, avec la bonne méthode qui estime la valeur la plus grande pour le paramètre tau qui est satisfait de toutes conditions de simulations.
  - Tau-leaping est plus rapide que l'algo de Gillespie.
    - Parce que chaque réaction est simulée individuellement dans l'algorithme de Gillespie, par contre on trouve des pas à sauter dans tau-leaping.
    - avec tau-leaping, le nombre Poisson sont sans limite, alors les fonctions de propension peuvent changer radicalement au cours de petits intervalles de temps et peut montrer des populations négatives.

# Maintenant



Je suis en train de commencer à écrit une article sur la persistance global.