dizzys: efficient deterministic/stochastic simulations in R for a metapopulation by using SIR/SEIR models

TRAN Thi Cam Giang

June 13, 2013

Abstract

Predicting the potential spread of an infectious disease is still a difficult problem for scientists. It requires much more than simple connecting subpopulations in a metapopulation and takes into account many factors about the pathogen and the affected subpopulation. Therefore, this 'dizzys' package allows us to simulate dynamics of an infectious disease through subpopulations by using the SIR/SEIR models and by implementing the direct algorithm of Gillespie in 1977 and the adaptive tau leaping to approximate the trajectory of a continuous-time stochastic process. Consequently, result returned is biological data in time horizon about the disease dynamic, we can perform analysis on this biological data. This vignette presents a few examples of SIR/SEIR applied to biological problems.

Introduction

Fundamentally, Kermack-McKendrick gave the first epidemic model to provide a mathematical description of the kinetic transmission of an infectious disease in an unstructured subpopulation. According to this model, today we have known well the SIR/SEIR deterministic epidemic models. This is the two basic models very popularly used by scientists. However, Keeling2008 [4] show that all the deterministic models are essentially fixed "clockwork" systems with the same starting conditions, exactly the same trajectory is always observed. It isn't right for dynamics of real pathogens in the real-world. So stochastic models are created and concerned with approximating or mimicking the random or probabilistic element from the deterministic models. Moreover, when the quantities in a system are small enough and extinction is probable to occur, then stochastic effects become critical to take into account. This is reason, in the 'dizzys' package, it permits us to obtain the dynamics of the deterministic and the approximate dynamics of the stochastic epidemic models.

Based on the stochastic models, their processes are in Markov process, it means that the future state of the process, conditional on the present state, is independent of the past. In the case, our package focus on simulating dynamics from a continuous-time Markov process for which the transition rates are constants, isn't a function of time. We use the exact algorithm of Gillespie in 1977 and the approximate algorithm described as the "adaptive tau-leaping algorithm". With these two algorithms, each has its private advantages and its private disadvantages. For the exact algorithm, it give us a really exact approach of simulating population-based time-to-event through two step with many iterations of 1) searching the time of next event by an exponentially distributed function and 2) searching the nature of next event. This Gillespie's solution becomes too slow and impractical as any one transition rate grows large. Hence, approximate models are born instead of the Gillespie's solution, they are concerned with larger transition rates and with increasing simulation speed while still maintaining reasonable accuracy. The "adaptive tau-leaping algorithm" known as an approximate method reduces the number of iterations by treating transition rates as constant over time periods for which this approximation leads to little error[5] .

The **dizzys** package in R implements both the exact solution and the approximate solution for the SIR and SEIR models by integrating the R package and the C++ implementation. We can choose one of the two solutions to simulate when the number of subpopulations in a metapopulation increases. We use C++ to perform the algorithms, in contrast, we use R to create interfaces. Therefore, new implementation is much faster than any pure R implementation.

Methods

In this section, first we will talk about the deterministic model, the stochastic model of the SEIR model. Then, we will have transformation the SEIR model into the SIR model through the usage of the two algorithms. We hope that the models and the algorithms should be well understood before obtainning simulation results.

Deterministic model:

To describe infectious diseases in a in a spatial context, we consider a metapopulation of n sub-populations. In subpopulation i of size N_i, disease dynamics can be deterministically described by the following set of differential equations:

$$\begin{aligned}
\frac{dS_i}{dt} &= \mu N_i - \lambda_i S_i - \mu S_i \qquad (1)\\
\frac{dE_i}{dt} &= \lambda_i S_i - \mu E_i - \sigma E_i \qquad (2)\\
\frac{dI_i}{dt} &= \sigma E_i - \mu I_i - \gamma I_i \qquad (3)\\
\frac{dR_i}{dt} &= \gamma I_i - \mu R_i \qquad (4)
\end{aligned}$$

where S_i, E_i, I_i et R_i are respectively the numbers of susceptible, exposed, infectious and recovered in this sub-population i. Individuals are born susceptible and die at a rate μ, become infected with the force of infection λ_i, infectious after a latency period of an average duration of $1/\sigma$ and recover at the rate γ. In case the infectious contact rate is constant, the equilibrium values of the variables S, E, I and R can be expressed analytically (see appendix). The force of infection depends not only on the total population size N_i and the number of infected I_i in subpopulation i, but also in other sub-populations :

$$\lambda_i = \left(1 - \sum_{\substack{k=1\\k\neq i}}^{n} \rho_{ik}\right) \beta_i \frac{I_i}{N_i} + \sum_{\substack{k=1\\k\neq i}}^{n} \rho_{ik} \frac{(1-\varepsilon_{ik})\beta_i N_k + \varepsilon_{ik}\beta_k N_i}{N_i N_k} I_k \qquad (5)$$

where $\sum_{\substack{k=1\\k\neq i}}^{n} \rho_{ik} < 1$, β_i is the contact rate in population i and $\rho_{ij} = \rho_{ji}$ ($0 \leqslant \rho_{ij} \leqslant 1$ and $\rho_{ii} = 1$) is the coupling between subpopulations i and j. Among the infections caused by contacts with infected from other subpopulations, $\varepsilon_{ij} = \varepsilon_{ji}$ ($0 \leqslant \varepsilon_{ij} \leqslant 1$) is the proportion of infections due to susceptible individuals visiting other populations as opposed to infected individuals from other populations visiting the focal population. See appendix for detail on the construction of this equation. We can verify that in the limit case on one single subpopulation in the metapopulation ($i = j$ et $n = 1$) we have

$$\lambda_i = \beta_i \frac{I_i}{N_i}. \qquad (6)$$

consider that the contact rate β_i is seasonally forced [3]:

$$\beta_i(t) = b_0 \left[1 + b_1 \cos\left(\frac{2\pi t}{T} + \varphi_i\right)\right] \qquad (7)$$

where b_0 and b_1 are the mean value and amplitude of the contact rate and T and φ_i are the period and the phase of the forcing.

Stochastic model using Gillespie's exact algorithm:

Based on the differential equations above, we give a stochastic version of this model. We use for that a population-based time-to-next-event model based on Gillespie's algorithm [2]. Table 1 lists all the events of the model, occurring in subpopulation i.

Table 1: Events of the stochastic version of the model of equations 1-4, occuring in subpopulation i.

Events	**Rates**	**Transitions**
birth	μN_i	$S_i \leftarrow S_i + 1$ and $N_i \leftarrow N_i + 1$
death of a susceptible	μS_i	$S_i \leftarrow S_i - 1$
death of an exposed	μE_i	$E_i \leftarrow E_i - 1$
death of an infected	μI_i	$I_i \leftarrow I_i - 1$
death of an immune	μR_i	$I_i \leftarrow I_i - 1$
infection	$\lambda_i S_i$	$S_i \leftarrow S_i - 1$ and $E_i \leftarrow E_i + 1$
becoming infectious	σE_i	$E_i \leftarrow E_i - 1$ and $I_i \leftarrow I_i + 1$
recovery	γI_i	$I_i \leftarrow I_i - 1$ and $R_i \leftarrow R_i + 1$

Stochastic model using “adaptive tau-leaping algorithm”:

In this step, we provide basic concepts for the adaptive tau-leaping algorithm by using the detailed description of Cao[5].

For the Markov process at time t, to describe a metapopulation of n subpopulations, we have:

state set: X(t)

$X(t) := [S_1(t), S_2(t), ..., S_n(t), E_1(t), E_2(t), ..., E_n(t), I_1(t), I_2(t), ..., I_n(t), R_1(t), R_2(t), ..., R_n(t)]$

each variables of $X(t)$ is defined on the non-negative integers.

set of allowable transitions: $\triangle_j$, for each allowable transition, j, we define a rate λ_j, by using a function independent on t but dependent on the current state $X(t)$, to calculate transition rates given the state ($\lambda(X)$) through the deterministic model, and a vector of n integers, $\triangle_j := [\triangle_{j,1}, ..., \triangle_{j,n}]$, that reflects the change in state if this transition were followed: $X(t) + \triangle_j$.

time process: modeling on a time-homogeneous process.

operation: with the SEIR model, the package simulates a trajectory from time 0 to a stopping time $tmax$. Based on the description of Cao and al.[2007], a good time period of length τ is during which all transition rates remain approximately constant and all n state variables remain greater than zero with probability$\sim$ 1. Then, by using the Poisson-dustributed number of transitions, that should have occurred during this period: $X(t+\tau) \approx X(t) + \sum_j y_j \triangle_j$ where $y_j \sim Poisson(\tau\lambda_j)$. To successfully apply this algorithm, we need to know that, transition rates frequently change and in balancing efficiency with accuracy when selecting these time periods to leap over.

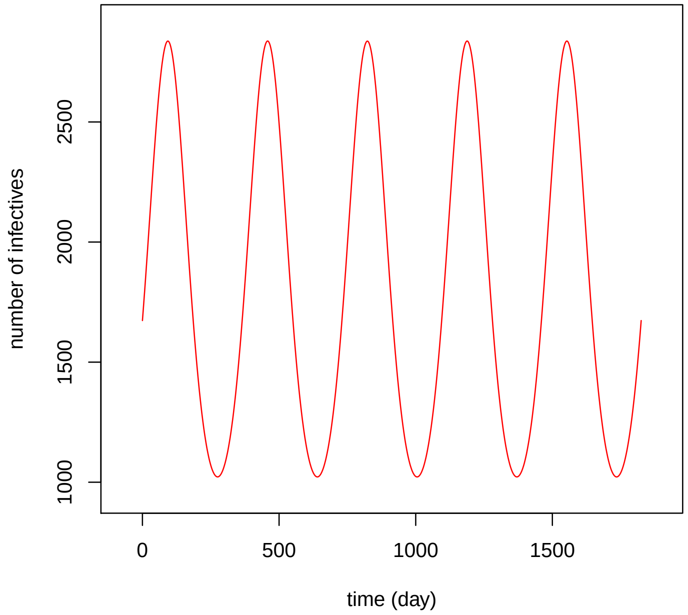
Transformation SEIR model into SIR model:

The SIR model used in this package is the SIR model with births and death**.** By observing this SEIR model, if we give a numerical value for the parameter σ then a SEIR model would have. On the other side, if we give Inf (to infinity) the parameter σ then we have a SIR model with birth and death (because, basically, a SEIR model tends to a SIR model when σ tends to infinity).

Example 1

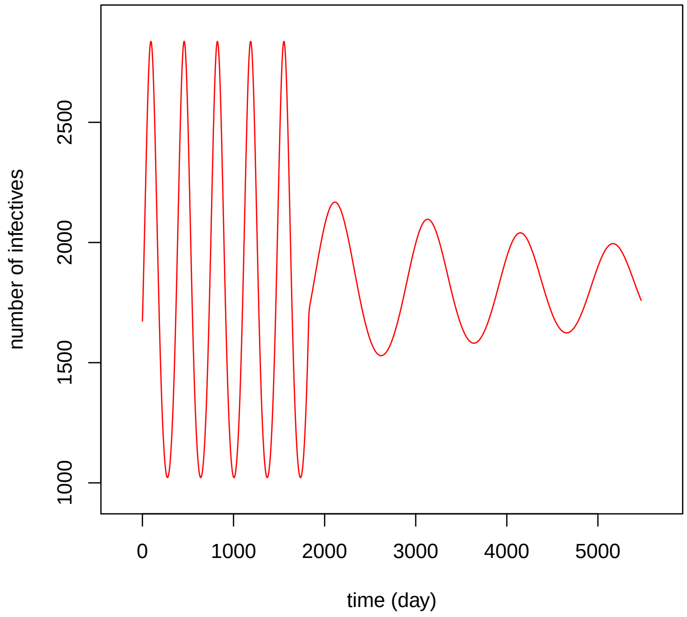
The deterministic SEIR model with one subpopulation by exploiting the 'seir' function in the package.

```
> 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 equilibrium values of variables.
> obj<- seir(type="deterministic",duration=5*365,mu=1/(70*365),beta0=1000/365,
+ beta1=.1,sigma=1/8,gamma=1/5,
+ T=365,phi=0,nbVilles=1,epsilon=0.0,rho=0.0,S=NULL,E=NULL,I=NULL,R=NULL,N=1e7)
> # Use the plot function of the seir class
> plot(obj,col="red",ylab="number of infectives", xlab="time (day)")
```



Now, we want to continue or to redo this simulation with other values of parameter, we can do it by exploiting the 'simul' function in the package.

```
> newobj<- simul(obj,duration= 10*365,continue=T, append=T, beta1=0.0, phi=pi/2)
> plot(newobj,col="red",ylab="number of infectives", xlab="time (day)")
>
```

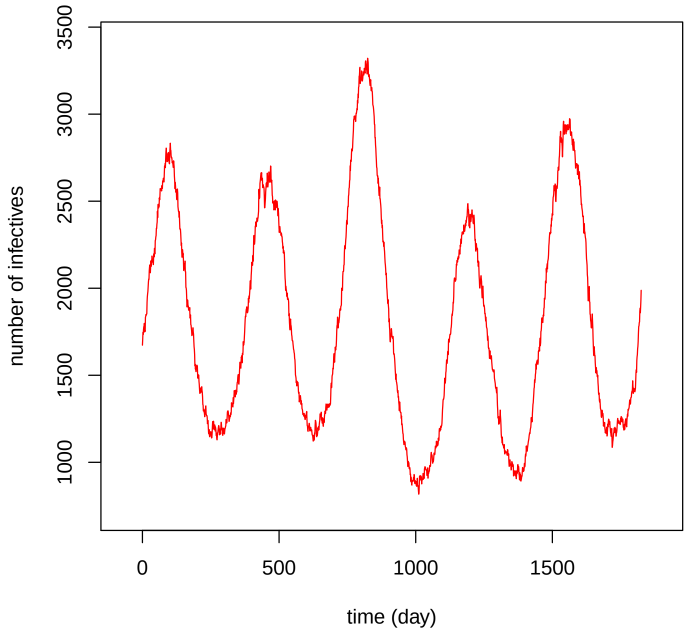


Example 2

The SEIR stochastic model using Gillespie's algorithm by using the 'seir' function.

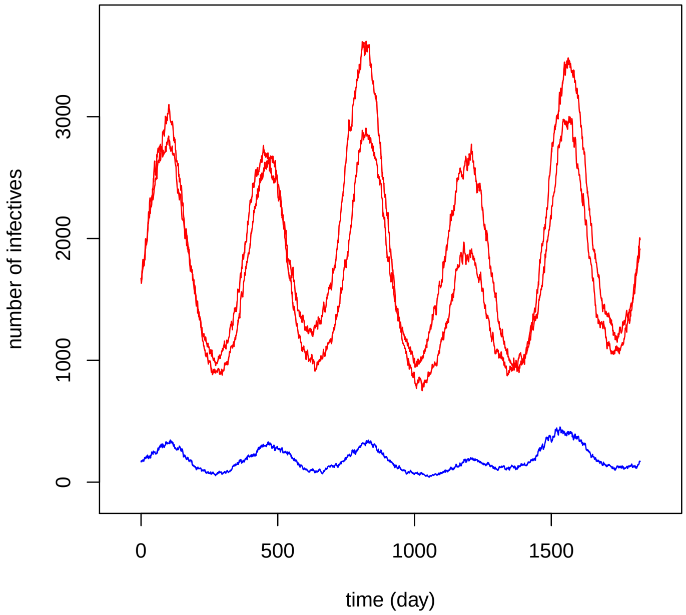
1) with one subpopulation:

```
> obj<- seir(type="stochastic",method="direct",duration=5*365,mu=1/(70*365),
+ beta0=1000/365,beta1=.1,sigma=1/8,gamma=1/5,T=365,phi=0,
+ nbVilles=1,epsilon=0.0,rho=0.0,S=NULL,E=NULL,I=NULL,R=NULL,N=1e7)
> plot(obj,col="red",ylab="number of infectives", xlab="time (day)")
```



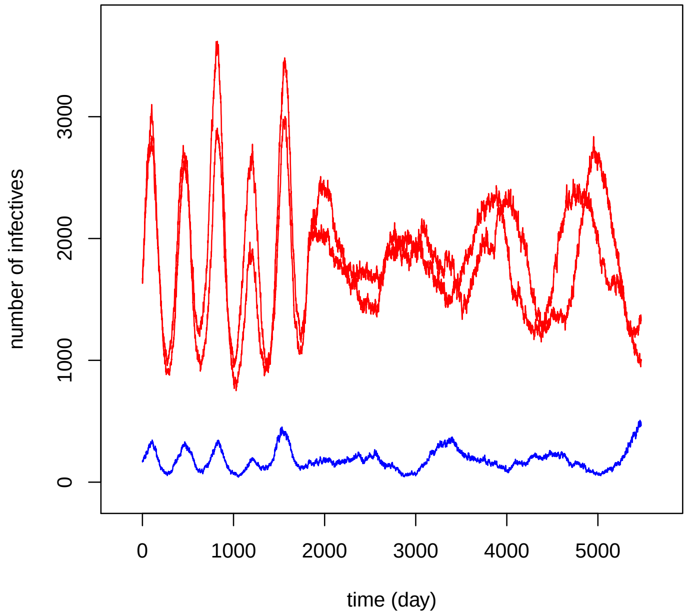
2) with three subpopulations and the different number of populations.

```
> obj<- seir(type="stochastic",duration=5*365,mu=1/(70*365),beta0=1000/365,beta1=.1,
+ sigma=1/8,gamma=1/5,nbVilles=3,N=c(1e7,1e6))
> plot(obj,col=c("red","blue"),ylab="number of infectives", xlab="time (day)")
```



3) continue or redo this siluation with other values of parameter, we can do it by exploiting the 'simul' function in the package.

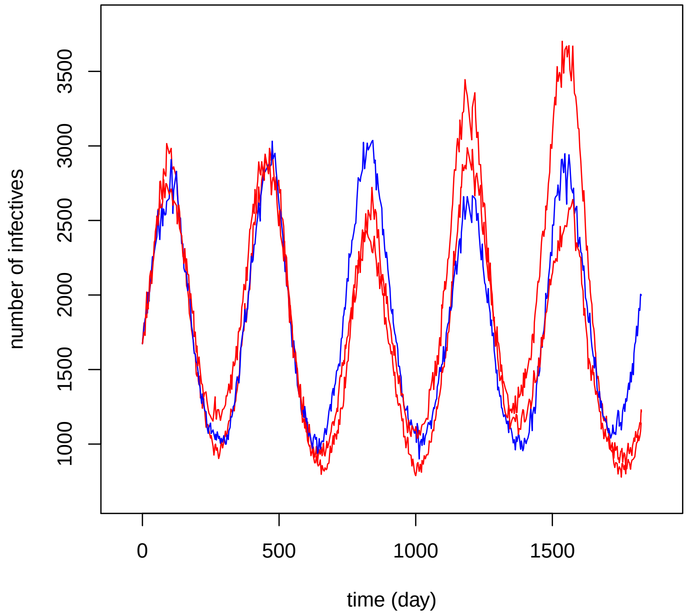
```
> newobj<- simul(obj,duration= 10*365,type="stoch",continue=T,append=T,beta1=0.0,phi=pi/2)
> plot(newobj,col=c("red","blue"),ylab="number of infectives",xlab="time (day)")
>
```



Example 3

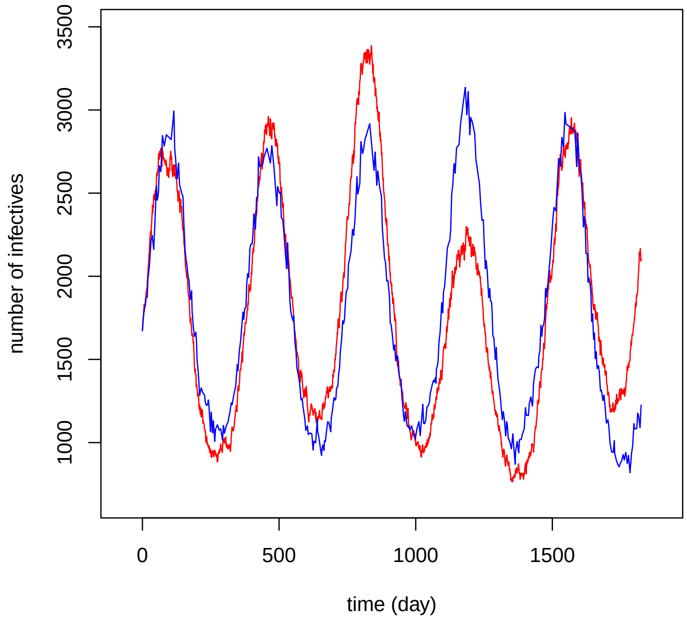
The SEIR stochastic model using “adaptive tau-leaping algorithm” by exploiting the ’seir’ function. To do this algorithm, we only make the parameter method="adaptivetau" as follows:

```
> obj<- seir(type="stochastic",method="adaptivetau",duration=5*365,mu=1/(70*365),
+ beta0=1000/365,beta1=.1,sigma=1/8,gamma=1/5,nbVilles=3,N=1e7)
> plot(obj,col=c("red","blue"),ylab="number of infectives", xlab="time (day)")
```



We can compare the result of the Gillespie'algorithm with the result of the adaptivetau algorithm:

```
> #obj1 with method="direct"
> obj1<- seir(type="stochastic",method="direct",duration=5*365,mu=1/(70*365),
+ beta0=1000/365,beta1=.1,sigma=1/8,gamma=1/5,nbVilles=1,N=1e7)
> plot(obj1,col="red",ylab="number of infectives", xlab="time (day)")
> #obj2 with method="adaptivetau"
> obj2<- seir(type="stochastic",method="adaptivetau",duration=5*365,mu=1/(70*365),
+ beta0=1000/365,beta1=.1,sigma=1/8,gamma=1/5,nbVilles=1,N=1e7)
> plot(obj2,col="blue",add=T)
```



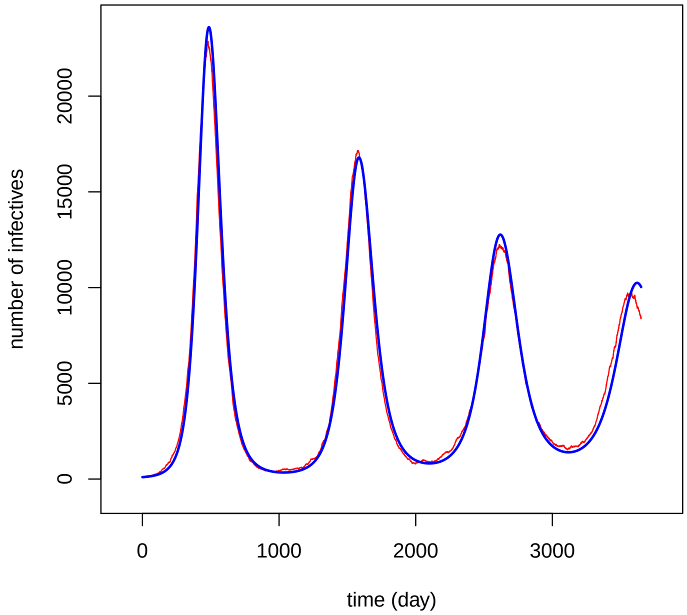
Example 4

In this example, we talk about the SIR model by exploiting the 'seir' function. To do this model, we only make the parameter "sigma=Inf". We refer to the book of Black2010 [1].

1) Comparing deterministic SIR model with stochastic SIR model.

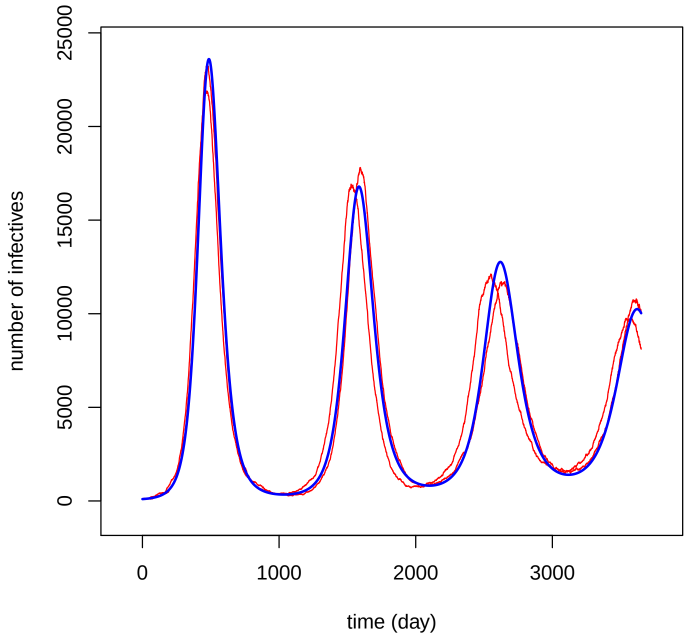
One subpopulation:

```
> S=700000; E=0; I=100; R=9299900;
> plot(seir(duration=10*365,type="stoch",S=S,E=E,I=I,R=R,beta0=1.175,beta1=0.0,
+ sigma=Inf,gamma=0.077),col="red",ylab="number of infectives", xlab="time (day)")
> plot(seir(duration=10*365,type="deter",S=S,E=E,I=I,R=R,beta0=1.175,
+ beta1=0.00,sigma=Inf,gamma=0.077),col="blue",add=T,lwd=2)
```



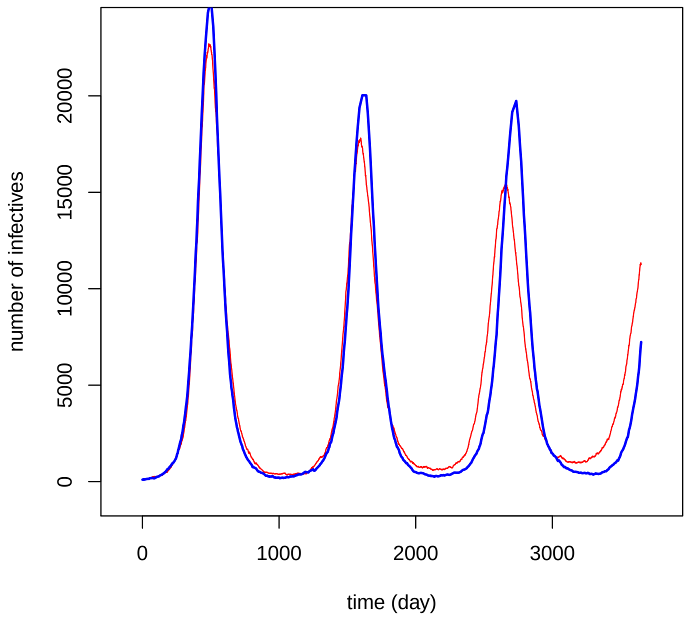
Two subpopulations:

```
> plot(seir(duration=10*365,type="stoch",nbVilles=2,S=S,E=E,I=I,R=R,beta0=1.175,
+ beta1=0.0,sigma=Inf,gamma=0.077),col="red",ylab="number of infectives", xlab="time (day)")
> plot(seir(duration=10*365,type="deter",S=S,E=E,I=I,R=R,beta0=1.175,
+ beta1=0.00,sigma=Inf,gamma=0.077),col="blue",add=T,lwd=2)
>
```



2) Comparing SIR model by using the direct and adaptivetau methods:

```
> plot(seir(duration=10*365,type="stoch",method="direct",nbVilles=1,
+ S=S,E=E,I=I,R=R,beta0=1.175, beta1=0.0,sigma=Inf,gamma=0.077),
+ col="red",ylab="number of infectives", xlab="time (day)")
> plot(seir(duration=10*365,type="stoch",method="adaptivetau",nbVilles=1,
+ S=S,E=E,I=I,R=R,beta0=1.175,beta1=0.0,sigma=Inf,gamma=0.077),add=T,col="blue",lwd=2)
>
```



Conclusion

Through the above simple examples, the dizzys package maintains quick runtimes and exact results on the SIR/SEIR models with the different number of subpopulations and the different simulation time. The package successfully implement both an exact solution and an approximate solution. Moreover, this hybrid R/C++ implementation appears to be faster than any pure R implementation as in the GillespieSSA package.

Acknowledgment

This package is based upon work supported by the Institute for Research and Development at Paris in France and by Professor Jean-Daniel Zucker and Marc Choisy.

References

[1] Andrew James Black. *THE STOCHASTIC DYNAMICS OF EPIDEMIC MODELS*. School of Physics and Astronomy, 2010.

[2] Daniel.T.Gillespie. Exact stochastic simulation of coupled chemical reactions. 1977.

[3] Sonia Altizer et al. Seasonality and the dynamics of infectious diseases. *Ecology Letters*, 9:467–484, 2006.

[4] Keeling and Rohani. *Modeling infectious diseases in Humans ans Animals.* Princeton University Press, 2008.

[5] Yang Cao Daniel T.Gillespie and Linda R.Petzold. The adaptive explicit-implicit tau-leaping method with automatic tau selection. 2007.