

## General Simulation Code for SI and SIR Model

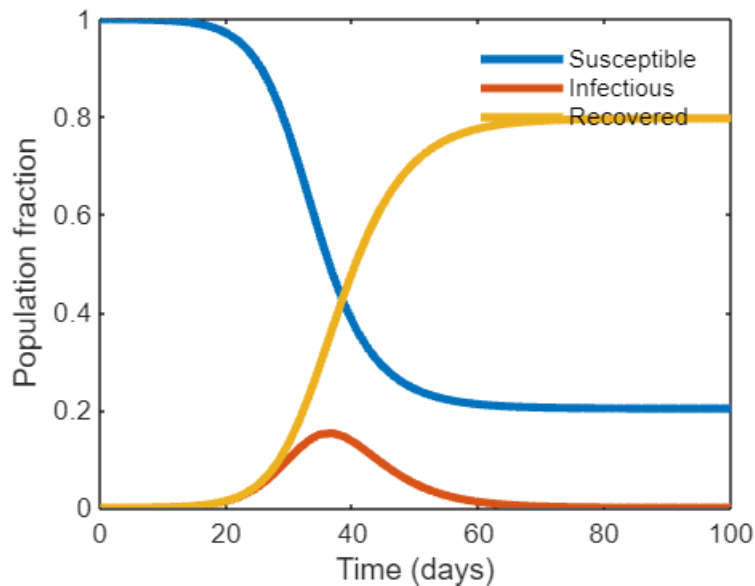
```
function dydt = si_model(t,y,parms)
% function dydt = si_model(t,y,parms)
% SI Model
S=y(1);
I=y(2);
% The model
dSdt = -parms.beta*S*I+parms.gamma*I;
dIdt = parms.beta*S*I-parms.gamma*I;
dydt = [dSdt; dIdt];
end
```

```
function dydt = sir_model(t,y,parms)
% function dydt = sir_model(t,y,parms)
% SIR Model
S=y(1);
I=y(2);
% The model
dSdt = -parms.beta*S*I;
dIdt = parms.beta*S*I-parms.gamma*I;
dRdt = parms.gamma*I;
dydt = [dSdt; dIdt; dRdt];
end
```

```
parms.c = 20; % Contacts per unit time (e.g., days)
parms.p = 0.025; % Probability of infectious contact
parms.beta = 0.5; % Transmission rate | beta = cp
parms.gamma = 1/4; % Recovery rate (days^-1)
parms.R0 = 2; % R0 = beta/gamma
parms.N = 10000;
parms.I0= 1;
parms.S0= parms.N-parms.I0;

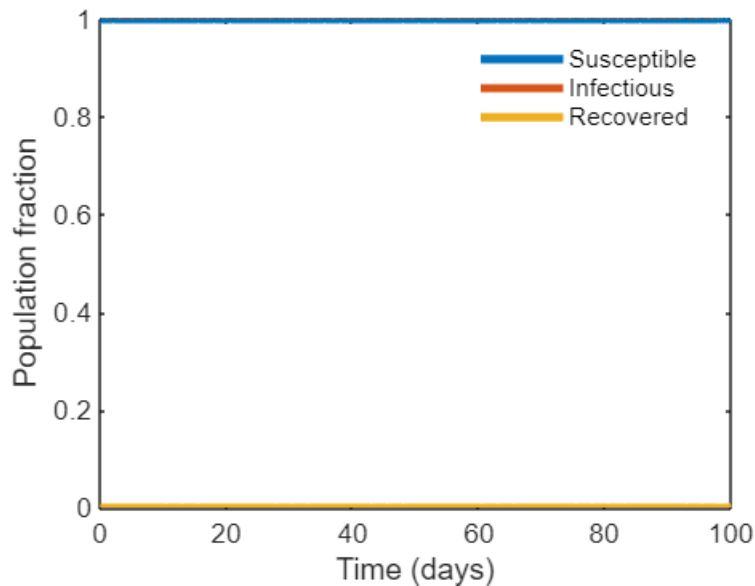
% Runs the model
[t,y]=ode45(@(t,y) sir_model(t,y,parms),[0 100],[parms.S0 parms.I0 0]/parms.N);

% Plots the results
tmph=plot(t,y);
set(tmph,'linewidth',3);
xlabel('Time (days)');
ylabel('Population fraction');
tmplh = legend('Susceptible','Infectious','Recovered');
legend('boxoff');
```



From the visualized model, I believe that the disease will not spread. We can see that at around 60 days, the values for each plateau towards an equilibrium. After 60 days, there are virtually no infectious with approximately 0.8 being recovered and 0.2 being susceptible.

```
% main data goes here
pars.c = 20; % Contacts per unit time (e.g., days)
pars.p = 0.01; % Probability of infectious contact
pars.beta = 0.2; % Transmission rate | beta = cp
pars.gamma = 1/4; % Recovery rate (days^-1)
pars.R0 = 0.8; % R0 = beta/gamma
pars.N = 10000;
pars.I0 = 1;
pars.S0 = pars.N - pars.I0;
% Run the model
[t,y] = ode45(@(t,y) sir_model(t,y,pars), [0 100], [pars.S0 pars.I0 0]/pars.N);
% Plot the results
tmph = plot(t,y);
set(tmph, 'linewidth', 3);
xlabel('Time (days)');
ylabel('Population fraction');
tmplh = legend('Susceptible', 'Infectious', 'Recovered');
legend('boxoff');
```



Here, I do not expect the disease to spread. As seen in the plot, there will be no infection with these parameters. It looks like 100 percent of the population is susceptible throughout the simulation, meaning no one got infected besides the initial one.

## Strength, Speed, and Size Evaluation

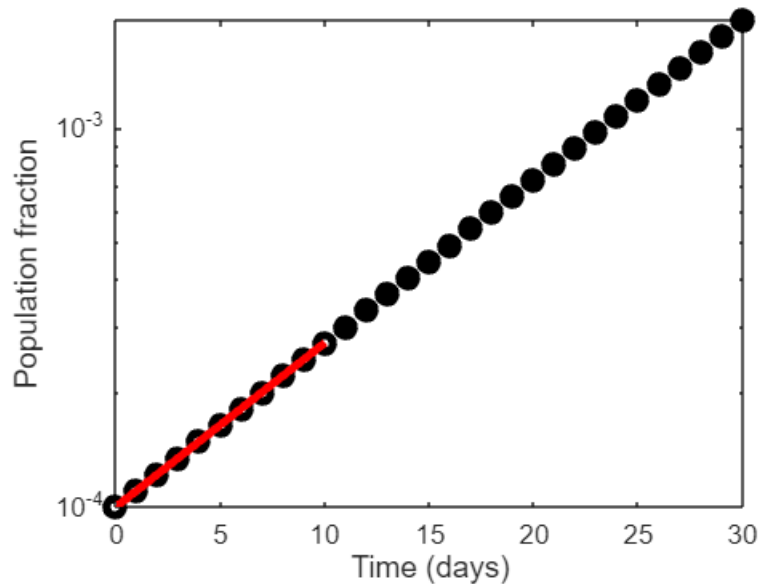
Beta = 0.5, Gamma = 0.4

```
pars.beta = 0.5; % Transmission rate | beta = cp
pars.gamma = 0.4; % Recovery rate (days^-1)
pars.R0 = pars.beta/pars.gamma; % R0 = beta/gamma
pars.N = 10000;
pars.I0= 1;
pars.S0= pars.N-pars.I0;
[t,y]=ode45(@(t,y) sir_model(t,y,pars),[0:1:10],[pars.S0 pars.I0 0]/pars.N);
[p,s]=polyfit(t,log(y(:,2)),1);
% Plot the data and overlay the best-fit exponential
tmph=semilogy(t,y(:,2),'ko');
set(tmph,'linewidth',3);
hold on
tmph=semilogy(t,exp(p(1)*t+p(2)),'r-');
set(tmph,'linewidth',3);
% Use solid points for the future k
[t,y]=ode45(@(t,y) sir_model(t,y,pars),[0:1:30],[pars.S0 pars.I0 0]/pars.N);
tmpi=find(t>10);
tmph=semilogy(t(tmpi),y(tmpi,2),'ko');
```

```

set(tmph,'linewidth',3,'markerfacecolor','k');
xlabel('Time (days)');
ylabel('Population fraction');
hold off

```



```

strength = pars.R0

```

```

strength = 1.2500

```

```

expected_speed = pars.beta - pars.gamma

```

```

expected_speed = 0.1000

```

```

estimated_speed = p(1)

```

```

estimated_speed = 0.0998

```

Beta = 1, Gamma = 0.5

```

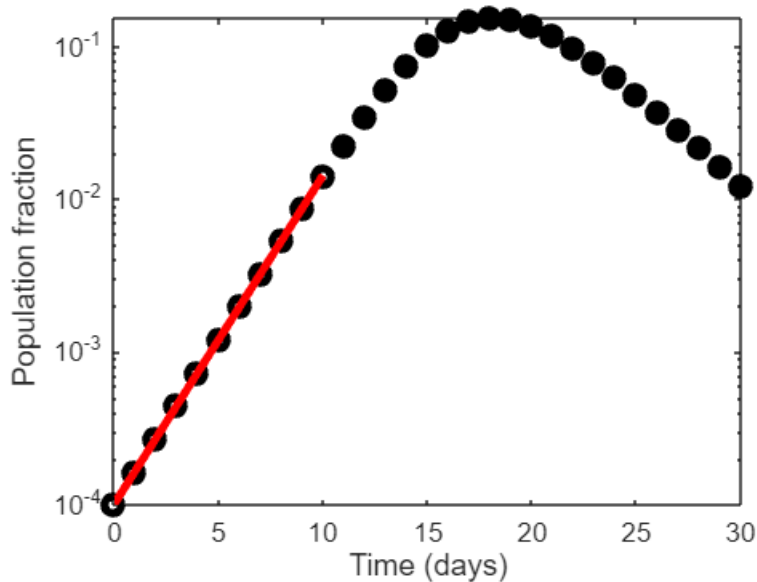
pars.beta = 1; % Transmission rate | beta = cp
pars.gamma = 0.5; % Recovery rate (days^-1)
pars.R0 = pars.beta/pars.gamma; % R0 = beta/gamma
pars.N = 10000;
pars.I0= 1;
pars.S0= pars.N-pars.I0;
[t,y]=ode45(@(t,y) sir_model(t,y,pars),[0:1:10],[pars.S0 pars.I0 0]/pars.N);
[p,s]=polyfit(t,log(y(:,2)),1);
% Plot the data and overlay the best-fit exponential
tmph=semilogy(t,y(:,2),'ko');
set(tmph,'linewidth',3);
hold on

```

```

tmp_h=semilogy(t,exp(p(1)*t+p(2)),'r-');
set(tmp_h,'linewidth',3);
% Use solid points for the future k
[t,y]=ode45(@(t,y) sir_model(t,y,pars),[0:1:30],[pars.S0 pars.I0 0]/pars.N);
tmp_i=find(t>10);
tmp_h=semilogy(t(tmp_i),y(tmp_i,2),'ko');
set(tmp_h,'linewidth',3,'markerfacecolor','k');
xlabel('Time (days)');
ylabel('Population fraction');
hold off

```



```
strength = pars.R0
```

```
strength = 2
```

```
expected_speed = pars.beta - pars.gamma
```

```
expected_speed = 0.5000
```

```
estimated_speed = p(1)
```

```
estimated_speed = 0.4954
```

Beta = 0.25, Gamma = 0.5

```

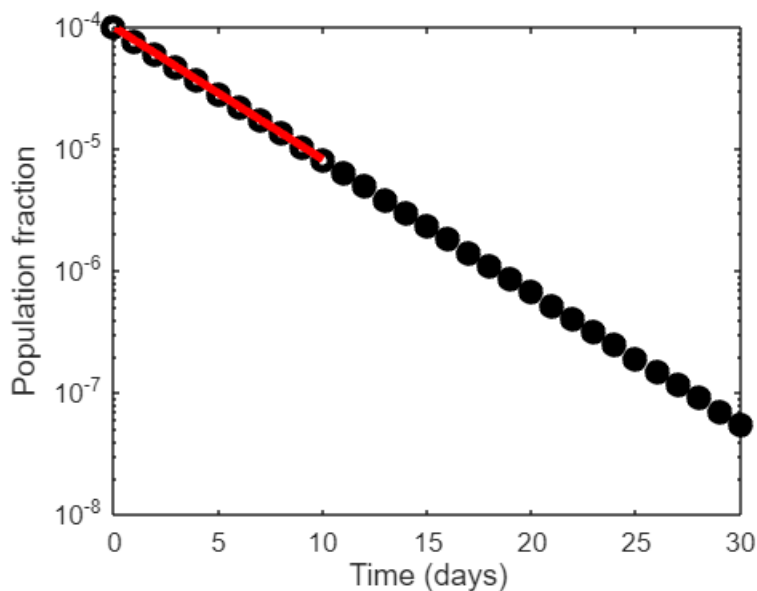
pars.beta = 0.25; % Transmission rate | beta = cp
pars.gamma = 0.5; % Recovery rate (days^-1)
pars.R0 = pars.beta/pars.gamma; % R0 = beta/gamma
pars.N = 10000;
pars.I0= 1;
pars.S0= pars.N-pars.I0;
[t,y]=ode45(@(t,y) sir_model(t,y,pars),[0:1:10],[pars.S0 pars.I0 0]/pars.N);

```

```

[p,s]=polyfit(t,log(y(:,2)),1);
% Plot the data and overlay the best-fit exponential
tmph=semilogy(t,y(:,2),'ko');
set(tmph,'linewidth',3);
hold on
tmph=semilogy(t,exp(p(1)*t+p(2)),'r-');
set(tmph,'linewidth',3);
% Use solid points for the future k
[t,y]=ode45(@(t,y) sir_model(t,y,pars),[0:1:30],[pars.S0 pars.I0 0]/pars.N);
tmpi=find(t>10);
tmph=semilogy(t(tmpi),y(tmpi,2),'ko');
set(tmph,'linewidth',3,'markerfacecolor','k');
xlabel('Time (days)');
ylabel('Population fraction');
hold off

```



```
strength = pars.R0
```

```
strength = 0.5000
```

```
expected_speed = pars.beta - pars.gamma
```

```
expected_speed = -0.2500
```

```
estimated_speed = p(1)
```

```
estimated_speed = -0.2500
```

Beta = 0.75, Gamma = 0.25

```

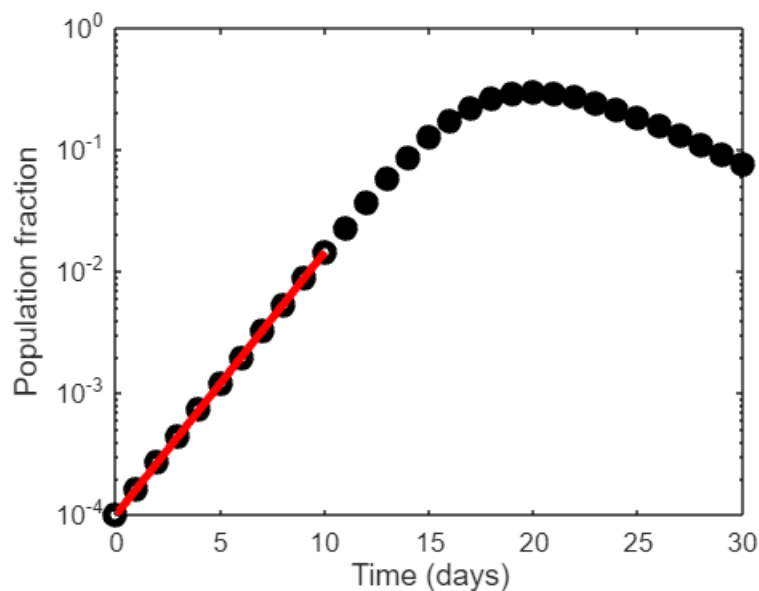
pars.beta = 0.75; % Transmission rate | beta = cp
pars.gamma = 0.25; % Recovery rate (days^-1)

```

```

pars.R0 = pars.beta/pars.gamma; % R0 = beta/gamma
pars.N = 10000;
pars.I0= 1;
pars.S0= pars.N-pars.I0;
[t,y]=ode45(@(t,y) sir_model(t,y,pars),[0:1:10],[pars.S0 pars.I0 0]/pars.N);
[p,s]=polyfit(t,log(y(:,2)),1);
% Plot the data and overlay the best-fit exponential
tmph=semilogy(t,y(:,2),'ko');
set(tmph,'linewidth',3);
hold on
tmph=semilogy(t,exp(p(1)*t+p(2)),'r-');
set(tmph,'linewidth',3);
% Use solid points for the future k
[t,y]=ode45(@(t,y) sir_model(t,y,pars),[0:1:30],[pars.S0 pars.I0 0]/pars.N);
tmpi=find(t>10);
tmph=semilogy(t(tmpi),y(tmpi,2),'ko');
set(tmph,'linewidth',3,'markerfacecolor','k');
xlabel('Time (days)');
ylabel('Population fraction');
hold off

```



```
strength = pars.R0
```

```
strength = 3
```

```
expected_speed = pars.beta - pars.gamma
```

```
expected_speed = 0.5000
```

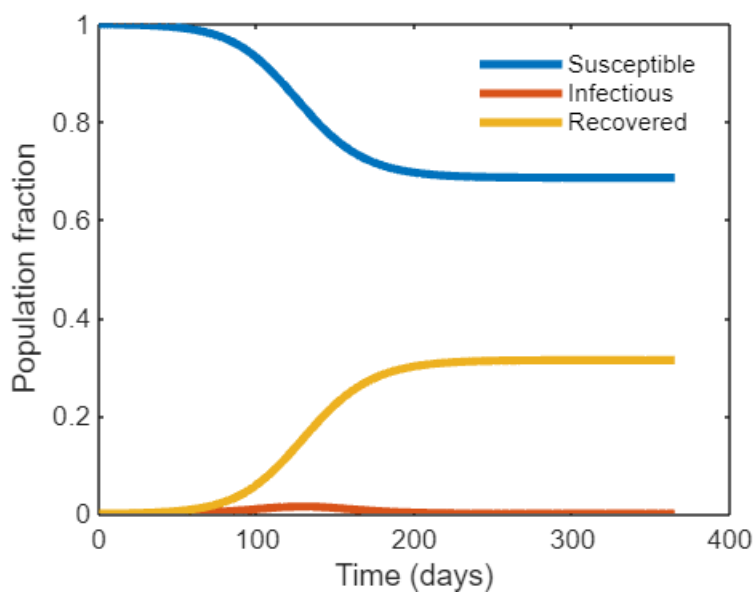
```
estimated_speed = p(1)
```

```
estimated_speed = 0.4974
```

%Transmission	Recovery	Strength	Speed
% $\beta$	$\gamma$	$R_0$	$r$
% 0.5	0.4	1.25	0.1
% 1	0.5	2	0.5
% 0.25	0.5	0.5	-0.25
% 0.75	0.25	3	0.5

## Comparing the SIR Model to the SI Model

```
%sir
pars.c = 20;
pars.p = 0.25;
pars.beta = 0.3; % Transmission rate | beta = cp
pars.gamma = 0.25; % Recovery rate (days^-1)
pars.R0 = 1.2; % R0 = beta/gamma
pars.N = 10000;
pars.I0 = 1;
pars.S0 = pars.N-pars.I0;
[t,y]=ode45(@(t,y) sir_model(t,y,pars),[0 365],[pars.S0 pars.I0 0]/pars.N);
figure
% Plot the results
tmph=plot(t,y);
set(tmph,'linewidth',3);
xlabel('Time (days)');
ylabel('Population fraction');
tmplh = legend('Susceptible','Infectious','Recovered');
legend('boxoff');
```

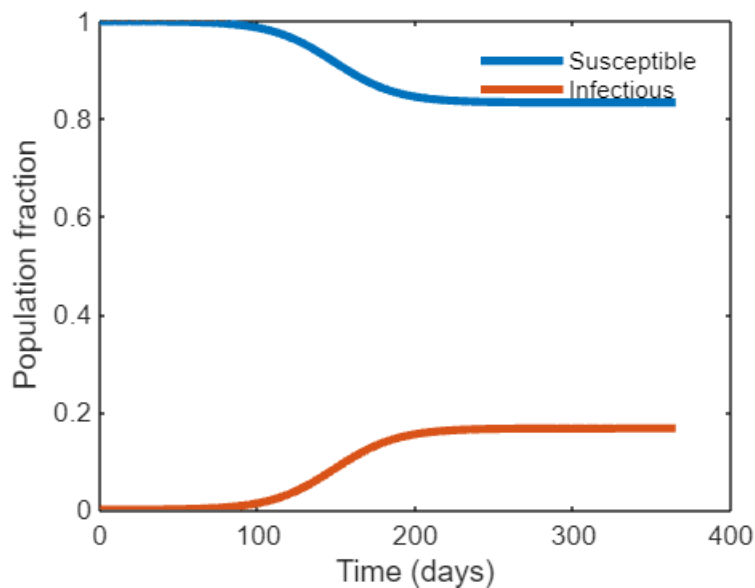




```

%si
pars.c = 20;
pars.p = 0.25;
pars.beta = 0.3; % Transmission rate | beta = cp
pars.gamma = 0.25; % Recovery rate (days^-1)
pars.R0 = 1.2; % R0 = beta/gamma
pars.N = 10000;
pars.I0 = 1;
pars.S0 = pars.N-pars.I0;
[t,y]=ode45(@(t,y) si_model(t,y,pars),[0 365],[pars.S0 pars.I0]/pars.N);
figure
% Plot the results
tmph=plot(t,y);
set(tmph,'linewidth',3);
xlabel('Time (days)');
ylabel('Population fraction');
tmplh = legend('Susceptible','Infectious');
legend('boxoff');

```



From my observation of the two plots, we see that there are some major differences. For example, in the SI model, since there are only S and I, I grows logistically where it plateaus around 0.2. However, for the SIR model, since infectious turns into recovered, there are less susceptible, more recovered, and way less infectious. This is because infectious are turning into recovered.

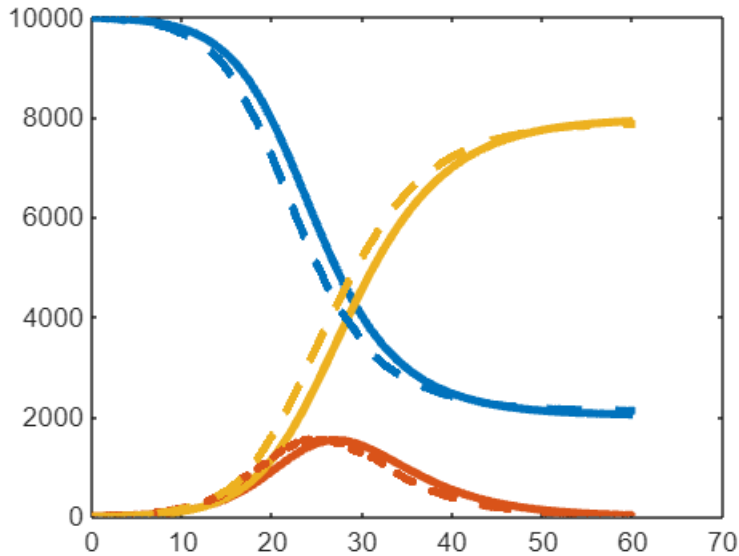
## Stochastic SIR Models

```
function [t,y] = stochsim_SIR(trange,y0,pars)
% function [t,y] = stochsim_SIR(trange,y0,pars)
%
% Simulates an SIR model via the Gillespie algorithm
% from t0 to tf in trange given initial
% conditions in y0 = [S0 I0 R0] and parameters
% in pars. Returns time and values
% Conditions
t0=trange(1);
tf=trange(2);
t(1)=t0;
y(1,:)=y0;
tcur=t0;
ycur=y0;
ind=1;
% Model
while (tcur<tf)
% Check to see if there is an infection
if (ycur(2)==0)
ind=ind+1;
t(ind)=tf;
y(ind,:)=ycur;
break;
end
% Rates
infrate = pars.beta*ycur(1)*ycur(2)/pars.N;
recreate = pars.gamma*ycur(2);
totrate = infrate + recreate;
dt = -1/totrate*log(rand);
tcur=tcur+dt;
% Event type
if (rand<(infrate/totrate)) % infection
ycur(2) = ycur(2)+1;
ycur(1) = ycur(1)-1;
else % recovery
ycur(3)=ycur(3)+1;
ycur(2)=ycur(2)-1;
end
ind=ind+1;
t(ind)=tcur;
y(ind,:)=ycur;
end
end
```

```

pars.c = 20;
pars.p = 0.025;
pars.beta = pars.c*pars.p; % Transmission rate | beta = cp
pars.gamma = 0.25; % Recovery rate (days^-1)
pars.tf = 60;
pars.R0 = pars.beta/pars.gamma; % R0 = beta/gamma
pars.N = 10000;
pars.I0 = 10;
pars.S0 = pars.N-pars.I0;
figure
[t,y]=ode45(@(t,y) sir_model(t,y,pars),[0 pars.tf],[pars.S0 pars.I0 0]/
pars.N);
% Plot the results
tmph=plot(t,pars.N*y);
set(tmph,'linewidth',3);
hold on
[tsim,ysim]=stochsim_SIR([0 pars.tf],[pars.S0 pars.I0 0],pars);
tmphsim=plot(tsim,ysim,'--');
set(tmphsim,'linewidth',3);
for i=1:3,
set(tmphsim(i),'color',get(tmph(i),'color'));
end

```



Here, we see that the stochastic and deterministic approaches lead to similar results in shape. Although there are small differences with exact points, the overall shape and trajectory between the two are still the same.

However, in the real world, these differences must be accounted for in order to infer and predict disease trajectories in real systems.