Soluzioni del I homework.

Modello di stato:

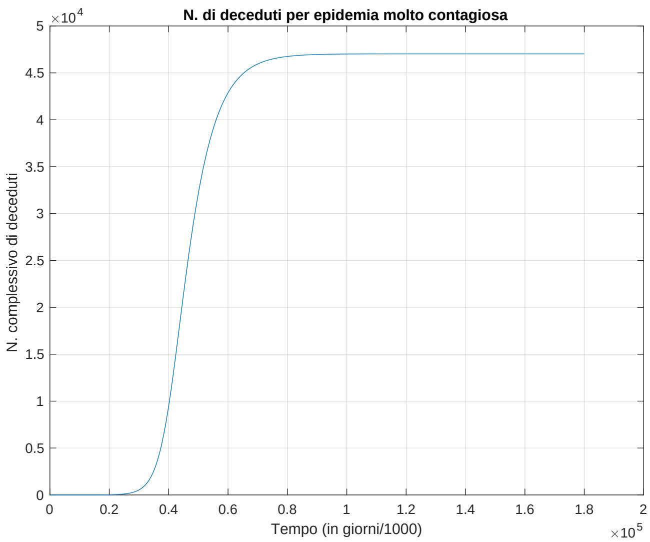
$$\begin{cases} \dot{S}(t) = -\beta S(t)I(t) \\ \dot{I}(t) = \beta S(t)I(t) - \alpha I(t) \\ \dot{R}(t) = \alpha I(t) \\ y(t) = 0.05R(t) \end{cases}$$

Caso (i) (malattia molto contagiosa):

Durata: 142 giorni

N. complessivo di deceduti: 47025 persone

N. di individui che non hanno contratto la malattia: 59510 persone

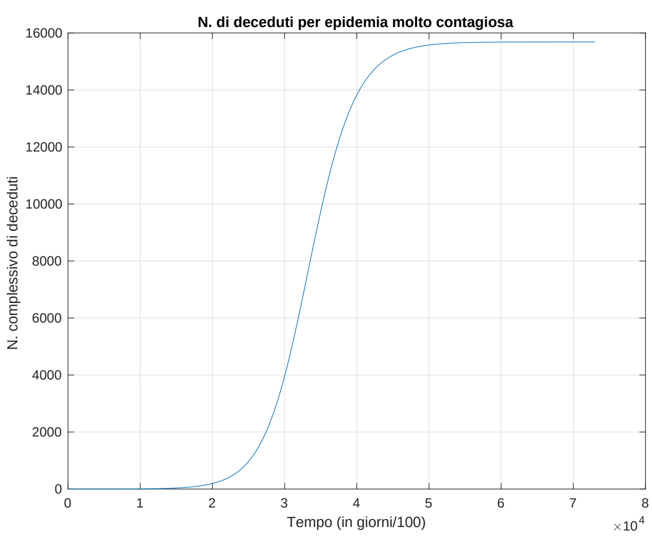


Caso (ii) (malattia poco contagiosa):

Durata: 703 giorni

N. complessivo di deceduti: 15686 persone

N. di individui che non hanno contratto la malattia: 686290 persone



Ragionamenti e i passaggi per ottenere la soluzione

Per prima cosa scegliamo le variabili di stato: a questo scopo dobbiamo chiederci che variabili dobbiamo conoscere a un certo istante per poter calcolare l'evoluzione futura dell'epidemia. Una scelta molto intuitiva è quella di selezionare le tre variabili
S: numero di individui suscettibili
I: numero di individui infetti
R: numero di individui rimossi
Calcoliamo le derivate delle tre variabili scelte per vedere se si tratta effettivamente di variabili di stato.

Sappiamo che il numero di suscettibili che, nell'unità di tempo, diventano infettivi (per effetto del contagio) è proporzionale, secondo il coefficiente β, al prodotto del numero dei suscettibili per il numero degli infettivi. Possiamo scrivere questo in termini matematici come segue:

$$\frac{dS}{dt} = \dot{S}(t) = -\beta S(t)I(t)$$

L'incremento degli infettivi è dato dalla differenza fra coloro che si infettano e coloro che passano alla classe dei rimossi, pertanto si ha:

$$\frac{dI}{dt} = \dot{I}(t) = \beta S(t)I(t) - \alpha I(t)$$

e

$$\frac{dR}{dt} = \dot{R}(t) = \alpha I(t).$$

In conclusione, il modello di stato è:

$$\begin{cases} \dot{S}(t) = -\beta S(t)I(t) \\ \dot{I}(t) = \beta S(t)I(t) - \alpha I(t) \\ \dot{R}(t) = \alpha I(t) \\ \\ y(t) = 0.05R(t) \end{cases}$$

Simulazione di epidemia relativa a malattia molto contagiosa inclusi grafico dei deceduti, e calcolo di durata, numero complessivo di deceduti e numero di persone suscettibili alla fine dell'epidemia

```
%%%% CASO DI EPIDEMIA MOLTO CONTAGIOSA

clear all
close all

b=1/(2*10^6); %si fissa beta (denotato da b)
a=1/6;  %si fissa alfa (denotato da a)
N=1000;
delt=1/N;%si fissa l'ampiezza degli intervallini a 1/N di giorno
         %(per determinare un'ampiezza adeguata ...
         % si procede per tentativi raddoppiando N (ossia dimezzando l'ampiezza ...
         % degli intervallini) fino a quando con un ulteriore ...
         % raddoppio di N la soluzione rimane essenzialmente uguale)
x0=[1000000;1;0]; %si fissa lo stato iniziale
S=x0(1); %si denota con S la prima componente dello stato corrente
I=x0(2); %si denota con I la seconda componente dello stato corrente
R=x0(2); %si denota con R la terza componente dello stato corrente
x=x0 %inizialmente la traiettoria di stato contiene solo lo stato iniziale
Tf=180; % si fissa a 6 mesi (180 giorni) la durata della simulazione ...
        %(per determinare una durata adeguata si procede per tentativi ...
        % aumentando la durata  fino a quando si constata che la ...
        % simulazione supera la fine dell'epidemia).


for t=0:delt:Tf  %in ciascun istante ...

Sp=-b*S*I; %si calcola la derivata della prima componente dello stato ...
Ip=b*S*I-a*I;  %la derivata della seconda componente dello stato ...
Rp=a*I;  %e la derivata della terza componente dello stato


S=S+Sp*delt; %si calcola la prima componente dello stato all'istante successivo
I=I+Ip*delt; %la seconda componente dello stato all'istante successivo
R=R+Rp*delt; %e la terza componente dello stato all'istante successivo
xs=[S;I;R]; %si sovrappongono le tre componenti ottenendo lo stato all'istante successivo
x=[x xs]; %si allunga la traiettoria di stato giustapponendo alla traiettoria
          %gia` calcolata fino a questo momento lo stato all'istante successivo
end
```

```
%Alla fine x e‘ una matrice con tre righe (il numero di componenti dello stato) e
%un numero di colonne pari al numero degli istanti t_i considerati +1.

y=0.05*x(3,:); %si calcola l’uscita (il 5% della terza riga di x)
plot(y) %si traccia il grafico di y
title(’N. di deceduti per epidemia molto contagiosa’) %si aggiunge il titolo
xlabel([’Tempo (in giorni/’,num2str(N),’)’]) % si aggiunge l’unita‘ in ascissa ...
ylabel(’N. complessivo di deceduti’) % e in ordinata
grid


%CALCOLO QUANDO AVVIENE LA FINE DELL’EPIDEMIA
for t=0:N:Tf*N  %avanzo di un intero giorno alla volta
    k=t+1; %k e‘ l’indice della colonna di x che contiente i dati al tempo t
           %(devo usare k=t+1 perche’ il primo indice e‘ 1 e il prmo tempo e‘ zero)
    if x(2,k)<1
        fine_ep=t; % fine_ep e‘ il primo istante misurato in giorni/N in cui ...
                   % il giorno inizia con un numero di infetti e‘ minore di 1.
        Durata=fine_ep/N %si calcola la durata come numero di giorni in cui
                         %il numero di infetti e‘ stato non minore di 1
                         %almeno in una parte della giornata

        break
    end
end

Sfine=x(1,fine_ep) %Numero di suscettibili ancora presenti alla fine dell’epidemia
Ifine=x(2,fine_ep) %Numero di infetti presenti alla fine dell’epidemia ...
                   %(deve essere <1)
Rfine=x(3,fine_ep) %Numero di rimossi alla fine dell’epidemia
Morti_totali=0.05*x(3,fine_ep) %Numero totale di morti alla fine dell’epidemia
```

Simulazione di epidemia relativa a malattia poco contagiosa inclusi grafico dei deceduti, e calcolo di durata, numero complessivo di deceduti e numero di persone suscettibili alla fine dell'epidemia

```
%%%% CASO DI EPIDEMIA POCO CONTAGIOSA

clear all
close all

b=1/(5*10^6); %si fissa beta (denotato da b)
a=1/6;  %si fissa alfa (denotato da a)

N=100;
delt=1/N; %si fissa l'ampiezza degli intervallini a 1/N di giorno
         %(per determinare un'ampiezza adeguata ...
         % si procede per tentativi raddoppiando N (ossia dimezzando l'ampiezza ...
         % degli intervallini) fino a quando con un ulteriore ...
         % raddoppio di N la soluzione rimane essenzialmente uguale)
x0=[1000000;1;0]; %si fissa lo stato iniziale
S=x0(1); %si denota con S la prima componente dello stato corrente
I=x0(2); %si denota con I la seconda componente dello stato corrente
R=x0(2); %si denota con R la terza componente dello stato corrente
x=x0 %inizialmente la traiettoria di stato contiene solo lo stato iniziale
Tf=730; % si fissa a 2 anni (730 giorni) la durata della simulazione ...
        %(per determinare una durata adeguata si procede per tentativi ...
        % aumentando la durata  fino a quando si constata che la ...
        % simulazione supera la fine dell'epidemia).


for t=0:delt:Tf  %in ciascun istante ...

Sp=-b*S*I; %si calcola la derivata della prima componente dello stato ...
Ip=b*S*I-a*I;  %la derivata della seconda componente dello stato ...
Rp=a*I;  %e la derivata della terza componente dello stato

S=S+Sp*delt; %si calcola la prima componente dello stato all'istante successivo
I=I+Ip*delt; %la seconda componente dello stato all'istante successivo
R=R+Rp*delt; %e la terza componente dello stato all'istante successivo
xs=[S;I;R]; %si sovrappongono le tre componenti ottenendo lo stato all'istante successivo
x=[x xs]; %si allunga la traiettoria di stato giustapponendo alla traiettoria
          %gia` calcolata fino a questo momento lo stato all'istante successivo
end
```

```
%Alla fine x e‘ una matrice con tre righe (il numero di componenti dello stato) e
%un numero di colonne pari al numero degli istanti t_i considerati +1.

y=0.05*x(3,:); %si calcola l’uscita (il 5% della terza riga di x)
plot(y) %si traccia il grafico di y
title(’N. di deceduti per epidemia molto contagiosa’) %si aggiunge il titolo
xlabel([’Tempo (in giorni/’,num2str(N),’)’]) % si aggiunge l’unita‘ in ascissa ...
ylabel(’N. complessivo di deceduti’) % e in ordinata
grid



%CALCOLO QUANDO AVVIENE LA FINE DELL’EPIDEMIA
for t=0:N:Tf*N  %avanzo di un intero giorno alla volta
    k=t+1; %k e‘ l’indice della colonna di x che contiente i dati al tempo t
           %(devo usare k=t+1 perche’ il primo indice e‘ 1 e il prmo tempo e‘ zero)
    if x(2,k)<1
        fine_ep=t; % fine_ep e‘ il primo istante misurato in giorni/N in cui ...
                   % il nuovo giorno inizia con un numero di infetti e‘ minore di 1.
        Durata=fine_ep/N %si calcola la durata come numero di giorni in cui
                         %il numero di infetti e‘ stato non minore di 1
                         %almeno per una parte della giornata

        break
    end
end

Sfine=x(1,fine_ep) %Numero di suscettibili ancora presenti alla fine dell’epidemia
Ifine=x(2,fine_ep) %Numero di infetti presenti alla fine dell’epidemia ...
                   %(deve essere <1)
Rfine=x(3,fine_ep) %Numero di rimossi alla fine dell’epidemia
Morti_totali=0.05*x(3,fine_ep) %Numero totale di morti alla fine dell’epidemia
```

Calcolo di durata, n. di deceduti e n. di individui che rimangono suscettibili

Tutte queste informazioni sono contenute nelle tre righe della matrice x ottenuta con il programma precedente.

Esaminando la seconda riga di x si nota che tutti i relativi valori sono minori di 1 a partire da un certo campione (chiamiamolo $\bar{k}$) che determina la durata della malattia.

Il valore del campione $\bar{k}$ della prima riga ci dà il numero di individui che non hanno contratto la malattia dopo che l'epidemia è terminata.

Il numero numero complessivo di deceduti dopo che l'epidemia è terminata è pari al 5% del campione $\bar{k}$ della terza riga della matrice x.

I relativi valori si possono ottenere esaminando "a mano" la matrice x ottenuta o con una semplice routine come quella riportata nel codice.

Osservazioni Si noti che nel caso di malattia poco contagiosa:
- la durata è molto pù lunga (quasi due anni contro meno di 5 mesi);
- il numero numero complessivo di deceduti è circa un terzo del caso di malattia molto contagiosa;
- il numero di individui che non hanno contratto la malattia dopo che l'epidemia è terminata è pari a quasi il 70% della popolazione contro meno del 6% del caso di malattia poco contagiosa

Soluzione del problema facoltativo

Il modo più semplice di affrontare il problema posto è considerare tre diversi modelli di stato: uno per ciascuna delle tre fasi considerate nella formulazione del problema.

FASE 1 (isolamento sociale)

In questa fase il modello è quello ricavato nella prima parte dell'esercizio con l'unica differenza che al posto del parametro β dobbiamo considerare $\beta_{rid} := \frac{\beta}{1+u(t)}$. Il modello quindi è In conclusione, il modello di stato è:

$$\begin{cases} \dot{S}(t) = -\frac{\beta}{1+u(t)}S(t)I(t) \\ \dot{I}(t) = \frac{\beta}{1+u(t)}S(t)I(t) - \alpha I(t) \\ \dot{R}(t) = \alpha I(t) \\ \\ y(t) = 0.05R(t) \end{cases}$$

dove $u(t)$ è costante: $u(t) = u \in [0, 10]$.

Tale modello vale per $0 \leq t \leq 60$ giorni.

FASE 2 (vaccinazione)

In questa fase aggiungiamo alle variabili di stato considerate fino a questo momento l'ulteriore variabile

V: numero di individui vaccinati

la cui derivata è

$$\dot{V}(t) = g := k(10 - u(t)), \qquad \text{con } k = 1200$$

dove $u(t)$ è costante: $u(t) = u \in [0, 10]$.

La dinamica delle variabili I e R è quella del modello senza ingressi (ricavata nella soluzione dell'esercizio precedente) mentre la dinamica della variabile S risente del termine $g := k(10 - u)$ che sottrae da S i vaccinati. Pertanto

$$\dot{S}(t) = -\beta S(t)I(t) - k(10 - u(t)), \qquad \text{con } k = 1200$$

dove $u(t)$ è costante: $u(t) = u \in [0, 10]$.

Attenzione: la presenza del termine negativo $-k(10 - u)$ potrebbe portare a valori negativi di $S(t)$. Tali valori sarebbero privi di senso. Pertanto la dinamica precedente è valida solo fino a quando ci sono persone che possono essere vaccinate ossia fino a quando $S(t) > 0$. Se $S(t)$ diventa nullo nell'istante t_0, da t_0 in poi, il termine $g = k(10 - u(t))$ sparisce sia dall'equazione per $\dot{S}(t)(t)$ sia da quella per

$\dot{V}(t) = 0$. Possiamo tenere conto di ciò considerando le equazioni

$$\dot{V}(t) = 1200(10 - u)\text{sign}[S(t)]$$

e

$$\dot{S}(t) = -\beta S(t)I(t) - 1200(10 - u)\text{sign}[S(t)]$$

dove la funzione sign[·] è definita da $\text{sign}[x] := \begin{cases} 1 & \text{se } x > 0 \\ 0 & \text{se } x = 0 \\ -1 & \text{se } x < 0 \end{cases}$

In conclusione, il modello di stato (del 4° ordine) è:

$$\begin{cases} \dot{S}(t) = -\beta S(t)I(t) - 1200(10 - u)\text{sign}[S(t)] \\ \dot{I}(t) = \beta S(t)I(t) - \alpha I(t) \\ \dot{R}(t) = \alpha I(t) \\ \dot{V}(t) = 1200(10 - u)\text{sign}[S(t)] \\ \\ y(t) = 0.05R(t) \end{cases}$$

Tale modello vale per $60 \leq t \leq 180$ (giorni) e va inizializzato ponendo le variabili S, I e R al tempo $t = 60$ al valore che avevano raggiunto alla fine della Fase I (corrispondente appunto a $t = 60$). Invece la variabile $V(60)$ va inizializzata a 0 perché all'inizio della seconda fase nessun individuo è ancora stato vaccinato.

Si noti che volendo si potrebbe anche considerare un modello del terzo ordine facendo a meno di introdurre la variabile V che non ha alcuna influenza nell'uscita. L'abbiamo introdotta solo perché può essere comodo tenere traccia anche del numero dei vaccinati.

FASE 3 (modello senza ingressi)

In questa fase il modello è quello ricavato nella prima parte dell'esercizio:

$$\begin{cases} \dot{S}(t) = -\beta S(t)I(t) \\ \dot{I}(t) = \beta S(t)I(t) - \alpha I(t) \\ \dot{R}(t) = \alpha I(t) \\ \\ y(t) = 0.05R(t) \end{cases}$$

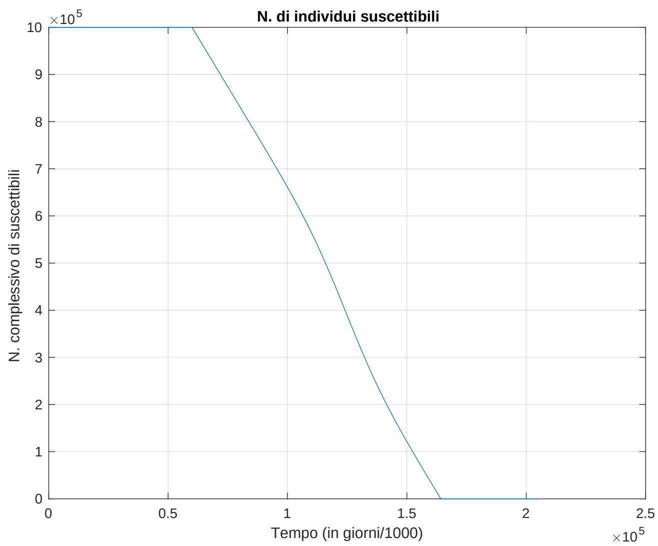
Tale modello vale per $t \geq 180$ (giorni) e fino alla fine dell'epidemia; va inizializzato ponendo le variabili S, I e R al tempo $t = 180$ al valore che avevano raggiunto alla fine della Fase II (corrispondente appunto a $t = 180$). Da quel momento in poi il numero $V(t)$ di vaccinati rimane costante e non serve nemmeno considerarlo.

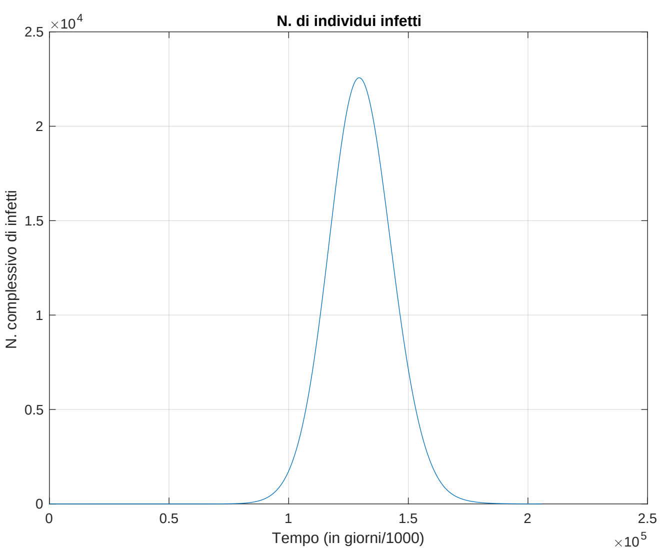
Si tratta ora di simulare le tre fasi dell'andamento dell'epidemia per ciascun valore dell'ingresso $u \in [0, 10]$ con una granularità di 0.1 e di scegliere il valore ottimo di u che minimizza il numero di deceduti alla fine dell'epidemia.

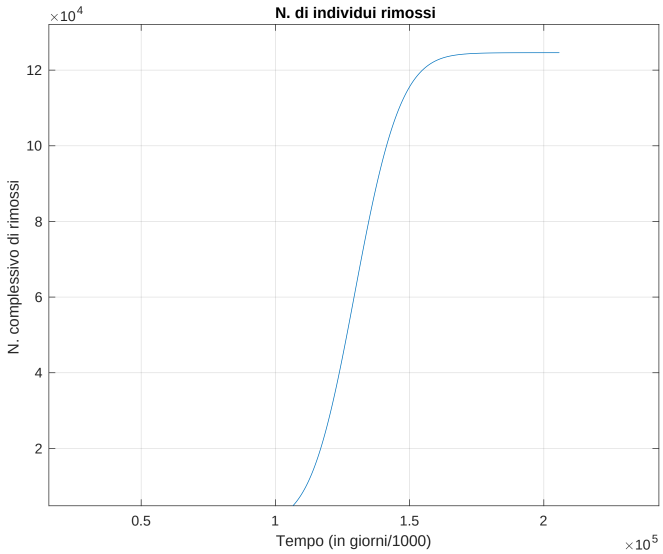
La simulazione mostra che il valore ottimo di u è

$$u^{\mathrm{ott}} = 3$$

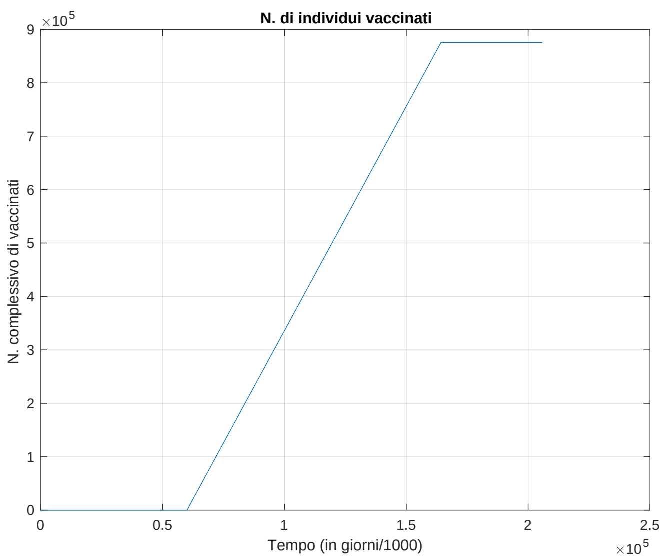
e, in corrispondenza a tale valore, alla fine dell'epidemia ci sono stati 6230 individui deceduti (da paragonare con gli oltre 47000 deceduti in assenza di politiche di contenimento dell'epidemia). È anche interessante notare che concentrare tutte le risorse nell'isolamento sociale (ossia scegliere $u = 10$) ritarda il decorso dell'epidemia ma non cambia il numero totale di deceduti che rimane esattamente a quello che si ha nel caso in cui non si attuano politiche di contenimento dell'epidemia. Nel caso di $u = u^{\mathrm{ott}} = 3$, l'epidemia si esaurisce dopo 206 giorni e i grafici che mostrano come evolvono le variabili $S(t)$, $I(t)$, $R(t)$ e $V(t)$ e il numero di deceduti nel tempo sono riportati di seguito.



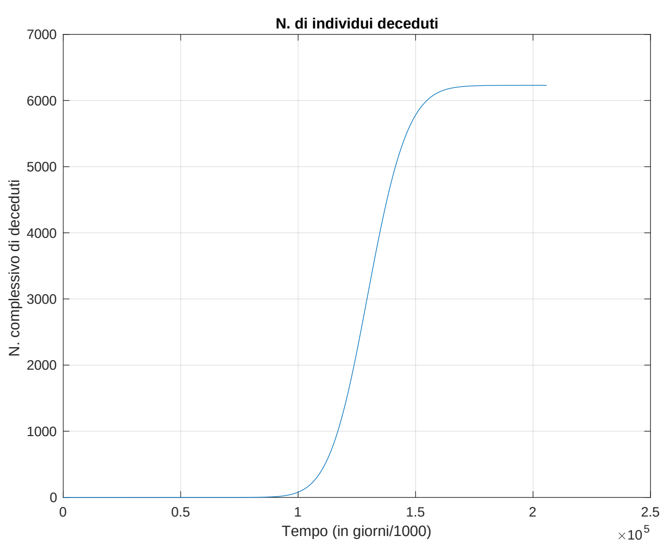




N. di individui rimossi
$\times 10^4$
12
10
8
6
4
2
N. complessivo di rimossi
0.5
1
1.5
2
Tempo (in giorni/1000)
$\times 10^5$



N. di individui vaccinati
$\times 10^5$
9
8
7
6
5
4
3
2
1
0
N. complessivo di vaccinati
0
0.5
1
1.5
2
2.5
Tempo (in giorni/1000)
$\times 10^5$



N. di individui deceduti
7000
6000
5000
4000
3000
2000
1000
0
N. complessivo di deceduti
0
0.5
1
1.5
2
2.5
Tempo (in giorni/1000)
$\times 10^5$

Di seguito si riporta il codice utilizzato.

```
%%%% APPROFONDIMENTI

clear all
close all

b=1/(2*10^6);%si fissa beta (denotato da b)
ubar=10;%si fissa la quantita totale di risore di controllo disponibili;
a=1/6;  %si fissa alfa (denotato da a)
N=1000;
delt=1/N;%si fissa l'ampiezza degli intervallini a 1/N di giorno
         %(per determinare un'ampiezza adeguata ...
         % si procede per tentativi raddoppiando N (ossia dimezzando l'ampiezza ...
         % degli intervallini) fino a quando con un ulteriore ...
         % raddoppio di N la soluzione rimane essenzialmente uguale)



for u=0:0.1:10

x0=[1000000;1;0]; %si fissa lo stato iniziale
S=x0(1); %si denota con S la prima componente dello stato corrente
I=x0(2); %si denota con I la seconda componente dello stato corrente
R=x0(2); %si denota con R la terza componente dello stato corrente
x=x0; %inizialmente la traiettoria di stato contiene solo lo stato iniziale
Tf=60;  %si fissa a 2 mesi (60 giorni) la durata della prima parte della epidemia ...
        %ossia la parte in cui il vaccino non e` ancora disponibile

b_rid=b/(1+u); %fisso il tasso di contagio ridotto grazie all'isolamento

for t=0:delt:Tf  %in ciascun istante ...

Sp=-b_rid*S*I; %si calcola la derivata della prima componente dello stato ...
Ip=b_rid*S*I-a*I;  %la derivata della seconda componente dello stato ...
Rp=a*I;  %e la derivata della terza componente dello stato


S=S+Sp*delt; %si calcola la prima componente dello stato all'istante successivo
I=I+Ip*delt; %la seconda componente dello stato all'istante successivo
R=R+Rp*delt; %e la terza componente dello stato all'istante successivo
xs=[S;I;R]; %si sovrappongono le tre componenti ottenendo lo stato all'istante successivo
```

```
x=[x xs]; %si allunga la traiettoria di stato giustapponendo alla traiettoria
          %gia` calocolata fino a questo momento lo stato all'istante successivo
end
%Alla fine x e` una matrice con due righe (il numero di componenti dello stato) e
%tante colonne quanti sono gli istanti t_i considerati. piu` precisamnete
%c'e` una colonna in piu` perche' anche all'istante Tf viene aggiunto il
%campione successivo. Pertanto, il valore dello stato all'istante Tf e`
%dato dalla penultima colonna del vettore x (e non dall'ultima).

l=size(x,2); %calcolo il n. di colonne della matrice x
x_di_Tf=x(:,l-1); %calcolo lo stato al tempo Tf
z=x(:,1:l-1); %salvo la traiettoria di stato nella nuova matrice z

Deceduti_2mesi=0.05*x_di_Tf(3,1); %calcolo il numero di deceduti nei primi due mesi


%INIZIO SECONDA PARTE DI SIMULAZIONE

g=1200*(ubar-u);
x0=x_di_Tf; %fisso lo stato iniziale (il nuovo istante iniziale e` il vecchio tf)
S=x0(1); %si denota con S la prima componente dello stato corrente
I=x0(2); %si denota con I la seconda componente dello stato corrente
R=x0(3); %si denota con R la terza componente dello stato corrente
V=0; %aggiungo la variabile di stato costituita dal numero (inizialmente nullo) dei vaccinati
x=[x0;0]; %inizialmente la traiettoria di stato contiene solo lo stato iniziale
Tf=120;  %si fissa a 4 mesi (120 giorni) la durata della prima parte della ...
        %ossia la parte in cui il vaccino non e` ancora disponibile

for t=0:delt:Tf  %in ciascun istante ...

Sp=-b*S*I-g; %si calcola la derivata della prima componente dello stato ...
Ip=b*S*I-a*I;  %la derivata della seconda componente dello stato ...
Rp=a*I;  %e la derivata della terza componente dello stato
Vp=g;


S=S+Sp*delt; %si calcola la prima componente dello stato all'istante successivo
if S<0
    S=0;
    g=0;
end
I=I+Ip*delt; %la seconda componente dello stato all'istante successivo
```

```
R=R+Rp*delt; %la terza componente dello stato all'istante successivo
V=V+Vp*delt; %e la quarta componente dello stato all'istante successivo
xs=[S;I;R;V]; %si sovrappongono le 4 componenti ottenendo lo stato all'istante successivo
x=[x xs]; %si allunga la traiettoria di stato giustapponendo alla traiettoria
          %gia` calocolata fino a questo momento lo stato all'istante successivo
end

l=size(x,2); %calcolo il n. di colonne della matrice x
x_di_Tf=x(:,l-1); %calcolo lo stato al tempo Tf
Deceduti_6mesi=0.05*x_di_Tf(3,1); %calcolo i morti dopo 6 mesi
Deceduti_II=Deceduti_6mesi-Deceduti_2mesi; %calcolo i morti nella seconda fase
z1=x(:,1:l-1); %salvo la traiettoria di stato nella nuova matrice z1




%INIZIO TERZA PARTE DI SIMULAZIONE


x0=x_di_Tf; %fisso lo stato iniziale (il nuovo istante iniziale e` il vecchio Tf)
S=x0(1); %si denota con S la prima componente dello stato corrente
I=x0(2); %si denota con I la seconda componente dello stato corrente
R=x0(3); %si denota con R la terza componente dello stato corrente
x=[x0(1); x0(2); x0(3)]; %inizialmente la traiettoria di stato contiene solo lo stato inizial


t=0;
while I>=1 %La terza parte continua fino a quando gli infetti non sono scesi al di sotto di 1
t=t+delt; %Misuro il tempo trascorso nella terza parte di simulazione
Sp=-b*S*I; %si calcola la derivata della prima componente dello stato ...
Ip=b*S*I-a*I;  %la derivata della seconda componente dello stato ...
Rp=a*I;  %e la derivata della terza componente dello stato

S=S+Sp*delt; %si calcola la prima componente dello stato all'istante successivo
I=I+Ip*delt; %la seconda componente dello stato all'istante successivo
R=R+Rp*delt; %la terza componente dello stato all'istante successivo
xs=[S;I;R]; %si sovrappongono le tre componenti ottenendo lo stato all'istante successivo
x=[x xs]; %si allunga la traiettoria di stato giustapponendo alla traiettoria
          %gia` calocolata fino a questo momento lo stato all'istante successivo
end
```

```
l=size(x,2); %calcolo il n. di colonne della matrice x
Tf=t; %calcolo il tempo finale della terza simulazione
x_di_Tf=x(:,l); %calcolo lo stato al tempo Tf
Deceduti_TOT=0.05*x_di_Tf(3,1); %calcolo i morti alla fine dell'epidemia
Deceduti_III=Deceduti_TOT-Deceduti_6mesi; %calcolo i morti nella terza fase

u
Deceduti=Deceduti_TOT
Durata_TOT=180+Tf

end



%%rifaccio tutta la simulazione con il valore ottimo di u

u=3 %fisso il valore ottimo di u

x0=[1000000;1;0]; %si fissa lo stato iniziale
S=x0(1); %si denota con S la prima componente dello stato corrente
I=x0(2); %si denota con I la seconda componente dello stato corrente
R=x0(2); %si denota con R la terza componente dello stato corrente
x=x0; %inizialmente la traiettoria di stato contiene solo lo stato iniziale
Tf=60;  %si fissa a 2 mesi (60 giorni) la durata della prima parte della epidemia ...
        %ossia la parte in cui il vaccino non e` ancora disponibile

b_rid=b/(1+u); %fisso il tasso di contagio ridotto grazie all'isolamento

for t=0:delt:Tf  %in ciascun istante ...

Sp=-b_rid*S*I; %si calcola la derivata della prima componente dello stato ...
Ip=b_rid*S*I-a*I;  %la derivata della seconda componente dello stato ...
Rp=a*I;  %e la derivata della terza componente dello stato



S=S+Sp*delt; %si calcola la prima componente dello stato all'istante successivo
I=I+Ip*delt; %la seconda componente dello stato all'istante successivo
R=R+Rp*delt; %e la terza componente dello stato all'istante successivo
xs=[S;I;R]; %si sovrappongono le tre componenti ottenendo lo stato all'istante successivo
x=[x xs]; %si allunga la traiettoria di stato giustapponendo alla traiettoria
          %gia` calocolata fino a questo momento lo stato all'istante successivo
end
%Alla fine x e` una matrice con due righe (il numero di componenti dello stato) e
```

```
%tante colonne quanti sono gli istanti t_i considerati. piu‘ precisamnete
%c’e‘ una colonna in piu‘ perche’ anche all’istante Tf viene aggiunto il
%campione successivo. Pertanto, il valore dello stato all’istante Tf e‘
%dato dalla penultima colonna del vettore x (e non dall’ultima).

l=size(x,2); %calcolo il n. di colonne della matrice x
x_di_Tf=x(:,l-1); %calcolo lo stato al tempo Tf
z=x(:,1:l-1); %salvo la traiettoria di stato nella nuova matrice z

Deceduti_2mesi=0.05*x_di_Tf(3,1); %calcolo il numero di deceduti nei primi due mesi


%INIZIO SECONDA PARTE DI SIMULAZIONE

g=1200*(ubar-u);
x0=x_di_Tf; %fisso lo stato iniziale (il nuovo istante iniziale e‘ il vecchio tf)
S=x0(1); %si denota con S la prima componente dello stato corrente
I=x0(2); %si denota con I la seconda componente dello stato corrente
R=x0(3); %si denota con R la terza componente dello stato corrente
V=0; %aggiungo la variabile di stato costituita dal numero (inizialmente nullo) dei vaccinat
x=[x0;0]; %inizialmente la traiettoria di stato contiene solo lo stato iniziale
Tf=120;  %si fissa a 4 mesi (120 giorni) la durata della prima parte della ...
        %ossia la parte in cui il vaccino non e‘ ancora disponibile

for t=0:delt:Tf  %in ciascun istante ...

Sp=-b*S*I-g; %si calcola la derivata della prima componente dello stato ...
Ip=b*S*I-a*I;  %la derivata della seconda componente dello stato ...
Rp=a*I;  %e la derivata della terza componente dello stato
Vp=g;


S=S+Sp*delt; %si calcola la prima componente dello stato all’istante successivo
if S<0
    S=0;
    g=0;
end
I=I+Ip*delt; %la seconda componente dello stato all’istante successivo
R=R+Rp*delt; %la terza componente dello stato all’istante successivo
V=V+Vp*delt; %e la quarta componente dello stato all’istante successivo
xs=[S;I;R;V]; %si sovrappongono le 4 componenti ottenendo lo stato all’istante successivo
x=[x xs]; %si allunga la traiettoria di stato giustapponendo alla traiettoria
```

```
         %gia‘ calocolata fino a questo momento lo stato all’istante successivo
end

l=size(x,2); %calcolo il n. di colonne della matrice x
x_di_Tf=x(:,l-1); %calcolo lo stato al tempo Tf
Deceduti_6mesi=0.05*x_di_Tf(3,1); %calcolo i morti dopo 6 mesi
Deceduti_II=Deceduti_6mesi-Deceduti_2mesi; %calcolo i morti nella seconda fase
z1=x(:,1:l-1); %salvo la traiettoria di stato nella nuova matrice z1




%INIZIO TERZA PARTE DI SIMULAZIONE


x0=x_di_Tf; %fisso lo stato iniziale (il nuovo istante iniziale e‘ il vecchio Tf)
S=x0(1); %si denota con S la prima componente dello stato corrente
I=x0(2); %si denota con I la seconda componente dello stato corrente
R=x0(3); %si denota con R la terza componente dello stato corrente
x=[x0(1); x0(2); x0(3)]; %inizialmente la traiettoria di stato contiene solo lo stato inizial


t=0;
while I>=1 %La terza parte continua fino a quando gli infetti non sono scesi al di sotto di 1
t=t+delt; %Misuro il tempo trascorso nella terza parte di simulazione
Sp=-b*S*I; %si calcola la derivata della prima componente dello stato ...
Ip=b*S*I-a*I;  %la derivata della seconda componente dello stato ...
Rp=a*I;  %e la derivata della terza componente dello stato

S=S+Sp*delt; %si calcola la prima componente dello stato all’istante successivo
I=I+Ip*delt; %la seconda componente dello stato all’istante successivo
R=R+Rp*delt; %la terza componente dello stato all’istante successivo
xs=[S;I;R]; %si sovrappongono le tre componenti ottenendo lo stato all’istante successivo
x=[x xs]; %si allunga la traiettoria di stato giustapponendo alla traiettoria
         %gia‘ calocolata fino a questo momento lo stato all’istante successivo
end


l=size(x,2); %calcolo il n. di colonne della matrice x
Tf=t; %calcolo il tempo finale della terza simulazione
x_di_Tf=x(:,l); %calcolo lo stato al tempo Tf
Deceduti_TOT=0.05*x_di_Tf(3,1); %calcolo i morti alla fine dell’epidemia
```

```
Deceduti_III=Deceduti_TOT-Deceduti_6mesi; %calcolo i morti nella terza fase

Deceduti=Deceduti_TOT
Durata_TOT=180+Tf



z=[z;zeros(1,size(z,2))]; %aggiungo una riga di zeri a z (zero vaccinati in tutta la Fase 1)
z=z(:,1:size(z,2)-1); %elimino l'ultima colonna di z che e` uguale alla prima di z1
z12=[z z1]; %giustappongo z e z1 per avere i dati delle prime 2 fasi in z12
V_TOT=z12(4, size(z12,2)); %calcolo il numero di vaccinati totali alla fine di Fase 2
x4=V_TOT*ones(1,size(x,2)); %produco un riga di tutti elementi uguali a V_TOT
x=[x;x4]; %aggiungo la riga x4 a x (i vaccinati sono V_TOT in tutta la Fase 3)
z12=z12(:,1:size(z12,2)-1); %elimino l'ultima colonna di z12 che e` uguale alla prima di x
z123=[z12 x]; %giustappongo z12 e x per avere tutti i dati in z123

plot(z123(1,:)) %si traccia il grafico di S
title('N. di individui suscettibili') %si aggiunge il titolo
xlabel(['Tempo (in giorni/',num2str(N),')']) % si aggiunge l'unita` in ascissa ...
ylabel('N. complessivo di suscettibili') % e in ordinata
grid

figure
plot(z123(2,:)) %si traccia il grafico di I
title('N. di individui infetti') %si aggiunge il titolo
xlabel(['Tempo (in giorni/',num2str(N),')']) % si aggiunge l'unita` in ascissa ...
ylabel('N. complessivo di infetti') % e in ordinata
grid

figure
plot(z123(3,:)) %si traccia il grafico di R
title('N. di individui rimossi') %si aggiunge il titolo
xlabel(['Tempo (in giorni/',num2str(N),')']) % si aggiunge l'unita` in ascissa ...
ylabel('N. complessivo di rimossi') % e in ordinata
grid

figure
plot(z123(4,:)) %si traccia il grafico di V
title('N. di individui vaccinati') %si aggiunge il titolo
xlabel(['Tempo (in giorni/',num2str(N),')']) % si aggiunge l'unita` in ascissa ...
ylabel('N. complessivo di vaccinati') % e in ordinata
grid
```

```
figure
plot(0.05*z123(3,:)) %si traccia il grafico dei deceduti
title('N. di individui deceduti') %si aggiunge il titolo
xlabel(['Tempo (in giorni/',num2str(N),')']) % si aggiunge l'unita` in ascissa ...
ylabel('N. complessivo di deceduti') % e in ordinata
grid
```