

Modelli Matematici per la Biologia – Esercitazione 5

a.a. 2006-2007

Dott. Simone Zuccher

25 Maggio 2007

Nota. Queste pagine potrebbero contenere degli errori: chi li trova è pregato di segnalarli all'autore (zuccher@sci.univr.it).

1 Complessità dei sistemi competitivi (cicli limite, ampie oscillazioni, attrattori strani)

1.1 Esercizio

Studiare, al variare dei parametri a, b, c (positivi), delle condizioni iniziali e del tempo finale, il comportamento del sistema

$$\begin{cases} x' = x(1-x) - \frac{axy}{x+c} \\ y' = by(1-y/x). \end{cases}$$

1.1.1 Risoluzione

Il sistema ammette più punti di equilibrio, ma essendo le popolazioni x e y solo positive o nulle noi ci concentriamo esclusivamente in questo range ottenendo come unico punto di equilibrio $(\frac{(1-a-c)+\sqrt{(1-a-c)^2+4c}}{2}, \frac{(1-a-c)+\sqrt{(1-a-c)^2+4c}}{2})$, che indicheremo con $(x_{\text{eq}}, y_{\text{eq}})$. Analizzando il sistema linearizzato attorno a $(x_{\text{eq}}, y_{\text{eq}})$, si scopre che la parte reale degli autovalori λ è negativa solo se

$$b > \frac{(a - \sqrt{(1-a-c)^2+4c})(1+a+c - \sqrt{(1-a-c)^2+4c})}{2a}.$$

Pertanto, assumendo $a, c > 0$ e plottando la superficie $b = \frac{(a - \sqrt{(1-a-c)^2+4c})(1+a+c - \sqrt{(1-a-c)^2+4c})}{2a}$ si ottiene visivamente la regione dello spazio (a, c, b) stabile e instabile (vedi figura 1). Come si può notare dalla figura 1, per $a < 1/2$ e tutti i valori di $b > 0, c > 0$, la stabilità del punto di equilibrio $(x_{\text{eq}}, y_{\text{eq}})$ è assicurata. Al contrario quando $a > 1/2$, affinché ci sia stabilità il generico punto (a, c, b) deve trovarsi al di sopra della superficie riportata

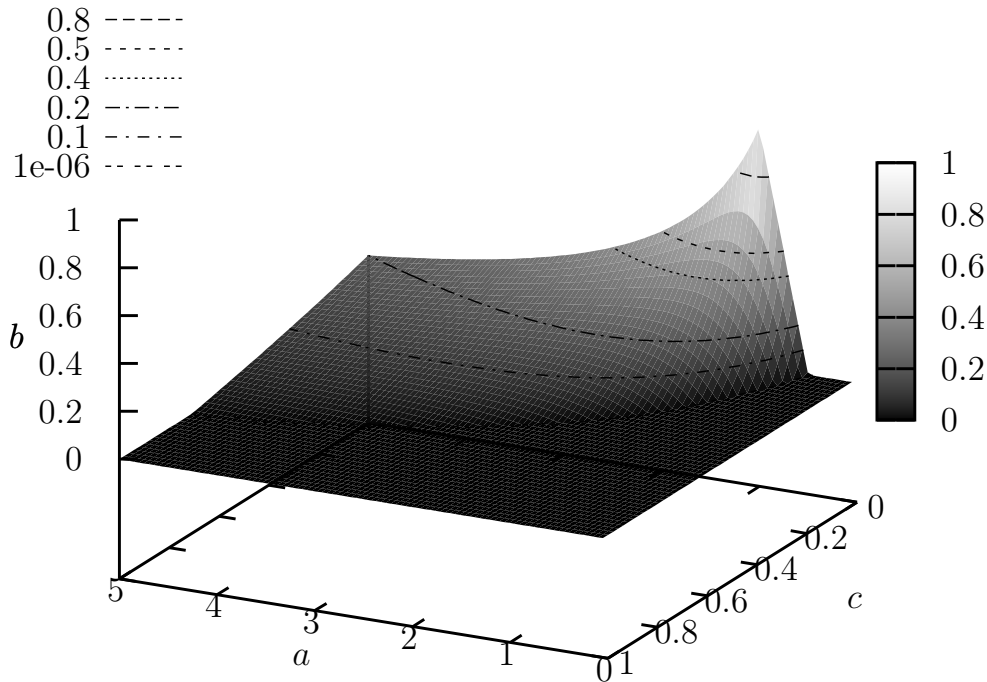


Figura 1: Il punto di equilibrio risulta stabile per le terne (a, c, b) che si trovano al di sopra della superficie riportata.

in figura 1. In presenza di autovalori reali negativi (x_{eq}, y_{eq}) è un nodo stabile, mentre diventa un fuoco stabile se gli autovalori sono complessi a parte reale negativa. Nel caso instabile si passa da nodo instabile (entrambi gli autovalori reali positivi) a fuoco instabile (autovalori complessi a parte reale positiva).

In figura 2 sono riportati i risultati ottenuti con $a = 0.1, b = 0.1, c = 0.1$, partendo dalla condizione iniziale $(x_0, y_0) = (0.1, 0.1)$. Come si può notare, il punto di equilibrio è stabile. Il caso $a = 0.1, b = 5, c = 1, (x_0, y_0) = (0.1, 0.1)$ (figura 3), ottenuto per valori dei parametri che assicurano la stabilità, mostra un chiaro fuoco stabile. Mantenendo fissa la condizione iniziale cambiando i parametri in modo da avvicinarsi al limite di stabilità, si osserva che la traiettoria compie sempre più giri attorno al punto di equilibrio stabile, indice del fatto che la parte reale degli autovalori, pur essendo negativa, è piccola rispetto alla parte immaginaria (si veda la figura 4, ottenuta con $a = 1, b = 0.5, c = 0.2$).

Nel caso in cui la condizione di stabilità non sia rispettata, il punto di equilibrio diventa instabile per cui, anche partendo da condizioni iniziali molto vicine ad esso, la traiettoria se ne allontana. Per l'analisi di questa situazione è possibile applicare il teorema di Poincaré-Bendixson che assicura l'esistenza di un ciclo limite cui tende la traiettoria quando si parta da condizioni iniziali interne ad esso. Quindi, partendo all'interno di questo ciclo limite la traiettoria si avvicinerà a tale ciclo e partendo all'esterno

altrettanto.

In figura 5 è riportato il caso $a = 1, b = 0.1, c = 0.1$, che assicura l'instabilità del punto $(x_{eq}, y_{eq}) \approx (0.27016, 0.27016)$. Partendo da una condizione iniziale $(x_0, y_0) = (0.27, 0.27)$ molto vicina a (x_{eq}, y_{eq}) , si può notare un'iniziale allontanamento dalla condizione iniziale e il susseguente carattere periodico della traiettoria quando raggiunge il ciclo limite stabile. La figura 6 differisce dalla 5 solo per le condizioni iniziali $(x_0, y_0) = (0.2, 0.01)$, che sono ora esterne al ciclo limite che si ottiene per $a = 1, b = 0.1, c = 0.1$. Come si può notare, la traiettoria viene attratta dal ciclo limite che risulta, pertanto, stabile.

Si noti una differenza sostanziale tra il Lotka-Volterra classico visto la volta scorsa e il modello più realistico. Ora il ciclo limite è stabile, nel senso che attrae le traiettorie. Al contrario, il ciclo limite del sistema classico è tale per cui una leggera variazione delle condizioni iniziali, ovvero una perturbazione nell'evoluzione ciclica, può portare a traiettorie che per alcuni tempi possono essere molto lontane (nel piano delle fasi). Questo si verifica, in particolare, quando la perturbazione avviene per valori delle variabili tali per cui il prodotto xy è piuttosto piccolo.

La figura 1 è stata ottenuta utilizzando lo script per GNU Gnuplot qui di seguito riportato.

```
# Name:      surface
# Author:    Simone Zuccher
# Created:   24 May 2007
# Purpose:   plot the stability bound for the realistic Lotka-Volterra system
#           u' = u(1-u) - auv/(u+d)
#           v' = bv(1 - v/u)
# Modified:
#
reset
set xlabel '$a$'
set ylabel '$c$'
set zlabel '$b$' -7,-7
set view 64, 207
set pm3d
set palette gray
set isosamples 50, 50
unset surface
set contour s
set cntrparam levels discrete 1e-6, .1, .2, .4, .5, .8
set ztics 0,.2
set cbtics 0,.2
set key 11, -1.3
surf(x,y) = ((x-sqrt((1-x-y)**2+4*y))*(1+x+y-sqrt((1-x-y)**2+4*y))/2./x >=0) \
? (x-sqrt((1-x-y)**2+4*y))*(1+x+y-sqrt((1-x-y)**2+4*y))/2./x : 0
splot [0:5] [0:1] surf(x,y) t 'Stability surface' w l
```

Utilizzando il file `realLV.m` per GNU Octave di seguito riportato si possono fare diverse prove al variare dei parametri a, b, d , della condizione iniziale e del tempo finale di integrazione. **Si ricordi di cambiare gli estremi della finestra di visualizzazione dipendentemente dalla condizione iniziale.**

```

% Name:      realLV.m
% Author:    Simone Zuccher
% Created:   24 May 2007
% Purpose:   solve the realistic Lotka-Volterra system
%           u' = u(1-u) - auv/(u+d)
%           v' = bv(1 - v/u)
%           given u0 and v0
% Input:     see file
% Output:    1. plot of u(t) versus v(t) together with the vector field
%           2. plot time histories of u(t), v(t)
% Modified:
%
% The non-negative equilibrium points are the following
%
%           2           2
%   SQRT(d  + (2 a + 2) d + a  - 2 a + 1) - d - a + 1
% [ u = -----,
%           2
%
%           2           2
%   SQRT(d  + (2 a + 2) d + a  - 2 a + 1) - d - a + 1
% v = ----- ]
%           2
%
% Clear all variables
clear all;

% Window ranges
xmin=0;
xmax=1;
ymin=0;
ymax=.5;

% Model constant
global aa;

% Give instructions
disp('');
disp('~~~~~');
disp('This script solves a real version of the Lotka-Volterra system:');
disp(' u' = u(1-u) - auv/(u+d)');
disp(' v' = bv(1 - v/u)');
disp('given a, b, d ');

% Set initial conditions
aa=input('Insert constants [a b d]: ');

```

```

% Compute b_stab (b<b_stab ensures stability)
bstab=(aa(1)-sqrt((1-aa(1)-aa(3))^2+4*aa(3)))*\
      (1+aa(1)+aa(3)-sqrt((1-aa(1)-aa(3))^2+4*aa(3)))/(2*aa(1));

if (aa(2)<bstab)
    disp('WARNING: a, b, d out of the stability region');
endif

% Equilibrium points
eq = [ (sqrt(aa(3)^2+(2*aa(1)+2)*aa(3)+aa(1)^2-2*aa(1)+1)-aa(3)-aa(1)+1)/2\
      (sqrt(aa(3)^2+(2*aa(1)+2)*aa(3)+aa(1)^2-2*aa(1)+1)-aa(3)-aa(1)+1)/2];
%eq = [eq; aa(3)/aa(4) aa(1)/aa(2)];
disp('Equilibrium points:');
disp(eq);

% Set initial conditions
x0=input('Insert initial conditions [x0 y0]: ');

% Set final time for integration
tmax=input('Insert final time: ');

disp('Initial condition:');
disp(x0);

% Time parameters
tmin=0;
dt=.01;
% Create time
t = tmin:dt:tmax;

% dx and dy used only for vectors
dx=abs(xmax-xmin)/30;
dy=abs(ymax-ymin)/30;
% rescales vector size
scale=0.027*max(abs(xmax-xmin),abs(ymax-ymin));

% Definition of the dynamical system
function xdot=dsys(x, t)
    global aa;
    u = x(1);
    v = x(2);
    xdot(1) = u*(1-u) - aa(1)*u*v/(u+aa(3));
    xdot(2) = aa(2)*v*(1 - v/u);
endfunction

__gnuplot_set__ nokey

```

```

setax=[xmin xmax ymin ymax];
axis(setax)

[X, Y] = meshgrid(xmin:dx:xmax, ymin:dy:ymax);
DX = X.*(1-X) - aa(1).*X.*Y./(X+aa(3));
DY = aa(2).*Y.*(1 - Y./X);
L = sqrt(DX.^2 + DY.^2);
mytitle=["Phase portrait. Initial conditions: x0=" num2str(x0(1)) \
        ", y0=" num2str(x0(2)) \
        "; a=" num2str(aa(1)) ", b=" num2str(aa(2))\
        ", c=" num2str(aa(3))];
__gnuplot_set__ nokey
__gnuplot_set__ xlabel 'x(t)'
__gnuplot_set__ ylabel 'y(t)'
title(mytitle)

% Plot vector field
quiver(X, Y, scale*DX./L, scale*DY./L)
hold on;

% Plot all equilibrium points
plot( eq(:,1), eq(:,2), '*k')

x = lsode("dsys", x0, t)';
plot( x(1,1), x(2,1), '*k', x(1,:), x(2,:), '-r')
hold off;

% Wait for keypressed
disp('Please press a key to continue...');
pause();
mytitle=["Time histories. Initial conditions: x0=" num2str(x0(1)) \
        ", y0=" num2str(x0(2)) \
        "; a=" num2str(aa(1)) ", b=" num2str(aa(2))\
        ", c=" num2str(aa(3))];
__gnuplot_set__ auto
__gnuplot_set__ xlabel 't'
__gnuplot_set__ ylabel 'x(t), y(t)'
title(mytitle)
__gnuplot_set__ key

% Plot time histories
plot( t, x(1,:), '-r;x(t);', t, x(2,:), '-g;y(t);')

```

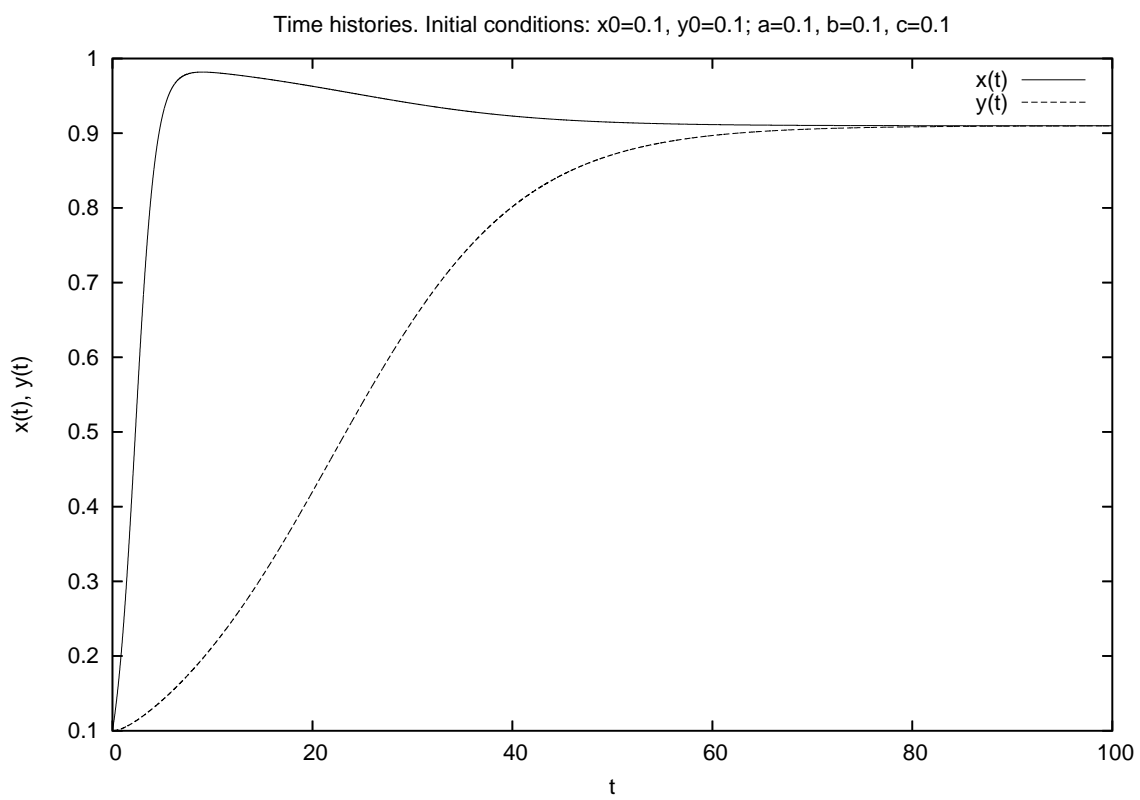
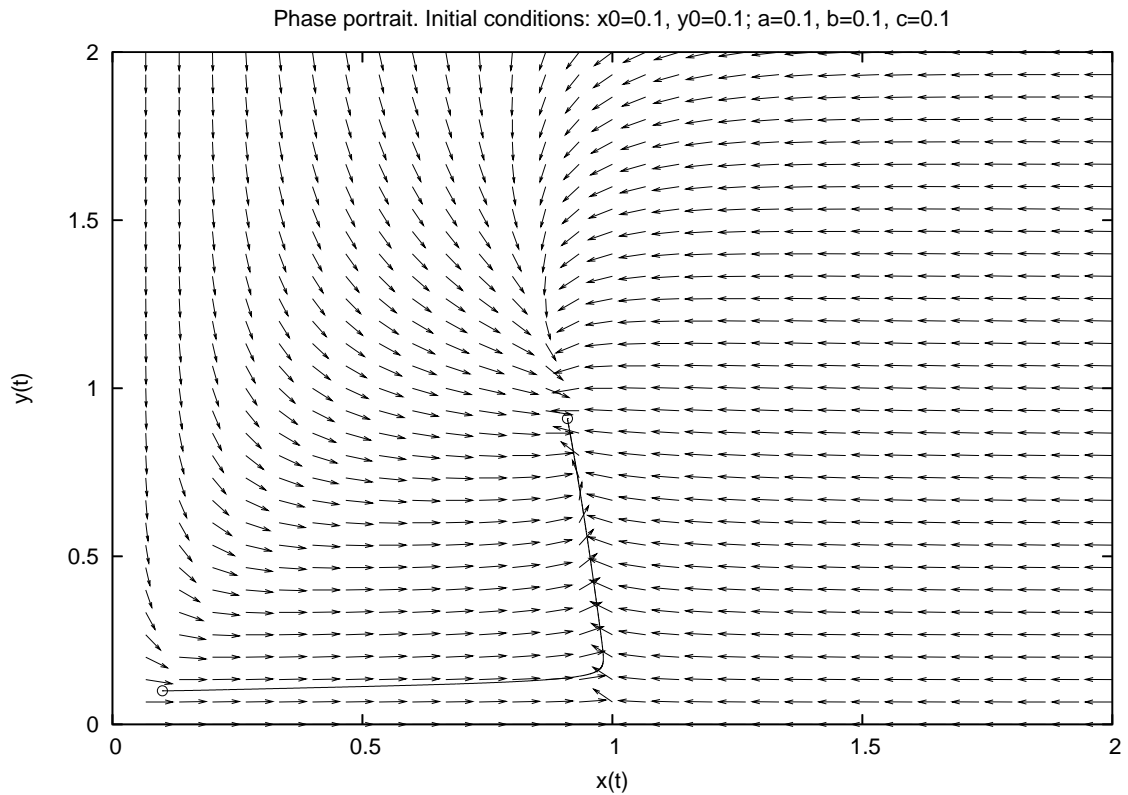


Figura 2: Ritratto di fase e storia temporale della versione realistica del modello Lotka-Volterra per $a = 0.1, b = 0.1, c = 0.1, (x_0, y_0) = (0.1, 0.1)$ e $t_f = 100$.

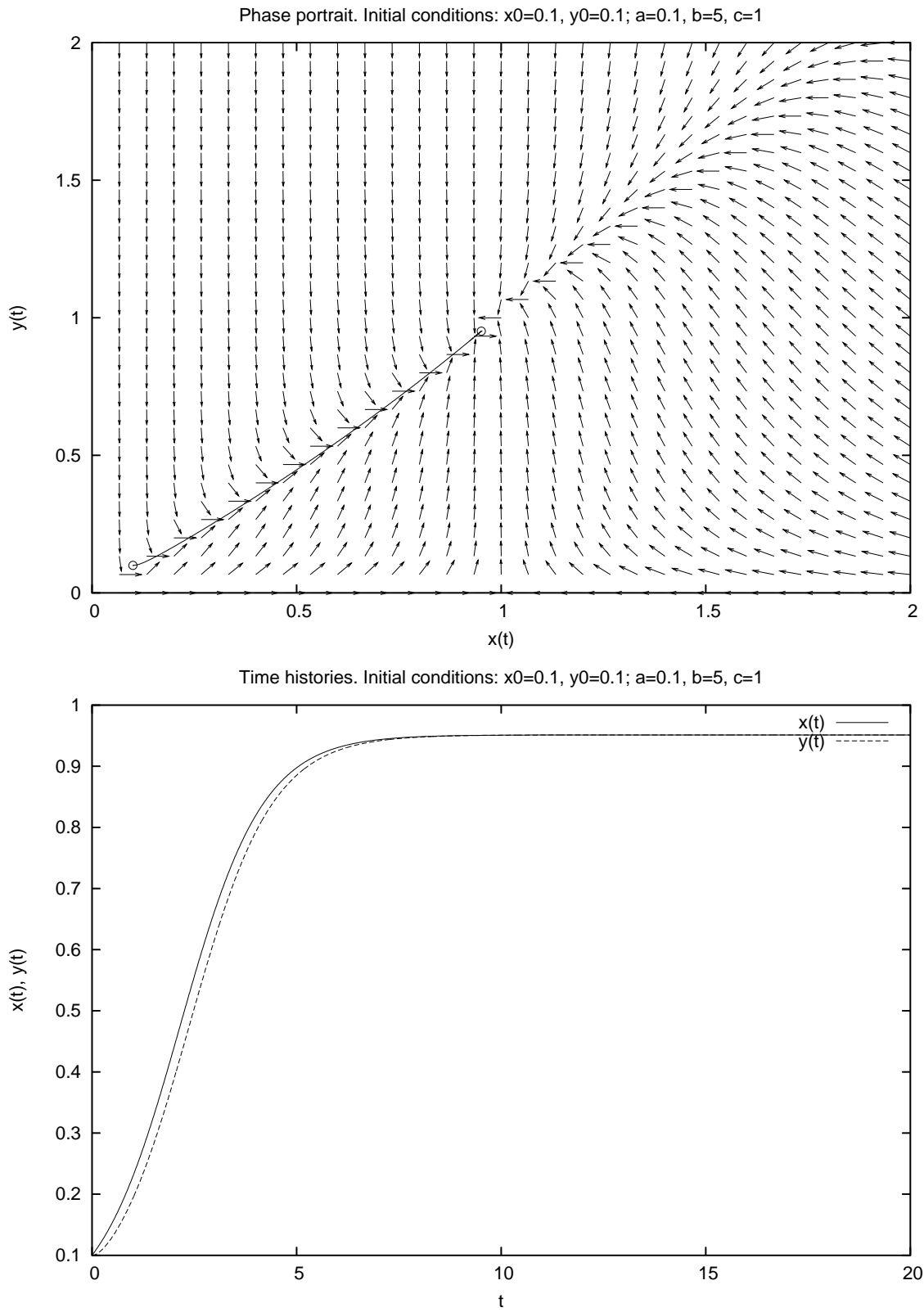


Figura 3: Ritratto di fase e storia temporale della versione realistica del modello Lotka-Volterra per $a = 0.1, b = 5, c = 1, (x_0, y_0) = (0.1, 0.1)$ e $t_f = 20$.

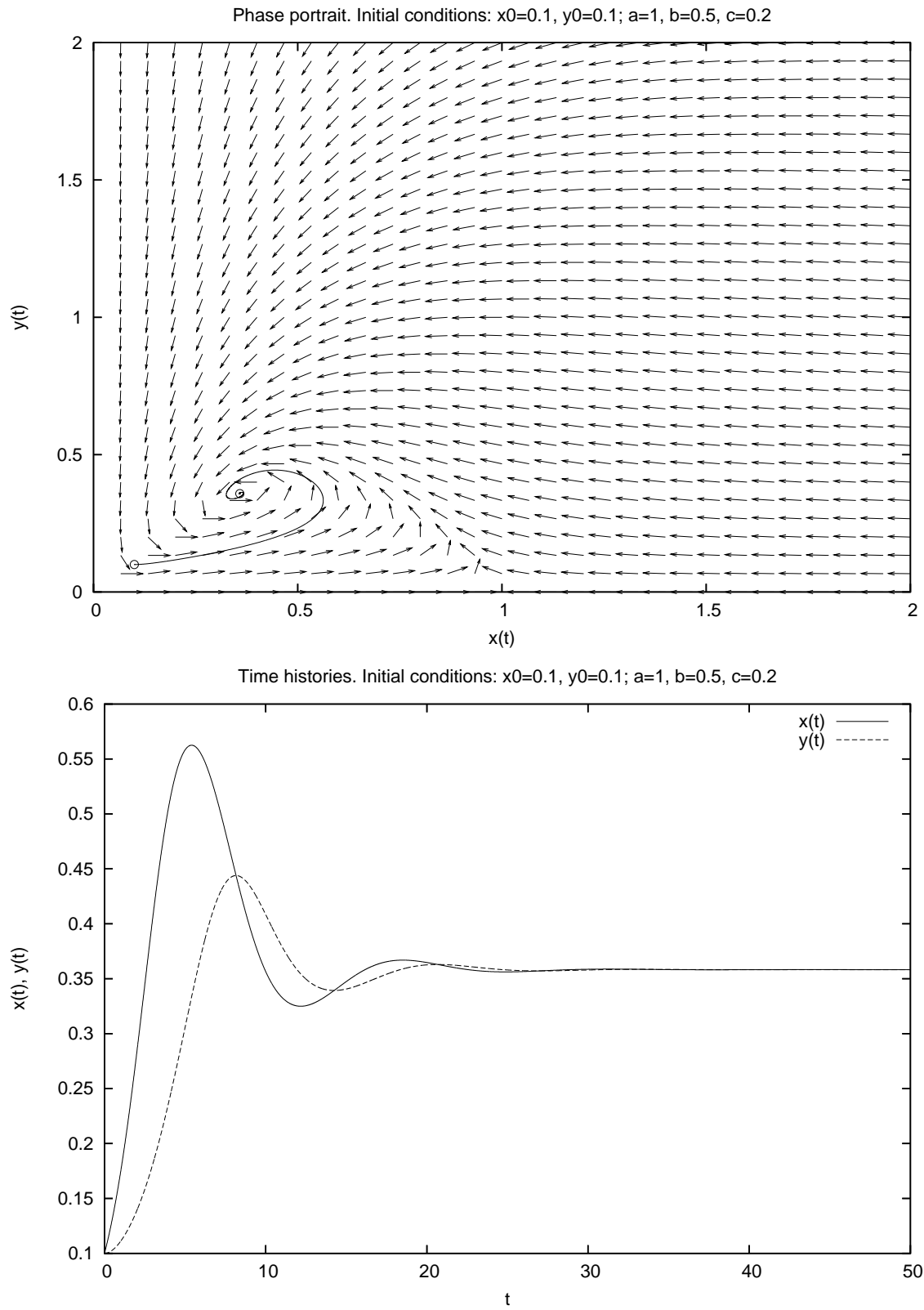


Figura 4: Ritratto di fase e storia temporale della versione realistica del modello Lotka-Volterra per $a = 1, b = 0.5, c = 0.2, (x_0, y_0) = (0.1, 0.1)$ e $t_f = 50$.

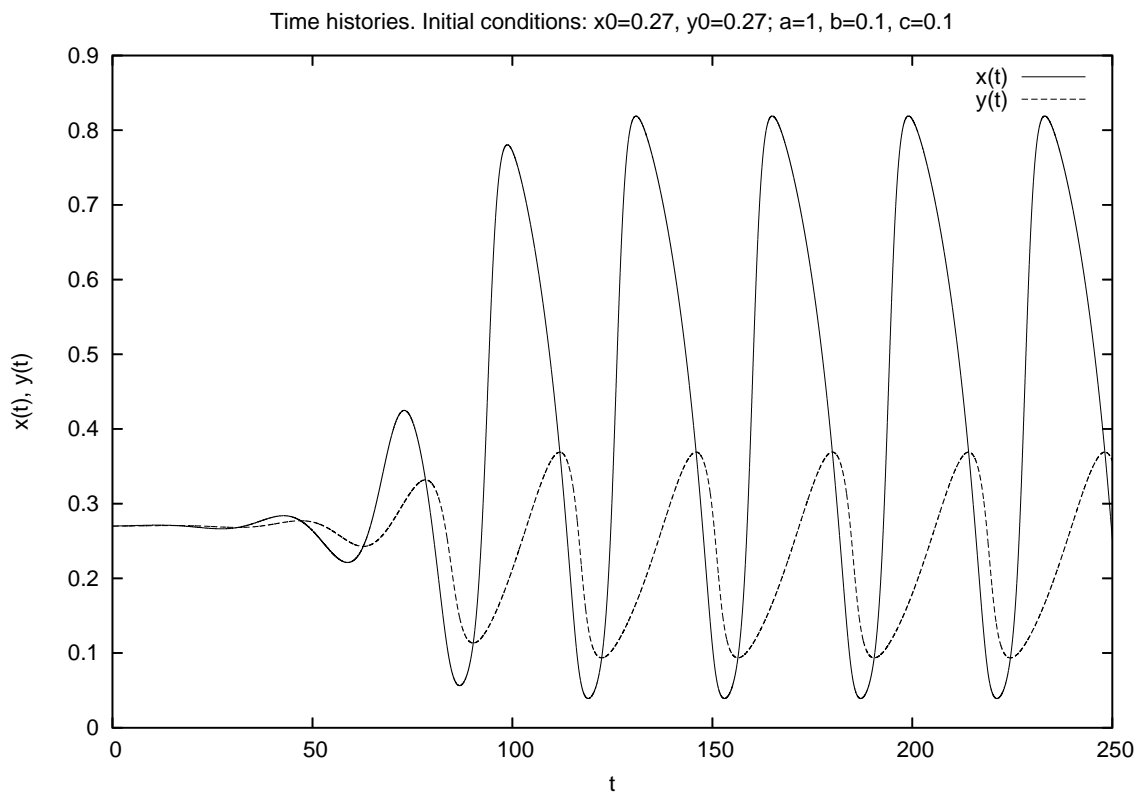
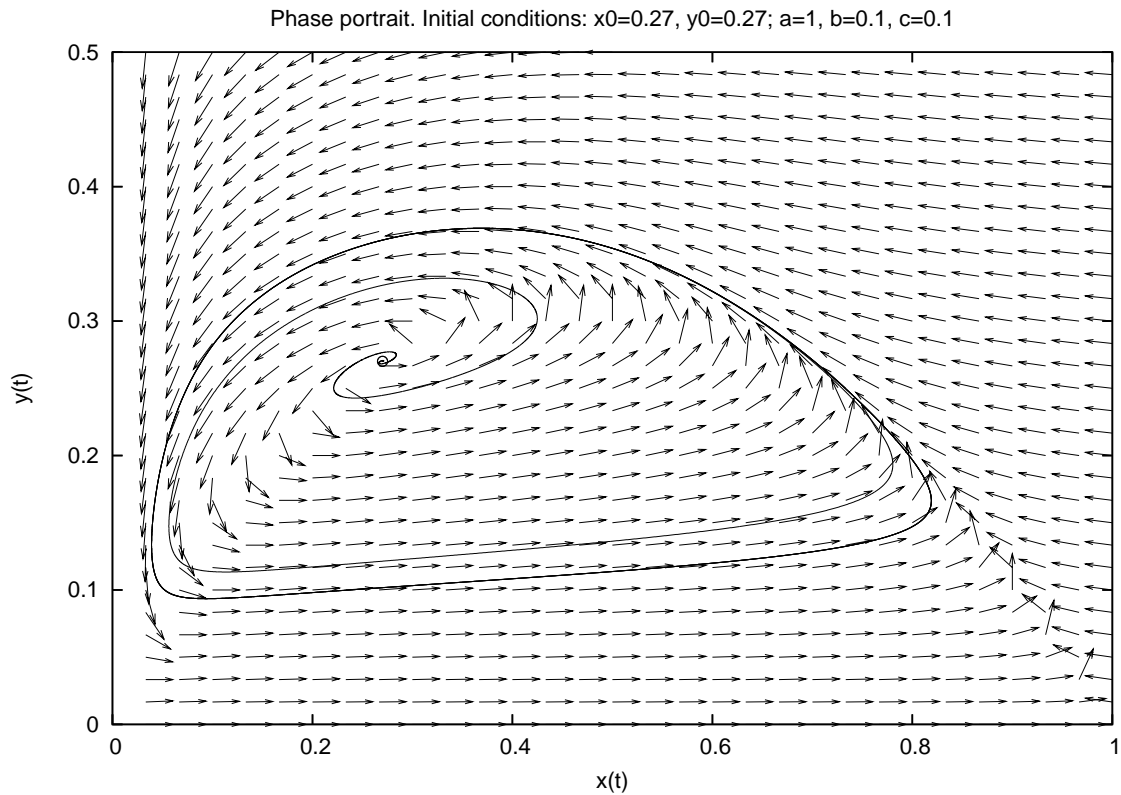


Figura 5: Ritratto di fase e storia temporale della versione realistica del modello Lotka-Volterra per $a = 1, b = 0.1, c = 0.1, (x_0, y_0) = (0.27, 0.27)$ e $t_f = 250$.

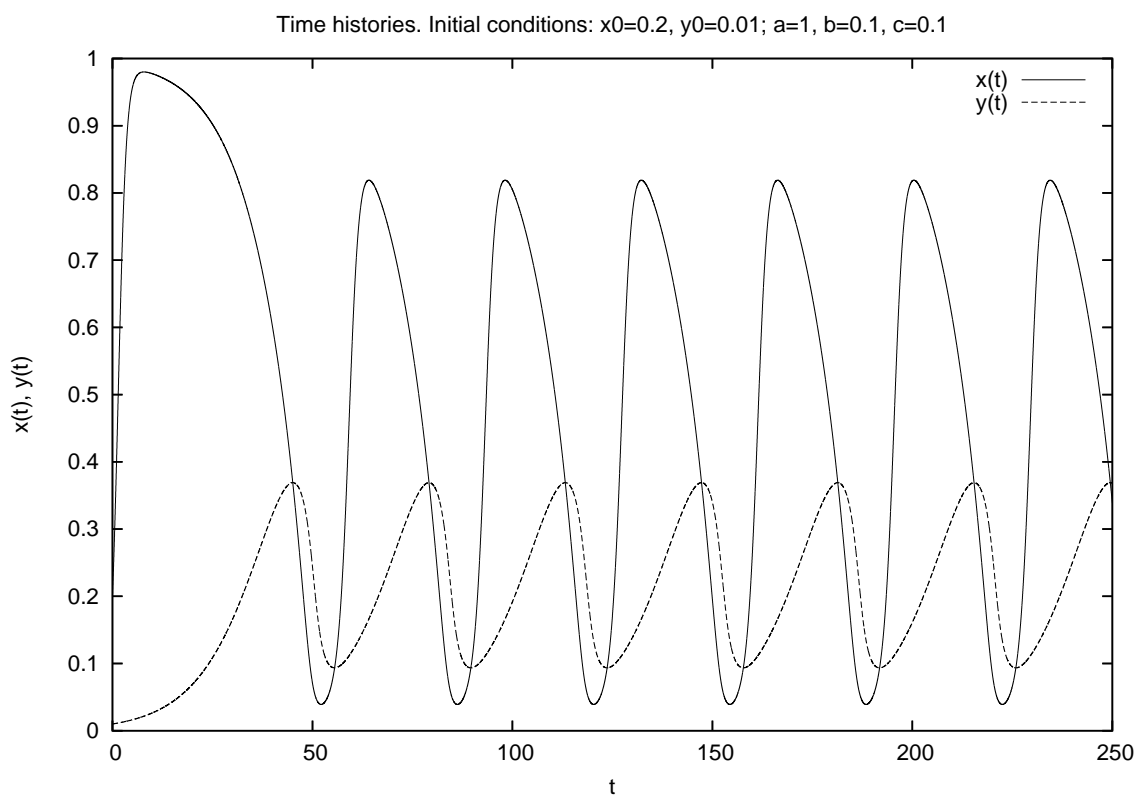
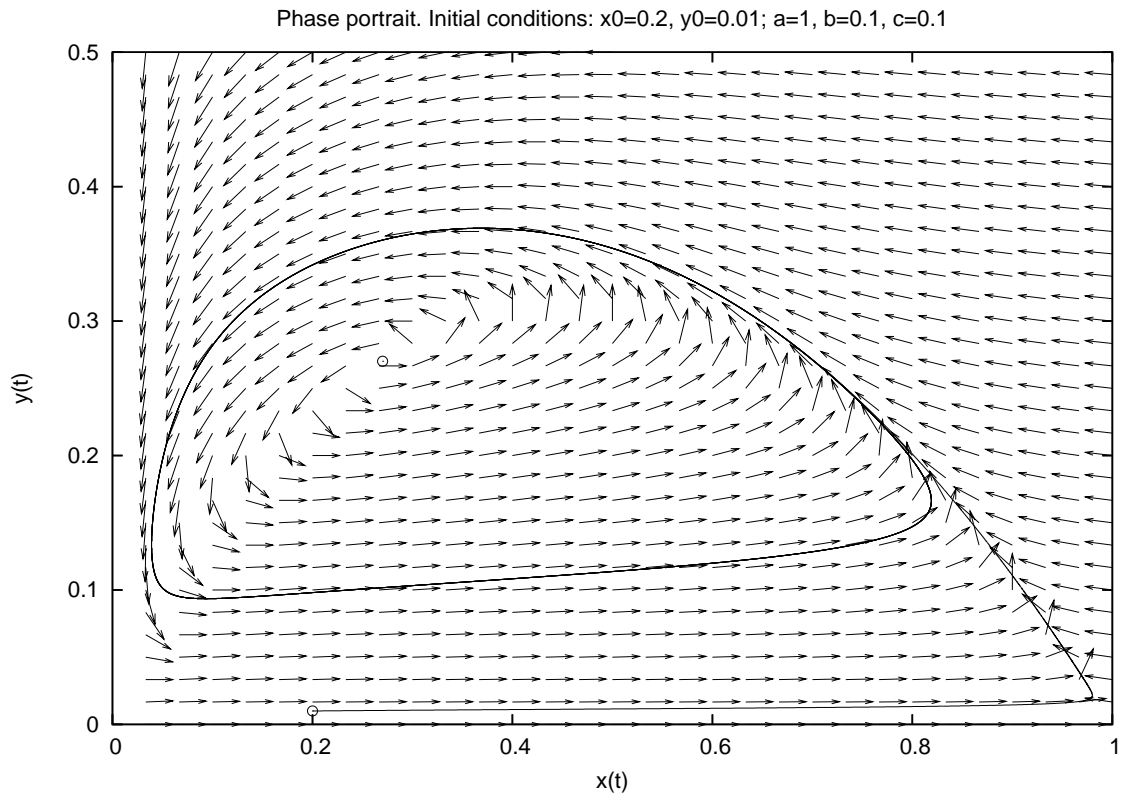


Figura 6: Ritratto di fase e storia temporale della versione realistica del modello Lotka-Volterra per $a = 1, b = 0.1, c = 0.1, (x_0, y_0) = (0.2, 0.01)$ e $t_f = 250$.

1.2 Esercizio

Studiare, al variare delle condizioni iniziali e del tempo finale, il comportamento del sistema

$$\begin{cases} x' = x \left[\left(1 + \frac{1}{1 + (x-2)^2} \right) - y \right] \\ y' = y [x - (y+1)]. \end{cases}$$

1.2.1 Risoluzione

Il sistema dato è un caso particolare di sistemi del tipo

$$\begin{cases} x' = x [F(x) - y] = f(x, y) \\ y' = y [x - G(y)] = g(x, y), \end{cases}$$

con $F(x) = 1 + \frac{1}{1 + (x-2)^2}$ e $G(y) = y + 1$, che ammette come punti di equilibrio l'origine e $(x_0, y_0) \approx (2.6823, 1.6823)$. Si può dimostrare che l'origine è un punto di sella (e quindi instabile) in quanto $\lambda_1 = F(0) > 0$ e $\lambda_2 = -G(0) < 0$, mentre l'altro punto di equilibrio, se è come in figura 7, risulta essere stabile (a piccole perturbazioni) qualsiasi siano $F(x)$ e $G(y)$ essendo la parte reale degli autovalori negativa.

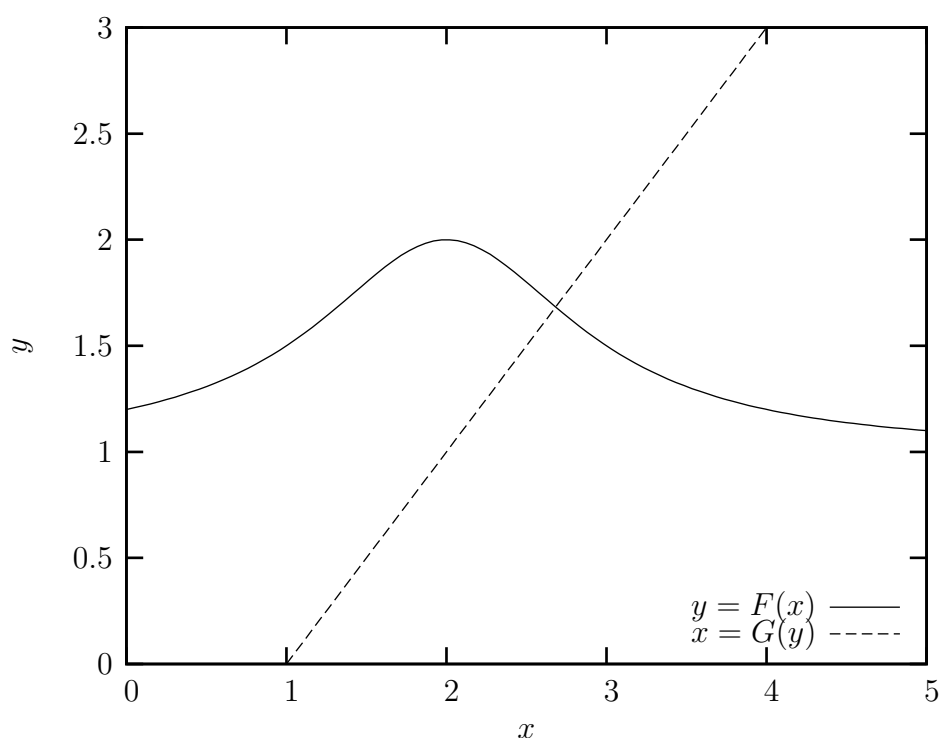


Figura 7: Luoghi dei punti $y = F(x)$ e $x = G(y)$, ovvero $f(x, y) = 0$ e $g(x, y) = 0$.

Pertanto, perturbando l'equilibrio, ovvero partendo da una condizione iniziale sufficientemente vicina al punto di equilibrio, le traiettorie ne vengono attratte e compiono orbite piuttosto ristrette riportandosi rapidamente sulla soluzione di equilibrio (vedi figura 8).

Al contrario, dipendentemente da quanto è lontana la condizione iniziale dal punto di equilibrio, si possono ottenere traiettorie con ampie oscillazioni (vedi figure 9–11).

La condizione che discrimina tra l'uno e l'altro caso, come si può facilmente intuire dalla figura 7, è dove la traiettoria interseca la linea $f(x, y) = 0$. Se questo avviene per $x < \bar{x}$, dove \bar{x} è la x dove $y = F(x)$ raggiunge il massimo, allora si hanno ampie oscillazioni, altrimenti no.

Utilizzando il file `tp.m` di seguito riportato si possono fare diverse prove al variare della condizione iniziale e del tempo finale di integrazione. **Si ricordi di cambiare gli estremi della finestra di visualizzazione dipendentemente dalla condizione iniziale.**

```
% Name:      tp.m
% Author:    Simone Zuccher
% Created:   24 May 2007
% Purpose:   solve the threshold phenomena system
%            u'= u(1+ 1/(1+(u-2)^2) - v)
%            v'= v(u - (v+1))
%            given u0 and v0
% Input:     see file
% Output:    1. plot of u(t) versus v(t) together with the vector field
%            2. plot time histories of u(t), v(t)
% Modified:
%
% The non-negative equilibrium points are the following
%
% [u = 2.353209961160612, v = 1.353209961160612]
%
%
% Clear all variables
clear all;

% Window ranges
xmin=0;
xmax=5;
ymin=0;
ymax=3;

% Equilibrium points
eq = [ 2.682327816337427 1.682327816337427];
disp('Equilibrium points:');
disp(eq);
```

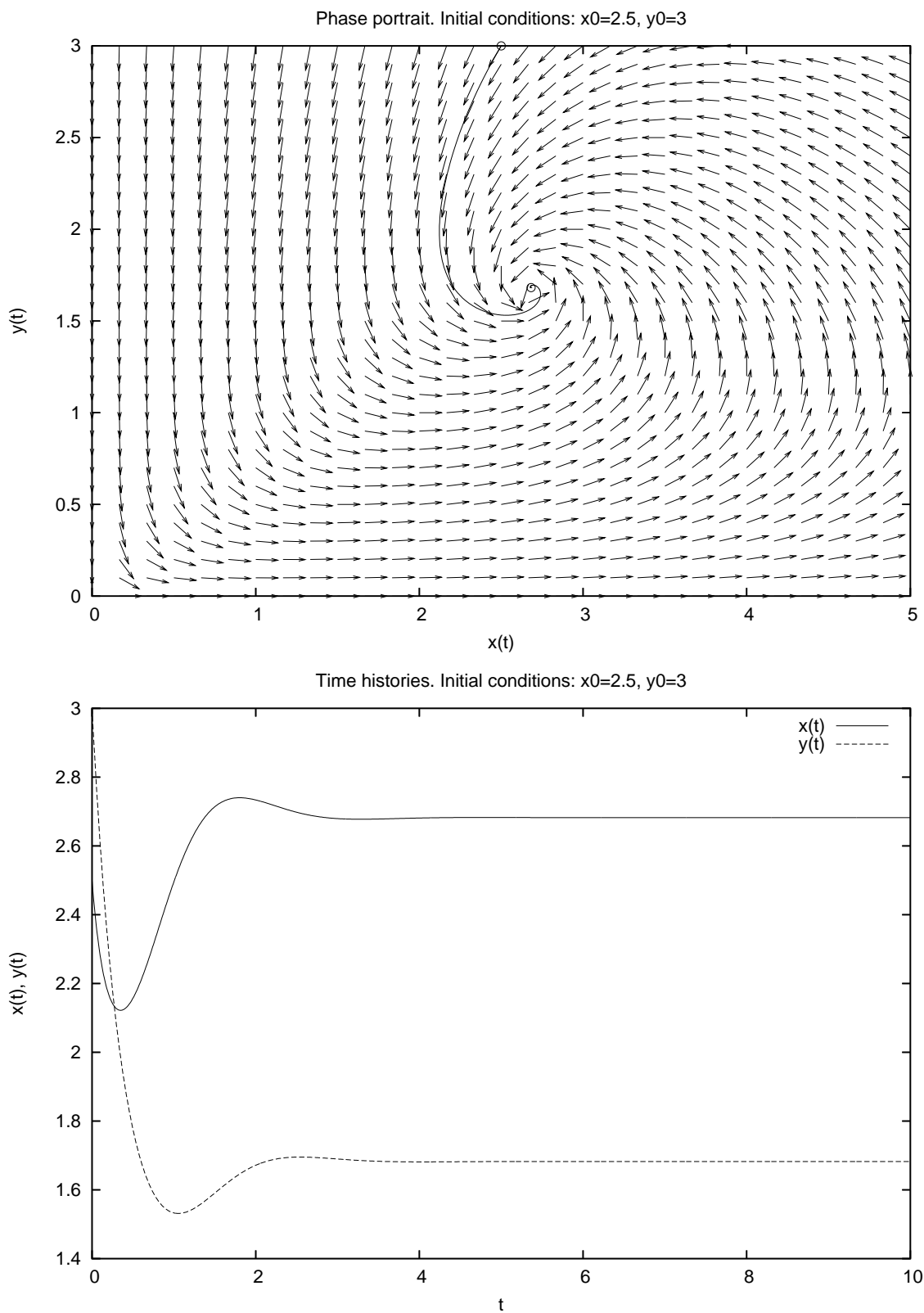


Figura 8: Ritratto di fase e storia temporale per $(x_0, y_0) = (2.5, 3.0)$ e $t_f = 10$.

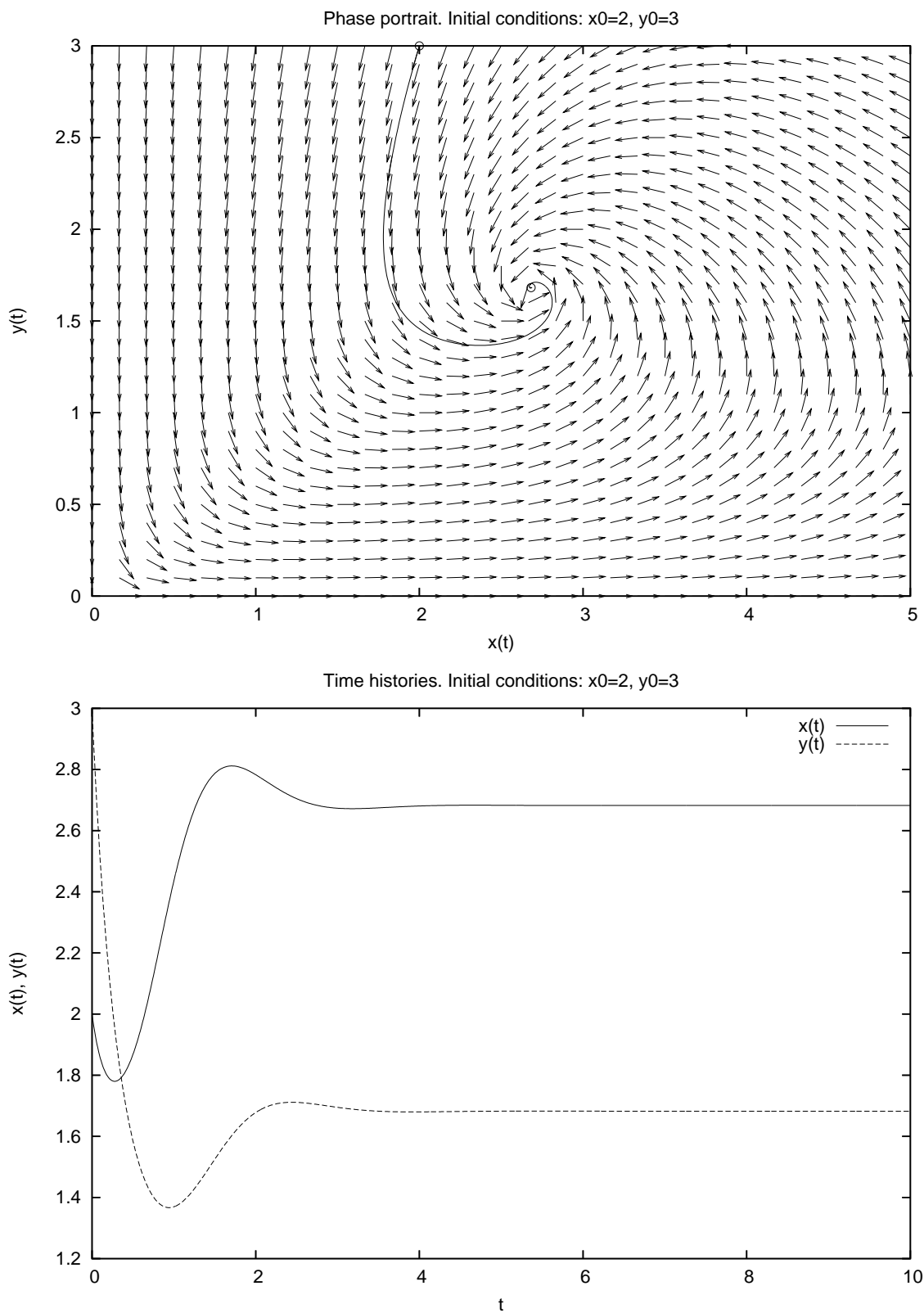


Figura 9: Ritratto di fase e storia temporale per $(x_0, y_0) = (2.0, 3.0)$ e $t_f = 10$.

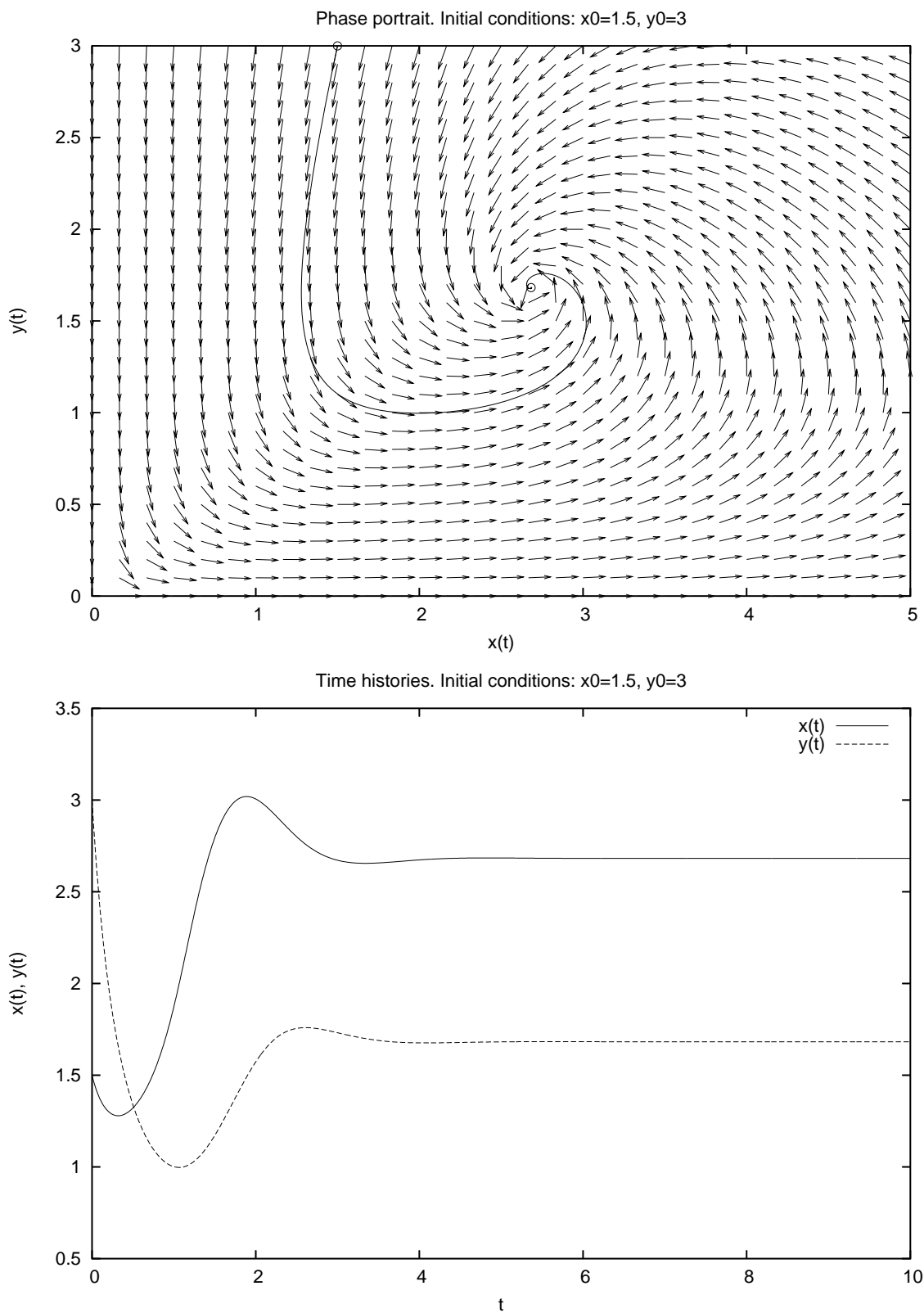


Figura 10: Ritratto di fase e storia temporale per $(x_0, y_0) = (1.5, 3.0)$ e $t_f = 10$.

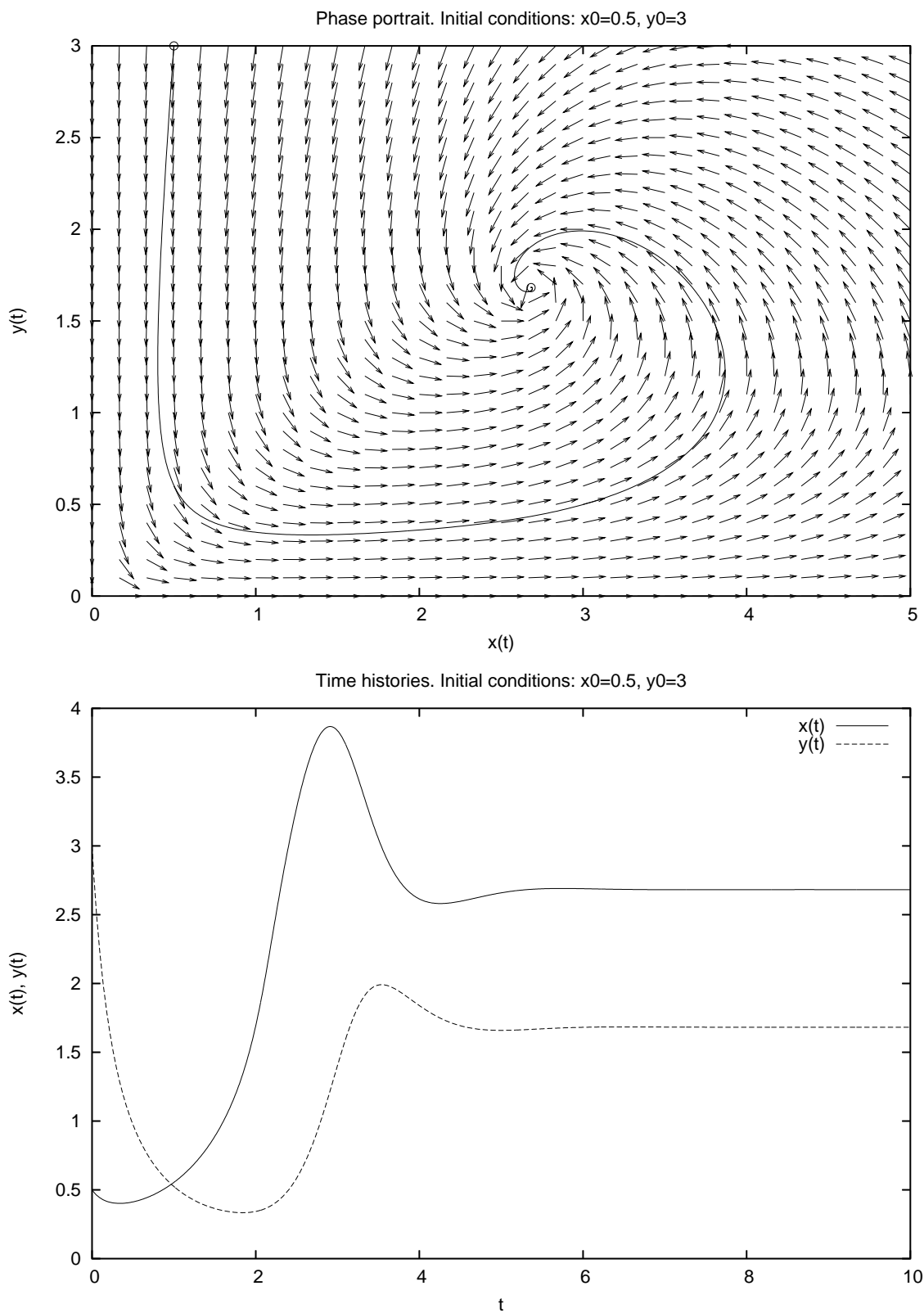


Figura 11: Ritratto di fase e storia temporale per $(x_0, y_0) = (0.5, 3.0)$ e $t_f = 10$.

```

% Set initial conditions
x0=input('Insert initial conditions [x0 y0]: ');

% Set final time for integration
tmax=input('Insert final time: ');

disp('Initial condition:');
disp(x0);

% Time parameters
tmin=0;
dt=.01;
% Create time
t = tmin:dt:tmax;

% dx and dy used only for vectors
dx=abs(xmax-xmin)/30;
dy=abs(ymax-ymin)/30;
% rescales vector size
scale=0.027*max(abs(xmax-xmin),abs(ymax-ymin));

% Definition of the dynamical system
function xdot=dsys(x, t)
    u = x(1);
    v = x(2);
    xdot(1) = u*(1+ 1/(1+(u-2)^2) - v);
    xdot(2) = v*(u - (v+1));
endfunction

__gnuplot_set__ nokey

setax=[xmin xmax ymin ymax];
axis(setax)

[X, Y] = meshgrid(xmin:dx:xmax, ymin:dy:ymax);
DX = X.*(1+ 1./(1+(X-2).^2) - Y);
DY = Y.*(X - (Y+1));
L = sqrt(DX.^2 + DY.^2);
mytitle=["Phase portrait. Initial conditions: x0=" num2str(x0(1)) \
        ", y0=" num2str(x0(2))];
__gnuplot_set__ nokey
__gnuplot_set__ xlabel 'x(t)'
__gnuplot_set__ ylabel 'y(t)'
title(mytitle)

% Plot vector field
quiver(X, Y, scale*DX./L, scale*DY./L)

```

```

hold on;

% Plot all equilibrium points
plot( eq(:,1), eq(:,2), '*k')

x = lsode("dsys", x0, t)';
plot( x(1,1), x(2,1), '*k', x(1,:), x(2,:), '-r')
hold off;

% Wait for keypressed
disp('Please press a key to continue...');
pause();
mytitle=["Time histories. Initial conditions: x0=" num2str(x0(1)) \
        ", y0=" num2str(x0(2))];
__gnuplot_set__ auto
__gnuplot_set__ xlabel 't'
__gnuplot_set__ ylabel 'x(t), y(t)'
title(mytitle)
__gnuplot_set__ key

% Plot time histories
plot( t, x(1,:), '-r;x(t);', t, x(2,:), '-g;y(t);')

```

1.3 Esercizio

Studiare, al variare delle condizioni iniziali e del tempo finale, il comportamento del sistema

$$\begin{cases} x' = a(y - x) \\ y' = -xz + bx - y \\ z' = xy - cw. \end{cases}$$

1.3.1 Risoluzione

Il sistema in questione è il ben noto sistema di Lorenz inizialmente utilizzato in campo meteorologico. Esso ammette tre punti di equilibrio, di cui uno è l'origine e gli altri sono simmetrici rispetto all'asse z e sono $(x, y, z) = (\sqrt{bc - c}, \sqrt{(b - 1)c}, b - 1)$ e $(x, y, z) = (-\sqrt{bc - c}, -\sqrt{(b - 1)c}, b - 1)$. Essi sono tutti instabili, con uno o due autovalori a parte reale positiva, come si può notare in figura 12.

Phase portrait. Initial conditions: $x_0=3, y_0=15, z_0=1, a=10, b=28, c=2.667$

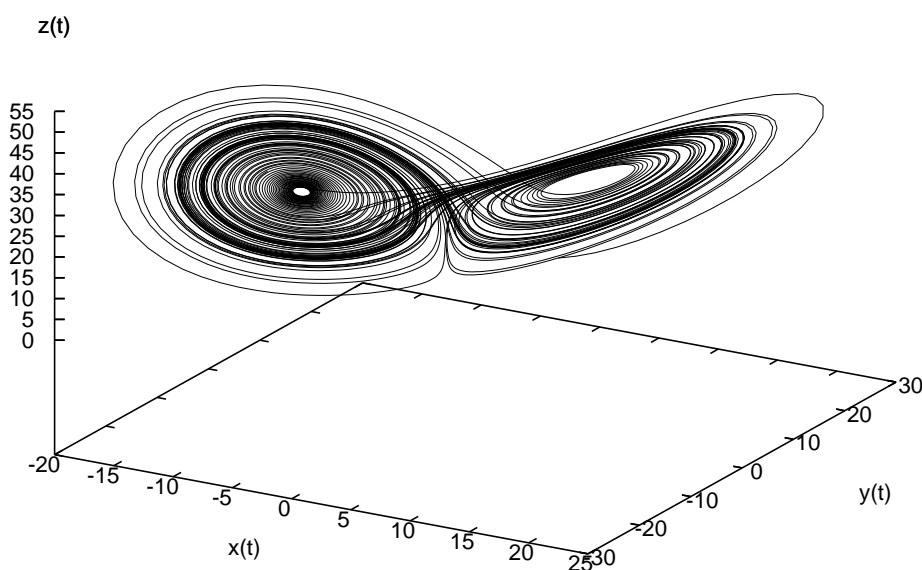


Figura 12: Attrattore di Lorenz ottenuto con $a = 10, b = 28, c = 8/3, (x_0, y_0, z_0) = (3, 15, 1)$ e $t_f = 100$.

Utilizzando il file `mylorenz.m` di seguito riportato si possono fare diverse prove al variare delle costanti a, b, c , della condizione iniziale e del tempo finale di integrazione.

```
% Name:      mylorenz.m
% Author:    Simone Zuccher
% Created:   24 May 2007
```

```

% Purpose: solve the classic Lorenz system
%          u' = a(v-u)
%          v' = -uw+bu -v
%          z' = uv-cw
%          given u0 and v0
% Input:   see file
% Output:  1. plot of u(t), v(t), w(t)
%          2. plot time histories of u(t), v(t), w(t)
% Modified:
%
% The non-negative equilibrium points are the following
%
% [u = 0, v = 0, w = 0]
%
% [u = SQRT(b c - c), v = SQRT(b - 1) SQRT(c), w = b - 1],
%
% [u = - SQRT(b c - c), v = - SQRT(b - 1) SQRT(c), w = b - 1],
%
%
% Clear all variables
clear all;

% Model constant
global aa;

% Set initial conditions
aa=input('Insert coefficients [a b c]: ');
%aa=[10 28 8./3.]

% Equilibrium points
eq = [ 0 0 0];
eq = [eq; sqrt(aa(2)*aa(3)-aa(3)) sqrt(aa(2)-1)*sqrt(aa(3)) aa(2)-1];
eq = [eq; -sqrt(aa(2)*aa(3)-aa(3)) -sqrt(aa(2)-1)*sqrt(aa(3)) aa(2)-1];

disp('Equilibrium points:');
disp(eq);

% Set initial conditions
x0=input('Insert initial conditions [x0 y0]: ');
%x0=[3 15 1];

% Set final time for integration
tmax=input('Insert final time: ');
%tmax=100;

disp('Initial condition:');

```

```

disp(x0);

% Time parameters
tmin=0;
dt=.01;
% Create time
t = tmin:dt:tmax;

% Definition of the dynamical system
function xdot=dsys(x, t)
    global aa;
    u = x(1);
    v = x(2);
    w = x(3);
    xdot(1) = aa(1)*(v-u);
    xdot(2) = -u*w+aa(2)*u - v;
    xdot(3) = u*v-aa(3)*w;
endfunction

x = lsode("dsys", x0, t');

mytitle=["Phase portrait. Initial conditions: x0=" num2str(x0(1)) \
        ", y0=" num2str(x0(2)) ", z0=" num2str(x0(3)) \
        ", a=" num2str(aa(1)) ", b=" num2str(aa(2)) \
        ", c=" num2str(aa(3))];
__gnuplot_set__ nokey
__gnuplot_set__ xlabel 'x(t)'
__gnuplot_set__ ylabel 'y(t)'
__gnuplot_set__ zlabel 'z(t)'
__gnuplot_set__ parametric
__gnuplot_set__ view
%__gnuplot_set__ view 120, 30
title(mytitle)
gspplot x

```

2 Esclusione competitiva

Questa è una piccola appendice a quanto visto l'esercitazione precedente riguardo alla competizione e cooperazione.

2.1 Esercizio

Studiare, al variare delle condizioni iniziali, dei coefficienti $\alpha, \beta, \gamma, \delta$ tutti positivi e tali che $0 < \delta \ll \beta$, e del tempo finale, il comportamento del sistema

$$\begin{cases} x' &= \alpha x - \beta x^2 - \gamma xy \\ y' &= \alpha y - (\beta - \delta)y^2 - \gamma xy. \end{cases}$$

2.1.1 Risoluzione

Il sistema dato è un caso particolare di un sistema dinamico del tipo

$$\begin{cases} x' &= \alpha_1 x + a_{11}x^2 + a_{12}xy \\ y' &= \alpha_2 y + a_{21}xy + a_{22}y^2 \end{cases} \quad (1)$$

con $\alpha_1 = \alpha_2 = \alpha$, $a_{12} = a_{21} = -\gamma$, $a_{11} = -\beta$ e $a_{22} = -(\beta - \delta)$. Questo, in pratica, significa che le due popolazioni hanno leggi di crescita praticamente simili tranne che per il coefficiente intraspecifico, che varia leggermente. Pertanto, si può studiare utilizzando lo script `compcoop.m`, riportato nell'esercitazione precedente.

In figura 13 viene riportato il caso $\alpha = 10, \beta = 0.1, \gamma = 0.099, \delta = 0.002$, $(x_0, y_0) = (40, 40)$. Come si può notare i punti di equilibrio sono tre, di cui uno solo è stabile e la soluzione si porta proprio su questo stato di equilibrio che prevede l'estinzione della popolazione x e la sopravvivenza della y . Questo succede perchè, pur essendo le due leggi di crescita molto simili, la popolazione x ha un coefficiente di competizione intraspecifico leggermente maggiore dell'altra. La figura 14 è stata ottenuta con gli stessi valori dei coefficienti ($\alpha = 10, \beta = 0.1, \gamma = 0.099, \delta = 0.002$), ma cambiando la condizione iniziale a $(x_0, y_0) = (95, 5)$ in modo da essere molto vicini al punto di equilibrio dove la specie x sopravvive e la specie y si estingue. Nonostante questo, la specie x si estingue e sopravvive solo la y . Si noti che, per questa scelta dei parametri, $\beta - \delta = 0.098 < \gamma$.

In figura 15 viene riportato il caso $\alpha = 10, \beta = 0.1, \gamma = 0.097, \delta = 0.002$, $(x_0, y_0) = (95, 5)$. Innanzitutto va sottolineato il fatto che i punti di equilibrio sono quattro, di cui quello stabile è il nuovo comparso. Questa volta la scelta dei coefficienti è tale per cui $\beta - \delta = 0.098 > \gamma$ ed il cambiamento sostanziale sta nel fatto che esiste un punto di equilibrio stabile che prevede la sopravvivenza di entrambe le specie. Questo succede partendo anche da altri dati iniziali, come mostrato in figura 16.

2.2 Esercizio

Si dimostri analiticamente come, al variare dei coefficienti $\alpha, \beta, \gamma, \delta$ tutti positivi e tali che $0 < \delta \ll \beta$, varia la natura dei punti di equilibrio del sistema

$$\begin{cases} x' &= \alpha x - \beta x^2 - \gamma xy \\ y' &= \alpha y - (\beta - \delta)y^2 - \gamma xy. \end{cases}$$

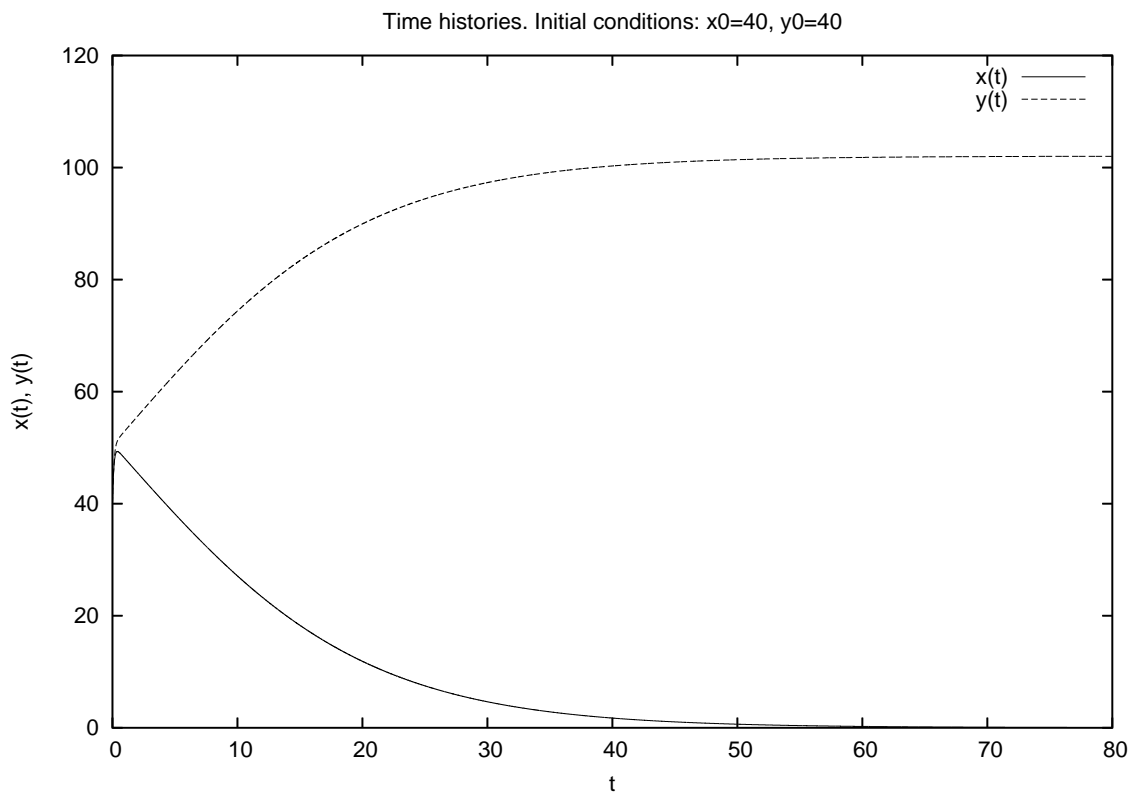
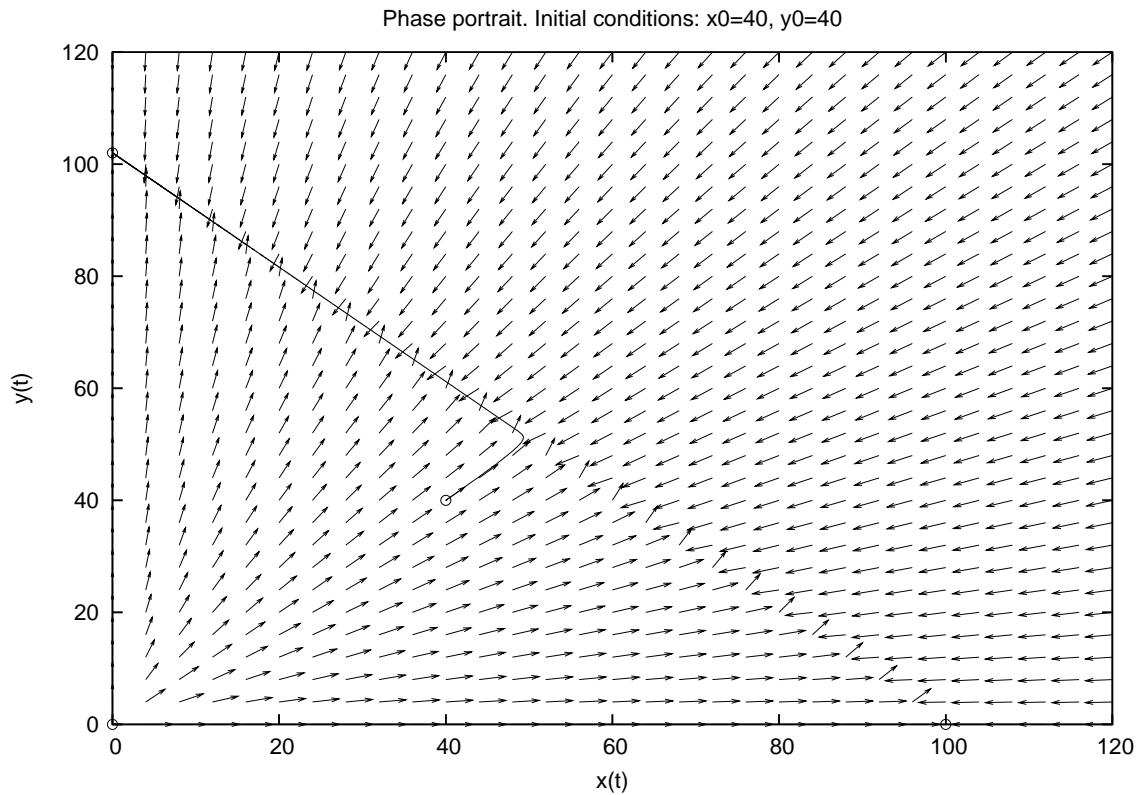


Figura 13: Ritratto di fase e storia temporale per $\alpha = 10, \beta = 0.1, \gamma = 0.099, \delta = 0.002$, $(x_0, y_0) = (40, 40)$ e $t_f = 80$.

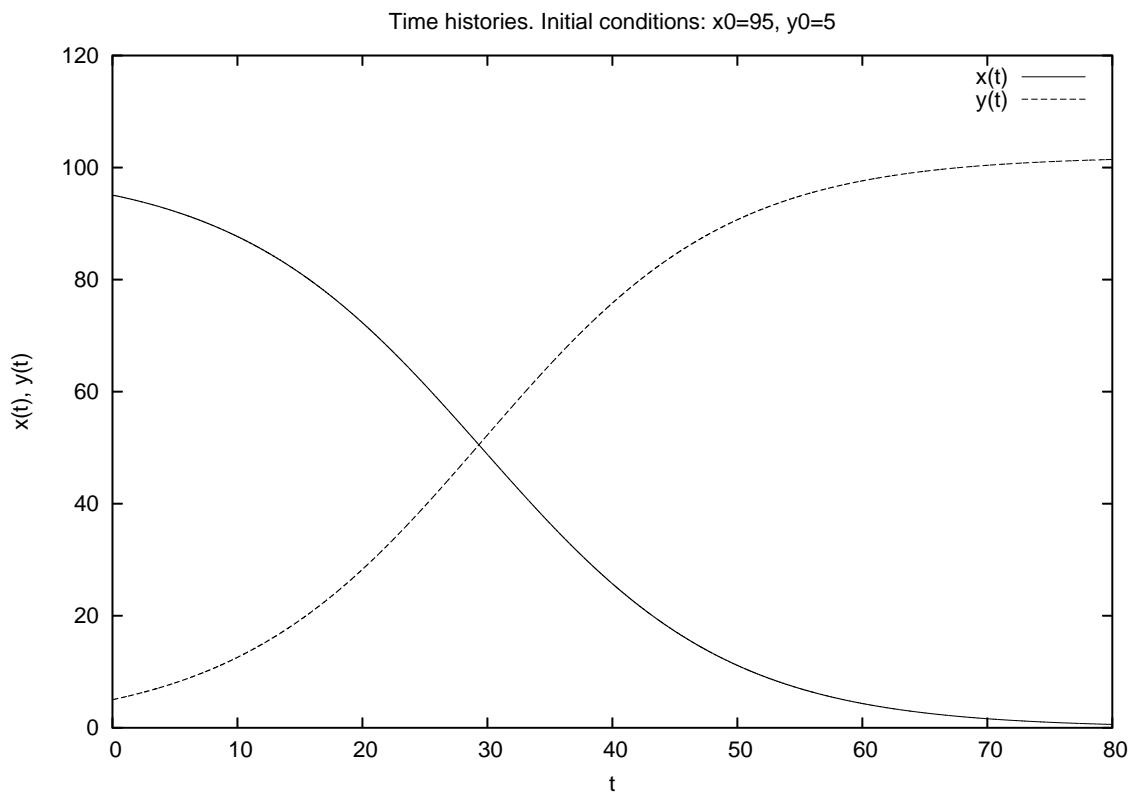
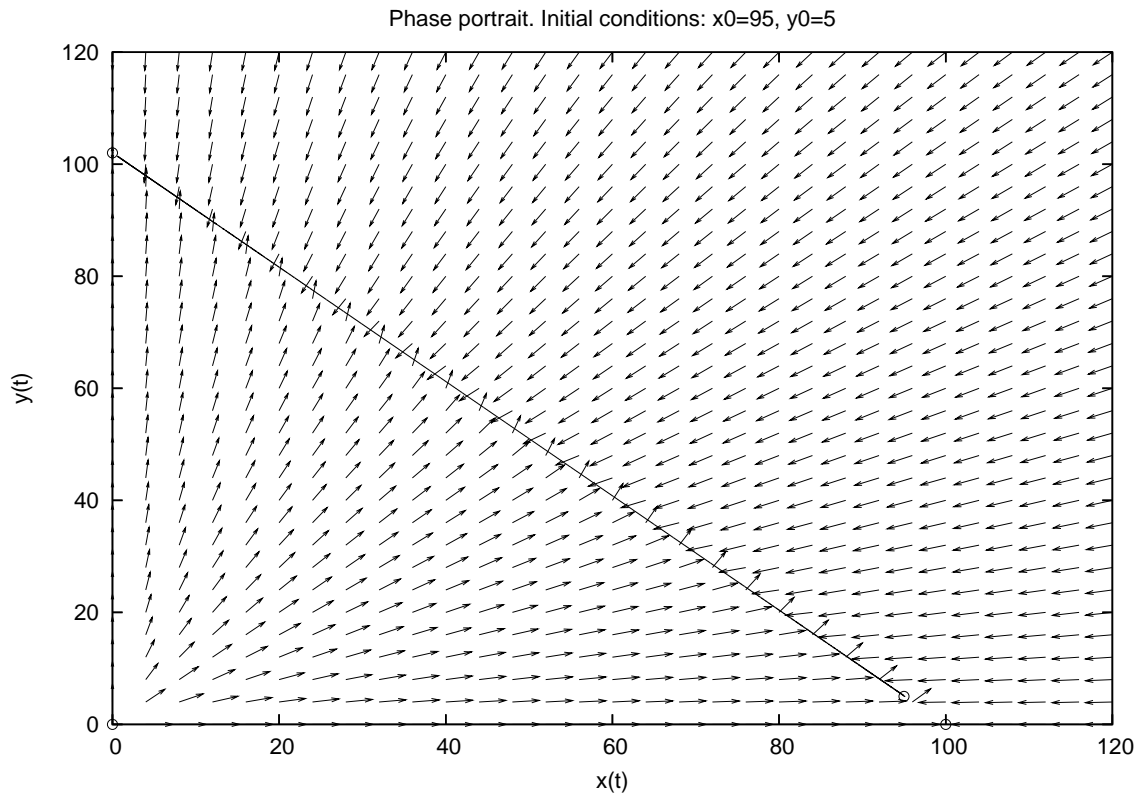


Figura 14: Ritratto di fase e storia temporale per $\alpha = 10, \beta = 0.1, \gamma = 0.099, \delta = 0.002$, $(x_0, y_0) = (95, 5)$ e $t_f = 80$.

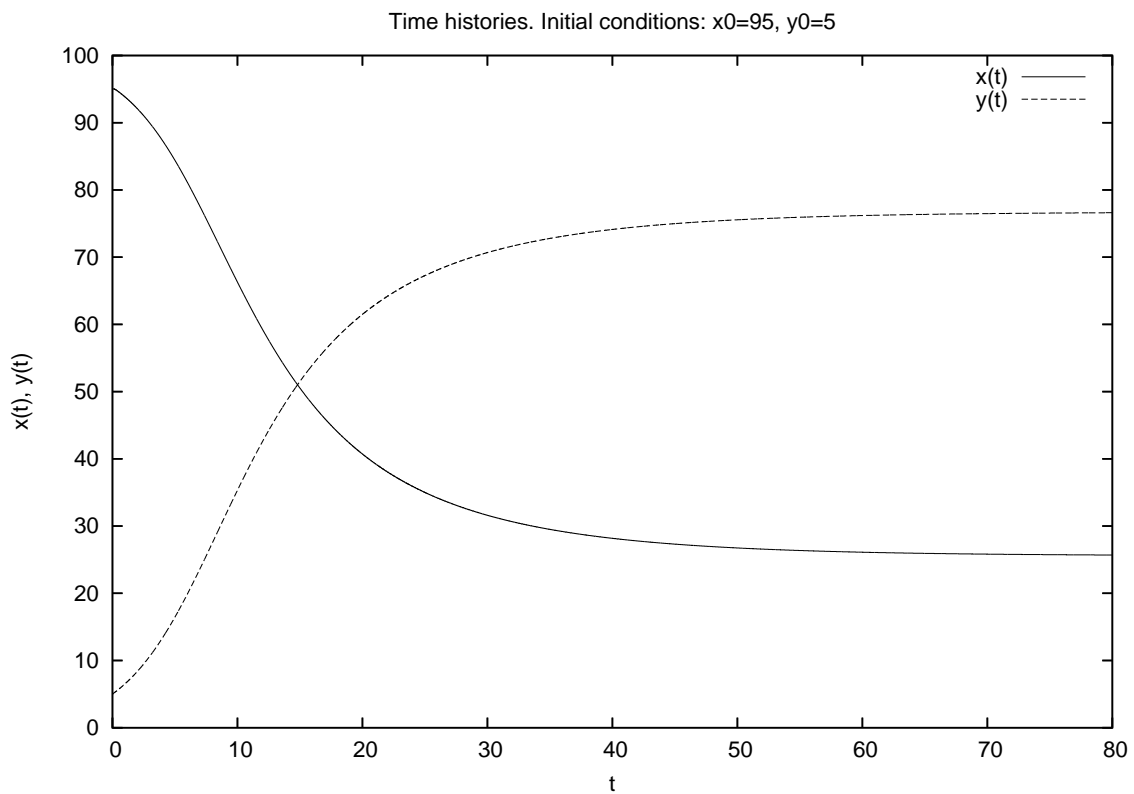
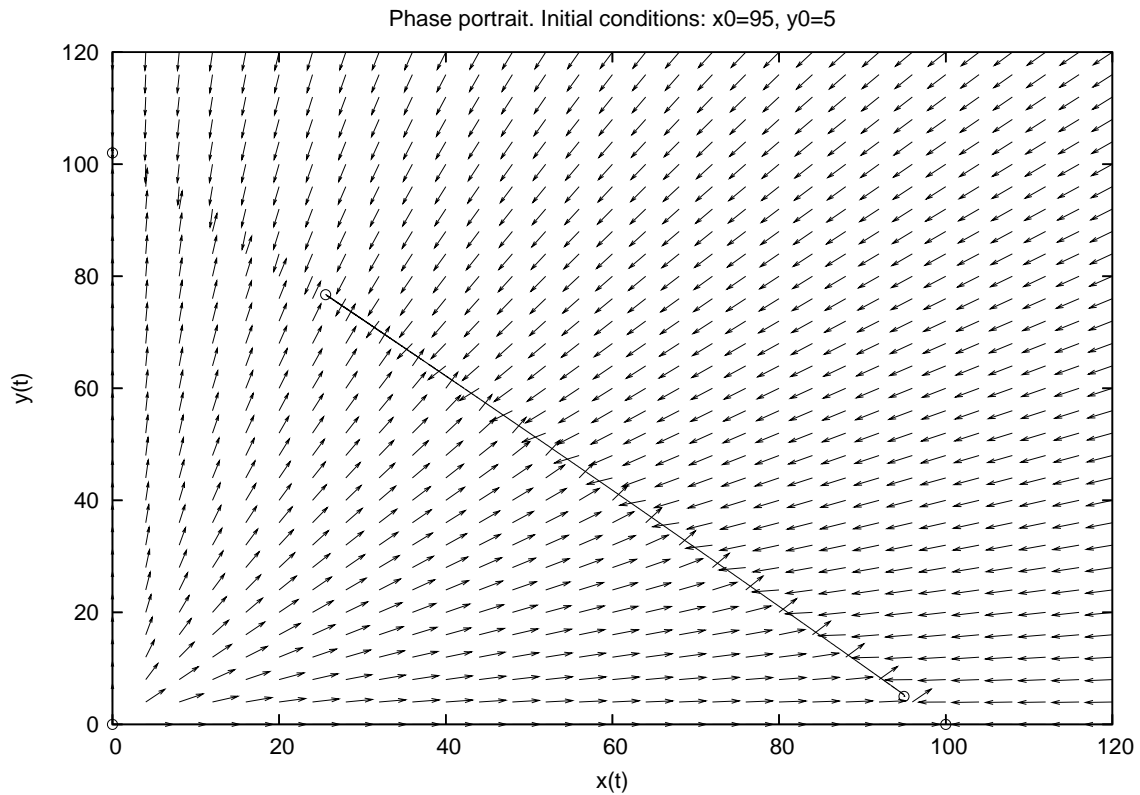


Figura 15: Ritratto di fase e storia temporale per $\alpha = 10, \beta = 0.1, \gamma = 0.097, \delta = 0.002$, $(x_0, y_0) = (95, 5)$ e $t_f = 80$.

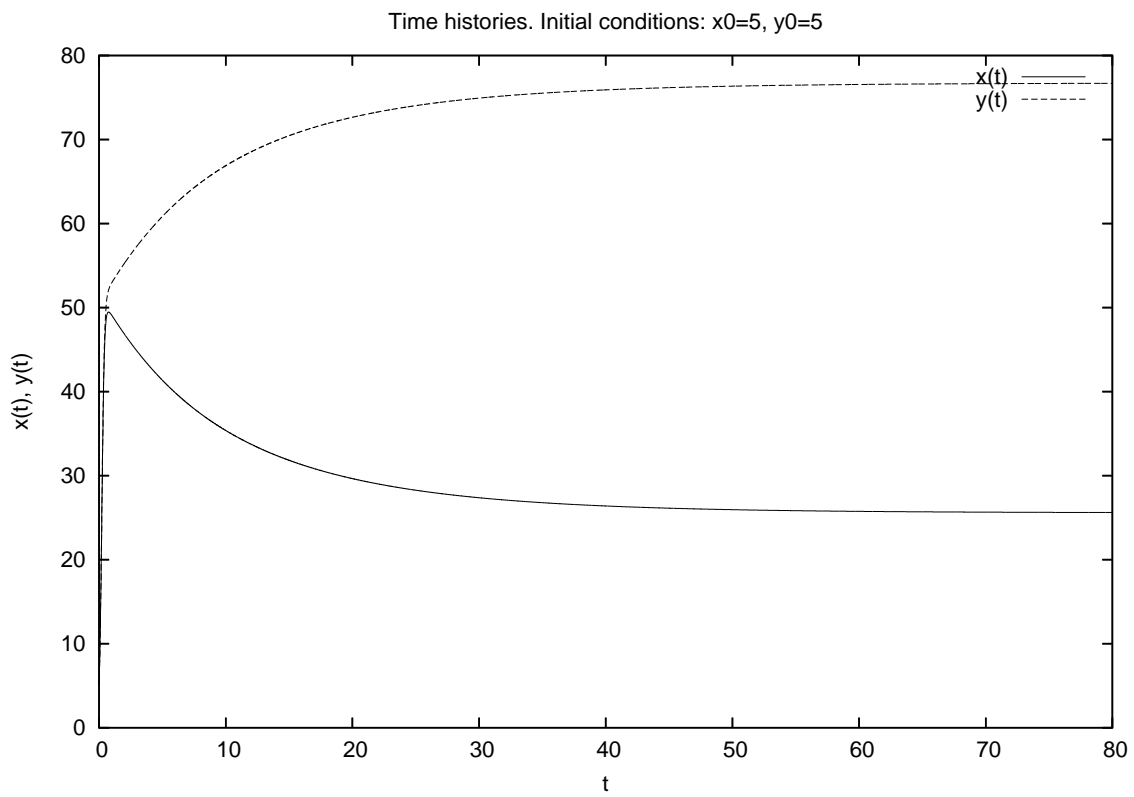
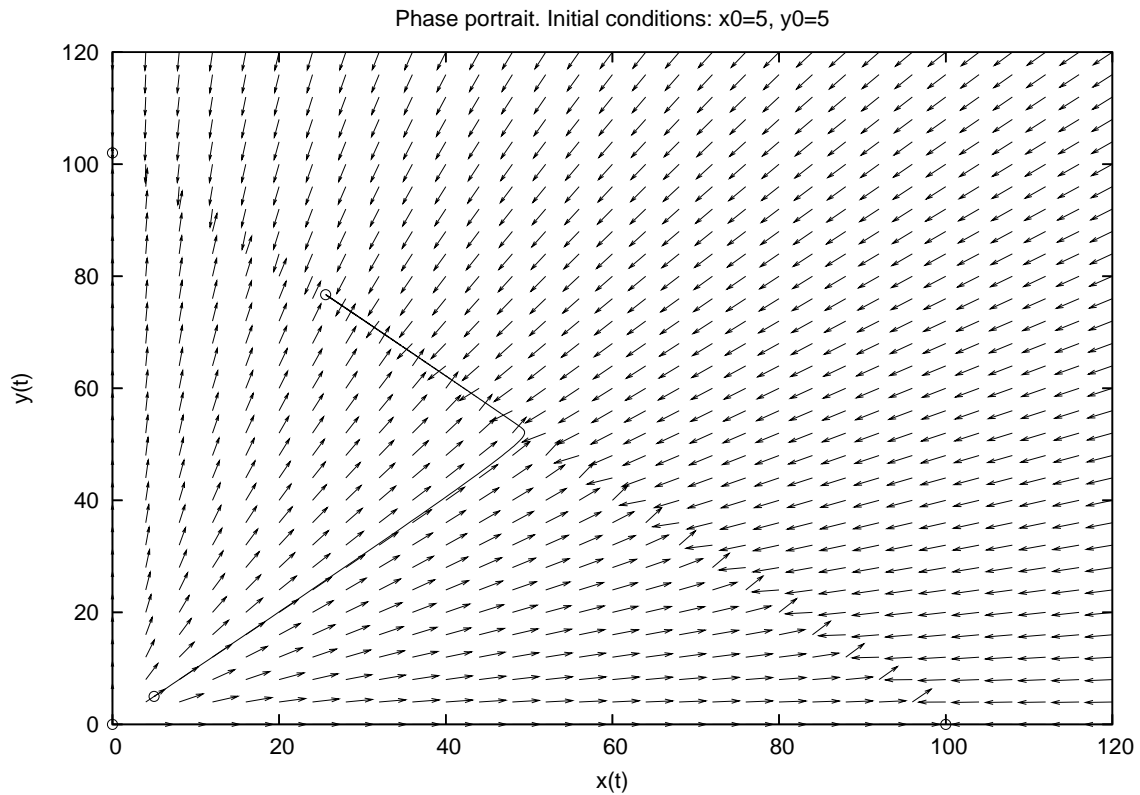


Figura 16: Ritratto di fase e storia temporale per $\alpha = 10, \beta = 0.1, \gamma = 0.097, \delta = 0.002$, $(x_0, y_0) = (5, 5)$ e $t_f = 80$.