Case Study: Bifurcation Analysis of Tumor-Immune ODE System

Abstract

1 Understanding Qualitative Changes

Simulations and null-cline analyzes illuminate the general behavior of the system. A Bifurcation Analysis gives the broader picture.

- Question: How do different QUALITATIVE behaviors arise as the PARAMETERS of the system are varied?

Recall the non-dimensionalized system of Kuznetsov [1]:

$$\frac{dE}{dt} = \sigma + \frac{\rho ET}{\eta + T} - \mu ET - \delta E$$

$$\frac{dT}{dt} = \alpha T(1 - \beta T) - ET$$

We carry out a bifurcation analysis on this system of equations.

1.1 Example question

Question: When is the tumor-free equilibrium stable?

Answer: When $\alpha < \frac{\sigma}{\delta}$ (using the non-dimensionalized parameters).

The parameters were estimated to be: $\alpha = 1.636, \sigma = .1181$, and $\delta = .3743$. Therefore, in this case the tumor-free equilibrium is UNSTABLE.
Question: In this case, what is the long-term fate of the system? Before we can answer, we need to see how the number of *EQUILIBRIA* and their

STABILITY change with the parameters.

Topic for discussion: Why would we be interested

1. Which parameters might reasonably be changed by treatment?
2. How? (For example, a bone marrow transplant might increase the value of σ; or immunotherapy - stimulation of the immune system by a vaccine, perhaps, might increase ρ or μ.)
3. How might surgery or radiation affect the model? (These therapies would reduce the number of cells, mainly the tumor cells, but also the immune cells. They may also reduce the immune response of the system to any remaining cancer by weakening the entire system.)

Try to sketch $f(T)$ and $g(T)$ on the same axes (the positive quadrant only) for various parameter values. The main point is: what features of these graphs will result in a qualitative change in the dynamics of the system. Intersections (in the positive quadrant) correspond to (physically relevant) equilibria, and the relative orientation of the two curves at the points of interesection determines the stability of the equilibria. Is it possible to identify parameter values which cause a change in the number of intersections, for example?
medskip

Biologically, reducing α corresponds to slowing the rate of tumor growth. The disappearance of the stable equilibrium corresponding to a high tumor burden.

1.2 The Effect of Shifting the T-Nullcline

Question: As α is decreased and the T-nullcline is shifted down, the patient is eventually cured. The first bifurcation occurs when the two equilibria coalesce and then disappear. This is called a saddle-node or fold bifurcation. The tumor-free equilibrium, however, remains unstable. This situation corresponds to the T-nullcline. As α is decreased further, the two equilibria coalesce and switch relative positions. At this point the two equilibria also switch stability, and the tumor-free equilibrium becomes stable. This type of bifurcation is called a transcritical bifurcation. Graphically, this corresponds to the T-nullcline and physically we would consider a patient described by these parameters as *disease free*, since the system would move towards the tumor-free state without any outside intervention.

1.3 The Effect of Varying the E-nullcline

The E-nullcline, i.e. the graph of the function $f(T)$, has different forms depending on the parameter values.

1.4 Intersections of the E and T Nullclines

For each of the T-nullclines, determine the number and stability of the equilibria. A Further Parameter Change Reduces the Possible Number of Equilibria. What is the maximum possible number of (positive) equilibria?

2 Determining Bifurcations Analytically

Conditions determining the shape of the graph of $f(T)$:

- The function $f(T)$ has horizontal asymptotes when the parameter, ρ, is:
$$\rho > \left(\sqrt{\eta\mu} + \sqrt{\delta}\right)^2 \text{ or } \rho < \left(\sqrt{\eta\mu} - \sqrt{\delta}\right)^2$$
- The function $f(T)$ has a maximum at a positive value of T when $\rho > \eta\mu$
- The condition for the asymptotes to be positive is:
$$\rho > \left(\sqrt{\eta\mu} + \sqrt{\delta}\right)^2$$

Rearranging this inequality gives the following conditions on the parameters for which asymptotes exist:

$$\rho > \left(\sqrt{\eta\mu} + \sqrt{\delta}\right)^2 \text{ or } \rho < \left(\sqrt{\eta\mu} - \sqrt{\delta}\right)^2$$

(2) $\rho > \eta\mu$. Details: Taking the derivative of $f(T)$ and setting it equal to zero gives :
$$\mu = \frac{\rho\eta}{\mu}$$
Solving for T gives:
$$T_{\max} = \sqrt{\frac{\rho\eta}{\mu}} - \eta.$$
Setting $T_{\max} > 0$ gives the result.

(3) $\rho > \left(\sqrt{\eta\mu} + \sqrt{\delta}\right)^2$ Details: In Question (1) we saw that asymptotes will exist only if ρ satisfies one of the two given inequalities. The values T_{Asympt} will be positive only if the expression: $\rho - \delta - \eta\mu > 0$, and hence ρ must satisfy the first of the two inequalities given in Answer (1).

2.1 Finding Equilibria Analytically

To find the equilibria of the system analytically, we can set $f(T) = g(T)$ to get a THIRD polynomial whose ROOTS are the EQUILIBRIA.

$$f(T) = \frac{\sigma}{\delta + \mu T - \frac{\rho T}{\eta + T}} = \alpha(1 - \beta T) = g(T)$$

$$\Longrightarrow A_3 T^3 + A_2 T^2 + A_1 T + A_0 = 0$$

where

$$A_3 = \beta\mu$$

$$A_2 = \beta(\mu\eta + \delta - \rho) - \mu$$

$$A_1 = \beta\delta\eta - \mu\eta - \delta + \rho + \frac{\sigma}{\alpha}$$

[

$$A_0 = \frac{\sigma}{\eta} - \delta\eta$$

2.2 Descartes Rule of Signs

Let $f(x) = a_n x^n + a_{n-1} x^{n-1} + \; + a_1 x + a_0$. The number of real positive roots of $f(x) = 0$ is less than or equal to the number of sign changes in the sequence $\{a_n, a_{n-1}, ..., a_1, a_0\}$.

The MATLAB code Bif3.m can be used to find the equilibria and their eigenvalues, and to plot them

```
%Bif3.m
% Script file for finding the equilibria for the system of ode's
% described in Kuznetsov's paper: nonlinear dynamics of
% immunogenic tumors, Bulletin of Mathematical Biology Vol.56, 1994.
%
% The equilibria are stored in the matrix Equil, one equilibrium per row.
% The eigenvalues are stored in the matrix Ev.
% The vector Stability stores the stability information:
```

```
% Stability(i) = 1 iff the ith equilibrium is stable,
% Stability(i) = -1 iff the ith equilibrium is unstable,
% Stability(i) = 0 if the stability of the ith equilibrium is undetermined
%
% Author: A.E.Radunskaya
% August, 2002
%
% The parameter values for the (non-dimensionalized) system are
% given in the vector "par", as noted below:
% par(1)=alpha;
% par(2)=beta;
% par(3)=delta;
% par(4)=eta;
% par(5)=mu;
% par(6)=rho;
% par(7)=sigma;
% The parameters from Kuznetsov's paper:
par=[1.636 .002 .3743 20.19 .00311 1.131 .1181];
Equil=KuzEquil(par);
[Ev,Stability]=EquilStability(Equil,par);
for i=1:length(Stability) % for each equilibrium
switch Stability(i)
case{1}, symbol='b*'; mcol='b';
case{-1}, symbol='ro'; mcol='r';
case{0}, symbol='g^'; mcol='g';
end;
plot(Equil(i,1),Equil(i,2),symbol,'MarkerFaceColor',mcol);
hold on
end;

%KuzEquil.m
% This function determines the equilibria of the system of two-dimensional
% equations given in the m-file ``Kuzode". It does so by finding
% the intersection of the two null-clines described in the m-files NullE and NullT,
% as well as the portion of the T-nullcline: {T=0}.

% This is accomplished by the subroutine ``NullIntersect"
%
% This is the system described in Kuznetsov's paper: nonlinear dynamics of
% immunogenic tumors, Bulletin of Mathematical Biology Vol.56, 1994
```

```
%
% Author: A.E.Radunskaya
% August, 2002
%
function Equil=KuzEquil(par)
alpha=par(1);
beta=par(2);
delta=par(3);
eta=par(4);
mu=par(5);
rho=par(6);
sigma=par(7);
% Initialize the array containing computed equilibria.
Equil=[];
% Loop through possible starting values for the routine fzero.
% If a new zero is found, store it in the array Equil.
options=optimset('TolX',.000001);
for startT=.1:10:500 % go one beyond the carrying capacity for the tumor cells
[z,fval]=fzero(@NullIntersect,startT,options,par);
ez=alpha*(1-beta*z);
if abs(fval) < .0001 % if fzero has been successful
if isempty(Equil)
Equil(1,2)=z;
Equil(1,1)=ez;
Lastz=z;
i=1;
elseif min(abs(ez-Equil(:,1))>.001) % if we haven't yet found this equilibrium, store
i=i+1;
Equil(i,2)=z;
Equil(i,1)=ez;
Lastz=z;
end; % if isempty or if not yet found
end; % if fval is small enough
end; % for loop
% Add the tumor-free equilibrium.
Equil(i+1,2)=0;
Equil(i+1,1)=sigma/delta;

%NullE.m
function E=NullE(T,par)
```

```
% This function determines the value of the effector cells along
% the null-cline: dE/dt = 0 for a given value of T, using the
% (non-dimensional) parameters in the vector par.
%
% This is equation (7) in Kuznetsov's paper: nonlinear dynamics of
% immunogenic tumors, Bulletin of Mathematical Biology Vol.56, 1994
alpha=par(1);
beta=par(2);
delta=par(3);
eta=par(4);
mu=par(5);
rho=par(6);
sigma=par(7);
E=sigma./(delta + mu*T - (rho*T./(eta + T)));

%NullT.m
function E=NullT(T,par)
% This function determines the value of the effector cells along
% the null-cline: dT/dt = 0 for a given value of T not equal to zero, using the
% (non-dimensional) parameters in the vector par.
%
% This is the equation for g(y) in Kuznetsov's paper: nonlinear dynamics of
% immunogenic tumors, Bulletin of Mathematical Biology Vol.56, 1994
alpha=par(1);
beta=par(2);
delta=par(3);
eta=par(4);
mu=par(5);
rho=par(6);
sigma=par(7);
E=alpha*(1 - beta*T);

%NullIntersect.m
function z=NullIntersect(T,par)
% Find the intersection of the two nullclines
% described by NullE and NullT
z=NullE(T,par)-NullT(T,par);

%EquilStability.m
% This function determines the stability of a set of equilibria
```

```
% by computing the Jacobian of the ODE and computing its eigenvalues.
% The eigenvalues are stored in the array, and the stability is stored
% in the vector Stability, with a '1' signifying stable, and a -1 signifying unstable
%
% Author:A.E.Radunskaya
% August, 2002
%
function [Ev, Stability]=EquilStability(Equil,par)
% Equil is an array, each row of which is an equilibrium of the system in question.
% The Jacobian is computed in the routine "KuzJac".
[r,c]=size(Equil); % the number of rows in Equil, r, is the number of equilibria.
for i=1:r % compute the stability of each equilibrium
J=KuzJac(Equil(i,:),par);
Ev(i,:)=eig(J)';
if max(real(Ev(i,:))) < 0 % if all eigenvalues are negative, the equilibrium is stabl
Stability(i) = 1;
elseif max(real(Ev(i,:))) > 0 % if there is at least one positive eigenvalue the equi
Stability(i)=-1;
else Stability(i) = 0; % otherwise, the stability is undetermined
end; % if eigenvalues are all negative
end; % for each equilibrium
```

Definition 2.1 *The stable manifold through a point, p in the state space of a system of differential equations is the set of initial values:*

$$W_s(p) = \{x_0 \mid \lim_{n\to\infty} x(t_n) = p\}$$

for some increasing infinite sequence of times, $\{t_1, t_2, ...\}$, $\lim_{n\to\infty} t_n = +\infty$, *and where* $x(t)$ *is a solution to the system of differential equations with* $x(0) = x_0$.

Definition 2.2 *The unstable manifold through p,* W_u, *is defined analagously, with the sequence of times decreasing to* $-\infty$*:*

$$W_u(p) = \{x_0 \mid \lim_{n\to\infty} x(t_n) = p\}$$

for some decreasing infinite sequence of negative times, $0 > t_1 > t_2 > ...$, $\lim_{n\to\infty} t_n = -\infty$, *with* $x(t)$ *a solution such that* $x(0) = x_0$.

The stable manifold through a saddle equilibrium in a planar system of ODEs is also called a separatrix, precisely because it separates the orbits into two sets. In our case, since there are two stable equilibria, these two sets of orbits actually belong to two different basins of attraction. In higher dimensions the situation can be quite a bit more complicated, but even in two dimensions we can observe some interesting phenomena.

Definition 2.3 *Let q be a stable equilibrium point of a system of differential equations, and let B be the set of initial values resulting in solutions which approach q:*

$$B = \{x_0 \mid x(0) = x_0, \text{ and } \lim_{t\to\infty} x(t) = q\}.$$

B is called the Basin of Attraction of q.

In our example, the clinical interpretation of the basin of attraction of the stable equilibrium, is THE SET OF STATES OF A PATIENTS SYSTEM WHICH, IN THE ABSENCE OF THERAPEUTIC INTERVENTION, WILL RESULT IN THE ESTABLISHMENT OF A RELATIVELY LARGE TUMOR.

A MATLAB file is included which draws the stable and unstable manifolds through the saddle equilibria for a given parameter set. MATLAB demo code: see Bif4.

```
%Bif4.m
Draw stable and unstable manifolds through the saddle equilibria
% in the 2-dimensional ODE system given in NonDimKuzOde
%
% Author: A.E.Radunskaya

% August 2, 2002
%
function Saddles=DrawManifolds(par)
% Parameters of the system are in the vector par as follows:
par=[1.636 .002 .3743 20.19 .00311 1.131 .1181];
% alpha=par(1);
% beta=par(2);
% delta=par(3);
% eta=par(4);
% mu=par(5);
% rho=par(6);
% sigma=par(7);
% This routines needs these M-files: KuzEquil, NullIntersect,
% NullE, NullT, EquilStability, KuzJac, NonDimKuzOde and OutOfRange
```

```
Equil=KuzEquil(par);
[Ev, Stability]=EquilStability(Equil,par);
I=find(Stability==-1);
Saddles=[Equil(I,1) Equil(I,2)];
[r,c]=size(Saddles);
options=odeset('Events',@OutOfRange);
for i=1:r
i
J=KuzJac(Saddles(i,:),par);
[V,L]=eig(J);
[v,ind]=min(diag(L)); % Find the stable eigenvector
y1=Saddles(i,:) + V(:,ind)'/norm(V(:,ind));
[val,term,dir]=OutOfRange(0,y1,[]);
if val~=0
[T1,Y1]=ode45(@NonDimKuzOde,[0 -20],y1,options,par); % Solve backwards for stable man
else
Y1=y1;
end;
y2=Saddles(i,:)-V(:,ind)'/norm(V(:,ind));
[val,term,dir]=OutOfRange(0,y2,[]);
if val~=0
[T2,Y2]=ode45(@NonDimKuzOde,[0,-20],y2,options,par);
else
Y2=y2;
end;
[v,j]=max(diag(L)); % Find the unstable eigenvector
y3=Saddles(i,:)+V(:,j)'/norm(V(:,j));
[val,term,dir]=OutOfRange(0,y3,[]);
if val~=0
[T3,Y3]=ode45(@NonDimKuzOde,[0,50],y3,options,par); % Solve forward for the unstable
else
Y3=y3;
end;
y4=Saddles(i,:)-V(:,j)'/norm(V(:,j));
[val,term,dir]=OutOfRange(0,y1,[]);
if val~=0
[T4,Y4]=ode45(@NonDimKuzOde,[0,50],y4,options,par);
else
Y4=y4;
end;
```

```
plot(Y1(:,1),Y1(:,2),'m',Y2(:,1),Y2(:,2),'m',...
Y3(:,1),Y3(:,2),'c',...
Y4(:,1),Y4(:,2),'c','LineWidth',1);
hold on
clear Y1 Y2 Y3 Y4
end
axis manual
axis([0 5 0 500]);
plot(Saddles(:,1),Saddles(:,2),'mo','MarkerFaceColor','m')
```

```
%OutOfRange.m
function [VALUE,ISTERMINAL,DIRECTION]=OutOfRange(T,Y,flag)
% Determines whether to stop the Ode Solver when the solution
% is out of the window given in WINDOW=[Xmin Xmax Ymin Ymax]
% This function is used as the Events function in the ODE options
%
WINDOW=[0 5 0 500];
ISOUT=min([Y(1)-WINDOW(1) WINDOW(2)-Y(1) Y(2)-WINDOW(3) WINDOW(4)-Y(2)]);
VALUE=ISOUT>=0;
ISTERMINAL=1;
DIRECTION=0;
```

```
%NonDimKuzOde
function xdot=NonDimKuzOde(t,x,par)
% The ODE for the non-dimensionalized system of differential equations in
% Kuznetsov's paper: nonlinear dynamics of
% immunogenic tumors, Bulletin of Mathematical Biology Vol.56, 1994
alpha=par(1);
beta=par(2);
delta=par(3);
eta=par(4)*ones(size(x(2,:)));
mu=par(5);
rho=par(6);
sigma=par(7)*ones(size(x(2,:)));
xdot=[sigma + (rho*x(1,:).*x(2,:))./(eta + x(2,:)) - mu*x(1,:).*x(2,:) - delta*x(1,:)
alpha * x(2,:) .*(ones(size(x(2,:))) - beta*x(2,:)) - x(1,:).*x(2,:)];
```

2.3 Bifurcation Diagram

The MATLAB code for generating the bifurcation diagram is given in Bif5.m.

```
% Bif5.m
% Script file for drawing a bifurcation diagram.
% In this case, we compute the value of the tumor population
% at the equilibria of the system given in the ODE file NonDimKuzOde
% as the parameter sigma, the immune source rate, is varied.
% The other parameter values are those determined to be ‘normal’.
%
% Unstable equilibria are plotted as dashed lines, and stable equilibria
% as solid lines.
%
% This file uses NonDimKuzOde, KuzEquil, EquilStability and KuzJac
% NullIntersect, NullE, and NullT are called by KuzEquil
%
% Author: A.E.Radunskaya
% August, 2002
%
par=[1.636 .002 .3743 20.19 .00311 1.131 .1181];
% par(1)=alpha;
% par(2)=beta;
% par(3)=delta;
% par(4)=eta;
% par(5)=mu;
% par(6)=rho;
% par(7)=sigma;
sigma=0:.005:1.2;
for i=1:length(sigma)
i;
par(7)=sigma(i);
Equil=KuzEquil(par);
[Ev, Stability]=EquilStability(Equil,par);
EQ{i}=Equil;
STAB{i}=Stability’;
end;
% Plotting
for i=1:length(sigma)
for j=1:length(STAB{i})
switch STAB{i}(j)
case{1}, symbol=’b.’;
case{-1}, symbol=’r.’;
 case{0}, symbol=’g’;
```

```
end;
plot(sigma(i),EQ{i}(j,2),symbol)
hold on
end;
end
```

2.3.1 Types of Bifurcations

Each bifurcation has a name. In this model we have three types:

- Transcritical when equilibria change STABILITY
- Saddle-node when equilibria APPEAR or DISAPPEAR
- Heteroclinic when the basins of attraction dramatically CHANGE SHAPE

A Heteroclinic Bifurcation occurs when boundaries of basins change, i.e. when the stable and unstable manifolds of saddle equilibria change their relative orientation.

Computing tip: If this is done in MATLAB, the function ginput is useful for selecting various initial conditions graphically. Also, the Events option in the MATLAB ODE solvers is helpful to avoid the solutions running off to infinity. The use of this option is illustrated in MATLAB demo code Bif4.m, using the Events function OutOfRange.m.

3 Prey predator populations

```
%prey_demo.m
% This file contains MATLAB code to guide you through problem set
% and more generally to teach you phase plane diagram analysis for
% differential equation.  Try changing the parameters and see how
% these changes effect the solutions.

clf;

global alpha beta;
more on; echo on;

% Consider the following system of first order differential equations
%
```

```
%  dy_1/dt = (1 - alpha y_2) y_1
%
%  dy_2/dt = (- 1 + beta y_1) y_2
%
%  where, variables y_1 and y_2 measure the sizes of the prey and
%  predator populations, respectively.  The quadratic cross terms
%  account for the interaction between the species.  This system is
%  known as the LOTKA-VOLTERRA predator-prey model.

% The first task when analyzing the system of differential equations
% is to plot both variables as a function of time.  For this one needs to
% specify an interval of time, initial conditions of the size of the prey
% and predator population and values for the parameters alpha and
% beta and then one needs to solve the system.

echo off;

T = input('What time interval do you want for the simulation? T=');
y_1_init = input('Specify an initial size of the predator population y_1_init=');
y_2_init = input('Specify an initial size of the prey population y_2_init=');
alpha = input('Pick a value for the parameter alpha alpha=');
beta = input('Pick a value for the parameter beta beta=');

echo on;

% To solve the differential equations we use RUNGE KUTTA method referred
% to as ode23. To read more about this code type help ode23 inside MATLAB.
% Function "prey" defines the differential equations, tspan is the period
% of simulation and y0 are the initial population values.  ode23 returns two
% arguments: t - different points in time between 0 and T; and
% y - value of the two populations at those times;

y0 = [y_1_init; y_2_init];  % Initial Values;
tspan = [0 T];              % Simulation Period;

[t y] = ode23('prey',tspan,y0);

% We can now plot the time paths of the two populations.

plot(t,y)
```

```
title('Lotke-Volterra Equation Time History');
xlabel('TIME');
ylabel('POPULATIONS');
legend(['y_1';'y_2']);

pause;

% The next task is to undertake the phase-plane analysis.  For this we first
% plot the isoclines.  The isoclines are obtained by setting the differential
% equations to 0.  Thus the isocline dy_1/dt = 0 is given by y_2 = 1/alpha
% while the isocline dy_2/dt = 0 is given by y_1 = 1/beta;

clf;

% Plot of the dy_1/dt = 0 isocline;

plot([0; round(max(y(:,1)))], [1/alpha;1/alpha]);

hold on;

% Plot of the dy_2/dt = 0 isocline;

plot([1/beta;1/beta],[0;round(max(y(:,2)))]);

xlabel('y_1');
ylabel('y_2');

% Next we determine the direction of the trajectory in each isosector
% (quadrant formed by the isoclines). For this we pick a point in each
% quadrant and determine the sign the differential equation at that point.  A
% negative value implies that the variable is decreasing while a positive
% value implies that the variable is increasing in that isosector.

% For the north-east isosector:

echo off;

p_1 = input('Pick a point on the horizontal axis in the isosector p_1=');
p_2 = input('Pick a point on the vertical axis in the isosector p_2=');
```

```
echo on;

   % The directions of the variables are:

pp = prey(0,[p_1;p_2])

% For the south-east isosector:

echo off;

p_1 = input('Pick a point on the horizontal axis in the isosector p_1=');
p_2 = input('Pick a point on the vertical axis in the isosector p_2=');

echo on;

   % The directions of the variables are:

pp = prey(0,[p_1;p_2])

% For the south-west isosector:

echo off;

p_1 = input('Pick a point on the horizontal axis in the isosector p_1=');
p_2 = input('Pick a point on the vertical axis in the isosector p_2=');

echo on;

   % The direction of the variables are:

pp = prey(0,[p_1;p_2])

% For the north-west isosector:

echo off;

p_1 = input('Pick a point on the horizontal axis in the isosector p_1=');
p_2 = input('Pick a point on the vertical axis in the isosector p_2=');
```

```
echo on;

   % The directions of the variables are:

pp = prey(0,[p_1;p_2])

% Finally, we can plot the system in phase space (y_1 and y_2);

plot(y(:,1),y(:,2));
title('Lotke-Volterra Equation - phase-plane plot');

% The graph of the phase plane (two dimensional) trajectories for both
% predator and prey depicts a closed orbit around the equilibrium point
% obtained by setting both differential equations equal to zero. The isoclines
% intersect at the equilibrium. By changing the parameter values and initial
% population size, it is possible to see different orbits.

%prey.m
function y_diff = prey(t,y)

global alpha beta;

y_1_init = y(1);
y_2_init = y(2);

y_1dot=y_1_init - alpha*y_2_init*y_1_init;
y_2dot=-y_2_init + beta*y_1_init*y_2_init;

y_diff=[y_1dot y_2dot]';
```

References

[1] Vladmir A. Kuznetsov, Iliya A. Makalkin, Mark A. Taylor, and Alan S. Perelson. Nonlinear dynamics of immunogenic tumors: Parameter estimation and global bifurcation analysis. Bulletin of Mathematical Biology, 56(2), 1994, 295–321.