

Chemostat_2nutrient_dynamics.m

```
function [ dSdt ] = chemostat_2nutrient_dynamics(t,S,params)
% this function defines the chemostat differential equation
%
% get the individual parameters from the vector
%
q    = params(1);
u    = params(2);
r    = params(3);
K    = params(4);
ep   = params(5);
L    = params(6);
F    = params(7);
v    = params(8);
%
% initialize the state's derivative
%
dSdt = zeros(size(S));
%
% put things in terms of individual variables
%
x = S(1); % cells
y = S(2); % glucose
z = S(3); % ammonium
%
% form the differential equation formulae
%
rxyz = r*x*y*z/((K+y)*(z+L));
dxdt  = (rxyz) - q*x;
dydt  = q*(u - y) - ep*(rxyz);
dzdt  = -F*(rxyz) + q*(v - z);
%
% put things back in terms of state vector variable
%
dSdt(1) = dxdt;
dSdt(2) = dydt;
dSdt(3) = dzdt;
end
```

Chemostat_2nutrients_script_C_lim_30

```
clear all

t0 = 0;      %initial time
t1 = 100;    % final time
x0 = 10;     % initial cell population
y0 = 10;     % initial glucose concentration
z0 = 10;     % initial ammonium concentration
%
% initial state for use in DE solver.
%
S0 = [x0;y0;z0];

%
```

```

% rate constants and problem parameters
%

ep = 14.2857;
u = 25;
q = 0.03;
v = 5;

% not sure which values to use for these parameters
yInf = 0.054;
zInf = 1.11;
xInf = 1.737;
K = 30;
L = 2;
r = q*(yInf+K)*(zInf+L)/(yInf*zInf);
F = (q*zInf-q*v)/(xInf*q);
%
% rate constants into parameter vector
%
params(1) = q;
params(2) = u;
params(3) = r;
params(4) = K;
params(5) = ep;
params(6) = L;
params(7) = F;
params(8) = v;
%
% define function for fixed parameter values
%
chemostat_2nutrient_dynamics_params =
@(t,S)chemostat_2nutrient_dynamics(t,S,params);

%
% call the matlab function to solve the DE for us.
%
[t,St] = ode45(chemostat_2nutrient_dynamics_params,[t0 t1],S0);
%
% Plot the results.
%
plot(t,St,'LineWidth',3)

legend('cells','glucose','ammonium')
xlabel('time','hrs')
ylabel('states','g')

```

Chemostat_2nutrient_script_N_lim_30

```

clear all

t0 = 0;      %initial time
t1 = 100;    % final time
x0 = 10;     % initial cell population
y0 = 10;     % initial glucose concentration

```

```

z0 = 10;      % initial ammonium concentration
%
% initial state for use in DE solver.
%
S0 = [x0;y0;z0];

%
% rate constants and problem parameters
%

ep = 25;
u = 46;
q = 0.03;
v = 0.65;

% not sure which values to use for these parameters
yInf = 15.33;
zInf = 0.0034;
xInf = 1.227;
K = 30;
L = 2;
r = q*(yInf+K)*(zInf+L)/(yInf*zInf);
F = (q*zInf-q*v)/(xInf*q);
%
% rate constants into parameter vector
%
params(1) = q;
params(2) = u;
params(3) = r;
params(4) = K;
params(5) = ep;
params(6) = L;
params(7) = F;
params(8) = v;
%
% define function for fixed parameter values
%
chemostat_2nutrient_dynamics_params =
@(t,S)chemostat_2nutrient_dynamics(t,S,params);

%
% call the matlab function to solve the DE for us.
%
[t,St] = ode45(chemostat_2nutrient_dynamics_params,[t0 t1],S0);
%
% Plot the results.
%
plot(t,St,'LineWidth',3)

legend('cells','glucose','ammonium')
xlabel('time','hrs')
ylabel('states','g')

```