Chemostat\_2nutrient\_script

clear all

t0 = 0; %initial time

t1 = 100; % final time

x0 = 10; % initial cell population

y0 = 12; % initial yeast concentration

z0 = 14; % initial nutrient concentration

%

% initial state for use in DE solver.

%

S0 = [x0;y0;z0];

%

% rate constants and problem parameters

%

r = 2;

K = 10;

ep = 1.1;

u = 6;

q = 0.03;

v = 12;

fp = 1.5;

L = 15;

%

% rate constants into parameter vector

%

params(1) = q;

params(2) = u;

params(3) = r;

params(4) = K;

params(5) = ep;

params(6) = v;

params(7) = fp;

params(8) = L;

%

% 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','food')

xlabel('time')

ylabel('states')

Chemostat\_2nutrient\_dynamics

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);

v = params(6);

fp = params(7);

L = 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

%

ryz = r\*(y/(K+y))\*(z/(L+z));

dxdt = (ryz)\*x - q\*x;

dydt = q\*(u - y) - ep\*ryz\*x;

dzdt = q\*(v - z) - fp\*ryz\*x;

%

% put things back in terms of state vector variable

%

dSdt(1) = dxdt;

dSdt(2) = dydt;

dSdt(3) = dzdt;

end

A close up of a map

Description automatically generated