Question2

Shuchen Ye

Content

8.1

Using the law of mass action, we have:

$$vE_1=k_1*E*S$$

$$vE_2=k_2*ES$$

$$vE_3=k_3*ES$$

$$vS_1=k_1*E*S$$

$$vS_2=k_2*ES$$

$$vES_1=k_1*E*S$$

$$vES_2=k_2*ES$$

$$vES_3=k_3*ES$$

$$vP=k_3*ES$$

We can see that for substance E, it is involved in three processes, so the rate can be divided into vE_1, vE_2, and vE_3. For substance S, it participates in two processes, so the rate can be divided into vS_1 and vS_2. For substance ES, it is involved in three processes, so the rate can be divided into vES_1, vES_2, and vES_3. The substance P only participates in one reaction process, so there is only one rate vP.

Combine them into four equations for the four species:

$$vE= -vE_1+ vE_2+ vE_3=-k_1*E *S+k_2*ES+k_3*ES$$

$$vS=- vS_1+vS_2=- k_1*E *S+ k_2*ES$$

$$vES= vES_1- vES_2- vES_3= k_1*E *S- k_2*ES- k_3*ES$$

$$vP= vP =k_3*ES$$

8.2

Since the initial concentration of E is 1 μM, the initial concentration of S is 10 μM, and the initial concentrations of ES and P are both 0. The rate constants are k_1=100/μM/min, k_2=600/min, and k_3=150/min.

We use the fourth-order Runge-Kutta method to write the Matlab code:

Matlab Code

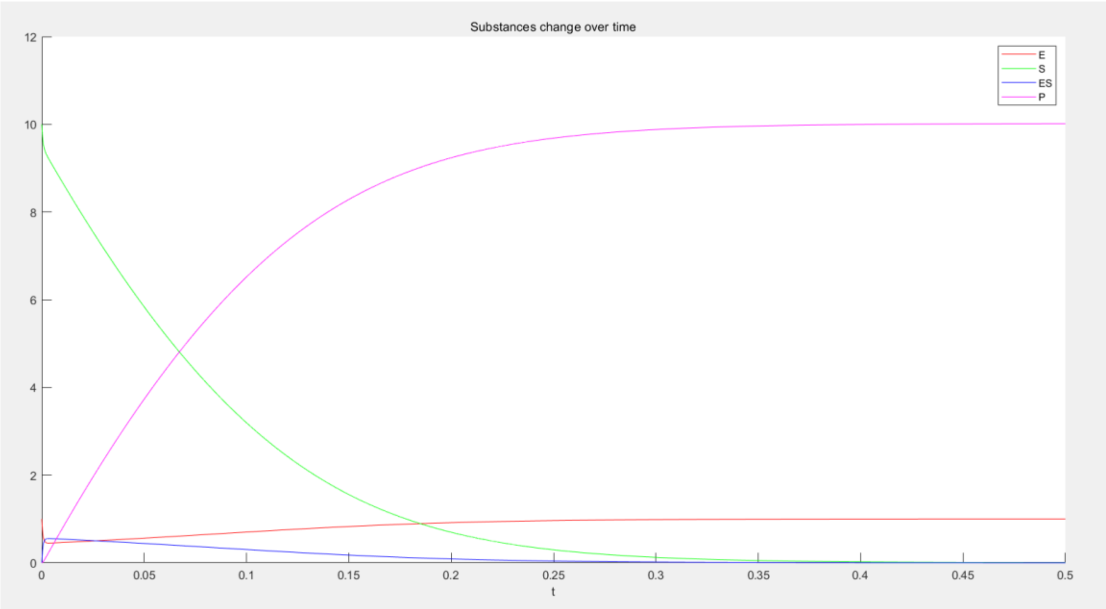
```
clear;
clc;
close all;

h=1e-5; %Step length
t=0:h:0.5; %generate a vector of the independent variable t

%% Create an array of calculation results x, y, z
N=length(t);
E=ones(1,N);
S=10*ones(1,N);
ES=zeros(1,N);
P=zeros(1,N);

%% Fourth-order Runge-Kutta iteration
for i=2:N
    t_n=t(i-1);
    E_n=E(i-1);
    S_n=S(i-1);
    ES_n=ES(i-1);
```

```
    P_n=P(i-1);

    kE1_1=100*E_n*S_n;
    kE1_2=600*ES_n;
    kE1_3=150*ES_n;
    kS1_1=100*E_n*S_n;
    kS1_2=600*ES_n;
    kES1_1=100*E_n*S_n;
    kES1_2=600*ES_n;
    kES1_3=150*ES_n;
    kP1=150*ES_n;

    kE2_1=100*(E_n+kE1_1*h/2)*(S_n+kS1_1*h/2);
    kE2_2=600*(ES_n+kES1_2*h/2);
    kE2_3=150*(ES_n+kES1_3*h/2);
    kS2_1=100*(E_n+kE1_1*h/2)*(S_n+kS1_1*h/2);
    kS2_2=600*(ES_n+kES1_2*h/2);
    kES2_1=100*(E_n+kE1_1*h/2)*(S_n+kS1_1*h/2);
    kES2_2=600*(ES_n+kES1_2*h/2);
    kES2_3=150*(ES_n+kES1_3*h/2);
    kP2=150*(ES_n+kES1_2*h/2);

    kE3_1=100*(E_n+kE2_1*h/2)*(S_n+kS2_1*h/2);
    kE3_2=600*(ES_n+kES2_2*h/2);
    kE3_3=150*(ES_n+kES2_3*h/2);
    kS3_1=100*(E_n+kE2_1*h/2)*(S_n+kS2_1*h/2);
    kS3_2=600*(ES_n+kES2_2*h/2);
    kES3_1=100*(E_n+kE2_1*h/2)*(S_n+kS2_1*h/2);
    kES3_2=600*(ES_n+kES2_2*h/2);
    kES3_3=150*(ES_n+kES2_3*h/2);
    kP3=150*(ES_n+kES2_2*h/2);

    kE4_1=100*(E_n+kE3_1*h/2)*(S_n+kS3_1*h/2);
    kE4_2=600*(ES_n+kES3_2*h/2);
    kE4_3=150*(ES_n+kES3_3*h/2);
    kS4_1=100*(E_n+kE3_1*h/2)*(S_n+kS3_1*h/2);
    kS4_2=600*(ES_n+kES3_2*h/2);
    kES4_1=100*(E_n+kE3_1*h/2)*(S_n+kS3_1*h/2);
    kES4_2=600*(ES_n+kES3_2*h/2);
    kES4_3=150*(ES_n+kES3_3*h/2);
    kP4=150*(ES_n+kES3_2*h/2);

    E(i)=E_n-
h/6*(kE1_1+2*kE2_1+2*kE3_1+kE4_1)+h/6*(kE1_2+2*kE2_2+2*kE3_2+k
```

```
E4_2)+h/6*(kE1_3+2*kE2_3+2*kE3_3+kE4_3);
    S(i)=S_n-
h/6*(kS1_1+2*kS2_1+2*kS3_1+kS4_1)+h/6*(kS1_2+2*kS2_2+2*kS3_2+k
S4_2);
    ES(i)=ES_n+h/6*(kES1_1+2*kES2_1+2*kES3_1+kES4_1)-
h/6*(kES1_2+2*kES2_2+2*kES3_2+kES4_2)-
h/6*(kES1_3+2*kES2_3+2*kES3_3+kES4_3);
    P(i)=P_n+h/6*(kP1+2*kP2+2*kP3+kP4);
end
%% Drawing
figure();
hold on;
plot(t,E,'r');
plot(t,S,'g');
plot(t,ES,'b');
plot(t,P,'m');
legend('E','S','ES','P');
xlabel('t');
title('Substances change over time');
hold off;
```

Result

The result is as follows:



We can see that enzyme E gradually increases after a brief decrease, and finally stabilizes at the original 1 μM. The intermediate ES gradually decreased after a brief increase and approached zero. Substance S also gradually decreases with the occurrence of chemical reaction, and substance P is generated.

After about 0.35 min, substance P was about 10 μM, while substance S approached 0, enzyme E remained at about 1, and ES was basically 0.

8.3

Since the interval of t is fixed, to find the time point with the fastest P generation rate, we only need to find the position where the slope of the P generation curve is the largest. So we add the following code after Fourth-order Runge-Kutta iteration:

Matlab Code

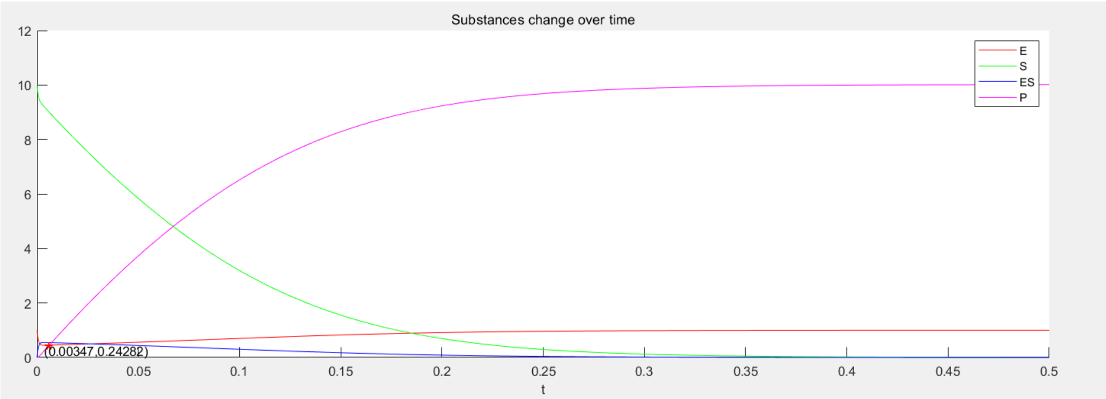
```
for i=2:N
    T(i-1)=P(i)-P(i-1);
end
a=find(T==max(T))-1;
```

And add the following code to the drawing section:

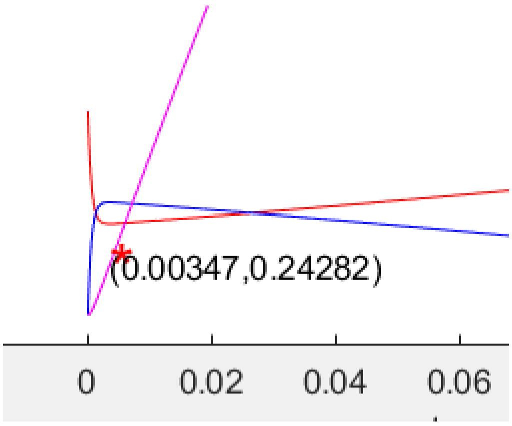
```
text(t(a),P(a),'*','color','r','FontSize',20);
text(t(a),P(a),['(',num2str(t(a)),',',num2str(P(a)),'
)']);
```

Then we can get this picture. If you look carefully, the point marked in the lower left corner is the point when Vm is the maximum value.

Result



Partial magnification is as follows:



The maximum Vm can be obtained by (P(a+1)-P(a))/(t(a+1)-t(a)):

Max(Vm)=82.9357

```
ans =

    82.9357
```

For the complete code, please see Q2.m.