

BENG221

Insulin-Mediated Blood Glucose Regulation

Jun Wang, Yiqian Wu and Yang Zhang

11/14/2014

Contents

1. Introduction.....	3
1.1 Background.....	3
1.2 Problem statement.....	5
2. Modeling strategy and assumptions.....	6
2.1 Diffusion module	6
2.2 PI control module.....	6
3. Analytical Solution	8
3.1 Particular solution.....	9
3.2 Homogeneous solution.....	9
3.3 Flux and Amount	11
4. The Proportion Integration Control System.....	14
4.1 System Principle Diagram	14
4.2 Transfer Function Analysis.....	14
4.3 The stability analysis.....	15
4.4 The Error Analysis.....	16
5. Discussion and Conclusion.....	19
5.1 Results.....	19
5.2 Limitations and future work.....	19
6. Reference	20
7. Appendix.....	21
Code of diffusion module	21
Code of control module	23

1. Introduction

1.1 Background

In humans, blood glucose regulation involves both fluid and central nervous system¹. However, fluid regulation is much dominant in blood glucose regulation by employing two hormones insulin and glucagon. Insulin and glucagon have antagonistic effects in glucose regulation that insulin secretion is triggered by high blood glucose level while glucagon the opposite.

After a meal, the food will get into the stomach and then into the small intestine where it is fully digested and then absorbed. In the small intestine wall, there is a layer of epithelial cell between the intestine lumen and interstitial space. The glucose will transport from intestine lumen to the epithelial cell, then from the epithelial cell to the interstitial space and eventually to the blood² (shown in figure 1).

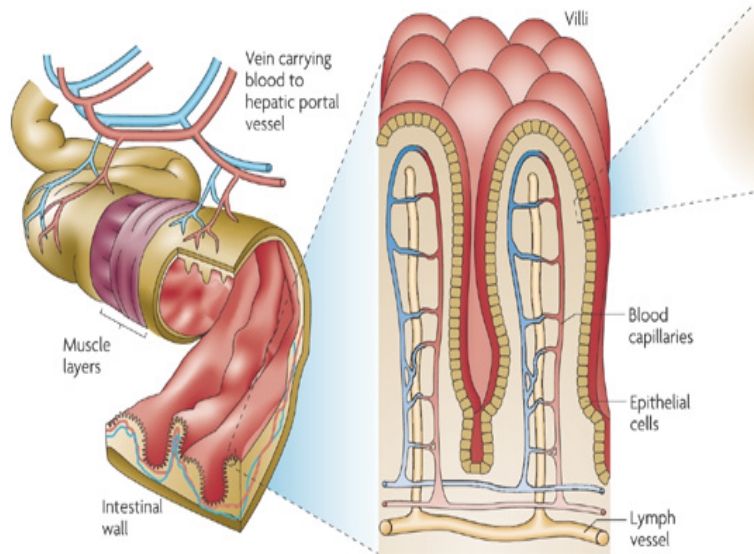


Fig. 1 Structure of small intestine wall

Figure1 from: http://csls-text2.c.u-tokyo.ac.jp/inactive/08_02.html³

Interestingly, the transport of glucose from intestine lumen to the blood experiences two stages. In the apical member of epithelial cell which faces the lumen, glucose is absorbed

Insulin-Mediated Blood Glucose Regulation

by active transportation, while it is transported out to the interstitial space from the basolateral membrane by facilitated diffusion (shown in figure 2). Therefore, the glucose in the epithelial cell is high, and the concentration in the interstitial space and blood reaches similar level with only some drag of time^{4,5}.

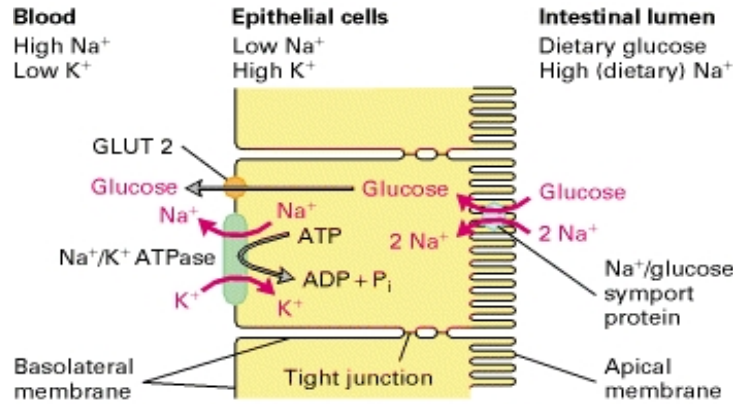


Fig. 2 Glucose transportation from intestine lumen to blood
Figure 2 from:

<http://www.ncbi.nlm.nih.gov/books/NBK21502/figure/A4123/?report=objectonly>⁶

Increase of glucose level in the blood whose normal glucose level maintains between (REF) will be sensed by the beta cell in the pancreas and trigger beta cell to secrete insulin⁷. Insulin will then stimulate the tissue cell to take up and utilize glucose. This in turn will lower the glucose level in the blood (shown in figure3). This process continues until the blood glucose gets back to the normal range. In healthy humans, blood glucose level will increase and reach a peak about 30 minutes after taking a meal, and lower to less than 200mg/dL after one hour, and back to normal range-less than 120mg/dL after 2 hours⁸.

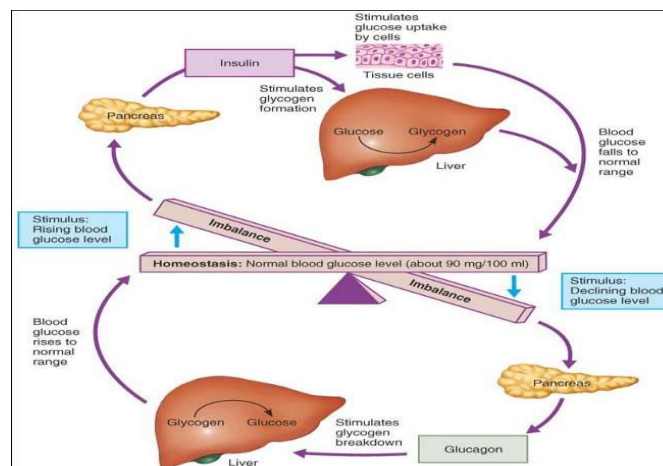


Fig. 3 Blood glucose regulation by insulin and glucagon
Figure from: <https://bohone09.wikispaces.com/Group+7>⁹

1.2 Problem statement

In this study, we model both the diffusion and control process happening in blood glucose regulation. In the diffusion module, we model glucose diffusion from epithelial cells across the interstitial space into the blood vessel. We also calculate the time needed for the blood glucose to reach a certain level of increase. We then model the control process of blood glucose regulation by insulin, and estimate the time required for the blood glucose level to be back to the normal condition.

2. Modeling strategy and assumptions

2.1 Diffusion module

In the diffusion process, we assume that the concentration of glucose in epithelia maintains at 230mg/dL. Because of this high concentration, glucose will diffuse from the epithelia to the interstitial space and then to the blood stream until the blood glucose level reaches 210mg/dL (an increase of 90mg/dL). We assume that this amount of increase is the threshold at which the insulin secretion will be triggered.

The amount of glucose diffused to the blood within this period = increased blood glucose level * blood volume = the time integral of flux at the interface * interface area A .

The diffusion model is shown in Fig.4.

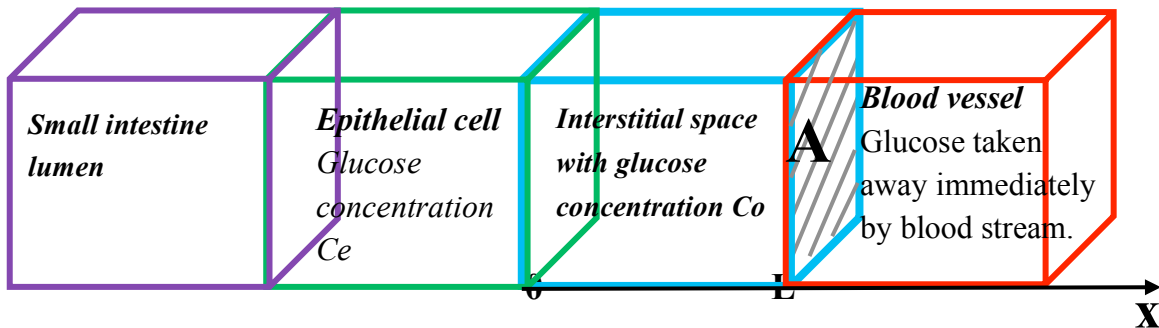


Fig.4. Diffusion model

2.2 PI control module

In the control process, we regard the increased blood glucose level as in the input and the change of blood glucose level as the output. We then calculate the time needed for the blood glucose level back to the normal range, which is the stage where the out put equals the input (shown in Fig.5).

When the blood glucose level reaches the threshold value $\Delta C_{\text{glu-input}}$, the speed of insulin

Insulin-Mediated Blood Glucose Regulation

secretion is proportional to that increase with coefficient K_1 . The decrease of blood glucose level is proportional to the accumulation of insulin, which is the integration of the speed of insulin secretion, with coefficient K_2 .

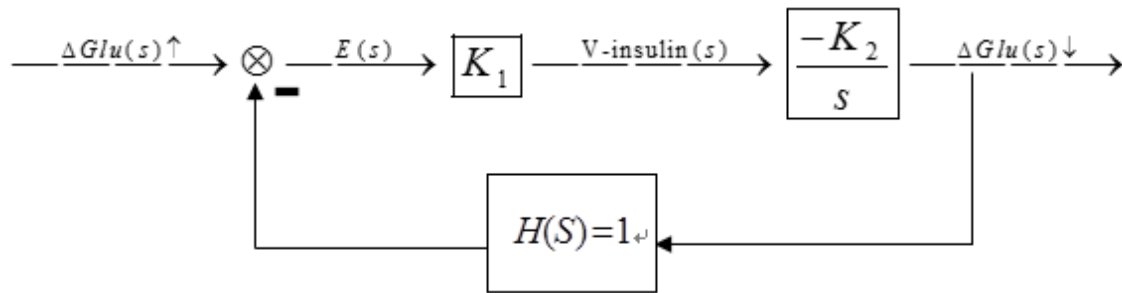


Fig 5. Control model

3. Analytical Solution

The governing equation of the diffusion process described above is the diffusion equation:

$$\frac{\partial u}{\partial t} = D \frac{\partial^2 u}{\partial x^2} \quad (1)$$

According to our model scenario, we make some simplifications. Basically, we assume that the glucose concentration within the epithelia remains constant, and the glucose diffusing into the blood vessel is taken away immediately by the blood. In other words, initial glucose concentration inside the interstitial space is C_o (I.C.), the glucose concentration at the interface between epithelia and interstitial space is C_e (B.C. at $x = 0$), and the glucose concentration at the interface between interstitial space and blood is C_L (B.C. at $x = L$). They are expressed as follows:

$$u(x, 0) = C_o \quad (2)$$

$$u(0, t) = C_e \quad (3)$$

$$u(L, t) = C_L \quad (4)$$

The values of these constants are listed in Table.1.

Constant	Value ^{7,8,10,11}
Diffusion coefficient D	2e-4 m ² /s
Diffusion length L	0.001 m
Initial concentration C_o	0.0009 kg/L
Left boundary concentration C_e	0.0023 kg/L
Right boundary concentration C_L	0.0009 kg/L
Interface area A	0.1 m ²
Glucose amount threshold S	0.006 kg

Table.1. Model constants.

Since the diffusion equation has inhomogeneous boundary conditions, we solve using the “poison tooth extraction” method. We first find the particular solution, then the homogeneous solution, and combine the two to get the full solution.

3.1 Particular solution

For steady state, the concentration no longer changes with time. Therefore:

$$\frac{\partial u_p(x,t)}{\partial t} = 0 \quad (5)$$

Hence, we can write $u_p(x, t)$ as $u_p(x)$ and rewrite (1) as:

$$D \frac{\partial^2 u_p(x)}{\partial x^2} = 0 \quad (6)$$

Solve for (6) we get:

$$u_p(x) = ax + b \quad (7)$$

Apply the boundary conditions to obtain a and b:

$$u_p(0, t) = b \quad \Rightarrow \quad b = C_e \quad (8)$$

$$u_p(L, t) = aL + b = C_L \quad \Rightarrow \quad a = \frac{C_L - C_e}{L} \quad (9)$$

Therefore, the particular solution $u_p(x)$ is:

$$u_p(x) = \frac{C_L - C_e}{L} x + C_e$$

3.2 Homogeneous solution

The homogeneous solution $u_H(x, t) = u(x, t) - u_p(x, t)$ satisfies the diffusion equation with homogeneous boundary conditions:

$$\frac{\partial u_H}{\partial t} = D \frac{\partial^2 u_H}{\partial x^2} \quad (10)$$

$$B.C. \begin{cases} u_H(0, t) = 0 \\ u_H(L, t) = 0 \end{cases} \quad (11)$$

Solve (10) by separation of variables:

$$u_H(x, t) = G(t)\phi(x) \quad (12)$$

$$\phi(x) \frac{dG(t)}{dt} = D \frac{d^2\phi(x)}{dx^2} G(t) \quad (13)$$

$$\frac{1}{D} \frac{dG(t)}{G(t)} = \frac{d^2\phi(x)}{\phi(x)} = -\lambda \quad (14)$$

For the time-dependent part, we have:

$$\frac{dG(t)}{dt} = -\lambda DG(t) \quad \Rightarrow \quad G(t) = G_0 e^{-\lambda Dt} \quad (15)$$

For the space-dependent part, there are three cases.

- i. For $\lambda = 0$, the solution is $\phi(x) = ax + b$ (16)

Apply the boundary conditions, we obtain $a = b = 0$, which is trivial.

ii. For $\lambda < 0$, the solution is $\phi(x) = ae^{\sqrt{-\lambda}x} + be^{-\sqrt{-\lambda}x}$ (17)

With boundary conditions, we still get the trivial solution $a = b = 0$.

iii. For $\lambda > 0$, the solution is $\phi(x) = a\cos(\sqrt{\lambda}x) + b\sin(\sqrt{\lambda}x)$ (18)

The boundary conditions yield:

$$\begin{cases} \phi(0) = a = 0 \\ \phi(L) = a\cos(\sqrt{\lambda}L) + b\sin(\sqrt{\lambda}L) = 0 \end{cases} \Rightarrow \begin{cases} a = 0 \\ \sin(\sqrt{\lambda}L) = 0 \end{cases} \quad (19)$$

Therefore, $\sqrt{\lambda} = \frac{n\pi}{L}, n = 1, 2, 3 \dots$ (20)

Since none of the individual n terms satisfies the initial condition, we use superposition and write the homogeneous solution as:

$$u_H(x, t) = \sum_{n=1}^{\infty} b_n \sin\left(\frac{n\pi}{L}x\right) e^{-D\left(\frac{n\pi}{L}\right)^2 t} \quad (21)$$

The full solution is therefore the summation of the homogeneous and particular solutions:

$$u(x, t) = u_H(x, t) + u_p = \sum_{n=1}^{\infty} b_n \sin\left(\frac{n\pi}{L}x\right) e^{-D\left(\frac{n\pi}{L}\right)^2 t} + \frac{C_L - C_e}{L}x + C_e \quad (22)$$

Apply initial conditions, we get:

$$u(x, 0) = u_p(x) + \sum_{n=1}^{\infty} b_n \sin\left(\frac{n\pi}{L}x\right) \quad (23)$$

Since $\sin(n\pi x/L)$ and $\sin(m\pi x/L)$ are orthogonal for all $n \neq m$, and

$$\int_0^L \sin\left(\frac{m\pi}{L}x\right) \sin\left(\frac{n\pi}{L}x\right) dx = \frac{L}{2} \text{ for } n = m, \quad (24)$$

we can compute b_n as follows:

$$b_n = \frac{2}{L} \int_0^L [u(x, 0) - u_p(x)] \sin\left(\frac{n\pi}{L}x\right) dx, \text{ where } \begin{cases} u_p(x) = \frac{C_L - C_e}{L}x + C_e \\ u(x, 0) = C_o \end{cases} \quad (25)$$

Perform integration by parts, we have:

$$b_n = \frac{2}{L} \int_0^L \left[C_o - \left(\frac{C_L - C_e}{L}x + C_e \right) \right] \sin\left(\frac{n\pi}{L}x\right) dx = \frac{2(C_L - C_o)}{n\pi} \cos(n\pi) - \frac{2(C_e - C_o)}{n\pi} \quad (26)$$

Therefore, the full solution of the model diffusion equation is:

$$u(x, t) = \sum_{n=1}^{\infty} \left[\frac{2(C_L - C_o)}{n\pi} \cos(n\pi) - \frac{2(C_e - C_o)}{n\pi} \right] \sin\left(\frac{n\pi}{L}x\right) e^{-D\left(\frac{n\pi}{L}\right)^2 t} + \frac{C_L - C_e}{L}x + C_e \quad (27)$$

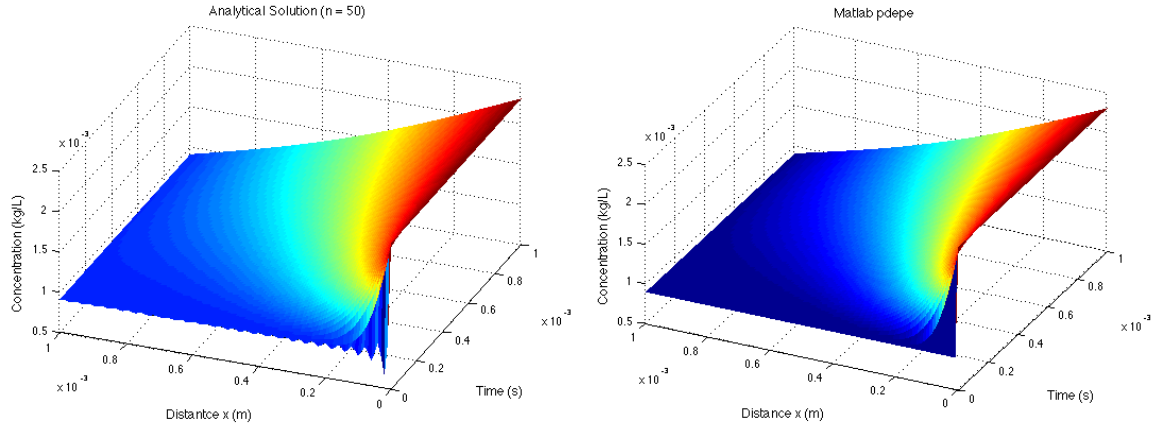


Fig.6. The glucose concentration in the interstitial space. Left: analytical solution; $n = 50$. Right: Matlab[®] pdepe function.

Fig.6 shows the change of glucose concentration in the interstitial space with time and space. The left figure is the plotted analytical solution with the first 50 terms. The right figure is plotted using the Matlab[®] built-in function pdepe. The two methods give out similar results, except for the Gibbs phenomenon (oscillations) in the left figure. One can see from Fig.6 that as time goes by, the glucose concentration tends to a linear distribution within the interstitial space.

3.3 Flux and Amount

As mentioned before, we assume that when the amount of glucose that has entered the blood stream reaches 6g, i.e., when the blood glucose level increases by 120mg/dL (5L blood/person), the insulin-mediated blood glucose regulation will be triggered (the third part of this report). For now, we want to calculate the amount of glucose entering the blood flow by investigating the flux. Flux is defined as:

$$J(x, t) = -D \frac{\partial u(x, t)}{\partial x} \quad (28)$$

Substitute equation (27) into (28), we have:

$$J(x, t) = -D \left\{ \frac{C_L - C_e}{L} + \sum_{n=1}^{\infty} \left[\frac{2(C_L - C_o)}{L} \cos(n\pi) - \frac{2(C_e - C_o)}{L} \right] e^{-D \left(\frac{n\pi}{L}\right)^2 t} \cos\left(\frac{n\pi}{L} x\right) \right\} \quad (29)$$

Fig.7 is plotted using the first 50 terms in the analytical solution of flux and shows the change of flux over time and space. It can be seen from Fig.7 that the flux is very large at the beginning of diffusion, but tends to a constant as time goes by.

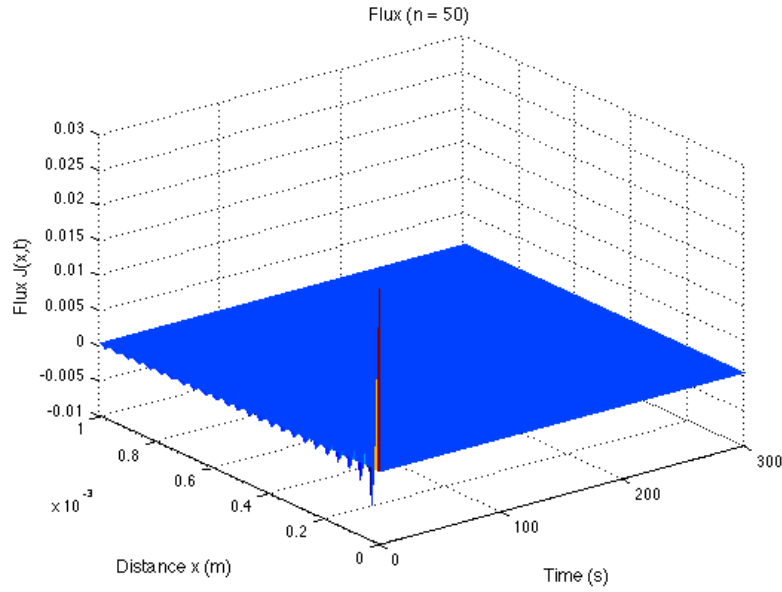


Fig.7. The change of flux over time and space.

Since the glucose enters blood via the interface between the interstitial space and the blood vessel ($x = L$), we only need to investigate the flux at $x = L$:

$$J(L, t) = -D \left\{ \frac{C_L - C_e}{L} + \sum_{n=1}^{\infty} \left[\frac{2(C_L - C_o)}{L} \cos(n\pi) - \frac{2(C_e - C_o)}{L} \right] e^{-D \left(\frac{n\pi}{L}\right)^2 t} \cos(n\pi) \right\} \quad (30)$$

The amount of glucose S that has entered the blood stream at time t_o can therefore be expressed as the integral of $J(L, t)$ over time period 0 to t_o multiplies the interface area A :

$$\begin{aligned} S(t_o) &= A \int_0^{t_o} J(L, t) dt \\ &= A \left\{ \frac{D(C_e - C_L)}{L} t_o + 2L \sum_{n=1}^{\infty} \left[\frac{(C_L - C_o)}{(n\pi)^2} \cos(n\pi) - \frac{(C_e - C_o)}{(n\pi)^2} \right] \cos(n\pi) \left[e^{-D \left(\frac{n\pi}{L}\right)^2 t_o} - 1 \right] \right\}, \quad (31) \end{aligned}$$

where A is the area of the interstitial space – blood vessel interface through which the glucose diffuses into the blood.

Insulin-Mediated Blood Glucose Regulation

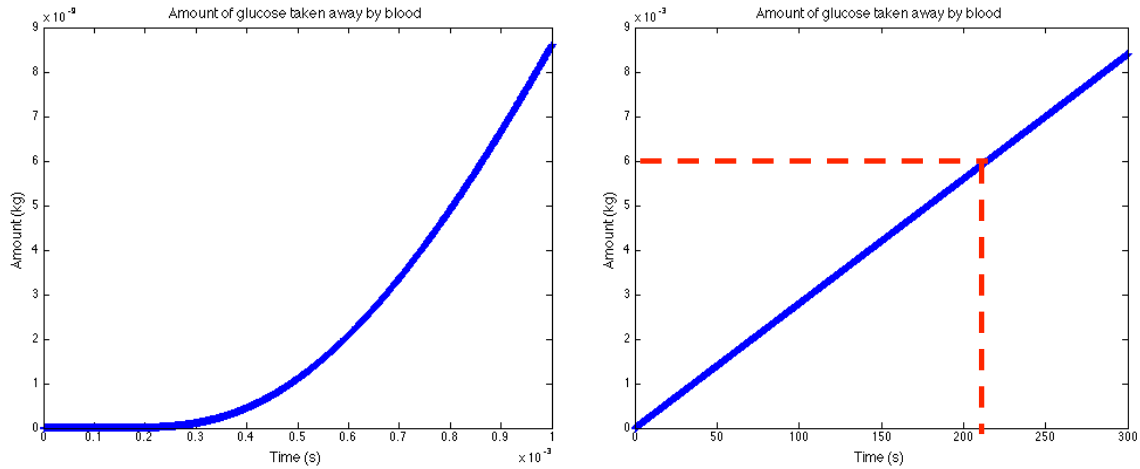


Fig.8. The amount of glucose taken away by the blood. Left: at the beginning of diffusion ($t < 0.001$ s). Right: steady state ($t \rightarrow \infty$).

We can see from Fig.8 that the amount of glucose entering the blood (the amount taken away by blood) increases nonlinearly at the beginning of the diffusion (the left figure). However, when time increases and the process approaches steady state, the amount of glucose that has entered the blood increases linearly (with a constant rate; the right figure).

As stated before, we assume that when this amount reaches a threshold of 6×10^{-3} kg (i.e., blood glucose level raised by 120 mg/dL), the insulin secretion will start. One can see from the right figure that when $\text{Amount} = 6 \times 10^{-3}$ kg, the corresponding time approximately equals 210 s. That is to say, the diffusion process takes $t_0 = 210$ s to increase the blood glucose level to the threshold for triggering the insulin-mediated glucose regulation.

4. The Proportion Integration Control System

4.1 System Principle Diagram

As we know, a negative feedback system exists in our body which adjusts and maintains the level of blood glucose. When the amount of blood glucose increases, then our body secretes more insulin which finally reduces the amount of blood glucose. The processes are shown in figure 8.

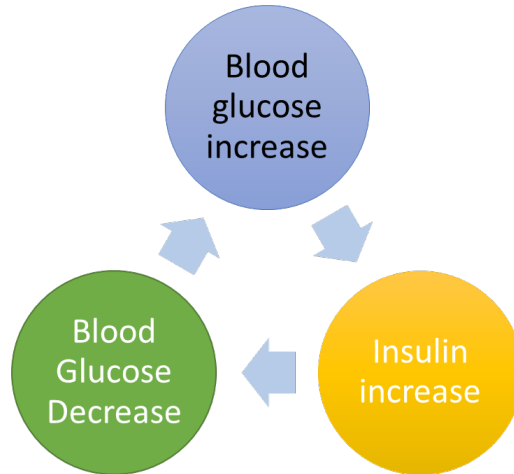


Fig. 8 Process of blood glucose adjustment

4.2 Transfer Function Analysis

We tried to use a PI control system to mimic the processes of blood glucose adjustment. In the time domain, two equations (32,33) are used to describe the process of blood glucose adjustment. Equation (32) refers to when the blood glucose reaches the threshold value $\Delta C_{glu-input}$, then speed of insulin secretion is proportional to it, and the coefficient is K_1 . In equation (33), the decrease of blood glucose is proportional to the accumulation of insulin, which is the integration of speed of insulin secretion, and the coefficient is K_2 ¹¹.

$$v_{insulin} = K_1 \Delta C_{glu-input} \quad (32)$$

$$\Delta C_{glu-output} = K_2 \int_0^t v_{insulin} dt \quad (33)$$

Via Laplace transform, the transfer function diagram in frequency domain was obtained,

shown in figure 9.

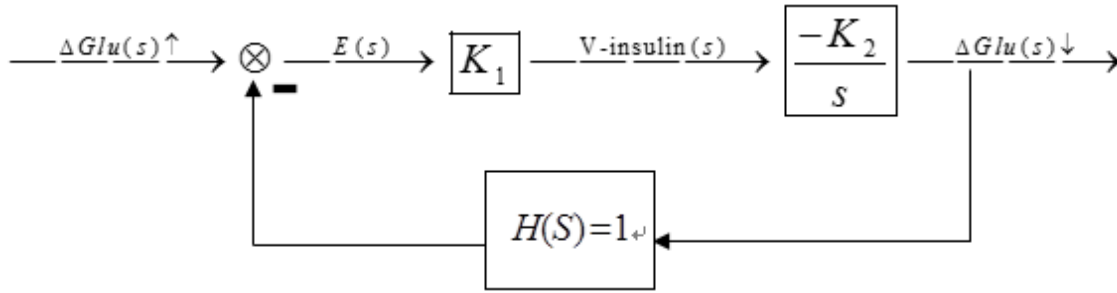


Fig. 9 Closed loop transfer function block diagram

The forward transfer function is:

$$F(s) = \frac{-K_1 K_2}{s} \quad (34)$$

The feed back function is:

$$H(s) = 1 \quad (35)$$

So the open-loop transfer function is

$$F(s)H(s) = \frac{-K_1 K_2}{s} \quad (35)$$

Then the transfer function of system was derived as equation (36), and the output, the decrease of blood glucose, was shown as equation (37).

$$\Phi(s) = \frac{F(s)}{1 + F(s)H(s)} = \frac{-K_1 K_2}{s - K_1 K_2} \quad (36)$$

$$Y(s) = \Phi(s) \cdot X(s) \quad (37)$$

4.3 The stability analysis

We can use bode diagram to analyze the stability of the system. The stability of the system can be judged via the value of phase at magnitude going across the zero line. This is one order system and its phase is 90 degrees. It is always stable.

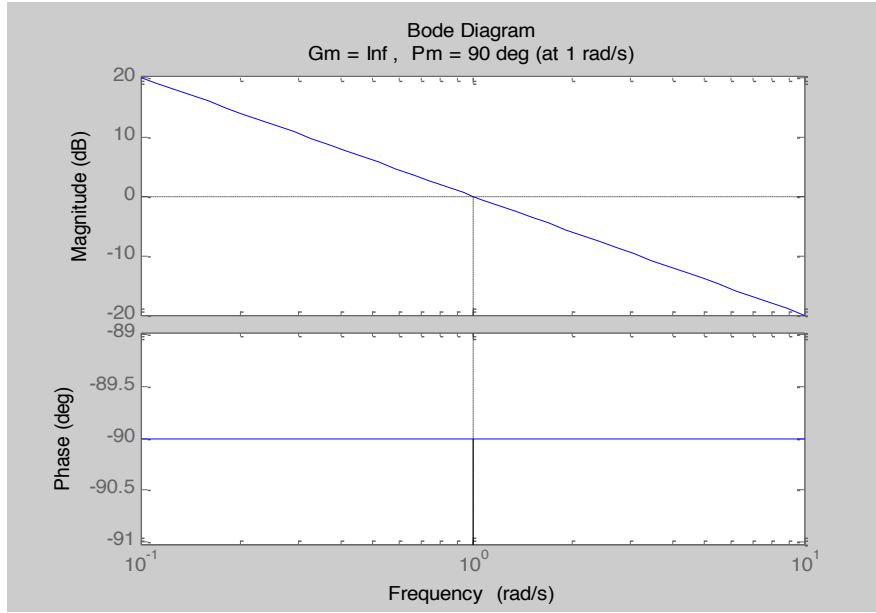


Fig.10 Bode Diagram of system

4.4 The Error Analysis

When the blood glucose increases, our body will secrete corresponding amount of insulin to suppress the increase of glucose. It takes time. How long it will take? We consider the time point when the error smaller than 0.02 as the end point.

$$E(s) = \frac{1}{1 + F(s)H(s)} \cdot X(s) = \frac{1}{1 - K_1K_2 / s} \cdot X(s) \quad (38)$$

When the input $x(t)$ is: $x(t) = C$ ($t > 0$)

$$\text{then } X(s) = \frac{C}{s}$$

$$Y(s) = \frac{-K_1K_2}{s - K_1K_2} \cdot \frac{C}{s} = C \left(\frac{1}{s} - \frac{1}{s - K_1K_2} \right) \quad (39)$$

$$y(t) = L^{-1}(Y(s)) = C(1 - e^{K_1K_2t}) \quad (40)$$

$$\text{steady-state error is: } e(t) = Ce^{K_1K_2t} \quad (41)$$

As stated before, we assume that when the blood glucose amount reaches a threshold of 6g (i.e., blood glucose level raised by 120 mg/dL), the insulin secretion will start. That

means the PI control system is initiated.

Substitute $k_1 \approx 0.3$, $k_2 \approx 0.001$ and $x(t) = 120 \text{ mg/dL}$ ($t \geq 0$) into the function, we got the solution $y(t)$.

$$y(t) = L^{-1}(Y(s)) = C(1 - e^{-K_1 K_2 t}) = 120(1 - e^{-3t/10000})$$

$$\frac{E(s)}{X(s)} = \frac{1}{1 + F(s)H(s)} = \frac{1}{1 - K_1 K_2 / s} \quad (42)$$

$$\frac{e(t)}{x(t)} = L^{-1}\left(\frac{E(s)}{X(s)}\right) = e^{-K_1 K_2 t} = e^{-3t/10000} \leq 2\%$$

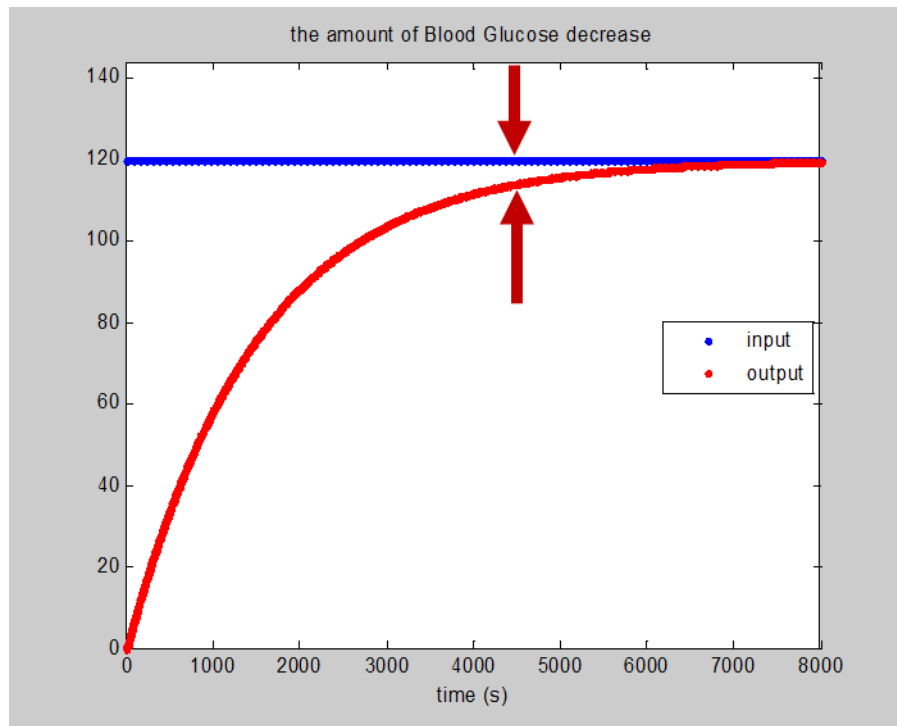


Fig.11 Input (blue line) and output (red line) of system.

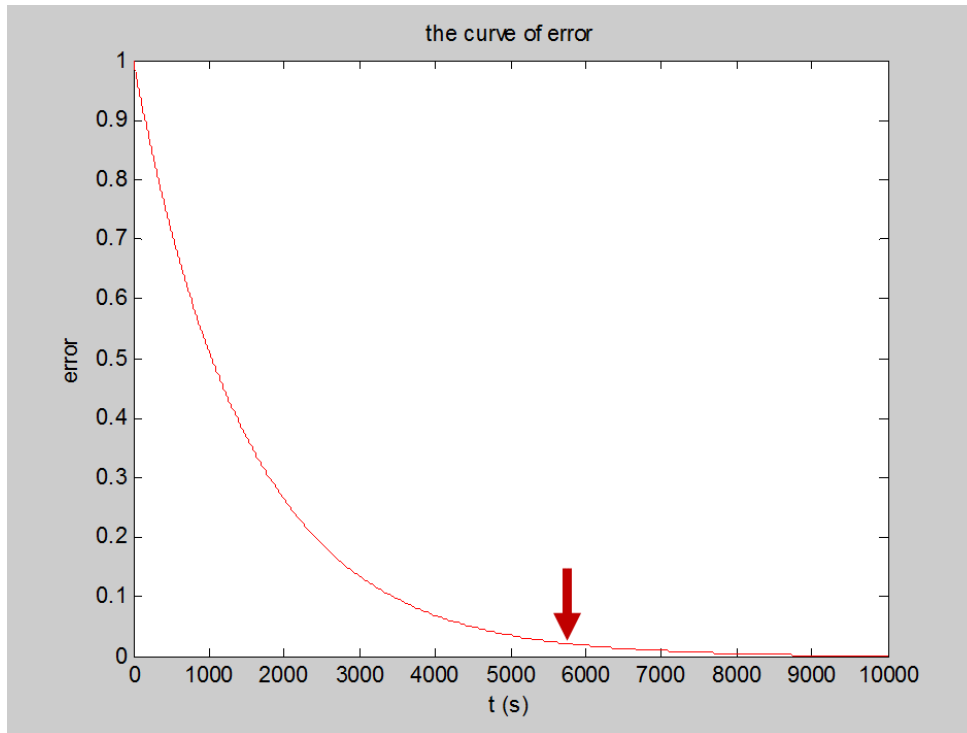


Fig. 12 The error of system.

When $t \geq \frac{\ln 0.02}{-K_1 K_2} = 12909 \text{ (s)} \approx 3.5 \text{ (h)}$, we have $\frac{e(t)}{x(t)} \leq 0.02$

5. Discussion and Conclusion

5.1 Results

Based on our calculations, the time periods for glucose diffusion and insulin regulation are 210 s and 3.5 h respectively. According to literature, the time for food digestion is about 0.33 h. Therefore, the whole period T , equal to the summation of the time of digestion, diffusion and adjustment, is close to 4 h. From the above-mentioned results, one can see that the diffusion process of glucose into blood happens much faster compared to the insulin-mediated regulation process of blood glucose. These meet the real biological situations.

In conclusion, using our models of diffusion and regulation, we can mimic the processes of blood glucose adjustment and calculate the span of period.

5.2 Limitations and future work

Firstly, in real case, insulin secretion is characterized as continuous basal secretion with peaks that occur after meals¹¹. No specific threshold has been reported. Actually, insulin secretes throughout the regulation process with different speeds^{7,11}. Secondly, glucose transportation is also a continuous process. The concentration of glucose in epithelium and blood changes with active and passive transportation rather than maintain at the constant level. Blood glucose will not increase to 210mg/dL from 90mg/dL abruptly. Hence, our model can be further improved by using time-dependent boundary conditions. Analytical solutions can be obtained by employing Green's function.

In the control module, we use the increased glucose concentration as the constant input. However, since the glucose diffusion speed changes with time, we could consider using flux as a continuous input. Meanwhile, only first-order control is adopted in our model to simplify the problem. Second-order or higher-order control models can also be considered in the future to better simulate the complex glucose-insulin relationship in reality.

6. Reference

1. Muntean C. et al (2012) The role of the central nervous system in glucose homeostasis. Romanian Journal of Diabetes Nutrition and Metabolic Diseases Volume 19, Issue 2, Pages 207–214
2. Molecular Cell Biology. 4th edition. Section 15.7 Transport across Epithelia
3. Digestion and absorption. 2010.
http://csls-text2.c.u-tokyo.ac.jp/inactive/08_02.html
4. Thennadil S. et al. (2001) Comparison of Glucose Concentration in Interstitial Fluid, and Capillary and Venous Blood During Rapid Changes in Blood Glucose Levels. DIABETES TECHNOLOGY & THERAPEUTICS Volume 3, Number 3
5. Rossetti P. et al (2010) Estimating Plasma Glucose from Interstitial Glucose: The Issue of Calibration Algorithms in Commercial Continuous Glucose Monitoring Devices. Sensors 2010, 10, 10936-10952
6. Transport across epithelia. 2000.
www.ncbi.nlm.nih.gov/books/NBK21502/figure/A4123/?report=objectonly
7. DeFronzo R. et al (1979) Glucose clamp technique: a method for quantifying insulin secretion and resistance. Am. J. Physiol. 237(3): E214-. E223
8. Expected Blood Glucose After a High-Carb Meal. Suzanne Robin, Demand Media
9. Periodontal Disease & Diabetes. 2005. <https://bohone09.wikispaces.com/Group+7>
10. Small intestine. Last modified on 17 November 2014.
http://en.wikipedia.org/wiki/Small_intestine
11. Ferrannini E. et al (1979) Pattern of insulin delivery after intravenous glucose injection in man and its relation to plasma glucose disappearance. J Clin Invest. 1979 Jul;64(1):243-54.

7. Appendix

Code of diffusion module

```

function Project
global D L Co Ce ns CL
Co = 0.00090; % I.C. in interstitial space
Ce = 0.0023; % kg/L. B.C.@ x=0
CL = 0.00090; % B.C.@ x=L
L = 0.001; % 1mm
D = 1e-9; %2e-4;
ns = 50; % number of terms
t_end = 1e2; % s
A = 1e-1; % area at x = L; mm^2; 30m^2
%S = 0.006; % kg; 6g

dx = 0.01*10^(-3); % step size in x dimension
dt = 1e-1; % step size in t dimension
xmesh = 0:dx:L; % domain in x; L
tmesh = 0:dt:t_end; % domain in t
nx = length(xmesh); % number of points in x dimension
nt = length(tmesh); % number of points in t dimension

%% Analytical solution
u = zeros(nt, nx);
for n = 1:ns
    Bn = 2*(CL-Co)*cos(n*pi)/(n*pi) - 2*(Ce-Co)/(n*pi);
    Kn=n*pi/L;
    On_x=sin(Kn*xmesh);
    u = u + exp(-D*(Kn^2)*tmesh)* Bn*On_x;
end
u=u + Ce*ones(nt,nx) + ones(1,nt)*((CL - Ce)*xmesh/L);
figure('Color','w'); h = surf(tmesh,xmesh,u);
set(h, 'LineStyle', 'none')
%surf(tmesh,xmesh,u);
%mesh(tmesh,xmesh,u);
title(['Analytical Solution (n = ', num2str(ns),)'],'FontSize',12)

```

```

xlabel('Time (s)','FontSize',12)
ylabel('Distantce x (m)','FontSize',12)
zlabel('Concentration (kg/L)','FontSize',12)

%% Flux
J = zeros(nt, nx);
for n = 1:ns
    Kn=n*pi/L;
    Bn = 2*(CL-Co)*cos(n*pi)/L - 2*(Ce-Co)/L;
    J = J - D*Bn*exp(-D*(Kn^2)*tmesh)*cos(Kn*xmesh);
end
J = J - (CL - Ce)*D/L*ones(nt,nx);
figure('Color','w'); h = surf(tmesh,xmesh,J);
set(h, 'LineStyle', 'none')
title(['Flux (n = ', num2str(ns),')'],'FontSize',12)
xlabel('Time (s)','FontSize',12)
ylabel('Distance x (m)','FontSize',12)
zlabel('Flux J(x,t)','FontSize',12)
zlim([-1e-6 1e-6])

%% Amount of glucose entering the blood flow
y = 0;
for n = 1:ns
    Bn = (2*L*cos(n*pi)/(n*pi)^2)*((CL-Co)*cos(n*pi) - (Ce-Co));
    y = y + Bn * (exp(-D*(n*pi/L)^2*tmesh) - 1);
end
y = A*(D*(Ce-CL)/L*tmesh + y);
figure('Color','w'); plot(tmesh,y,'+')
title(['Amount of glucose taken away by blood'],'FontSize',12)
xlabel('Time (s)','FontSize',12)
ylabel('Amount (kg)','FontSize',12)

%% pdepe
sol_pdepe = pdepe(0, @pdefun,@icfun,@bcfun,xmesh,tmesh);

figure('Color','w'); h = surf(tmesh,xmesh,sol_pdepe);
max(max(sol_pdepe));
set(h, 'LineStyle', 'none')

```

```
xlabel('Time (s)','FontSize',12);  
ylabel('Distance x (m)','FontSize',12);  
title('Matlab pdepe','FontSize',12);  
zlabel('Concentration (kg/L)','FontSize',12);
```

```
function [c,f,s] = pdefun(x,t,u,DuDx)
```

```
global D
```

```
c = 1;
```

```
f = D*DuDx;
```

```
s = 0;
```

```
function u0 = icfun(x)
```

```
global Co
```

```
u0 = Co;
```

```
function [pl,ql,pr,qr] = bcfun(xl,ul,xr,ur,t)
```

```
global CL Ce
```

```
pl = ul-Ce;
```

```
ql = 0;
```

```
pr = ur-CL;
```

```
qr = 0;
```

Code of control module

```
close all;
```

```
GH=tf([1],[1,0]);
```

```
[mag,pha,w]=bode(GH);
```

```
margin(mag,pha,w);
```

```
%% analysis of the system
```

```
C0=120;
```

```
K=1/1500;
```

```
x=C0*(t>0);
```

```
y=C0*(1-exp(-K*t));
```

```
t=linspace(0,10000,10000);
```

```
figure() % plot the curve of output of system
```

```
plot(t,x,'b.',t,y,'r');
```

```
xlabel('time (s)');  
title('input--output of system');  
legend('input','output','location','east')  
axis([-10 8000 0 1.2*C0]);
```

```
e=C0*exp(-K*t);  
e=exp(-K*t);  
figure()% plot the curve of error of system  
plot(t,e,'r-');  
xlabel('t (s)');  
ylabel('error');  
title('the curve of error');  
axis([-10 8000 0 1.2*C0]);
```

```
T_steady=log(0.02)/(-K);% calculate the time when the erroe of system can be neglected.
```