Identification and low-complexity regime-switching insulin control of type I diabetic patients

Ali Hariri[1], Le Yi Wang[2]

[1]DTE Energy, One Energy Plaza, Detroit, Michigan, USA;
[2]ECE Department, Wayne State University, Detroit, USA.
Email: hariria@dteenergy.com, U.S.lywang@wayne.edu

ABSTRACT

This paper studies benefits of using simplified regime-switching adaptive control strategies in improving performance of insulin control for Type I diabetic patients. Typical dynamic models of glucose levels in diabetic patients are nonlinear. Using a linear time invariant controller based on an operating condition is a common method to simplify control design. On the other hand, adaptive control can potentially improve system performance, but it increases control complexity and may create further stability issues. This paper investigates patient models and presents a simplified switching control scheme using PID controllers. By comparing different switching schemes, it shows that switched PID controllers can improve performance, but frequent switching of controllers is unnecessary. These findings lead to a control strategy that utilizes only a small number of PID controllers in this scheduled adaptation strategy.

Keywords: Insulin Control; Diabetes; Switching Control; Modeling; Adaptation

1. INTRODUCTION

Insulin is a hormone that is necessary for converting the blood sugar, or glucose, into usable energy. The human body maintains an appropriate level of insulin. Diabetes are caused by lack of insulin in the body. There are two major types of diabetes, called type 'I' and type 'II' diabetes. Type 'I' diabetes are called Insulin Dependent Diabetes Mellitus (IDDM), or juvenile onset diabetes mellitus. Type 'II' diabetes are known as Non-Insulin Dependent Diabetes Mellitus (NIDDM) or Adult-Onset Diabetes (AOD) [1-7]. This paper is focused on type 'I' diabetes. Type 'I' diabetes develop when the pancreas stop producing insulin. Consequently, insulin must be provided through injection or continuous infusion to control glucose levels.

The insulin infusion rate to a diabetic patient can be administrated based on the glucose (sugar) level inside the body. Over the years many mathematical models have been developed to describe the dynamic behavior of human glucose/insulin systems. The most commonly used model is the minimal model introduced by Bergman, *et al.* [6,8-13]. The minimal model consists of a set of three differential equations with unknown parameters. Since diabetic patients differ dramatically due to variations of their physiology and pathology characteristics, the parameters of the minimal model are significantly different among patients. Based on such models, a variety of control technologies have been applied to glucose/insulin control problems [14-17].

This paper studies benefits of using simplified adaptation control strategies in improving performance of insulin control for Type I diabetic patients. Typical dynamic models of glucose levels in diabetic patients are nonlinear. Using a linear time-invariant controller based on an operating condition is a common method to simplify control design. On the other hand, adaptive control can potentially improve system performance, but it increases control complexity and may create further stability issues. This paper investigates patient model identification and presents a simplified switching control scheme using PID controllers. By comparing different switching schemes, it shows that switched PID controllers can improve performance, but frequent switching is unnecessary. These findings lead to a control strategy that utilizes only a small number of PID controllers in this scheduled adaptation strategy.

Many methods and techniques have been investigated, tested, and studied for controlling the glucose level in type 'I' diabetes patients. Lynch and Bequette [14] tested the glucose minimal model of Bergman [10] to design a Model Predictive Control (MPC) to control the glucose level in diabetes patients. Also in that study the nonlinear

mathematical model was linearized about the steady-state values of time variant variables. Fisher [15] used the glucose insulin minimal model to design a semi closed-loop insulin infusion algorithm based on three hourly plasma glucose samplings. The study concentrated on the glucose level, and did not take in consideration of some important factors such as the free plasma insulin concentration and the rate at which insulin is produced as the level of glucose rises. Furler [16,18] modified the glucose insulin minimal model by removing the insulin secretion and adding insulin antibodies to the model. The algorithm calculates the insulin infusion rate as a function of the measured plasma glucose concentration. The linear interpolation was used to find the insulin rate. The algorithm neglected some important variations in insulin concentration and other model variables. Ibbini, Masadeh and Amer [17] tested the glucose minimal model to design a semi closed-loop optimal control system to control the glucose level in diabetes patients.

The rest of this paper is organized as follows. Section 2 discusses basic model structures, experimental data, and model simulations. Section 3 is focused on model parameter identification. The Levenberg-Marquardt algorithm is employed to obtain model parameters iteratively. Control design and switching strategies are presented in Section 4. By comparing fixed PID controllers with switching strategies, we first show that adaptation can be beneficial. Further studies show that using a small set of PID controllers in switching control is a feasible and desirable approach, which simplifies switching scheme without performance degeneration.

2. MODELS

Since the level of the glucose inside the human being body changes significantly in response to food intake and other physiological and environment conditions, for control design it is necessary to derive mathematics models to capture such dynamics [8-10,19-21]. Many mathematical models have been developed to describe the human glucose/insulin system. Such models are highly nonlinear and usually very complex. The most commonly used and simplified model is the Minimal Model introduced by Bergman, *et al.* [8-10], which is still nonlinear. To further simplify the model for control design, a common practice is to locally linearize the Minimal Model under a given operating condition.

2.1. Model Structures

The minimal model of the glucose and insulin is perhaps the simplest model that is physiologically based and represents well for the observed glucose/insulin dynamics of a diabetic patient. The insulin enters or exits the

interstitial insulin compartment at a rate that is proportional to the difference $I(t) - Ib$ of plasma insulin $I(t)$ and the basal insulin level *Ib* [22,23]. If the level of insulin in the plasma is below the insulin basal level, insulin exits the interstitial insulin compartment; and if the level of insulin in the plasma is above the insulin basal level, insulin enters the interstitial insulin compartment. Insulin also can flee the interstitial insulin compartment through another route at a rate that is proportional to the insulin amount inside the interstitial insulin compartment. On the other hand, glucose enters or exits the plasma compartment at a rate that is proportional to the difference $G(t) - G_b$ of the plasma glucose level $G(t)$ and the basal glucose level G_b. If the level of glucose in the plasma is below the glucose basal level, the glucose exits the plasma compartment; and if the level of glucose in the plasma is above the glucose basal level, glucose enters the glucose compartment. Glucose also can flee the plasma compartment through another route at a rate that is proportional to the glucose amount inside the interstitial insulin compartment.

Currently, the most widely used model in physiological research on the metabolism of glucose is the Minimal Model. This model structure describes experimental data well with the smallest set of identifiable and meaningful parameters. The Minimal Model consists of two parts: the minimal model of glucose disappearance and the minimal model of insulin kinetics.

$$\frac{dG(t)}{dt} = -\left[P_1 + x(t) \right] G(t) + P_1 G_b \tag{1}$$

$$\frac{dx(t)}{dt} = -P_2 x(t) + P_3 \left[I(t) - I_b \right] \tag{2}$$

$$\frac{dI(t)}{dt} = -n I(t) + \gamma \left[G(t) - h \right] t \tag{3}$$

where $G(t)$ (mg/dL) is the blood glucose level in plasma; $I(t)$ (μU/mL) is the insulin concentration level in plasma; $x(t)$ (min^{-1}) is the variable which is proportional to insulin in the remote compartment, G_b (mg/dL) is the basal blood glucose level in plasma; I_b (μU/mL) is the basal insulin level in plasma; t (min) is the time interval from the glucose injection. **Eqs.1** and **2** represent the glucose disappearance and **Eq.3** represents the insulin kinetics. The initial conditions of the above three differential equations are as the following:
$G(0) = G_0, x(0) = 0, I(0) = I_0$

The model parameters carry some physiological meanings that can be summarized as follows. P_1 (min^{-1}) describes the "glucose effectiveness" which represents the ability of blood glucose to enhance its own disposal at the basal insulin level. P_2 (min^{-1}) describes the

decreasing level of insulin action with time. P_3 (min^{-2} · $(\mu U/mL)^{-1}$) describes the rate in which insulin action is increased as the level on insulin deviates from the corresponding baseline. γ ($(\mu U/mL) \cdot (mg/dL)^{-1} \cdot min^{-1}$) denotes the rate at which insulin is produced as the level of glucose rises above a "target glycemia" level. n (min^{-1}) represents fractional insulin clearance. h (mg/dL) is the pancreatic "target glycemia" level. G_0 (mg/dL) is the theoretical glucose concentration in plasma extrapolated to the time of glucose injection $t = 0$ [8-10,24]. I_0 ($\mu U/mL$) is the theoretical plasma insulin concentration at $t = 0$. $\mu U/mL$ is the conventional unit to measure the insulin level and has the following conversion formula: 1 micro-unit/milliliter = 6 picomole/liter (1$\mu U/mL$ = 6 pmol/L) [25,26]. P_1, P_2, P_3, n, γ, h, G_0 and I_0 are the parameters to be estimated.

A fourth differential equation will be added to the set of the Minimal Model equations to represent a first-order pump dynamics:

$$\frac{dU_1(t)}{dt} = \frac{1}{a}\left(-U_1(t) + u(t)\right) \tag{4}$$

where $U_1(t)$ is the infusion rate, $u(t)$ is the input command, and a is the time constant of the pump.

2.2. Experimental Data

A new approach was developed by Bergman, *et al.* [8-10] to compute the pancreatic responsiveness and insulin sensitivity in the intact organism. This approach uses computer modeling to investigate the plasma glucose and insulin dynamics during a Frequently Sampled Intravenous Glucose Tolerance (FSIGT). An amount of glucose was injected at $t = 0$ over a period of time equal to 60 seconds [8-10,27]. The blood samples were taken from a fasting subject at regular intervals of time, and then analyzed for glucose and insulin content. Glucose was measured in triplicate by the glucose oxidize technique on an automated analyzer. The coefficient of variation of a single glucose determination was about ± 1.5%. Insulin was measured in duplicate by radioimmunoassay, with dextrin-charcoal separation using a human insulin standard. **Table 1** shows the FSIGT test data for a normal individual.

2.3. Model Simulation

Implementation of the minimal model can be achieved by using computer simulation tools. The two differential **Eqs.1** and **2** of the minimal model that correspond to the glucose kinetics are modeled here by using the MATLAB/Simulink software. In this model, the insulin $I(t)$ is considered as an input and the glucose $G(t)$ as an output. The values of the input $I(t)$ at a time interval are given in **Table 1**. The simulation diagram of the minimal model for the glucose kinetic is shown in **Figure 1**. The output of the system, glucose $G(t)$, is

Table 1. FSIGT test data for a normal individual.

Sampling time (minutes)	Glucose level (mg/dL)	Insulin level ($\mu U/mL$)
0	92	11
2	350	26
4	287	130
6	251	85
8	240	51
10	216	49
12	211	45
14	205	41
16	196	35
19	192	30
22	172	30
27	163	27
32	142	30
42	124	22
52	105	15
62	92	15
72	84	11
82	77	10
92	82	8
102	81	11
122	82	7
142	82	8
162	85	8
182	90	7

shown in **Figure 2** for a normal individual with the following parameters: $P_1 = 3.082 \times 10^{-2}$, $P_2 = 2.093 \times 10^{-2}$, $P_3 = 1.062 \times 10^{-5}$, $G_0 = 350$, $G_b = 92$, $I_b = 11$, [8-10].

3. PARAMTERS ESTIMATION

3.1. Parameter Estimation

Parameter estimation is to determine the values of model parameters that provide the best fit to measured data, based on some error criteria such as least-squares. The method of least squares assumes that the best-fit curve of a given set of data is the curve that has the minimal sum of the deviations squared from a given set of data [28-31]. Given a set of data $(x_1, y_1), (x_2, y_2), (x_3, y_3), \cdots, (x_N, y_N)$, where the independent variable is x and the dependent variable is y, under a selected model function form $f(x)$ the least squares (LS) estimation seeks to minimize

$$\Pi = \sum_{i=1}^{N}\left[y_i - f(x_i)\right]^2 \tag{5}$$

When the function is an m-th degree polynomial

$$f(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_m x^m \tag{6}$$

we have

$$\begin{aligned}\Pi &= \sum_{i=1}^{N}\left[y_i - f(x_i)\right]^2 \\ &= \sum_{i=1}^{N}\left[y_i - \left(a_0 + a_1 x_i + a_2 x_i^2 + \cdots + a_m x_i^m\right)\right]^2\end{aligned} \tag{7}$$

The unknown coefficients $a_0, a_1, a_2, \cdots, a_m$, can be estimated to yield a least squares error.
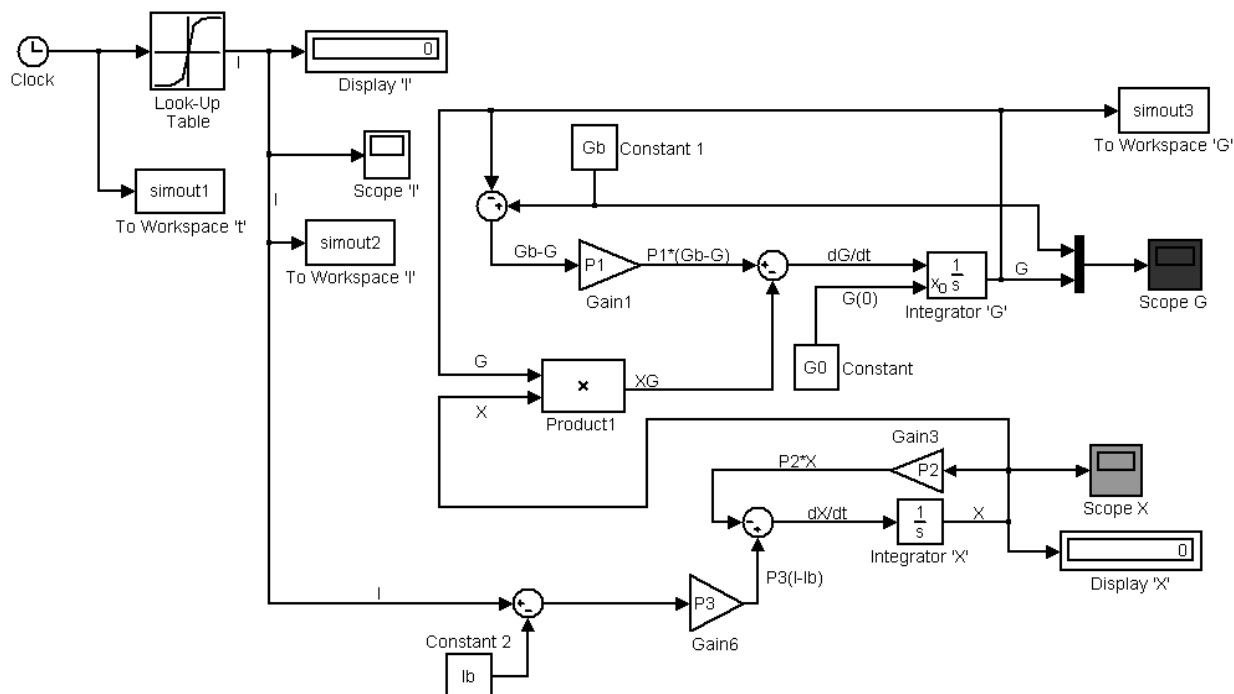
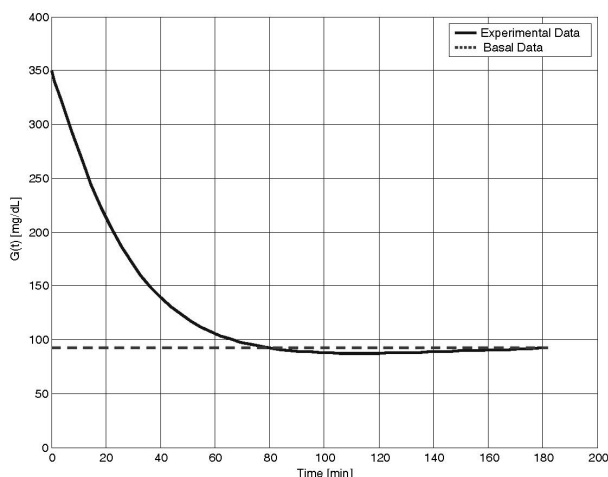**Figure 1.** Simulation diagram of the glucose kinetic model.

Figure 2. The output of the minimal model for the glucose kinetics

Model parameters can be obtained iteratively to reduce computational complexity. It starts with an initial guess of the unknown parameters. Each iteration updates the current estimate based on new observations. Suppose there are m base functions $f_1, f_2, \cdots, f_m$ of n parameters $p_1, p_2, \cdots, p_n$. The functions and the parameters can be represented as follows:

$$\boldsymbol{f}^{\mathrm{T}} = (f_1, f_2, \cdots, f_m)$$
$$\boldsymbol{p}^{\mathrm{T}} = (p_1, p_2, \cdots, p_n) \tag{8}$$

The least square method is to find the values of the unknown parameters $p_1, p_2, \cdots, p_n$ for which the cost function is minimum, *i.e.*

$$S(\boldsymbol{P}) = \frac{1}{2} \boldsymbol{f}^{\mathrm{T}} \quad \boldsymbol{f} = \frac{1}{2} \sum_{i=1}^{m} f_i^2(\boldsymbol{p}) \tag{9}$$

The Levenberg-Marquardt algorithm is an iterative technique that seeks the minimum of a multivariate function that is expressed as the sum of squares of nonlinear real-valued functions [28]. It has become a standard technique for nonlinear least-squares problems. Levenberg-Marquardt can be thought of as a combination of steepest descent and the Gauss-Newton method. When the current solution is far from the correct one, the algorithm behaves like a steepest descent method which is guaranteed to converge. When the current solution is close to the correct solution, it becomes a Gauss-Newton method.

The Levenberg-Marquardt algorithm is an iterative procedure. Let $\hat{\boldsymbol{x}} = f(\boldsymbol{p})$ be the parametrized model function. The minimization starts after an initial guess for the parameter vector p is provided. The algorithm is locally convergent, namely it converges when the initial guess is close to the true values. In each iteration step, the parameter vector $\boldsymbol{p}$ is updated by a new estimate $\boldsymbol{p} + \varepsilon$, where ε is a small correction term that can be determined by a Taylor Series expansion which leads to the following approximation:

$$f(\boldsymbol{p} + \varepsilon_p) \approx f(\boldsymbol{p}) = \boldsymbol{J}\varepsilon_p \tag{10}$$

where, $\boldsymbol{J}$ is the Jacobian of $\boldsymbol{f}$ at $\boldsymbol{p}$

$$J = \frac{\partial f(\boldsymbol{p})}{\partial \boldsymbol{p}} \qquad (11)$$

Levenberg-Marquardt iterative initiates at the starting point p_0, and produces a series of vectors p_1, p_2, p_3 etc, that converge towards a local minimizer p^+ of $\boldsymbol{f}$. At each step, it is required to find the ε which minimizes the value of

$$\left\| x - f\left(p + \varepsilon_p\right) \right\| \approx \left\| x - f(p) - J\varepsilon_p \right\|$$

That gives the following:

$$\left\| x - f\left(p + \varepsilon_p\right) \right\| \approx \left\| x - \hat{x} - J\delta_p \right\| = \left\| e - J\varepsilon_p \right\| \qquad (12)$$

where ε_p is the solution to a linear least squares problem. The minimum is achieved when the term $J\varepsilon_p - e$ is orthogonal to column space J. Based on that, the following can be concluded

$$J^{\mathrm{T}}\left(J\varepsilon_p - e\right) = 0 \qquad (13)$$

Eq.13 can be rearranged as the following

$$J^{\mathrm{T}}\left(J\varepsilon_p - e\right) = J^{\mathrm{T}}e \qquad (14)$$

The Levenberg-Marquardt algorithm solves a slight variation of **Eq.14**, which is known as the augmented normal equation

$$N\varepsilon_p = J^{\mathrm{T}}e \qquad (15)$$

where the diagonal elements of N are computed as $N_{ii} = \eta + \left[J^{\mathrm{T}}J\right]_{ii}$ for $\eta > 0$, while the rests of the matrix N are identical to those of the matrix $\left[J^{\mathrm{T}}J\right]$. η is called the damping parameter. If the updated parameter vector, $p + \varepsilon_p$, where ε_p is computed from **Eq.15**, yields a reduction in the residual value or error e, then the update is valid and the process repeats with a decreased damping parameter η. Otherwise, the damping parameter is increased and the augmented normal **Eq.15** is solved again. Then the process iterates until a value of ε_p that reduces error is found. The MATLAB Software has the Optimization Toolbox which has a command called *Lsqnonlin* for this algorithm.

This algorithm is applied to our problem here. The FSIGT data sample in **Table 1** consists of 24 samples. The FSIGT samples were taken over a period of 182 minutes. The unknown parameters of the minimal model **Eq.1**, **Eq.2**, **Eq.3** were estimated by utilizing the Levenberg-Marquadrt Algorithm. The parameters to be estimated were given an initial guess, then the algorithm was used to update the parameters using the sequential data in **Table 1**. A MATLAB program was written to estimate the unknown parameters. The estimated values of those parameters are shown in **Table 2**.

Table 2. Estimated Parametres.

Parameters	Normal Individual #1	Normal Individual #2
P_1	0.032299	0.049519
P_2	0.0092644	41.5953
P_3	5.3004e–006	1.8577e–004
n	0.29858	0.14653
γ	0.0068676	1.0113e–005
h	90.3709	196.0531
G_0	295.6801	318.84
I_0	401.7177	203.2434

The values of the parameters shown in **Table 2** were implemented in the simulation diagram of the minimal model shown on **Figure 1**. The values of the glucose levels of both individuals are shown in **Table 3**.

The graphs of both experimental and simulated data for normal individual #1 and #2 are shown in **Figure 3** and **Figure 4** respectively.

These plots show that the two graphs (experimental and simulated) are close to each other, leading to the conclusion that the estimated values of parameters are close to the actual values. In general, the relative errors indicate how good an estimate is, relative to the true values. Although absolute errors are useful, they do no necessarily give an indication of the importance of an error. If the experimental value is denoted by G, and the estimated (or simulated) value is denoted by G, then the relative error is defined as

$$\text{Relative Error} = \frac{G - \hat{G}}{G} \qquad (16)$$

Table 3. Simulated data for normal individuals #1 and #2.

| Sampling time during the test (minutes)	Normal Individual #1 Glucose Level (mg/dL)	Normal Individual #2 Glucose Level (mg/dL)
0	94	94
2	295.6801	318.84
4	282.1308	297.2223
6	268.2993	277.7749
8	254.8580	260.2561
10	242.1020	244.4545
12	230.1353	230.1878
14	218.9702	217.2973
16	208.5776	205.6436
19	198.9116	195.1034
22	185.6659	181.1443
27	173.7810	169.1246
32	156.6159	152.6775
42	142.2917	139.8434
52	120.5167	122.0036
62	105.8327	111.1272
72	96.35277	104.4947
82	90.67314	100.4500
92	87.73594	97.98329
102	86.65621	96.47894
122	86.77255	95.56149
142	89.03143	94.66075
162	92.55181	94.32573
182	95.65837	94.20113
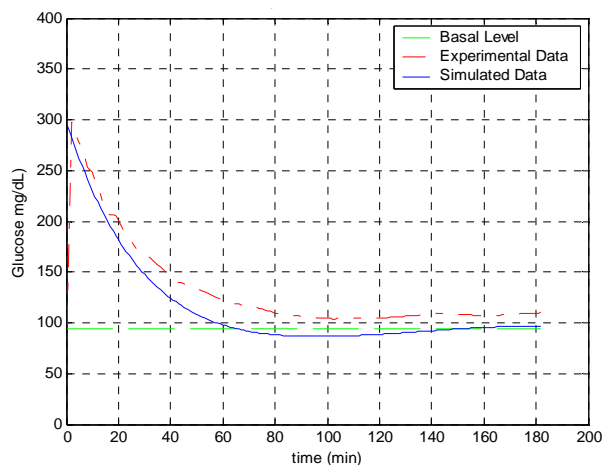
Figure 3. Plot of glucose level G(t) for normal individual #1.
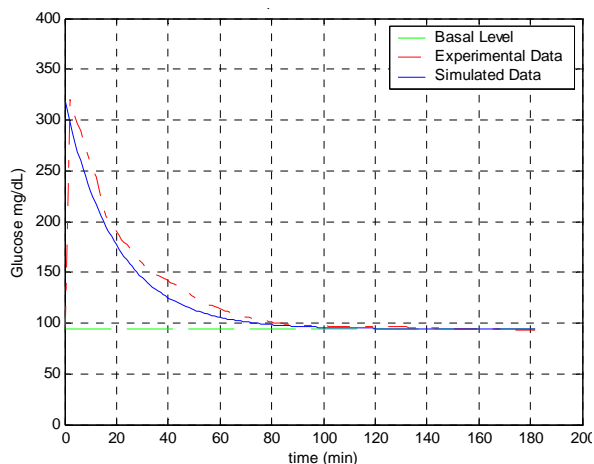

Figure 4. Plot of glucose level G(t) for normal individual #2.

And the Square Relative Error can be expressed as

$$\text{Square Relative Error} = \left(\frac{G - \hat{G}}{G}\right)^2 \quad (17)$$

When the data is sampled over a certain period of time, the mean squared relative error (MSRE) can be used. The MSRE is defined as

$$\text{MSRE} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{G_i - \hat{G}_i}{G_i}\right)^2, \text{ for } i = 1, 2, \cdots, n \quad (18)$$

where G_i is the experimental value at sample i. $\hat{G}_i$ is the estimated value at sample i. n is the number of samples of a data set.

The Square Relative Error between the experimental data and the simulated data of the glucose level for individual #1 and #2 are calculated and shown in **Table 4** and **Table 5** respectively. Normally the Mean Square Relative Error is expressed in percentage format. Below is the percentage error for both individuals: the percent-

Table 4. Error data for normal individuals #1.

Experimental Data, G(t)	Simulated Data, $\hat{G}(t)$	Square Relative Error
94	94	0
298	295.6801	6.060466e–005
284	282.1308	4.3317e–005
272	268.2993	0.0001851148
253	254.8580	5.393524e–005
248	242.1020	0.0005656016
235	230.1353	0.0004285321
217	218.9702	8.243416e–005
208	208.5776	7.710311e–006
205	198.9116	0.0008820624
191	185.6659	0.0007799362
172	173.7810	0.0001072176
164	156.6159	0.002027272
141	142.2917	8.391868e–005
132	120.5167	0.007568141
120	105.8327	0.01393832
116	96.35277	0.0286871
108	90.67314	0.02573902
106	87.73594	0.02968814
104	86.65621	0.02781129
105	86.77255	0.03013514
109	89.03143	0.03356148
107	92.55181	0.01823304
110	95.65837	0.01699855

Table 5. Error data for normal individuals #2.

Experimental Data, G(t)	Simulated Data, $\hat{G}(t)$	Square Relative Error
94	94	0
320	318.84	1.314063e–005
303	297.2223	0.0003635986
289	277.7749	0.001508629
272	260.2561	0.001864179
258	244.4545	0.002756472
244	230.1878	0.003204405
223	217.2973	0.0006539557
205	205.6436	9.857924e–006
194	195.1034	3.235126e–005
182	181.1443	2.210359e–005
169	169.1246	5.439401e–007
152	152.6775	1.986409e–005
139	139.8434	3.681781e–005
122	122.0036	8.715616e–010
112	111.1272	6.073247e–005
105	104.4947	2.315634e–005
100	100.4500	2.024906e–005
98	97.98329	2.909034e–008
97	96.47894	2.88559e–005
97	95.56149	0.0002199282
95	94.66075	1.275242e–005
94	94.32573	1.200794e–005
93	94.20113	0.0001668063

age MSRE for individual #1 = 0.99028%; the percentage MSRE for individual #2 = 0.04596%.

3.2. Model Linearization

Now let us recall the four differential **Eqs.1-4** that define the proposed mathematical model and denote them as $f_1(G)$, $f_2(x)$, $f_3(I)$, and $f_4(U_1)$. The result is

$$f_1(G) = \frac{dG(t)}{dt} = -\left[P_1 + x(t)\right]G(t) + P_1 G_b \qquad (19)$$

$$f_2(x) = \frac{dx(t)}{dt} = -P_2 x(t) + P_3\left[I(t) - I_b\right] \qquad (20)$$

$$f_3(I) = \frac{dI(t)}{dt} = -nI(t) + \gamma\left[G(t) - h\right]t + U_1(t) \qquad (21)$$

$$f_4(U_1) = \frac{dU_1(t)}{dt} = \frac{1}{a}\left(-U_1(t) + U(t)\right) \qquad (22)$$

The above equations can be written and arranged as follows.

The above equations can be further simplified as

$$f_1(G) = -P_1 G(t) - x(t)G(t) + P_1 G_b \qquad (23)$$

$$f_2(x) = -P_2 x(t) + P_3 I(t) - P_3 I_b \qquad (24)$$

$$f_3(I) = -nI(t) + \gamma t G(t) - \gamma h t + U_1(t) \qquad (25)$$

$$f_4(U_1) = -\frac{1}{a}U_1(t) + \frac{1}{a}U(t) \qquad (26)$$

The above system is a nonlinear system due to the presence of the nonlinear term that appears in **Eq.23**. The nonlinear term is $x(t)G(t)$. Now let us make the following definitions

$$x_1(t) = G(t),\ x_2(t) = x(t),\ x_3(t) = I(t),\ x_4(t) = u_1(t)$$

Then **Eqs.23** to **26** become

$$\frac{dx_1(t)}{dt} = -P_1 x_1(t) - x_1(t)x_2(t) + P_1 G_b \qquad (27)$$

$$\frac{dx_2(t)}{dt} = -P_2 x_2(t) + P_3 x_3(t) - P_3 I_b \qquad (28)$$

$$\frac{dx_3(t)}{dt} = \gamma t x_1(t) - nx_3(t) + x_4(t) - \gamma h t \qquad (29)$$

$$\frac{dx_4(t)}{dt} = -\frac{1}{a}x_4(t) + \frac{1}{a}u(t) \qquad (30)$$

The Jacobian matrices (J_x and J_u) of the model can be derived as

$$J_x = \begin{bmatrix} -P_1 - x_2 & -x_1 & 0 & 0 \\ 0 & -P_2 & P_3 & 0 \\ \gamma t & 0 & -n & 1 \\ 0 & 0 & 0 & -\dfrac{1}{a} \end{bmatrix}_{x=x_0,\ u=u_0}$$

and $J_u = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \dfrac{1}{a} \end{bmatrix}_{x=x_0,\ u=u_0}$ (31)

where the point (x_0, u_0) is the equilibrium point. The equilibrium point can be calculated by setting the state equations to zero and solving

$$-P_1 x_{10} - x_{10}x_{20} + P_1 G_b = 0 \qquad (32)$$

$$-P_2 x_{20} + P_3 x_{30} - P_3 I_b = 0 \qquad (33)$$

$$\gamma t x_{10} - nx_{30} + x_{40} - \gamma h t = 0 \qquad (34)$$

$$-\frac{1}{a}x_{40} + \frac{1}{a}u_0 = 0 \qquad (35)$$

where x_{10}, x_{20}, x_{30}, x_{40} and u_0 are the values of the state variables and the input at the operating point (*i.e.* the equilibrium point). At the equilibrium point, $u_0 = 0$,

Eq.35 becomes $-\dfrac{1}{a}x_{40} = 0$, that gives

$$x_{40} = 0 \qquad (36)$$

Substituting the value of x_{40} in **Eq.34** results in $\gamma t x_{10} - nx_{30} - \gamma h t = 0$ and

$$x_{30} = \frac{(x_{10} - h)\gamma t}{n} \qquad (37)$$

The value of x_{30} can be substituted in **Eq.33**

$$-P_2 x_{20} + P_3 \frac{(x_{10} - h)\gamma t}{n} - P_3 I_b = 0$$

to obtain

$$x_{20} = \frac{P_3 \gamma t x_{10}}{P_2 n} - \frac{P_3 \gamma t h}{P_2 n} - \frac{P_3 I_b}{P_2} \qquad (38)$$

Now, by substituting the value of x_{20} in **Eq.32** we have

$$-\frac{P_3 \gamma t}{P_2 n}(x_{10})^2 + \left(-P_1 + \frac{P_3 \gamma t h}{P_2 n} + \frac{P_3 I_b}{P_2}\right)x_{10} + P_1 G_b = 0 \quad (39)$$

The above equation is a 2nd order equation and can be solved by using the quadratic formula

$$x_{10} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a} \qquad (40)$$

where $a = -\dfrac{P_3 \gamma t}{P_2 n}$, $b = -P_1 + \dfrac{P_3 \gamma t h}{P_2 n} + \dfrac{P_3 I_b}{P_2}$, $c = P_1 G_b$

There are 2 possible values of x_{10}. Since x_{20} and x_{30} are expressed in term of x_{10}, there will be 2 possible values for each. Based on that, the controllability test will be studied to check which value of x_{10} is accepted.

3.3. A Case Study

The simulation diagram shown in **Figure 1** was used to simulate the data of a diabetic patient. The value of the parameters for a diabetic patient is shown below. P$_1$ = 0, P$_2$ = 0.81/100, P$_3$ = 4.01/1000000, I$_0$ = 192, G$_0$ = 337, γ = 2.4/1000, h = 93, n = 0.23, a = 2, G$_b$ = 99, I$_b$ = 8, [10].

The output of the simulated system is shown in **Figure 5**. By examining **Figure 5**, it can be clearly seen that the glucose level does not come down to the basal level after injecting an amount of 337 mg/dL of glucose inside a diabetic patient. **Figure 5** shows that the level of the glucose inside a diabetic patient decreases for the first 100 minutes, starts increasing afterward, and reaches the value of around 310 mg/dL after 3 hours from the time the glucose was injected. The goal is to bring down the value of the glucose inside a diabetic patient to the normal level or at least into a small neighborhood of the basal level. The above goal can be achieved by designing a PID feedback controller. The controller is to regulate the infusion rate and inject the required amount of the insulin inside the diabetic patient, and in turn the insulin will work inside the patient to bring down the level of glucose to the normal level or at least to the neighborhood of the normal level.

4. CONTOL DESIGN

We start with a linearized state space model for our systems. The general form of the state space is defined in the following equation:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \tag{41}$$

The proposed mathematical model at the equilibrium point (x_0, u_0) can be written in the state space form as shown below:

$$\dot{x} = \begin{bmatrix} -P_1 - x_{20} & -x_{10} & 0 & 0 \\ 0 & -P_2 & P_3 & 0 \\ \gamma t & 0 & -n & 1 \\ 0 & 0 & 0 & -\dfrac{1}{a} \end{bmatrix} x + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \dfrac{1}{a} \end{bmatrix} u \tag{42}$$

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} x$$
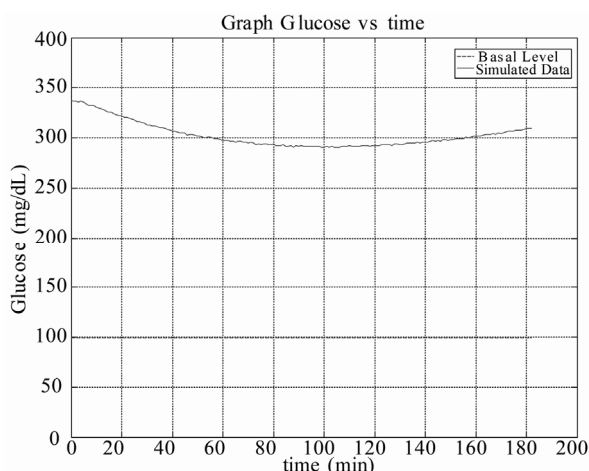
Figure 5. Output of the simulated system for diabetic.

where u is the input and y is the output of the system. The data of a diabetic person shown in Section 3.3 was used and the equilibrium point (x_0, u_0) was calculated as time varies from $t = 1$ min to $t = 182$ min. The two values for x_{10} that were calculated before were substituted in **Eq.42** and the controllability test was performed. It was found that only one value (the one obtained from the sign) makes the system to be controllable, hence only this value is used in the subsequent development.

4.1. Design of PID Controllers for Diabetic Patient

When designing a controller, the designer must define the specifications that need to be achieved by the controller. Normally the maximum overshoot (Mp) of the system step response should be small. Commonly a range between 10% and 20% is acceptable. Also the settling time (ts), is an important factor. The objective here is to design a PID controller, so that the closed-loop system has the following specifications: small steady-state error for a step input; less than 10% overshoot; settling time less than 60 minutes.

The patient dynamic system was expressed in the state space representation in (42). For an overshoot less than 10%, a damping ratio must be greater than 0.59, and a settling time less than 60 minutes implies that $(\zeta\omega_n)$ must be greater than 0.067.

The PID Controllers can be designed based on the models at different operating points. The following list contains the models at $t = 1, 20, 40, 60, 90, 120, 150$ and 182 minutes, and the corresponding PID controllers. Since **B** and **C** do not change with time, they are fixed in all cases as:

$$B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0.5 \end{bmatrix} \text{ and } C = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}$$

The control design is done by applying the root locus method and then evaluated by using the step response. For example, if the model at $t = 1$ min is used, after substituting the above numerical values (42), the root locus plot can be generated by the Matlab functions

MATLAB Program
Plotting the open loop system Root locus using MATLAB
[num, den] = ss2tf(A,B,C,D); rlocus(num, den) axis([–0.6 0.1 –0.5 0.5]) sgrid(0.59,0) sigrid(0.067)

The open loop poles are shown in **Figure 6**. These poles are located at –0.0040 + 0.0045*j*, –0.0040 – 0.0045*j*, –0.2302, –0.5. The four poles are stable, but the first two poles are very close to the imaginary axis and hence represent the slowest dynamics. The controller takes the form

$$Gc(S) = \frac{K(S+z_1)(S+z_2)}{S}$$

where K is the value of the gain where the root locus intersects with the line of the damping ratio. The z_1 and z_2 represent the value of the zeros to be added and may be selected to cancel the slowest poles of the dynamic system. Hence, select $z_1 = -0.004 + 0.0045j$, $z_2 = -0.004 - 0.0045j$.

The controller is * (please see the equation below) resulting in the system matrix A and the PID controller as

$$A_1 = \begin{bmatrix} 0 & -859.6667 & 0 & 0 \\ 0 & -0.0081 & 0.00000401 & 0 \\ 0.0024 & 0 & -0.23 & 1 \\ 0 & 0 & 0 & -0.5 \end{bmatrix},$$

$$G_C^1(S) = \frac{K(S^2 + 0.008S + 0.00003625)}{S}$$

The PID Parameters are:

$$K_p = 0.0444, \ K_i = 2.0094 \times 10^{-4}, \ K_d = 5.59$$

The design specifications of the system require the maximum overshoot to be less than 10% and the settling time to be less than 60 minutes. After inserting the PID controller in series with the patient system and connecting them in a unity feedback, we get the maximum overshoot 5.71% and the settling time 47.5 minutes (see **Figure 7**).

t = 20 minutes:

$$A_{20} = \begin{bmatrix} 0 & -131.33 & 0 & 0 \\ 0 & -0.0081 & 0.00000401 & 0 \\ 0.048 & 0 & -0.23 & 1 \\ 0 & 0 & 0 & -0.5 \end{bmatrix},$$

$$G_C^{20}(S) = \frac{K(S^2 + 0.0076S + 0.0001)}{S}$$

The PID Parameters are:

$$K_p = 0.2160, \ K_i = 0.0031, K_d = 28.4$$
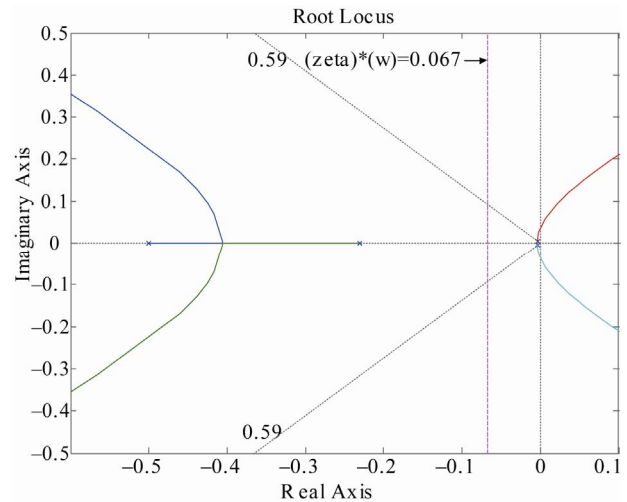
Figure 6. Output of the simulated system for Root Locus.
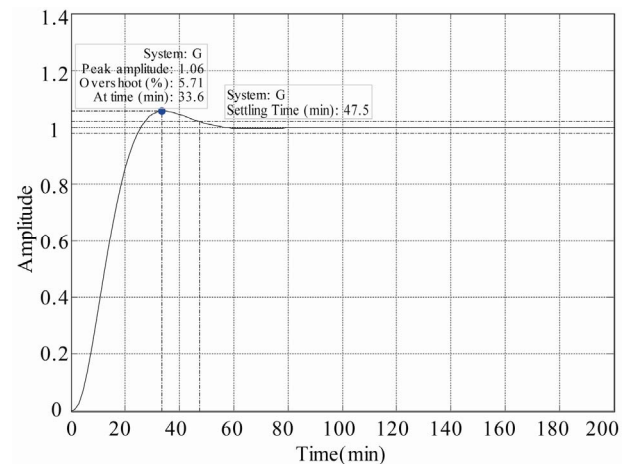
Figure 7. Output of the unit step response, using the model at $t = 1$ minute with $K = 5.5$.

t = 40 minutes:

$$A_{40} = \begin{bmatrix} 0 & -112.17 & 0 & 0 \\ 0 & -0.0081 & 0.00000401 & 0 \\ 0.096 & 0 & -0.23 & 1 \\ 0 & 0 & 0 & -0.5 \end{bmatrix},$$

$$G_C^{40}(S) = \frac{K(S^2 + 0.0072S + 0.0002)}{S}$$

The PID Parameters are:

$$K_p = 0.2374, \ K_i = 0.0061, K_d = 32.7$$

$$* Gc(S) = \frac{K(S - (-0.0040 + 0.0045j))(S - (-0.0040 - 0.0045j))}{S}$$

t = 60 minutes:

$$A_{60} = \begin{bmatrix} 0 & -105.78 & 0 & 0 \\ 0 & -0.0081 & 0.00000401 & 0 \\ 0.1440 & 0 & -0.23 & 1 \\ 0 & 0 & 0 & -0.5 \end{bmatrix},$$

$$G_C^{60}(S) = \frac{K(S^2 + 0.007S + 0.0003)}{S}$$

The PID Parameters are:

$$K_p = 0.0444, \ K_i = 2.0094 \times 10^{-4}, \ K_d = 5.59$$

t = 90 minutes:

$$A_{90} = \begin{bmatrix} 0 & -101.52 & 0 & 0 \\ 0 & -0.0081 & 0.00000401 & 0 \\ 0.2160 & 0 & -0.23 & 1 \\ 0 & 0 & 0 & -0.5 \end{bmatrix},$$

$$G_C^{90}(S) = \frac{K(S^2 + 0.0064S + 0.0004)}{S}$$

The PID Parameters are:

$$K_p = 0.2331, \ K_i = 0.0138, \ K_d = 36.4$$

t = 120 minutes:

$$A_{120} = \begin{bmatrix} 0 & -99.39 & 0 & 0 \\ 0 & -0.0081 & 0.00000401 & 0 \\ 0.288 & 0 & -0.23 & 1 \\ 0 & 0 & 0 & -0.5 \end{bmatrix},$$

$$G_C^{120}(S) = \frac{K(S^2 + 0.0058S + 0.0005)}{S}$$

The PID Parameters are:

$$K_p = 0.2234, \ K_i = 0.0187, \ K_d = 37.9$$

t = 150 minute:

$$A_{150} = \begin{bmatrix} 0 & -98.11 & 0 & 0 \\ 0 & -0.0081 & 0.00000401 & 0 \\ 0.36 & 0 & -0.23 & 1 \\ 0 & 0 & 0 & -0.5 \end{bmatrix},$$

$$G_C^{150}(S) = \frac{K(S^2 + 0.0054S + 0.0006)}{S}$$

The PID Parameters are:

$$K_p = 0.2032, \ K_i = 0.0229, \ K_d = 37.7$$

t = 182 minute:

$$A_{182} = \begin{bmatrix} 0 & -97.21 & 0 & 0 \\ 0 & -0.0081 & 0.00000401 & 0 \\ 0.4368 & 0 & -0.23 & 1 \\ 0 & 0 & 0 & -0.5 \end{bmatrix},$$

$$G_C^{182}(S) = \frac{K(S^2 + 0.0048S + 0.0007)}{S}$$

The PID Parameters are:

$$K_p = 0.1870, \ K_i = 0.0281, \ K_d = 38.5$$

4.2. Individual PID Controllers

Non-adaptive PID controllers use a fixed PID controller for the entire control period and rely on its robustness to maintain control performance. For each individual PID controller (with its transfer function found in the previous subsection for $t = 1$, 20, 40, 60, 90, 120, 150 and 182 min) the simulation results on $G(t)$ and $u(t)$ are shown below in **Figures 8-15**.

Under the individual PID controllers, the output $G(t)$, the glucose level, did not really meet the design specification, and the glucose level is not near or at least in a small neighborhood of the glucose basal level. The overshoot of the system was too high and beyond the acceptable level. Also the settling time was not even close to where it should be as per the design requirement. And the steady state error was not satisfactory. A new method should be developed and implemented to meet all the design specifications. This method is explained in detail in the following section.

4.3. Switched PID Controllers

The individual PID controllers could not lower the glucose level $G(t)$ of the patient to the neighborhood of the glucose basal level. Consequently, we introduce a new control-switching scheme that adapts controllers to meet design specifications. The switching control consists of the following items: One "time clock," one "switch case" block, one "if action case" block, one "merge" block, and eight "off-on switches." All these blocks are connected together to form the wiring diagram of the Control-Switching Scheme.

The functions of the control switching scheme is detailed in **Figure 16**. The "Time Clock" is to provide the "Switch Case" block with time as a signal input to activate it.

The "Switch Case" block receives a single input from the clock, which it uses to form case conditions that determine which subsystem to execute. Each output port case condition is attached to a Switch Case Action sub-
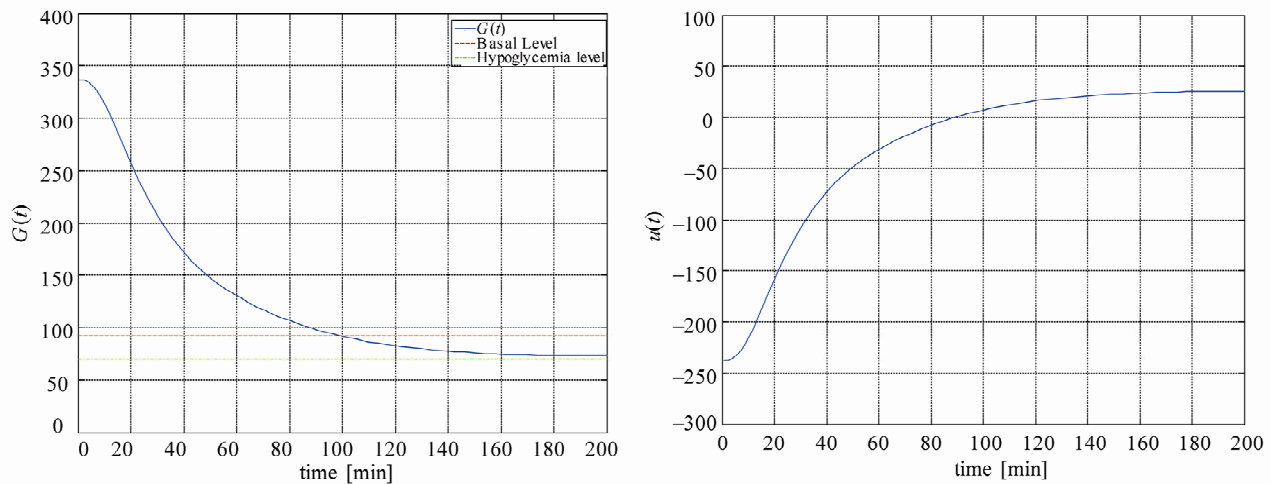
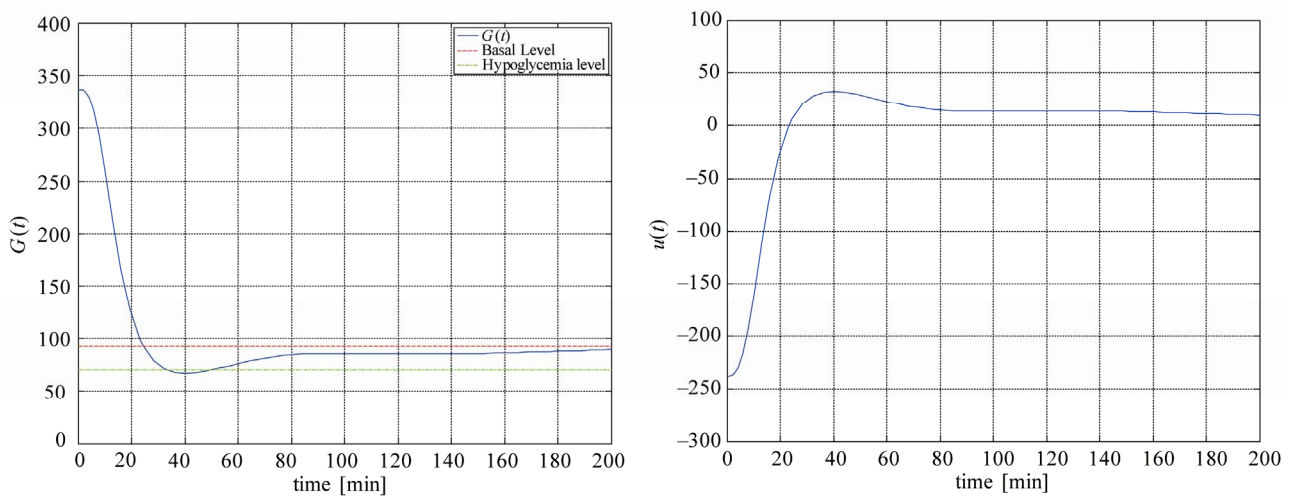**Figure 8.** $G(t)$ and $u(t)$ using G_C^1.
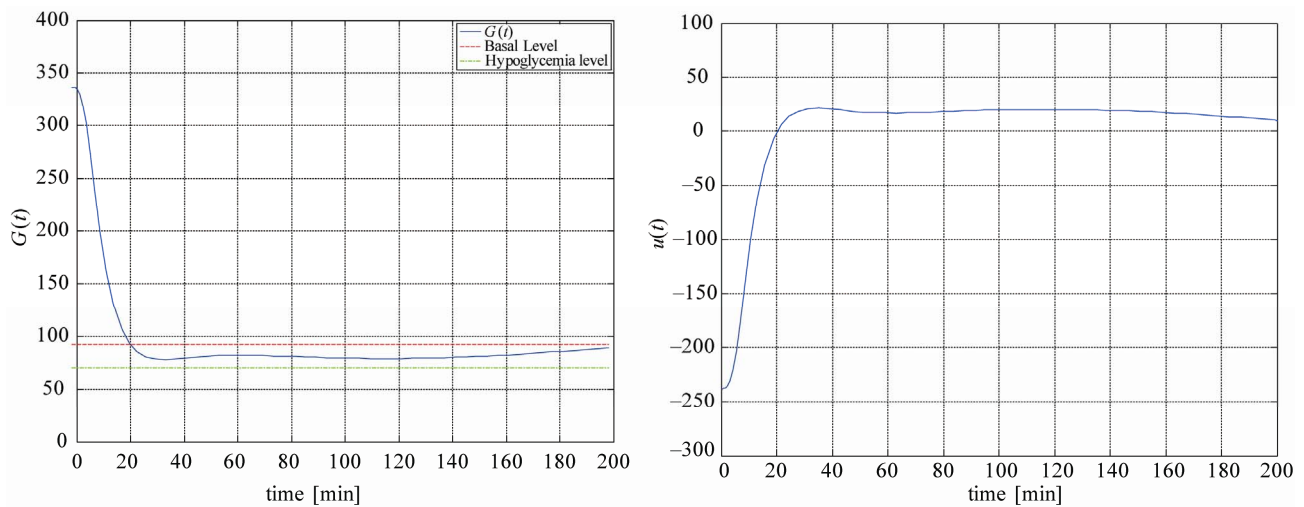

Figure 9. $G(t)$ and $u(t)$ using G_C^{20}.

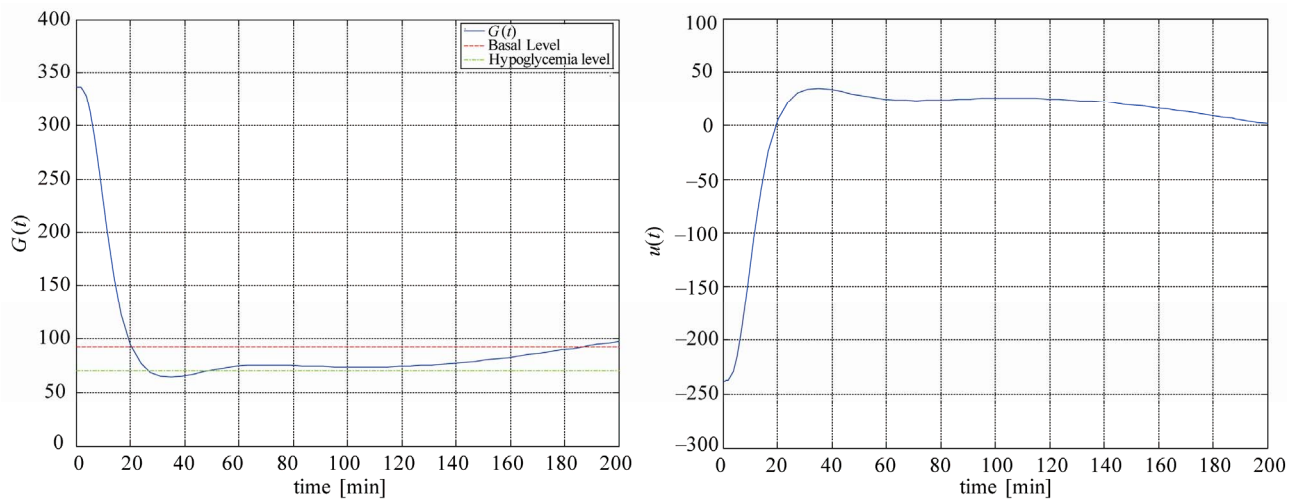**Figure 10.** $G(t)$ and $u(t)$ using G_C^{40}.

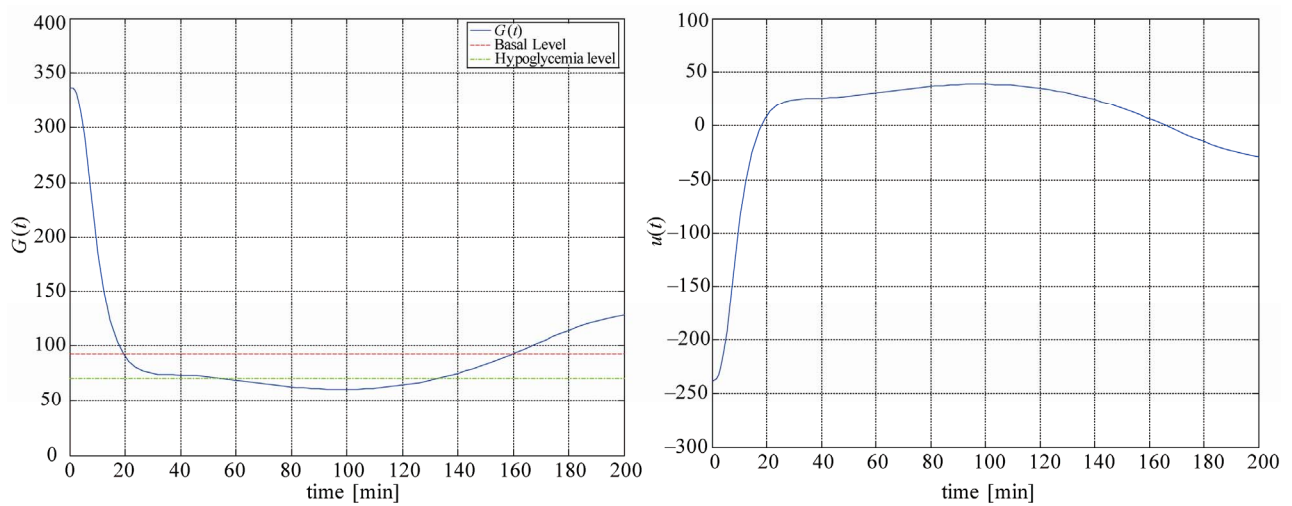**Figure 11.** $G(t)$ and $u(t)$ using G_C^{60}.

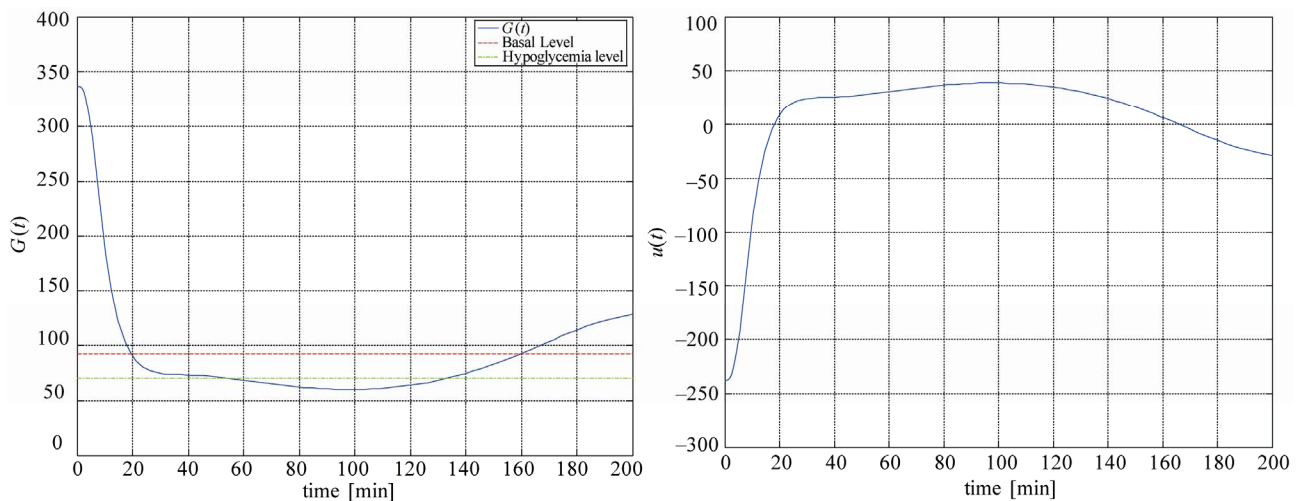**Figure 12.** $G(t)$ and $u(t)$ using G_C^{90}.

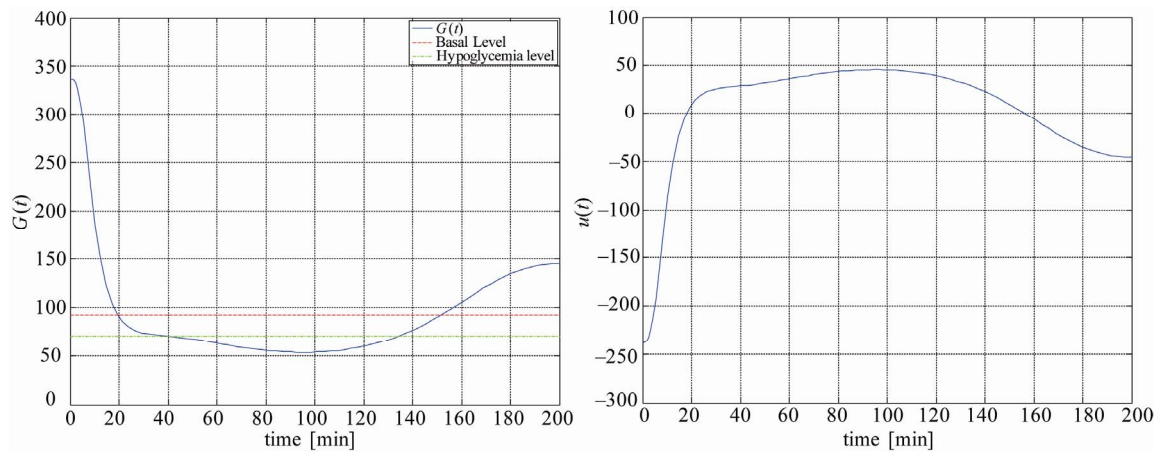**Figure 13.** $G(t)$ and $u(t)$ using G_C^{120}.

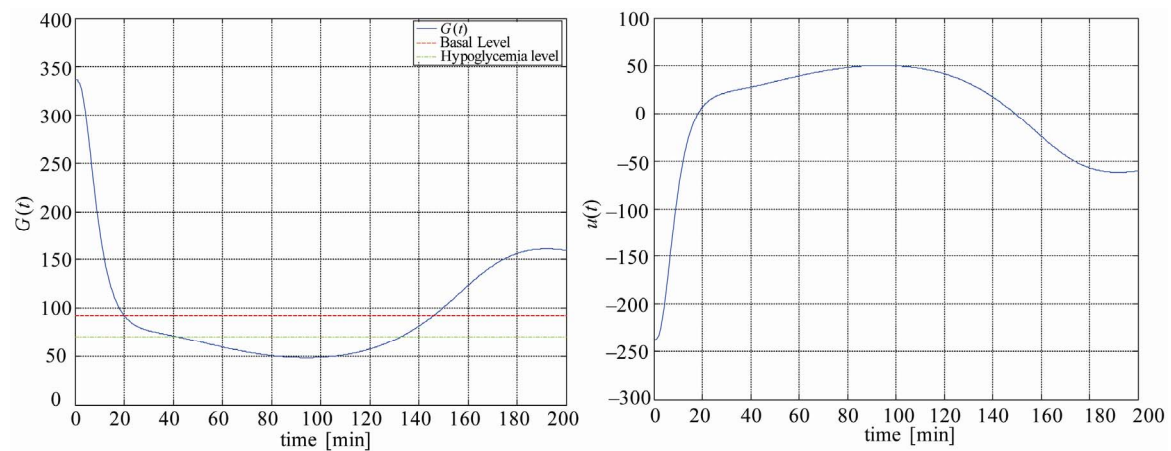**Figure 14.** $G(t)$ and $u(t)$ using G_C^{150}.
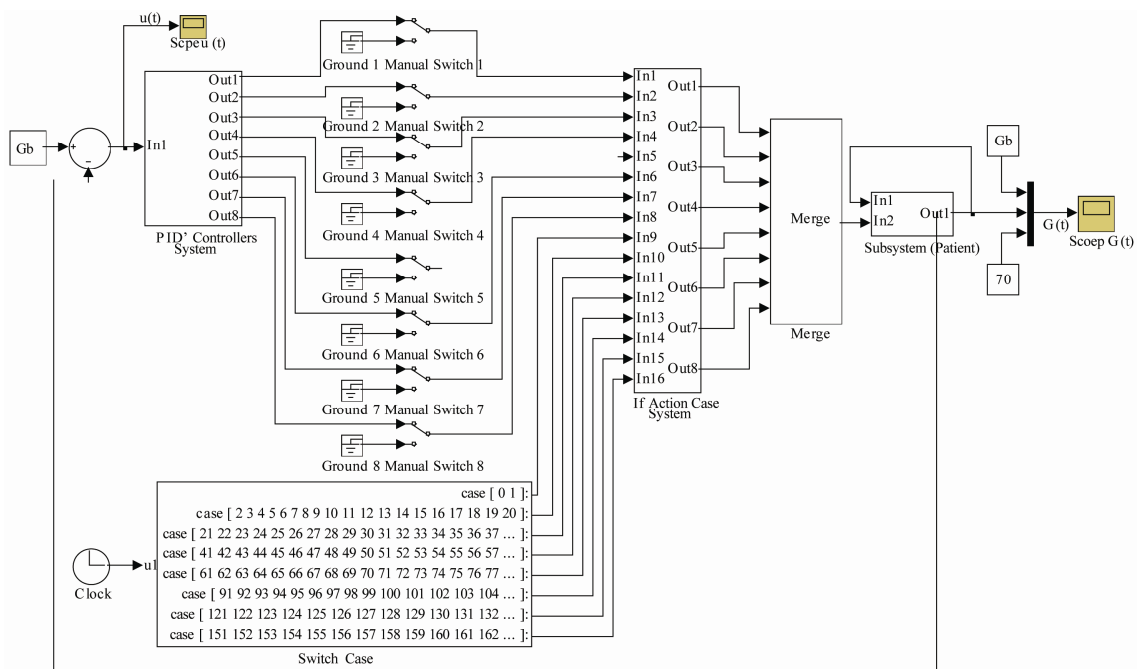

Figure 15. $G(t)$ and $u(t)$ using G_C^{182}.

Figure 16. Control-Switching Scheme Simulation Diagram.

system. The cases are evaluated top-down, starting with the top case. If a case value corresponds to the actual value of the input, its Switch Case Action subsystem is executed. The "Switch Case" model is divided into eight time interval zones as shown in **Table 6**.

The "If Action Case" block consists of eight "If- Action Subsystem." The "If-Action Case" implements Action Subsystems used in if-statement and switch control flow statements. Action Subsystems execute their programming in response to the conditional outputs of an If-statement or Switch Case block. A schematic diagram of the "If Action Case" block is shown in **Figure 17**. The "Merge" block combines its inputs into a single output line whose value at any time is equal to the most recently computed output of its driving blocks. The number of inputs can be specified by setting the block's inputs parameter. The "Off-On Switches" are used to turn the PID controllers OFF or ON.

In general when the time clock is running, it feeds the "Switch Case" block with an input signal which in turn switches on the "If-Action Case" block as per the time interval that was specified in **Table 6**. Based on the status of the "If-Action Case", a specific PID controller will be turned on and executed to control the output of the system.

For zone 1, the time interval is between 0 - 1 minute, during this period of time the "Switch Case" is enabling input "In9" of the "If-Action Case". When input "In9" is enabled, it will only execute the input "In1" to the output "Out1". The input "In1" is connected to the first PID Controller. That means only the first PID Controller, (PID Controller 1), is working. At the end of the first minute, the "Switch Case" will switch to zone 2 which runs from the beginning of the minute number 2 and will last until the end of the minute number 20. During this period of time, the "Switch Case" is enabling input "In10" of the "If-Action Case". When input "In10" is enabled, it will only execute the input "In2" to the output "Out2". The input "In2" is connected to the second PID Controller. That means only the second PID Controller, (PID Controller 20), is working. The same procedure will be followed until the "Switch Case" switches between the eight time zones that were specified in **Table 6**. In turn the PID Controllers will be executed based on the status of the "If-Action Case".

The Control-Switching Scheme Diagram shown in **Figure 16** was simulated with all the PID controllers executed (connected to the circuit). The output $G(t)$ of the system is shown in **Figure 18**. It can be clearly seen that the PID controllers are able to bring the glucose level from 337 mg/dL to the basal level (99 mg/dL) within 40 minutes. But in about 70 minutes the value of the glucose starts going below the basal level, and it

Table 6. Swithing Time Interval.

Zone number	Time Interval (minutes)
1	0 - 1
2	2 - 20
3	21 - 40
4	41 - 60
5	61 - 90
6	91 - 120
7	121 - 150
8	150 - 182

went further below the minimum value that the glucose can be at. In this case the person will be classified as a patient with the hypoglycemia (low sugar), and that is not acceptable.

The Control-Switching Scheme Diagram was then simulated in which all the PID controllers are executed except the eighth PID Controller G_C^{182}. The graph of the output, $G(t)$ of the system is shown in **Figure 19**. It can be seen the same problem still exists. Again in this case the person will be classified as a patient with the hypoglycemia.

The same procedure was repeated but with all the PID controllers executed except the PID Controllers G_C^{150} and G_C^{182} with the graph of the output shown in **Figure 20**; and excluding G_C^{120}, G_C^{150}, and G_C^{182} with the simulation result shown **Figure 21**. Again in both cases the person will be classified as a patient with the hypoglycemia.

When we exclude G_C^{90}, G_C^{120}, G_C^{150}, and G_C^{182}, the output $G(t)$ of the system, shown in **Figure 22**,. reaches the glucose basal level (99 mg/d/L) within 40 minutes, and it stays in that neighborhood. The input of the PID Controller system is shown in **Figure 23**.

For verification, the same control strategy is evaluated on patient #2. Following the same modeling procedure that was performed for the diabetic patient #1, the model parameters are identified as $P_1 = 0$, $P_2 = 0.42/100$, $P_3 = 2.56/1000000$, $I_0 = 209$, $G_0 = 297$, $\gamma = 3.72/1000$, $h = 154$, $n = 0.22$, $a = 2$, $G_b = 100$, $I_b = 8$, [10].

Without control, the above data was implemented in model simulation. The output of the simulation diagram is shown in **Figure 24**, which shows that without proper control the glucose level does not come down to the basal level after injecting an amount of 297 mg/dL of glucose inside a diabetic patient. The level of the glucose inside a diabetic patient decreases for the first 120 minutes and starts increasing afterward and reaches the value of about 270 mg/dL after 3 hours from the time the glucose was injected.

The same control-switching scheme that was performed for diabetic patient #1 is repeated for diabetic patient #2. The values of the parameters for the first four PID controllers (at t =1, 20, 40 and 60 minutes) are
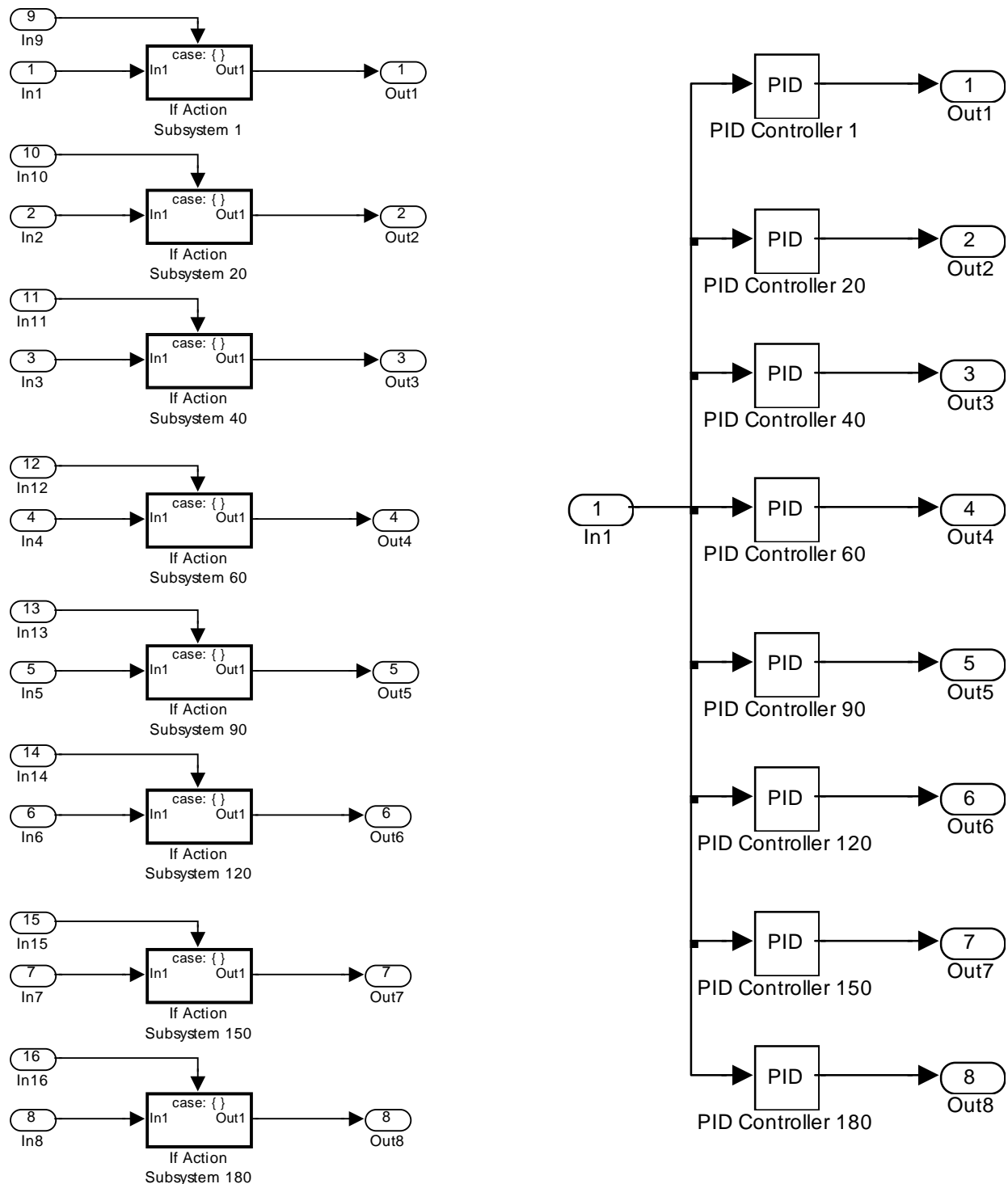
Figure 17. "If Action Case" system and PID controller switching function module.

summarized in **Table 7**.

The control-switching scheme was simulated for diabetic patient #2 by using only the first four PID controllers. The output $G(t)$ of the system is shown in **Figure 25**. The output $G(t)$ reaches the glucose basal level

(100 mg/d/L) within 60 minutes and it stays in that neighborhood. The graph of the input of the PID Controller system is shown in **Figure 26**.

Based on the simulation results, although adaptive control can potentially improve control performance, it is
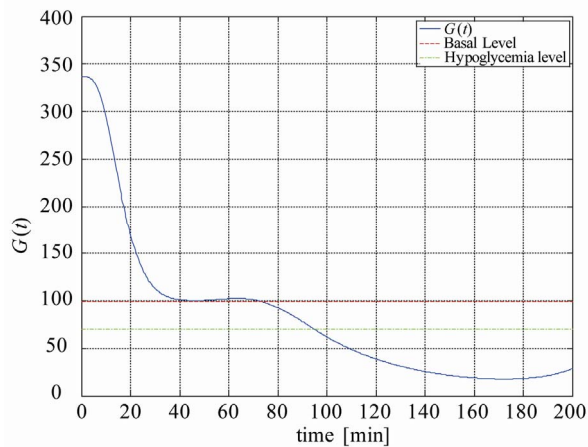
Figure 18. Plot of G(*t*) when all PID Controllers are executed.
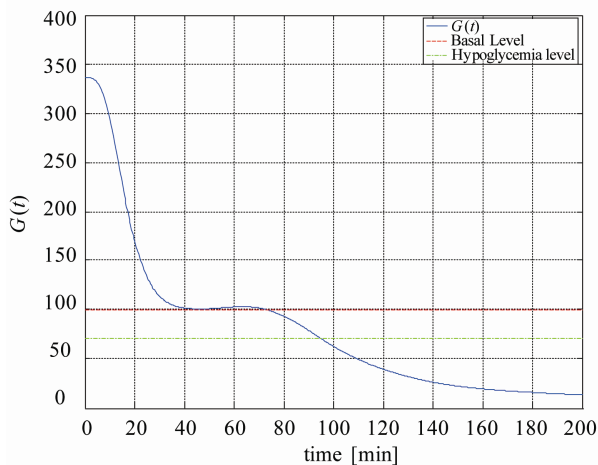
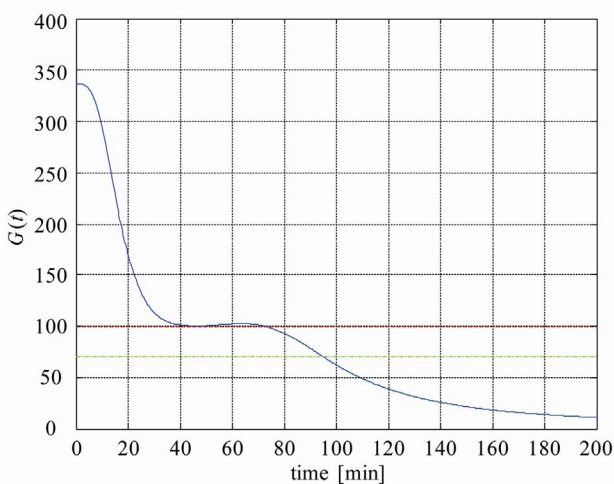**Figure 19.** Plot of G(*t*) when all PID Controllers except controller G_C^{182} are executed.

Figure 20. Plot of G(*t*) when all PID Controllers except controller G_C^{150} and G_C^{182} are executed.
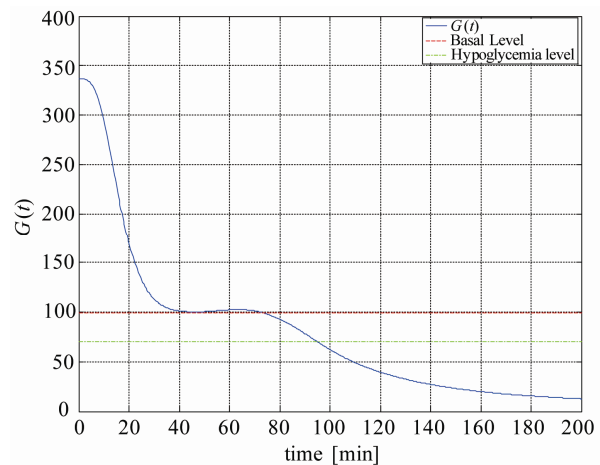

Figure 21. Plot of G(*t*) when all PID Controllers except controller G_C^{120}, G_C^{150} and G_C^{182} are executed.
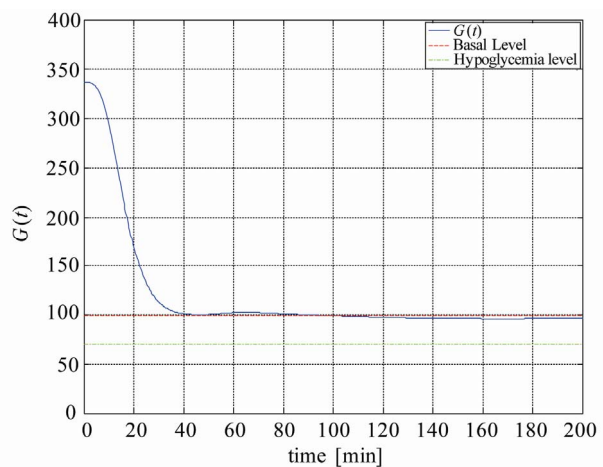

Figure 22. Plot of G(*t*) when all PID Controllers except controller G_C^{90}, G_C^{120}, G_C^{150} and G_C^{182} are executed.
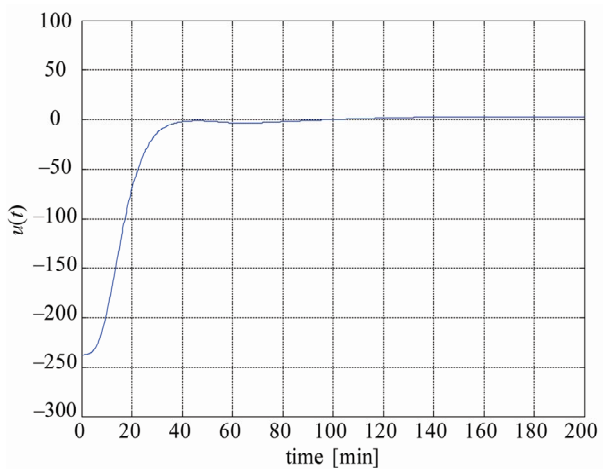

Figure 23. Plot of *u*(*t*) when all PID Controllers except controller G_C^{90}, G_C^{120}, G_C^{150} and G_C^{182} are executed.
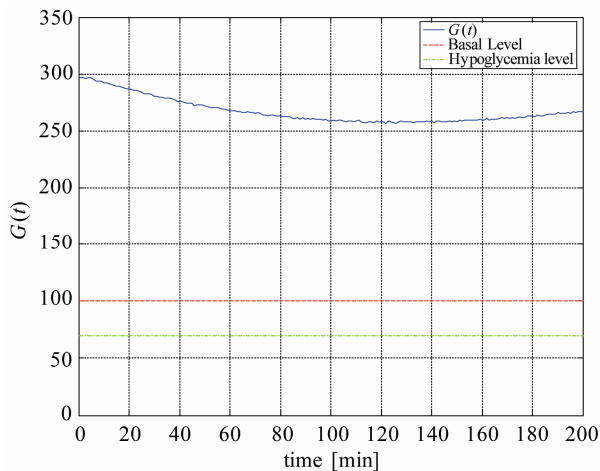
Figure 24. Output of the Simulation Diagram for diabetic patient #2.
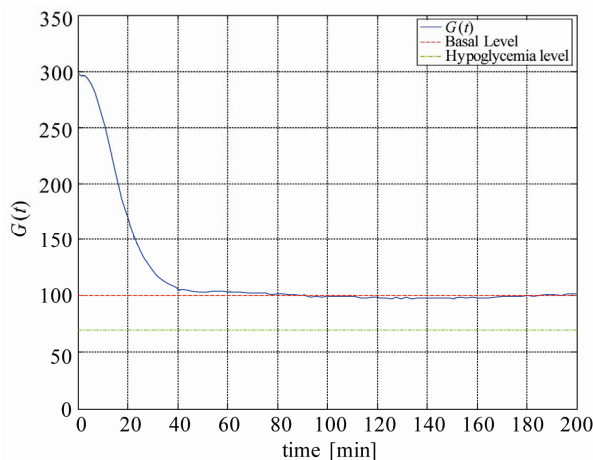

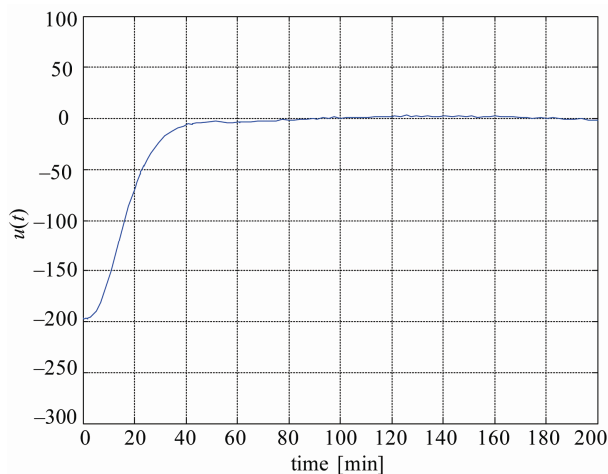**Figure 25.** Plot of $G(t)$ when PID Controllers 1, 20, 40 and 60 are executed.

Figure 26. Plot of $u(t)$ when PID Controllers 1, 20, 40 and 60 are executed.

Table 7. PID Controller for diabetic patient #2.

	PID Controllers Parameters			
	PID at $t = 1$ **min**	**PID at** $t = 20$ **min**	**PID at** $t = 40$ **min**	**PID at** $t = 60$ **min**
K_p	0.0412	0.1069	0.0868	0.0768
K_i	2.7402×10^{-4}	0.0047	0.0086	0.014
K_d	10.1	30.6	33.1	33.6

sometimes unnecessary, or even harmful when switching overly frequently. Our results show that when the switching scheme is limited to the first four PID controllers, the performance is in fact enhanced. This may be related to the fact that some PID controllers are more robust with respect to the model variations. On the other hand, in comparison to individual controllers, the control-switching scheme achieves design specification while all individual controllers fail to deliver the required performance.

5. CONCLUSIONS

This study reveals that typical PID controllers may not be sufficient to deliver satisfactory control performance in glucose level control problems. This is mainly due to the nonlinear nature of patient dynamic models and limited robustness of the PID controllers. An adaptive control that switches controllers based on operating conditions can potential enhance control performance. However, the switching control scheme must be carefully designed to ensure that control specifications be met. Our results show that overly frequent switching of controllers may have detrimental effects on control performance. We show that by reducing the number of PID controllers in the switching scheme, not only control complexity is reduced, but performance is actually enhanced.

REFERENCES

[1] Karam, J.H., Grodsky, G.M. and Forsham, P.H. (1963) Excessive insulin response to glucose in obese subjects as measured by immunochemical assay. *Diabetes*, **12**, 196-204.

[2] Ginsberg, H., Olefsky, J.M. and Reaven, G.M. (1974) Further evidence that insulin resistance exists in patients with chemical diabetes. *Diabetes*, **23**, 674-678.

[3] Reaven, G.M. and Olefsky, J.M. (1977) Relationship between heterogeneity of insulin responses and insulin resistance in normal subjects and patients with chemical diabetes. *Diabetologia*, **13**, 201-206. doi:10.1007/BF01219700

[4] Lerner, R.L. and Porte, D. (1972) Acute and steady state insulin responses to glucose in nonobese, diabetic subjects. *Journal of Clinical Investigation*, **51**, 1624-1631. doi:10.1172/JCI106963.

[5] Reaven, G.M. (1980) Insulin-independent diabetes mellitus: Metabolic characteristics. *Metabolism Clinical and Ex-*

perimental, **29**, 445-454.
doi:10.1016/0026-0495(80)90170-5

[6] Bergman, R.N. and Cobelli, C. (1980) Minimal modeling, partition analysis, and the estimation of insulin sensitivity. *Federation Proceedings*, **39**, 110-115.

[7] Shen, S-W., Reaven, G.M. and Farquhar, J.W. (1970) Comparison of impedance to insulin-mediated glucose uptake in normal and diabetic subjects. *Journal of Clinical Investigation*, **49**, 2151-2160.
doi:10.1172/JCI106433.

[8] Bergman, R.N., Ider, Y.Z., Bowden, C.R. and Cobelli, C. (1979) Quantitative estimation of insulin sensitivity. *American Journal of Physiology*, **236**, E667-E677.

[9] Pacini, G. and Bergman, R.N. (1986) MINMOD, A computer program to calculate insulin and pancreatic responsivity from the frequently sampled intravenous glucose tolerance test. *Computer Methods and Programs in Biomedicine*, **23**, 113-122

[10] Bergman, R.N., Phillips, L.S. and Cobelli, C. (1981) Physiologic evaluation of factors controlling glucose tolerance in man, measurement of insulin sensitivity and β-cell glucose sensitivity from the response to intravenous glucose. *Journal of Clinical Investigation*, **68**, 1456-1467. doi:10.1172/JCI110398

[11] Bergman, R.N. and Urquhart, J. (1971) The pilot gland approach to the study of insulin secretory dynamics. Recent Progress in Hormone Research, **27**, 583-605.

[12] Nomura, M., Shichiri, M., Kawamori, R., Yamasaki, Y., Iwama, N. and Abe, H. (1984) A mathematical insulin-secretion model and its validation in isolated rat pancreatic islets perifusion. *Computers and Biomedical Research*, **17**, 570-579.
doi:10.1016/0010-4809(84)90021-1

[13] Buchanan, T.A., Metzger, B.E., Freinkel, N. and Bergman, R.N. (1990) Insulin sensitivity and B-cell responsiveness to glucose during late pregnancy in lean and moderately obese women with normal glucose tolerance or mild gestational diabetes. *American Journal of Obstetrics and Gynecology*, **162**, 1008-1014.

[14] Lynch, S.M. and Bequette, B.W. (2001) Estimation-based model predictive control of blood glucose in type i diabetics: A simulation study. *IEEE Transations on Biomedical Engineering conferences*, 79-80.

[15] Fisher, M.E. (1991) A semi-closed loop algorithm for the control of blood glucose levels in diabetes. *IEEE Transations on Biomedical Engineering conferences*, 57-61.

[16] Furler, S.M., Kraegen, E.W., Bell, D.J., Smallwood, R.H., and Chisolm, D.J. (1985) Blood glucose control by intermittent loop closure in the basal mode: Computer simulation studies with a diabetic model. *Diabetes Care*, **8**, 553-561.

[17] Ibbini, M.S., Masadeh, M.A. and Amer, M.M.B. (2004) A semiclosed-loop optimal control system for blood glucose level in diabetics. *Journal of Medical Engineering*

& Technology, **28**, 189-196.
doi:10.1080/03091900410001662332.

[18] Furler, S.M., Kraegen, E.W., Smallwood, R.H. and Chisholm, D.J. (1985) Blood glucose control by intermittent loop closure in the basal mode: computer simulation studies with a diabetic model. *Diabetes Care*, **8**, 553-561. doi:10.2337/diacare.8.6.553

[19] Insel, P.A., Liljenquist, J.E., Tobin, J.D., Sherwin, R.S., Watkins, P., Andres, R. and Berman, M. (1975) Insulin control of glucose metabolism in man. *Journal of Clinical Investigation*, **55**, 1057-1066.
doi:10.1172/JCI108006

[20] Grodsky, G.M. (1972) A threshold distribution hypothesis for packet storage of insulin and its mathematical modeling. *Journal of Clinical Investigation*, **51**, 2047-2059.
doi:10.1172/JCI107011

[21] Sherwin, R.S., Kramer, K.J., Tobin, J.D., Insel, P.A., Liljenquist, J.E., Berman, M. and Andres, R. (1974) A model of the kinetics of insulin in man. *Journal of Clinical Investigation*, **53**, 1481-1492.

[22] Turner, R.C., Holman, R.R., Mathews, D., Hockaday, T.D.R. and Peto, J. (1979) Insulin deficiency and insulin resistance interaction in diabetes: Estimation of their relative contribution by feedback analysis from basal insulin and glucose concentrations. *Metabolism Clinical and Experimental*, **28**, 1086-1096.
doi:10.1016/0026-0495(79)90146-X

[23] Lerner, R.L. and Porte, D. (1971) Relationships between intravenous glucose loads, insulin responses and glucose disappearance rate. *Journal of Clinical Endocrinology & Metabolism*, **33**, 409-417

[24] Bolie, V.W. (1961) Coefficients of normal blood glucose regulation. *Journal of Applied Physiology*, **16**, 783-788.

[25] Guyton, A.C. and Hall, J.E. (1996) Text book of medical physiology, 9th Edition, Saunders.

[26] The American Diabetes Association is a US leading non-profit health organization providing diabetes research, information and advocacy since 1940.

[27] Gaetano, A.D. and Arino, O. (2000) Mathematical modeling of the intravenous glucose tolerance test. *Journal of Mathematical Biology*, **40**, 136-168.
doi:10.1007/s002850050007

[28] Marquardt, D.W. (1963) An algorithm for least-squares estimation of non-linear parameters. *Journal of the Society for Industrial and Applied Mathematics*, **11**, 431-441.
doi:10.1137/0111030

[29] Goodwin, C. and Payne, R.L. (1977) Dynamic System Identification. Academic Press, Inc., New York.

[30] Lourakis, M. (2005) A brief description of the levenberg-marquardt algorithm implemented by levmar. Institute of Computer Science, Foundation for Research and Technology.

[31] De Groot, M.H. and Schervish, M.J. (2002) Probability and statistics. 3rd Edition, MA: Addison-Wesley, Boston.