FPGA IMPLEMENTATION OF SECOND ORDER SLIDING MODE BASED ON ST ALGORITHM FOR TYPE I DIABETES

* Eman Hassony Jadoo¹

Dr. Ekhlas Hameed Karam²

- 1) M.Sc. Student, Computer Engineering Department, Mustansiriyah University, Baghdad, Iraq.
2) Assistant Prof., Computer Engineering Department, Mustansiriyah University, Baghdad, Iraq.

Abstract: Diabetes is an increasing worldwide health problem. In particular, patients with type 1 diabetes who require strict control of blood glucose because they have insulin production insufficient. In this paper, the Second Order Sliding Mode Controller (SOSMC) depending on Super Twisting Algorithm (STA) has suggested to control the blood glucose levels of humans and regulate it. The Bergman Minimal (BEM) is the mathematical model explaining the dynamics behavior of blood glucose concentration. The parameters of the suggested SOSMC have optimized by using Chaotic Particle Swarm Optimization (CPSO) algorithm. The simulation part has done to validate the efficiency of the suggested controller depending on MATLAB / Simulink. Also, the SOSMC controller scheme has implemented by Field Programmable Gate Array (FPGA) and downloaded to the Spartan-6 (XC6SLX45T-3FGG484) board. The performance results obtained for the suggested controller in calculating the insulin infusion intensity for different patients indicate its efficacy and robustness in controlling blood glucose levels for type 1 diabetes.

Keywords: *Type I Diabetes, Second Order Sliding Mode Control, Chaotic Particle Swarm Optimization, BEM model, FPGA.*

1. Introduction

Diabetes is a medical condition triggered by an existing high blood glucose level for a long

amount of time leading to inadequate insulin supply in the blood [1].

The pancreatic beta-cells of the human body are essential for producing insulin that controls glucose intake. In diabetes, beta cells fail to generate adequate insulin production in the blood of the human body and are unable to regulate the level of glucose. Diabetes is divided into two principal types. Type 1 diabetes mellitus (T1DM) and Type 2 diabetes mellitus (T2DM) in the first type, the patient's body can't generate sufficient amount of insulin, so doses of insulin have to be administered into the human body to control blood glucose rates, whereas the second type starts with insulin resistance, a disease in which cells do not react adequately to insulin. This model is known as noninsulin-dependent [2]. Blood Glucose Level (BGL) in a healthy human body's blood is within (70-110 mg / dL) ranges. Diabetes can be identified in a human body if the natural relationship of glucose and insulin is not regulated by the body. Therefore, blood glucose levels need to be regulated by injecting insulin [3]. The mutual way to prevent the effects of high blood glucose levels are to insulin injection

*Corresponding Author: emanhassony06@gmail.com

at a time while there has expected to be an increase in blood glucose. The development known as an artificial pancreas is a closed-loop glucose regulator system, in general, has three main components, a glucose monitor, the pump of insulin, and the control techniques that deliver the necessary insulin dose according to glucose measurement [4]. Figure 1 shows the closed-loop control system for the level of blood glucose.

Several studies use a closed-loop controller to keep blood glucose diabetic concentration within the appropriate range, such as Kaveh and Shtessel used high order sliding mode controller (HOSMC) to regulate the levels of blood glucose [5]. Garcia-Gabin et al. presented the new technique for controlling of the glucose in a plasma composed of a sliding mode controller for feedback and meal compensation for feed forward [6]. In a [7] a hybrid of robust SMC and Smith predictor (SP) configurations have been used. Regulating blood glucose level of type 1 diabetic by using FPGA based digital PID in [8]. Nasrin et al. suggest a Sliding Mode Control (SMC) based on BackStepping technique [9]. As well as the fractional backstepping sliding mode control to regulate the level of blood glucose in patients with type 1 diabetes [10].

Figure 1. Block diagram for insulin distribution mechanism in closed loop.

In this paper, the SOSMC has used for regulating the levels of glucose, the CPSO

algorithm was used to improve the controller parameters and then, the implementation of the FPGA is applied using an FPGA Spartan 6 Package. The FPGA has been chosen as the better choice to provide potential substitutes to speed up hardware implementation. The paper has organized as follows: BEM mathematical model of blood glucose system presented in section 2. The details of the SOSMC described in section 3, the CPSO algorithm illustrated in section 4. FPGA implementation will explain in Section 5. The simulation results have explained in section 6 then the conclusions are listed in the last section.

2. Bergman Glucose-Insulin Regulation Model

Specific mathematical models been suggested to explain the complexities of diabetes, and to compare the interaction between glucose models and insulin delivery that helps design a model for diabetes. Among these models, a common reference model in the literature, the minimal Bergman model approaches the dynamic response of blood glucose concentration in a diabetic to insulin injections. The Bergman model is composed of three differential equations [8, 9]:

$$\begin{aligned}\dot{G}(t) &= -p_1(G(t) - G_b) - X(t)G(t) + D(t) \\ \dot{X}(t) &= -p_2X(t) + p_3(I(t) - I_b) \\ \dot{I}(t) &= -n(I(t) + I_b) + \gamma[G(t) - h]^+ t + u(t)\end{aligned}\quad (1)$$

Where $G(t)$ is the concentration of glucose in plasma [mg/dL], $X(t)$ proportional to the concentration of insulin in remote compartment [1/min], $I(t)$ is the concentration of insulin in plasma [mU/dL], and $u(t)$ is rate of insulin injected [mU/min] the control signal, $(p_1, p_2, p_3, n, h, \gamma)$ are model parameters. G_b, I_b are the “basal” base line or steady state

value of blood glucose and the basal value for insulin respectively.

In the last equation of this model, the term $\gamma[G(t) - h]^+$ is the internal regulatory function for the formulation of insulin secreted in the body, and this has not found in diabetics. The value of p_1 will be greatly reduced because there is no produce insulin, it can approach to zero [8]. $D(t)$ Is disturbance signal (meal disturbance) can be modeled by a decaying exponential function of the following form [11]:

$$D(t) = A \exp(-B(t - t_{meal})) \quad (2)$$

Where B represents a rate of the meal absorption, A is size of the meal and t_{meal} represents the starting period of the digestion of the meal.

3. Sliding Mode Controller Design Based on ST Algorithm

SMC is the robust and simple method of synthesizing controllers depending on the concepts of Variable Structural Control (VSC) for linear and nonlinear processes.

The discrete control has a high transformation frequency, which causes a "chattering-phenomenon," which considers unwanted property that occurs in the control operation of SMC [12]. Different methods have used to overcome the chattering phenomena like replacing the $\text{sign}(s)$ by boundary function like $\text{sat}(s)$, using terminal SMC, integral SMC, and other different methods. One of the most effective methods to work around this problem is using a Second Order Sliding Mode Control. There are various SOSMC algorithms, like the Twisting (TW), Sub-Optimal (SO), and Super-Twisting (ST). ST algorithm does not need a differentiator to obtain the first sliding variable derivative needed to find the control output for the algorithms [12]. The ST used design steps

similar to the classical SMC. It applies the same sliding surface as in Eq. (3); The sliding surface could introduce in the following:

$$s(t) = \gamma e(t) - \dot{e}(t) \quad (3)$$

Where $e(t)$ and $\dot{e}(t)$ are error and derivative of the error respectively, $e(t)$ is given by:

$$e(t) = r(t) - y(t) \quad (4)$$

Where $r(t)$ is the reference input (Basal Value) and $y(t)$ is the output signal (measured glucose)

The constant γ has chosen to be positive. The choice of γ decides the convergence rate of the tracking error.

The ST algorithm has represented as follows [12, 13]:

$$u = u_1 + u_2, \quad (5)$$

$$u_1 = -a_1 \text{sign}(s) \quad (6)$$

$$u_2 = -a_2 |s|^{0.5} \text{sign}(s), \quad (7)$$

Where a_1 and a_2 are positive bounded constants. The control law of the super twisting SOSMC is given by:

$$u(t) = -a_1 \text{sign}(s(t)) - a_2 |s(t)|^{0.5} \text{sign}(s(t)) \quad (8)$$

Figure 2 show the block diagram for the SOSMC based on super twisting algorithm.

Figure 2. The SOSMC block diagram.

3.1. Chaotic Particle Swarm Optimization (CPSO) Algorithm

The common evolutionary techniques are the Particle Swarm Optimization (PSO) algorithms, which is to adopt a random sequence for their parameter. A population of candidate solutions has initialized with the PSO algorithm and is called a particle. For a D-dimensional search space, N particles travel around the problem [14].

The position of the i^{th} particle at the i^{th} iteration is represented by

$x_i(t) = (x_{i1}, x_{i2}, \dots, x_{iD})$. The particle velocity for the i^{th} may be described as

$$v_i(t) = (v_{i1}, v_{i2}, \dots, v_{iD}).$$

The best position that has so far been visited by the i^{th} particle is represented as $p_i = (p_{i1}, p_{i2}, \dots, p_{iD})$ which is also called pbest. The global best position attained by the whole swarm is called the global best (gbest) and represented as $p_g(t) = (p_{g1}, p_{g2}, \dots, p_{gD})$.

The velocity vector at the i^{th} iteration has represented as $v_i(t) = (v_{g1}, v_{g2}, \dots, v_{gD})$. At the next iteration, the velocity and position of the particle have calculated according to (11, 12)

$$v_i(t+1) = wv_i(t) + c_1r_1(pbest_i(t) - x_i(t)) + c_2r_2(gbest_i(t) - x_i(t)) \quad (11)$$

$$x_i(t+1) = x_i(t) + v_i(t) \quad (12)$$

Where w is inertia weight factor, c_1, c_2 are called acceleration coefficients, r_1, r_2 are random value in the range $[0,1]$. The parameters w, r_1 , and r_2 are the key factors that affected the convergence behavior [15]. In the CPSO algorithm the parameters c_1 and c_2 are modified by using the logistic map based on the following equation:

$$M(t+1) = \mu \times M(t) \times (1 - M(t)) \quad 0 \leq M(t) \leq 1 \quad (13)$$

Where $M(1)$ generated randomly for each independent run and $\mu = 4$. Then, add a new modification of velocity as in equations (14).

$$v_i(t+1) = wv_i(t)M(t) \times r_1(pbest_i(t) - x_i(t)) + (1 - M(t)) \times r_2(gbest_i(t) - x_i(t)) \quad (14)$$

The important advantages of the chaotic optimization algorithm (COA) are summarized as Easy to implement, short run time, and speed of search [16]. The flowchart that represented this algorithm has illustrated below.

Figure 3. General flowchart of the CPSO algorithm.

3.2. FPGA Implementation

The design of the suggested controller using the Xilinx 14.6i architecture set, it is implemented on Spartan-6 (XC6SLX45T-3FGG484) and the hardware implementation is dependent on the Xilinx System Generator (XSG) tool which makes the software of MATLAB adopted an environment that enables the Simulink to be

easily used in the implementation of any FPGA kit system. Then when you begin using XSG, you need to identify the boundary of the FPGA dependent architecture using the "In" and "Out" gateways. The block function "Gateway In" is to convert floating-point representations into fixed-point representations, so that FPGA can process them [17]. The "Gateway Out" block does the same, translation of FPGA data into floating-point or other MATLAB-readable formats. In this controller, a fixed point sign number of 10 bits with 3 bits binary point (fix_10_3) are used to represent the input and the output signals represented by (fix_57_42). Figure 4 and Figure 5 show the device utilization summary for the FPGA design with 1% utilization of hardware platform slices respectively.

Figure 4. FPGA design of SOSMC.

Size Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Registers	438	54,376	1%	
Number used as Flip-Flops	438			
Number used as Latches	0			
Number used as Latch-thrus	0			
Number used as AND/OR logic	3			
Number of Slice LUTs	2,871	23,388	12%	
Number used as logic	2,772	23,388	12%	
Number using OR output only	1,278			
Number using OR output only	94			
Number using OR and OR	465			
Number used as ROM	0			
Number used as Memory	27	6,488	1%	
Number used as Dual-Port RAM	0			
Number used as Single-Port RAM	0			
Number used as Shift Register	27			
Number using OR output only	3			
Number using OR output only	0			
Number using OR and OR	24			
Number used exclusively as route-thrus	72			
Number with same-size register load	30			
Number with same-size carry load	42			
Number with other load	0			
Number of occupied Slices	885	6,822	12%	
Number of MUXes used	2,660	13,644	19%	
Number of LUT Flip-Flop pairs used	3,108			

Figure 5. The Device Utilization Summary of the SOSMC.

4. Simulation Results

The results of simulations using BEM model as addressed in Eq. (1), the model parameters have listed in Table 1 and the suggested controller based on the CPSO algorithm is offered in this section for a BGL of 70 mg/dl. The BEM model response without controller is shown in Figure 6.

Table1. Parameters for the Bergman Minimal Model [5,6].

Parameter	Normal	Patient1	Patient2	Patient3
p_1	0.0317	0	0	0
p_2	0.0123	0.02	0.0072	0.0142
p_3	4.92	5.3×10^{-6}	2.16×10^{-6}	9.94×10^{-5}
n	0.2659	0.3	0.2465	0.2814
γ	0.0039	-	-	-
h	79.0353	-	-	-
G_b	70	70	70	70
I_b	7	7	7	7

In this paper, The simulations are performed dynamically for three patients with different average weight and initial conditions of 220, 200, and 180mg / dL, respectively for patients 1, 2, and 3. The meal disturbance in a simulation as provided in Eq.(2) the value of its parameters are $A = 0.5$, $b = 0.05$, and $t_{meal} = 400\text{min}$.

Figure 6. Disturbance meal signal (exogenous glucose infusion).

Figure 7. Glucose output of three patients and healthy person with disturbance (open-loop glucose regulatory system).

The simulation results show that the normal person's glucose level is stabilized at the basal level in the presence of the meal disturbance, while the patient's level of glucose remains dangerous outside the range. The simulation second part is the suggested controller applied to the system and a patient is checked in the presence of the disturbance (meal). Three sets of

parameters for three separate patients were used to test the robustness of the control algorithm to adjust the parameter. The CPSO algorithm parameters are listed here, as seen in Table 2.

Table 2. The parameters of CPSO algorithm.

CPSO Parameters	Acronym	Value
Number of particles	pop_size	20
Max. Number of Iterations	$Iter_{max}$	80
Inertia weight factor	w	0.9
Acceleration constant	$c_1 \& c_2$	1.5
Random values	$r_1 \& r_2$	0-1
Control parameter	μ	4
Chaotic initial value	$M(1)$	0.3

Table 3 illustrates the optimal parameters for SOSMC controller gotten from the CPSO algorithm.

Table 3. Optimal controller parameters.

Controller	Parameter	Value
SOSMC	γ	0.15
	a_1	9.25
	a_2	0.00013

Figure 7 shows the response of BEM model for three patients after applying the suggested controller to regulate the BGL according to Table 3 parameters.

Figure 7. Closed-loop of the glucose regulation system based on SOSMC.

Figure 8. Insulin infusion rate based on SOSMC.

It can notice from simulation results of the suggested controller that the glucose output with controller tracks the desired BGL with small settling time (t_s), study state error ($e_{s.s.}$), and Mean Absolute Percentage Error (MAPE) between both the value of glucose within the control method and that of the standard model according to the formula below:

$$MAPE = \frac{1}{n} \sum_{t=0}^n \left| \frac{BGL_{desired} - BGL_{measured}}{BGL_{desired}} \right| \quad (15)$$

Where n is simulation duration, $BGL_{desired}$ is a value of glucose returned by the reference model, and $BGL_{measured}$ its actual system output under the controller Figure 9 illustrates the simulation results of the controlled glucose output system in MATLAB and FPGA.

Also it can be noticed both connection give approximately same output response.

Table 4. The simulation result's evaluation parameters for the three patients

Patients	t_s (min.)	MAPE	$e_{s.s.}$
Patient#1	508.94	0.0565	0.0031
Patient#2	496.14	0.0386	0.0005
Patient#3	476.25	0.0295	0.0008

Figure 9. The simulation results (with and without) FPGA implementation for one patient using SOSMC.

5. Conclusions

In this paper, the SOSMC based on STA has suggested for the blood glucose levels regulation system. The BEM mathematical model is considered. The optimal parameters for suggested controller have determined by the CPSO algorithm. Also, the SOSMC implemented by MATLAB/Simulink, and then the SOSMC also implemented by FPGA due to its great data storage capacity, low consumption of energy, and high speed of operation. In the presence of meal disturbance, the closed-loop control method has tested for various patients with specific parameters and it has been demonstrated that the glucose level is regulated in a suitable period at its base value.

Acknowledgements

The authors thank all the colleagues in the Computer Engineering Department, College of Engineering, Mustansiriyah University, for their unlimited support.

Conflict of interest

The publication of this article causes no conflict of interest.

6. References

- Kovács, L., & Paláncz, B. (2007, July). *"Glucose-insulin control of type1 diabetic patients in H^2/H^∞ space via computer algebra"*. In International Conference on Algebraic Biology (pp. 95-109). Springer, Berlin, Heidelberg.
- Sylvester, D. D., & Munje, R. K. (2017). *"Back stepping SMC for blood glucose control of type-1 diabetes mellitus patients"*. International Journal of Engineering Technology Science and Research, 4, 1-7.
- Keith-Hynes, P., Mize, B., Robert, A., & Place, J. (2014). *"The diabetes assistant: a smartphone-based system for real-time control of blood glucose"*. Electronics, 3(4), 609-623.
- Li, C., & Hu, R. (2007, October). *"PID control based on BP neural network for the regulation of blood glucose level in diabetes"*. In 2007 IEEE 7th International Symposium on Bioinformatics and Bioengineering (pp. 1168-1172). IEEE.
- Kaveh, P., & Shtessel, Y. B. (2006, March). *"Blood glucose regulation in diabetics using sliding mode control techniques"*. In 2006 Proceeding of the Thirty-Eighth Southeastern Symposium on System Theory (pp. 171-175). IEEE.
- Garcia-Gabin, W., Vehí, J., Bondia, J., Tarín, C., & Calm, R. (2008). *"Robust sliding mode closed-loop glucose control with meal compensation in type 1 diabetes mellitus"*. IFAC Proceedings Volumes, 41(2), 4240-4245.
- Abu-Rmileh, A., & Garcia-Gabin, W. (2011). *"Smith predictor sliding mode closed-loop glucose controller in type 1 diabetes"*. IFAC Proceedings Volumes, 44(1), 1733-1738.
- Geramipour, A., Khazaei, M., Marjaninejad, A., & Khazaei, M. (2013). *"Design of FPGA-based digital PID controller using Xilinx SysGen® for regulating blood glucose level of type-I diabetic patients"*. Int J Mechatron Electr Comput Technol, 3(7), 56-69.
- Parsa, N. T., Vali, A. R., & Ghasemi, R. (2014). *"Back stepping sliding mode control of blood glucose for type I diabetes"*. World Academy of Science, Engineering and Technology, International Journal of Medical, Health, Biomedical, Bioengineering and Pharmaceutical Engineering, 8(11), 779-783.
- Heydarinejad, H., & Delavari, H. (2017). *"Fractional order back stepping sliding mode control for blood glucose regulation in type I diabetes patients"*. In Theory and Applications of Non-integer Order Systems (pp. 187-202). Springer, Cham.
- Fisher, M. E. (1991). *"A semiclosed-loop algorithm for the control of blood glucose levels in diabetics"*. IEEE transactions on biomedical engineering, 38(1), 57-61.

12. Djouima, M., Azar, A. T., Drid, S., & Mehdi, D. (2018). *"Higher Order Sliding Mode Control for Blood Glucose Regulation of Type 1 Diabetic Patients"*. International Journal of System Dynamics Applications (IJSDA), 7(1), 65-84.
13. Levant A. *"Sliding order and sliding accuracy in sliding mode control"*. Int J Control 1993;58(6):1247–63.
14. Wang, D., Tan, D., & Liu, L. (2018). *"Particle swarm optimization algorithm: an overview"*. Soft Computing, 22(2), 387-408.
15. Hadi, E. A. (2019). *"Multi Objective Decision Maker for Single and Multi Robot Path Planning"* , MSc.thesis ,University of Technology, Iraq.
16. Hosseinpourfard, R., & Javidi, M. M. (2015). *"Chaotic pso using the Lorenz system: An efficient approach for optimizing nonlinear problems"*. Çankaya Üniversitesi Bilim ve Mühendislik Dergisi, 12(1).
17. Abood, L. H., Karam, E. H., & Issa, A. H. (2018, December). *"FPGA Implementation of Single Neuron PID Controller for Depth of Anesthesia Based on PSO"*. In 2018 Third Scientific Conference of Electrical Engineering (SCEE) (pp. 247-252). IEEE.