Population-Based Incremental Learning Algorithm for Identification of Blood Glucose Dynamics Model for Type-1 Diabetic Patients

Juan D. Hoyos* Freddy Bolanos Monica Vallejo Pablo S. Rivadeneira
Universidad Nacional de Colombia, Facultad de Minas, Carrera 80 N 65 - 223, Medellin, Colombia

*judhoyosgi@unal.edu.co bolanosm@unal.edu.co mavallev@unal.edu.co psrivade@unal.edu.co

Abstract—This paper describes the implementation results of a Population Based Incremental Learning (PBIL) algorithm, for model identification concerning blood glucose evolution over time, for Type - 1 diabetic patients. The adaptive PBIL algorithm adjusts the model parameters dynamically, for the sake of inferring autonomous glucose measures, in case of an input sensor disconnection, also provides information about the internal physiological parameters of the subject, which can be used to characterize the patient and design a control strategy for this proposed model. Obtained results in virtual patients are promising and allow to predict the behavior of blood glucose on a lack of input sensor information, it is expected to reproduce the results in real Type-1 Diabetic Patients.

Keywords—Population-Based Incremental Learning (PBIL), Adaptive modelling, Glucose Dynamics Model, Type-1 Diabetes.

Type of the submission: Regular Research Paper

I. INTRODUCTION

Diabetes is a disease that affects millions of people worldwide. It is predicted that by 2030 there will be a total of 366 million people who will suffer this illness [1]. In this paper, a model is proposed to describe the dynamics of blood glucose, for type 1 diabetes patients, which is also called diabetes mellitus (T1DM). This type of diabetes is an autoimmune disease, usually caused by hereditary factors, where the antibody system begins to destroy almost completely the β cells, located specifically in the pancreas. These β cells are responsible for producing the type of enzyme called insulin, which carry out the function of allowing glucose to enter the cells to be converted into energy and go through with their functions. By not producing the necessary amount of this enzyme or hormone, those patients with T1DM must administer it externally.

Among the consequences related to the lack of insulin, there is the possible permanence for a long period of time in high or low glucose levels, these clinical conditions are known as hyperglycemia (> 120mg/dl) and hypoglycemia (< 70mg/dl) respectively. As a consequence, patients who experience one of these states exhibit dizziness, breathing problems, kidney failure, impaired mental function and even death [2]. That is why it is necessary to perform an adequate management of the disease, monitoring and applying the adequate dose of insulin for the patient to remain at safe glucose levels. The normal range of levels in which glucose must be, is referred to as normoglycemia.

Nowadays, patients may use a device called insulin pump, which performs the function that the pancreas should accomplish, which is the administration of the insulin that patient needs, depending on his current glucose level. Such a level is measured through a Continuous Glucose Monitoring (CGM), which estimates the blood glucose values, and it depends on the amount of carbohydrates ingested by the subject, since once assimilated in the digestion, they are finally entered in the form of glucose into the bloodstream.

This insulin pump could perform a closed loop control as reported in previous works [3, 4], which highlights the relevance of having a model in several types of control for the sake of describing the dynamics of blood glucose in patients with T1DM. Several types of model have been proposed in literature [3, 5, 6]. Some of them include physical parameters of the patient and other work whit stochastic data such as the use of neural networks [7]. Some of such models exhibit certain limitations, such as the type of inputs considered, since they are impulsive inputs generally. In addition to being able to characterize a patient, the model can also carry out a function as an estimator, predicting glucose levels in the event that a sensor disconnection occurs at certain times of the day. Therefore, glucose levels must be predicted in a certain prediction horizon for patient safety.

When proposing a model, there is also the need to identify its parameters, task that is generally carried out by defining a cost function, converting the identification problem into an optimization one. Being a non-convex optimization problem, there is a need to find a heuristic optimization technique. Several methods have been used, such as the one reported in [8], where the parameters of a model composed of three states were identified through genetic algorithms. Many other population-based algorithms have not been used to solve this kind of biological system identification, so it was decided to use the Population-Based Incremental Learning algorithm (PBIL), since it has proven to be an efficient method for solving non-convex problems such as those shown in [9, 10].

Several modifications have been proposed in different heuristic algorithms, in which some features of the algorithm are readjusted as the algorithm works, in order to have better
This paper describes a model intended to deliver estimations of blood insulin. The model is adjusted in real time by changing its parameters, by means of a Population-Based Incremental Learning Algorithm, which is additionally implemented with an adaptive feature, for the sake of speeding up the convergence time, without sacrificing quality of the solutions. As a comparison reference, a quite more complex model was used, which was implemented in the Uva/Padova Simulator [12]. Some data from real patients was also used to testing the model. By using an arbitrary point of time, data from both real and virtual patients were divided in two separated sets. For a given patient, data from first time interval were used for adjusting the model parameters, by means of the PBIL algorithm. Second set of data were used to measuring the ability of the model of predicting the behavior of blood glucose. The remaining of this document is organized as follows. Section II describes the PBIL algorithm, and its adaptive feature. Section III depicts the blood insulin model, and the parameters related to the optimization performed. The results of model testing are shown in section IV. Finally, concluding remarks are presented at the end of this paper.

II. POPULATION-BASED INCREMENTAL LEARNING (PBIL)

Population-Based Incremental Learning (PBIL) is an optimization approach which derives both from Competitive Neural Networks (CNN), and Genetic Algorithms (GA) [13]. It uses stochastic search information in order to adjust a probability array, which progressively converges toward an optimal solution. Such a probability array is adjusted in an iterative fashion, by taking into account the solutions that have been considered earlier by the algorithm. The feature of using a population of solutions for the space exploration, precludes the local optimum issue, and has the potential of allowing parallel-search implementations.

The PBIL searching process performs a concurrent exploration of the solutions space, allowing the iterative updating of the probability array, which stores information concerning the bests solutions found so far. Let us suppose an optimization problem in which a feasible solution may be represented as a binary string. In such a case, the probability array might then take the form of a vector, in which each entry stores a probability associated by position with each bit of the optimal solution. If a given probability entry approaches or converges to one, that would be the associated value of the corresponding bit in the optimal solution. Figure 1 depicts a probability vector for a given binary PBIL optimization.

Each probability ($P_i$) of the vector depicted in Figure 1 represents a single binary attribute of the solution to be optimized. At the beginning of the searching process, the whole vector of probabilities is set to $1/2$, allowing a uniform and wide exploration of the solutions space. For each algorithm iteration, a new population of potential solutions is created by using the probabilities in the vector (i.e. those probabilities close to zero, are prone to generate solutions with the corresponding bit set to zero, and vice versa).

Algorithm 1 PBIL Algorithm

1: procedure PBIL
2: \hspace{1cm} $P_i \leftarrow 1/2; \forall \ 1 \leq i \leq N$ \hspace{0.5cm} $\triangleright$ Maximum diversity
3: \hspace{1cm} while $\text{Entropy}(P) > \text{Tol}$ do
4: \hspace{1.5cm} $P \leftarrow \text{Generate}_\text{Pop}(P)$
5: \hspace{1.5cm} $\text{Best} \leftarrow \text{Find}_\text{Best}(P)$
6: \hspace{1.5cm} $E \leftarrow \text{Entropy}(P)$
7: \hspace{1.5cm} $LR \leftarrow \text{Learning}_\text{Rule}(E)$
8: \hspace{1.5cm} $P \leftarrow \text{Adjust}_\text{Prob}(\text{Best}, LR)$
9: \hspace{1cm} return $\text{round}(P)$ \hspace{0.5cm} $\triangleright$ Solution pointed by P

Algorithm 1 shows the main steps involved in the PBIL optimization algorithm. As mentioned before, by setting all entries of $P$ vector to 0.5, maximum diversity of population at early stages of the search process is guaranteed. At each algorithm iteration, a new population of solutions is generated starting from the probability vector ($P$). Such a task is performed by the $\text{Generate}_\text{Pop}$ routine. In order to guide the searching process, the best solution of such population is then chosen. Such a best solution ($\text{Best}$ in Algorithm 1) will be used to slightly change the probabilities in the $P$ vector. The latter is performed by the $\text{Adjust}_\text{Prob}$ routine.

However, in the adaptive version of the algorithm, it is necessary to iteratively adjust the rate of change performed to the probabilities in $P$ vector. Such rate is labeled as the $LR$ parameter in Algorithm 1 and the way in which LR changes as the algorithm converges, is referred to as the Learning Rule [14]. $P$ vector entropy provides a measure of the convergence status of the algorithm, and the learning rule allows to favor either exploration or exploitation of the solutions space at different stages of the convergence process. Entropy routine in Algorithm 1 performs the $P$ entropy computation by means of the Shannon formulation, as shown in Equation 1. On the other hand, Learning Rule routine computes the $LR$...
parameter according to a linear rule behavior.

\[ E = -\frac{1}{N} \cdot \sum_{i} \log(P_i) \cdot P_i \]  

(1)

The linear learning rule \cite{14} starts with low values of the \( LR \) parameter, for the sake of allowing a wide solutions space exploration (i.e. the probabilities in \( P \) change slowly, allowing to perform a detailed searching of the optimal). The \( LR \) parameter is increased as the algorithm converges, as an inverse proportional function of the entropy, which means that at final iterations of the algorithm, \( LR \) will has its maximum value, as shown in Equation (2). Such high value of \( LR \) allows the exploitation of the solutions space, which is nothing but the speedup of the convergence process, by performing stronger changes to probabilities in array \( P \).

\[ LR = [LR_{max} - E \cdot (LR_{max} - LR_{min})] \]  

(2)

As mentioned before, the updating of the probabilities in \( P \) vector is performed by means of the Update_Prob routine. In more complex probability arrays (such as matrices or three dimensional sets) a modified Hebbian Rule is used for such probability adjusting \cite{15}. For the binary vector case, such updating takes the form of the Equation (3), where suffixes Old and New refers to the old and new (updated) version of the probability entry, and \( S_{i} \) refers to the corresponding binary entry in the best solution found at a given iteration.

\[ P_{i_{\text{New}}} = [1 + LR \cdot (2 \cdot S_{i} - 1)] \cdot P_{i_{\text{Old}}} \]  

(3)

Several criteria may be used in order to stop the iterative process in a PBIL optimization. The \( P \) vector entropy is the more often alternative, since it measures the status of convergence of the PBIL algorithm. At early stages of the searching process, associated entropy values are high, indicating a high population diversity (or alternatively, high probability dispersion). As the algorithm converges and probabilities trend either to zero or one, entropy values decrease toward zero. Since waiting to the entropy values to reach zero may be very restrictive, a tolerance (\( Tol \) in Algorithm \cite{1}) is used as a minimum value. The algorithm ends by rounding the probabilities in the \( P \) vector, in order to approximating its entries either to zero or one, for the sake of deriving the optimal solution.

III. MODEL OF DIABETIC TYPE 1 PATIENTS

The model structure is taken from \cite{3} and it has the form

\[ \dot{x}(t) = Ax(t) + B \begin{pmatrix} u_{i}(t) \\ u_{m}(t) \end{pmatrix} + E, \quad y(t) = x_{1}(t) \]  

(4)

where \( x(t) \) is the state vector representing the glycemia, insulinemia, subcutaneous insulin concentration, \( \text{CHO} \) plasma concentration, and \( \text{CHO} \) concentration in the second compartment. The model has six parameters to be identified and it is a stable system with equal poles and affine term. In \cite{3} is predicted the behavior up to 24 hrs (1 day) by using the same set of parameters.

The term \( k_{0} \) is the main difference with similar structures given in \cite{5, 6}. This term is associated to a glucose self-regulation mechanism which allows the glucose level to be driven toward a basal state. This self-regulation effect may represent renal clearance of glucose at high glucose concentration or counter-regulatory hormone effect at low glycaemia. The values reported in \cite{3} for this parameter are 0.0039+/−0.0005 1/min. In consequence, accepted physiologic equilibrium are recovered, i.e. i) with no insulin and no meals, glycaemia rises up to a limit. This limit is imposed to the physical impossibility of the liver to produce glucose. In contrast with the model given in \cite{6}, ii) with basal insulin and no meals, a basal glucose level is achieved, and iii) with basal insulin and meal intake, after a long interval of time, basal glucose is recovered. This parameter \( k_{0} \) dynamically has an important impact. The model considering \( k_{0} = 0 \) is unstable and with a \( k_{0} \neq 0 \) is stable, as it happens in reality. It is also important to note that in \cite{3} is modeled insulin absorption and meal absorption with two compartment models with diffusion time constant equals. On the other hand, in the seminal work of \cite{16} these compartmental models have different time constants. In this work, it is considered that this simplification is not necessary and that this restriction degrades the model fit. Our hypothesis is that this structure can become flexible to achieve a better fit.

A. Long-term T1DM model

The proposed model comprises five compartments describing insulin absorption and action, meal absorption dynamics, and glucose dynamics. The model uses subcutaneous insulin delivery and carbohydrate intake as inputs and blood glucose as the model output. To enable a long-term validity and a fit with clinical data, the model has realistic equilibrium points. It is an elementary glucose, insulin and digestion model with five states and eleven independent parameters; and its state-space
representation is given by the structure (4) but with matrices:

\[
A = \begin{pmatrix}
-\theta_1 & -\theta_2 & 0 & \theta_3 & 0 \\
0 & \frac{1}{\theta_4} & \frac{\theta_5}{\theta_4} & 0 & 0 \\
0 & 0 & -\frac{\theta_6}{\theta_4} & 0 & 0 \\
0 & 0 & 0 & -\frac{\theta_7}{\theta_4} & 0 \\
0 & 0 & 0 & 0 & -\frac{\theta_8}{\theta_4} \\
\end{pmatrix}, \quad B = \begin{pmatrix}
\theta_2 \\
\theta_6 \\
\theta_8 \\
\theta_10 \\ \theta_{12}
\end{pmatrix}, \quad C = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 \\
\end{pmatrix},
\]

(8)

where \( x_1 \) is the glycemia [mg/dl] (or glucose in the plasma compartment), \( x_2 \) and \( x_3 \) are delivery rates of insulin [U/min] in the first and second insulin absorption compartment (related to the plasma and interstitial fluid), respectively, \( x_4 \) and \( x_5 \) are delivery rates of carbohydrates [g/min] in the first and second meal absorption compartment, respectively. Input \( u_i \) is the insulin pump delivery rate [U/min], \( u_{m} \) is the delivery rate of carbohydrates [g/min] due to the meals. The parameters have the following constraints: \( \theta_4, \theta_6, \theta_8, \theta_{10} > 0 \), and \( 0 < \theta_5, \theta_7, \theta_9, \theta_{11} <= 1 \). The states of this model have a physiological meaning, and consequently, the parameters also have a physiological meaning. \( \theta_4 \) and \( \theta_6 \) are the diffusion time constants in the insulin compartments, while \( \theta_8 \) and \( \theta_{10} \) are the diffusion time constants in the two digestion compartments. The gains \( \theta_5 \) and \( \theta_7 \) are insulin diffusion effectiveness in each compartment. Similarly, \( \theta_9 \) and \( \theta_{11} \) are meal absorption effectiveness in each compartment. Parameter \( \theta_{12} \) is the net balance between endogenous glucose production and insulin independent glucose consumption (e.g. by the brain). \( \theta_2 \) is the insulin action effectiveness and \( \theta_3 \) is the carbohydrate bioavailability.

This model structure with its constraints is more interesting than the structures given in [3, 6, 8], since it allows different time constants for each compartment and gains lesser than one. The gains lesser than one allow to model the deactivation of the insulin molecule while diffusing through a compartment and the incomplete digestion (or absorption) of the calculated intake when diffusing through each compartment. These features are taken from [16]. The results about identification of the model are presented in Section IV.

IV. RESULTS

The parametric model just described above, was used to approximate the behavior of blood glucose in type I diabetes patients. As a comparison reference, data of 33 virtual patients, among children, teenagers and adults, supplied by the UVA/Padova TIDM simulator were used [12]. Real patients data were also used, by using an insulin pump for collecting real time measures of blood glucose. The system used for this purpose was the Medtronic Paradigm Veo 754. In any case, sampling time was five minutes, which is an usual feature of insulin pumps, and blood glucose measuring systems.

The optimization was performed over the model parameters, described in previous section, by using the PBIL algorithm presented in Section II. The search over the solutions space, which includes each potential combination of model parameters was guided by the fitness function depicted in Equation (9). As can be seen in such equation, squared error was used as cost function to minimize, with a tolerance value of 10\(^{-12}\).

\[
\text{fitness} = ||\text{Glucose}(t) - y(t)||^2
\]

(9)

A tuning process was performed over the PBIL algorithm, in order to set the best values for its execution, by taking into account both convergence times, and quality of the found solutions. At the end of several executions of the algorithm, this trade off values were obtained, for the various algorithm parameters, such as population size, learning rates and so on. Table I summarizes the results of such a tuning, which summarizes an exhaustive adjusting process, with several executions of the PBIL algorithm, evaluating each time both its performance, and the best solutions found.

### Table I

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Population size</th>
<th>( \text{Gen}_{\text{max}} )</th>
<th>( LR_{\text{min}} )</th>
<th>( LR_{\text{max}} )</th>
<th>Tolerance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>100</td>
<td>20000</td>
<td>0.05</td>
<td>0.2</td>
<td>( 10^{-12} )</td>
</tr>
</tbody>
</table>

A. Virtual Patients

After the tuning process, the algorithm was used to optimize the model parameters of 33 virtual patients. PBIL optimization was performed fifteen times for each of these patients, in order to reduce the uncertainty of the tests. Figure 2 depicts the evolution process of the fifteen executions of the PBIL algorithm, for the number 23 patient. In addition to identifying the parameters of the model, it was also necessary to perform the identification of the initial conditions for the states \( X_2(0) \), \( X_3(0) \), referred in the equation (4) just because the behavior of the model depends on the initial condition of insulin states. An initial value of 0 for states \( X_4(0) \), \( X_5(0) \) was assumed.

The optimization process starts by setting an arbitrary time in the data set. The data before this time point are used to train the model, which is nothing but the iterative adjusting of the model parameters by means of the PBIL algorithm. By minimizing the squared error between real data, and the model output (\( \text{Glucose}(t) \) and \( y(t) \) in Equation (9), respectively), it is expected that the model to be able to predict future behavior of blood glucose. As can be seen in Figure 2 the algorithm always converges quite before 200 iterations and reaches a minimum value of the target function (local minimum), similar to results in [9]. These feature allows to predict the performance of the PBIL optimization and making budgets for its execution in real time.

Figure 3 shows the real and modeled data for the training process just described, and after the chosen random time point, which will be referred to as validation time. In such a
The linear learning rule described in Equation (2) is also related to the evolution of the entropy. Different learning rules may lead to several behaviors regarding the exploration and exploitation features of the algorithm.

As in Figure 3, Figure 5 depicts the real data compared to training and validation data, for several days of measures, this time for the number 24 patient. Both figures show a daily coefficient, referred as $CV_i$, which is an empirical measurement used in the medical environment, to confine the amount of change of glucose expected for a given patient. Such a parameter is given in an percentage fashion (with values between 0 and 1), and may be used in the future for estimations of the model quality, or in order to establish if a given set of measures is exhibiting weird behavior, with respect to the expected or predicted data. For Figure 5, the absolute error obtained of the validation data, with respect to the real glucose measures, was of 9.77%.

A good trade off between performance (i.e. convergence time) and quality of the solutions was achieved by means of the tuning process described earlier. A feasible evidence of the latter may be viewed in Figure 4. Figure shows the evolution of the PBIL vector entropy, as a function of the number of iterations of the algorithm. A good searching process implies a balance between exploration and exploitation of the solutions space. Higher values of entropy (at the beginning of the searching process) imply exploration feature, whereas lower values of entropy are related to exploitation of the solutions space. Figure 4 suggest that such a balance between these two features is present in the tuned PBIL algorithm.

Figure 6 shows the relative error between validation and real data for the glucose values of the 33 patients, by taking into account the fifteen executions of the PBIL algorithm for each patient. As can be seen in figure 6, there is a great variability in the algorithm solutions for some patients, as is the case for virtual patient #30, corresponding to the group of younger
patients.

As can be seen from figure 7, a relative error of validation of 15.84% is obtained, which is greater than the average error obtained for virtual patients. This behavior can be explained as a consequence of the following factors:

- Presence of noise associated with the glucose sensor used "Continuous Glucose Monitor" (CGM) from Minimed Paradigm Veo 754 system.
- Uncertainty associated with the information records that the real patient must make in the insulin pump, such as the amount of carbohydrates ingested and the exact moment of each intake.
- The model does not consider variables such as stress and physical activity performed by the patient, which can significantly influence the glucose dynamics in the patient.
- High coefficients of variation (CV) in data, which could also imply a change in the physiological parameters of the patient.

In conclusion, the unreliability of the real data leads to greater errors in the validation process of the model, which poses interesting challenges that are still pending to be solved, such as including variables in the proposed model, finding better filtering and estimation techniques, and developing an automatic correction in the CHO inputs which introduce error in the measurement. One advantage of using the PBIL algorithm to perform the identification is the fact that some parameters can be defined as integer values, since the individual resolution can be changed by parameter. This is an advantage to be able to perform the correction process of CHO entries, since these values are of integer type, as well as the sample in which the meal was registered.

C. Graphical User Interface

The development of the graphical user interface, shown in figure 8, has been an additional result obtained in this work. This interface turns out to be a versatile tool that allows medical personnel and researchers to easily and efficiently manage the data of each patient. As shown in the figure the data delivered by the insulin pump in format (*.csv) can be entered directly.

The data collected for each patient is stored in a database, which can then be used for the identification of individual parameters. The interface also allows to select the dates from which you want to extract the data. Additionally, some controls can be observed in the lower part to modify the amount of data used for the identification and validation of the model. Finally, the interface allows saving the results of the PBIL algorithm shown in the lower part of the graph. It is a tool that is expected to be used by medical personnel to calibrate the insulin pump, and deliver the appropriate dose of insulin for each patient’s state throughout the day, which will help to reduce episodes of hypoglycemia and hyperglycemia.
V. CONCLUSION

A method for training (identifying) a simple blood glucose model for diabetes type I patients has been presented and tested. Such a method is based on the PBIL algorithm and performs a non-exhaustive search over the solutions space, for the sake of optimizing the squared error between real and modeled data. Performance of the proposed solution is promising. A mean absolute error of 6% was found when using validation data (when the model is used as a predictor), which leads to reliable estimations. Performance of the model is also satisfying, with less than 200 iteration prior to reaching the algorithm convergence. Finally, a graphic user interface was developed and used in the tests reported in this paper.

ACKNOWLEDGEMENT

The authors would like to thank to Universidad Nacional de Colombia, by its support in the development of this work.

REFERENCES


