Hindawi
Computational and Mathematical Methods in Medicine
Volume 2018, Article ID 4091497, 8 pages
https://doi.org/10.1155/2018/4091497

Research Article

Control of Blood Glucose for Type-1 Diabetes by Using Reinforcement Learning with Feedforward Algorithm

Phuong D. Ngo,[1] **Susan Wei**,[2] **Anna Holubová**,[3] **Jan Muzik**,[3] **and Fred Godtliebsen**[1]

[1]*UiT The Arctic University of Norway, Tromsø, Norway*
[2]*The University of Melbourne, Australia*
[3]*Czech Technical University, Prague, Czech Republic*

Correspondence should be addressed to Phuong D. Ngo; phuong.ngo@uit.no

Received 9 August 2018; Revised 18 November 2018; Accepted 28 November 2018; Published 30 December 2018

Guest Editor: Ka L. Man

Background. Type-1 diabetes is a condition caused by the lack of insulin hormone, which leads to an excessive increase in blood glucose level. The glucose kinetics process is difficult to control due to its complex and nonlinear nature and with state variables that are difficult to measure. *Methods*. This paper proposes a method for automatically calculating the basal and bolus insulin doses for patients with type-1 diabetes using reinforcement learning with feedforward controller. The algorithm is designed to keep the blood glucose stable and directly compensate for the external events such as food intake. Its performance was assessed using simulation on a blood glucose model. The usage of the Kalman filter with the controller was demonstrated to estimate unmeasurable state variables. *Results*. Comparison simulations between the proposed controller with the optimal reinforcement learning and the proportional-integral-derivative controller show that the proposed methodology has the best performance in regulating the fluctuation of the blood glucose. The proposed controller also improved the blood glucose responses and prevented hypoglycemia condition. Simulation of the control system in different uncertain conditions provided insights on how the inaccuracies of carbohydrate counting and meal-time reporting affect the performance of the control system. *Conclusion*. The proposed controller is an effective tool for reducing postmeal blood glucose rise and for countering the effects of external known events such as meal intake and maintaining blood glucose at a healthy level under uncertainties.

1. Introduction

Type-1 diabetes is a chronic condition that is characterized by an excessive increase in blood glucose level because the pancreas does not produce insulin hormone due to the autoimmune destruction of pancreatic beta cells. High blood glucose can lead to both acute and chronic complications and eventually result in failure of various organs.

Until today, there are many challenges in control of the blood glucose in type-1 diabetes. One of them is that the glucose kinetics process is complex, nonlinear, and only approximately known [1]. There are also many external known and unknown factors that affect the blood glucose level such as food intakes, physical activities, stress, and hormone changes. Generally, it is difficult to predict and quantify those factors and disturbances.

By using control theories, various studies have been conducted to design a control system for patients with type-1 diabetes. For example, Marchetti et al. [2], derived an improved proportional-integral-derivative controller for blood glucose control. Soylu et al. [3] proposed a Mamdani type fuzzy control strategy for exogenous insulin infusion. Model predictive control has also been widely used in type-1 diabetes and artificial pancreas development [4, 5]. Recently, together with the development of artificial intelligence and machine learning, reinforcement learning (RL) has emerged as a data-driven method to control unknown nonlinear systems [6, 7] and has been used as a long-term management tool for chronic diseases [8, 9]. The biggest advantage of RL compared to other methods is that the algorithm depends only on interactions with the system and does not require a well represented model of the environment. This especially

makes RL well suited for type-1 diabetes since the modelling process of the insulin-kinetic dynamics is complex and requires invasive measurements on the patient or must be fit through a large dataset. Hence, by using RL as the control algorithm, the modelling process can be bypassed, which makes the algorithm not susceptible to any modelling error.

In diabetes, controlling of blood glucose require actions that are made at specific instance throughout the day in terms of insulin doses or food intakes. The actions are based on the current observable states of the patients (e.g., blood glucose measurement and heart rate). The effectiveness of the actions is calculated by how far the measured blood glucose value is compared to the healthy level. In RL, an agent makes decision based on the current state of the environment. The task of the algorithm is to maximize a cumulative reward function or to minimize a cumulative cost function. Based on these similarities in the decision-making process between a human being and a RL agent, RL may be key to the development of an artificial pancreas system.

When dealing with meal disturbances, modelling of glucose ingestion is the norm as well as the first step in designing a controller for disturbance rejection [10]. Feed-forward control was proven to be an effective tool to improve disturbance rejection performance [11, 12]. In control system theory, feed-forward is the term that describes a controller that utilizes the signal obtained when there is a (large) deviation from the model. Compared to feed-back control, where action is only taken after the output has moved away from the setpoint, the feed-forward architecture is more proactive since it uses the disturbance model to suggest the time and size of control action. Furthermore, building a meal disturbance model is simpler and requires less data to fit than finding the insulin-glucose kinetics. Based on the model, necessary changes in insulin actions can be calculated to compensate for the effects of carbohydrate on the blood glucose level.

A challenge in the control of the blood glucose is the lack of real-time measurement techniques. With the development of continuous glucose measurement sensors, blood glucose level can be measured and provided to the controller in minute intervals. However, blood glucose value alone is usually not enough to describe the states of the system for control purpose. Therefore, an observer is needed to estimate other variables in the state space from the blood glucose measurement. In this paper, the Kalman filter was chosen for that purpose since it can provide an optimal estimation of the state variables when the system is subjected to process and measurement noises [13, 14].

Vrabie et al. [15] established methodologies to obtain optimal adaptive control algorithms for dynamical systems with unknown mathematical models by using reinforcement learning. Based on that, Ngo et al. [16] proposed a reinforcement learning algorithm for updating basal rates in patients with type-1 diabetes. This paper completes the framework for blood glucose control with both basal and bolus insulin doses. The framework includes the reinforcement learning algorithm, the feed-forward controller for compensating food intake and the Kalman filter for estimating unmeasurable state variables during the control process. This paper also conducts simulations under uncertain information to evaluate the robustness of the proposed controller.

2. Methods

2.1. Problem Formulation. The purpose of our study is to design an algorithm to control the blood glucose in patients with type-1 diabetes by the means of changing the insulin concentration. The blood glucose metabolism is a dynamic system in which the blood glucose changing over time as the results of many factors such as food intake, insulin doses, physical activities, and stress level. The learning process of RL is based on the interaction between a decision-making agent and its environment, which will lead to an optimal action policy that results in desirable states [17]. The RL framework for type-1 diabetes includes the following elements:

(i) The state vector at time instance k consists of the states of the patient:

$$\mathbf{x}_k = \left[\, g(k) - g_{\mathrm{d}}(k) \;\; \chi(k) \,\right]^T, \tag{1}$$

where $g(k)$ and $g_{\mathrm{d}}(k)$ the are measured and desired blood glucose levels, respectively, and $\chi(k)$ is the interstitial insulin activity (defined in the appendix).

(ii) The control variable (insulin action) u_k, which is part of the total insulin i_k (a combination of the basal and the bolus insulin (Figure 1)):

$$i_k = u_{\mathrm{basal}}(k) + u_{\mathrm{bolus}}(k) = u_k + u_{\mathrm{basal}}(0) + u_{\mathrm{bolus}}(k), \tag{2}$$

where $u_{\mathrm{basal}}(k)$ and $u_{\mathrm{bolus}}(k)$ are the basal and bolus at time instance k, respectively.

(iii) The cost received one time-step later as a consequence of the action. In this paper, the cost was calculated by the following quadratic function:

$$r_{k+1} = \mathbf{x}_k^T \mathbf{Q} \mathbf{x}_k + u_k^T R u_k, \tag{3}$$

where $\mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 0.1 \end{bmatrix}$ and $R = 0.01$. Each element in matrix $\mathbf{Q}$ and the value of R indicate the weighting factors of the cost function. The element in the first row and the first column of $\mathbf{Q}$ has the highest value, which corresponds to the weighting of the difference between the measured blood glucose and the prescribed healthy value. Since our ultimate goal is to reduce this difference, the factor of this measurement should have the largest value in the cost function. The element in the second row and second column of $\mathbf{Q}$ corresponds to the weighting of the interstitial insulin activity. The value of R indicates the weighting factor of the action (basal update). Minimizing the cost function, therefore, becomes the problem of minimizing the difference

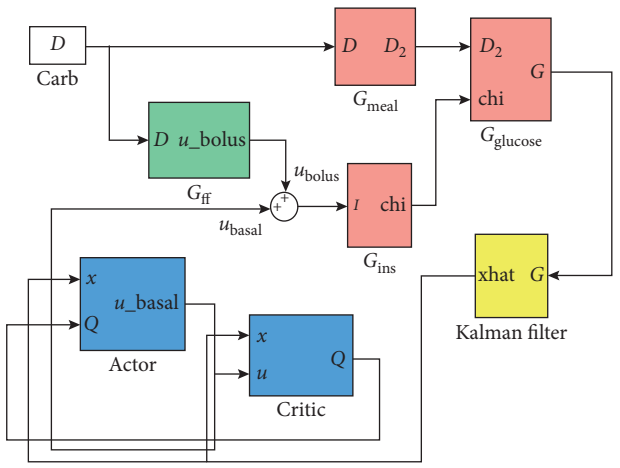

Figure 1: Control system diagram.

between the measured blood glucose and the desired value, the interstitial insulin activity, and the change in basal insulin.

At time instance $k+1$, a sequence of observations would be $\mathbf{x}_k$, u_k, r_{k+1}, $\mathbf{x}_{k+1}$ and u_{k+1}. Based on this observation, the agent receives information about the state of the patient and chooses an insulin action. The body reacts to this action and transitions to a new state. This determines the cost of the action.

For the control design purpose, the blood glucose model (Appendix) was divided into three submodels: the meal (G_{meal}), the insulin (G_{ins}), and the glucose kinetics (G_{glucose}). The controller has three main components: the actor, the critic, and the feedforward algorithm. The actor is used to estimate the action-value function, the critic's task is to obtain optimal basal insulin, and the feedforward algorithm is used to propose the bolus insulin profile for disturbance compensation (food intake). The purpose of the Kalman filter is to estimate unmeasurable states of the patient.

2.2. Basal Update by Actor and Critic. When the patient is in a fasting condition, the controller only needs to change the basal insulin level through the actor and the critic. Based on the current state $\mathbf{x}_k$, the actor proposes an insulin action u_k through the policy $\pi: u_k = \pi(\mathbf{x}_k)$. The updated basal rate is obtained from u_k as follows:

$$u_{\text{basal}} = u_k + u_{\text{basal}}(0). \tag{4}$$

After each action, the patient transforms into a new state, and the cost associated with the previous action can be calculated using equation (3). The action-value function (Q-function) of action u is defined as the accumulation of cost when the controller takes action $u_k = u$ at time instance k and then continues following policy $\pi(\mathbf{x}_{k+1})$:

$$Q_k^\pi(\mathbf{x}, u) = \mathbb{E}_\pi\left\{\sum_{i=0}^{\infty} \gamma^i r_{k+i+1} \mid \mathbf{x}_k = \mathbf{x}, u_k = u\right\}, \tag{5}$$

where γ (with $0 < \gamma \leq 1$) is the discount factor that indicates the weighting of future cost in the action-value function.

The action-value function depends on the current state and the next action. It was shown that the action-value function satisfies the following recursive equation (Bellman equation) [15, 17]:

$$Q_k^\pi(\mathbf{x}, u) = r_k + \gamma Q_{k+1}^\pi(\mathbf{x}, u). \tag{6}$$

Since the state space and action space are infinite, function approximation was used in this paper for estimation of the Q-function. In this case, the Q-function was approximated as a quadratic function of vectors $\mathbf{x}_k$ and u_k:

$$Q_k^\pi(\mathbf{x}, u) \approx \mathbf{z}_k^T \mathbf{P} \mathbf{z}_k, \tag{7}$$

where the symmetric and positive definite matrix $\mathbf{P}$ is called the kernel matrix and contains the parameters that need to be estimated. Vector $\mathbf{z}_k$ is the combined vector of $\mathbf{x}_k$ and u_k:

$$\mathbf{z} = \begin{bmatrix} \mathbf{x}_k^T & u_k^T \end{bmatrix}^T. \tag{8}$$

With Kronecker operation, the approximated Q-function can be expressed as a linear combination of the basis function $\Phi(\mathbf{z}_k) = \mathbf{z}_k \otimes \mathbf{z}_k$:

$$Q_k^\pi(\mathbf{x}, u) \approx \mathbf{z}_k^T \mathbf{P} \mathbf{z}_k = \mathbf{w}^T(\mathbf{z}_k \otimes \mathbf{z}_k) = \mathbf{w}^T \Phi(\mathbf{z}_k), \tag{9}$$

where $\mathbf{w}$ is the vector that contains elements of $\mathbf{P}$ and $\otimes$ is the Kronecker product.

By substituting $Q_k^\pi(\mathbf{x}, u)$ in equation (6) by $\mathbf{w}^T \Phi(\mathbf{z}_k)$ and using the policy iteration method with the least square algorithm [15], elements of vector $\mathbf{w}$ can be estimated. Matrix $\mathbf{P}$ can then be obtained from $\mathbf{w}$ using the Kronecker transformation.

By decomposing the kernel matrix $\mathbf{P}$ into smaller matrices $\mathbf{P}_{\mathbf{xx}}$, $\mathbf{P}_{\mathbf{x}u}$, $\mathbf{P}_{u\mathbf{x}}$, and $\mathbf{P}_{uu}$, the approximated Q-function can be written as follows:

$$Q_k^\pi(\mathbf{x}, u) = \frac{1}{2}\begin{bmatrix} \mathbf{x}_k \\ u_k \end{bmatrix}^T \mathbf{P} \begin{bmatrix} \mathbf{x}_k \\ u_k \end{bmatrix} = \frac{1}{2}\begin{bmatrix} \mathbf{x}_k \\ u_k \end{bmatrix}^T \begin{bmatrix} \mathbf{P}_{\mathbf{xx}} & \mathbf{P}_{\mathbf{x}u} \\ \mathbf{P}_{u\mathbf{x}} & \mathbf{P}_{uu} \end{bmatrix} \begin{bmatrix} \mathbf{x}_k \\ u_k \end{bmatrix}. \tag{10}$$

The current policy is improved with actions that minimize the Q-function $Q_k^\pi(\mathbf{x}, u)$. This can be done by first taking the partial derivative of the Q-function and then solving $\partial Q_k^\pi(\mathbf{x}, u)/\partial u = 0$. The optimal solution can thereafter be obtained as follows [15]:

$$u_k = -\mathbf{P}_{uu}^{-1} \mathbf{P}_{u\mathbf{x}} \mathbf{x}_k. \tag{11}$$

With that, the update of basal insulin is

$$u_{\text{basal}} = -\mathbf{P}_{uu}^{-1} \mathbf{P}_{u\mathbf{x}} \mathbf{x}_k + i_{\text{be}}, \tag{12}$$

where i_{be} is the equilibrium basal plasma insulin concentration.

2.3. Bolus Update by Feedforward Algorithm. When the patient consumes meals, in addition to the basal insulin, the controller calculates and applies boluses to compensate for the rise of blood glucose as the results of carbohydrate in the food. The feedforward algorithm first predicts how much blood glucose level will rise and then suggests a bolus profile

to counter the effects of the meal. The starting time of the bolus doses was also calculated by the algorithm based on the meal intake model.

Since the meal intake model (equations (A.1) and (A.2)) and the insulin model (equation (A.4)) are linear time-invariant (LTI) models, they can be transformed from state space equations into transfer functions as follows:

$$G_{\text{meal}}(s) = \frac{D_2(s)}{D(s)} = \mathbf{C}_{\text{meal}}\left(s\mathbf{I} - \mathbf{A}_{\text{meal}}\right)^{-1}\mathbf{B}_{\text{meal}} = \frac{A_G}{\left(s\tau_D + 1\right)^2},$$

$$G_{\text{ins}}(s) = \frac{D_2(s)}{D(s)} = \mathbf{C}_{\text{ins}}\left(s\mathbf{I} - \mathbf{A}_{\text{ins}}\right)^{-1}\mathbf{B}_{\text{ins}} = \frac{p_3}{s + p_2}, \tag{13}$$

where

$$\begin{aligned}
\mathbf{A}_{\text{meal}} &= \begin{bmatrix} -1/\tau_D & 0 \\ 1/\tau_D & -1/\tau_D \end{bmatrix}, \\
\mathbf{B}_{\text{meal}} &= \begin{bmatrix} A_G \\ 0 \end{bmatrix}, \\
\mathbf{C}_{\text{meal}} &= \begin{bmatrix} 0 & 1/\tau_D \end{bmatrix}, \\
\mathbf{A}_{\text{ins}} &= -p_2, \\
\mathbf{B}_{\text{ins}} &= p_3, \\
\mathbf{C}_{\text{ins}} &= 1.
\end{aligned} \tag{14}$$

Descriptions and values of τ_D, p_2, and p_3 are shown in Tables 1 and 2. The transfer function from the meal intake $D(s)$ to the blood glucose level $g(s)$ can be calculated as

$$F(s) = \frac{g(s)}{D(s)} = \left(G_{\text{meal}}(s) + G_{\text{ff}}(s)G_{\text{ins}}(s)\right)G_{\text{glucose}}(s). \tag{15}$$

In order to compensate for the meal, the gain of the open loop system $F(s)$ must be made as small as possible. Hence, the feedforward transfer function was chosen such that $G_{\text{meal}}(s) + G_{\text{ff}}(s)G_{\text{ins}}(s) \longrightarrow 0$, which leads to

$$G_{\text{ff}}(s) = -G_{\text{meal}}(s)G_{\text{ins}}^{-1}(s) = \frac{-A_G\left(s + p_2\right)}{p_3\left(\tau_D s + 1\right)^2}. \tag{16}$$

The meal compensation bolus in s-domain can be calculated from the feedforward transfer function:

$$u_{\text{bolus}}(s) = G_{\text{ff}}(s)D(s) = \frac{-A_G\left(s + p_2\right)}{p_3\left(\tau_D s + 1\right)^2}D(s). \tag{17}$$

Hence, the feedforward action becomes the output of the following dynamic system, which can be solved easily using any ordinary differential equation solver:

$$\begin{aligned}
&p_3\tau_D^2\ddot{u}_{\text{bolus}}(t) + 2p_3\tau_D\dot{u}_{\text{bolus}}(t) + p_3u_{\text{bolus}}(t) \\
&\quad = -A_G\left(\dot{D}(t) + p_2D(t)\right).
\end{aligned} \tag{18}$$

2.4. Kalman Filter for Type-1 Diabetes System. Since the interstitial insulin activity, the amounts of glucose in compartments 1 and 2 cannot be measured directly during implementation, Kalman filter was used to provide an estimation of the state variables from the blood glucose level. The discretized version of the type-1 diabetes system can be written in the following form:

$$\begin{aligned}
\mathbf{x}_K(k+1) &= \mathbf{A}_K\mathbf{x}_K(k) + \mathbf{B}_K\mathbf{u}_K(k) + \mathbf{H}_Kw(k), \\
y_K(k) &= \mathbf{C}_K\mathbf{x}_K(k) + v(k),
\end{aligned} \tag{19}$$

where $\mathbf{x}_K(k) = \begin{bmatrix} D_1 & D_2 & g(k) - g_{\text{d}}(k) & \chi(k) \end{bmatrix}^T$, $\mathbf{u}_K(k) = \begin{bmatrix} D(k) & i(k) \end{bmatrix}^T$, and matrices $\mathbf{A}_K$, $\mathbf{B}_K$, $\mathbf{C}_K$ are linearized coefficient matrices of the model:

$$\begin{aligned}
\mathbf{A}_K &= \begin{bmatrix} -1/\tau_D & 0 & 0 & 0 \\ 1/\tau_D & -1/\tau_D & 0 & 0 \\ 0 & 1/\tau_D & -p_1 - g_{\text{d}} & 0 \\ 0 & 0 & 0 & -p_2 \end{bmatrix}, \\
\mathbf{B}_K &= \begin{bmatrix} A_G & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & p_3V \end{bmatrix}, \\
\mathbf{C}_K &= \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix},
\end{aligned} \tag{20}$$

matrix $\mathbf{H}_K$ is the noise input matrix: $\mathbf{H}_K = \begin{bmatrix} 0 & 0 & 0 & p_3V \end{bmatrix}^T$, the output value $y_K(k) = g(k) - g_{\text{d}}(k)$ is the measured blood glucose deviation from the desired level, $w(k)$ is the insulin input noise, and $v(k)$ is the blood glucose measurement noise with zero-mean Gaussian distribution. The variances of $w(k)$ and $v(k)$ are assumed to be as follows:

$$\begin{aligned}
E\left(w^2(k)\right) &= R_w, \\
E\left(v^2(k)\right) &= R_v.
\end{aligned} \tag{21}$$

Table 1: Parameters and constants of the insulin-glucose kinetics model.

Name	Description	Value
p_1	Glucose effectiveness	$0.2\,\text{min}^{-1}$
p_2	Insulin sensitivity	$0.028\,\text{min}^{-1}$
p_3	Insulin rate of clearance	$10^{-4}\,\text{min}^{-1}$
A_G	Carbohydrate bioavailability	$0.8\,\text{min}^{-1}$
τ_D	Glucose absorption constant	10 min
V	Plasma volume	2730 g
i_{be}	Equilibrium basal plasma insulin concentration	$7.326\,\mu$IU/ml

Table 2: Variables of the insulin-glucose kinetics model.

Name	Description	Unit
D	Amount of CHO intake	mmol/min
D_1	Amount of glucose in compartment 1	mmol
D_2	Amount of glucose in compartment 2	mmol
$g(t)$	Plasma glucose concentration	mmol/l
$\chi(t)$	Interstitial insulin activity	min^{-1}
$i(t)$	Plasma insulin concentration	μIU/ml

Based on the discretized model, a Kalman filter was implemented through the following equation:

$$\hat{\mathbf{x}}(k+1 \mid k) = \mathbf{A}_k \cdot \hat{\mathbf{x}}(k \mid k-1) + \mathbf{B}_k \cdot \mathbf{u}_K(k) + \mathbf{L}[y(k) - \mathbf{C} \cdot \hat{\mathbf{x}}(k \mid k-1)], \tag{22}$$

where $\hat{\mathbf{x}}(k+1 \mid k)$ denotes the estimation of $\mathbf{x}(k+1)$ based on measurements available at time k. The gain $\mathbf{L}$ is the steady-state Kalman filter gain, which can be calculated by

$$\mathbf{L} = \mathbf{M}\mathbf{C}^T\left(\mathbf{C}\mathbf{M}\mathbf{C}^T + R_v\right)^{-1}, \tag{23}$$

where $\mathbf{M}$ is the solution of the corresponding algebraic Riccati equation [13, 14, 18]:

$$\mathbf{M} = \mathbf{A}\mathbf{M}\mathbf{A}^T + \mathbf{B}R_w B - \mathbf{A}\mathbf{M}\mathbf{C}^T\left(\mathbf{C}\mathbf{M}\mathbf{C}^T + \mathbf{R}\right)^{-1}\mathbf{C}\mathbf{M}\mathbf{A}^T. \tag{24}$$

By assuming the noise variances to be $R_w = R_v = 0.01$, the Kalman filter gain was calculated from equation (23) as

$$\mathbf{L} = \begin{bmatrix} 0 & 0 & 8.32 \cdot 10^{-4} & -6.40 \cdot 10^{-7} \end{bmatrix}. \tag{25}$$

2.5. Simulation Setup. First, a pretraining of the algorithm was conducted on the type-1 diabetes model in the scenario where the patient is in a fasting condition (without food intake). The purpose of the pretraining simulation is to obtain an initial estimation of the action-value function for the algorithm. The learning process was conducted by repeating the experiment multiple times (episodes). Each episode starts with an initial blood glucose of 90 mg/dL and ends after 30 minutes. The objective of the algorithm is to search and explore actions that can drive the blood glucose to its target level of 80 mg/dL.

By using the initial estimation of the action-value function, the controller was then tested in the daily scenario with food intake. Comparisons were made between the proposed reinforcement learning with the feedforward (RLFF) controller, the optimal RL (ORL) controller [15], and the proportional-integral-derivative (PID) controller. The ORL was designed with the same parameters and pretrained in the same scenario as with the RLFF. The PID controller gains were chosen, which produces a similar blood glucose settling time as the RLFF:

$$u_k = K_p\left(g(k) - g_{\mathrm{d}}(k)\right) + K_i \sum_k \left(g(k) - g_{\mathrm{d}}(k)\right) + K_d\left(g(k) - g(k-1)\right), \tag{26}$$

where

$$\begin{aligned} K_p &= 1, \\ K_i &= 0.001, \\ K_d &= 0.01. \end{aligned} \tag{27}$$

In order to understand the effects of different food types on the controlled system, two sets of simulations were conducted for food that has slow and fast glucose absorption rates while containing a similar amount of carbs. Absorption rates in the simulations are characterized by parameter τ_D from the model, where $\tau_D = 50$ corresponds to food with a slow absorption rate and $\tau_D = 10$ corresponds to food with a fast absorption rate. The amount of carbohydrate (CHO) per meal can be found in Figure 2.

Next, the performance of the proposed controller was evaluated under uncertainties of meal information. Two cases of uncertainties were considered: uncertain CHO estimation case and uncertain meal-recording time. In the uncertain CHO estimation, the estimated CHO information that provided to the controller was assumed to be a normal distribution with a standard deviation of 46% from the correct carbohydrate value shown in Figure 2. The standard deviation value was used based on the average adult estimates and the computerized evaluations by the dietitian [19]. For the uncertain meal-recording time, the estimated meal starting time is assumed to be a normal distribution with a standard deviation of two minutes from the real starting time. This standard deviation value was randomly selected because systematic research on the accuracy of meal-time recording for patients with type-1 diabetes could not be found. For each case, multiple simulations were conducted in the same closed-loop system with its corresponding random variables. From the obtained results, the mean and standard deviation for blood glucose responses at each time point will be calculated and analyzed.

3. Results

After pretraining in the no-meal scenario, the Q-function was estimated as follows:

$$Q_k^{\pi}(\mathbf{x}, u) = \begin{bmatrix} \mathbf{x}_k^T & u_k^T \end{bmatrix} \begin{bmatrix} 4.454 \cdot 10^2 & -8.870 \cdot 10^4 & -0.084 \\ -8.870 \cdot 10^4 & 3.538 \cdot 10^7 & 33.630 \\ -0.084 & 33.630 & 0.010 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{x}_k \\ u_k \end{bmatrix}. \tag{28}$$

The initial basal policy was obtained from the initial Q-function and equation (12):

$$u_{\mathrm{basal}}(k) = 8.86\left(g(k) - 80\right) - 3534.11\chi(k) + 7.326. \tag{29}$$

The initial estimation of the Q-function and the initial basal policy were used for subsequent testing simulations of the control algorithm.

During the simulation with correct meal information, blood glucose responses of the RLFF, the ORL, and the PID are shown in Figures 3 and 4. The insulin concentration during the process can also be found in Figures 5 and 6. With slow-absorption food, the fluctuation range of blood glucose was approximately ±30 mg/dL for all three controllers from the desired value (Figure 3). However, with fast absorption glucose meals, the fluctuation range of the postmeal blood glucose level was within ±40 mg/dL with the RLFF compared to ±60 mg/dL with the ORL and is significantly smaller than the fluctuation range ±80 mg/dL of the PID (Figure 4).

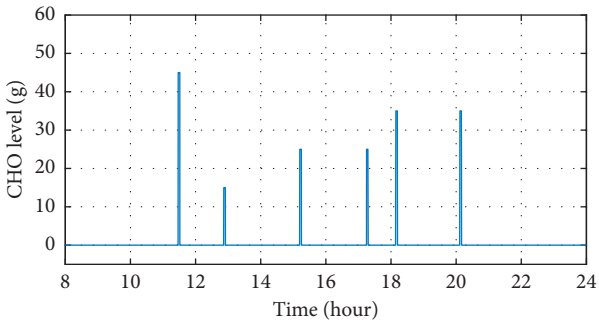

Figure 2: CHO consumed throughout the day.

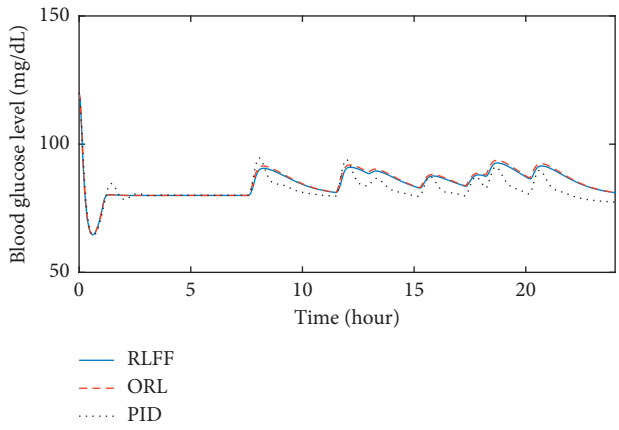

Figure 3: Comparison of the blood glucose responses in the nominal condition for slow glucose absorption food.

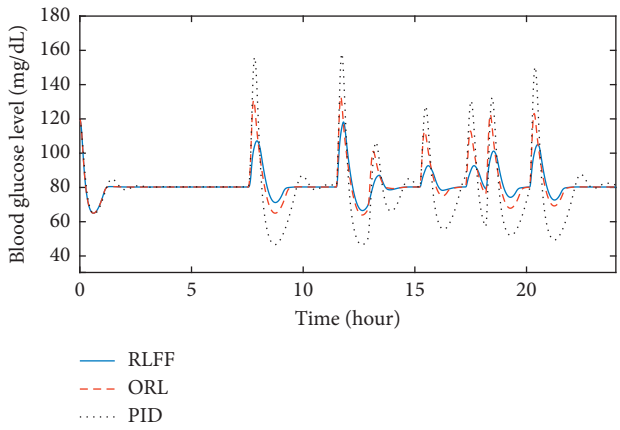

Figure 4: Comparison of the blood glucose responses in the nominal condition for fast glucose absorption food.

Figures 7 and 8 show the blood glucose variation under uncertain meal time and CHO counting. The upper and lower bounds in shaded areas show the mean blood glucose value plus and minus the standard deviation for each instance. Under uncertain meal information, the upper bound was kept to be smaller than 40 mg/dL from the desired blood glucose value for fast glucose absorption food and 15 mg/dL for slow glucose absorption food. The lower bound is smaller than 15 mg/dL from the desired value for

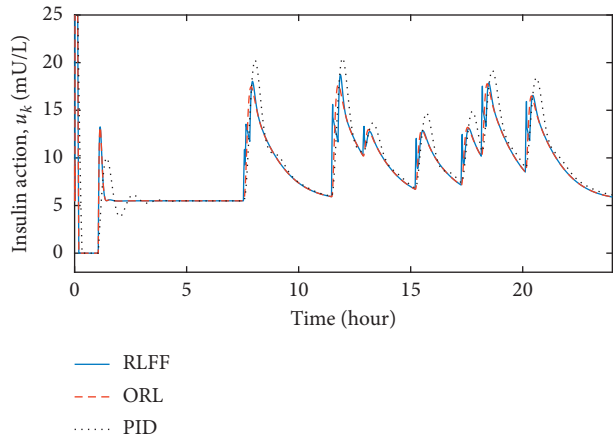

Figure 5: Comparison of insulin concentrations in the nominal condition for slow glucose absorption food.

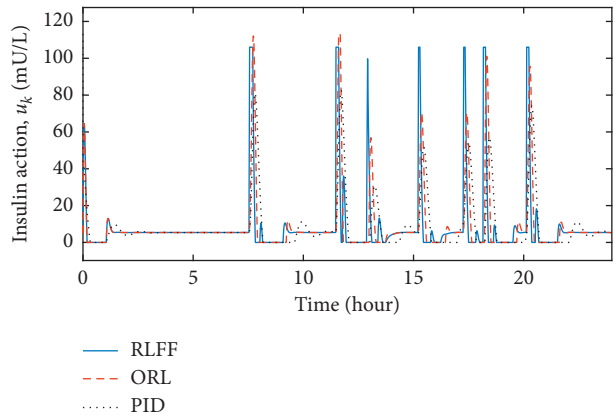

Figure 6: Comparison of insulin concentrations in the nominal condition for fast glucose absorption food.

fast glucose absorption food and 5 mg/dL for slow glucose absorption food.

4. Discussion

The controller has shown its capability to reduce the rise of postmeal blood glucose in our simulations. It can be seen in Figures 3 and 4 that three controllers were able to stabilize the blood glucose. However, when using the RLFF, the added bolus makes the insulin responses much faster when there is a change in blood glucose level, which reduces the peak of the postmeal glucose rise by approximately 30 percent compared to the ORL and 50 percent compared to the PID in the fast-absorption case. It can also be seen that the undershoot blood glucose (the distance between the lowest blood glucose and the desired blood glucose value) of the PID controller is much larger than that of the RLFF and the ORL. The RLFF has the smallest glucose undershoot among the three controllers. Low blood glucose value (hypoglycemia) can be very dangerous for patients with type-1 diabetes. Therefore, simulation results show the advantage of using RLFF in improving safety for patients. In general, with the feedforward algorithm, the proposed algorithm is an

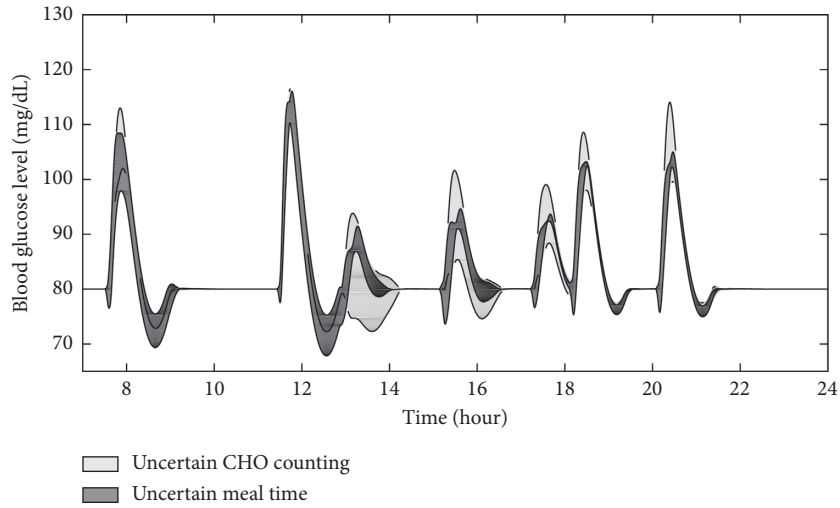

Figure 7: Blood glucose responses under uncertainties for fast glucose absorption food.

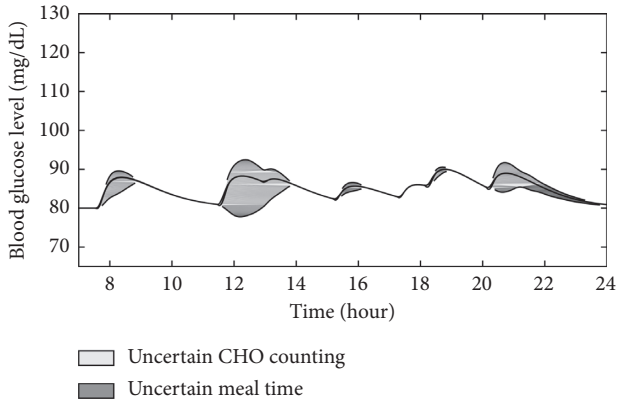

Figure 8: Blood glucose responses under uncertainties for slow glucose absorption food.

effective tool for countering the effects of external events such as meal intake.

Among uncertainties, carb counting created more effect on the variation of the blood glucose than meal-time recording, especially with slow absorbing food. The uncertainty in recording meal time may also lead to larger undershot of blood glucose below the desired level as can be seen in Figure 7. Following the same trend as previous simulations, the fluctuation range of the blood glucose with slow absorbing food is smaller than the fluctuation range with fast glucose absorbing food. In general, the control algorithm kept the blood glucose at the healthy level although uncertainties affect the variation of the responses. However, an accurate carbohydrate counting and accurate meal-time recording method are still important for the purpose of blood glucose control in order to completely avoid the chance of getting hypoglycemia.

5. Conclusion

The paper proposes a blood glucose controller based on reinforcement learning and feedforward algorithm for type-1 diabetes. The controller regulates the patient's glucose level using both basal and bolus insulin. Simulation results of the proposed controller, the optimal reinforcement learning, and the PID controller on a type-1 diabetes model show that the proposed algorithm is the most effective algorithm. The basal updates can stabilize the blood glucose, and the bolus can reduce the glucose undershoot and prevent hypoglycemia. Comparison of the blood glucose variation under different uncertainties provides understandings of how the accuracy of carbohydrate estimation and meal-recording time can affect the closed-loop responses. The results show that the control algorithm was able to keep the blood glucose at a healthy level although uncertainties create variations in the blood glucose responses.

Appendix

Blood Glucose Model

In this paper, the insulin-glucose process was used as the subject in our simulations. The model is described by the following equations [20–23]:

$$\frac{dD_1(t)}{dt} = A_G D(t) - \frac{D_1(t)}{\tau_D}, \tag{A.1}$$

$$\frac{dD_2(t)}{dt} = \frac{D_1(t)}{\tau_D} - \frac{D_2(t)}{\tau_D}, \tag{A.2}$$

$$\frac{dg(t)}{dt} = -p_1 g(t) - \chi(t) g(t) + \frac{D_2(t)}{\tau_D}, \tag{A.3}$$

$$\frac{d\chi(t)}{dt} = -p_2 \chi(t) + p_3 V\left(i(t) - u_{\mathrm{be}}\right), \tag{A.4}$$

where variable descriptions and parameter values are given in Tables 1 and 2. In this model, the inputs are the amount of CHO intake D and the insulin concentration $i(t)$. The output of the model is the blood glucose concentration $g(t)$. It is assumed that the blood glucose is controlled by using an

insulin pump, and there is no delay between the administered insulin and the plasma insulin concentration.

Abbreviations

RL: Reinforcement learning
RLFF: Reinforcement learning with feedforward algorithm
ORL: Optimal reinforcement learning
PID: Proportional-integral-derivative
LTI: Linear time-invariant
CHO: Carbohydrate.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

The research has been funded by financial support from Tromsø Forskningsstiftelse. The publication charges for this article have been funded by a grant from the publication fund of UiT, the Arctic University of Norway.

References

[1] Q. Wang, P. Molenaar, S. Harsh et al., "Personalized state-space modeling of glucose dynamics for type 1 diabetes using continuously monitored glucose, insulin dose, and meal intake," *Journal of Diabetes Science and Technology*, vol. 8, no. 2, pp. 331–345, 2014.

[2] G. Marchetti, M. Barolo, L. Jovanovic, H. Zisser, and D. E. Seborg, "An improved PID switching control strategy for type 1 diabetes," *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 3, pp. 857–865, 2008.

[3] S. Soylu, K. Danisman, I. E. Sacu, and M. Alci, "Closed-loop control of blood glucose level in type-1 diabetics: a simulation study," in *Proceedings of International Conference on Electrical and Electronics Engineering (ELECO)*, pp. 371–375, Bursa, Turkey, November 2013.

[4] D. Boiroux, A. K. Duun-Henriksen, S. Schmidt et al., "Overnight glucose control in people with type 1 diabetes," *Biomedical Signal Processing and Control*, vol. 39, pp. 503–512, 2018.

[5] H. Lee and B. W. Bequette, "A closed-loop artificial pancreas based on model predictive control: human-friendly identification and automatic meal disturbance rejection," *Biomedical Signal Processing and Control*, vol. 4, no. 4, pp. 347–354, 2009.

[6] M. K. Bothe, L. Dickens, K. Reichel et al., "The use of reinforcement learning algorithms to meet the challenges of an artificial pancreas," *Expert Review of Medical Devices*, vol. 10, no. 5, pp. 661–673, 2014.

[7] M. De Paula, L. O. Ávila, and E. C. Martínez, "Controlling blood glucose variability under uncertainty using reinforcement learning and Gaussian processes," *Applied Soft Computing*, vol. 35, pp. 310–332, 2015.

[8] C. J. C. H. Watkins and P. Dayan, "Technical note: Q-learning," in *Reinforcement Learning*, pp. 55–68, vol. 292Springer US, Boston, MA, USA, 1992.

[9] J. Pineau, M. G. Bellemare, A. J. Rush, A. Ghizaru, and S. A. Murphy, "Constructing evidence-based treatment strategies using methods from computer science," *Drug and Alcohol Dependence*, vol. 88, no. S2, pp. S52–S60, 2007.

[10] K. Lunze, T. Singh, M. Walter, M. D. Brendel, and S. Leonhardt, "Blood glucose control algorithms for type 1 diabetic patients: a methodological review," *Biomedical Signal Processing and Control*, vol. 8, no. 2, pp. 107–119, 2013.

[11] S. P. Bhattacharyyta, "Disturbance rejection in linear systems," *International Journal of Systems Science*, vol. 5, no. 7, pp. 633–637, 1974.

[12] H. Zhong, L. Pao, and R. de Callafon, "Feedforward control for disturbance rejection: model matching and other methods," in *Proceedings of 24th Chinese Control and Decision Conference (CCDC)*, pp. 3528–3533, Taiyuan, China, May 2012.

[13] F. Lewis, *Optimal Estimation*, John Wiley & Sons, Inc., Hoboken, NJ, USA, 1986.

[14] G. F. Franklin, J. D. Powell, and M. L. Workman, *Digital Control of Dynamic Systems*, Addison-Wesley, Boston, MA, USA, 2nd edition, 1990.

[15] D. Vrabie, K. G. Vamvoudakis, and F. L. Lewis, *Optimal Adaptive Control and Differential Games by Reinforcement Learning Principles*, Institution of Engineering and Technology, vol. 81, 1st edition, 2012.

[16] P. D. Ngo, S. Wei, A. Holubova, J. Muzik, and F. Godtliebsen, "Reinforcement-learning optimal control for type-1 diabetes," in *Proceedings of 2018 IEEE EMBS International Conference on Biomedical & Health Informatics (BHI)*, pp. 333–336, Las Vegas, NV, USA, March 2018.

[17] R. Sutton and A. Barto, *Reinforcement Learning: An Introduction*, MIT Press, Cambridge, MA, USA, 1st edition, 1998.

[18] MathWorks, *MATLAB Optimization Toolbox: User's Guide (r2018a)*, MathWorks, Natick, MA, USA, 2018.

[19] A. S. Brazeau, H. Mircescu, K. Desjardins et al., "Carbohydrate counting accuracy and blood glucose variability in adults with type 1 diabetes," *Diabetes Research and Clinical Practice*, vol. 99, no. 1, pp. 19–23, 2013.

[20] R. N. Bergman, Y. Z. Ider, C. R. Bowden, and C. Cobelli, "Quantitative estimation of insulin sensitivity," *American Journal of Physiology-Endocrinology and Metabolism*, vol. 236, no. 6, p. E667, 1979.

[21] R. Hovorka, V. Canonico, L. J. Chassin et al., "Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes," *Physiological Measurement*, vol. 25, no. 4, pp. 905–920, 2004.

[22] M. E. Wilinska, L. J. Chassin, H. C. Schaller, L. Schaupp, T. R. Pieber, and R. Hovorka, "Insulin kinetics in type-1 diabetes: continuous and bolus delivery of rapid acting insulin," *IEEE Transactions on Biomedical Engineering*, vol. 52, no. 1, pp. 3–12, 2005.

[23] A. Mösching, *Reinforcement Learning Methods for Glucose Regulation in Type 1 Diabetes*, Ecole Polytechnique Federale de Lausanne, Lausanne, Switzerland, 2016.