mathematics

MDPI

Article

Proof of Concept Control of a T1DM Model Using Robust Fixed-Point Transformations via Sliding Mode Differentiators

Bence Czakó *,†, Dániel András Drexler † and Levente Kovács †

Physiological Controls Research Center, Óbuda University, 1034 Budapest, Hungary
* Correspondence: czako.bence@uni-obuda.hu
† These authors contributed equally to this work.

Abstract: Type 1 Diabetes Mellitus (T1DM) is a disease where insulin production is obstructed in the pancreas, and exogenous administration of the hormone must be utilized. Automatic control of the administration can be achieved using the Artificial Pancreas (AP) concept, whose performance is heavily reliant on the underlying control algorithm. A Robust Fixed-Point Transformations (RFPT)-based control strategy was designed to automate the insulin delivery process, which incorporates a Sliding Mode Differentiator (SMD) to provide higher order derivatives of the blood glucose level. Inter-patient variability, carbohydrate disturbances, and real-life sampling were included in the validation of the method. Results showed that the algorithm could regulate the blood glucose level, with a significant overshoot at the beginning of the control action due to the adaptive nature of the controller. Results indicate that the design requires additional modifications to be feasible in practice, including an extended validation with more virtual patients and realistic simulation settings in the future. Nevertheless, the current control algorithm has several attractive features, which are discussed with respect to PID and Model Predictive Control (MPC).

Keywords: type 1 diabetes mellitus; artificial pancreas; automated insulin delivery; robust fixed-point transformations; sliding mode differentiators

MSC: 93-10

Citation: Czakó, B.; Drexler, D.A.; Kovács, L. Proof of Concept Control of a T1DM Model Using Robust Fixed-Point Transformations via Sliding Mode Differentiators. *Mathematics* **2023**, *11*, 1210. https://doi.org/10.3390/math11051210

Academic Editor: Biao Tang

Received: 20 January 2023
Revised: 22 February 2023
Accepted: 27 February 2023
Published: 1 March 2023

1. Introduction

Automatic control of physiological systems is one of the most challenging areas of control theory. In physiological systems, measurements and input actions often have significant limitations, while many sources of disturbance might be apparent, which are hard to characterize or compensate for. Some successful applications of physiological control include the optimization of drug delivery, automated anesthesia, or the automated treatment of Diabetes Mellitus (DM).

DM is one of modern society's most prevalent diseases, with an estimated 536.6 million cases in 2021, projected to increase to 783.2 million cases in 2045 [1]. In general, DM is a collection of diseases with insufficient insulin production by the pancreas in response to carbohydrate (CH) intake. DM has several different manifestations, each originating from a specific biological factor. Of all the possible variants, type 1 (T1DM) and type 2 (T2DM) DM have the most significant prevalence, thus being the target of an automated solution. While T1DM is an inherited autoimmune disease, where the malfunction of the pancreas starts at an early age, T2DM is a disease connected to the patient's lifestyle. A significant difference between the two conditions is that T1DM cannot be cured completely (currently), but T2DM could be reversed in some instances by introducing measures that target unhealthy lifestyle choices.

As a consequence, early attempts to implement an automated scheme targeted the T1DM condition, due to the limited number of interplaying factors that lead to the disease

compared to the T2DM variant [2]. A significant achievement in this regard was the development of the Artificial Pancreas (AP) concept, where an external device controls the exogenous insulin administration instead of relying on the self-administration of the patient by an insulin pen [3]. Such a device can provide tighter blood glucose (BG) regulation, thus leading to an improved quality of life for the patient [4]. The two main physical components of the AP concept are the Continuous Glucose Monitoring Sensor (CGMS), providing essentially real-time measurements on the BG level of the patient, and an insulin pump, which can administer the insulin in a precise manner [5]. In order to automate the process, a sophisticated control algorithm must also be implemented, which calculates the insulin administration based on the measured BG levels. The implementation of such an algorithm is difficult in practice, which can be attributed to the unique physiological characteristics of the patient, as well as the absence of information on the CH intake, which presents a disturbance in the control operation. Numerous algorithms exist in the literature, in which PID-control- and Model-Predictive-Control-based approaches were tested in clinical trials [6,7]. Nevertheless, a large number of other alternatives were also investigated in the research community, including Sliding Mode Control [8], Tensor-Product-based control [9], or fuzzy control [10]. A comprehensive review of the available control algorithms can be found in the surveys [11,12].

In our current research, the application of the Robust Fixed Point Transformation (RFPT)-based control was investigated on the T1DM problem. The method can be classified as an adaptive nonlinear control method, where a fixed-point iteration eliminates parametric errors and other disturbances arising from the Input–Output (IO) linearization principle [13,14]. Using the RFPT in the context of T1DM control was reported in [15,16] and [17]. These works were proof-of-concept studies, with several limitations regarding the possible real-life application.

Since the RFPT method relies on the IO linearization principle, a major issue that must be addressed is the acquisition of high-order derivatives of the measured variable, namely the BG level. Estimating the derivatives is a highly nontrivial task and has been one of the fundamental issues in control theory. One possible approach is the use of Sliding Mode Differentiators (SMD), which can provide higher-order derivatives of online sampled signals, developed initially by [18]. Both continuous and discrete time formulations are available in the approach of [19,20]. The algorithm uses a dynamical system to estimate the derivatives of a given signal, where each state variable represent a derivative order. When the system is solved, the exact values of the derivatives are obtained in finite time due to the discontinuity in the structure of the ODEs.

In the current paper, we aim to develop an RFPT controller which is validated through simulations, where multiple real-life factors are addressed. Our novelty in this work is that the design of the controller is based on a rigorous theoretical foundation developed in previous papers [15,21], rather than ad hoc observations. Additionally, the controller is also augmented with an SMD, which can provide higher-order derivatives of online sampled signals. Moreover, this is the first time in the literature that the SMD is connected to the RFPT methodology, which can be beneficial for the future development of the control technique. The implementation of the algorithm was based on an identifiable virtual patient (IVP) model, introduced in [22], and the design was tested using virtual patients in silico, where parametric uncertainty, CH disturbance, and sampling time were also considered.

In Section 2, preliminaries of the IO linearization principle are presented in conjunction with the RFPT method and the SMD. In Section 3, the application of the theory is shown on the IVP model and the controller design is performed. In Section 4, simulation results of the closed-loop system are presented, with additional analysis on the achieved results. Conclusions and future research directions are presented in Section 5.

2. Materials and Methods

The RFPT method is an extension of the IO linearization principle, where an iteration aims to eliminate the mismatch between the predicted output of the system model and

its measurement through an additional dynamic variable. Convergence of the iteration eventually results in the original IO linearized version of the controller. In the original IO linearization, a state transformation is calculated, which renders the original system into a chain of integrators and a scalar nonlinear differential equation in single input, single output (SISO) cases. The input signal is then defined so that the nonlinear terms cancel out in the nonlinear equation, and the system is fully linearized through a virtual input. This linear system can then be controlled using traditional linear control strategies, such as PID or state feedback control. Nevertheless, when the system parameters are inexact or other disturbances are present during the closed-loop control, the linearization fails, resulting in unstable behavior for most systems. RFPT aims to bridge this difference by utilizing a discrete-time iteration, such that the linearization is preserved and the operation of the linear controller is maintained.

This section shows the theoretical background of the method for continuous SISO systems, highlighting its benefits and drawbacks. We first introduce the IO linearization principle based on the exposition of [23].

2.1. Input–Output Linearization

A general nonlinear SISO input-affine system is given by

$$\begin{aligned} \dot{x} &= f(x) + g(x)u \\ y &= h(x) \end{aligned} \tag{1}$$

where $x = x(t) \in \mathbb{R}^n$ is the state of the system at time t, $u = u(t) \in \mathbb{R}$ is the input of the system at time t, while $y = y(t) \in \mathbb{R}$ is the output of the system at time t, and $t \in \mathbb{R}^+$, and f, g, h are continuously differentiable vector fields. The output of the system is differentiated, which leads to

$$\dot{y} = \frac{\partial h}{\partial x}\dot{x} = \frac{\partial h}{\partial x}[f(x) + g(x)u] \triangleq L_f h + L_g h\, u \tag{2}$$

where $L_f h$ and $L_g h$ are the Lie derivatives of h along the vector fields f and g, respectively. This notation has the property

$$\begin{aligned} L_f^0 h &= h \\ L_f^k h &= L_f L_f^{k-1} h = \frac{\partial (L_f^{k-1} h)}{\partial x} f, \quad k \geq 1. \end{aligned} \tag{3}$$

Assuming that the input u of the system appears in the r-th derivative, the following condition can be formulated

$$L_g L_f^{r-1} h = \frac{\partial (L_f^{r-1} h)}{\partial x} g \neq 0, \tag{4}$$

where r is the relative degree of the system. The relative degree r $(1 \leq r \leq n)$ of a nonlinear input-affine system in a region $D_0 \subset D$ exists if

$$\begin{aligned} &L_g L_f^{i-1} h(x) = 0,\ i = 1, ..., r-1 \\ &L_g L_f^{r-1} h(x) \neq 0 \end{aligned} \tag{5}$$

for all $x \in D_0$. It is also assumed that the relative degree equals the state dimension, such that $r = n$. This assumption implies that no zero dynamics are present in the transformed system. If this assumption is not satisfied, the controller can still provide adequate closed-

loop performance when the zero dynamics are stable. As a consequence, the coordinate transform is given as

$$\begin{pmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_n \end{pmatrix} = \mathcal{T}(x) = \begin{pmatrix} h(x) \\ L_f h(x) \\ \vdots \\ L_f^{r-1} h(x) \end{pmatrix}, \tag{6}$$

where $\mathcal{T} : D_x \subset \mathbb{R}^n \to D_\xi \subset \mathbb{R}^n$ must be a diffeomorphism. Using this transformation, a normal form can be obtained

$$\begin{aligned} \dot{\xi}_1 &= \xi_2 \\ \dot{\xi}_2 &= \xi_3 \\ &\vdots \\ \dot{\xi}_{n-1} &= \xi_n \\ \dot{\xi}_n &= \alpha(\xi) + \beta(\xi) u \\ y &= \xi_1. \end{aligned} \tag{7}$$

In (7), the system is divided into a series of integrators and a nonlinear differential equation with the components

$$\begin{aligned} \alpha(\xi) &= [L_f^n h(x)]_{x=\mathcal{T}_c^{-1}(\xi)} \\ \beta(\xi) &= [L_g L_f^{n-1} h(x)]_{x=\mathcal{T}_c^{-1}(\xi)}. \end{aligned} \tag{8}$$

The nonlinear term $\dot{\xi}_n$ in (7) can then be used to define the input as

$$u = \frac{v - \alpha(\xi)}{\beta(\xi)}, \tag{9}$$

which eliminates the nonlinearities, from which $\dot{\xi}_n = v$, where v is a virtual input. By combining equations (7) and (9), the linear system

$$\begin{aligned} \dot{\xi} &= A\xi + Bv \\ y &= C\xi \end{aligned} \tag{10}$$

is obtained, which can be stabilized in the origin using state feedback $v = -K\xi$, where K is determined either by LQR control or pole placement. In practice, (9) is often obtained from an uncertain model with vector fields f^*, g^*, h^*, which have a similar structure to the vector fields in (1), thus the equilibrium remains the same qualitatively in both cases. By combining (7) and (9), the following identity emerges

$$\dot{\xi}_n = \alpha(\xi) + \beta(\xi) \left(\frac{v - \alpha^*(\xi)}{\beta^*(\xi)} \right) \tag{11}$$

with the modified affine terms

$$\begin{aligned} \alpha^*(\xi) &= [L_{f^*}^n h^*(x)]_{x=\mathcal{T}_c^{-1}(\xi)} \\ \beta^*(\xi) &= [L_{g^*} L_{f^*}^{n-1} h^*(x)]_{x=\mathcal{T}_c^{-1}(\xi)}. \end{aligned} \tag{12}$$

This does not lead to a linear system anymore; hence, it requires further compensation, for which the RFPT provides an iterative solution.

2.2. Robust Fixed-Point Transformation

The normal form can be discretized with the Euler method, for example

$$\xi_n^+ = \xi_n + \Delta t\left[\alpha(\boldsymbol{x}) + \beta(\boldsymbol{x})\left(\frac{v - \alpha^*(\boldsymbol{x})}{\beta^*(\boldsymbol{x})}\right)\right], \tag{13}$$

where $\Delta t > 0$, $\xi := \xi(k\Delta t)$ and $\xi^+ := \xi((k+1)\Delta t), k \in \mathbb{N}_0$. The following requirement is made on (11) to eliminate the nonlinearities

$$\mathcal{E}(g) \equiv \alpha(\boldsymbol{\xi}) + \beta(\boldsymbol{\xi})\left(\frac{g - \alpha^*(\xi)}{\beta^*(\xi)}\right) = v, \tag{14}$$

where $v = -K\xi$ is the linear state feedback control and g is a new input which should satisfy (14). One can define a fixed-point iteration to find the root g_* of (14) through the associated equation $\mathcal{E}(g) - v + g = g$, with the additional property that $\mathcal{E}(g_*) = g_* = v$. The convergence of the iteration can be ensured on a closed domain $g \in [a, b] \in R$, if the inequality

$$\left|\left|\frac{\mathrm{d}\mathcal{E}(g)}{\mathrm{d}g} + 1\right|\right| < 1 \tag{15}$$

can be satisfied on the whole region, which is often not the case due to the structural properties of $\mathcal{E}$. These properties of $\mathcal{E}$ can be altered by introducing the so-called deform function

$$\mathcal{G}(\mathcal{E}(g_i) - v, g_i, \theta) = g_{i+1}, \tag{16}$$

where the function is dependent on the error $\mathcal{E}(g_i) - v$, the iteration variable g_i, and the design parameters $\theta \in \mathbb{R}^m$. An initial condition g_0 for the iteration must also be supplied. One must impose the additional property

$$\mathcal{G}(\mathcal{E}(g_*) - v, g_*, \theta) = g_* \rightarrow \mathcal{E}(g_*) = v \tag{17}$$

such that the fixed-point solution indeed coincides with the linear controller v. If, for some θ, the closed interval convergence condition can be satisfied, i.e.,

$$\left|\left|\frac{\mathrm{d}\mathcal{G}(\mathcal{E}(g) - v, g, \theta)}{\mathrm{d}g}\right|\right| < 1 \tag{18}$$

then the iteration $g^+ = \mathcal{G}(\mathcal{E}(g) - v, g, \theta)$ will converge to g_* due to the Banach fixed-point theorem. Since $\mathcal{E}(g)$ can only be accessed through direct measurement on the r-th derivative of the output of the process, the iteration is performed online when measurements are available, i.e., $g = g_k = g(k\Delta t)$. The normal form augmented with the deform function is given as

$$\begin{aligned}
\xi_1^+ &= \xi_1 + \Delta t\xi_2 \\
&\vdots \\
\xi_n^+ &= \xi_n + \Delta t\left[\alpha(\boldsymbol{\xi}) + \beta(\boldsymbol{\xi})\left(\frac{g - \alpha^*(\boldsymbol{\xi})}{\beta^*(\boldsymbol{\xi})}\right)\right] \\
g^+ &= \mathcal{G}(\mathcal{E}(g) - v, g, \theta) \\
v &= -K\xi \\
y &= \xi_1.
\end{aligned} \tag{19}$$

By adequately tuning θ, the sequence $g \rightarrow g^*$ renders the system essentially linear in a fixed number of steps from which the linear controller can finish the control task. In practice, when $|g^+ - g^*| < \epsilon$, where ϵ is a small number, then $g^+ = g$, since the iteration is in the vicinity of the fixed point. We note that the requirement (18) is similar to the linearization theorem for nonlinear discrete-time systems.

As aforementioned, $\mathcal{E}(g)$ can only be accessed by directly measuring the r-th derivative of the system's output. Nonetheless, this is infeasible in practice, such that an estimation must be constructed from the output signal directly, for which Sliding Mode Differentiators provide a possibility.

2.3. Sliding Mode Differentiator

SMDs originate from the sliding mode control theory, closely related to the IO linearization principle [18]. SMDs can provide exact differentials of a signal $f(t)$ up to order r, if an upper bound L is available such that $|f^{(r+1)}| < L$, where L is the Lipschitz constant. A dynamical system can then be constructed in the form of

$$\begin{aligned}
\dot{z}_0 &= -\tilde{\lambda}_r L^{1/(r+1)} |z_0 - f(t)|^{\frac{r}{r+1}} \operatorname{sign}(z_0 - f(t)) + z_1 \\
\dot{z}_1 &= -\tilde{\lambda}_{r-1} L^{2/(r+1)} |z_0 - f(t)|^{\frac{r-1}{r+1}} \operatorname{sign}(z_0 - f(t)) + z_2 \\
&\vdots \\
\dot{z}_{r-1} &= -\tilde{\lambda}_1 L^{r/(r+1)} |z_0 - f(t)|^{\frac{1}{r+1}} \operatorname{sign}(z_0 - f(t)) + z_r \\
\dot{z}_r &= -\tilde{\lambda}_0 L \operatorname{sign}(z_0 - f(t)) + z_r,
\end{aligned} \tag{20}$$

where $\tilde{\lambda}_r = \lambda_r$, $\tilde{\lambda}_0 = \lambda_0$, and $\tilde{\lambda}_i = \lambda_i \tilde{\lambda}_{i+1}^{i/(i+1)}$. The parameters λ_i are free parameters, often determined empirically. A possible choice for the coefficients of a fifth-order differentiator is $\boldsymbol{\lambda} = [1.1, 1.5, 2, 3, 5, 8]$ [20]. In practice, the larger the coefficient values, the faster the convergence of the system with larger sensitivity for the noise in the function $f(t)$. The initial condition associated with (20) is given as $\boldsymbol{z} = [f(0), 0, \ldots, 0]$. Additional material on SMD can be found in [24].

In the case of the RFPT method, the differentiated function $f(t)$ is the measured BG level $y(t)$ such that the value of $\mathcal{E}(g)$ is estimated by $z_r(t)$, where r is the relative degree. This estimation is essential since $\mathcal{E}(g)$ forms a basis of the error between the output of the system and the linear controller. In the case of T1DM control, there is no possible way to physically measure the r-th derivative of the BG level, only to estimate it, which justifies the use of the SMD. In particular, at each sampling instant k in (19), the system (20) is integrated with initial condition $\boldsymbol{z} = [y(k\Delta t), z_1(k\Delta t^-), \ldots, z_r(k\Delta t^-)]$, where the notation $z_r(k\Delta t^-)$ denotes the endpoint of the previous solution curve of $\boldsymbol{z}$.

In the following, the Identifiable Virtual Patient (IVP) model is presented, on which the control design is based.

3. Controller Design

In order to design the controller, a suitable model is required, which was chosen to be the IVP model which can be found in [25]. The model describes the insulin interactions in the body using four state variables, where the CH intake is represented as a disturbance:

$$\begin{aligned}
\dot{x}_1 &= -\frac{1}{\tau_1} x_1 + \frac{1}{\tau_1} \frac{u}{C_I} \\
\dot{x}_2 &= -\frac{1}{\tau_2} x_2 + \frac{1}{\tau_2} x_1 \\
\dot{x}_3 &= -k_1 x_3 + k_1 S_I x_2 \\
\dot{x}_4 &= -(GEZI + x_3) x_4 + EGP + R_a,
\end{aligned} \tag{21}$$

where x_1 (mg/dL) is the catheter site subcutaneous insulin concentration, x_2 (mg/dL) is the insulin plasma concentration, x_3 (mg/dL) is the insulin effect related to the decrease in plasma glucose, which is denoted by x_4 (mg/dL), R_a (mg/dL/min) is the rate of appearance of glucose following meals, and u (U/min) is the exogenous insulin administration rate. In this model, u is the input signal, while R_a is a disturbance term. The output for this model

is the BG level, $y = x_4$, which can be measured with a CGMS device directly. A description of the model parameters and their corresponding dimensions can be found in Table 1.

Table 1. Model parameters.

Parameter	Unit	Description
τ_1	min	Pharmacokinetic time constant
τ_2	min	Pharmacokinetic time constant
C_I	mL/min	Insulin clearance
k_1	1/min	Time constant of insulin action
S_I	mL/μU/min	Insulin sensitivity
$GEZI$	1/min	Glucose effectiveness at zero insulin
EGP	mg/dL/min	Endogenous glucose production

The model was also augmented with a sensor model, based on the research of [26]. The sensor model is derived from measurements taken with a Dexcom G6 sensor, which can provide BG measurements with a sampling time of 5 min. The sensor model can be decomposed into two parts, namely a fixed error IG_s, originating from calibration imprecision, and a random noise v_k, characterized by a Gaussian white noise. The calibration error is given by the continuous time system

$$\dot{x}_5 = \frac{1}{\tau}(x_4 - x_5) \tag{22}$$

$$x_{5s} = (a_0 + a_1 t + a_2 t^2)x_5 + b_0 \tag{23}$$

where the coefficients were experimentally determined to be $\tau = 3.1$, $a_0 = 1$, $a_1 = 0.002$, $a_2 = 0$, and $b_0 = 7.3$ [26]. The random noise is a discrete-time autocorrelated white noise as

$$v_k = \alpha_1 v_{k-1} + \alpha_2 v_{k-2} + w_k \tag{24}$$

$$w_k, v_0, v_1 \sim \mathcal{N}(0, \sigma), \tag{25}$$

where $\alpha_1 = 1.3$, $\alpha_2 = -0.46$, and $\sigma = 3.2$. Each measurement is then given as

$$\hat{y}_k = x_{5s}(\Delta t k) + v_k, \tag{26}$$

where $\hat{y}_k$ is the noise-corrupted BG measurement directly used in the SMD.

To design the controller, the steps outlined in Section 2 must be followed. First, the relative degree must be determined by differentiating the output $y = x_4$ with respect to time until the input appears explicitly. Here, the relative degree was found to be $r = 4$, since the input is explicitly present in the fourth derivative of the output signal. The second step is the calculation of the coordinate transform $\mathcal{T}$ from (6), which was performed using Wolfram Mathematica due to the complexity of the calculations. The transformation is given by

$$\begin{pmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \\ \xi_4 \end{pmatrix} = \mathcal{T}(\boldsymbol{x}) = \begin{pmatrix} x_4 \\ \gamma_2 - \gamma_1 x_4 \\ \dot{R}_a + k_1\gamma_3 x_4 + \gamma_1(-\gamma_2 + \gamma_4 x_4) \\ \ddot{R}_a + \gamma_5 + \gamma_6 - \gamma_4(\dot{R}_a + \gamma_7) \end{pmatrix} \tag{27}$$

where the auxiliary variables are defined as

$$\begin{aligned}
\gamma_1 &= (GEZI + x_3)\\
\gamma_2 &= EGP + R_a\\
\gamma_3 &= (-S_I x_2 + x_3)\\
\gamma_4 &= (GEZI + x_3)\\
\gamma_5 &= (k_1(S_I(-x_1 + x_2 + k_1\tau_2 x_2) - k_1\tau_2 x_3)x_4)/\tau_2\\
\gamma_6 &= 2k_1(S_I x_2 - x_3)(-b + ax_4)\\
\gamma_7 &= k_1 c x_4 - a(b - ax_4).
\end{aligned} \tag{28}$$

In the following step, the inverse transformation rule $\mathcal{T}^{-1}$ must be calculated. The inverse transformation is given by the vector

$$\begin{pmatrix} x_1\\ x_2\\ x_3\\ x_4 \end{pmatrix} = \mathcal{T}^{-1}(\boldsymbol{x}) = \begin{pmatrix} -\delta_1(\delta_2 + \delta_3 - \delta_4 + \delta_5 + \delta_6 + \delta_7))\\ (\delta_8 - \delta_9)/(k_1 S_I \xi_1^2)\\ \delta_{10}/\xi_1\\ \xi_1 \end{pmatrix} \tag{29}$$

where the auxiliary terms are

$$\begin{aligned}
\delta_1 &= (1/(k_1 S_I \xi_1^3)\\
\delta_2 &= -k_1 R_a \xi_1^2 - \ddot{R}_a \tau_2 \xi_1^2 + GEZIk_1\xi_1^3 + R_a\xi_1\xi_2\\
\delta_3 &= k_1 R_a \tau_2 \xi_1\xi_2 + k_1\xi_1^2\xi_2 - 2R_a\tau_2\xi_2^2 - \xi_1\xi_2^2\\
\delta_4 &= k_1\tau_2\xi_1\xi_2^2 + 2\tau_2\xi_2^3 - \dot{R}_a\xi_1(\xi_1 + k_1\tau_2\xi_1 - 2\tau_2\xi_2)\\
\delta_5 &= R_a\tau_2\xi_1\xi_3 + \xi_1^2\xi_3 + k_1\tau_2\xi_1^2\xi_3 - 3\tau_2\xi_1\xi_2\xi_3\\
\delta_6 &= EGP(\xi_1\xi_2 - 2\tau_2\xi_2^2 + k_1\xi_1(-\xi_1 + \tau_2\xi_2) + \tau_2\xi_1\xi_3)\\
\delta_7 &= \tau_2\xi_1^2\xi_4\\
\delta_8 &= \dot{R}_a\xi_1 + EGPk_1\xi_1 + k_1R_a\xi_1 - GEZIk_1\xi_1^2\\
\delta_9 &= EGP\xi_2 - R_a\xi_2 - k_1\xi_1\xi_2 + \xi_2^2 - \xi_1\xi_3\\
\delta_{10} &= (EGP + R_a - GEZI\xi_1 - \xi_2).
\end{aligned} \tag{30}$$

To calculate the input rule that cancels out the nonlinearities, the terms $\alpha(\boldsymbol{x})$ and $\beta(\boldsymbol{x})$ must be determined, where $\alpha(\boldsymbol{x})$ is described by

$$\begin{aligned}
\alpha(\boldsymbol{x}) = {} & x_4(k_1((x_3 - S_I x_2)k_1^2 + (S_I(x_1 - x_2)k_1)/\tau_2)\\
& + (S_I k_1(C_I\tau_1 x_1 - C_I\tau_1 x_2 + C_I\tau_2 x_1))/(C_I\tau_1\tau_2^2))\\
& - ((x_3 - S_I x_2)k_1^2 + (S_I(x_1 - x_2)k_1)/\tau_2)\\
& (3EGP + 3R_a - 3x_4(GEZI + x_3))\\
& + (GEZI + x_3)(x_4((x_3 - S_I x_2)k_1^2\\
& + (S_I(x_1 - x_2)k_1)/\tau_2)\\
& + (GEZI + x_3)(\dot{R}_a - (GEZI + x_3)\\
& (EGP + R_a - x_4(GEZI + x_3))\\
& + k_1 x_4(x_3 - S_I x_2))\\
& - 2k_1(x_3 - S_I x_2)\\
& (EGP + R_a - x_4(GEZI + x_3)) - \ddot{R}_a)\\
& - 3k_1(x_3 - S_I x_2)(\dddot{R}_a - \dot{R}_a + (GEZI + x_3)\\
& (EGP + R_a - x_4(GEZI + x_3))\\
& - k_1 x_4(x_3 - S_I x_2)),
\end{aligned} \tag{31}$$

while $\beta(\boldsymbol{x})$ is given as

$$\beta(\boldsymbol{x}) = (-(S_I k_1 x_4)/(C_I \tau_1 \tau_2)). \tag{32}$$

Using (9), $\alpha(\boldsymbol{x})$, and $\beta(\boldsymbol{x})$, the transformed input has the structure

$$\begin{aligned} u = {} & (1/(k_1 S_I x_4))(CI\tau_1\tau_2(-g - (k_1(S_I((-\tau_2)x_1 \\ & + \tau_1(-x_1 - k_1\tau_2 x_1 + x_2 + k_1\tau_2 x_2 + k_1^2\tau_2^2 x_2)) \\ & - k_1^2\tau_1\tau_2^2 x_3)x_4)/(\tau_1\tau_2^2) + 3k_1(S_I x_2 - x_3) \\ & (\ddot{R}_a - \dot{R}_a + EGPGEZI + GEZIR_a + EGPx_3 \\ & + R_a x_3 - GEZI^2 x_4 + k_1 S_I x_2 x_4 \\ & - 2GEZIx_3x_4 - k_1 x_3 x_4 - x_3^2 x_4) + (1/\tau_2) \\ & ((GEZI + x_3)((-\ddot{R}_a)\tau_2 - EGPGEZI^2\tau_2 \\ & - GEZI^2 R_a\tau_2 + 2EGPk_1 S_I\tau_2 x_2 + 2k_1 R_a S_I\tau_2 x_2 \\ & - 2EGPGEZI\tau_2 x_3 - 2EGPk_1\tau_2 x_3 \\ & - 2GEZIR_a\tau_2 x_3 - 2k_1 R_a\tau_2 x_3 - EGP\tau_2 x_3^2 \\ & - R_a\tau_2 x_3^2 + \dot{R}_a\tau_2(GEZI + x_3) + GEZI^3\tau_2 x_4 \\ & + k_1 S_I x_1 x_4 - k_1 S_I x_2 x_4 - 3GEZIk_1 S_I\tau_2 x_2 x_4 \\ & - k_1^2 S_I\tau_2 x_2 x_4 + 3GEZI^2\tau_2 x_3 x_4 \\ & + 3GEZIk_1\tau_2 x_3 x_4 + k_1^2\tau_2 x_3 x_4 - 3k_1 S_I\tau_2 x_2 x_3 x_4 \\ & + 3GEZI\tau_2 x_3^2 x_4 + 3k_1\tau_2 x_3^2 x_4 + \tau_2 x_3^3 x_4)) \\ & - (3k_1(S_I(-x_1 + x_2 + k_1\tau_2 x_2) - k_1\tau_2 x_3) \\ & (-EGP - R_a + (GEZI + x_3)x_4))/\tau_2)). \end{aligned} \tag{33}$$

Note that instead of v, the deformed variable g appears in this equation as the virtual input (immediately in the first line). In some instances, the input signal u might become negative, violating the positivity constraint imposed on the system. To this end, a saturation is imposed on the input signal, such that

$$u = \begin{cases} \bar{u}, \text{if } \bar{u} \leq u \\ u, \text{if } 0 < u < \bar{u} \\ 0, \text{otherwise,} \end{cases} \tag{34}$$

where $\bar{u}$ is the maximum input rate that the insulin pump can handle. The value of $\bar{u}$ was chosen to be $\bar{u} = 30[U/h]$, which corresponds to the typical maximum basal administration rate in commercial insulin pumps [27].

The corresponding linear system in (33) is described by the matrices

$$\begin{aligned} A &= \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \\ B &= \begin{pmatrix} 0 & 0 & 0 & 1 \end{pmatrix}^{\mathsf{T}} \\ C &= \begin{pmatrix} 1 & 0 & 0 & 0 \end{pmatrix}. \end{aligned} \tag{35}$$

This system is controllable and could be regulated by a linear state feedback controller that tracks a setpoint value of r_c with x_4. The state feedback law is given in the form of

$$\begin{aligned} v &= k_f r_c - K\boldsymbol{\xi} \\ k_f &= -1/(C(A - BK)^{-1}B), \end{aligned} \tag{36}$$

where k_f is a corrector term, and K is the gain, which can either be found by pole placement or the minimization of the LQR cost. The deform function $\mathcal{G}(\mathcal{E}(g_i) - v, g_i, \theta)$ was chosen to be

$$\mathcal{G}(\mathcal{E}(g) - v, g, \theta) = (g + K_c)(1 + B_c \tanh(A_c(\dot{\xi}_4 - v))) - K_c, \tag{37}$$

which was previously applied in a wide variety of systems [28]. Here, $\dot{\xi}_4 = \mathcal{E}(g)$ from (14), and the parameter vector has the elements $\theta = [A_c,\ B_c,\ K_c]^\intercal$.

The controller can be tuned by first choosing the closed-loop poles p_{cl} of the linear controller, using only the IO linearized version, finding the gain matrix K, and then tuning the parameters of the deform function. In the first step, one sets every model parameter instance in the control loop to the average system parameters and then finds a proper choice for p_{cl}, which is $p_{cl} = [-0.04, -0.03, -0.02, -0.01]$ in this case. This choice ensures that no overshoot is present during the setpoint tracking, which is important to avoid negative insulin administration. For this choice, the corresponding gain matrix was $K = [1.57 \cdot 10^{-7}, 3.64 \cdot 10^{-5}, 2.8 \cdot 10^{-3}, 9 \cdot 10^{-1}]$ with the correction term $k_f = 1.57 \cdot 10^{-7}$. The reference to be tracked was $r_c = 120$.

After tuning the linear controller, the parameters of the deform function have to be determined. By keeping the same system parameter settings, i.e., using the approximate parameters in every function, one determines first the initial condition of the iteration, which was $g_0 = k_f r_c - K\mathcal{T}(\boldsymbol{x}_0)$. Then, the elements of θ are determined, for which a rule of thumb is that B_c is -1, while A_c and K_c differ by order of magnitude [28]. As a consequence, the parameters $B_c = -1$, $K_c = 10^{-1}$, and $A_c = 10^{-2}$ lead to $g(t) = v(t)$, which means that the iteration converges to the proper fixed point. Additionally, the convergence tolerance was set to $\delta = 10^{-12}$.

The last step is to find the Lipschitz constant L of the SMD since the rest of the parameters are fixed. In this case, its value was determined through computer simulations, which resulted in the value of $L = 10^{-14}$. Figure 1 shows a schematic diagram of the entire control loop.

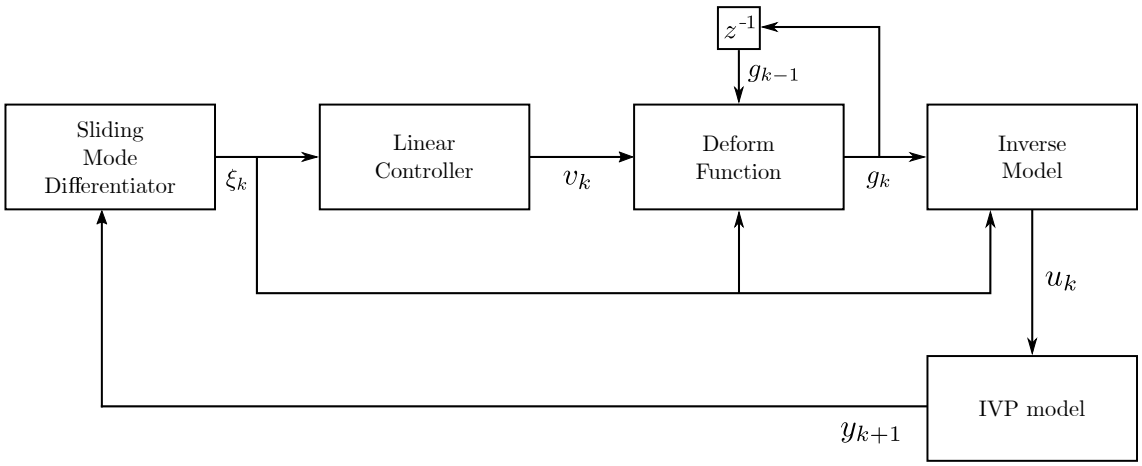

Figure 1. Schematic diagram of the proposed control loop. The SMD first produces the derivatives $\boldsymbol{\xi}$ of $y = x_4$. Then, the linear controller calculates $v = k_f r_c + K\xi$, which is used in the deform function to create the error $\dot{\xi}_4 - v$ and perform one step of the RFPT iteration. The resulting iterate is then used to calculate the input signal for the system via the inverse model, and the cycle repeats at each step.

4. In Silico Results

The approach was validated in silico using ten virtual patients, each representing a unique parameter set for the IVP model. The parameters of the virtual patients can be seen in Table 2. Each simulation ran for six days with a fixed CH intake profile. On each day, three meals were simulated, with varying CH contents, which can be seen in Figure 2. A limit was imposed on the BG level, such that it cannot exit the closed region

$x_4 \in [50; 400]$; otherwise, the subject enters into a deadly hypoglycemic or hypoglycemic state, respectively.

Table 2. Parameters of the virtual patients, associated with the IVP model.

	EGP	$GEZI$	C_I	S_I	τ_1	τ_2	vk_1
p_1	1.33	2.2×10^{-3}	2010	8.1×10^{-4}	49	47	1×10^{-2}
p_2	0.6	4.4×10^{-3}	1281	9.6×10^{-4}	41	10	1.2×10^{-2}
p_3	1.07	3.5×10^{-3}	909	4.6×10^{-4}	71	70	2.3×10^{-2}
p_4	0.98	1.6×10^{-5}	1813	3.8×10^{-4}	91	70	8.1×10^{3}
p_5	0.6	4.3×10^{-3}	1535	2.1×10^{-4}	46	46	9.6×10^{-3}
p_6	0.6	1×10^{-3}	588	4.1×10^{-4}	68	30	9.2×10^{-3}
p_7	1.11	2.3×10^{-3}	1806	8.2×10^{-4}	60	60	1×10^{-2}
p_8	1.3	1×10^{-8}	540	3.7×10^{-4}	95	37	1×10^{-2}
p_9	1.27	6.4×10^{-3}	875	2.6×10^{-4}	131	21	1×10^{-2}
p_{10}	0.61	1×10^{-3}	1309	6×10^{-4}	53	53	1×10^{-2}

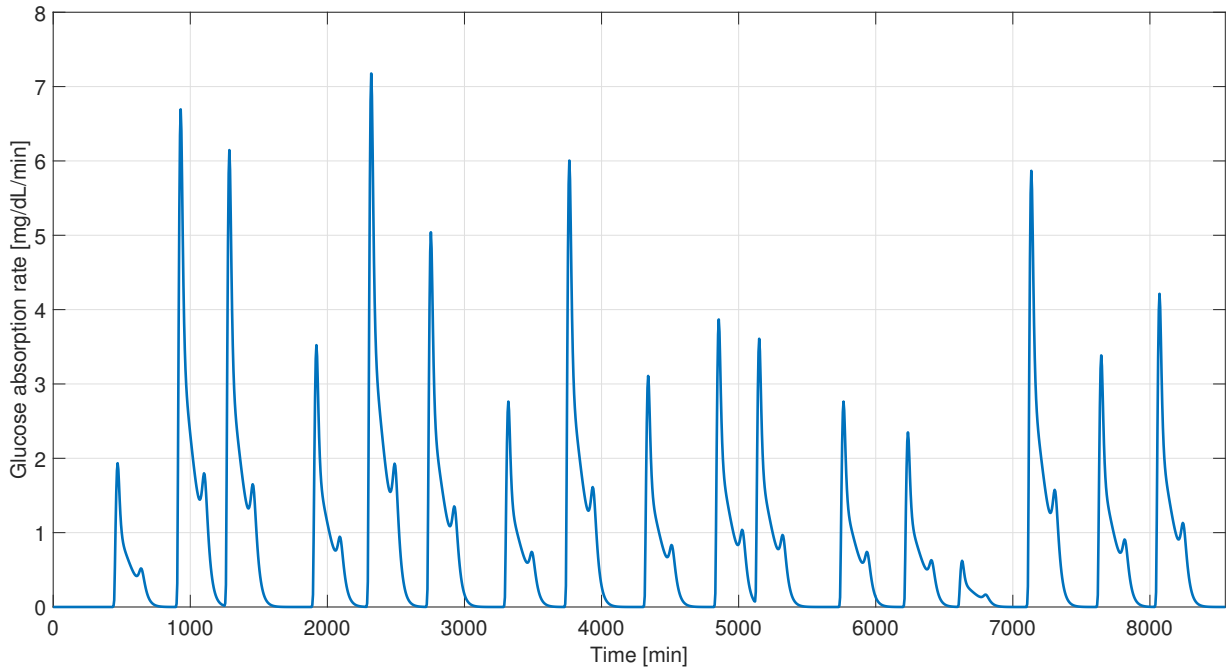

Figure 2. Rate of glucose appearance after CH absorption for the simulation.

Some additional considerations were introduced before running the simulation. Since CGMS sensors have a reasonably large sampling time, they had to be included in the simulation. Currently, commercial CGMS devices permit the sampling of BG levels 5 min apart, and as a consequence, this was chosen to be the dead time between measurements, such that $\Delta t = 5$. Between each measurement sample, the system was simulated with a step size of $\delta t = 0.1$, which was also the refresh rate of the linear controller (36), which produces v. The SMD was integrated once a new measurement was available from the previous measurement, updated with the latest measured value.

It was also assumed that no information is present on the R_a disturbance, hence in the inverse model $R_a = \dot{R}_a = \ddot{R}_a = \dddot{R}_a = 0$. Moreover, only x_4 was used in the inverse model (33), since it could be directly measured, and the remaining terms were set to $x_1 = x_2 = x_3 = 0$ in this equation, which leads to the simplified inverse model

$$u = -(C_I \tau_1 \tau_2 (g + (GEZI(-\tau_2 x_4 GEZI^3 + EGP \tau_2 GEZI^2))/\tau_2))/(S_I p_2 x_4). \quad (38)$$

An unexpected consequence of this choice was that the control performance increased since the inverse transformation (29) turned out to be numerically ill conditioned. In practice, one can only measure x_4, while the other states are hidden. By estimating the derivatives $\boldsymbol{\xi}$ of $y = x_4$ using the SMD, these derivatives can be used in the inverse transformation (29) to obtain estimates on x_1, x_2, and x_3. However, the SMD produces noisy estimates, and the cubic terms further exacerbate the errors in the inverse transformations ($\bar{\xi}_1$ in δ_1, for example), which leads to unreliable state estimates. While omitting these terms leads to an inaccurate inverse, this discrepancy still involves smaller errors than using the ill-conditioned transformations.

The deform function (37) was also only updated when a new measurement was available since the effect of g_k on $\mathcal{E}(g_k)$ could only be determined through direct sampling. Between each measurement, with the previously defined sampling $\Delta t = 0.1$, the previous iteration g_{k-1} was used in the inverse model.

The results were assessed using the time-in-range (TIR) metric [29] and the Control Variability Grid Analysis (CVGA) [30]. The TIR metric defines different regions and determines what percentage of the BG level stays in the given areas during the simulation, whereas the CVGA plot shows the maximal and minimal BG levels attained during closed-loop therapy.

The first simulation with the initially tuned parameters led to unsatisfactory results in general. Of the ten virtual patients, only eight could stay within limits on the BG level. Additionally, three of the remaining cases were grouped in the E region of the CVGA plot, indicating life-threatening BG levels. Consequently, the control parameters were fine-tuned, so better performance was achieved. Through simulations the control parameters were empirically modified to $p_{cl} = p_{cl} \times 0.8$, $K_c = 10$, $A_c = 10^{-4}$, and $L = 10^{-16}$. By decreasing the eigenvalues of p_{cl}, the response time of the linear controller can be decreased, thus lowering the overshoot in the BG level due to the disturbances. This modification improved general performance; however, two virtual patients were still out of range. The simulation results can be seen in Figures 3–6, and the corresponding CVGA plot in Figure 7. Table 3 contains each patient's CVGA and TIR metrics, where an additional column shows the extreme cases where hypoglycemia or hyperglycemia occurred. The best results were achieved in the case of p_1, p_7, p_8, and p_9, where the TIR was in the safe range (70–180 (mg/dL)) for more than 80% of the time. However, as one can see, only two sets of parameters entered the safe region B on the CVGA plot, while the others remained in the D and E zones. While this seems to be a poor result at first glance, another perspective can be gained by inspecting Figure 3.

As one can see, the out-of-range BG levels occurred in the region $t \in [0; 2500]$, where there was a significant deviation from the tracked setpoint. However, after 2500 min, the controller could provide a tight tracking of the BG level, irrespective of the R_a magnitude. Each BG trajectory is contracted in the optimal region, indicated by the green dash-dotted lines. This initial deviation is due to the iterative nature of the controller, where if the initial condition g_0 is poorly estimated, the convergence is slower to the desired fixed point, as Figure 5 shows. In Figure 3, this slow convergence is what one can see, where at the beginning of the simulation, the trajectories corresponding to each virtual patient have a significant deviation in between, but as time progresses, this deviation vanishes and each trajectory evolves close to each other.

The adaptation period can be significantly shortened by supplying better initial conditions at the beginning of the simulation. For example, taking the last value of g in the simulation and using it as an initial condition improves the results drastically, which can be seen in Figures 3, 4 and 6. The beginning of the simulations is significantly better, and the trajectories are bounded by the red dashed lines, except for p_7, which barely enters into hypoglycemia. This observation is further confirmed by the bottom CVGA plot in Figure 7, and the TIR values in Table 4.

Nevertheless, finding better initial estimates for g is nontrivial in practice since it implicitly requires knowledge of the uncertainty of the model parameters. One possible

way to reduce the variation during adaptation is to restrict the CH intake at the beginning of the operation of the controller. Another solution could be to redefine A_c to be time-varying. Values closer to K_c lend to acceleration in convergence, but poor adaptation in the long run, while using smaller values has the opposite effect. A better result could be achieved by using larger values at the beginning of the control cycle and then smoothly varying them to smaller values, which will be a focus of future research.

Discussion

The proposed method is an adaptive control method, similar to previous designs, shown in [31,32]. While the previous demonstration showed that the choice of initial condition for the deform function significantly determines its performance, the controller could handle inter-patient variability by solely using the mean of the parameters. This contrasts with PID and Model Predictive Control (MPC), the two most popular approaches in T1DM control.

The RFPT controller has similar computational characteristics as PID control in the sense that it does not require extensive computational hardware. The most important benefit of the method lies in its adaptability. In general, PID control cannot consider the individual characteristics of each patient, and hence can only give a suboptimal performance in many cases. However, in this case, the PID should outperform the RFPT, since the initial condition of the iteration cannot be given precisely in practice. Additionally, RFPT is more cumbersome in its theoretical exposition, which could make it less preferable in practice.

One can also compare the RFPT with MPC implementations. MPC is an optimal control method that formulates the control problem as an optimization task. As a direct consequence, the obtained control signal is optimal with respect to a cost function, which often incorporates the minimization of the exogenous insulin applied to the patient. By contrast, RFPT does not possess any optimal traits due to the simulation-based tuning of the control parameters. However, the optimization procedure often requires complex algorithms, which makes it computationally less attractive than the RFPT. Moreover, the performance of the MPC significantly relies on the internal model of the process, which, in the presence of parameter uncertainties, can lead to instability. In practice, this problem is challenging to solve and requires advanced mathematical apparatus, thus the RFPT might be a viable alternative.

One important remark is that, as shown in the theoretical introduction, the RFPT contains a linear controller due to the IO linearization framework, which can be of any type. Consequently, the state feedback controller used in this study can be replaced by a PID controller or a linear MPC, combining the benefits of both approaches.

Table 3. CVGA and TIR metrics, corresponding to the slow adaptation.

			TIR (%)				
	Extermity	**CVGA**	**< 50**	**50 − 70**	**70 − 180**	**180 − 250**	**250 <**
p_1		B	0	0	0.84	0.13	0.02
p_2		E	0	0	0.67	0.18	0.15
p_3		Lower D	0	0.14	0.74	0.11	0.01
p_4		Upper D	0	0	0.71	0.19	0.1
p_5		E	0	0.09	0.67	0.14	0.09
p_6	hypo		0.14	0.16	0.58	0.1	0.02
p_7		Lower D	0	0.06	0.8	0.13	0.01
p_8		Lower D	0	0.08	0.81	0.1	0.01
p_9		B	0	0	0.85	0.14	0.02
p_{10}	hypo		0.07	0.18	0.59	0.13	0.03

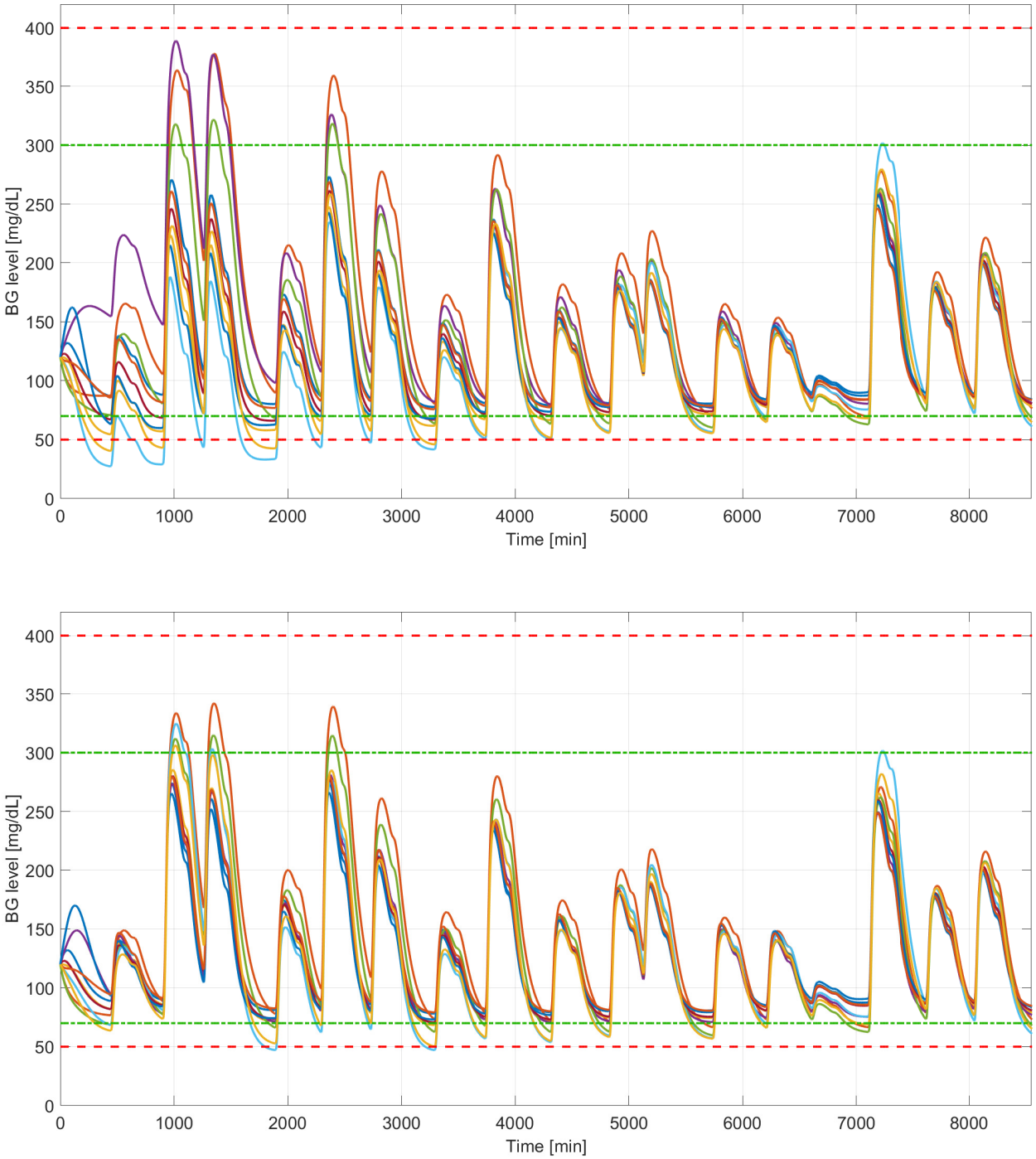

Figure 3. BG levels of the two simulations. The top figure shows the slow adaptation case, while the bottom corresponds to the fast. The red dashed line indicates a tolerable region of BG level, while the green dash-dotted line shows the optimal region.

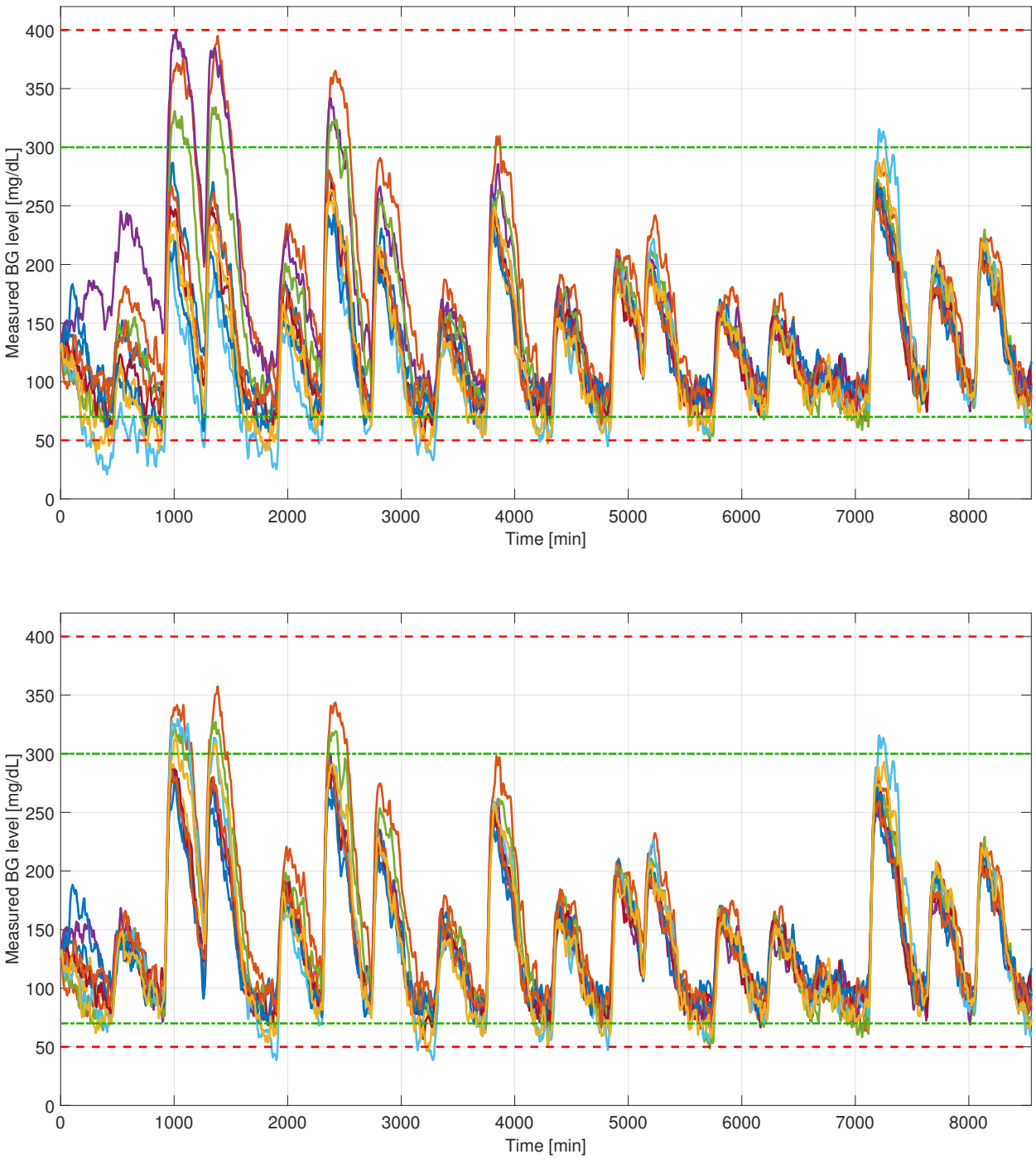

Figure 4. Measured and noise-corrupted BG levels. The top figure shows the slow adaptation case, while the bottom corresponds to the fast. The red dashed line indicates a tolerable region of BG level, while the green dash-dotted line shows the optimal region.

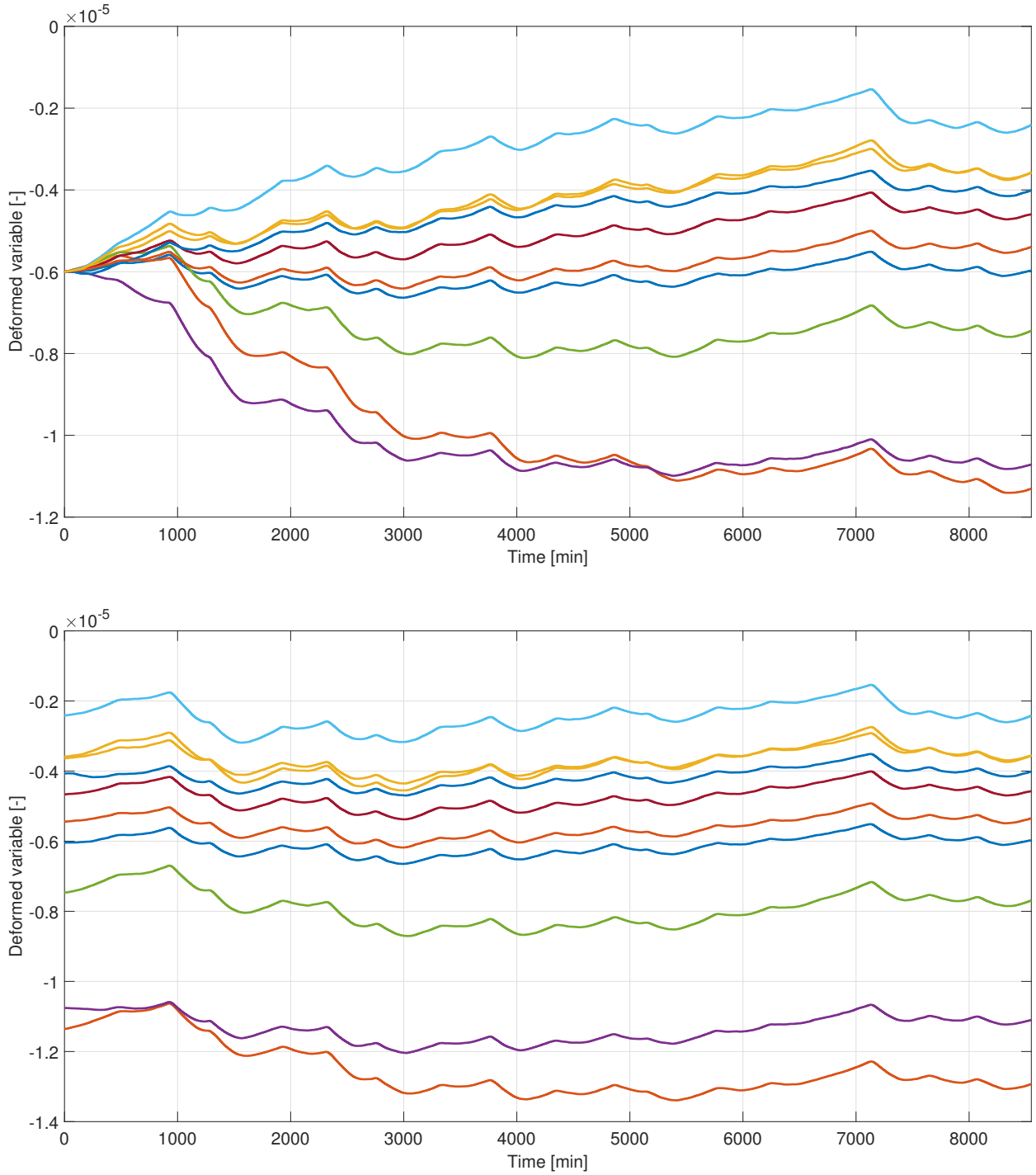

Figure 5. Evolution of the deform variable g associated with the closed-loop control. The top figure shows the slow adaptation case, while the bottom corresponds to the fast.

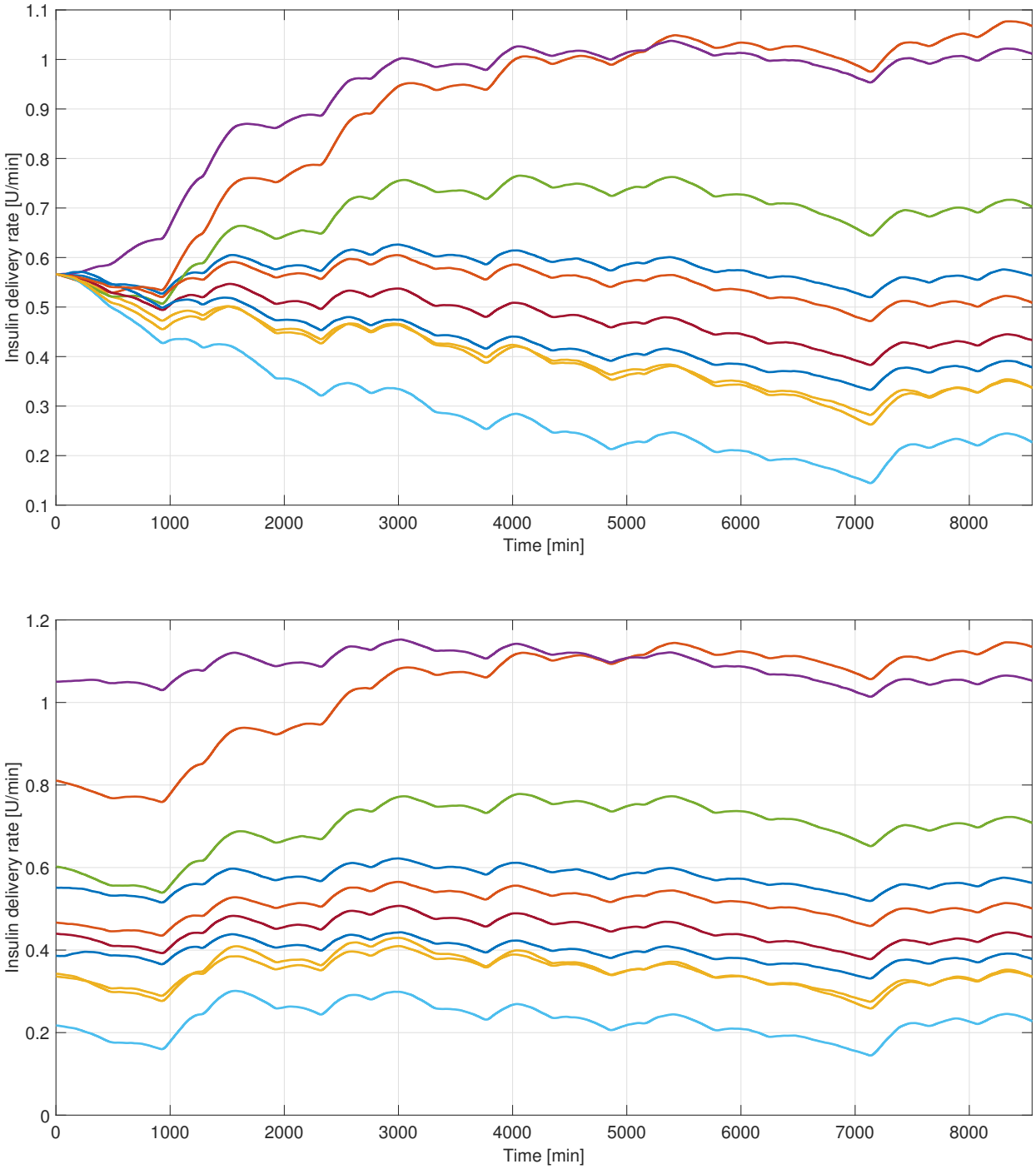

Figure 6. The computed closed-loop insulin delivery rates. The top figure shows the slow adaptation case, while the bottom corresponds to the fast. The signals closely resemble the deformed signals. One can see that the input signals in both cases respected the actuator limitations, hence the control strategies were feasible in practice.

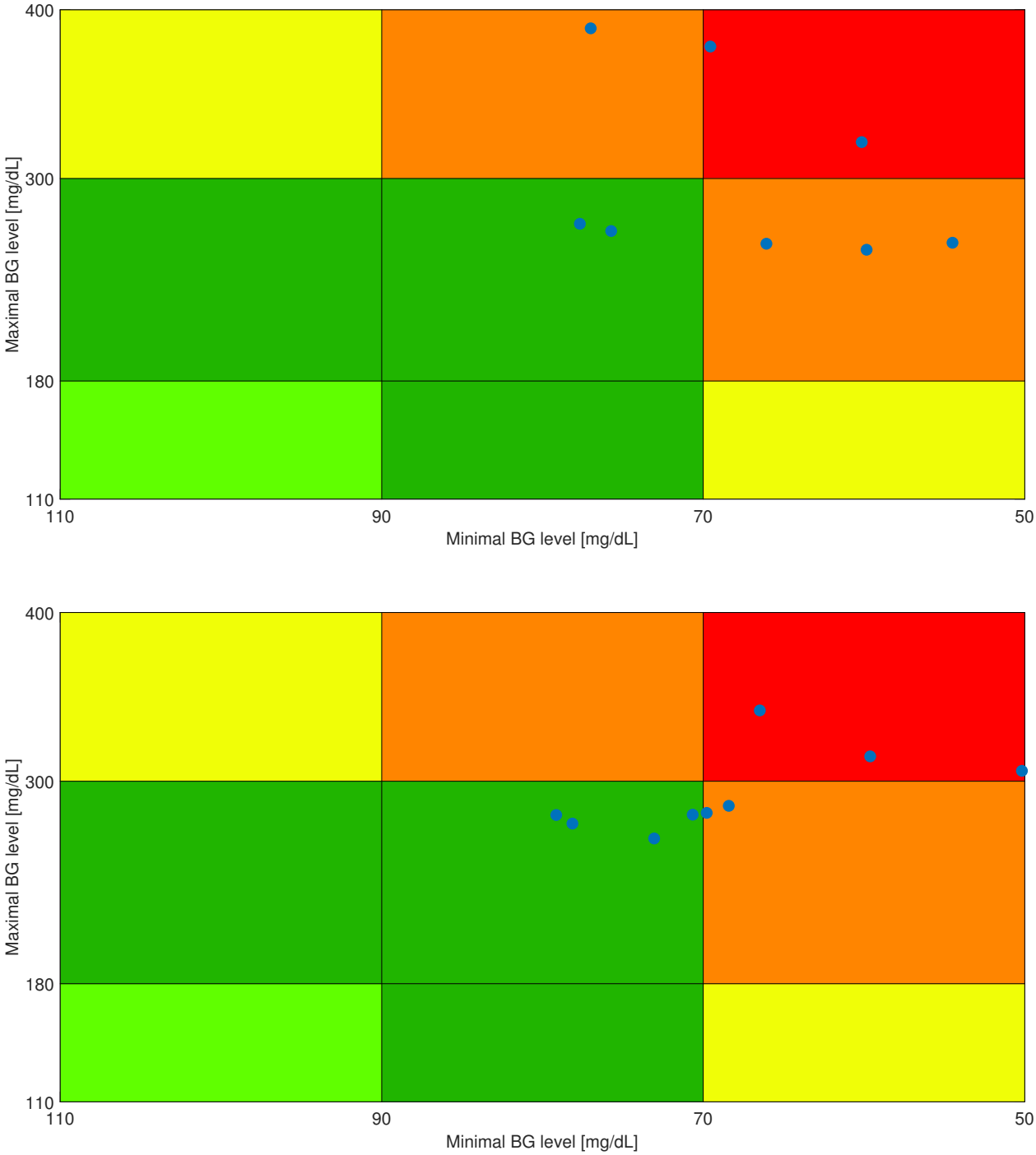

Figure 7. CVGA plot of the BG levels attained during the simulation. The top figure shows the slow adaptation case, while the bottom corresponds to the fast. The green zones (A,B) indicate safe levels, yellow zones (C) are the acceptable levels, orange zones (D) are the on the margin of safety, and the red zone (E) indicates life-threatening scenarios.

Table 4. CVGA and TIR metrics, corresponding to the fast adaptation.

			TIR (%)				
	Extermity	**CVGA**	< 50	$50 - 70$	$70 - 180$	$180 - 250$	$250 <$
p_1		B	0	0	0.84	0.13	0.03
p_2		E	0	0.03	0.69	0.17	0.12
p_3		Lower D	0	0.01	0.81	0.13	0.04
p_4		Lower D	0	0	0.83	0.14	0.03
p_5		E	0	0.11	0.66	0.14	0.09
p_6	hypo		0.02	0.13	0.64	0.13	0.08
p_7		B	0	0	0.83	0.13	0.04
p_8		B	0	0	0.85	0.13	0.02
p_9		B	0	0	0.83	0.14	0.03
p_{10}		E	0	0.17	0.63	0.13	0.07

5. Conclusions

In the present work, an RFPT-based proof-of-concept controller was introduced. The theory of the method was presented in conjunction with the design of the controller. The scheme was augmented with an SMD, providing higher-order derivatives essential for the controller's operation outside the in silico domain. The controller was simulated with a fixed CH intake protocol, and the results were analyzed using both CVGA and TIR metrics. Results showed that the controller could not provide satisfactory tracking at the beginning of its operation, but the effect of disturbances was significantly lowered after an adaptation period.

In order to create a reliable controller, several modifications must be implemented in future research. First, the convergence speed must be accelerated by either finding a better estimation for the initial condition of the iteration or using time-varying control parameters. Another possibility is to use alternative deform functions, which is also the scope of future design. In terms of validation, the number of virtual patients should be increased, and a number of different, realistic CH profiles should be used during simulation studies. Moreover, the model on which the controller is validated should be changed to a different, more elaborate one.

Author Contributions: Conceptualization, B.C.; methodology, B.C.; software, B.C.; validation, B.C.; formal analysis, B.C.; investigation, B.C.; resources, B.C.; data curation, B.C.; writing—original draft preparation, B.C.; writing—review and editing, B.C. and D.A.D.; visualization, B.C.; supervision, D.A.D. and L.K.; project administration, D.A.D. and L.K.; funding acquisition, D.A.D. and L.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Research, Development and Innovation Fund of Hungary grant number 2019-1.3.1-KK-2019-00007 under the 2019-1.3.1-KK funding scheme. Bence G. Czakó was supported by the Eötvös Loránd Research Network Secretariat under grant agreement no. ELKH KÖ-40/2020 ('Development of cyber-medical systems based on AI and hybrid cloud methods').

Data Availability Statement: No new data were created or analyzed in this study. Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DM	Diabetes Mellitus
MPC	Model Predictive Control
CH	Carbohydrate
T1DM	Type 1 Diabetes Mellitus
T2DM	Type 2 Diabetes Mellitus
AP	Artificial Pancreas
BG	Blood Glucose
CGMS	Continuous Glucose Monitoring Sensor
RFPT	Robust Fixed-Point Transformations
IO	Input–Output
Sliding Mode Differentiator	SMD
IVP	Identifiable Virtual Patient
TIR	Time-in-Range
CVGA	Control Variability Grid Analysis

References

1. Sun, H.; Saeedi, P.; Karuranga, S.; Pinkepank, M.; Ogurtsova, K.; Duncan, B.B.; Stein, C.; Basit, A.; Chan, J.C.; Mbanya, J.C.; et al. IDF Diabetes Atlas: Global, regional and country-level diabetes prevalence estimates for 2021 and projections for 2045. *Diabetes Res. Clin. Pract.* **2022**, *183*, 109119. [CrossRef] [PubMed]
2. Bekiari, E.; Kitsios, K.; Thabit, H.; Tauschmann, M.; Athanasiadou, E.; Karagiannis, T.; Haidich, A.B.; Hovorka, R.; Tsapas, A. Artificial pancreas treatment for outpatients with type 1 diabetes: systematic review and meta-analysis. *BMJ* **2018**, *361*, k1310. [CrossRef] [PubMed]
3. Thabit, H.; Hovorka, R. Coming of age: the artificial pancreas for type 1 diabetes. *Diabetologia* **2016**, *59*, 1795–1805. [CrossRef]
4. Chakrabarty, A.; Healey, E.; Shi, D.; Zavitsanou, S.; Doyle, F.J.; Dassau, E. Embedded Model Predictive Control for a Wearable Artificial Pancreas. *IEEE Trans. Control. Syst. Technol.* **2020**, *28*, 2600–2607. [CrossRef] [PubMed]
5. Ekhlaspour, L.; Nally, L.M.; El-Khatib, F.H.; Ly, T.T.; Clinton, P.; Frank, E.; Tanenbaum, M.L.; Hanes, S.J.; Selagamsetty, R.R.; Hood, K.; et al. Feasibility Studies of an Insulin-Only Bionic Pancreas in a Home-Use Setting. *J. Diabetes Sci. Technol.* **2019**, *13*, 1001–1007. [CrossRef] [PubMed]
6. Weinzimer, S.A.; Steil, G.M.; Swan, K.L.; Dziura, J.; Kurtz, N.; Tamborlane, W.V. Fully Automated Closed-Loop Insulin Delivery Versus Semiautomated Hybrid Control in Pediatric Patients With Type 1 Diabetes Using an Artificial Pancreas. *Diabetes Care* **2008**, *31*, 934–939. [CrossRef]
7. Buckingham, B.A.; Forlenza, G.P.; Pinsker, J.E.; Christiansen, M.P.; Wadwa, R.P.; Schneider, J.; Peyser, T.A.; Dassau, E.; Lee, J.B.; O'Connor, J.; et al. Safety and Feasibility of the OmniPod Hybrid Closed-Loop System in Adult, Adolescent, and Pediatric Patients with Type 1 Diabetes Using a Personalized Model Predictive Control Algorithm. *Diabetes Technol. Ther.* **2018**, *20*, 257–262. [CrossRef] [PubMed]
8. Kaveh, P.; Shtessel, Y.B. Blood glucose regulation using higher-order sliding mode control. *Int. J. Robust Nonlinear Control.* **2008**, *18*, 557–569. [CrossRef]
9. Eigner, G.; Bojthe, I.; Pausits, P.; Kovacs, L. Investigation of the TP modeling possibilities of the Hovorka T1DM model. In Proceedings of the 2017 IEEE 15th International Symposium on Applied Machine Intelligence and Informatics (SAMI), Herl'any, Slovakia, 26–28 January 2017. [CrossRef]
10. Fereydouneyan, F.; Zare, A.; Mehrshad, N. Using a fuzzy controller optimized by a genetic algorithm to regulate blood glucose level in type 1 diabetes. *J. Med. Eng. Technol.* **2011**, *35*, 224–230. [CrossRef]
11. Mehmood, S.; Ahmad, I.; Arif, H.; Ammara, U.; Majeed, A. Artificial Pancreas Control Strategies Used for Type 1 Diabetes Control and Treatment: A Comprehensive Analysis. *Appl. Syst. Innov.* **2020**, *3*, 31. [CrossRef]
12. Trevitt, S.; Simpson, S.; Wood, A. Artificial Pancreas Device Systems for the Closed-Loop Control of Type 1 Diabetes. *J. Diabetes Sci. Technol.* **2015**, *10*, 714–723. [CrossRef] [PubMed]
13. Tar, J.K.; Rudas, I.J. Geometric Approach to Nonlinear Adaptive Control. In Proceedings of the 2007 4th International Symposium on Applied Computational Intelligence and Informatics, Timisoara, Romania, 17–18 May 2007. CrossRef]
14. Czakó, B.G.; Drexler, D.A.; Kovács, L. Discrete time derivation of the Robust Fixed-Point Transformation method. *IFAC-PapersOnLine* **2022**, *55*, 535–540. [CrossRef]
15. Czako, B.; Drexler, D.A.; Kovacs, L. Control of a T1DM Model Using Robust Fixed-Point Transformations Based Control with Disturbance Rejection. In Proceedings of the 2022 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR), Cluj-Napoca, Romania, 19–21 May 2022. [CrossRef]
16. Kovács, L. A robust fixed point transformation-based approach for type 1 diabetes control. *Nonlinear Dyn.* **2017**, *89*, 2481–2493. [CrossRef] [PubMed]

17. Eigner, G.; Horvath, P.; Tar, J.K.; Rudas, I.J.; Kovacs, L. Application of Robust Fixed Point Control in Case of T1DM. In Proceedings of the 2015 IEEE International Conference on Systems, Man, and Cybernetics, Hong Kong, China, 9–12 October 2015. [CrossRef]
18. Levant, A. Higher-order sliding modes, differentiation and output-feedback control. *Int. J. Control* **2003**, *76*, 924–941. [CrossRef]
19. Levant, A.; Livne, M.; Yu, X. Sliding-Mode-Based Differentiation and Its Application. *IFAC-PapersOnLine* **2017**, *50*, 1699–1704. [CrossRef]
20. Barbot, J.P.; Levant, A.; Livne, M.; Lunz, D. Discrete sliding-mode-based differentiators. In Proceedings of the 2016 14th International Workshop on Variable Structure Systems (VSS), Nanjing, China, 1–4 June 2016. [CrossRef]
21. Czako, B.G.; Drexler, D.A.; Kovacs, L. Continuous time Robust Fixed Point Transformations based control. In Proceedings of the 2019 IEEE AFRICON, Accra, Ghana, 25–27 September 2019. [CrossRef]
22. Kanderian, S.S.; Weinzimer, S.; Voskanyan, G.; Steil, G.M. Identification of Intraday Metabolic Profiles during Closed-Loop Glucose Control in Individuals with Type 1 Diabetes. *J. Diabetes Sci. Technol.* **2009**, *3*, 1047–1057. [CrossRef] [PubMed]
23. Khalil, H.K. *Nonlinear Systems*; Prentice Hall: Upper Saddle River, NJ, USA, 2002.
24. Barbot, J.P.; Levant, A.; Livne, M.; Lunz, D. Discrete differentiators based on sliding modes. *Automatica* **2020**, *112*, 108633. [CrossRef]
25. Kanderian, S.S.; Weinzimer, S.A.; Steil, G.M. The Identifiable Virtual Patient Model: Comparison of Simulation and Clinical Closed-Loop Study Results. *J. Diabetes Sci. Technol.* **2012**, *6*, 371–379. [CrossRef]
26. Vettoretti, M.; Battocchio, C.; Sparacino, G.; Facchinetti, A. Development of an Error Model for a Factory-Calibrated Continuous Glucose Monitoring Sensor with 10-Day Lifetime. *Sensors* **2019**, *19*, 5320. [CrossRef]
27. 2020 Consumer Guide on Insulin Pumps. Avaliable online: http://main.diabetes.org/dforg/pdfs/2020/2020-cg-insulin-pumps.pdf (accessed on 19 January 2023).
28. Tar, J.; Bitó, J.; Nádai, L.; Tenreiro Machado, J. Robust fixed point transformations in adaptive control using local basin of attraction. *Acta Polytech. Hung.* **2009**, *6*, 21–38.
29. Advani, A. Positioning time in range in diabetes management. *Diabetologia* **2019**, *63*, 242–252. [CrossRef] [PubMed]
30. Magni, L.; Raimondo, D.M.; Man, C.D.; Breton, M.; Patek, S.; Nicolao, G.D.; Cobelli, C.; Kovatchev, B.P. Evaluating the Efficacy of Closed-Loop Glucose Regulation via Control-Variability Grid Analysis. *J. Diabetes Sci. Technol.* **2008**, *2*, 630–635. [CrossRef]
31. Turksoy, K.; Cinar, A. Adaptive Control of Artificial Pancreas Systems—A Review. *J. Healthc. Eng.* **2014**, *5*, 1–22. [CrossRef] [PubMed]
32. Nath, A.; Deb, D.; Dey, R. Robust observer-based adaptive control of blood glucose in diabetic patients. *Int. J. Control* **2020**, *94*, 3054–3067. [CrossRef]