Long-Term Prediction for T1DM Model During State-Feedback Control

Péter Szalay[1], Zoltán Benyó[1] and Levente Kovács[2]

Abstract— **Avoiding low glucose concentration is critically important in type-1 diabetes treatment. Predicting the future plasma glucose levels could ensure the safety of the patient. However, such estimation is no trivial task. The current paper proposes a predictor framework which stems from Unscented Kalman filter and works during closed-loop control, that can predict hazardous glucose levels in advance. Once the blood glucose concentration starts to rise, the predictor activates and estimates future glucose levels up to 3 hours, confirming whether the controller can endanger the patient. The capabilities of the framework is presented through simulations based on the SimEdu validated in-silico simulator.**

I. INTRODUCTION

Diabetes Mellitus (DM) is a collective term referring to several chronic metabolic diseases typically characterized by elevated blood glucose levels, glucose excretion in the urine, and consequentially with an increase in urine volume. Type I. DM (T1DM) is driven by autoimmune mediated destruction of pancreatic β-islet cells, which leads to insufficient insulin production and elevated blood glucose levels [1]. As a result immediate insulin treatment is needed to maintain normal glucose concentration (normoglycemia). In the past few decades the automation of insulin treatment has been extensively researched [2], [3], [4], and it is usually referred to as Artificial Pancreas (AP). Some implementations of AP has matured to clinical testing [5], [6] with promising results.

One of the biggest dangers with any form of diabetes treatment is hypoglycemia (a condition where the blood glucose concentration is going below 3.9 mmol/L), which can result in a variety of symptoms and can lead to loss of consciousness, seizures and even death [1]. Thus, for any (semi-) automated treatment, it is crucial to avoid hypoglycemia.

This is however, no easy task for a variety of reasons including:

1) If the controller can only administer insulin, it has no means to elevate glucose levels.
2) Even rapid-acting insulin - if injected subcutaneously - has significantly slower effect on the plasma glucose concentration than meal intake or physical activity.
3) The dynamics of the human metabolism concerning glucose is slower when the glucose levels are lower compared to when they are high [7].

[1]Péter Szalay and Zoltán Benyó are with the Department of Control Engineering and Information Technology, Budapest University of Technology and Economics, Budapest, Hungary szalaip@iit.bme.hu, benyo@iit.bme.hu

[2]Levente Kovács is with the University Research and Innovation Center, and Physiological Controls Group, Biomatics Institute, John von Neumann Faculty of Informatics, Óbuda University, Budapest, Hungary kovacs.levente@nik.uni-obuda.hu

Thus, to reduce the duration and severity of hyperglycemic episodes (glucose concentration exceeding 7.8 mmol/L), the controller must inject insulin shortly after a meal intake. While in the attempt to avoid high glucose levels, hypoglycemia can occur if too much insulin is administered. It is suspected, that relying solely on feedback, where the timing and amount of meal intake of the patient is unknown, cannot achieve clinically acceptable quality tight glycemic control [8], [9].

Predicting future glucose concentrations could help to avoid such severe outcomes. Various methods have been introduced to achieve reliable estimations [10], [11].

The goal of the work presented here was to develop a predictor algorithm capable of detecting potential hypoglycemic episodes during tight glycemic control by calculating a confidence interval for future glucose concentration values. Given a controller tuned to suppress high glucose levels, if the proposed algorithm can detect potential hypoglycemia early enough to intervene, the safety of pure feedback controllers can be improved significantly.

Results have been demonstrated on in-silico simulations of the SimEdu simulator [13]. The model used during these simulation is presented in Section II. Section III is about the Linear Parameter-Varying (LPV) State-Feedback controller. This controller requires an observer providing estimation of state variables. Our choice was a modified version of the Unscented Kalman Filter (UKF), which also serves as the core of the predictor algorithm presented in Section IV. The controller was purposefully modified to allow low glucose levels. The results of the Simulation are detailed in Section V.

II. MODELING

A 11th order model (1) has been used for the simulations, which was introduced by [12] and later updated in [13]. It will be referred to as the Cambridge-model. The state variables of the model are: $C(t)$ glucose concentration in the subcutaneous tissue [mmol/L], $Q_1(t)$ and $Q_2(t)$ the masses of glucose in accessible and non-accessible compartments [mmol], $x_1(t)$, $x_2(t)$ and $x_3(t)$ remote effect of insulin on glucose distribution, disposal and endogenous glucose production respectively [1/min], $I(t)$ insulin concentration in plasma [mU/L], $S_1(t)$ and $S_2(t)$ insulin masses in the accessible and non-accessible compartments [mU], $G_1(t)$ and $G_2(t)$ masses of ingested glucose in the stomach and gut [mmol/kg]. The input of the system is the $u(t)$ injected insulin flow of rapid-acting insulin [mU/min] is the input of the system, while the $D(t)$ amount of ingested carbohydrates [mmol/min] and the $Phy(t)$ effect of physical activity

[mmol/min] are considered as disturbances. The output is the $C(t)$ glucose concentration in the subcutaneous tissue. The measurement noise is modeled by additive white noise. Further details about the model can be found in [13] and [14].

$$
\begin{aligned}
\dot{C}(t) &= -k_{a,int}C(t) + \frac{k_{a,int}}{V_G}Q_1(t) \\
\dot{Q}_1(t) &= -\left(\frac{F_{01}^s}{Q_1(t)+V_G} + x_1(t)\right)Q_1(t) + k_{12}Q_2(t) - \\
&\quad -R_{cl}max\{0, Q_1(t) - R_{thr}V_G\} - Phy(t) + \\
&\quad +EGP_0 max\{0, 1 - x_3(t)\} + min\left\{U_{G,ceil}, \frac{G_2(t)}{t_{max}}\right\} \\
\dot{Q}_2(t) &= x_1(t)Q_1(t) - \left(k_{12} + x_2(t)\right)Q_2(t) \\
\dot{x}_1(t) &= -k_{b1}x_1(t) + S_{IT}k_{b1}I(t) \\
\dot{x}_2(t) &= -k_{b2}x_2(t) + S_{ID}k_{b2}I(t) \\
\dot{x}_3(t) &= -k_{b3}x_3(t) + S_{IE}k_{b3}I(t) \\
\dot{I}(t) &= \frac{k_a}{V_I}S_2(t) - k_eI(t) \\
\dot{S}_2(t) &= -k_aS_2(t) + k_aS_1(t) \\
\dot{S}_1(t) &= -k_aS_1(t) + u(t) \\
\dot{G}_2(t) &= (G_1(t) - G_2(t))min\left\{\frac{1}{t_{max}}, \frac{U_{G,ceil}}{G_2(t)}\right\} \\
\dot{G}_1(t) &= D(t) - G_1(t)min\left\{\frac{1}{t_{max}}, \frac{U_{G,ceil}}{G_2(t)}\right\}
\end{aligned}
\tag{1}
$$

A high-order model is rather difficult to work with, therefore (1) was only used in simulations and the following reduced model is the basis of observer and controller design:

$$
\begin{aligned}
\dot{Q}_1(t) &= -(F_aQ_1(t) + F_b + x_1(t))Q_1(t) + k_{12}Q_2(t) - \\
&\quad -R_{cl}max\{0, Q_1(t) - R_{thr}V_G\} - Phy(t) + \\
&\quad +EGP_0 max\left\{0, 1 - \frac{k_aS_{IE}}{V_Ik_e}S_2(t)\right\} + G_2(t) \\
\dot{Q}_2(t) &= x_1(t)Q_1(t) - \left(k_{12} + \frac{k_aS_{ID}}{V_Ik_e}S_2(t)\right)Q_2(t) \\
\dot{x}_1(t) &= k_{b1}\left(\frac{k_aS_{IT}}{V_Ik_e}S_2(t) - x_1(t)\right) \\
\dot{S}_2(t) &= -k_aS_2(t) + k_aS_1(t) \\
\dot{S}_1(t) &= -k_aS_1(t) + u(t) \\
\dot{G}_2(t) &= \frac{G_1(t) - G_2(t)}{t_{max}} \\
\dot{G}_1(t) &= \frac{U_{G,ceil}D(t) - G_1(t)}{t_{max}}
\end{aligned}
\tag{2}
$$

where $F_{01}^s(Q_1(t) + V_G)^{-1} \approx F_aQ_1(t) + F_b$. The output of this reduced model is $C(t) \approx Q_1(t)/V_G$.

III. CONTROLLER DESIGN

Without closed-loop control and with meal intake, unknown and various unmeasured disturbances affecting the human metabolism the blood glucose levels can vary in a wide range for a diabetes patient [1]. However, with a successful closed-loop control the output can be kept in a narrow range, making prediction easier even if normoglycemia is not achieved. The controller presented here achieves just that: makes the behavior of the patient easier to predict. Our choice was linear parameter-varying (LPV) control, tuned for disturbance rejection, and damping oscillatory transients. Then, to show the capabilities of the predictor, the latter property has been removed. This modified controller will bring the simulated patient to hypoglycemia after every meal intake, which the predictor can detect.
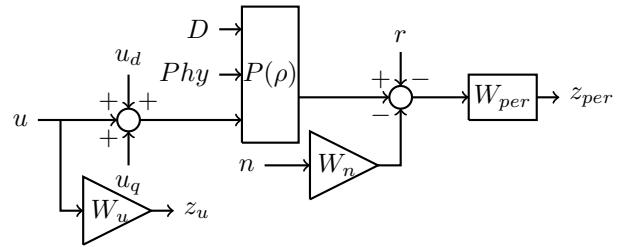

Fig. 1. Nominal reduced LPV model with weighting functions.

The model (2) can be transformed into an LPV system [15]. For the LPV form, let us introduce the following notation for the scheduling variables:

$$
\begin{aligned}
\rho(t) = \left(\begin{array}{c} \rho_1(t) \\ \rho_2(t) \end{array}\right) = \left(\begin{array}{c} Q_1(t) \\ Q_2(t) \end{array}\right) \\
Q_{1,min} \le Q_1(t) \le Q_{1,max}. \\
Q_{2,min} \le Q_2(t) \le Q_{2,max}.
\end{aligned}
\tag{3}
$$

Further components - called weighting functions - can represent control signal constraints, disturbance and measurement noise properties, uncertainties or tracking performance requirements [16]. Based on [20] such weighting functions extended the model as shown on Fig. 1. The inputs are: the u control signal, the u_q quantization noise, the u_d control signal disturbance, the D meal intake, the Phy physical activity, the n measurement noise, and the r reference signal. The outputs are the z_{per} tracking error and the z_u control signal constraint performance signal. $P(\rho)$ represents the nominal T1DM model (2) in LPV form, W_u is a scalar weight that equals to the inverse of the maximal allowed insulin signal, while W_n indicate measurement noise level. Finally, W_{per} is the transfer function of a first order system that defines tracking performance (4): the low frequency components of the tracking error is requested to be 10 times smaller than for higher frequencies.

$$
W_{per} = \frac{0.6s + 0.1}{60s + 1}
\tag{4}
$$

It is possible to add weighting functions representing model uncertainties, making the controller robust [16].

For this extended model we can define a vector for state feedback:

$$
K(\rho) = K_0 + \prod_{i=1}^{2}\rho_i(t)K_i
\tag{5}
$$

which can be calculated by solving a convex optimization problem [18], [19]. The resulting state-feedback controller will have the following properties:

1) The $\mathscr{H}_\infty$ norm of the controlled system from the disturbances input $d(t)$ to constrained control signal $z_u(t)$ is less than 1, and similarly from $d(t)$ to $z_{per}(t)$ tracking performance output is less than $\sqrt{\gamma(\rho)}$.
2) The 'frozen' poles of the controlled system will be withing the area shown on Fig. 2. This will ensure that the controller can be realized in discrete time and the transients will be less oscillatory.
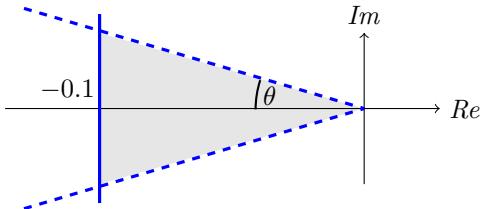
Fig. 2.   Constrains on the closed-loop poles

However, the second property has been modified so that the imaginary part of the closed-loop poles are no longer constrained (dashed line in Fig.2). Therefore, hypoglycemia can occur.

Finally, because of the $\max\{\}$ functions present in (2) one can define not one but four separate LPV models depending on the value of $Q_1(t)$ and $x_3(t)$ (or $k_a S_{IE} S_2(t)/(V_I k_e)$ in the reduced model) as presented in [15].

1) Endogenous glucose production (EGP) is active and there is no renal extraction of glucose: ($x_3(t) \leq 1$) and $Q_1(t) < R_{thr} V_G$;
2) EGP is active ($x_3(t) \leq 1$), and renal extraction is active $Q_1(t) > R_{thr} V_G$;
3) No EGP ($x_3(t) > 1$), and there is no renal extraction $Q_1(t) \leq R_{thr} V_G$;
4) No EGP ($x_3(t) > 1$), and renal extraction is active $Q_1(t) > R_{thr} V_G$;

Therefore, four different controllers can be defined for the four different LPV models, and the final control rule will switch between them. Let us introduce switching functions $sw_i(Q_1(t), S_2(t)) \in \mathbb{R}$ (or $sw_i(t)$ for short), $i = 1, \ldots, 4$ so that $\sum_{i=1}^{4} sw_i(Q_1(t), S_2(t)) = 1$. The control rule will be as follows:

$$u(t) = \sum_i^4 sw_i(t) \mathrm{K}_i(\rho(t))\Big(\mathrm{x}(t) - \mathrm{x}_0\Big) + u_0 \qquad (6)$$

where x_0 is the steady state values of the extended state vector $\mathrm{x}(t)$ and u_0 is the steady state value of $u(t)$ (the output equals to the reference signal and all disturbances are zero).

A. Observer

To use state-feedback control a reliable state observer is needed. Unscented Kalman Filter was chosen as it offers a higher accuracy than the widely used Extended Kalman Filter, but simultaneously uses less computational power than e.g. particle filters. UKF uses a small number of deterministic samples, called sigma-points, to represent the probability distribution of the system state [21]. To calculate these sigma points for a given vector of stochastic variables the mean and the factor of the covariance matrix is needed. The parameters of the UKF were chosen so that the sigma points do not spread far from the mean value. To increase numerical robustness it was implemented as a square-root filter. This means that the matrix factors needed for the

sigma point calculation is updated directly using QR decomposition, instead of Cholesky factorization [20], [22], [23]. The observer works with a discrete nonlinear system of the form (7) presented below, built from the combination of the reduced model (2) and the weighting functions.

$$\begin{aligned} \mathrm{x}_{k+1} &= \mathrm{f}(\mathrm{x}_k, \mathrm{w}_k, u_k, k) \\ y_k &= \mathrm{C}\mathrm{x}_k + z_k \end{aligned} \qquad (7)$$

where $\mathrm{x}_k \in \mathbb{R}^n$, $\mathrm{w}_k \in \mathbb{R}^m$, $\mathrm{f} : \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \times \mathbb{R} \to \mathbb{R}^n$ and C is an $1 \times n$ real vector. UKF and observers using sigma points in general rely on the assumption that the state variables and disturbances have Gaussian distribution. However, it has been reported [24] that the blood glucose concentration usually follows lognormal distribution instead of the Gaussian distribution. Therefore, we introduced the transformation $\mathcal{T}$, which substitutes selected elements of a vector or matrix with their natural logarithm if they are positive real numbers. With the help of this transformation the sigma points acquired from the sigma point selection strategy will reflect lognormal distribution. The variables assumed to have lognormal distribution are states $Q_1(t)$, $Q_2(t)$, $G_1(t)$, $G_2(t)$ and the disturbance $D(t)$. It is assumed, that: $\mathcal{T}(\mathrm{w}_k) \sim \mathcal{N}(\mu_{w,k}, \mathbf{Q}_k)$, and $z_k \sim \mathcal{N}(0, R_k)$. Let us introduce the notation $\mathcal{X}$ for a set of N of sigma points, Σ for covariance matrix, $\tilde{\mathrm{f}}(.)$ for $\mathcal{T}(\mathrm{f}(.))$ and ω_m, ω_c positive real weights associated with $\mathcal{X}$. Furthermore let $Triang\{\mathbf{M}\}$ of a matrix $\mathbf{M}$ stand for the transpose of the upper triangular result $\mathbf{\Gamma}$ of the QR decomposition (8):

$$\begin{aligned} Triang\{\mathbf{M}\} &= \mathbf{\Gamma}^T \\ \mathbf{\Theta}\mathbf{\Gamma} &= QR\{\mathbf{M}^T\} \end{aligned} \qquad (8)$$

The proposed observer algorithm is as follows:

0. *Initialization*:

   a) The initial estimation $\hat{\mathrm{x}}_0$ and estimation error co-variance matrix $\mathbf{\Sigma}_0$ are chosen, so that $\mathcal{T}(\mathrm{x}_0) \sim \mathcal{N}(\mathcal{T}(\hat{\mathrm{x}}_0), \mathbf{\Sigma}_0)$.
   b) Calculate the initial set of sigma points $\mathcal{X}_0$. The disturbances and measurement noise are embedded into $\mathcal{X}_0$.

1. *Estimation*:
   $\mathcal{X}_k = \left(\begin{array}{ccc} \mathcal{X}_k^{(x)T} & \mathcal{X}_k^{(w)T} & \mathcal{X}_k^{(z)T} \end{array} \right)^T$ and control signal u_k is known.

   a) Propagate the sigma points through the nonlinear function of the system, while applying transformation:

   $$\begin{aligned} \mathcal{X}_{k+1}^{(\bar{x})} &= \tilde{\mathrm{f}}(\mathcal{T}^{-1}(\mathcal{X}_k^{(x)}), \mathcal{T}^{-1}(\mathcal{X}_k^{(w)}), u_k) \\ \mathcal{X}_{k+1}^{(\bar{x})} &= (\xi_{k+1,1}^{(\bar{x})} \cdots \xi_{k+1,N}^{(\bar{x})}) \end{aligned} \qquad (9)$$

   b) Calculate the initial estimation of the transformed state vector:

   $$\bar{\mathrm{x}}_{k+1} = \sum_{i=1}^{N} \omega_i^{(m)} \xi_{k+1,i}^{(\bar{x})} \qquad (10)$$

c) Calculate the following:

$$\mathbf{P}^{(x)} = \sum_{i=1}^{N} \sqrt{\omega_i^{(c)}} (\xi_{k+1,i}^{(\bar{x})} - \bar{\mathbf{x}}_{k+1}) \qquad (11)$$

d) Propagate the transformed sigma points to the output:

$$\begin{aligned} \mathcal{Y}_{k+1} &= \mathbf{C}\mathcal{T}^{-1}(\mathcal{X}_{k+1}^{(\bar{x})}) + \mathcal{X}_k^{(z)} \\ \mathcal{Y}_{k+1} &= (\xi_{k+1,1}^{(y)} \cdots \xi_{k+1,N}^{(y)}) \end{aligned} \qquad (12)$$

e) Calculate the initial estimation of the measured output:

$$\bar{y}_{k+1} = \sum_{i=1}^{N} \omega_i^{(m)} \xi_{k+1,i}^{(y)} \qquad (13)$$

f) Calculate the following:

$$\mathbf{P}^{(y)} = \sum_{i=1}^{N} \sqrt{\omega_i^{(c)}} (\xi_{k+1,i}^{(y)} - \bar{y}_{k+1}) \qquad (14)$$

2) *Update*:

The actual value of the measured output y_{k+1} is measured.

a) The factor $\boldsymbol{\Sigma}_{k+1}^{\frac{1}{2}}$ can be calculated with QR decomposition as follows:

$$\begin{aligned} & \begin{pmatrix} P^{(yy)} & 0 & 0 \\ \mathrm{P}_k^{(xy)} & \boldsymbol{\Sigma}_{k+1}^{\frac{1}{2}} & 0 \end{pmatrix} = \\ & = Triang \left\{ \begin{pmatrix} \mathbf{P}^{(y)} \\ \mathbf{P}^{(x)} \end{pmatrix} \right\} \end{aligned} \qquad (15)$$

b) Kalman gain is calcualted as follows:

$$\mathrm{G}_{k+1} = \mathrm{P}^{(xy)} \left(P^{(yy)} \right)^{-1} \qquad (16)$$

c) Update the estimated state vector:

$$\begin{aligned} \hat{\mathbf{x}}_{k+1} = \quad & \mathcal{T}^{-1} \left(\bar{\mathbf{x}}_{k+1} + \mathrm{G}_{k+1}(y_{k+1} - \right. \\ & \left. - \bar{y}_{k+1}) \right) \end{aligned} \qquad (17)$$

d) Calculate the new sigma point set $\mathcal{X}_{k+1}$.

3) $k = k + 1$ and continue from *Estimation*.

IV. PREDICTOR

Most observer algorithms contain the following two steps: (1) *Estimation* - calculating what the state variables should be knowing the previous step and the model dynamics, and (2) *Update* - correcting the *Estimation* based on the difference between the expected and measured output. Output measurement is not available in the case of prediction, but *Estimation* can still be performed. Kalman filter takes this further by calculating not only the expected values, but also the estimation error covariance matrix, hence providing a confidence interval for the prediction [25], [26]. The future output may be uncertain, but the fact that the output is crossing a threshold can be a vital information.

When a controller is tuned incorrectly, hypoglycemia can occur after meal intake. In the event of this, the controller may react too fast to the increasing blood glucose concentration, administer too large a dose of insulin and then although hyperglycemia may be reduced in both duration and severity, the effect of the insulin will persist long after high glucose concentration is averted. At this point the control signal hits zero and the controller cannot influence the system anymore.

Let us realize the predictor this way: When the glucose levels exceed a defined threshold after the *Update* phase of the observer, keep updating a separate set of sigma points for a given horizon. Modify the model so that no further meal intake is expected. If the predicted output with $\pm$ one sigma confidence goes below 3.9 mmol/L within the prediction horizon before the control signal reaches zero, a supervisor system could use this information to intervene and modify the control signal to avoid hypoglycemia.

The applied control law in this particular case is state-feedback. Therefore, the state estimation error of the observer cannot be neglected. For this purpose two sets of sigma points are used. The first set will have four different signals embedded: state variables, disturbance, measurement noise and estimation error: $\mathcal{X}_k^{(1)} = \left(\mathcal{X}_k^{(x,1)T} \quad \mathcal{X}_k^{(w,1)T} \quad \mathcal{X}_k^{(z,1)T} \quad \mathcal{X}_k^{(e,1)T} \right)^T$. The second set is used only to predict the error coming from state estimation. The first set of sigma points will be calculated using the vector $(\mathrm{x}_k^T \quad \mathrm{w}_k^T \quad 0 \quad 0)^T$ as mean values and covariance matrix $\boldsymbol{\Sigma}_k^{(1)}$ (18). For the second set $(\mathrm{x}_k^T \quad \mathrm{w}_k \quad 0)$ and $\boldsymbol{\Sigma}_k^{(2)}$ (19) will be used.

$$\left(\boldsymbol{\Sigma}_k^{(1)} \right)^{\frac{1}{2}} = diag \left\{ \boldsymbol{\Sigma}_k^{\frac{1}{2}}, \mathbf{Q}_k^{\frac{1}{2}}, \sqrt{R_k}, 0 \right\} \qquad (18)$$

$$\left(\boldsymbol{\Sigma}_k^{(2)} \right)^{\frac{1}{2}} = diag \left\{ 0, \mathbf{Q}_k^{\frac{1}{2}}, \sqrt{R_k} \right\} \qquad (19)$$

where $diag\{\}$ represents block-diagonal matrix.

The proposed algorithm is as follows:

0. *Initialization*:

The prediction starts once the Update step of the Observer is completed. The first set of sigma points $\mathcal{X}_k^{(1)}$ is calculated from the mean and covariance (18), and a second set $\mathcal{X}_k^{(2)}$ using (19). Reduced or zero meal intake is assumed in w_k. $l = k$ will be the starting value of the loop counter.

1) *Estimation*:

a) Calculate the control signal in accordance with the control rule (6) for both sigma point sets: $U_l^{(1)}$ from $\left(\mathcal{X}_l^{(x,1)} + \mathcal{X}_l^{(e,1)} \right)$ and $U_l^{(2)}$ from $\mathcal{X}_l^{(x,2)}$.

b) Propagate both sigma point sets through the nonlinear function of the model, while applying transformation to get: $\mathcal{X}_{l+1}^{(\bar{x},1)}$ and $\mathcal{X}_{l+1}^{(\bar{x},2)}$.

c) Calculate the initial estimation of the transformed state vector for both sigma point sets: $\bar{\mathrm{x}}_{l+1}^{(1)}, \bar{\mathrm{x}}_{l+1}^{(2)}$.

d) Calculate the same matrix as in (11) for both sets to acquire $\mathbf{P}_1^{(x)}$ and $\mathbf{P}_2^{(x)}$.

e) Propagate the transformed sigma points to the output to get $\mathcal{Y}_{l+1}^{(1)}$ and $\mathcal{Y}_{l+1}^{(2)}$.

f) Calculate $\mathbf{P}_2^{(y)}$ for only the second set of sigma points the same way as in (13) and (14).

g) The factor of the new covariance matrix of the state variables can be calculated directly from $\mathbf{P}_1^{(x)}$ using QR decomposition:

$$\left(\begin{array}{cc} \boldsymbol{\Sigma}_{l+1}^{\frac{1}{2}} & 0 \end{array} \right) = Triang\left\{ \mathbf{P}_1^{(x)} \right\} \qquad (20)$$

h) The estimation error covariance matrix will be updated using $\mathbf{P}_2^{(x)}$ and $\mathbf{P}_2^{(y)}$:

$$\begin{pmatrix} * & 0 & 0 \\ * & \boldsymbol{\Sigma}_{e,l+1}^{\frac{1}{2}} & 0 \end{pmatrix} = \\ = Triang\left\{ \begin{pmatrix} \mathbf{P}_2^{(y)} \\ \mathbf{P}_1^{(x)} \end{pmatrix} \right\} \qquad (21)$$

i) Update the sigma point set $\mathcal{X}_{l+1}^{(1)}$ with the vector $\left(\ (\bar{\mathbf{x}}_{l+1}^{(1)})^T \quad \mathbf{w}_{l+1}^T \quad 0 \quad 0 \ \right)^T$ and the block diagonal matrix $\boldsymbol{\Sigma}_{l+1}^{(1)}$ (22), and $\mathcal{X}_{l+1}^{(2)}$ with the vector $\left(\ (\bar{\mathbf{x}}_{l+1}^{(1)})^T \quad \mathbf{w}_{l+1}^T \quad 0 \ \right)^T$ and the block diagonal matrix $\boldsymbol{\Sigma}_{l+1}^{(2)}$ (23).

$$\left(\boldsymbol{\Sigma}_{l+1}^{(1)} \right)^{\frac{1}{2}} = \\ = diag\left\{ \boldsymbol{\Sigma}_{l+1}^{\frac{1}{2}}, \mathbf{Q}_{l+1}^{\frac{1}{2}}, \sqrt{R_{l+1}}, \boldsymbol{\Sigma}_{e,l+1}^{\frac{1}{2}} \right\} \qquad (22)$$

$$\left(\boldsymbol{\Sigma}_{l+1}^{(2)} \right)^{\frac{1}{2}} = \\ = diag\left\{ \boldsymbol{\Sigma}_{e,l+1}^{\frac{1}{2}}, \mathbf{Q}_{l+1}^{\frac{1}{2}}, \sqrt{R_{l+1}} \right\} \qquad (23)$$

j) $l = l + 1$ and repeat while $l \leq k + N_p$, with N_p being the prediction horizon.

V. RESULTS

The capabilities of the proposed framework were tested using MATLAB-simulink environment. The controller was purposefully modified to allow oscillatory transients. The controlled system is now very likely to go into severe hypoglycemia following a meal intake. The prediction horizon was set to 3 hours. The simulation started at 6:00 a.m., and lasted for 24 hours. 3 meals were administered to the virtual patient:

- Breakfast at 9:30 a.m. containing 45 g carbohydrates (CHO),
- lunch at 1:30 p.m. with 75 g CHO,
- and finally 85 g CHO for dinner at 7:30 p.m.

The predictor was turned on once the estimated output went above 6 mmol/L. Two example of prediction is presented on Fig. 3 and Fig. 4.

It is visible that the predictor can detect potential hypoglycemic episodes, even before severe hyperglycemia is reached. Furthermore, the insulin signal is sill not zero when this detection happens. An appropriate supervisor system could intervene and avoid low glucose levels. It is also visible, that real output is usually covered by the $\pm$ one sigma interval of the prediction. The mean value of the prediction is usually below the actual output. The reason is that the observer estimates the state variables related to
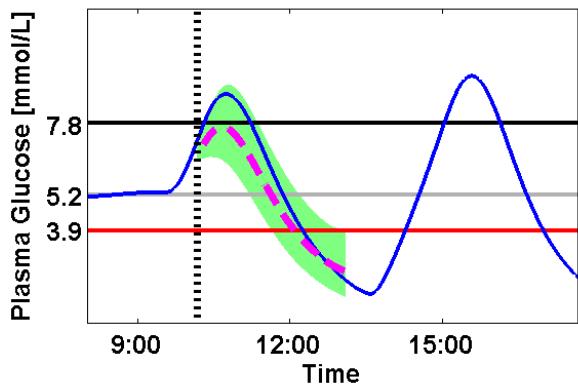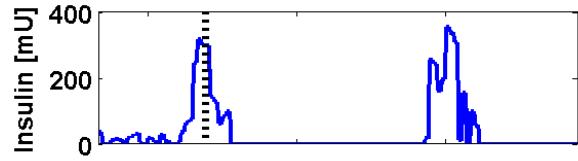

Fig. 3. Prediction after the first meal. (a) Bottom: The three solid horizontal lines represent the border of hypoglycemia (3.9 mmol/L), the border of hyperglycemia (7.8 mmol/L) and the reference signal (5.2 mmol/L). The other solid line is the real glucose concentration, while the dashed line is the 3-hour prediction. The transparent area around the prediction is the $\pm\sigma$ confidence interval. The horizontal dotted line is the time instance when the prediction takes place. (b) Top: Control signal (solid line) and starting time of prediction (dotted line).
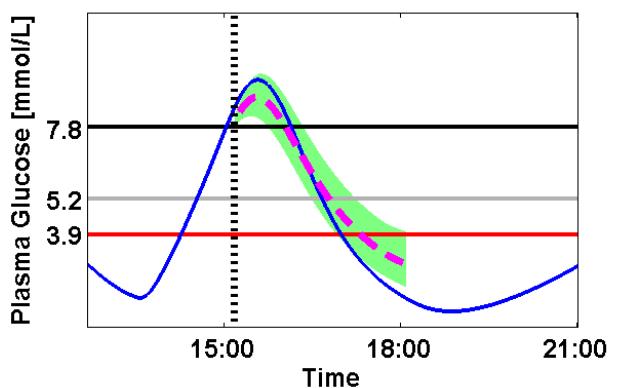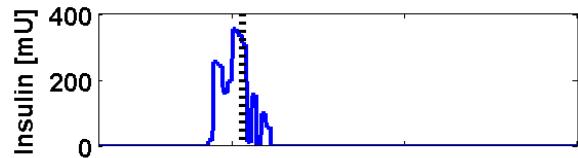

Fig. 4. Prediction after the second meal. (a) bottom: Solid line - real glucose concentration. The three horizontal lines - border of hypoglycemia (3.9 mmol/L), border of hyperglycemia (7.8 mmol/L) and the reference signal (5.2 mmol/L). Dashed line - prediction of the glucose concentration. Transparent area - $\pm\sigma$ confidence interval. Horizontal dotted line - start of prediction. (b) top: Control signal (solid line), starting time of prediction (dotted line).

glucose ingestion ($G_1(t)$ and $G_2(t)$) with a little delay. This, combined with the assumption that there will be no more meal intake, causes the predictor to estimate lower

glucose levels. It is trivial that if another meal intake occurs within the prediction horizon the predicted and real glucose concentration will deviate from each other. Situations like this must be handled by the supervisor algorithm, and hence it is not in the scope of the investigation.

The proposed predictor is a general framework, that could be used with any T1DM models and controllers which are compatible with Kalman filter. The more effective the controller is in disturbance rejection, the tighter the confidence interval will be. The Unscented Kalman filter could be substituted as well, with an EKF, a sigma-point filter with different sigma point selection strategy or a particle filter, etc. Sophisticated stochastic models of meal intake, physical activity and other disturbances can improve the quality of prediction even further.

VI. CONCLUSIONS

A Kalman filter-based predictor framework was proposed to detect future hypoglycemic episodes during closed-loop tight glycemic control. The predictor was able to predict potential hypoglycemia up to 3 hours in advance during simulation using the Cambridge-model. The predictions occurred before the control signal dropped to zero, hence allowing intervention. Future works would focus on the development of a supervisor algorithm, which uses this predictor to enhance pure feedback controllers.

REFERENCES

[1] A. Fonyó, E. Ligeti, Physiology (in Hungarian), Budapest, Hungary: Medicina, 2008.

[2] F. Chee, T. Fernando, Closed-Loop Control of Blood Glucose, Springer-Verlag, Berlin, Germany, 2007, 1st edition.

[3] C. Cobelli, E. Renard, B. Kovatchev, Artificial Pancreas: Past, present and future, Diabetes, vol. 60, no. 11, pp. s672–2682, 2011.

[4] V. N. Shah, A. Shoskes, B. Tawfik, S. K. Garg, Closed-Loop System in the Management of Diabetes: Past, Present, and Future, Diabetes Technol. Ther., vol. 16, no. 8, pp. 477–490, 2014.

[5] R. Hovorka, D. Elleri, H. Thabit, J. M. Allen, L. Leelarathna, R. El-Khairi, K. Kumareswaran, K. Caldwell, P. Calhoun, C. Kollman, H. R. Murphy, C. L. Acerini, M. E. Wilinska, M. Nodale, D. B. Dunger, Overnight closed-loop insulin delivery in young people with type 1 diabetes: a free-living, randomized clinical trial, Diabetes Care, vol. 37, no. 5, pp. 1204–1211, 2014.

[6] J. Kropff, S. Del Favero, C. Toffanin, R. Visentin, M. Monaro, M. Messori, F. Di Palma, G. Lanzola, A. Farret, F. Boscari, S. Galasso, P. Magni, A. Avogaro, P. Keith-Hynes, B.P. Kovatchev, D. Bruttomesso, C. Cobelli, J. H. DeVries, E. Renard, L. Magni: 2 month evening and night closed-loop glucose control in patients with type 1 diabetes under free-living conditions: a randomised crossover trial, Lancet Diabetes Endocrinol., 2015.

[7] M. Somogyi, M. Kirstein, Insulin as a cause of extreme hyperglycemia and instability, Weekly Bulletin of the St Louis Medical Society, vol. 32, pp. 498–510, 1938.

[8] V. N. Shah, A. Shoskes, B. Tawfik, S. K. Garg, Closed-loop system in the management of diabetes: past, present, and future, Diabetes. Technol. Ther., vol. 16, no. 8, pp.477–490, 2014.

[9] B. Kovatchev, C. Cobelli, E. Renard, S. Anderson, M. Breton, S. Patek, W. Clarke, D. Bruttomesso, A. Maran, S. Costa, A. Avogaro, C. Dalla Man, A. Facchinetti, L. Magni, G. De Nicolao, J. Place, A. Farret, Multinational study of subcutaneous model-predictive closed-loop control in type 1 diabetes mellitus: summary of the results, J. Diabetes Sci. Technol., vol. 4, no. 6, pp. 1374–1381, 2010.

[10] F. Stahl, R. Johansson, Receding horizon prediction by Bayesian combination of multiple predictors, in Proc. IEEE 51st Annual Conference on Decision and Control, pp. 5278–5285, 2012.

[11] K. Plis, R. Bunescu, C. Marling, J. Shubrook, F. Schwartz, A Machine Learning Approach to Predicting Blood Glucose Levels for Diabetes Management, 28th AAAI Conference on Artificial Intelligence, pp. 35–39, 2014.

[12] R. Hovorka, V. Canonico, L. Chassin, U. Haueter, M. Massi-Benedetti, M. O. Federici, T. Pieber, H. Schaller, L. Schaupp, T. Vering, M. Wilinska, Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes, Physiol. meas., vol. 25, pp. 905–920, 2004.

[13] M. Wilinska, L. Chassin, C. Acerini, J. Allen, D. Dunger, R. Hovorka, Simulation environment to evaluate closed-loop insulin delivery systems in type 1 diabetes, J. Diab. Sci. Techn., vol. 4, no. 1, pp. 132–144, 2010.

[14] P. Szalay, P. I. Sas, L. Barkai, L. Kovács, Nonlinear analysis of Type 1 Diabetes Models by Differential Geometric Approach, in Proc. 8th IFAC Symposium on Biological and Medical Systems, Budapest, Hungary, 2012, pp. 55–60.

[15] P. Szalay, G. Eigner, L. Kovacs, Linear Matrix Inequality-Based Robust Controller Design for Type-1 Diabetes Model, in Proc. 19th IFAC World Congress, vol. 19, no. 1, Cape Town, South Africa, 2014, pp. 9247-9252.

[16] K. Zhou, Robust and Optimal Control, Prentice Hall, New Jersey, 1996.

[17] C. Scherer, S. Weiland, Linear Matrix Inequalities in Control, Lecture Notes, Delft Center for Systems and Control, Delft, The Netherlands, 2004.

[18] M. Grant, S. Boyd, CVX: Matlab software for disciplined convex programming, version 2.0 beta. http://cvxr.com/cvx, September 2013.

[19] M. Grant, S. Boyd, Graph implementations for nonsmooth convex programs, Recent Advances in Learning and Control (a tribute to M. Vidyasagar), Lecture Notes in Control and Information Sciences, V. Blondel, S. Boyd, and H. Kimura, editors, Springer, pp. 95–110, 2008.

[20] P. Szalay, A. Molnár, M. Müller, G. Eigner, I. Rudas, Z. Benyó, L. Kovács, Comparison of sigma-point filters for state estimation of diabetes models, in Proc. 2014 IEEE Int. Conference on Systems, Man and Cybernetics, pp. 2476–2481, 2014,

[21] S. J. Julier, J. K. Uhlmann, Unscented filtering and nonlinear estimation, Proceedings of the IEEE, vol. 93, no. 3, pp. 401–422, 2004.

[22] M. G. Rutten, Square-root unscented filtering and smoothing, in Proc. IEEE 8th Int. Conf. on Intelligent Sensors, Sensor Networks and Information Processing, pp. 294–299, 2013.

[23] P. Szalay, L. Szilágyi, Z. Benyó, L. Kovács, Sensor Drift Compensation Using Fuzzy Interference System and Sparse-Grid Quadrature Filter in Blood Glucose Control, Lecture Notes in Computer Science: Neural Information Processing, Springer International Publishing, vol. 8835, pp. 445.-453, 2014.

[24] A. K. Duun-Henriksen, S. Schmidt, R. Røge, J. Bech, K. Nørgaard, J. B. Jørgensen, H. Madsen, Model identification using stochastic differential equation grey-box models in diabetes, Journal of Diabetes Science and Technology, vol. 7, pp. 431-440, 2014.

[25] Qi Wenjuan, Deng Zili, Several Weighting Fusion Kalman Predictors with Colored Measurement Noises and their Accuracy Comparison, in Proc. Fifth International Conference on Measuring Technology and Mechatronics Automation, pp. 989–992, 2013.

[26] G. G. Rigatos, P. Siano, Power corporations' default probability forecasting using the Derivative-free nonlinear Kalman Filter, in Proc. IEEE 13th International Conference on Industrial Informatics, pp.1165–1170, 2015.