

Meal estimation from Continuous Glucose Monitor data using Kalman filtering and hypothesis testing

Odd Martin Staal^{1,2}, Steinar Sælid², Anders Lyngvi Fougner¹ and Øyvind Stavadahl¹

Abstract—A method for estimating meal inputs from Continuous Glucose Monitoring (CGM) data is presented. The method is based on Kalman filtering and hypothesis testing, and provides estimates of the time the meal was initiated and the carbohydrate content of the meal. The sensitivity to model correctness is evaluated, and suggestions for how the method can be tuned and extended are given. The method is tested on synthetic data from two simple, individualisable models of glucose dynamics as well as on real CGM data. The method has potential as a meal detector and estimator in a data cleaning settings as well as in a real-time, artificial pancreas (closed-loop glucose control) setting. Further research is needed to determine its performance on larger data sets and compare it to other methods.

I. INTRODUCTION

Blood glucose (BG) dynamics models aim to predict future blood glucose levels based on information about past and current blood glucose, meals, insulin, exercise and other inputs. Examples of BG dynamics models are given in [1]–[4]. Such models are used in the context of artificial pancreas, advisor apps for people with diabetes, or in glucose measurement equipment where estimates are needed during periods of noisy or missing signal. Another application is the use in *in silico* trials of new equipment, where e.g. artificial pancreas systems can be tested in a simulator. The UVA-Padova model has been used in the T1DMS simulator for this purpose [5].

One of the drawbacks of real-life use of BG dynamics models is that they rely heavily on precise and timely information about the inputs to the model to make a good prediction. Two main inputs that are involved are meals (usually represented by the amount of carbohydrates in g, hereafter abbreviated CHO) and insulin injections (represented by the injected insulin dose in international units, U). The accuracy of such inputs is important also in other contexts, for instance when identifying or individualizing the model through parameter estimation.

Identification of BG dynamics models have traditionally been possible only when using detailed and accurate data from clinical research studies, where the amounts and times of carbohydrates (often in the form of pure glucose) and insulin administered to the patient are accurately known. In data from free-living use of continuous glucose monitor

(CGM) and insulin pump systems the information about meals is often lacking, incomplete or inaccurate. This is one of the hurdles preventing this large amount of data from being usable for precise model identification in research settings, and methods that are able to reconstruct missing meal and insulin information in such data would be valuable. Insulin injection data is recorded in an accurate and timely manner by the insulin pump, for individuals who use those, and is available for use in offline and online processing. Precise and timely estimation of meal information is more difficult. There have been efforts towards automatic meal detection systems based on different external systems that detect chewing [6] or abdominal sounds [7], [8]. A current drawback of these systems is that they may *detect* the food intake time, but have little ability to estimate the carbohydrate content of the meal.

Other researchers have investigated systems for automatic carbohydrate estimation based on e.g. images [9]. The drawback here is that while the estimate of the carbohydrate content may be more accurate through computer vision, meal announcement by the user is still needed, as he or she will have to remember to take a picture of every meal. The risk that the user will forget the CHO estimation task altogether for many meals is still present even with such systems in place.

A third option investigated by many research groups is to detect the meal from CGM data [10]–[21]. For real-time use of a meal estimator in an artificial pancreas (AP) the foremost concern with this approach is the latency from a meal is eaten until it is detected and a compensatory insulin bolus can be injected by the AP. The latency of detection can be split into physiological, sensor and algorithmic delay. The physiological delay originates from digestion taking a while to break down and transport the glucose of the food into the blood, and diffusion from the blood to the interstitial fluid where the CGM sensor resides. Next, the CGM sensor has delays related to detecting, aggregating, filtering and transmitting the glucose measurement to the piece of equipment that is responsible for the meal estimation. It is common in CGM systems that a new sample is produced only every 5 minutes. Finally, the estimation algorithm needs to process a certain number of samples to achieve confidence that a meal is in effect. At this point the total delay may well be close to 40 minutes [17]. Ideally the insulin meal bolus should have been injected at the time of the meal, or even before, and setting it 40 minutes after the meal will increase the probability of unwanted hyperglycemia and subsequent hypoglycemia [22], [23].

Invited paper, submitted 15 March 2019. Accepted 19 July 2019.

*This work was supported by the Research Council of Norway, project numbers 242167 and 248872.

¹Dept. of Engineering Cybernetics, Norwegian University of Science and Technology, Trondheim, Norway. odd.m.staal@ntnu.no, oyvind.stavadahl@ntnu.no, anders.fougner@ntnu.no

²Prediktor Medical, Gamle Fredrikstad, Norway. ss@prediktor.no

Lacking external systems for meal detection, meal announcement by the user is currently the main option for obtaining meal information to use in BG dynamics models in free-living settings. This implies that the user must estimate the relevant quantity (e.g. grams of carbohydrates in the meal) and provide the information to the system at the time of intake. The data is either logged manually, or input into different logging systems in insulin pumps or CGMs, for instance in the insulin pump bolus calculator logs, since the user typically must enter a meal CHO estimate for the pump to calculate a proposed insulin bolus. One of the problems with relying on user meal announcements is that the carbohydrate estimation by the user is often inaccurate, both in time and value. An example of this is reported by [9], where the CHO estimation by diabetes patients was off by 28 g on average. In other cases, users will forget to announce the meal entirely. The influence of such errors on glucose prediction and model identification could be large.

In this paper we investigate a model-based method for meal estimation from CGM data using a hypothesis testing Kalman filter (KF). The method has offline applications, e.g. in cleaning data sets to be used for parameter estimation in BG dynamics models. It has potential applications in conjunction with meal detection in AP systems as it can run in on-line, real-time settings.

The method is inspired by [24], [25], and is denoted CHP in the following, after Chan, Hu and Plant. A difference is that our method uses hypothesis testing, allowing us to test all hypothetical inputs to the system in a window of evaluated past samples. Also, the estimated meal size and time is given by the CHP estimator directly when a detection is made. To the best of our knowledge, the use of the CHP algorithm for meal detection from CGM data is novel.

Our method has similarities with other recently published methods for meal detection and estimation:

- Mahmoudi *et al.* [10] use a Kalman filter that is running a linear individualized model to detect meals, using tests on the innovation sequence from the Kalman filter, and thresholding on the estimate of the CHO input, which is modelled as double integrated white noise. They use a Rauch-Tung-Striebel smoother after a detection is made, to estimate the meal size, and test their method on simulated data.
- Kölle *et al.* [11], [12] use a different approach where a Moving Horizon Estimator (MHE) is used together with Linear Discriminant Analysis (LDA) to detect meals. The MHE is used to estimate the states in an augmented version of Bergman’s minimal model from a moving window of measurements. One of the states in the model is the glucose rate of appearance in plasma, R_a . The LDA is trained to classify profiles of R_a within the window into classes ”meal onset” and ”no meal onset”. The method flags meals only, and does not report an estimate for the time of the meal or its value, but could likely be extended to do so.

II. METHOD

This section describes the CHP estimator used to estimate meals from CGM data.

A. Kalman filter input detection

A BG dynamics model is running in a Kalman filter estimator, which uses CGM readings to update its internal states. The BG dynamics model is assumed to be linear, of the form:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}) = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad (1)$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k \quad (2)$$

Inputs are assumed to be impulse-like. This is a common representation of meals and insulin boluses in BG dynamics model models. They are turned into discrete-time impulses by dividing by the step time Δt .

The system A and B matrices are transformed to discrete form by constructing the matrix $\mathbf{E} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ 0 & 0 \end{bmatrix}$, computing the matrix exponential of \mathbf{E} and picking out the discretized matrices from the resulting matrix: [26].

$$e^{\mathbf{E}\Delta t} = \begin{bmatrix} \mathbf{A}_d & \mathbf{B}_d \\ 0 & \mathbf{I} \end{bmatrix} \quad (3)$$

where \mathbf{A}_d is the discrete transition matrix and \mathbf{B}_d is the discrete input matrix.

Process and measurement noises are added as per standard Kalman filtering theory:

$$\mathbf{x}_{k+1} = \mathbf{A}_d\mathbf{x}_k + \mathbf{B}_d\mathbf{u}_k + \mathbf{v}_k \quad (4)$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{w}_k \quad (5)$$

$$\mathbf{v}_k \sim \mathcal{N}(0, \mathbf{Q}) \quad \mathbf{w}_k \sim \mathcal{N}(0, \mathbf{R}) \quad (6)$$

A time step of $\Delta t = 1\text{min}$ and 5min has been used in this work. One minute is low enough to give small discretization errors in most BG dynamics models using forward Euler numerical integration. Five minutes was used when testing against real CGM data.

We work in the following with a system where we test the hypothesis that an impulse input has been applied at time step k_u in the past. We use k to denote the most recent (current) time step or a generic time step. Variables with a superscript u belong to the hypothesis that an input has been applied, while variables without the u superscript belong to the no-input hypothesis, or are common for both hypotheses. We emphasize that the no-input filter can and should be provided with all the *known* inputs from a data set, including any meals that were announced.

In the Kalman filter the state estimate is computed by first performing a time update and then a measurement update, given by the following equations:

$$\bar{\mathbf{x}}_k = \mathbf{A}_d\hat{\mathbf{x}}_{k-1} + \mathbf{B}_d\mathbf{u}_k \quad (7)$$

$$\hat{\mathbf{x}}_k = \bar{\mathbf{x}}_k + \mathbf{K}_k(\mathbf{y}_k - \mathbf{C}\bar{\mathbf{x}}_k) \quad (8)$$

where $\bar{\mathbf{x}}_k$ is the *a priori* estimate, $\hat{\mathbf{x}}_k$ is the *a posteriori* estimate, and \mathbf{K}_k is the Kalman gain matrix at time step k . It

is commonly the case in glucose data sets that measurements are not available for every time step. To handle this we set $\hat{\mathbf{x}}_k$ equal to $\bar{\mathbf{x}}_k$ for the time steps where the measurement is missing, by setting K_k equal to zero at these times. When measurements are available, the K_k matrix is computed using the normal Kalman filter equations, see e.g. [27].

Combining Eqs. (7) and (8) to eliminate the *a priori* estimate we get the following expression for the propagation of the *a posteriori* estimate:

$$\begin{aligned}\hat{\mathbf{x}}_k &= (I - K_k C)\bar{\mathbf{x}}_k + K_k \mathbf{y}_k \\ &= (I - K_k C)(A_d \hat{\mathbf{x}}_{k-1} + B_d \mathbf{u}_k) + K_k \mathbf{y}_k \\ &= M_k A_d \hat{\mathbf{x}}_{k-1} + M_k B_d \mathbf{u}_k + K_k \mathbf{y}_k\end{aligned}\quad (9)$$

where we have substituted $M_k = (I - K_k C)$.

If no input is applied, we have $\mathbf{u} \equiv \mathbf{0}$, and the propagation of the Kalman filter's *a posteriori* estimates can be written

$$\begin{aligned}\hat{\mathbf{x}}_k &= M_k A_d \hat{\mathbf{x}}_{k-1} + K_k \mathbf{y}_k \\ \hat{\mathbf{x}}_{k+1} &= M_{k+1} A_d \hat{\mathbf{x}}_k + K_{k+1} \mathbf{y}_{k+1} \\ &= M_{k+1} A_d (M_k A_d \hat{\mathbf{x}}_{k-1} + K_k \mathbf{y}_k) + K_{k+1} \mathbf{y}_{k+1} \\ &\vdots\end{aligned}\quad (10)$$

On the other hand, if an impulse control input with magnitude \mathbf{u}_k is applied at time k, the estimates would instead be

$$\begin{aligned}\hat{\mathbf{x}}_k^u &= M_k A_d \hat{\mathbf{x}}_{k-1} + M_k B_d \mathbf{u}_k + K_k \mathbf{y}_k \\ &= \hat{\mathbf{x}}_k + M_k B_d \mathbf{u}_k \\ \hat{\mathbf{x}}_{k+1}^u &= M_{k+1} A_d \hat{\mathbf{x}}_k^u + K_{k+1} \mathbf{y}_{k+1} \\ &= M_{k+1} A_d (M_k A_d \hat{\mathbf{x}}_{k-1} + M_k B_d \mathbf{u}_k + K_k \mathbf{y}_k) \\ &\quad + K_{k+1} \mathbf{y}_{k+1} \\ &= \hat{\mathbf{x}}_{k+1} + M_{k+1} A_d M_k B_d \mathbf{u}_k \\ &\vdots\end{aligned}\quad (11)$$

We see that the difference between the $\hat{\mathbf{x}}^u$ and $\hat{\mathbf{x}}$ series is only an additive propagation of the $B\mathbf{u}$ term.

The no-input Kalman filter innovations are

$$\varepsilon_k = \mathbf{y}_k - C\hat{\mathbf{x}}_k. \quad (12)$$

With an input, the innovation at time k would be

$$\varepsilon_k^u = \mathbf{y}_k - C\hat{\mathbf{x}}_k^u. \quad (13)$$

We define the following sequences starting at k_u , the hypothesized time step of the input:

$E = [\varepsilon_{k_u} \varepsilon_{k_u+1} \dots \varepsilon_k]^T$ are the no-input innovations.

$E_u = [\varepsilon_{k_u}^u \varepsilon_{k_u+1}^u \dots \varepsilon_k^u]^T$ are the with-input innovations.

The sequences are related by

$$E = E_u + \Psi \mathbf{u}_{k_u} \quad (14)$$

where the matrix Ψ is given by

$$\Psi = \begin{bmatrix} HM_{k_u} B_d \\ HM_{k_u+1} A_d M_{k_u} B_d \\ \vdots \\ C \left(\prod_{j=k_u+1}^k M_j A_d \right) M_{k_u} B_d \end{bmatrix} = \begin{bmatrix} T_{k_u}^0 \\ T_{k_u}^1 \\ \vdots \\ T_{k_u}^{k-k_u} \end{bmatrix} \quad (15)$$

The term $T_{k_u}^i$ describes how applying an unknown control input at time k_u affects the innovations i time steps later in time. The T terms are scalars for single-input single-output (SISO) systems. For multiple-input multiple-output (MIMO) systems T is a $n_y \times n_u$ matrix. In the following we assume the SISO case, which is applicable when only the meal input is to be estimated using CGM glucose measurements, which is the most relevant use case.

We are now set to find the most likely time step of a control input, \hat{k}_u . This is found as in [28] by computing the statistic:

$$\Delta L_k(k_u) = \frac{\left| \sum_{i=0}^{k-k_u} T_{k_u}^i \varepsilon_{k_u+i} \right|^2}{2 \sum_{i=0}^{k-k_u} (T_{k_u}^i)^2} \quad (16)$$

for every candidate time step k_u . The k_u giving the largest value of ΔL_k is the most likely time of input, and is called \hat{k}_u . To reduce false detections, the ΔL_k must be above a certain threshold ΔL_{min} for a detection to be flagged.

Using weighted least squares optimization to minimize $E_u^T E_u$ we obtain the optimal input estimate given that the input happened at \hat{k}_u as

$$\hat{u} = (\Psi^T \Sigma^{-1} \Psi)^{-1} \Psi^T \Sigma^{-1} E \quad (17)$$

where the weighting matrix Σ contains the innovation covariances from the Kalman filter, a diagonal matrix in the SISO case. If the system is MIMO, Σ is a block diagonal matrix.

If we write out Eq.17 in details for the scalar SISO case, we get

$$\hat{u} = \frac{\sum_{i=0}^{k-k_u} T_{k_u}^i \varepsilon_{k_u+i} / \omega_{k_u+i}}{\sum_{i=0}^{k-k_u} (T_{k_u}^i)^2 / \omega_{k_u+i}} \quad (18)$$

Here, ω_k is the innovation process variance at time k, given by

$$\omega_k = C \bar{P}_k H + R_k \quad (19)$$

where R_k is the measurement variance from Eq. 6 and \bar{P}_k is the *a priori* state estimation covariance from the Kalman filter.

The estimated \hat{u} is in the form of a discretized impulse, so to go back to a continuous impulse we need to multiply it with Δt to get the estimated meal size.

Once an input has been detected we can update the state estimate by the following equations:

$$\hat{x}_{k,new} = M_{k_u}^k \hat{u} \quad (20)$$

$$\hat{P}_{k,new} = \hat{P}_k + \frac{(M_{k_u}^k)^T T_{k_u}^k M_{k_u}^k}{\sum_{i=0}^{k-k_u} (T_{k_u}^i)^2} \sigma_\varepsilon^2 \quad (21)$$

where

$$M_{k_u}^k = \left(\prod_{j=k_u+1}^k M_j A_d \right) M_{k_u} B_d \quad (22)$$

and σ_ε^2 is the variance of the innovation process of the no-input Kalman filter, i.e. $C^T \bar{P} C + R$.

B. Models

The input estimation described above needs a model describing the glucose dynamics. We have tested it on two simple models that are described in the following. The models are individualizable to some extent, and while they lack a physiological grounding they are observable and thus suitable for estimation use.

1) *Model A : 3 state model with first order input dynamics*: Model A is a very simple linear model, described by the following equations:

$$\mathbf{x} = \begin{bmatrix} G_p \\ I \\ M \end{bmatrix} \begin{array}{l} \text{- Plasma glucose} \\ \text{- Plasma insulin} \\ \text{- Meal glucose rate of appearance} \end{array}$$

$$\mathbf{f}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} \dot{G}_p \\ \dot{I} \\ \dot{M} \end{bmatrix} = \begin{bmatrix} \theta_1 - \theta_2 I + \theta_4 M \\ -\frac{1}{\theta_3} I + u_i \\ -\frac{1}{\theta_5} M + u_m \end{bmatrix}$$

Here u_i is the insulin injection [U], and u_m is the meal intake [g]. θ s are person dependent parameters. Nominal values for the parameters of model A used in this work are $\theta_{A0} = [0 \ 0.04 \ 30 \ 0.015 \ 30]^T$

2) *Model B : 5 state model with second order input dynamics*: The model described by Magdelaine *et al.* [3] exists in a reduced version intended to improve parameter identifiability [29]. This model has been adopted as model B in our work, and is described below.

$$\mathbf{x} = \begin{bmatrix} G_p \\ I \\ I_2 \\ M \\ M_2 \end{bmatrix} \begin{array}{l} \text{- Plasma glucose} \\ \text{- Plasma insulin} \\ \text{- Input compartment insulin} \\ \text{- Meal glucose rate of appearance} \\ \text{- Input compartment meal} \end{array}$$

$$\mathbf{f}(\mathbf{x}, \mathbf{u}) = \begin{bmatrix} \dot{G}_p \\ \dot{I} \\ \dot{I}_2 \\ \dot{M} \\ \dot{M}_2 \end{bmatrix} = \begin{bmatrix} \theta_1 - \theta_2 I + \theta_4 M \\ \frac{1}{\theta_3} (I_2 - I) \\ -\frac{1}{\theta_3} I_2 + u_i \\ \frac{1}{\theta_5} (M_2 - M) \\ -\frac{1}{\theta_5} M_2 + u_m \end{bmatrix}$$

Inputs u_m and u_i have the same meanings as in Model A. Nominal values for the parameters of model B used in this work are $\theta_{B0} = [0 \ 0.04 \ 30 \ 0.02 \ 20]^T$. Model A can be considered a reduced version of Model B.

An illustration of the two models' ability to roughly approximate a glucose trajectory is shown in Fig. 1. The nominal parameter sets for model A and B have been found through manual parameter tuning to the $P1_{real}$ dataset described in Sec. III-A).

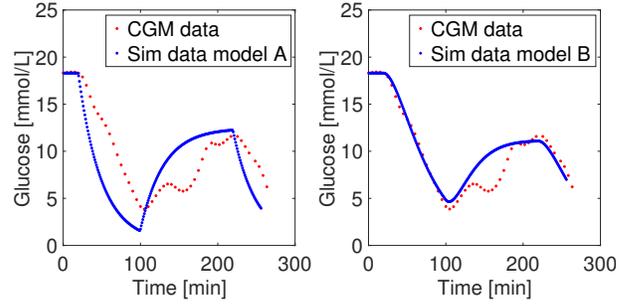


Fig. 1. *Left*: Simulation of Model A with parameter set θ_{A0} . *Right*: Simulation of Model B with parameter set θ_{B0} . Both are plotted against a limited set of CGM data from a person with T1DM containing two insulin boluses and one meal ($P1_{real}$ described in Sec. III-A).

3) *Strictly non-negative states*: The models described, although seemingly linear, contain a hidden non-linearity; all states are strictly non-negative. The states of models A and B describe glucose, insulin and meals, which are never negative. We enforced this in the Kalman filter measurement update by setting any states that were estimated to below zero, to zero.

C. Adaptations for glucose estimation

Some extra logic was added to the estimator to make it more suitable for glucose tracking. The CHP method was originally intended for tracking aircraft and detecting inputs in the form of pilot manoeuvres, which can be both positive and negative. In the glucose case, only positive meal inputs are allowed. Thus, if the meal size estimated in Eq. 18 is negative, a detection is not flagged.

It is also possible to threshold based on the estimated meal size, in addition to or as a replacement of the ΔL_{min} threshold. We then only flag detections having a corresponding estimated meal value higher than a certain amount. In the work reported here, we have used both types of thresholds, using 10 g for the estimated meal threshold.

Finally, the estimator behaviour after a detection can be discussed. In initial tests we required the estimator to wait for a number of samples equal to the number of stored matrices backward in time (N_{back}) after a detection before a new detection was possible. Another option is to keep the estimator running, this will often lead to estimating a meal for subsequent samples and updating the meal size as more data arrives.

1) *Tuning of parameters*: The method has several tunable parameters. The Kalman filter has the noise parameters Q (process noise covariance), R (measurement noise covariance) and P_0 (initial state covariance) that need to be set. The CHP estimator has a threshold for detection ΔL_{min} and a window length of backward samples N_{back} that can be tuned.

The measurement covariance matrix R in the Kalman filter describes the uncertainty in the measurements and is thus given by the uncertainty of CGM systems. We used $R = 0.16$ (mmol L⁻¹)² which is comparable to 60 (mg dL⁻¹)², or 0.18 (mmol L⁻¹)², which was used by Mahmoudi *et al.* [10].

The value of the initial state covariance P_0 has been set to $10^3 \mathbf{I}$ in this work, i.e. the initial state is unknown. The P_0 matrix only has an effect in the start of the estimation, after some measurements the state covariance will converge to a value determined by Q and R .

The process noise given by Q controls how rapidly the state covariance increases during time stepping, and the balance of the process noise to the measurement noise determines how close to the measurements the filter will stay. Setting the process noise too high will make the Kalman filter follow the measurements closely, and no meal will ever be detected. Setting the process noise too low will make the Kalman filter more insistent on its predictions, which makes prediction errors more likely, increasing the probability of detecting meals. So the process noise is directly related to the estimator's ability to detect meals, and its false positive/false negative rate. The process noise Q should be optimized together with ΔL_{min} and N_{back} to give the best performance of the meal detector on real data. This requires annotated data sets where the meal times and values are logged.

Initial setting of Q can be done based on synthetic data, by using the assumption that the model difference between model A and B is not larger than the difference from model A or B to real-life glucose dynamics. Given the simplicity of models A and B this should be safe to assume. Q was set to $1e^{-6} \mathbf{I}\Delta t$ for both models. The sampling interval of the data was 1 minute, the estimator window N_{back} was 30 minutes (30 samples), and $\Delta L_{min} = 20$ in the following, unless otherwise indicated.

2) *Use of insulin data:* When trying the method on real data sets from the Ohio study [30], we saw many cases where providing the insulin inputs directly into the model as a control input resulted in a false meal detection. This happens when the insulin input forces the Kalman filter to give lower predictions, because the model response is more rapid than the real response. This in turn causes the CHP estimator to falsely detect a meal. This led us to run the estimator without using the insulin information. In many cases this improved the detections, giving less false positives.

One solution to this problem would be to individualize model parameters, either by reducing the insulin sensitivity parameter or increasing the insulin transport time constant. More complex models could also be considered. While model individualization may be feasible and desirable in some uses of this method, e.g. in a personalized artificial pancreas, it is not desirable for data cleaning purposes. Then it would be an advantage if the method did not need to be individually tailored to each data set.

An alternative way to use the insulin information is proposed to this end. Instead of entering insulin information directly into the model through u_i , we let it affect the process noise on the insulin state(s), weighted by the square of the magnitude of the insulin input:

$$Q_{total} = Q + u_i^2 Q_{insAdd} \quad (23)$$

where Q_{insAdd} is a covariance matrix with zeros for all states except insulin states. Q_{total} is used instead of Q in the

Kalman filter time update. This has the effect of signaling to the Kalman filter that the insulin state is more uncertain from this point onward, which makes the Kalman filter stay closer to the measurements. We used noise magnitudes of $10 - 2$ for the nonzero elements of Q_{insAdd} in this work, but this should also be considered a tunable parameter.

3) *Handling 5 min sampling intervals:* To make the method appropriate for processing real CGM data, it is desirable to use data with a 5 min sampling interval. The step size of the Kalman filter and CHP estimator can be changed to 5 min, but it has implications for the estimation accuracy. We used 5 minutes sampling interval when testing against the real CGM data.

III. TESTS AND RESULTS

A. Data

We have tested the algorithm on different data sets, both simulated and real. Data set $P1_{real}$ is a 4-hour CGM recording from a person with type 1 diabetes, with two insulin boluses (15 and 10 U) and one meal (27 g). Data set $P1_{sim}$ is synthetic, generated by fitting the models A and B to the $P1_{real}$ data. The parameter vectors used when simulated are θ_{A0} and θ_{B0} described in Sec. II-B. Data set *Ohio P X day Y* is real CGM data from the Ohio data set [30], Patient id X, day Y, where Y is the number of days since the start of the CGM recording for each patient. These data contain basal and bolus insulin information, announced meals and several cases of unannounced meals. For the initial testing reported here we selected 24-hour data sets that had the least amount of obvious data errors.

B. Ideal case

The method was first tested on ideal case simulated data, $P1_{sim}$, using the same model in the estimator as was used to generate the data, with 1 minute sampling interval. The results are shown in Fig. 2, and show that the estimator works as intended. The estimators both estimate the meal to be at 99 min (true value 100 min) with a value of 28.1 g and 28.5 g for model A and B respectively (true value 27 g). The estimator using model B detects the meal 17 minutes after it was input to the model, while the estimator using model A detected the meal 7 minutes after the meal was input. The faster detection time of model A is due to the more simple dynamics of this model giving a more abrupt signal change as a response to the meal, and does not imply that it is a better model for use in meal detection.

C. Model dependency

The method's sensitivity to modelling inaccuracy was investigated by switching models for data generation and estimation, i.e. running the meal estimator with model A on simulated data generated with model B, and vice versa. The results are shown in Fig. 3. As could be expected, the estimator is influenced by model mismatch. The most obvious error is in the CHO content estimation, which becomes quite wrong for the estimator using model B when faced with data generated from model A, estimating the 27 g

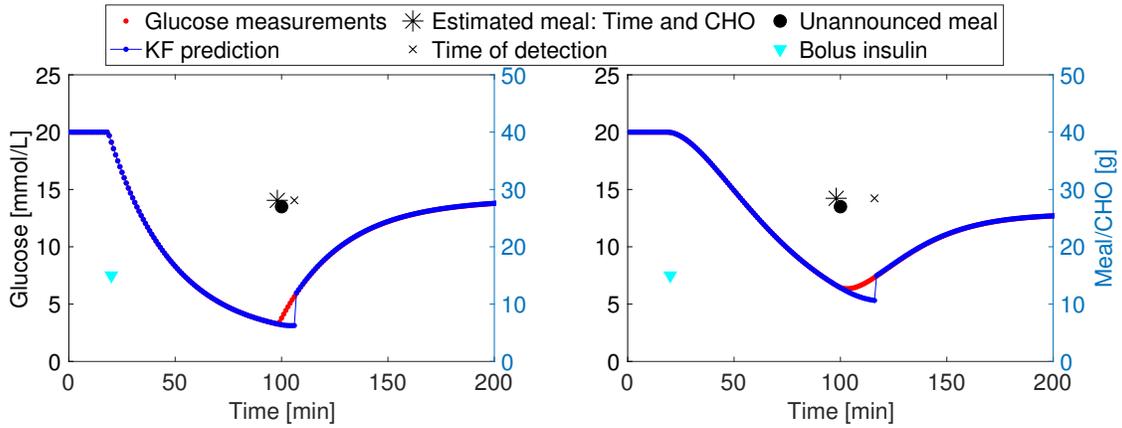


Fig. 2. The meal estimation running on ideal data $P1_{sim}$. *Left*: Model A data estimated with model A meal estimator. *Right*: Model B data estimated with model B meal estimator. The black dot signifies the true time and value of the input, the black star signifies what the estimator estimated it to be (time and value), and the black x is when the detection was made.

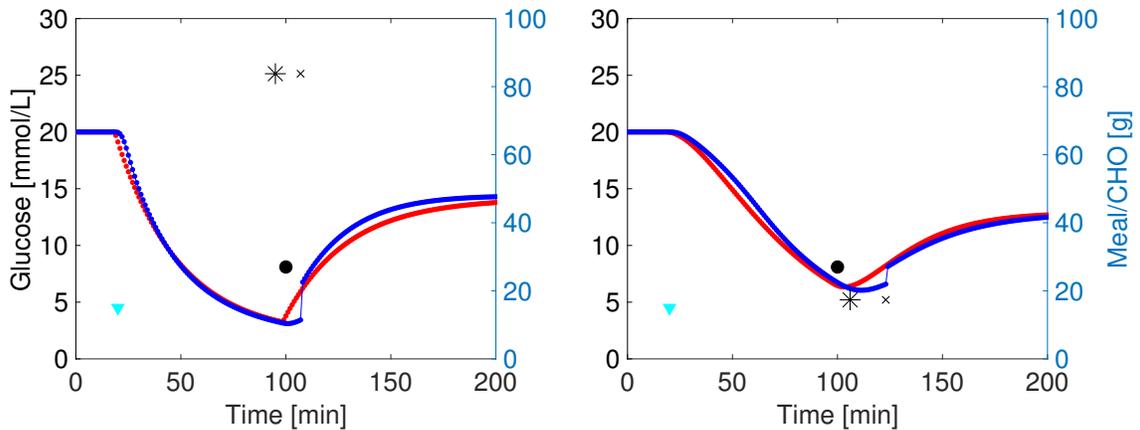


Fig. 3. Meal estimation result when running on mismatched models for data and estimator, using the scenario described for $P1_{sim}$ in Sec. III-A. *Left*: Simulated data from Model A estimated using model B. *Right*: Simulated data from model B estimated using model A. Legend as in Fig. 2. Note how the model mismatch leads to error in the CHO amount estimation.

meal to be above 80 g. This indicates that model adaptation to the data is likely needed. We see that the time of the detected meal is still approximately correct.

D. Tests on real CGM data

The estimator was modified to use 5 min sampling interval in order to process data from the Ohio study [30]. The performance of the meal detector using model B on a selected real data set is shown in Fig. 4. We see that the meals are detected close to that which was announced by the user. The second meal is detected as two separate meals. This data set illustrates some of the difficulty in using these data to determine the detection time; the meals input by the user seem to not have been given at the time of eating the meal, but rather some time after, since the glucose curve has been going up for a while when the meal announcement occurs, at least for the two last meals. If we use the user-logged meal time as the true meal times, we get detection times of 55, 45 and 15 minutes for the three meals. If we instead define the true meal time as the time of the last sample that did not show a glucose increase, we got detection times of 40,

65 and 45 minutes for the data set shown in Fig. 4. These figures are comparable to other methods that use CGM data to detect meals mentioned in Sec. I.

IV. DISCUSSION

The method proposed in this paper has potential as a meal detector, as it has the ability to estimate both the time and CHO value of meal. The method provides a straight-forward way to include known inputs like basal and bolus insulin and announced meals, thus enabling meal detection based on all the available data.

The method is applicable for real-time estimation, and provides a way to incorporate the estimated meal directly into the estimate at the time of detection without having to backtrack to the estimated input time of the meal, due to the state update described in Eqs. 20 – 22.

The meal value estimate is often seen to be off compared to the true value in the limited testing we performed. This is due to model mismatch, and further research and development is needed to see if this can be improved. More rigorous testing against properly annotated real CGM data is needed

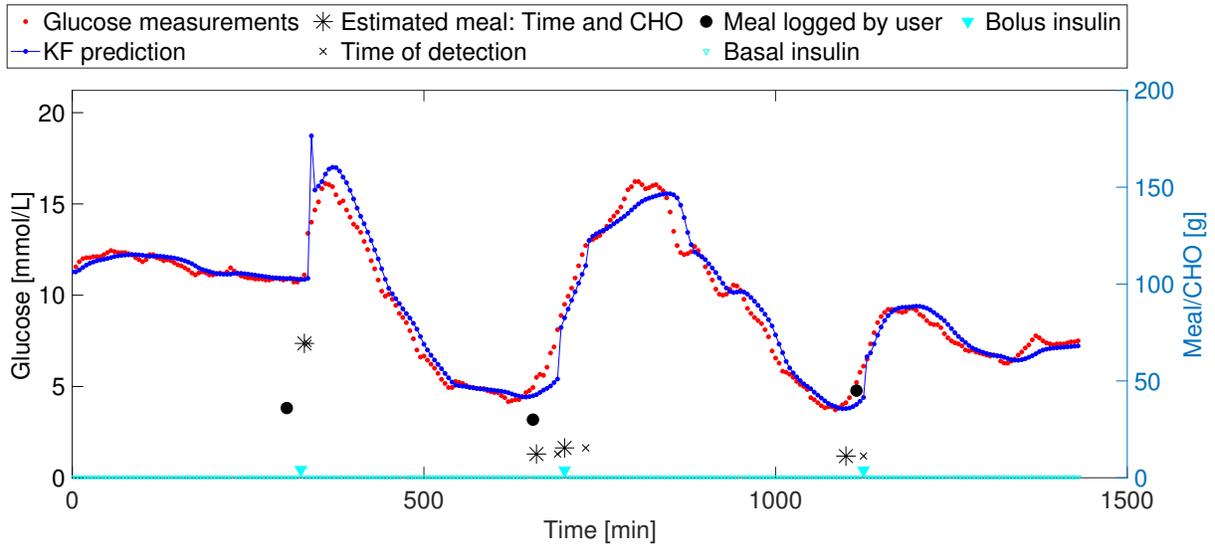


Fig. 4. Performance of the estimation algorithm using model B on data set Ohio P559 day 17. The meals logged by the user (black dots) are treated as unannounced meals, i.e. not input to the algorithm

to conclude on the applicability of this method in meal detection in real-time, real-life data. The need for accurate individualization of the method must also be investigated further.

Further research is needed to compare the newly proposed method with the other approaches for meal detection mentioned in Sec. I, to see how they perform against each other, both in an artificial pancreas setting and in an offline, data cleaning setting, and to further investigate the performance of the method on more real CGM data sets than the limited selection investigated here. Testing how the algorithm reacts to more sources of uncertainty, such as CGM errors (e.g. pressure induced sensor attenuation) or blocked insulin infusion sets will also need to be investigated.

A. Suggestions for further improvement

There are likely several more cases of glucose-specific logic that could be added to the estimator to improve performance and make it more robust. We suggest some possibilities for expansion and improvement of the method in the following.

1) Tuning parameters based on large sets of real data:

The parameters that are tunable in this method were listed in Sec. II-C.1 and III-D. The parameters have been found from a limited set of simulated and real data. The tuning of the parameters could and should be based on larger sets of data, and whether or not individualized parameters are needed should be investigated.

2) *Handling of 5 minute sampling interval:* A run of the estimator using 5 min data found that the input estimation still works at this sampling interval, detecting meals and estimating the meal intake time close to the actual time, however the estimation of the CHO content of the meal deteriorated. To keep the estimator step-size at 1 min or lower we could use a smoother in a fixed-lag mode when each 5-min sample arrives to create the missing 1-min

samples between the previous and current 5-min sample, then process all the smoothed samples in the input estimator at this time. While this does nothing to improve the real-time meal detection time, it may improve the CHO estimate. It also has the advantage of increasing the resolution of the estimate of the time of the detected meal to single minutes instead of 5 minutes, something that could perhaps be relevant when computing a compensatory insulin dosage. A suitable smoother for such an approach could be a Rauch-Tung-Striebel smoother operating in fixed lag mode [27].

3) Integration with other systems for meal detection:

An interesting extension of the method would be to adjust the process noise of the estimator based on inputs from other systems. For instance the audio based meal detector described in [7] could be combined with the estimator proposed in this paper. When the abdominal audio detector does not indicate a meal, the CHP estimator can use a higher Q so that it allows more meal-unrelated fluctuations, avoiding false detections. When the abdominal audio detector does indicate a meal may be present the Q matrix of the estimator is made smaller, increasing the probability of detection. This could improve the detection times while keeping the added benefit of meal quantification.

4) *Extension to nonlinear models:* The method we describe would work well in real time due to the low computational effort required. The restriction to linear models is a drawback, as many models of glucose-insulin dynamics are non-linear. While it is possible to use the Extended Kalman filter or the Unscented Kalman filter to handle non-linear transition and/or measurement functions in the filter part of the estimator, the CHP estimator relies on the linearity of the model in order to arrive at the closed form in Eq. 14, enabling least squares estimation. So, while extension of this concept to nonlinear systems is feasible, it will require more book-keeping of linearized transition matrices $A_{d,k}$, and

the optimization problem turns into a search and probably longer computation times. That being said, the time between samples of 5 minutes commonly found in CGM systems allows for a lot of computation between each sample even by embedded systems, so this is likely not a restriction that needs to be considered.

V. CONCLUSION

We have presented a method for meal detection and meal estimation based on Kalman filtering and the Chan-Hu-Plant method of input estimation, that provides an estimate of the consumed amount of CHO and the time of the meal. The method has been tested on synthetic and real data sets. The method has potential as a meal detector and estimator. It needs further testing against real CGM data that has been annotated with meal information, and comparison to other methods for meal detection. The method has potential use in glucose data cleaning settings, for imputing missing meal information. It could possibly also play a role in real-time artificial pancreas settings, although the latency of the detection is an issue in this usage scenario.

REFERENCES

- [1] S. S. Kanderian, S. A. Weinzimer, and G. M. Steil, "The identifiable virtual patient model: comparison of simulation and clinical closed-loop study results," *J Diabetes Sci Technol*, vol. 6, no. 2, pp. 371–379, Mar 2012.
- [2] R. Hovorka, V. Canonico, L. J. Chassin, U. Haueter, M. Massi-Benedetti, M. Orsini Federici, T. R. Pieber, H. C. Schaller, L. Schaupp, T. Vering, and M. E. Wilinska, "Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes," *Physiol Meas*, vol. 25, no. 4, pp. 905–920, Aug 2004.
- [3] N. Magdelaine, L. Chaillous, I. Guilhem, J. Y. Poirier, M. Krempf, C. H. Moog, and E. L. Carpentier, "A long-term model of the glucose-insulin dynamics of type 1 diabetes," *IEEE Transactions on Biomedical Engineering*, vol. 62, no. 6, pp. 1546–1552, June 2015.
- [4] C. Dalla Man, R. A. Rizza, and C. Cobelli, "Meal simulation model of the glucose-insulin system," *IEEE Trans Biomed Eng*, vol. 54, no. 10, pp. 1740–1749, Oct 2007.
- [5] The Epsilon Group. T1DMS type 1 diabetes metabolic simulator. <https://tegvirginia.com/software/t1dms/>.
- [6] R. Zhang and O. Amft, "Monitoring chewing and eating in free-living using smart eyeglasses," *IEEE journal of biomedical and health informatics*, vol. 22, no. 1, pp. 23–32, 2018.
- [7] K. Kölle, A. L. Fougner, R. Ellingsen, S. M. Carlsen, and Ø. Stavadahl, "Feasibility of early meal detection based on abdominal sound," *IEEE Journal of Translational Engineering in Health and Medicine. In press*, 2019.
- [8] K. A. Al Mamun and N. McFarlane, "Integrated real time bowel sound detector for artificial pancreas systems," *Sensing and bio-sensing research*, vol. 7, pp. 84–89, 2016.
- [9] D. Rhyner, H. Loher, J. Dehais, M. Anthimopoulos, S. Shevchik, R. H. Botwey, D. Duke, C. Stettler, P. Diem, and S. Mougiakakou, "Carbohydrate estimation by a mobile phone-based system versus self-estimations of individuals with type 1 diabetes mellitus: a comparative study," *Journal of medical Internet research*, vol. 18, no. 5, 2016.
- [10] Z. Mahmoudi, F. Cameron, N. K. Poulsen, H. Madsen, B. W. Bequette, and J. B. Jørgensen, "Sensor-based detection and estimation of meal carbohydrates for people with diabetes," *Biomedical Signal Processing and Control*, vol. 48, pp. 12–25, 2019.
- [11] K. Kölle, A. L. Fougner, and Ø. Stavadahl, "Meal detection based on non-individualized moving horizon estimation and classification," in *IEEE Conference on Control Technology and Applications (CCTA) 2017*. IEEE, 2017, pp. 529–535.
- [12] K. Kollé, T. Biester, S. Christiansen, A. L. Fougner, and O. Stavadahl, "Pattern recognition reveals characteristic postprandial glucose changes: Non-individualized meal detection in diabetes mellitus type 1," *IEEE Journal of Biomedical and Health Informatics*, 2019.
- [13] F. Cameron, G. Niemeyer, and B. A. Buckingham, "Probabilistic evolving meal detection and estimation of meal total glucose appearance," *Journal of Diabetes Science and Technology*, vol. 3, no. 5, pp. 1022–1030, 2009.
- [14] F. Cameron, G. Niemeyer, D. M. Wilson, B. W. Bequette, K. S. Benassi, P. Clinton, and B. A. Buckingham, "Inpatient trial of an artificial pancreas based on multiple model probabilistic predictive control with repeated large unannounced meals," *Diabetes technology & therapeutics*, vol. 16, no. 11, pp. 728–734, 2014.
- [15] E. Dassau, B. W. Bequette, B. A. Buckingham, and F. J. Doyle, "Detection of a meal using continuous glucose monitoring (CGM): implications for an artificial β -cell," *Diabetes care*, 2008.
- [16] 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.
- [17] R. A. Harvey, E. Dassau, H. Zisser, D. E. Seborg, and F. J. Doyle III, "Design of the glucose rate increase detector: a meal detection module for the health monitoring system," *Journal of diabetes science and technology*, vol. 8, no. 2, pp. 307–320, 2014.
- [18] J. Xie and Q. Wang, "A variable state dimension approach to meal detection and meal size estimation: in silico evaluation through basal-bolus insulin therapy for type 1 diabetes," *IEEE Transactions on Biomedical Engineering*, vol. 64, no. 6, pp. 1249–1260, 2016.
- [19] K. Turksyoy, S. Samadi, J. Feng, E. Littlejohn, L. Quinn, and A. Cinar, "Meal detection in patients with type 1 diabetes: a new module for the multivariable adaptive artificial pancreas control system," *IEEE journal of biomedical and health informatics*, vol. 20, no. 1, pp. 47–54, 2015.
- [20] C. Ramkissoon, P. Herrero, J. Bondia, and J. Vehi, "Unannounced meals in the artificial pancreas: detection using continuous glucose monitoring," *Sensors*, vol. 18, no. 3, p. 884, 2018.
- [21] K. Turksyoy, E. Littlejohn, and A. Cinar, "Multimodule, multivariable artificial pancreas for patients with type 1 diabetes: regulating glucose concentration under challenging conditions," *IEEE Control Systems Magazine*, vol. 38, no. 1, pp. 105–124, 2018.
- [22] L. M. Huyett, E. Dassau, H. C. Zisser, and F. J. Doyle III, "Glucose sensor dynamics and the artificial pancreas: the impact of lag on sensor measurement and controller performance," *IEEE Control Systems Magazine*, vol. 38, no. 1, pp. 30–46, 2018.
- [23] Ø. Stavadahl, A. L. Fougner, K. Kölle, S. C. Christiansen, R. Ellingsen, and S. M. Carlsen, "The artificial pancreas: a dynamic challenge," *IFAC-PapersOnLine*, vol. 49, no. 7, pp. 765–772, 2016.
- [24] Y. Chan, A. Hu, and J. Plant, "A Kalman filter based tracking scheme with input estimation," *IEEE transactions on Aerospace and Electronic Systems*, no. 2, pp. 237–244, 1979.
- [25] P. Bogler, "Tracking a maneuvering target using input estimation," *IEEE transactions on Aerospace and Electronic Systems*, no. 3, pp. 298–310, 1987.
- [26] R. A. DeCarlo, *Linear systems: A state variable approach with numerical implementation*. Prentice-Hall, Inc., 1989.
- [27] F. Gustafsson, *Statistical Sensor Fusion*. Studentlitteratur AB, Lund, Sweden, 2010.
- [28] D. Magill, "Optimal adaptive estimation of sampled stochastic processes," *IEEE Transactions on Automatic Control*, vol. 10, no. 4, pp. 434–439, 1965.
- [29] N. Magdelaine, L. Chaillous, I. Guilhem, J.-Y. Poirier, M. Krempf, F.-G. Anne-Laure, C. H. Moog, and E. Le Carpentier, "A relevant glucose-insulin model: Validation using clinical data," in *ATTD 2016; 9th International Conference on Advanced Technologies & Treatments of Diabetes, Feb 2016, Milan, Italy*, 2016.
- [30] C. Marling and R. Bunesco, "The OhioT1DM dataset for blood glucose level prediction," in *The 3rd International Workshop on Knowledge Discovery in Healthcare Data, Stockholm, Sweden*, 2018.