Towards a Model-Based Meal Detector for Type I Diabetics

Sanjian Chen  
*University of Pennsylvania*, sanjian@seas.upenn.edu

James Weimer  
*University of Pennsylvania*, weimerj@seas.upenn.edu

Michael R. Rickels  
*University of Pennsylvania*, rickels@mail.med.upenn.edu

Amy Peleckis  
*University of Pennsylvania*, amy.peleckis@uphs.upenn.edu

Insup Lee  
*University of Pennsylvania*, lee@cis.upenn.edu

Follow this and additional works at: [http://repository.upenn.edu/cis_papers](http://repository.upenn.edu/cis_papers)

Part of the [Computer Engineering Commons](http://repository.upenn.edu/cis_papers), and the [Computer Sciences Commons](http://repository.upenn.edu/cis_papers)

**Recommended Citation**


This paper is posted at ScholarlyCommons. [http://repository.upenn.edu/cis_papers/782](http://repository.upenn.edu/cis_papers/782)  
For more information, please contact libraryrepository@pobox.upenn.edu.
Towards a Model-Based Meal Detector for Type I Diabetics

Abstract
Blood glucose management systems are an important class of Medical Cyber-Physical Systems that provide vital everyday decision support service to diabetics. An artificial pancreas, which integrates a continuous glucose monitor, a wearable insulin pump, and control algorithms running on embedded computing devices, can significantly improve the quality of life for millions of Type 1 diabetics. A primary problem in the development of an artificial pancreas is the accurate detection and estimation of meal carbohydrates, which cause significant glucose system disturbances. Meal carbohydrate detection is challenging since post-meal glucose responses greatly depend on patient-specific physiology and meal composition.

In this paper, we develop a novel meal-time detector that leverages a linearized physiological model to realize a (nearly) constant false alarm rate (CFAR) performance despite unknown model parameters and uncertain meal inputs. Insilico evaluations using 10,000 virtual subjects on an FDA-accepted maximal physiological model illustrate that the proposed CFAR meal detector significantly outperforms a current state-of-the-art meal detector that utilizes a voting scheme based on rate-of-change (RoC) measures. The proposed detector achieves 99.6% correct detection rate while averaging one false alarm every 24 days (a 1.4% false alarm rate), which represents an 84% reduction in false alarms and a 95% reduction in missed alarms when compared to the RoC approach.

Keywords
Time series analysis, Medical Information Systems

Disciplines
Computer Engineering | Computer Sciences

Comments

This conference paper is available at ScholarlyCommons: http://repository.upenn.edu/cis_papers/782
ABSTRACT
Blood glucose management systems are an important class of Medical Cyber-Physical Systems that provide vital everyday decision support service to diabetics. An artificial pancreas, which integrates a continuous glucose monitor, a wearable insulin pump, and control algorithms running on embedded computing devices, can significantly improve the quality of life for millions of Type 1 diabetics. A primary problem in the development of an artificial pancreas is the accurate detection and estimation of meal carbohydrates, which cause significant glucose system disturbances. Meal carbohydrate detection is challenging since post-meal glucose responses greatly depend on patient-specific physiology and meal composition.

In this paper, we develop a novel meal-time detector that leverages a linearized physiological model to realize a (nearly) constant false alarm rate (CFAR) performance despite unknown model parameters and uncertain meal inputs. In silico evaluations using 10,000 virtual subjects on an FDA-accepted maximal physiological model illustrate that the proposed CFAR meal detector significantly outperforms a current state-of-the-art meal detector that utilizes a voting scheme based on rate-of-change (RoC) measures. The proposed detector achieves 99.6% correct detection rate while averaging one false alarm every 24 days (a 1.2% false alarm rate), which represents an 84% reduction in false alarms and a 95% reduction in missed alarms when compared to the RoC approach.  

Categories and Subject Descriptors
G.3 [Probability and Statistics]: Time series analysis; J.3 [Life and Medical Sciences]: Medical Information Systems

1. INTRODUCTION
Approximately 29 million people in the U.S. have diabetes, among which about 5% have Type 1 diabetes [14], an auto-immune disease that destroys a person’s pancreas’ ability to release insulin. Type 1 diabetics depend on everyday insulin infusion or injection to maintain their glucose level within the acceptable range where too much insulin can cause life-threatening hypoglycemia (extremely low glucose level) and too little insulin can cause nerve-damaging hyperglycemia (high glucose level) [2]. Unfortunately, meal carbohydrates are a major disturbance to one’s blood glucose level, and therefore every Type 1 diabetic faces a lifelong control challenge: he/she has to carefully titrate insulin doses for every meal so that post-meal hyperglycemia is effectively controlled without risking hypoglycemia.

In recent years, Continuous Glucose Monitoring (CGM) technology has become more popular, which drives a whole class of Medical Cyber-Physical System (MCPS), most notably the artificial pancreas (AP), that aims to facilitate glucose management for Type 1 diabetics. At the AP system’s core are a CGM sensor, a wearable insulin pump for insulin infusion and boluses, and algorithms that control insulin titration [8]. Reliably predicting meals is difficult in real-life situations, thus all AP systems depend on certain kinds of meal declaration/detection mechanisms. Meal detection is a safety critical problem, where an incorrectly identified meal may trigger the system to either deliver too much insulin unnecessarily or deliver too little insulin, both of which have harmful (if not deadly) consequences.

Currently, most Type 1 diabetics who use CGM sensors and wearable insulin pumps manually input the time and estimated carb count of each meal into the device, which then calculates a suggested insulin dose. Unfortunately, self-reported meal information is inherently unreliable [12]. Thus, more dependable meal detection methods are necessary to ensure patient safety. This paper proposes a novel meal detection method that leverages a linear model of glu-
cose and insulin responses that is inspired by a first-principle minimal physiological model [3].

The proposed detector exhibits consistent detection performance in in-silico evaluations using an FDA-accepted Type 1 diabetes physiological model across a wide range of virtual subject parameter settings. Specifically, it achieved a 99.6% detection rate (one miss every 90 days) with a low false alarm rate of once every 24 days. This is a significant improvement compared to a state-of-the-art voting algorithm that detects meals based on glucose rate-of-change (RoC) [12]: in the same in-silico experiment, the voting algorithm detects 93.2% of meals (one miss every 5 days) while raising a false alarm every 4 days.

The remainder of this paper is organized as follows: Section 2 reviews related work; Section 3 gives the problem formulation; Section 4 presents two major classes of glucose-insulin physiological models; Section 5 introduces the Constant False Alarm Rate (CFAR) meal detector; Section 6 compares the performance of our detector with a state-of-the-art meal detector in an in-silico evaluation using an FDA-accepted physiological model.

2. RELATED WORK

Input detection via hypothesis testing is a mature research area where approaches can be roughly classified as model-driven [6, 36] or data-driven [4, 10, 23, 26, 27, 28]. When applied to MCPS problems, model-driven approaches (e.g., likelihood ratio testing) suffer due to modeling nonlinearities and unknowns. In contrast, data-driven MCPS approaches (e.g., machine learning classification) are considered promising [1, 20], but suffer from over-fitting when accurately annotated real-world training data is scarce.

Specific to the problem of meal detection and estimation, various algorithms have been proposed [12, 5, 19]. Most existing algorithms detect meals based on the RoC of glucose values, more precisely, first and/or second derivatives derived from CGM readings [12, 19]. The RoC metrics are estimated in real-time and are either directly compared to pre-tuned thresholds to make a decision, or used to estimate meal impulses. One of the most prominent RoC meal detectors is described in [12]. It utilizes a voting algorithm to detect meal disturbances using combinations of glucose derivatives and Kalman filtering. A pilot evaluation on real patients using the RoC detector suggests an average delay of 30 minutes between the onset of a meal and detection [12].

Although RoC-based meal detectors are easy to understand and implement, they have some fundamental limitations considering everyday diabetes management. The post-meal glucose response of a Type 1 diabetic depends on many factors and can greatly vary across different patients. Most insulin-dependent Type 1 diabetics are instructed to take insulin boluses around meal times, where both the timing and the amount of meal boluses can change postprandial plasma glucose responses [9]. Moreover, the exact breakdown of protein, fat, and carbohydrates in a meal also impacts glucose responses [37]. Additionally, the complex interaction between glucose and insulin metabolism systems can trigger RoC-based meal detection which was actually a drop in “insulin on board” [13]. Therefore, threshold based RoC meal detectors are likely to generate numerous false alarms and can miss true meal events when the postprandial glucose rise is not as prominent.

Other approaches to meal detection use physiological models to generate a population of possible post-meal glucose shapes [5]. Based on a residual generated using the difference between a no-meal realization of the population models and the actual CGM measurements, near-perfect meal detection can be achieved using population-based approaches but requires significantly longer to make an accurate decision when compared to RoC-based approaches [12, 5, 19]. Moreover, this performance is dependent on an accurate estimate of patient specific parameters (e.g. insulin time action profile), which in real-world scenarios requires periodic controlled studies on each patient where the meal events are accurately logged.

An alternative approach to event detection that tends to minimize the detection time while maximizing performance in problems with scarce data and uncertain models is the CFAR detector [30, 32]. CFAR detectors leverage parameterized model structure to generate sufficient statistics that are maximally invariant to unknown parameters such that a threshold test achieves a (nearly) constant false alarm rate [30]. Although not previously applied to the diabetic meal detection problem, these detectors have been shown to work well in other CPS applications with structured dynamics and unknown parameters, specifically in detecting faults in networked systems [34, 35], building heating, ventilating and air conditioning (HVAC) systems [31], smart grids [33], and most recently in a MCPS application to detect critical pulmonary shunts in infants [16].

3. PROBLEM FORMULATION

All AP systems need accurate estimates of the meal carbohydrate disturbances. To estimate the meal carbohydrate disturbances requires an accurate (and timely) estimate of when meals occur. Current approaches to meal detection lack detection accuracy (RoC-based approaches) or lack timely detection guarantees (population-based approaches). Thus, the following summarizes our problem statement:

3.1 Problem Statement

This paper addresses the meal detection problem, where given recent glucose level measurements and insulin inputs, design a run-time monitor that accurately and quickly detects the ingestion of carbohydrates.

4. PHYSIOLOGICAL MODELS

First-principle models of glucose physiology broadly fall into two categories: maximal models and minimal models [7]. Maximal models use fine-grain compartmental submodels to describe the dynamics of glucose and insulin. These models are mostly used for simulation purposes, since controller design for non-linear models with unknown parameters is difficult. On the other hand, the minimal models use only a few coarse-grain compartments to model the physiology, and they have a simple structure that is convenient for linearization and control design [15]. This section introduces an FDA-accepted maximal model, which will later be used in in-silico evaluations, and a minimal model, which inspires the process modeling in designing our CFAR detector.

4.1 A Maximal Model

A maximal model that describes the glucose-insulin responses with meals has been proposed [21, 22], based on which the UVa/Padova Type 1 Diabetes Mellitus Metabolic
Simulator (T1DMS) has been developed [18]. This model is an FDA-accepted substitute for animal testing in pre-clinical trials when evaluating certain control algorithms [11]. It consists of a set of continuous-time differential equations with 13 state variables and 32 physiological parameters. The model includes three sub-systems: the insulin sub-system, the meal glucose absorption, and the glucose kinetics. Due to space constraints, this section only sketches the model equations with brief explanations of the state variables. Extensive details of the model, including the modeling rationale and meanings of the variables & parameters, can be found in a series of publications [21, 22, 18, 11].

The insulin sub-system describes the transportation of insulin from the subcutaneous injection site to other compartments of the body such as the liver, plasma, and tissues. This sub-system has seven state variables which evolve according to the following equations [21, 22]:

\[
\begin{align*}
\dot{I}_p(t) &= -(m_2 + m_4)I_p(t) + m_1I_1(t) + k_{a1}S_1(t) + k_{a2}S_2(t) \\
\dot{X}(t) &= P_{2V}/V_I p(t) - P_{2V}X(t) - P_{2V} + I_0 \\
\dot{I}_1(t) &= k_{f1}/V_I p(t) - k_{i1}I_1(t) \\
\dot{I}_2(t) &= k_{i1}I_1(t) - k_{f2}I_2(t) \\
\dot{I}_3(t) &= m_2* I_p(t) - (m_1 + m_3)I_1(t) \\
\dot{S}_1(t) &= -(k_{a1} + k_3)S_1(t) + u(t) \\
\dot{S}_2(t) &= k_3S_1(t) - k_{a2}S_2(t).
\end{align*}
\]

In the above equations, \(I_p\) represents the mass of plasma insulin. \(X(t)\) is a remote insulin signal that also appears in the glucose kinetics. \(I_1\) and \(I_2\) represent a delayed insulin signal that governs the endogenous glucose production. \(I_3\) represents the liver insulin. \(S_1\) and \(S_2\) represent a two-compartment subcutaneous insulin process. \(u(t)\) is the subcutaneous insulin injection/infusion input (wearable insulin pumps for Type 1 diabetics inject insulin into the subcutaneous tissue). Table 2 lists the physiological entities that the parameters represent. Parameters listed as "rate parameters" in Table 2 are those simply govern the flux rate in/out of a compartment. Details about the parameters can be found in the maximal modeling literature [21, 22, 18].

The meal absorption sub-system models how meal carbohydrates pass the stomach, intestine, and finally becomes glucose appearing in the plasma [22]. The stomach is represented by two compartments: one for the solid phase and the other for the liquid phase. The dynamics are modeled by the following equations [22]:

\[
\begin{align*}
Q_{sto}(t) &= Q_{sto1}(t) + Q_{sto2}(t) \\
Q_{sto1}(t) &= -k_{gr}Q_{sto1}(t) + m(t) \\
Q_{sto2}(t) &= -k_{cmp}(Q_{sto}Q_{sto1}(t)) + k_{gr}Q_{sto1}(t) \\
Q_{int}(t) &= k_{abs}Q_{sto1}(t) + k_{cmp}(Q_{sto}Q_{sto2}(t) \\
k_{cmp}(Q_{sto}) &= k_{min} + k_{max} - k_{min}/2 \times \left( \frac{\tanh(\alpha *(Q_{sto} - b + D))}{\tanh(\beta *(Q_{sto} - d + D))} + 2 \right) \\
\alpha &= 5/(2*D*(1 - b)), \ \beta = 5/(2*D + d)
\end{align*}
\]

\(Q_{sto2}\) represents glucose in the solid phase and liquid phase, respectively. \(Q_{int}\) is the amount of glucose in the intestine. \(k_{cmp}(Q_{sto})\) is a non-linear function that represents the rate of carbohydrates emptying from the stomach. \(D\) is the total amount of ingested glucose in the last meal. \(m(t)\) is the input of meal carbohydrates. The physiological meanings of parameters can be found in Table 2.

The insulin and absorbed meal glucose interact through the glucose kinetics, which is modeled by three state variables. The equations are given as follows [22]:

\[
\begin{align*}
R_a(t) &= f * k_{abs} * Q_{int}/BW \\
G_p(t) &= -k_1 * G_p(t) + k_2 * G_1(t) + (0, p_1 - k_p2 * G_p - k_p3 * I_3(t)) \\
&- F_{isc} - (0, k_{a1} *(G_p(t) - k_{a2})) + R_a(t) \\
G_1(t) &= -(V_m0 + V_{max} * X(t)) * G_1(t) \quad (3c) \\
&+ k_1 * G_p(t) - k_2 * G_1(t) \\
G_m(t) &= -k_{a1} * G_1(t) + k_{a2}/V_g * G_2(t). \\
G_p \quad \text{represents the plasma glucose concentration.} \\
G_1 \quad \text{represents glucose in the rapidly equilibrating tissue.} \\
G_m \quad \text{represents the subcutaneous glucose.}
\end{align*}
\]

The maximal model of the T1DMS consists of all the 13 differential equations presented above. The insulin sub-system is a linear model. The meal sub-system contains a non-linear parameter \(k_{cmp}(Q_{sto})\). The glucose kinetics sub-system has several non-linear terms, such as the max operators and state product \(X(t)G_1(t)\). Most of the model parameters, as listed in Table 2, are not easily identifiable on patients. Because of the non-linearity and unknown parameters, it is very difficult to directly use the maximal model in control design. Instead, the model is used primarily for simulation purposes. The FDA accepted 300 virtual subjects, each of which is a realization of the entire parameter vector, that are sampled from a joint distribution. The parameter distribution was drawn from clinical experimental data obtained from individuals who underwent a triple tracer meal protocol [22].

**4.2 Minimal Models**

Minimal models represent another class of first-principle glucose-insulin models. The basic idea is to lump together compartments to describe the dominating dynamics of the glucose-related physiology in a minimal number of compartments. One of the most commonly accepted minimal models is described in [3], and referred to as the Bergman model. The Bergman model uses a single lumped compartment to model insulin and another lumped compartment to model glucose in plasma. Insulin governs the changes of glucose level either directly or through another remote compartment. Under this compartmentalization scheme, at most three state variables are needed describe the glucose-insulin physiology: plasma glucose, plasma insulin, and insulin in the remote compartment. Seven minimal models are proposed in [3], from the simple insulin-independent models (models No. 1 to 3), to the more elaborated forms (models No. 4 to 7). Not all the models use all of the three states.
The Bergman model No. 4, whose equations are given as follows, has a linear form and explicitly describes insulin-dependent glucose uptake,

\[ \dot{G}(t) = p_1 G(t) + p_2 \cdot I(t) + p_3 \]

where \( G(t) \) and \( I(t) \) represent plasma glucose and insulin, respectively, and \( p_1, p_2, \) and \( p_3 \) are unspecified model parameters.

While the Bergman model describes the plasma glucose-insulin dynamics, a second order, two-compartment meal model is presented in [15]:

\[ \dot{g}(t) = -\frac{1}{k_g} g(t) + \frac{A_g}{k_g} D_g(t) \]
\[ \dot{m}(t) = -\frac{1}{k_c} m(t) + \frac{1}{k_c} g(t), \]

where \( g(t) \) represents glucose in the first compartment and \( m(t) \) represents the plasma glucose appearance, which is an input to the Bergman model. \( D_g(t) \) is the meal carbohydrate input. \( A_g \) is the carbohydrate bioavailability and \( k_c \) is the time of maximum glucose rate of appearance.

Lastly, the insulin pathway from subcutaneous tissue to plasma can be modeled by the following second order process [24, 17],

\[ \dot{x}(t) = -k_e x(t) + u(t - \tau) \]
\[ \dot{I}(t) = -k_i I(t) + \frac{k_o}{V_d} x(t), \]

where \( x(t) \) and \( I(t) \) are insulin in the subcutaneous compartment and plasma, respectively, \( k_e \) and \( k_i \) are rate parameters, \( V_d \) is the insulin volume, and \( u(t - \tau) \) represents the insulin input with a time delay \( \tau \).

Combining equations 4, 5, and 6 results in a fifth-order linear model that describes the glucose-insulin kinetics given meal carbohydrate inputs and subcutaneous insulin inputs. This linear model will be used in the CFAR detector design.

5. **CFAR MEAL DETECTOR DESIGN**

In this section, we introduce the CFAR meal detector with a bin-counting heuristic which counts the number of decisions generated using dual parameter-invariant statistics evaluated at each time step. The remainder of this section details the dual parameter-invariant statistics and the bin-counting heuristic, respectively.

5.1 **Parameter Invariant Design**

The design of CFAR detectors utilizing parameter invariant statistics originates from statistical signal processing [29]. Our previous work [32] proposes a general framework of parameter invariant design in medical systems. In this paper, we apply the general methodology in the context of the meal detection problem.

Our primary goal lies in designing a detector (or monitor) that has consistent detection performance on a large population of Type 1 diabetics given a wide range of meal and insulin inputs. Because many of the patient-specific physiological parameters (e.g., those in the maximal models) cannot be identified, the detector is designed to be invariant to model parameters. In addition, the amount of meal carbohydrates is manually reported by users in current Type 1 diabetes management systems, which can be unreliable for a variety of reasons, e.g., the patient may forget to report or discount the meal portion. Therefore, the detector is also designed to be invariant to the exact magnitude of meal inputs. Insulin bolus times and doses are used by our detector since those are pump logged information.

We start with the linear minimal model in equations 4, 5, and 6. Following standard control theory techniques [25], the state-space linear model can be transformed into a \( z \)-domain representation and then written in a discrete time matrix form \( y = F \cdot \theta + G \cdot v + \sigma_n \), in which \( y \) represents the outputs, \( F \) represents the process model, \( \theta \) represents model parameters, \( G \) represents input response, \( v \) represents inputs, and \( \sigma_n \) are zero-mean Gaussian noises.

The CFAR detector runs in a sliding window fashion. It has a few critical time windows which are illustrated in Figure 1. At each time step, the detector is given a vector of \( T \) past measurements \( y \in \mathbb{R}^T \) and insulin bolus inputs within the time window \( w \). To be invariant to the unknown model parameters, the CFAR detector first eliminates the effects of parameters \( \theta \) by projecting \( y \) onto the null space of \( F \) so that the term \( F \cdot \theta \) becomes zero.

The core of the detector is a bi-directional meal hypothesis test. The detector first hypothesizes that a meal happened \( w \) steps back from the current time (meals are treated as impulses in the detector model), \( w \cdot d \) is a detector parameter that stays constant at run-time once it is chosen. The null hypothesis \( H_0 \) states that a meal indeed happened in a time window around next to the hypothesized meal time (the \( H_0 \) window in Figure 1). The event hypothesis \( H_1 \) states that a meal actually happened in an even earlier time window (the \( H_1 \) window in Figure 1). The sizes of \( H_0 \) and \( H_1 \) windows are design parameters to be chosen. The input response matrices \( G_0 \) and \( G_1 \) represent the hypothesized meal time windows of \( H_0 \) and \( H_1 \), respectively. When testing \( H_0 \) in the direction of \( H_1 \), the detector eliminates the effect of \( H_0 \) input by projecting \( G_1 \) onto the null space of \( G_0 \). It then calculates a statistic \( t_0(y) \) as a ratio, where the numerator represents the remaining energy in the glucose measurements explained by signal under \( H_1 \) and the denominator represents the energy that is not explained by \( H_1 \). Then the detector tests \( H_1 \) in the direction of \( H_0 \) and calculates \( t_1(y) \) in exactly the same way but in the opposite direction.

The statistic \( t_0(y) \) assumes \( H_0 \) is true and tests \( H_1 \). The statistic \( t_1(y) \) assumes \( H_1 \) is true and tests \( H_0 \). To guarantee a minimum level of performance, the detector rejects \( H_0 \) when \( t_0(y) > \eta_0 \), where \( \eta_0 \) is related to the probability of false alarms. And similarly, the detector rejects \( H_1 \) when \( t_1(y) > \eta_1 \), where \( \eta_1 \) is related to the probability of missed detection.

The detector makes a meal detection decision based on the bi-directional hypothesis tests. When \( H_1 \) is rejected and \( H_0 \) is not, it claims a meal happened in the \( H_0 \) zone. When \( H_0 \) is rejected and \( H_1 \) is not, it claims a meal happened in the \( H_1 \) zone. When both are not rejected, it means there is not enough power in the signal to make a decision. When both are rejected, it means there are residual energy in the measurements that are explained by both hypotheses’ signals.

Figure 1 demonstrates how the CFAR detector works on simulated scenarios generated by the FDA-accepted maximal model. The CGM measurements are sampled at one minute time steps. The true meal happens around time 2160.
Figure 1: A meal detection example of the CFAR detector.

Table 1: Credit addition rules for the bin-counting process.

<table>
<thead>
<tr>
<th>$t_0(y)$</th>
<th>$t_1(y)$</th>
<th>Credit Adding Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>$&gt; \eta_0$</td>
<td>$&gt; \eta_1$</td>
<td>Both Rejecting (+1 to each)</td>
</tr>
<tr>
<td>$\leq \eta_0$</td>
<td>$\leq \eta_1$</td>
<td>Accept $H_0$ (+2 to $H_0$)</td>
</tr>
<tr>
<td>$\leq \eta_0$</td>
<td>$&gt; \eta_1$</td>
<td>Accept $H_1$ (+2 to $H_1$)</td>
</tr>
<tr>
<td>$&gt; \eta_0$</td>
<td>$\leq \eta_1$</td>
<td>Insufficient Power (assign no credits)</td>
</tr>
</tbody>
</table>

5.2 Bin-Counting Meal Alarms

In the previous section, we develop a detector that generates a decision at each time step. At run-time, the detector runs in a sliding window fashion, with the relative positions of the windows fixed once the detector parameters are chosen (see Figure 1). As the detector approaches a true meal event (the ground truth meal times are unknown to the detector), it will first pass through the $H_0$ window, and then the $H_1$ window. Therefore, one meal event will accumulate a few $H_0$ claims and then some $H_1$ claims as the detector windows slide through. We leverage this phenomenon and develop a bin-counting heuristic to generate more robust meal detection alarms.

The bin-counting process creates one bin per each sample time and registers a number of “meal hits” under it. Table 1 presents the credit adding rules. At every step, if the detector claims $H_0$, then two credits are added to every bin in the $H_0$ window; similarly, if the detector claims $H_1$, two credits are added to every bin in the $H_1$ window. If both $H_0$ and $H_1$ are rejected, then one credit is added to every bin in both $H_0$ and $H_1$ windows. If both $H_0$ and $H_1$ are accepted, it means neither $H_0$’s nor $H_1$’s statistic is greater than the threshold, and thus no credit is assigned. The credit adding rules are also highlighted in Figure 1.

A peak in the bin-count signal means that the detector makes a number of decisions at different time steps that all point to the same meal time, indicating a strong positive correlation.
hit of a meal. On the other hand, if a bin only receives one or two counts (a typical width of $H_0$ and $H_1$ windows is 5 sample steps), it means the detector did not generate consistent decisions as the bin passed through the $H_0$ and $H_1$ windows, indicating a possibility of false positive.

Two design parameters, a threshold $t$ and a minimum width $w$, are used to define positive peaks in the bin-count curve: a peak is characterized by at least $w$ consecutive bin counts that are above $t$. At each time step, the detector examines the bin-count signal (up until the current step) and raise a meal alarm if a new peak emerges. In Figure 1, the bin-count curve crosses the threshold of $t = 12$ around time 2170 (the widths of $w_d$, $H_0$ and $H_1$ windows are all set to 5 sample periods) and stays above it for at least $w = 3$ steps, therefore, a meal alarm is declared by the CFAR detector around the time 2182. The bin-count peaks emerge with a time lag from the detector’s perspective because of how the CFAR detector works: it adds bin counts to past time windows, $H_0$ and $H_1$, according to current hypothesis test results. The example also demonstrates that the alarming sensitivity of our detector can be tuned by varying $t$ and $w$: the detector may raise the meal alarm sooner with smaller $t$ and $w$, at the cost of more false alarms. We also note that there is a few steps delay between the actual meal time and when the bin-count curves starts rising. This delay phenomenon is consistently observed in the simulation studies and we believe this is related to the fact that there is a delay from the onset of eating to CGM starts changing, which makes sense physiologically: in the maximal model, meal carbohydrates have to pass several digestion compartments before affecting the plasma glucose. In our future work we will incorporate the meal delay into the detector design in order to accurately identify the onset of a meal.

6. CASE STUDY

We evaluate the CFAR meal detector on an in-silico diabetic database that is generated by the FDA-accepted maximal model described in Section 4.1. We compare the performance of our detector with a state-of-the-art RoC-based voting meal detection algorithm published in [12].

6.1 In-Silico Diabetic Database

The academic version of the FDA-accepted Type 1 Diabetic Simulator [18] (T1DMS) comes with ten virtual subjects that are drawn from the same parameter distribution with the FDA-accepted virtual subject population. Each virtual subject is realization of the 32 parameters of the physiological model (see Table 2 for the complete list of parameters). To thoroughly test the performance of our detector across a wide range of possible patient parameters, we randomly sampled 10,000 virtual subjects from the parameter space spanned by the ten T1DMS virtual subjects. The 10,000 virtual subjects are sampled from an independent uniform distribution in the hypercubes spanned by the T1DMS subjects. Because real patients’ parameters are actually from a joint distribution based on clinical study data [22], we over-sample the parameter space by assuming no knowledge about the joint distribution, which is good for testing the robustness of the detectors.

We run simulations of the FDA-accepted maximal model (with the virtual subjects’ parameters) by feeding meals and insulin inputs that mimic the real-life scenario of a Type 1 patient. Three meals with random amounts of carbohydrates are fed to a virtual subject every simulation day. Whenever a meal is given, a meal bolus that is calculated based on a randomized meal insulin ratio is also fed into the simulator. In addition, we set a checkpoint once every hour except overnight time (22PM - 6AM) to mimic the scenario that a patient may check his/her own blood sugar and take correction insulin boluses when the glucose reading is too high. The dose of each correction bolus is calculated based on a randomly drawn insulin sensitivity value. The simulation settings, including the amount of carbohydrates in each meal, meal insulin ratio, and insulin sensitivity are independently and randomly sampled from ranges that are consistent with de-identified clinical data on over 60 Type 1 diabetics’ CGM readings and pump logs, which are collected in a collaboration study with the Penn Diabetes Center.

The complete in-silico database consists of simulated glucose data sampled at a one minute period and the corresponding insulin & meal inputs. The database currently includes two sets of experimental data: 3-day simulation data of 10,000 virtual subjects (data set No. 1) and 15-day simulation data of 1,000 virtual subjects (data set No. 2). The first set is to test a detector’s performance across a large number of virtual subjects and the second set can be used to test a detector’s performance for a prolonged per-subject simulation period.

6.2 In-Silico Experimental Results

We test the CFAR meal detector and the RoC-based three out of four voting meal detector [12] on the in-silico diabetic databases described before. The two detectors are run on the exactly same set of virtual subjects’ simulated data and the CGM measurements are sampled at a one-minute period. A receiver operating characteristic (ROC) curve is generated for each detector by changing the detection sensitivity. For the CFAR meal detector, the bin-counting threshold $t$ is varied from 16 to 11 to change its sensitivity. The rest of the CFAR detector parameters are chosen as follows: the training data window width $w_d$ is 300 samples (or 5 hours), the lengths of the $w$, $H_0$ and $H_1$ windows are all 5 samples, and the bin-counting $w$ is set to be 3. For the three out of four RoC voting meal detector, the RoC threshold is varied.

Figure 2 shows the ROC curves of two detectors when evaluated on the first experiment data set of 10,000 virtual subjects (the ROC curves of two detectors running on the second data set of 1,000 virtual subjects are highly consistent with the results on the 10,000 data set; the two are almost identical). The CFAR detector’s best operation point (circled out in Figure 2) is at the 99.6% detection rate and 1.4% false alarm rate. That is on average about one missed detection every 90 days and one false alarm every 24 days. To achieve the same detection rate, the RoC voting detector has a 39.6% false alarm rate, which is two false alarms every day, 48 times higher than our detector. To achieve the same false alarm rate, the RoC voting detector has a 78.5% detection rate, which is about one missed detection every 1.5 days, 60 times higher than our detector. At one of its best operation points (circled out in Figure 2), the RoC voting detector has a 93.2% detection rate and a 8.4% false alarm rate; that is on average about one miss every 5 days and one false alarm every 4 days, i.e., 18 times more missed detection and 6 times more false alarms than our detector. The CFAR detector significantly outperforms the RoC voting detector. In practice, a false alarm made by the meal detector may trigger an AP system to unnecessarily deliver insulin boluses, putting the patient at a greater risk of hypo-
glycemia; on the other hand, a missed detection may result in insufficient insulin to compensate post-meal glucose rises and leave the patient suffer from hyperglycemia.

Figure 3 shows the cumulative detection percentage over the time delay since the onset of a meal. The curves in Figure 3 are obtained when both detectors are running at their operating points marked in Figure 2. Initially, the CFAR detector’s detection time is about less than one minute (that is less than one sample time) late than the rate-of-change voting detector, an insignificant difference given that the glucose physiology changes slowly on a minute scale (most commercial CGM sensors transmit one reading every 5 minutes). After the crossing point at around 70%, the CFAR detector quickly reaches its 99.6% detection rate before the 30 minute delay, whereas the RoC voting detector reaches its maximum 93.2% detection rate around the 40 minute delay mark.

Figure 4 compares the number of per-subject missed meals of the two detectors. The results demonstrate the consistency of detection performance on different subjects, i.e., whether the detector performs particularly bad for certain subjects. In this test we used the data set No. 2 (1,000 subjects, 15 simulation days on each) since we want to examine the per-subject performance of each detector. Over 15 simulation days, each subject has 44 meals to be tested (3 meals a day; the very first meal is excluded as the simulator is still warming up). The CFAR detector has 137 total misses on 98 subjects, out of which 74 subjects (or 76%) have only one miss. The maximum number of per-subject misses is 5. The RoC voting detector has 2063 total misses on 298 subjects, out of which 79 subjects (or 26%) have one miss, and 21 subjects have over half of the meals missed, including 2 subjects on whom all meals are missed. The results clearly demonstrate the power of parameter invariant design: the detector exhibits a much more consistent performance over a large virtual population, comparing to the RoC voting detector.

The in-silico evaluation shows that the CFAR detector achieves a high detection rate with a low false alarm rate across a large virtual patient population and a wide range of meal/insulin simulation settings. It significantly outperforms the existing RoC based voting meal detector. The new detector achieves a near-perfect 99.6% detection rate across a wide range of physiological parameters and simulation settings, with a low average false alarm rate of once every 24 days.

This preliminary work is intended to motivate the study of providing guarantees regarding meal (disturbance) detection and ultimately the prediction of near future glucose values to inform critical actions that aim at preventing unsafe conditions such as hypoglycemia. Our future work includes extensive evaluation of the meal detector on clinical data and developing techniques to formally bound the performance of the detector.

7. CONCLUSION

In this paper, we develop a CFAR meal detector based on an augmented version of the Bergman minimal model. In-silico evaluation using the FDA-accepted maximal physiological model shows that the detector significantly outperforms the existing RoC based voting meal detector. The new detector achieves a near-perfect 99.6% detection rate across a wide range of physiological parameters and simulation settings, with a low average false alarm rate of once every 24 days.

This preliminary work is intended to motivate the study of providing guarantees regarding meal (disturbance) detection and ultimately the prediction of near future glucose values to inform critical actions that aim at preventing unsafe conditions such as hypoglycemia. Our future work includes extensive evaluation of the meal detector on clinical data and developing techniques to formally bound the performance of the detector.
8. REFERENCES


Table 2: Physical meanings, units, and a typical Type 1 diabetic value of the parameters in the T1DMS maximal model

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Physical Meaning</th>
<th>Units</th>
<th>Typical Type 1 value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m_1$</td>
<td>Rate parameter</td>
<td>min $^{-1}$</td>
<td>0.20</td>
</tr>
<tr>
<td>$m_2$</td>
<td>Rate parameter</td>
<td>min $^{-1}$</td>
<td>0.28</td>
</tr>
<tr>
<td>$m_3$</td>
<td>Rate parameter</td>
<td>min $^{-1}$</td>
<td>0.30</td>
</tr>
<tr>
<td>$m_4$</td>
<td>Rate parameter</td>
<td>min $^{-1}$</td>
<td>0.11</td>
</tr>
<tr>
<td>$k_1$</td>
<td>Rate parameter</td>
<td>min $^{-1}$</td>
<td>0.0084</td>
</tr>
<tr>
<td>$P_{bw}$</td>
<td>Rate parameter</td>
<td>min $^{-1}$</td>
<td>0.0034</td>
</tr>
<tr>
<td>$V_i$</td>
<td>Insulin volume</td>
<td>liter/kg</td>
<td>0.053</td>
</tr>
<tr>
<td>$I_0$</td>
<td>Basal insulin level</td>
<td>pmol/liter</td>
<td>108</td>
</tr>
<tr>
<td>$BW$</td>
<td>Body weight</td>
<td>kg</td>
<td>86</td>
</tr>
<tr>
<td>$k_1$</td>
<td>Rate parameter</td>
<td>min $^{-1}$</td>
<td>0.063</td>
</tr>
<tr>
<td>$k_2$</td>
<td>Rate parameter</td>
<td>min $^{-1}$</td>
<td>0.13</td>
</tr>
<tr>
<td>$k_{p1}$</td>
<td>Extrapolated EGP</td>
<td>mg/kg/min</td>
<td>4.6</td>
</tr>
<tr>
<td>$k_{p2}$</td>
<td>Liver glucose effectiveness</td>
<td>min $^{-1}$</td>
<td>0.0042</td>
</tr>
<tr>
<td>$k_{p3}$</td>
<td>Insulin action on liver</td>
<td>mg/kg/min per pmol/liter</td>
<td>0.0095</td>
</tr>
<tr>
<td>$V_{m0}$</td>
<td>Model parameter</td>
<td>mg/kg/min</td>
<td>4.9</td>
</tr>
<tr>
<td>$V_{xx}$</td>
<td>Model parameter</td>
<td>mg/kg/min per pmol/liter</td>
<td>0.054</td>
</tr>
<tr>
<td>$K_{m0}$</td>
<td>Model parameter</td>
<td>mg/kg</td>
<td>237</td>
</tr>
<tr>
<td>$V_g$</td>
<td>Glucose volume</td>
<td>dL/kg</td>
<td>1.8</td>
</tr>
<tr>
<td>$k_{s1}$</td>
<td>Rate parameter</td>
<td>min $^{-1}$</td>
<td>0.0034</td>
</tr>
<tr>
<td>$k_{s2}$</td>
<td>Rate parameter</td>
<td>min $^{-1}$</td>
<td>0.016</td>
</tr>
<tr>
<td>$k_d$</td>
<td>Rate parameter</td>
<td>min $^{-1}$</td>
<td>0.016</td>
</tr>
<tr>
<td>$k_{sc}$</td>
<td>Rate parameter</td>
<td>min $^{-1}$</td>
<td>0.095</td>
</tr>
<tr>
<td>$k_{gr1}$</td>
<td>Rate of grinding</td>
<td>min $^{-1}$</td>
<td>0.039</td>
</tr>
<tr>
<td>$b$</td>
<td>Meal related parameter</td>
<td>unitless</td>
<td>0.79</td>
</tr>
<tr>
<td>$d$</td>
<td>Meal related parameter</td>
<td>unitless</td>
<td>0.18</td>
</tr>
<tr>
<td>$k_{max}$</td>
<td>Rate parameter</td>
<td>min $^{-1}$</td>
<td>0.039</td>
</tr>
<tr>
<td>$k_{min}$</td>
<td>Rate parameter</td>
<td>min $^{-1}$</td>
<td>0.074</td>
</tr>
<tr>
<td>$f$</td>
<td>Meal related parameter</td>
<td>unitless</td>
<td>0.9</td>
</tr>
<tr>
<td>$k_{abs}$</td>
<td>Rate parameter</td>
<td>min $^{-1}$</td>
<td>0.21</td>
</tr>
<tr>
<td>$k_c1$</td>
<td>Glomerular filtration rate</td>
<td>min $^{-1}$</td>
<td>0.0005</td>
</tr>
<tr>
<td>$k_c2$</td>
<td>Renal threshold of glucose</td>
<td>mg/kg</td>
<td>339</td>
</tr>
<tr>
<td>$F_{inc}$</td>
<td>Insulin-independent glucose utilization</td>
<td>mg/kg/min</td>
<td>1</td>
</tr>
</tbody>
</table>