Document downloaded from:

http://hdl.handle.net/10251/200345

This paper must be cited as:

Faccioli, S.; Sala-Mira, I.; Diez, J.; Facchinetti, A.; Sparacino, G.; Del Favero, S.; Bondía Company, J. (2022). Super-twisting-based meal detector for type 1 diabetes management: Improvement and assessment in a real-life scenario. Computer Methods and Programs in Biomedicine. 219:1-10. https://doi.org/10.1016/j.cmpb.2022.106736

The final publication is available at

https://doi.org/10.1016/j.cmpb.2022.106736

Copyright Elsevier

Additional Information

Revised Manuscript clean version (Without Track changes) Click here to view linked References

Super–twisting-based meal detector for type 1 diabetes management: improvement and assessment in a real-life scenario

Faccioli S.[a,1], Sala-Mira I.[b,1], Diez J. L.[b,c], Facchinetti A.[a], Sparacino G.[a], Del Favero S.[a,*], Bondia J.[b,c]

[a] *Department of Information Engineering - DEI, University of Padova, 35131, PD, Italy*
[b] *Instituto Universitario de Automática e Informática Industrial, Universitat Politècnica de València, 46022, Valencia, Spain*
[c] *Centro de Investigación Biomédica en Red de Diabetes y Enfermedades Metabólicas Asociadas - CIBERDEM, 28028, Madrid, Spain*

Abstract

Background and Objectives. Hybrid automated insulin delivery systems rely on carbohydrate counting to improve postprandial control in type 1 diabetes. However, this is an extra burden on subjects and it introduces a source of potential errors that could impact control performances. In fact, carbohydrates estimation is challenging, prone to errors, and it is known that subjects sometimes struggle to adhere to this requirement, forgetting to perform this task. A possible solution is the use of automated meal detection algorithms. In this work, we extended a super–twisting-based meal detector suggested in the literature and assessed it on real-life data.

Abbreviations: AID, automated insulin delivery; BG, blood glucose; CGM, continuous glucose monitor; CHO, carbohydrates; FN, false negative; FP, false positive; MD, meal detector; SAP, sensor-augmented pump; STMD, super–twisting-based meal detector; T1D, type 1 diabetes; TP, true positive

*Corresponding author. Tel.: +39-049-827-7747
Email address: sdelfave@dei.unipd.it (Del Favero S.)

[1] These authors contributed equally

Methods. To reduce the false detections in the original meal detector, we implemented an implicit discretization of the super–twisting and replaced the Euler approximation of the glucose derivative with a Kalman filter. The modified meal detector is retrospectively evaluated in a challenging real-life dataset corresponding to a 2-week trial with 30 subjects using sensor-augmented pump control. The assessment includes an analysis of the nature and riskiness of false detections.

Results. The proposed algorithm achieved a recall of 70 [13] % (median [interquartile range]), a precision of 73 [26] %, and had 1.4 [1.4] false positives-per-day. False positives were related to rising glucose conditions, whereas false negatives occurred after calibrations, missing samples, or hypoglycemia treatments.

Conclusions. The proposed algorithm achieves encouraging performance. Although false positives and false negatives were not avoided, they are related to situations with a low risk of hypoglycemia and hyperglycemia, respectively.
Keywords: type 1 diabetes, automated insulin delivery system, postprandial control, meal detection

1. Introduction

Type 1 diabetes (T1D) is a chronic endocrine disorder related to an endogenous insulin deficiency due to autoimmune destruction of the pancreatic β-cells. Insulin promotes glucose disposal by muscles and adipose tissues and inhibits hepatic glucose production. Thus, the lack of insulin leads to high plasma glucose concentration (hyperglycemia). Hyperglycemia is associated

with long-term complications such as retinopathy, nephropathy, neuropathy, or heart disease [1]. As a result, people suffering from T1D need an exogenous insulin replacement to maintain glucose levels in a safe range (normoglycemia). However, an insulin over-delivery might cause a low glucose concentration (hypoglycemia), leading to anxiety, palpitations, cognitive dysfunction, seizures, and, in severe cases, coma or even death [1]. Automated insulin delivery (AID) is a promising technology to manage tight glucose control and, consequently, reduce the risks of hyperglycemia and hypoglycemia. Several clinical trials have reported that AID systems (aka artificial pancreas) significantly improve blood glucose (BG) control as compared to traditional intensive insulin therapies [2]. The main improvement occurs in the nocturnal period, but diurnal glucose is still challenging: one of the principal reasons is the considerable impact of meal ingestion.

In postprandial control, continuous glucose monitor (CGM) lags and subcutaneous insulin absorption dynamics limit the AID effectiveness in postmeal glucose absorption compensation [3]. The main consequence is a sustained hyperglycemia event right after the meal. In addition, an over reaction of the AID might lead to late hypoglycemia due to the stacked insulin on board [3]. Therefore, most AID systems have adopted carbohydrate (CHO) counting to overcome the challenges of postprandial control [4]. In AID systems requiring CHO counting, i.e., hybrid AID, users must estimate the meal CHO content for the AID to deliver a pre-meal insulin bolus. This pre-meal bolus advances the insulin needs to compensate for the glucose rise despite absorption delays, improving postprandial control. However, CHO counting might degrade AID performance due to timing delays, announcement omis-

sions, and inaccurate CHO estimations [5, 6]. In addition, the extra burden on the users constitutes a reason for disconformity with the technology [7].

Therefore, several approaches have been studied in the literature to remove CHO counting: automated meal detection (MD) [8, 9, 10], disturbance–observer-based control [11], model predictive control [12], reinforcement learning [13], multi-hormone AID systems [14], etc. MD is the traditional alternative to CHO counting since it is simple and usually forms an independent module from the main AID controller. AID integrates MD to increase the control aggressiveness against the meal, either by delivering insulin boluses [15, 16, 17] or changing the controller tuning [18]. MD design usually comprises two stages: 1) to calculate signals and features sensitive to the meal, and 2) to build a decision logic to determine the meal occurrence. Except for a few works that incorporate other measurements (heart rate [9], abdominal sound [19], head movement [20], etc.), features are usually related to CGM readings. Frequent features are CGM values, approximations of the CGM derivative [9], glucose predictions, innovation terms or residuals [15, 16, 21], and estimations of the meal rate of glucose appearance [22]. Combinations of those signals have also been used as features: for instance, authors in [22] applied the cross-covariance of the estimated glucose and the estimated rate of glucose appearance. Note that the abovementioned signals are estimated by multiple model simulations [9] or observers (such as Kalman filters [16, 21, 22], sliding mode observers [15], moving-horizon estimators [8, 23]). About the decision logic, the most extended method is to apply *ad-hoc* thresholds to the features. Other techniques rely on interval thresholds [21], hypothesis tests [16, 17], or binary classifiers [8, 24]. Besides meal detec-

tion, another relevant feature for meal-announcement-free control is estimating the meal CHO content, addressed, for example in [25, 16, 23]. An accurate estimation of the meal CHO content simplifies the control algorithm and allows more aggressiveness since bolusing strategies can be adopted. However, other algorithms for unannounced meal compensation can also be conceived without the explicit information of the meal CHO content. For example, [18] and [15] replace the CHO estimation with inner states or disturbances estimations, respectively. Therefore, this work will exclusively focus on the meal detection problem rather than estimating meal CHO content.

This work aims at improving the implementation and validation of the MD algorithm proposed in [15]. This algorithm determines the meal occurrence with thresholds on a glucose derivative estimation and a residual signal generated by a super–twisting observer. Despite its satisfactory performance, the algorithm has two main limitations. On one hand, its validation setting was not challenging enough since it used simulation data of only 10 virtual subjects. On the other hand, the algorithm is prone to false detections because either the glucose or the super–twisting discretization uses explicit Euler approximations leading to oscillations [15, 26]. Hence, to reduce the oscillations in the residuals, the implicit discretization method in [26] is implemented. Also, a Kalman filter is used to obtain a smooth estimation of the glucose derivative. Finally, a retrospective validation is performed with a challenging dataset of 30 subjects in free-living conditions [27] [2].

Hereafter, this article is organized as follows. Section 2 includes the new features of the MD as compared to the work in [15]. Furthermore, this section explains the tuning procedure and the validation. Then, in section

3, an exhaustive analysis of the tuning and validation results is performed. Finally, section 4 closes the article with the conclusions.

2. Materials and Methods

2.1. Super–twisting-based meal detector

The super–twisting-based meal detector (STMD) used in this work extends the algorithm proposed in [15]. The STMD algorithm is built upon three elements: 1) a *super–twisting observer* generates the residuals – a signal that represents the inconsistency between the glucose and the glucose estimation –, 2) a *Kalman filter* estimates the glucose derivative, and 3) a set of *decision rules* determines the occurrence of the meal based on the residuals and glucose derivative.

A schematic flowchart representing the sequence of the computations of the proposed algorithm is depicted in figure 1: every step will be described in detail in the following sections. Moreover, a list of variables and constants is provided in table 1.

2.1.1. Residual generation

A super–twisting observer was used to estimate the glucose given its robustness to noise and unmatched disturbances [28]. In the ideal case –

[2]The source of the data is the JDRF Artificial Pancreas Consortium coordinating center (JDRFAPPCC), but the analyses, content and conclusions presented herein are solely the responsibility of the authors and have not been reviewed or approved by the JDRFAPPCC. The public CTR3 dataset was financed by JDRF through the grants JDRF 22-2011-649 and JDRF 17-2013-509.

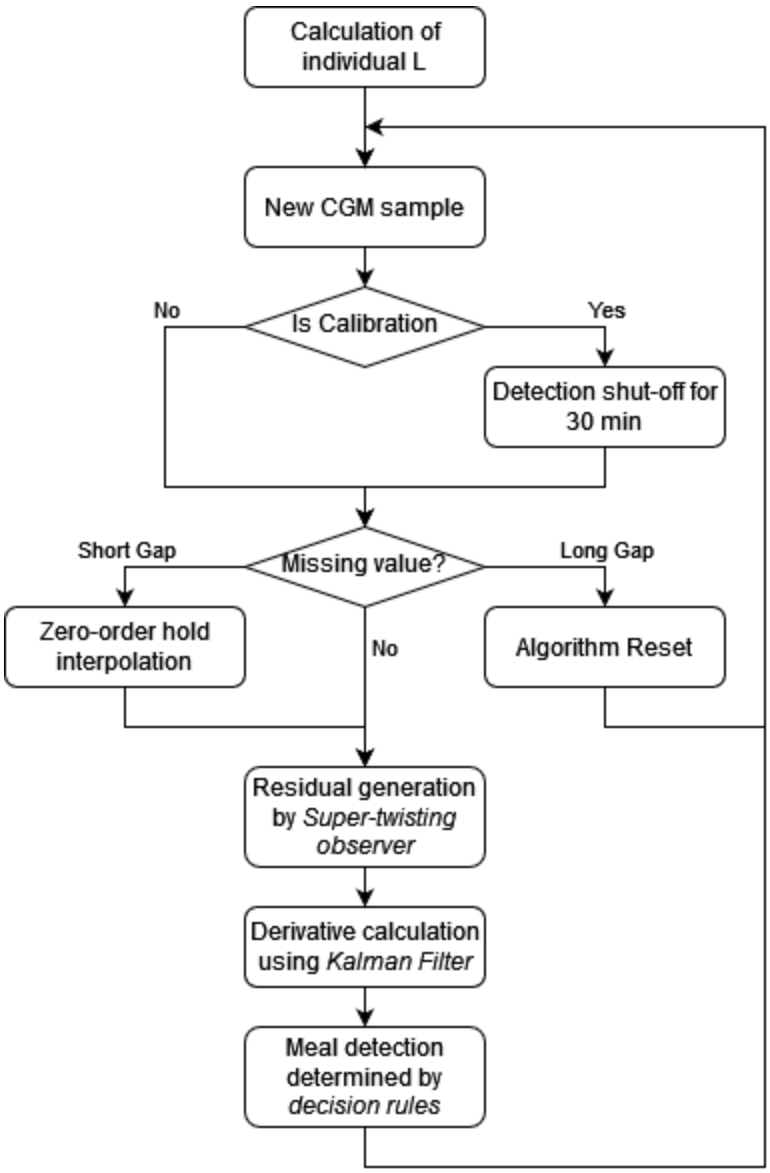


Figure 1: Flowchart of the proposed algorithm

continuous-time measurements without noise – the super–twisting equations read as:

$$\begin{cases} \dot{\widehat{ig}}(t) = u(t) + k_1 \left| res_{ig}(t) \right|^{0.5} \text{sign}(res_{ig}(t)) \\ \dot{u}(t) = k_2 \text{sign}(res_{ig}(t)) \end{cases} \tag{1}$$

where $ig(t)$ is interstitial glucose (ideal CGM reading), $\widehat{ig}(t)$ is the super–twisting estimate of $ig(t)$, $res_{ig}(t)$ is the ideal residual signal ($res_{ig}(t) = ig(t) - \widehat{ig}(t)$), $u(t)$ is an auxiliary signal, and the gains k_1 and k_2 need to be designed.

As shown in [15], the observer given by (1) ensures that the residual dynamics follows the super–twisting equations [29]:

$$\begin{cases} \dot{res}_{ig}(t) = \xi(t) - k_1 \left| res_{ig}(t) \right|^{0.5} \text{sign}(res_{ig}(t)) \\ \dot{\xi}(t) = F(t) - k_2 \text{sign}(res_{ig}(t)) \end{cases} \tag{2}$$

where $F(t)$ is a lumped disturbance term that depends on the meal rate of glucose appearance, its derivative, and other perturbing factors such as unmodeled dynamics (see [15] for more details). If $|F(t)| \leq L$, for $L > 0$, and gains k_1 and k_2 are selected as

$$k_1 = 1.1L \qquad k_2 = 1.5L^{0.5} \tag{3}$$

then the residuals $res_{ig}(t)$ will converge to 0, after a finite time [28, 29].

However, the only glucose-related available signal is the CGM readings. Since this time series is a discrete-time noisy signal, the super-twisting observer must be discretized, with a new timestamp k. Here we hence need to introduce $cgm(k)$ as the CGM readings signal, $\widehat{cgm}_{ST}(k)$ as the super–twisting estimation of $cgm(k)$, and $res(k) = cgm(k) - \widehat{cgm}_{ST}(k)$.

Note that the feature of convergence after a finite time is interesting for the meal detection purpose. Indeed, disturbances lower than L will have a null impact on $res_{ig}(t)$ – due to the absence of noise and the infinite sampling rate –, and a reduced impact on $res(k)$. Nevertheless, when $|F(k)| > L$ (with $F(k)$ being the discrete version of $F(t)$), $res(k)$ will still significantly deviate from 0. Therefore, we can apply a simple threshold-based logic on the $res(k)$ signal to discriminate the meal occurrence (see section 2.1.3).

Implicit discretization. The discretization of sliding–mode-based algorithms is a highlighted concern since chattering, an oscillatory behavior in the states when the sliding mode regime is reached, limits the performance [30]. The super–twisting algorithm mitigates the chattering effect, as compared to first-order sliding-mode observers, by hiding the discontinuity in the derivative. Nonetheless, explicit discretization methods still cause numerical chattering [30]. Sala-Mira *et al.* [15] alleviated the chattering issue with subsampling and filtering strategies, but at the price of higher computational load and delays. As an alternative, in this article, implicit discretization methods are utilized since they reduce and even suppress chattering [26, 31]. The implicit method resorts from a set-valued interpretation of the sign function. Hence, the direct discretization of expression (1) is defined as:

$$\begin{cases} \widehat{cgm}_{ST}(k) - \widehat{cgm}_{ST}(k-1) = hu(k) + hk_1 \left|res(k)\right|^{0.5} \lambda(k) \\ u(k) - u(k-1) = hk_2\lambda(k) \\ \lambda(k) \in \text{msign}(res(k)) \end{cases} \tag{4}$$

where h denotes the sampling time, k is the discrete iteration, and msign$(\cdot)$ refers to the set-valued sign function defined as [26]:

$$\text{msign}(res(k)) := \begin{cases} 1 \text{ if } res(k) > 0 \\ [-1,1] \text{ if } res(k) = 0 \\ -1 \text{ if } res(k) < 0 \end{cases} \tag{5}$$

Expression (4) might seem to be non-implementable given the presence of $\widehat{cgm}_{ST}(k)$ in both sides of the equality. However, by exploiting some properties of convex sets (such as the relation of (5) to the normal cone [31, 32]), the following implementable implicit discretization of (1) is obtained [26]:

$$g(k) = \widehat{cgm}_{ST}(k-1) + hu(k-1) \tag{6a}$$

$$\tilde{g}(k) = cgm(k) - g(k) \tag{6b}$$

$$\zeta(k) = \frac{4}{h^2 k_1^2}\left(\tilde{g}(k) - k_2 h^2\right) \tag{6c}$$

$$\widehat{cgm}_{ST}(k) = \begin{cases} g(k) + h^2 k_2 + \dfrac{h^2 k_1^2}{2}\left(\sqrt{1+\zeta(k)} - 1\right) \text{ if } \tilde{g}(k) > h^2 k_2 \\ cgm(k) \text{ if } |\tilde{g}(k)| \leq h^2 k_2 \\ g(k) - h^2 k_2 - \dfrac{h^2 k_1^2}{2}\left(\sqrt{1-\zeta(k)} - 1\right) \text{ if } \tilde{g}(k) < -h^2 k_2 \end{cases} \tag{6d}$$

$$\lambda(k) = proj\left([-1,1], \frac{\tilde{g}(k)}{h^2 k_2}\right) \tag{6e}$$

$$u(k) = hk_2\lambda(k) + u(k-1) \tag{6f}$$

where $proj([-1,1],a)$ is the Euclidean projection operator defined as

$$b = proj([-1,1],a) = \begin{cases} 1 \text{ if } a > 1 \\ a \text{ if } |a| < 1 \\ -1 \text{ if } a < -1 \end{cases} \tag{7}$$

for scalars $a \in \mathbb{R}$ and $b \in \mathbb{R}$.

Estimation of disturbance $F(k)$. As it will be shown in section 3.1, the knowledge of $F(k)$ will be useful to set the parameter L. Since $F(k)$ is a lumped disturbance that is not available in real-time, an estimation of $F(k)$, i.e., $\widehat{F}(k)$, is needed. To this end, the concept of the equivalent control in the sliding mode is applied [33]. According to [34], to maintain the sliding regime, the average value of $k_2\text{sign}(res(k))$, i.e, the equivalent injection term $k_2\text{sign}(res(k))|_{eq}$, in (2) must be equal to $F(k)$. Although the equivalent term is a theoretical concept, an approximation can be obtained by filtering the discontinuous term. Consequently, an estimation of the disturbance $F(k)$ is achieved by [33]

$$\widehat{F}(k) = \frac{h}{\tau}\left(k_2\text{sign}(res(k-1)) - \widehat{F}(k-1)\right) + \widehat{F}(k-1) \tag{8}$$

where τ is a positive constant which is set to 5 min in this work.

2.1.2. Kalman filter

Sala-Mira *et al.* [15] estimated the glucose derivative by a numerical approximation. This approximation is not robust to sensor noise. In this work, a smooth estimation of the derivative provided by a Kalman filter is used instead.

At discrete time, the Kalman filter is implemented by difference equations that recursively estimate the unknown state vector $x(k)$ of a dynamic

system exploiting vectors of noisy measurements $y(k)$ causally related to it. In particular, we can define a discrete state-space model as [35]:

$$\begin{cases} x(k+1) = Ax(k) + Cw(k) \\ y(k) = Gx(k) + v(k) \end{cases} \tag{9}$$

where $x(k)$ is the $n \times 1$ state vector; $y(k)$ is the output vector; $v(k)$ is the measurement noise (white zero-mean noise with variance σ_v^2 , and covariance matrix R); $w(k)$ is the scalar innovation process (white zero-mean noise with variance σ_w^2, and covariance matrix Q); A is the $n \times n$ state matrix, C the $n \times 1$ noise-state vector, and G is the $1 \times n$ output vector. Note that the subsequent formulas hold only if $w(k)$ is uncorrelated from $v(\ell)$ for all k and ℓ.

The linear minimum variance estimate of the state vector obtainable from the measurement collected until time k –indicated by $\hat{x}(k|k)$– can be updated up to time $k+1$ by using the following linear recursive equations [36]:

$$\begin{cases} \hat{x}(k+1|k+1) = A\hat{x}(k|k) + K(k+1)[y(k+1) - GA\hat{x}(k|k)] \\ K(k+1) = P(k+1|k)G^T[GP(k+1|k)G^T + R]^{-1} \\ P(k+1|k+1) = [I_n - K(k+1)G]P(k+1|k) \\ P(k+1|k) = AP(k|k)A^T + CQC^T \end{cases} \tag{10}$$

where $P(k|k)$ (size $n \times n$) is the covariance matrix of the estimation error affecting $\hat{x}(k|k)$, $K(k)$ (size $n \times 1$) is the Kalman gain matrix, I_n (size $n \times n$) is the identity matrix, and $P(0|0)$ and $\hat{x}(0|0)$ are the initial conditions.

Now, an *a priori* description of the unknown signal is necessary. In this work, we assumed that process noise drives the second derivative of glucose

with time. Thus, we used the following third-order (three-state) model [37]:

$$A = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad G = [1\ 0\ 0]$$
$$x(k) = \begin{bmatrix} \widehat{cgm}(k) \\ \widehat{der}(k) \\ \widehat{f}(k) \end{bmatrix}, \quad y(k) = cgm(k) \tag{11}$$

where $\widehat{cgm}$, $\widehat{der}$, and $\widehat{f}$ represent glucose, rate of change of glucose, and the second derivative of glucose with respect to time, respectively. The advantage of the three-state model is that it captures dynamics near the maximum (peak) and minimum (valley) values of glucose. Note that considering a third-order model is an assumption commonly adopted on the regularity of the process when information that we have / we want to use is minimal [37, 38].

Finally, the Q and R matrices, i.e., the process and the measurement noise covariance matrices (respectively), are key parameters in determining the performance of the Kalman filter. They are defined as

$$Q = \sigma_w^2, \quad R = \sigma_v^2 \tag{12}$$

where σ_w^2 and σ_v^2 are usually unknown: in this work, we selected $\sigma_w^2 = 0.01$ and $\sigma_v^2 = 4$, as done in [37].

2.1.3. Decision rules

Two decision rules are then applied to the residuals and the glucose derivative to determine the meal occurrence. In particular, a new detection

is considered if the residuals exceed a certain threshold ($res(k) > ThRes$) and glucose increases at a certain rate ($\widehat{der}(k) > ThDer$) [15]. Moreover, two consecutive meal detections will be separated at least by $TW_{shut} = 90$ min, in order to avoid multiple detections triggered by the same meal.

Alarm silencing strategy. Glucose data was collected (see section 2.2) using a CGM sensor that needed calibration twice a day. These calibrations could lead to big leaps in the CGM trace: unfortunately, the super–twisting observer is not able to interpret this anomalous behaviour, leading to spikes in the $res(k)$ time series. For this reason, to avoid spurious false detections, we decided to shut off the detection after a calibration for 30 min.

2.2. Real dataset

2.2.1. Data collection

Data was collected in a multicenter clinical trial [27] (`www.clinicaltrials.gov`: NCT02137512), whose main phase was aimed to assess the feasibility of a long-term AID system. In particular, in an initial phase of the study, 30 subjects spent 2 weeks at home using sensor-augmented pump (SAP) therapy, which was considered for baseline comparison. In this work, we used the data provided by this 14-day long phase to assess the performance of our STMD algorithm. The study and all experimental procedures were approved by local IRB/ethical committee.

Insulin was infused with a Roche Accu-Check Spirit Combo® insulin pump (Roche Diabetes Care, Inc., Indianapolis, IN, USA), whereas glucose data were recorded using a DexCom G4® sensor (DexCom, Inc., San Diego, CA, USA) with a sample time of 5 min. Moreover, individuals were instructed

to manually deliver a proper amount of insulin for all meals, by inserting in the system all CHO intake. For further details on the experiment, we refer to [27].

Finally, it should be noticed that the algorithm was tested on a dataset collected in free-living conditions. Datasets of this type are substantially more challenging than data collected in inpatient clinical trials (i.e., on hospitalized trials) or data collected in outpatient trials with strictly regimented protocol, because of the much larger incidence of unknown disturbances and confounding factors.

2.2.2. Preprocessing

Dealing with experimental data of different nature (e.g., CGM and CHO intake) poses some technical issues related to device synchronization, completeness of stored data, and reliability of the patient's provided information. Firstly, to fix synchronization problems, all signals were aligned to the same time grid equally sampled at $T_S = 5$ min. Secondly, we selected only portions of data that have certain meal information. Even if subjects were instructed to insert CHO intake in the system, some missed meal announcements still occurred. Lastly, for what concerns glucose missing information, we defined "long gap" as more than 30 minutes of missing CGM data, while "short gap" is any gap lasting less than 30 min. Short gaps were then filled using a zero-order hold; conversely, when a long gap occurred, the algorithm was reset, switching it on again when the CGM time series was recovered.

To conclude, note that other, more advanced, interpolation options are available. However, investigating them now seems less relevant since their impact on the overall performance is expected to be limited as missed values

are only 1.1% of all considered CGM samples.

2.3. Meal detection assessment

Here we introduce the following definitions:

- true positive (TP): when a meal is rightly detected within a detection window DW, starting from the meal at time k_m and up to TW_{max} min, i.e., $DW = [k_m,\ k_m + TW_{max}]$;
- false positive (FP): when a detection at k_d is triggered and no meal occurred in the window $[k_d - TW_{max},\ k_d]$;
- false negative (FN): when a meal occurs and no detection is issued in the DW interval;
- late detection (LD): detections raised in the interval $DW_{ld} = [k_m + TW_{max},\ k_m + TW_{max} + TW_{LD}]$. Clearly, these episodes are not counted as TP, as they were not timely triggered, and they increase the count of FN. Nonetheless, they can not be considered as erroneous, so they do not increase the FP count.

Note that the calculation of true negative (TN) is of limited interest [39], since we are dealing with an unbalanced dataset (only 3/4 meal occurrences per day of monitoring).

In this work, we set $TW_{max} = 180$ min, and $TW_{LD} = 120$ min. A detection window of 180 minutes is needed since the dataset included cases where the glucose levels remain constant or even decrease within the first 1-2 h after the subject registered the meal. Given such slow absorption periods,

a $TW_{max} = 180$ has been considered convenient for a fair evaluation of the algorithm. A similar time window length was considered in [40].

Once *TP*, *FP*, and *FN* have been found, the following metrics were used to evaluate the proposed super–twisting-based meal detector:

$$precision = 100 \cdot \frac{TP}{TP + FP} \tag{13}$$

$$recall = 100 \cdot \frac{TP}{TP + FN} \tag{14}$$

$$F1\_score = 2 \cdot \frac{precision \cdot recall}{precision + recall} \tag{15}$$

Additionally, since the dataset is unbalanced, we evaluated the daily number of *FP* generated by the algorithm (FP-per-day). Moreover, we calculated the detection time (DT) as the time between the meal occurrence and when the alarm has raised. Please note that we did not know the exact real meal time, but we had to rely on the time reported by the user. Finally, we reported the carbohydrate content related to *FN*s (CHO_{FN}).

2.3.1. Refinements on performance evaluation

Here we will describe some refinements used in the evaluation of the STMD.

Firstly, note that in a meal announcement-free AID environment, hypoglycemia treatments (fast-acting carbohydrates consumption to avoid severe hypoglycemia) do not need insulin injection. However, although the proposed meal detector is part of a fully-automated AID system, in this work –focusing only on the meal detection algorithm– we treated hypoglycemia treatments as normal CHO consumption.

Lastly, it could happen in real-data that multiple meals occur very close to each other. For example, a subject can eat the main dish, and shortly after insert in the system a dessert consumption; moreover, to avoid hypoglycemia, a patient could need multiple consecutive hypoglycemia treatments. To this end, if the data contained meals closer than 30 min, we considered only the second one, not counting the first one neither as a *TP* nor as a *FP*.

3. Results and Discussions

3.1. Tuning of parameters

This section investigates the tuning of the STMD parameters, i.e., L, *ThRes*, and *ThDer*.

Regarding the upper bound of the disturbance $F(k)$, i.e., L, we performed a preliminary analysis to find the best way to tune this crucial parameter. In particular, using our real data, we found that the estimated disturbance $\hat{F}(k)$ calculated over night is quite different from the disturbance estimated just after a meal occurrence. In fact, figure 2 shows the estimated disturbance $\hat{F}(k)$ of all patients, represented as *errorbar*, i.e., mean and standard deviation over the population. On one hand, on the left panel, we can see the distribution between all meals: 0 indicates mealtime, and the blue bars represent the estimation of $\hat{F}(k)$ after a meal occurrence. On the other hand, the right panel shows the distribution of $\hat{F}(k)$ during all night periods. It can be easily seen that the estimated disturbance during the night is smaller than the one calculated after a meal occurrence.

For this reason, we defined and used a personalized L (and, consequently, the gains k_1 and k_2) as the maximum estimated disturbance $\hat{F}(k)$ calculated

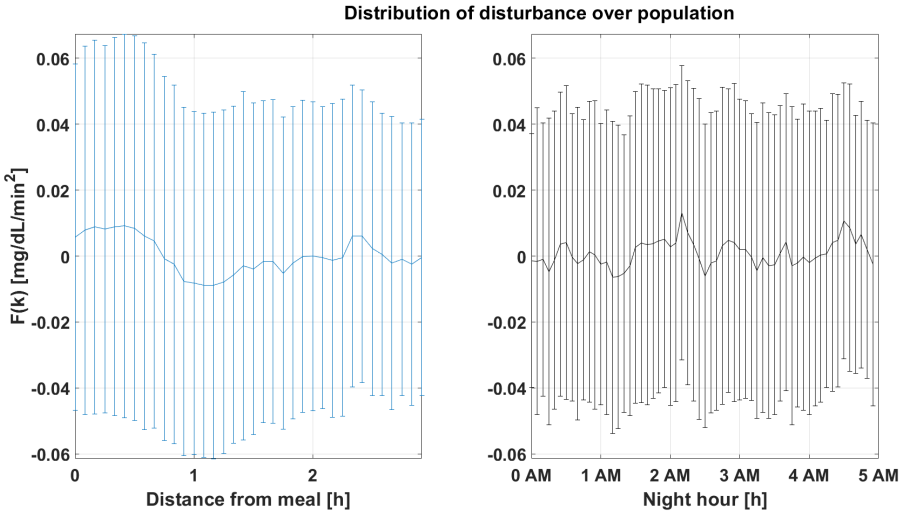


Figure 2: Distribution over the population of the estimated disturbance $\hat{F}(k)$, after a meal (left panel), or during the night period (right panel)

over the first night for each patient.

For what it concerns *ThRes* and *ThDer*, instead, for each parameters combination, the performance of our algorithm on the population was considered as a point in the [*precision*, *recall*] space. For instance, figure 3 depicts the performance obtained in this space, where ideal performance is achieved at the top right corner (i.e., *precision*=100%, *recall*=100%). Each curve was obtained for different values of *ThDer* and colored depending on it, while different points of the same curve were obtained using different values of *ThRes*.

In figure 3, it can be noticed that, as expected, smaller values of both *ThDer* and *ThRes* are associated with more aggressive configurations, showing higher recall in exchange for a lower precision. To find a suitable trade-off between precision and recall, we selected the combination of [*ThRes*, *ThDer*]

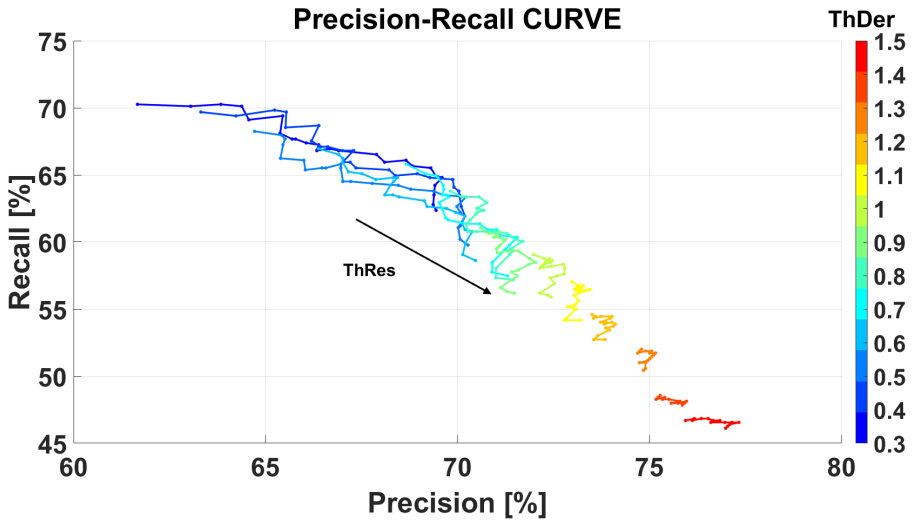


Figure 3: Precision *vs.* Recall analysis: each curve is obtained using different values of $ThDer$, while each point of a curve is obtained for different values of $ThRes$

that maximizes the population *F1_score*. A similar approach could be applied to obtain a personalized *ThRes*, *ThDer*, by maximizing the individual *F1_score* rather than population one. However, a population-based approach was used instead because some individuals contain only a few registered meals (e.g., 11 out of 30 subjects have less than 20 registered meals for 14 days).

3.2. Performance of STMD on real data

Table 2 reports the performances of the STMD algorithm: all metrics are expressed as median [interquartile range (iqr)]. Table 3 shows instead how the STMD compares to other meal detectors evaluated with free-living condition data in the literature. It should be remarked that due to the use of different datasets, preprocessing techniques, and evaluation metrics, the comparison only serves as an illustration and must be interpreted with caution. The proposed algorithm showed a recall of 70 [13] %, very similar to

most literature strategies [8, 9, 10], that showed recall values between 70 and 79 %. However, some literature works showed outperforming results with a recall of about 99 % [10] or 92.5 % [8]. Conversely, the proposed method achieves similar FP-per-day to "LDA CGM" or "Threshold Ra" in [8] while outperforming the other algorithms.

For what concerns the detection time, as anticipated, we calculated DT as the difference between the meal time reported by the user and its detection. The proposed STMD algorithm achieved on average a $DT = 45$ min, in line with most literature works [41, 42]. Nevertheless, a number of papers showed better performance in terms of DT, e.g., [8]. Please note, however, that authors in [8] estimated the meal time retrospectively: the detection time was calculated based on a shifted time instant where glucose rate of change was greater than 1 mg/dL/min. Finally, comparing the DT obtained in this work with the one obtained in the original work [15] (i.e., on average $DT =$ 30 min), we may say that real data are much more challenging than simulated data. In fact, illness or physical activity, for example, can moderately change the dynamics of postprandial periods in real data with respect to simulated one, which cannot such features yet. As indicated in Section 2.3, the dataset utilized in this work features slow postprandial absorption periods; thus, the STMD algorithm may need time to detect the corresponding meals.

3.3. False detections analysis

In this section, we will investigate the clinical impact of FPs and FNs.

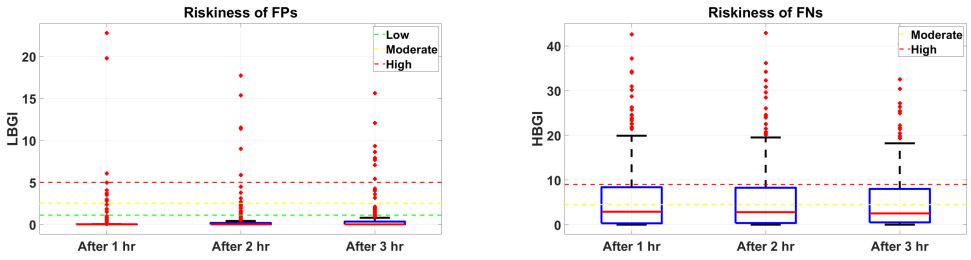
Before showing the results, we introduce two metrics widely used to quantitatively evaluate the quality of glycemic control: low BG index (LBGI) and high BG index (HBGI), [43]. In particular, let $cgm(k_1)$, $cgm(k_2)$, ..., $cgm(k_N)$

be N CGM readings of a subject, and let us introduce the hypoglycemic risk $rl(cgm)$ and the hyperglycemic $rh(cgm)$ associated to each reading as

$$\begin{aligned} rl(cgm(k_i)) &= \begin{cases} r(cgm(k_i)) & \text{if} \quad cgm(k_i) < 112.5 mg/dL \\ 0 & \text{otherwise} \end{cases} \\ rh(cgm(k_i)) &= \begin{cases} r(cgm(k_i)) & \text{if} \quad cgm(k_i) >= 112.5 mg/dL \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{16}$$

where $r(cgm(k_i)) = 10 \cdot cgm_{mod}(k_i)^2$, and $cgm_{mod}(k_i) = 1.509 \cdot (log(cgm(k_i))^{1.084} - 5.381)$. Finally, the indexes are defined as:

$$\begin{cases} LBGI = \dfrac{1}{N} \displaystyle\sum_{i=1}^{N} rl(cgm(k_i)) \\ HBGI = \dfrac{1}{N} \displaystyle\sum_{i=1}^{N} rh(cgm(k_i)) \end{cases} \tag{17}$$



(a) (b)

Figure 4: Boxplots of LBGI and HBGI for FP and FN occurrences

3.3.1. FPs' nature and riskiness

Figure 4a reports the boxplot of the LBGI calculated 1, 2, and 3 hours after each *FP* occurrence (i.e., with N= 12, 24, and 36, respectively). Note that *FP*s could be dangerous since they can lead to hypoglycemia. In fact, when

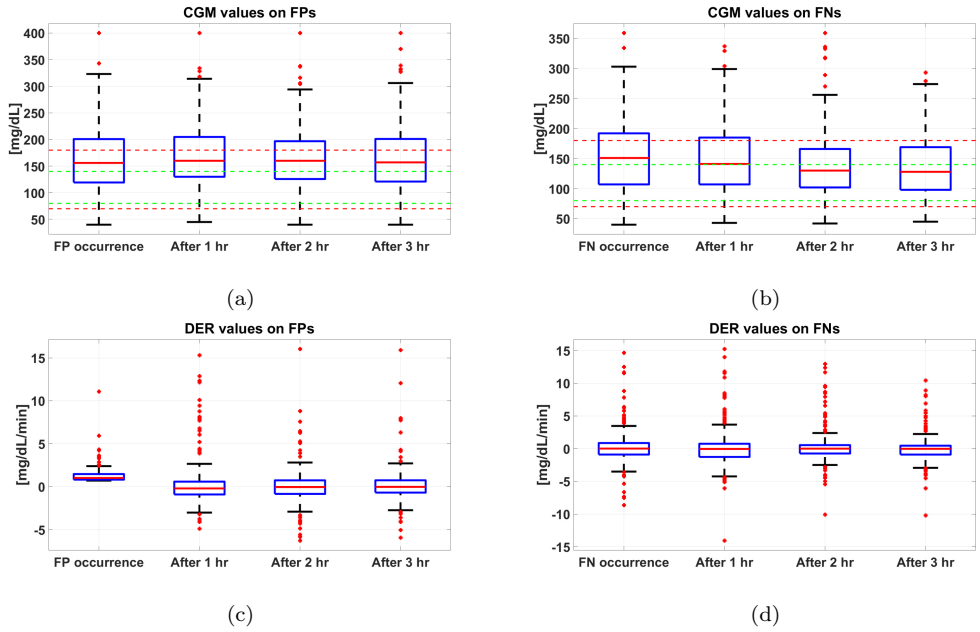


Figure 5: Boxplots of CGM and DER values for FP and FN occurrences

used within a meal announcement-free AID system, if the STMD wrongly detects a meal, the subsequent bolusing strategy will automatically inject unwanted insulin. Now, figure 4a shows that after *FP* occurrences, the risk of hypoglycemia is, on average, very low. In this case, an insulin injection would not worsen an already risky situation.

Figures 5a and 5c show the distribution of *cgm* and *der* values related to *FP*s, right when they occurred and 1, 2, and 3 hours afterwards. We can see that, on average, a *FP* occurs with a *cgm* value quite high, nearly 160 mg/dL, and with a positive derivative: the glucose trace is then not in a dangerous zone, and it is, moreover, increasing. Figure 6 shows an example of one of these events. In the top panel, *cgm* values are depicted in blue, its estimation in dashed red, and the detection in yellow triangle; in the

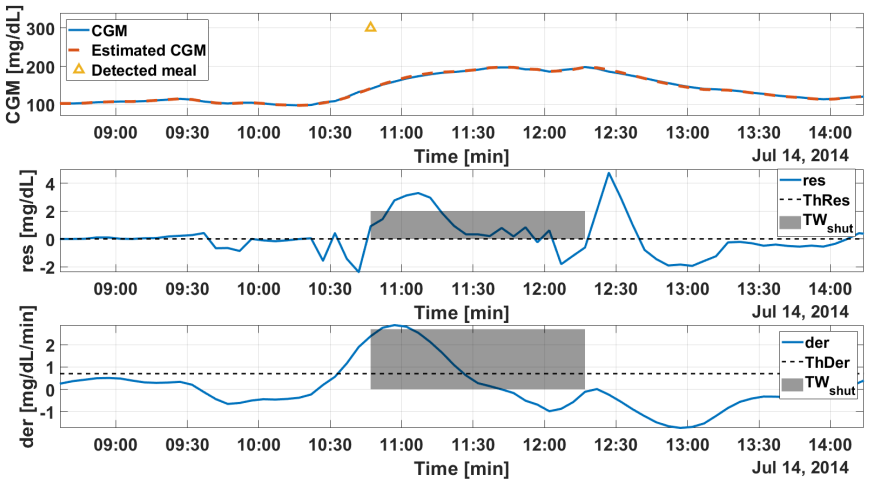


Figure 6: Example of a FP

middle panel is represented the *res* signal, while in the bottom panel the *der* signal. We can easily see that the *cgm* trace is rising, even if there is no meal intake. Although not caused by a meal, these curves are still occurrences where the *cgm* is rising like after a real CHO consumption. We calculated the percentage of *FP*s that were similar to missed meal announcements, as the ratio between the number of events where Δcgm >50 within 3 hours with starting *cgm* >70, and the total number of *FP*s: this percentage was approximately 19 %.

To conclude, nearly 20 % of *FP*s had a meal-like shape, while on average all *FP*s occurred with high BG values and positive derivatives.
Insulin dosing in these cases would not be so risky, as these episodes would likely need insulin anyway.

3.3.2. FNs' nature and riskiness

Figure 4b reports the boxplot of the HBGI calculated for the *FN* occurrences. Note that FNs could be dangerous for hyperglycemia; if the STMD is not able to detect a meal, a possible dangerous increase in the BG levels will likely lead to hyperglycemia. Fortunately, figure 4b shows that after *FN* occurrences, the risk of hyperglycemia is on average moderate. Even in this case, using the STMD within a meal announcement-free AID environment would not worsen an already dangerous situation.

To assess the behaviour of *FN*s using our STMD algorithm, we calculated the percentage of *FN* that were hypoglycemia treatments (FN_{hypo}). Moreover, to deal with real data issues, the STMD was shut-off in two different cases: after a calibration and after a long period of missing data. In both events, a number of meals could happen right during these times, but the STMD will not be able to detect them. To quantify the number of such events, we calculated the percentage of *FN* that happened after a calibration (FN_{cal}), and the percentage of *FN* that happened after a reset of the algorithm due to missing samples (FN_{res}).

In particular, $FN_{hypo} = 14.3$ %: nearly 15 % of *FN*s are not dangerous since they will not need an insulin injection in a meal announcement-free AID environment. Then, $FN_{cal} = 9.7$ % and $FN_{res} = 19.9$ %: therefore, a large number of *FN*s are related to calibration and missed data management. Fortunately, if the current generation of CGM systems (e.g., DexCom G6®, [44]) had been used, the number of *FN* would have decreased: indeed, these systems are calibration-free, and they embed smart algorithms to decrease the number of missing samples.

Finally, figures 5b and 5d report the distribution of *cgm* and *der* values related to *FN*s, right when they occurred and 1, 2, and 3 hours afterwards. The derivative is, on average, close to 0, and *cgm* values are not increasing with time. Based on this analysis, several *FN*s happened when meals had small and slow postprandial peaks, possibly due to an over reaction of the feed-forward action of the SAP therapy. Moreover, a number of other *FN*s are related to meals with small carbohydrate content (i.e., on average $CHO_{FN} =$ 32 g), and the feedback action of the AID controller will likely deal with them.

4. Conclusions and future works

Nowadays, hybrid AID systems permit a very effective and safe T1D management. Nonetheless, their effectiveness can be further improved: meal announcing is a significant burden for T1D subjects that may forget or inaccurately estimate carbohydrate, leading to a performance deterioration. A promising solution is the use of an automated meal detection algorithm, the first step towards a fully-automated AID.

In this work, we extended a super–twisting-based meal detector (STMD) proposed in [15], implementing an implicit discretization and approximating the glucose derivative with a Kalman filter. We assessed the performance of the proposed STMD in a challenging real-life dataset corresponding to a 2-week length trial with 30 subjects using sensor-augmented pump control.

The algorithm achieved promising results: a recall of 70 [13]% (median [interquartile range]), a precision of 73 [26]%, and 1.4 [1.4] false positives-per-day. Moreover, an analysis of incorrect detections –performed to determine

the nature and riskiness of false detections– showed that false positives were related to rising glucose conditions, whereas false negatives occurred after calibrations, missing samples, or hypoglycemia treatments. False detections were thus related to situations with a low risk of hypoglycemia and hyperglycemia.

4.1. Work limitations and future developments

As described in section 2.3.1, in this work, we treated hypoglycemia treatments as normal CHO consumptions and considered them to calculate the performance metrics. However, we need to highlight that, when used within a meal announcement-free AID system, the presented STMD algorithm should not treat hypoglycemia treatments as meals since they will not need an insulin injection.

Moreover, as introduced both in sections 3.3.1 and 3.3.2, an extensive analysis of the impact of incorrect detections should be performed only after the insertion of the STMD algorithm in a fully-automated AID system. In fact, the real riskiness of both *FP*s and *FN*s should be assessed using the AID system without meal announcement in its entirety. For example, *FN*s could be easily handled by the feedback action of the closed-loop algorithm, while meal-like occurrences not related to CHO consumption (that were considered *FP*s in this work) could still benefit by the injection of a proper amount of insulin dose by a smart bolusing strategy.

Therefore, future works will focus on the integration of the improved version of the STMD with the other modules presented in [15] and on the performance assessment of the achieved meal announcement-free AID system in dedicated clinical trials.

Funding

This work was supported by the Spanish Ministry of Economy, Industry and Competitiveness (grant number DPI2016-78831-C2-1-R); the European Union (FEDER funds); the Agencia Estatal de Investigación (PID2019-107722RB-C21/AEI/10.13039/501100011033); the Generalitat Valenciana through FSE funds (grant number ACIF/2017/021); the Italian Ministry of Education, Universities and Research (MIUR) through the project "Learn4AP: Patient-Specific Models for an Adaptive, Fault Tolerant Artificial Pancreas" (SIR initiative, project ID: RBSI14JYM2); the MIUR under the initiative "Departments of Excellence" (Law 232/2016).

Conflict of interest statement

None

Acknowledgements

Authors want to thank anonymous reviewers for their constructive suggestions.

References

[1] J.-C. Orban, E. Van Obberghen, C. Ichai, Acute Complications of Diabetes, in: Metabolic Disorders and Critically Ill Patients, Springer International Publishing, Cham, 2018, Ch. 15, pp. 341–363. doi:10.1007/978-3-319-64010-5_15.

[2] J. Fuchs, R. Hovorka, Closed-loop control in insulin pumps for type-1 diabetes mellitus: safety and efficacy, Expert Review of Medical Devices 17 (7) (2020) 707–720. doi:10.1080/17434440.2020.1784724.

[3] V. Gingras, N. Taleb, A. Roy-Fleming, L. Legault, R. Rabasa-Lhoret, The challenges of achieving postprandial glucose control using closed-loop systems in patients with type 1 diabetes, Diabetes, Obesity and Metabolism 20 (2) (2018) 245–256. doi:10.1111/dom.13052.

[4] L. Leelarathna, P. Choudhary, E. G. Wilmot, A. Lumb, T. Street, P. Kar, S. M. Ng, Hybrid closed-loop therapy: Where are we in 2021?, Diabetes, Obesity and Metabolism 23 (3) (2021) 655–660. doi:10.1111/dom.14273.

[5] C. K. Boughton, S. Hartnell, J. M. Allen, R. Hovorka, The importance of prandial insulin bolus timing with hybrid closed-loop systems, Diabetic Medicine 36 (12) (2019) 1716–1717. doi:10.1111/dme.14116.

[6] C. Roversi, M. Vettoretti, S. Del Favero, A. Facchinetti, P. Choudhary, G. Sparacino, Impact of Carbohydrate Counting Error on Glycemic Control in Open-Loop Management of Type 1 Diabetes: Quantitative Assessment Through an in silico Trial, Journal of Diabetes Science and Technology (2021) 193229682110123doi:10.1177/19322968211012392.

[7] P. V. Commissariat, L. K. Volkening, D. A. Butler, E. Dassau, S. A. Weinzimer, L. M. Laffel, Innovative features and functionalities of an

artificial pancreas system: What do youth and parents want?, Diabetic Medicine (December 2020) (2020) 1–9. doi:10.1111/dme.14492.

[8] K. Kölle, T. Biester, S. Christiansen, A. L. Fougner, Ø. Stavdahl, Pattern Recognition Reveals Characteristic Postprandial Glucose Changes: Non-Individualized Meal Detection in Diabetes Mellitus Type 1, IEEE Journal of Biomedical and Health Informatics 24 (2) (2020) 594–602. doi:10.1109/JBHI.2019.2908897.

[9] M. Zheng, B. Ni, S. Kleinberg, Automated meal detection from continuous glucose monitor data through simulation and explanation, Journal of the American Medical Informatics Association 26 (12) (2019) 1592–1599. doi:10.1093/jamia/ocz159.

[10] J. Weimer, S. Chen, A. Peleckis, M. R. Rickels, I. Lee, Physiology-Invariant Meal Detection for Type 1 Diabetes, Diabetes Technology and Therapeutics 18 (10) (2016) 616–624. doi:10.1089/dia.2015.0266.

[11] R. Sanz, P. Garcia, J.-L. Diez, J. Bondia, Artificial Pancreas System With Unannounced Meals Based on a Disturbance Observer and Feedforward Compensation, IEEE Transactions on Control Systems Technology 29 (1) (2021) 454–460. doi:10.1109/TCST.2020.2975147.

[12] J. P. Corbett, P. Colmegna, J. Garcia-Tirado, M. D. Breton, Anticipating Meals with Behavioral Profiles in an Artificial Pancreas System - An Informed Multistage Model Predictive Control Approach, IFAC-PapersOnLine 53 (2) (2020) 16305–16310. doi:10.1016/J.IFACOL.2020.12.652.

[13] S. Lee, J. Kim, S. W. Park, S.-M. Jin, S.-M. Park, Toward a Fully Automated Artificial Pancreas System Using a Bioinspired Reinforcement Learning Design: In Silico Validation, IEEE Journal of Biomedical and Health Informatics 25 (2) (2020) 536–546. doi:10.1109/JBHI.2020.3002022.

[14] A. Haidar, J. F. Yale, L. E. Lovblom, N. Cardinez, A. Orszag, C. M. Falappa, N. Gouchie-Provencher, M. A. Tsoukas, A. El Fathi, J. Rene, D. Eldelekli, S. O. Lanctôt, D. Scarr, B. A. Perkins, Reducing the need for carbohydrate counting in type 1 diabetes using closed-loop automated insulin delivery (artificial pancreas) and empagliflozin: A randomized, controlled, non-inferiority, crossover pilot trial, Diabetes, Obesity and Metabolism 23 (6) (2021) 1272–1281. doi:10.1111/DOM.14335.

[15] I. Sala-Mira, J.-L. Díez, B. Ricarte, J. Bondia, Sliding-mode disturbance observers for an artificial pancreas without meal announcement, Journal of Process Control 78 (2019) 68–77. doi:10.1016/j.jprocont.2019.03.008.

[16] A. E. Fathi, E. Palisaitis, B. Boulet, L. Legault, A. Haidar, An Unannounced Meal Detection Module for Artificial Pancreas Control Systems, in: 2019 American Control Conference (ACC), Vol. 2019-July, IEEE, 2019, pp. 4130–4135. doi:10.23919/ACC.2019.8814932.

[17] Z. Mahmoudi, F. Cameron, N. K. Poulsen, H. Madsen, B. W. Bequette, J. B. Jørgensen, Sensor-based detection and estimation of meal carbohydrates for people with diabetes, Biomedical Signal Processing and Control 48 (2019) 12–25. doi:10.1016/J.BSPC.2018.09.012.

[18] E. Fushimi, P. Colmegna, H. De Battista, F. Garelli, R. Sánchez-Peña, Unannounced meal analysis of the ARG algorithm, Proceedings of the American Control Conference 2019-July (2019) 4740–4745. doi:10.23919/acc.2019.8814719.

[19] K. Kolle, A. L. Fougner, R. Ellingsen, S. M. Carlsen, O. Stavdahl, Feasibility of Early Meal Detection Based on Abdominal Sound, IEEE Journal of Translational Engineering in Health and Medicine 7 (2019) 1–12. doi:10.1109/JTEHM.2019.2940218.

[20] S. A. Rahman, C. Merck, Y. Huang, S. Kleinberg, Unintrusive Eating Recognition using Google Glass (2015). doi:10.4108/icst.pervasivehealth.2015.259044.

[21] L. Meneghetti, A. Facchinetti, S. D. Favero, Model-Based Detection and Classification of Insulin Pump Faults and Missed Meal Announcements in Artificial Pancreas Systems for Type 1 Diabetes Therapy, IEEE Transactions on Biomedical Engineering 68 (1) (2021) 170–180. doi:10.1109/TBME.2020.3004270.

[22] C. Ramkissoon, P. Herrero, J. Bondia, J. Vehi, Unannounced Meals in the Artificial Pancreas: Detection Using Continuous Glucose Monitoring, Sensors 18 (3) (2018) 884. doi:10.3390/s18030884.

[23] N. Paoletti, R. Holloway, H. Chen, S. A. Smolka, S. Lin, Committed Moving Horizon Estimation for Meal Detection and Estimation in Type 1 Diabetes Synthesis of probabilistic systems View project Formal Analysis of Self-Adaptive Systems View project Committed Moving Horizon

Estimation for Meal Detection and Estim, in: 2019 American Control Conference, 2019.

[24] F. Zheng, S. Bonnet, E. Villeneuve, M. Doron, A. Lepecq, F. Forbes, Unannounced Meal Detection for Artificial Pancreas Systems Using Extended Isolation Forest, in: 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), IEEE, 2020, pp. 5892–5895. doi:10.1109/EMBC44109.2020.9176856.

[25] S. Samadi, M. Rashid, K. Turksoy, J. Feng, I. Hajizadeh, N. Hobbs, C. Lazaro, M. Sevil, E. Littlejohn, A. Cinar, Automatic Detection and Estimation of Unannounced Meals for Multivariable Artificial Pancreas System, Diabetes Technology & Therapeutics 20 (3) (2018) 235–246. doi:10.1089/dia.2017.0364.
URL http://www.liebertpub.com/doi/10.1089/dia.2017.0364http://online.liebertpub.com/doi/10.1089/dia.2017.0364

[26] X. Xiong, R. Kikuuwe, S. Kamal, S. Jin, Implicit-Euler Implementation of Super-Twisting Observer and Twisting Controller for Second-Order Systems, IEEE Transactions on Circuits and Systems II: Express Briefs PP (8) (2015) 1–1. doi:10.1109/tcsii.2019.2957271.

[27] S. M. Anderson, D. Raghinaru, J. E. Pinsker, F. Boscari, E. Renard, B. A. Buckingham, R. Nimri, F. J. Doyle, S. A. Brown, P. Keith-Hynes, M. D. Breton, D. Chernavvsky, W. C. Bevier, P. K. Bradley, D. Bruttomesso, S. Del Favero, R. Calore, C. Cobelli, A. Avogaro, A. Farret, J. Place, S. Ly, T. T. andShanmugham, M. Phillip, E. Dassau, I. S. Dasanayake, C. Kollman, J. W. Lum, R. W. Beck, B. Kovatchev, for the

Control to Range Study Group, Multinational Home Use of Closed-Loop Control Is Safe and Effective, Diabetes Care 7 (39) (2016) 11143–11150.

[28] A. Levant, Robust Exact Differentiation via Sliding mode Technique, Automatica 34 (3) (1998) 379–384.

[29] J. Davila, L. Fridman, A. Levant, Second-order sliding-mode observer for mechanical systems, IEEE Transactions on Automatic Control 50 (11) (2005) 1785–1789. doi:10.1109/TAC.2005.858636.

[30] Y. Yan, X. Yu, C. Sun, Discretization behaviors of a super-twisting algorithm based sliding mode control system, in: 2015 International Workshop on Recent Advances in Sliding Modes (RASM), IEEE, 2015, pp. 1–5. doi:10.1109/RASM.2015.7154656.

[31] B. Brogliato, A. Polyakov, D. Efimov, The Implicit Discretization of the Supertwisting Sliding-Mode Control Algorithm, IEEE Transactions on Automatic Control 65 (8) (2020) 3707–3713. doi:10.1109/TAC.2019.2953091.

[32] V. Acary, O. Bonnefon, B. Brogliato, Nonsmooth Modeling and Simulation for Switched Circuits, Vol. 69 of Lecture Notes in Electrical Engineering, Springer Netherlands, Dordrecht, 2011. doi:10.1007/978-90-481-9681-4.

[33] C. Edwards, Y. Shtessel, Adaptive dual-layer super-twisting control and observation, International Journal of Control 89 (9) (2016) 1759–1766. doi:10.1080/00207179.2016.1175030.

[34] V. I. Utkin, Sliding Modes in Control and Optimization, Springer Berlin Heidelberg, Berlin, Heidelberg, 1992. doi:10.1007/978-3-642-84379-2.

[35] L. Ljung, System identification - Theory for the user, 1987. doi:10.1016/0005-1098(89)90019-8.

[36] B. Friedland, Treatment of bias in recursive filtering, IEEE Transactions on Automatic Control 14 (4) (1969) 359–367. doi:10.1109/TAC.1969.1099223.

[37] B. W. Bequette, Continuous glucose monitoring: Real-time algorithms for calibration, filtering, and alarms, Journal of Diabetes Science and Technology 4 (2) (2010) 404–418. doi:10.1177/193229681000400222.

[38] H. Shu, E. P. Simon, L. Ros, Third-order kalman filter: Tuning and steady-state performance, IEEE Signal Processing Letters 20 (11) (2013) 1082–1085. doi:10.1109/LSP.2013.2277668.

[39] T. Saito, M. Rehmsmeier, The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets, PLoS ONE 10 (3) (2015) 1–21. doi:10.1371/journal.pone.0118432.

[40] E. Villeneuve, S. Lachal, C. Desir, P.-Y. Benhamou, S. Franc, G. Charpentier, E. Huneker, M. Doron, Increasing the safety of unannounced meal detection for artificial pancreas closed-loop with patient's hourly meal schedule, in: 2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC), IEEE, 2020,

pp. 5093–5096. doi:10.1109/EMBC44109.2020.9176470.
URL https://ieeexplore.ieee.org/document/9176470/

[41] E. Palisaitis, A. El Fathi, J. E. von Oettingen, A. Haidar, L. Legault, A Meal Detection Algorithm for the Artificial Pancreas: A Randomized Controlled Clinical Trial in Adolescents With Type 1 Diabetes, Diabetes Care 44 (2) (2021) 604–606. doi:10.2337/dc20-1232.

[42] R. A. Harvey, E. Dassau, H. Zisser, D. E. Seborg, F. J. Doyle, Design of the Glucose Rate Increase Detector, Journal of Diabetes Science and Technology 8 (2) (2014) 307–320. doi:10.1177/1932296814523881.

[43] B. P. Kovatchev, M. Straume, D. J. Cox, L. S. Farhy, Risk analysis of blood glucose data: A quantitative approach to optimizing the control of Insulin Dependent Diabetes, Journal of Theoretical Medicine 3 (1) (2000) 1–10. doi:10.1080/10273660008833060.

[44] Dexcom, Online site, [accessed july, 14th 2021] (2021).
URL https://www.dexcom.com/it-IT

Table 1: List of symbols

Symbol	Description
$ig(t)$	Continuous glucose reading
$cgm(k)$	Discrete noisy glucose monitor reading
Super-twisting-based residual generator	
k_1, k_2	Gains
$\hat{F}(k)$	Estimation of the lumped disturbance
L	Upper bound of the lumped disturbance
$cgm_{ST}(k)$	Estimation of the $cgm(k)$ signal
$res(k)$	Residual signal
Kalman filter	
σ_w	Standard deviation of process noise
σ_v	Standard deviation of output noise
$K(k)$	Kalman filter Gain
$\widehat{cgm}(k)$	Estimation of $cgm(k)$ signal
$\widehat{der}(k)$	Estimation of the first derivative of $cgm(k)$
$\widehat{f}(k)$	Estimation of the second derivative of $cgm(k)$
Decision logic	
ThRes	Threshold on $res(k)$
ThDer	Threshold on the derivative of $cgm(k)$
Performance metrics	
DW	Detection window
TW_{max}	Maximum elapsed time between mealtime and detection
TW_{LD}	Time window for late detections

TP [-]	**FN [-]**	**FP [-]**
16 [10]	6 [4]	7 [3]
Recall [%]	**Precision [%]**	**F1_score [%]**
70 [13]	73 [26]	68 [16]
FP-per-day [-]	**CHO_{FN} [g]**	**DT [min]**
1.4 [1.4]	32 [32]	45 [45]

Table 2: Event detection metrics (median [iqr] among patients)

Algorithm	**Recall [%]**	**FP-per-day [-]**	**DT [min]**
LDA Ra (in Kölle *et al.* [8])	92.5 [2.0]	1.5 [0.4]	18.59 [1.84]
LDA CGM (in Kölle *et al.* [8])	89.5 [4.0]	1.41 [0.42]	11.35 [2.05]
Threshold Ra (in Kölle *et al.* [8])	74.5 [3.5]	1.47 [0.53]	30.94 [5.59]
GRID (in Kölle *et al.* [8])	21.5 [7]	2.78 [0.41]	43.93 [4.7]
PAIN (in Weimer *et al.* [10])	99 [11.26]	1.88 [0.72]	-
Dassau *et al.* (in Weimer *et al.* [10])	73.9[20.5]	1.62 [1.27]	-
Lee *et al.* (in Weimer *et al.* [10])	70.26 [20.07]	1.69 [1.21]	-
Harvey *et al.* (in Weimer *et al.* [10])	79.90 [15.10]	1.64 [1.34]	-
STMD	70 [13]	1.4 [1.4]	45 [45]

Table 3: Performances' comparison between proposed STMD algorithm and other literature works on real data (as median [iqr])