INTERNATIONAL JOURNAL OF ADAPTIVE CONTROL AND SIGNAL PROCESSING
Int. J. Adapt. Control Signal Process. 2016; **30**:173–185
Published online 28 January 2015 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/acs.2538

Robust set-membership parameter estimation of the glucose minimal model

Pau Herrero[1,*,†], Benoît Delaunay[2], Luc Jaulin[2], Pantelis Georgiou[1], Nick Oliver[3] and Christofer Toumazou[1]

[1]*Centre for Bio-Inspired Technology, Institute of Biomedical Engineering, Imperial College London, UK*
[2]*Lab-STICC, ENSTA Bretagne, France*
[3]*Charing Cross Hospital, Imperial College Hospitals NHS Trust, UK*

SUMMARY

The minimal model of glucose-insulin dynamics is currently being used in several diabetes-related applications, such as investigating the glucose metabolism, and in the developments of model predictive controllers and fault detection techniques for automatic blood glucose control (i.e., artificial pancreas). Different approaches have been proposed to identify this model, but none of them is capable of providing guaranteed robust enclosures for its parameters, something very desired in applications such as the artificial pancreas, where robustness is paramount. This paper presents a novel approach for guaranteed set-membership parameter estimation of the minimal model based on the well-renowned Set Inversion via Interval Analysis (SIVIA) algorithm. Because the computational complexity of this algorithm is the main barrier for its applicability, an efficient vectorial implementation of SIVIA was employed. Clinical data from a standard intravenous glucose tolerance test were used to prove the validity of the presented approach. Finally, Modal Interval Analysis was used to reduce the numerical overestimation due to the dependency problem of interval arithmetic and significantly speeding up the computations.

Received 7 May 2013; Revised 7 May 2014; Accepted 21 December 2014

KEY WORDS: robust set-membership; parameter estimation; glucose minimal model; artificial pancreas; fault detection

1. INTRODUCTION

The minimal model of plasma glucose and insulin kinetics has been developed and used by Bergman and coworkers since the 1970s [1] to investigate glucose metabolism *in vivo* in physiological, pathological, and epidemiological studies from a standard intravenous glucose tolerance test (IVGTT). In a typical IVGTT, blood samples are taken from a fasting subject at regular intervals of time, following a single intravenous injection of glucose. Then, these blood samples are analyzed for glucose and insulin content [2]. More recently, in the context of automatic blood glucose control, also referred to as artificial pancreas (i.e., a continuous glucose monitor, an insulin pump, and a control algorithm) [3, 4], the minimal model has been employed in different applications, such as glucose forecasting in a model predictive controller [5], fault detection [6], and as a core of a type 1 diabetes subject simulator for *in silico* testing of glucose controllers [7]. In the set-membership framework, the minimal model has been used for obtaining robust estimates of the glucose absorption profile after the ingestion of a mixed meal [8], for predicting postprandial blood glucose levels [9] and for robust fault detection in insulin pumps [10]. However, none of these works have used a guaranteed set-membership approach to identify the parameters of the minimal model. Instead, empirical

*Correspondence to: Pau Herrero, Centre for Bio-Inspired Technology, Institute of Biomedical Engineering, Imperial College, London, UK.
†E-mail: pherrero@imperial.ac.uk

knowledge was used to assign the size of the intervals associated to the model parameters. In this paper, we propose the use of an bounded-error parameter estimation technique based on interval analysis [11] to accomplish this goal. In particular, an efficient vectorial implementation of the Set Inversion via Interval Analysis (SIVIA) algorithm [12] is used. Finally, to reduce the numerical overestimation due to the dependency problem of interval arithmetics, and consequently speeding up the computations, Modal Interval Analysis [13] is employed.

2. THE GLUCOSE MINIMAL MODEL

The glucose minimal model involves two physiologic compartments: a plasma compartment and an interstitial tissue compartment. Figure 1 shows a graphical representation of the glucose minimal model, where the *k*s are rate constants characterizing either material fluxes (solid lines) or control actions (dashed lines) and D is the glucose injection.

To render the minimal model uniquely identifiable [14], it must be reparameterized, and mathematically one has

$$\dot{X}(t) = -p_2 X(t) - p_3[I(t) - I_{SS}], \text{ with } X(0) = 0, \tag{1}$$

$$\dot{G}(t) = -X(t)G(t) + p_1[G_{SS} - G(t)], \text{ with } G(0) = G_0, \tag{2}$$

where $p_1 = k_1 + k_5$, $p_2 = k_3$ and $p_3 = k_2(k_4 + k_6)$; and G_{SS} and I_{SS} are glucose and insulin steady-state concentrations. Note that insulin concentration is considered as deviation from the basal value I_{SS} and that I_{SS} and G_{SS} are end-test values, taken to be the average of the last two-three data points. Finally, to provide physiological meaning to model parameters the following reparameterization can be done

$$S_G = p_1, \tag{3}$$

$$S_I = \frac{p_3}{p_2}, \tag{4}$$

where S_G is the glucose effectiveness, which measures the ability of glucose *per se*, at basal insulin, to stimulate glucose disappearance and to inhibit endogenous production by the liver (min^{-1}); S_I is the insulin sensitivity, which measures the ability of insulin to enhance the glucose per se stimulation of its disappearance and the glucose *per se* inhibition of endogenous production (min^{-1} per unit of insulin concentration, typically $mU \cdot l^{-1}$); p_2 is the insulin action parameter (min^{-1}); and G_0 is the glucose concentration extrapolated at time 0 ($mg \cdot dl^{-1}$).

In order to estimate the minimal model parameters, different parameter estimation techniques have been employed, the most common ones being the weighted least squares technique [15] and the Bayesian method [16]. However, none of these techniques provides guaranteed estimates, which may be desired in some applications such as robust control. For this purpose, a robust set-membership parameter estimation technique is required.

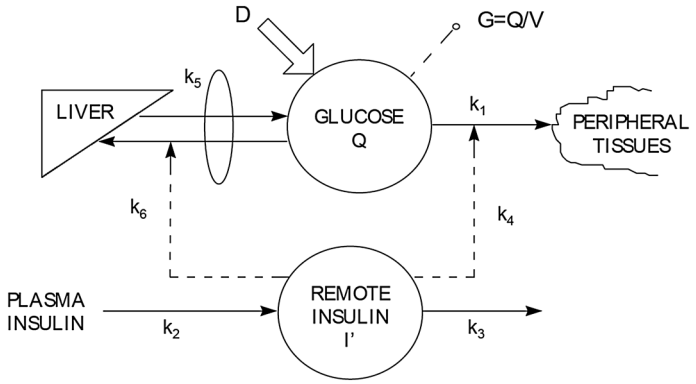


Figure 1. The glucose minimal model.

DOI: 10.1002/acs

3. PARAMETER ESTIMATION

Dynamical systems, such as the glucose minimal model stated in (1) and (2), are often modeled by differential equations, for example, an ODE

$$\frac{dy}{dt} = h(y, x, t), \tag{5}$$

where y is the output of the system, t is the time, and $x \in \mathbb{R}^n$ is the vector of parameters. In this equation, h is given while x is unknown. If possible, the integration of the ODE gives the equation

$$y(t) = f(x, t). \tag{6}$$

In most cases, observations of the system, that is, measurements, are given. Suppose that a set of data $(t_j, \tilde{y}_j)$ is known, where each $\tilde{y}_j$ is expected to be approximately $y(t_j)$. The model driven inverse problem consists in finding x such that

$$\forall j,\ \tilde{y}_j = f(x, t_j). \tag{7}$$

The problem of finding x that satisfies (7) is generally unsolvable using symbolic methods and, if the problem is non-convex, global numerical methods cannot find any reliable estimate. This problem has to be relaxed, which leads to numerical data fitting.

3.1. Weighted least squares estimation

A standard procedure to solve the problem of finding x that satisfies (7) is to minimize a cost function quantifying the distance between the output of this mathematical model y and the observed behavior of the system $\tilde{y}$ to be modeled [15]. For instance, the weighted least squares method corresponds to the minimization of the expression

$$\sum_{j=1}^{m} w_j (y(t_j) - \tilde{y}_j)^2, \tag{8}$$

where $\forall j, w_j$ are weights that are usually chosen inversely proportional to the variance of the measurement errors, so as to guarantee the optimal performance of least squares estimators. However, this approach may be weak if the objective function has multiple local minima, because the optimisation process may be very sensitive to initial values, and if one is interested in a global minimum (i.e., robust control), the entire feasible set has to be examined.

3.2. Bayesian estimation

Another common parameter identification technique used for identifying the minimal model is Bayesian estimation [16]. Bayesian parameters estimation techniques benefit over traditional methods in that an entire distribution of parameter probabilities is developed and prior knowledge of the system can be incorporated into the estimation task. This technique is especially usefully when the minimal model is identified together with other models, such as a gastrointestinal model [17]. Such additional information is entered as a mean and standard deviation for one or more of the parameters in the model. The values can come from previous individual experiments, analysis of a population, or from published results. However, this technique assumes that a prior distribution of parameter probabilities is known, which is not always the case. In addition, Bayesian estimation does not solve the problem of local minima and therefore is not suitable for guaranteed estimation of the parameters.

DOI: 10.1002/acs

3.3. Robust set-membership estimation

An alternative approach to the presented parameter estimation techniques, which allows to obtain guaranteed estimation of the parameters, is based on the hypothesis that a set of acceptable errors has been defined. The task then is to characterize the set of all values of the parameter vector such that the error remains acceptable. It is important to note that this approach requires known prior error bounds, which may not always be available.

The data driven inverse problem considers some error, which leads to the expression

$$\forall j,\ \tilde{y}_j = f(x, t_j) + e_j. \tag{9}$$

Now suppose that reliable bounds of e_j are known, namely, $a_j \leqslant e_j \leqslant b_j$. Then the problem is to find x such that:

$$\forall j,\ \tilde{y}_j - b_j \leqslant f(x, t_i) \leqslant \tilde{y}_j - a_j. \tag{10}$$

Different algorithms based on interval analysis [11] have been proposed in the context of nonlinear bounded-error estimation [18, 19], and refined using consistency techniques [20] and interval Taylor series method [21]. In this paper, we propose the use of an efficient implementation of the well-renowned SIVIA algorithm [18] to identify the minimal glucose model parameters from standard IVGTT data.

4. SET INVERSION VIA INTERVAL ANALYSIS

Let f be a function from $\mathbb{R}^n \to \mathbb{R}^p$ and let $\mathbb{Y}$ be a subset of $\mathbb{R}^p$, where $(n, p) \in \mathbb{N}^{*2}$. Set inversion is the characterization of the set defined by

$$\mathbb{X} = \{\mathbf{x} \in \mathbb{R}^n \mid f(\mathbf{x}) \in \mathbb{Y}\} = f^{-1}(\mathbb{Y}). \tag{11}$$

For any $\mathbb{Y} \subset \mathbb{R}^p$, for any function f admitting a convergent *inclusion function* $[f](.)$ from $\mathbb{IR}^n \to \mathbb{IR}^p$, being $\mathbb{IR}$ the set of real intervals [11] ; and by choosing an *inclusion test* $[t]$ defined by

$$[t](\mathbf{x}) = \begin{cases} \textit{true} & \text{if } [f]([\mathbf{x}]) \subset \mathbb{Y}, \\ \textit{false} & \text{if } [f]([\mathbf{x}]) \cap \mathbb{Y} = \emptyset, \\ \textit{undefined} & \text{otherwise,} \end{cases} \tag{12}$$

where $[x]$ is a vector of interval (an interval box).

Set Invertion via Interval Analysis algorithm [18] approximates the set defined by (11) by means of three sets of axis-aligned boxes of $\mathbb{R}^n$ $(\mathcal{S}, \mathcal{N}, \mathcal{E})$ of $\mathbb{R}^n$, also referred to as *pavings*, such that

$$\mathcal{S} \subset \mathbb{X} \subset (\mathcal{S} \cup \mathcal{E}), \tag{13}$$

$$(\mathcal{N} \cap \mathbb{X}) = \emptyset, \tag{14}$$

$$\forall [\mathbf{x}] \in \mathcal{E},\ \mathit{Width}([\mathbf{x}]) < \epsilon, \tag{15}$$

where ϵ is an arbitrary positive number that allows to control the accuracy of the approximated set and $Width$ is a real valued function that returns the maximum relative width of an interval box $[x]$ with respect to the initial box $[x_0]$, that is,

$$\mathit{Width} : [\mathbf{x}] = \bigotimes_{i \in [\![1,n]\!]} [x_i] \longmapsto \max_{i \in [\![1,n]\!]} \frac{\mathit{width}([x_i])}{\mathit{width}([x_{0_i}])}, \tag{16}$$

being *width* defined for a single interval as $\mathit{width} : [x] = [a, b] \mapsto |b - a|$. Algorithm 1 describes the classic implementation of SIVIA algorithm.

DOI: 10.1002/acs

Algorithm 1 Classic implementation of SIVIA Algorithm

Require: • $\mathbb{Y} \subset \mathbb{R}^p$ • $[\mathbf{x_0}] \in \mathbb{IR}^n$ • $[f] : \mathbb{IR}^n \to \mathbb{IR}^p$ • $\epsilon > 0$
Ensure: • $\mathcal{S}$, $\mathcal{N}$, and $\mathcal{E}$ such as • $\mathcal{S} \subset (\mathbb{X} \cap [\mathbf{x_0}]) \subset \mathcal{S} \cup \mathcal{E}$ • $\mathcal{N} \cap \mathbb{X} = \emptyset$ • $width([\mathbf{x}]) < \epsilon \ (\forall [\mathbf{x}] \in \mathcal{E})$

```
1: function SIVIA([f], 𝕐, [x0], ϵ)
2:     S ← N ← E ← ∅
3:     L ← {[x0]}
4:     while L ≠ ∅ do
5:         [x] ← pop(L)                    ▷ pop: Retrieves and removes the first interval box from a list
6:         if [f]([x]) ⊂ 𝕐 then push(S, [x])          ▷ push: Adds an interval box to a list
7:         else if [f]([x]) ∩ 𝕐 = ∅ then push(N, [x])
8:         else if Width([x]) < ϵ then push(E, [x])   ▷ Width: Returns width of largest interval
9:         else
10:            {[x1], [x2]} ← Bisect([x])   ▷ Bisect: Bisects a box and returns resulting boxes
11:            push(L, [x1])
12:            push(L, [x2])
13:        end if
14:    end while
15:    return (S, N, E)
16: end function
```

4.1. An efficient MATLAB implementation of Set Inversion via Interval Analysis

Typical programming languages used to implement SIVIA have been C++, FORTRAN 90, and ADA, which provide good computational efficiency, in terms of time and memory, and have operator overloading capabilities that allow implementing a user-friendly interval arithmetic that facilitates the writing of arithmetic expressions involving interval variables. However, these programming languages present the disadvantage of having a relatively slow learning curve and may have portability issues between different platforms, something that limits their use in many areas of science and engineering.

On the other side, high-level, numerically oriented (HLNO) programming languages such as MATLAB, SCILAB and OCTAVE are extensively used by engineers, physicists, and mathematicians because of their user-friendliness, portability, good technical support, extensive number of toolboxes, big online community of users, good documentation, and powerful data plotting tools. However, these languages are not particularly known for their computational efficiency, mainly because of their interpreted nature, and can be specially inefficient if they are not used in the way they are meant to be used, for example, using explicit *for loops* instead of array computations or built-in functions. Therefore, an efficient implementation of SIVIA in an HLNO programming language is desired to facilitate and promote its use in the scientific and engineering communities. Different attempts to implement SIVIA algorithm using HLNO programming languages have been done. The (SCS Toolbox, Technical University of Catalonia (Spain)) [22] is a MATLAB implementation of SIVIA, which is based on the interval arithmetic library INTLAB [23]. However, the computational efficiency of this implementations remains low compared to other existing implementations in C++.

The basic idea behind this novel implementation of SIVIA, which has a computational efficiency comparable to its C++ counterpart, is to evaluate all the boxes of the list $\mathcal{L}$ from Algorithm 1 in a vectorial way instead of processing them one by one. Therefore, the *inclusion test* $[t]$, *Width* and *Bisect* functions from Algorithm 1 need to be extended to their vectorial form. For this purpose, a vectorial interval arithmetic was implemented. The details of the vectorial SIVIA implementation, referred to as VSIVIA, can be found in [12]. In addition, the MATLAB code implementation with its technical documentation and including the examples presented in this paper, is available for free utilization [24].

Example 4.1

Consider the problem of a bolus intravenous injection of a drug into a patient [21]. The solution of the ODE system that models the pharmacokinetics of the drug distribution between the central

DOI: 10.1002/acs

Table I. Drug concentration data over time for a bolus intravenous injection of 800 mg of a drug into a patient.

t	0.1	0.25	0.5	0.75	1	1.5	2	2.5	3	4	6	8	10	12
$\tilde{y}$	16.1	14.3	12.0	10.3	9.0	7.2	6.1	5.2	4.6	3.7	2.5	1.7	1.18	0.81

compartment (blood) and the peripheral compartment (tissue) and the elimination from the central compartment is

$$y(t) = a \cdot e^{-\alpha \cdot t} + b \cdot e^{-\beta \cdot t}. \tag{17}$$

This expression depends on four parameters a, b, α, β, which can be used to express the distribution and elimination rates (k_{el} is the elimination rate and k_{cp}, k_{pc} are distribution rates) as follows:

$$k_{pc} = \frac{\alpha\beta + b\alpha}{a + b}, \quad k_{el} = \frac{\alpha\beta}{k_{pc}}, \quad k_{cp} = \alpha + \beta - k_{pc} - k_{el}. \tag{18}$$

For a bolus intravenous injection of 800 mg of a drug into a patient, the data for the concentration in the central compartment over a period of time are given in Table I.

Considering a relative $\pm 5\%$ error in the data $\tilde{y}$, we aim at identifying the set of parameters (a, α, b, β) such that the model expressed by (17) is consistent with the data set from Table I. Such a problem can be stated as the following set inversion problem:

$$\mathbb{X} = \left\{ (a, b, \alpha, \beta) \in ([a], [b], [\alpha], [\beta]) \mid \forall t,\ \hat{y}(t) = a \cdot e^{-\alpha \cdot t} + b \cdot e^{-\beta \cdot t} \right\}, \tag{19}$$

which can be solved by SIVIA algorithm.

Assuming that $(a, \alpha, b, \beta) \in [1, 100] \times [0, 10] \times [1, 100] \times [0, 1]$, with a $\epsilon = 10^{-3}$ (relative), VSIVIA solves the problem in 8.6 s (Intel Core 2 Duo E8500 (3.16 Ghz) – 4 GB RAM) with a total of 465,629 boxes processed. It is important to remark that the same problem solved with a classical implementation of SIVIA in MATLAB takes 59 min.

The corresponding bounding boxes for the inner and outer approximation for solution set $\mathbb{X}$ are given by:

$$Inn : \bigcup_{[\mathbf{x}] \in \mathcal{S}} [\mathbf{x}] = [8.15, 11.16] \times [0.95, 1.82] \times [6.70, 8.55] \times [0.17, 0.20], \tag{20}$$

$$Out : \bigcup_{[\mathbf{x}] \in \mathcal{E}} [\mathbf{x}] = [5.44, 11.64] \times [0.14, 2.06] \times [6.12, 11.54] \times [0.16, 1.00]. \tag{21}$$

Figure 2 shows the inner and outer envelopes corresponding to interval simulation of (17) with the inner and outer approximations obtained with vectorial Set Inversion via Interval Analysis. Table II shows the 2D projections corresponding to the obtained inner and outer approximations.

4.2. Robust Set Inversion via Interval Analysis algorithm

Set Inversion via Interval Analysis algorithm relies on the hypothesis that the prior bound for the error is correct, which is not always realistic. However, some data points may be outliers. Such outliers may for instance result from sensor failures, from an optimistic choice of the error bound and also from the fact that the model structure is unable to describe the process behavior accurately enough. The associated error should then be allowed to escape the feasible range defined by the prior bounds. Otherwise, the set $\mathbb{X}$ might become unrealistically small or even empty. One way to deal with this problem within SIVIA is the *q-relaxed intersection* [25] which consists of allowing a number of components $f_i(\mathbf{x})$ of $f(\mathbf{x})$, where $i = \{1, .., q\}$, not to fall within the feasible range of $\mathbb{Y}_i$ defined by the prior bounds, that is, such that $f(\mathbf{x})_i \notin \mathbb{Y}_i$. The *q-relaxed intersection* has been implemented in the vectorial implementation of the SIVIA algorithm.

DOI: 10.1002/acs

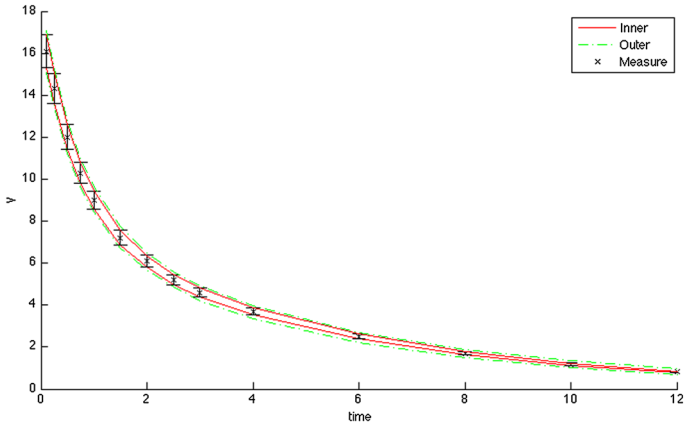
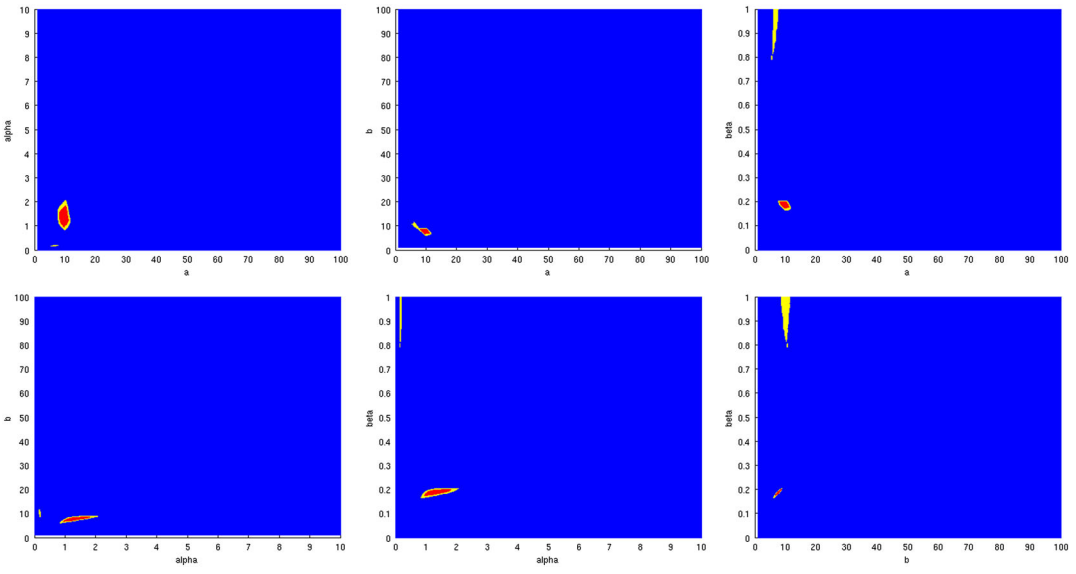


Figure 2. Inner (solid red line) and outer (dotted green line) envelopes corresponding to simulating (17) with the obtained inner and outer approximations with vectorial Set Inversion via Interval Analysis.

Table II. 2D projections of the obtained inner and outer approximations. Blue area represents the non-solution boxes ($\mathcal{N}$), yellow area are the undefined boxes ($\mathcal{E}$) and red area are the solution boxes ($\mathcal{S}$).



5. MINIMAL MODEL BOUNDED-ERROR PARAMETER ESTIMATION

The problem of identifying the parameters of the glucose minimal model [1] using data from a standard IVGTT can be expressed as the following set inversion problem:

$$\mathbb{X} = \left\{ (p_1 \times p_2 \times p_3 \times G_0) \in ([p_1] \times [p_2] \times [p_3] \times [G_0]) \mid \forall t_k,\ G(t_k) \in [\hat{G}](t_k) \right\}, \tag{22}$$

where $[\hat{G}](t_k)$ is an interval containing the plasma glucose measurement at times t_k, with $k = i, \cdots, n$, and $G(t_k)$ are the solutions at times t_k of solving the ODE system stated in (1) and (2). In order to characterize the set defined by (22) with SIVIA, an inclusion function for $G(t_k)$ is needed. Since the analytical solution of this ODE system is not available, this needs to be obtained numerically. For this purpose, a validated numerical integration method for ODEs can be used [26].

DOI: 10.1002/acs

5.1. Guaranteed ODEs integration

To get an inclusion function for $G(t_k)$, one possible approach would consist of using one of the guaranteed ODE solvers provided by interval analysis, for example, AWA [27], COSY [28] or VNODE [29]. However, these solvers are unable to provide accurate enclosures for the solutions when the parameters of the system are uncertain, as the problem being treated in this work. One approach to overcome this drawback is to bound the solutions of uncertain dynamical systems using deterministic dynamical systems. Relatively efficient guaranteed numerical integrators can then be used to compute the corresponding bounding solutions [19, 30–33].

However, these techniques still present a significant computational complexity, especially when combined with branch-and-bound techniques, for example, SIVIA algorithm, which limits their applicability to simple problems with a low number of variables. In this work, a tradeoff between complexity and reliability has been considered and only the propagation of the uncertainty associated to the initial states, model parameters, and inputs has been taken into account. This decision is supported by the fact that the observed discretization error is very small compared to the system's uncertainty (results not shown in this paper). Furthermore, we can consider any discretization/modeling error to be taken into account by the chosen bound on the a priori error.

The discrete equations corresponding to the model expressed by (22) are

$$X(k+1) = X(k) + [-p_2 X(k) - p_3[I(k) - I_{SS}]]T_s, \tag{23}$$

$$G(k+1) = G(k) + [-X(k)G(k) + p_1[G_{SS} - G(k)]]T_s, \tag{24}$$

where k indicates the current sample and T_s is the step size. Note that, despite using the same notation as in (1) and (2), variables and parameters in (23) and (24) are their interval counterparts.

5.2. Reducing interval overestimation

Interval computations have the inconvenience of overestimating the results due to the dependency problem, that is, multiple instances of variables, and due to the wrapping effect [11]. Different techniques have been proposed to reduce such overestimation [34–36]. In this work we propose reducing the dependency problem by means of the theory of Modal Interval Analysis (MIA) [37]. In particular, the *D-transformation* of the interpretable rational extension (fR) is employed. Considering a continuous real function $f(\mathbf{x})$, the *D-transformation* consists of applying the *dual operator* to any instance of a variable when the monotony sense of the function with respect to this instance is contrary to the monotony sense with respect to the variable, being the $dual$ operator defined as

$$dual([a,b]) = [b,a], \tag{25}$$

where a is the lower bound of an interval and b its upper bound. It is important to note that by applying the *D-transformation* the following inclusion is satisfied

$$[f]([\mathbf{x}]) \subseteq fR(D([\mathbf{x}])). \tag{26}$$

Example 5.1

Consider the trivial problem of computing an interval extension of the function $f(x) = x - x$, where $x \in [-1, 1]$. Note that its natural interval extension is $[f]([-1,1]) = [-1,1] - [-1,1] = [-2,2]$, which is an overestimated approximation of the range of the function due to the dependency problem. Let us now reduce such overestimation by applying the interpretable rational extension $fR([-1,1])$ of MIA. First of all, the studied function can be re-written as $f(x) = x_1 - x_2$, where the subindexes represent the different incidences of the variable x. Then, the monotony of f with respect to x and with respect to each one of its instances (x_1 and x_2), considered as different variables, is computed as follows:

$$\begin{aligned} \partial f(x)/\partial x &= 0, \\ \partial f(x)/\partial x_1 &= 1 \geqslant 0, \\ \partial f(x)/\partial x_2 &= -1 \leqslant 0. \end{aligned}$$

DOI: 10.1002/acs

Table III. Data from a standard intravenous glucose tolerance test.

t (min)	0	2	4	6	8	10	12	14	16	19	22	27
G (mg/dL)	92	350	287	251	240	216	211	205	196	192	172	163
I (mU/mL)	11	26	130	85	51	49	45	41	35	30	30	27
t (min)	32	42	52	62	72	82	92	102	122	142	162	182
G (mg/dL)	142	124	105	92	84	77	82	81	82	82	85	90
I (mU/mL)	30	22	15	15	11	10	8	11	7	8	8	7

Finally, by applying the *D-transformation* and the modal interval arithmetic, the following approximation is obtained

$$fR(D([x])) = [x_1] - dual([x_2]) = [-1, 1] - [1, -1] = [0, 0],$$

which in this particular case corresponds to the range of the function.

In order to reduce the dependency problem on (23) and (24), in addition to applying the *D-transformation*, symbolic manipulations were carried out to eliminate multiple instances of variables. Thus, the following equations were obtained

$$X(k+1) = X(k) + [[1 - p_2 T_s] + p_3[I(k) - I_{SS}]]T_s, \tag{27}$$

$$G(k+1) = \begin{cases} G(k) + [1 - [X(k)T_s + p_1]] + dual(p_1)G_{SS}T_s & \text{if}(G(k) \geqslant G_{SS}) \\ G(k) + [1 - [X(k)T_s + dual(p_1)]] + p_1 G_{SS} T_s & \text{if}(G(k) \leqslant G_{SS}) \\ G(k) + [1 - [X(k)T_s + p_1]] + p_1 G_{SS} T_s & \text{otherwise} \end{cases} \tag{28}$$

5.3. Data

Table III shows a typical response from a non-diabetic subject to a single intravenous injection of glucose taken from [38]. These data set was employed to test the proposed approach for identifying the minimal model parameters.

The selected glucose measurement error was assumed to be, whichever is larger, $\pm 4\%$ of the measured value or 4 mg/dL [39] and $\pm 1\%$ for the insulin concentration measurements [40]. The 5 initial plasma glucose and plasma insulin measurements (i.e., 8 min) of the IVGTT data were discarded to account for the single compartment approximation of glucose kinetics implemented in the minimal model [38]. Finally, the robust version of the SIVIA algorithm , that is, SIVIA + q-relax intersection was employed (Section 4.2), in order to deal with the outliers (e.g., diluted sample). In particular, three data points ($q = 3$) were allowed not to satisfy the *inclusion test*.

5.4. Results

Assuming that $(p_1, p_2, p_3, g_0) \in [0, 0.05] \times [0, 0.1] \times [0, 5e^{-5}] \times [200, 300]$, which bounds have been taken from population studied on the minimal model available in the literature [41], and a relative $\epsilon = 0.01$, VSIVIA algorithm computes in 790 s on an Intel Xeon X5650 2.66GHz–12 Cores–96GB, the inner and outer approximations of the solution set represented by the 2D projections shown in figures of Figure 3. Note that to achieve a bigger inner approximation, a smaller ϵ could be employed at the expense of increasing the computation time. In particular, a relative $\epsilon = 0.01$ was the highest one that allowed finding an inner approximation of the solution set. Being $[\mathbf{x}] = [p_1] \times [p_2] \times [p_3] \times [g_0]$, the corresponding bounding boxes of the obtained approximations are

$$Inner : \bigcup_{[\mathbf{x}] \in \mathcal{S}} [\mathbf{x}] = [0.0207, 0.0211] \times [0.0312, 0.0320] \times [0.1641e^{-4}, 0.1680e^{-4}] \times [269.5, 270.3], \tag{29}$$

$$Outer : \bigcup_{[\mathbf{x}] \in \mathcal{E}} [\mathbf{x}] = [0, 0.0375] \times [0.0023, 0.0852] \times [0.0352e^{-4}, 0.5^{-4}] \times [229.6, 300]. \tag{30}$$

DOI: 10.1002/acs

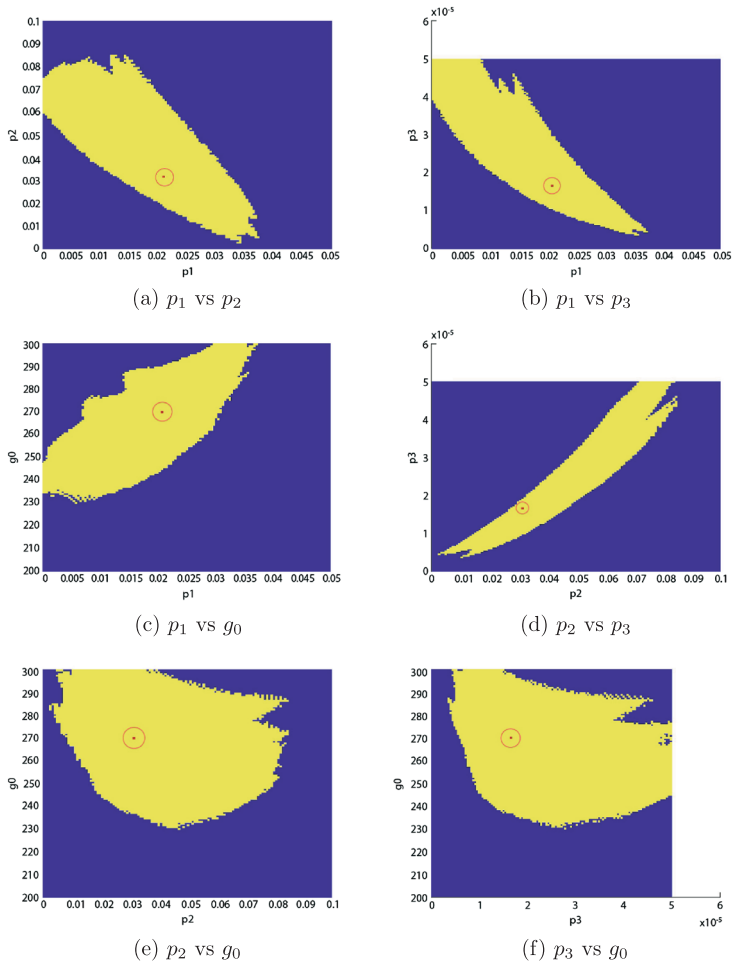


Figure 3. 2D projections of the obtained inner and outer approximations. Blue area represents the non-solution boxes ($\mathcal{N}$), yellow are the undefined boxes ($\mathcal{E}$) and red area (i.e., dot surrounded by a circle) are the solution boxes ($\mathcal{S}$).

Figure 4 shows the resulting envelope of simulating the interval boxes belonging the obtained inner and outer approximations. By applying the VSIVIA algorithm to the same problem, but without using the *D-transformation* to reduce overestimation, the resulting computation time was 2800 s. In addition, no inner approximation was achieved, and the volume of outer approximation was bigger (31).

$$Outer: \bigcup_{[\mathbf{x}]\in\mathcal{E}} [\mathbf{x}] = [0, 0.0387] \times [0, 0.0984] \times [0.0234e^{-4}, 0.5^{-4}] \times [218.7500, 300]. \tag{31}$$

As a matter of reference, the following solution was obtained for the same problem (no sensor noise considered) using the weighted least square technique parameter identification technique implemented in MLAB (Civilized Software®) [42]: $p_1 = 0.02649 \pm 0.01367$, $p_2 = 0.02543 \pm 0.02922$, $p_3 = 1.28169e^{-5} \pm 1.51621e^{-5}$, and $g_0 = 279.112 \pm 15.3880$. It is important to note that the inner approximation obtained with VSIVIA is included in the solution provided by MLAB and that the solution by MLAB is included in the outer approximation by VSIVIA. Finally, a classical MATLAB implementation of SIVIA was employed the solve the stated problem, but after 1 day of calculations no result was provided and the execution was canceled.

DOI: 10.1002/acs

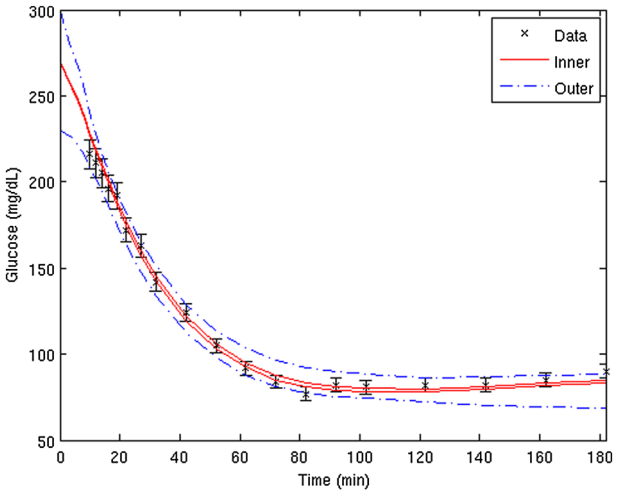


Figure 4. Resulting envelope of simulating the interval boxes belonging the inner (solid red line) and outer approximation (blue dotted line) of the solution set and intervals associated to the plasma glucose measurements (vertical bars).

6. CONCLUSIONS

Set Inversion via Interval Analysis algorithm has been proven to be a suitable tool for estimating the parameter of the glucose minimal model using data from a standard IVGTT when measurement errors and uncertainty associated to the initial are taken into account by means of interval bounds. An efficient vectorial implementation of SIVIA algorithm in MATLAB was able to approximate the model parameters in a reasonable computation time (13 min). Modal Interval Analysis has been proven to help reducing the numerical overestimation due to the dependency problem of interval arithmetics and significantly speeding up the computations of VSIVIA algorithm (from 45 min to 13 min) and improving the accuracy of the results. In a real-world setting, the presented technique would be employed offline to individualize the glucose–insulin minimal model parameters using retrospective data from an IVGTT test. Such a task could be carried out by trained clinicians using software installed on personal computer. Then, the obtained solution (i.e., interval bounds) could be used in real time applications (e.g., fault detection [10] or hypoglycemia detection [9]) embedded in hardware, such as a sensor-augmented insulin pump [43] or a handheld device (e.g., smartphone) communicating with an insulin pump and a continuous glucose sensor [44]. Finally, although the proposed robust parameter identification technique has only been used to identify the parameters of the IVGTT glucose minimal model (i.e., cold glucose kinetics), it could also be used to identify other minimal models of the glucose–insulin system such as the single and two-compartment minimal model of hot glucose kinetics and the minimal model of C-peptide and insulin kinetics.

ACKNOWLEDGEMENTS

This work has been funded by the Wellcome Trust and ENSTA-Bretagne.

REFERENCES

1. Bergman RN, Ider YZ, Bowden CR, Cobelli C. Quantitative estimation of insulin sensitivity. *American Journal of Physiology* 1979; **236**(6):E667–E677.
2. Robert H, Stefan L, Filip L, Thomas N. A simple intravenous glucose tolerance test for assessment of insulin sensitivity. *Theoretical Biology and Medical Modelling* 2011; **8**(1):12.
3. Hovorka R. Closed-loop insulin delivery: from bench to clinical practice. *Nature Reviews Endocrinology* 2011; **7**(7):385–395.
4. Reddy M, Herrero P, El-Sharkawy M, Pesl P, Jugnee N, Thomson H, Pavitt D, Toumazou C, Johnston D, Georgiou P, Oliver N. Feasibility study of a bio-inspired artificial pancreas in adults with type 1 diabetes. *Diabetes Technology and Therapeutics* 2014; **16**(9):550–557.

DOI: 10.1002/acs

5. Gillis R, Palerm CC, Zisser H, Jovanovic L, Seborg DE, Doyle FJ. Glucose estimation and prediction through meal responses using ambulatory subject data for advisory mode model predictive control. *Journal of Diabetes Science and Technology* 2007; **1**(6):825–833.
6. Vega-Hernandez O, Campos-Cornejo F, Campos-Delgado DU, Espinoza-Trejo DR. Increasing security in an artificial pancreas: Diagnosis of actuator faults. In *Health Care Exchanges, 2009. PAHCE 2009. Pan American*. IEEE: Mexico City, 2009; 137–142.
7. Kanderian SS, Weinzimer S, Voskanyan G, Steil GM. Identification of intraday metabolic profiles during closed-loop glucose control in individuals with type 1 diabetes. *Journal of Diabetes Science and Technology* 2009; **3**(5):1047–1057.
8. Herrero P, Bondia J, Palerm CC, Vehi J, Georgiou P, Oliver N, Toumazou C. A simple robust method for estimating the glucose rate of appearance from mixed meals. *Journal of Diabetes Science and Technology* 2011; **6**(1):153–162.
9. Garcia-Jaramillo M, Calm R, Bondia J, Vehi J. Prediction of postprandial blood glucose under uncertainty and intra-patient variability in type 1 diabetes: a comparative study of three interval models. *Computer Methods and Programs in Biomedicine* 2012; **108**(1):224 –233.
10. Herrero P, Calm R, Vehi J, Armengol J, Georgiou P, Oliver N, Toumazou C. Robust fault detection system for insulin pump therapy using continuous glucose monitoring. *Journal of Diabetes Science and Technology* 2012; **6**(5):1131–1141.
11. Moore R. *Interval Analysis*. Prentice-Hall: Englewood Cliffs, NJ, USA, 1966.
12. Herrero P, Delaunay B, Jaulin L, Georgiou P, Toumazou C. An efficient implementation of the sivia algorithm in a high-level numerical programming language. *Reliable Computing* 2012; **16**:224 –233.
13. Gardenyes E, Sainz MÁ, Jorba L, Calm R, Estela R, Mielgo H, Trepat A. Modal intervals. *Reliable Computing* April 2001; **7**(2):77–111.
14. Walter E. *Identifiability of State Space Models*. Springer-Verlag: Berlin, Germany, 1982.
15. Steil GM, Volund A, Kahn SE, Bergman RN. Reduced sample number for calculation of insulin sensitivity and glucose effectiveness from the minimal model. Suitability for use in population studies. *Diabetes* 1993; **42**(2):250–256.
16. Callegari T, Caumo A, Cobelli C. Generalization of map estimation in SAAM II: validation against ADAPT II in a glucose model case study. *Annals of Biomedical Engineering* 2002; **30**(7):961–968.
17. Dalla-Man C, Caumo A, Basu R, Rizza R, Toffolo G, Cobelli C. Minimal model estimation of glucose absorption and insulin sensitivity from oral test: validation with a tracer method. *American Journal of Physiology-Endocrinology and Metabolism* 2004; **287**(4):E637–E643.
18. Jaulin L, Walter E. Guaranteed nonlinear parameter estimation from bounded-error data via interval analysis. *MCS* 1993; **35**(2):123–137.
19. Kieffer M, Walter E. Guaranteed estimation of the parameters of nonlinear continuous-time models: contributions of interval analysis. *International Journal of Adaptive Control and Signal Processing* 2011; **25**(3):191–207.
20. Jaulin L. Interval constraint propagation with application to bounded-error estimation. *Automatica* 2000; **36**: 1547–1552.
21. Granvilliers L, Cruz J, Barahona P. Parameter estimation using interval computations. *SIAM Journal on Scientific Computing* 2004; **26**(2):591–612.
22. Tornil-Sin S, Puig V, Escobet T. Set computations with subpavings in MATLAB: the SCS toolbox. *IEEE International Symposium on Computer-Aided Control System Design - Part of 2010 IEEE Multi-Conference on Systems and Control*, Yokohama, Japan, 2010; 1403–1408.
23. Rump SM. INTLAB - interval laboratory. In *Developments in Reliable Computing*, Csendes Tibor (ed.). Kluwer Academic Publishers: Dordrecht, 1999; 77–104. http://www.ti3.tu-harburg.de/rump/.
24. Herrero P, Delaunay B, Jaulin L, Georgiou P, Toumazou C. VSIVIA: Vectorial set inversion via interval analysis. *Software Distribution* 2012. http://www3.imperial.ac.uk/people/p.herrero-vinias/research.
25. Jaulin L, Walter E, Didrit O. Guaranteed robust nonlinear parameter bounding. *Proceedings of CESA'96 IMACS Multiconference (Symposium on Modelling, Analysis and Simulation)*, Vol. 2, Lille, France, 1996; 1156–1161.
26. Olivier Bouissou, Alexandre Chapoutot, Adel Djoudi. Enclosing temporal evolution of dynamical systems using numerical methods. In *Nasa Formal Methods*, vol. 7871, Guillaume Brat, Neha Rungta, Arnaud Venet (eds)., *Lecture Notes in Computer Science*. Springer: Berlin Heidelberg, 2013; 108–123.
27. Lohner R. Computation of guaranteed enclosures for the solutions of ordinary initial and boundary value-problem. In *Computational Ordinary Differential Equations*, Cash JR, Gladwell I (eds). Clarendon Press: Oxford, 1992; 425–435.
28. Hoefkens J, Berz M, Makino K. Efficient high-order methods for ODEs and DAEs. In *Automatic Differentiation: From Simulation to Optimization*, Corliss G, Faure C, Griewank A (eds). Springer: New York, 2001; 341–351.
29. Nedialkov NS, Jackson KR. Methods for initial value problems for ordinary differential equations. In *Perspectives on Enclosure Methods*, Kulisch U, Lohner R, Facius A (eds). Springer: Vienna, 2001; 219–264.
30. Gouze JL, Rapaport A, Hadj-Sadok ZM. Interval observers for uncertain biological systems. *Journal of Ecological Modelling* 2000; **133**(1-2):45–56.
31. Walter E, Kieffer M. Interval analysis for guaranteed nonlinear estimation. *13th IFAC Symposium on System Identification (SYSID)*, Rotterdam, 2003; 259–270.
32. Raissi T, Ramdani N, Candau Y. Set membership state and parameter estimation for systems described by nonlinear differential equations. *Automatica* 2004; **40**(10):1771–1777.
33. Bernard O, Gouze JL. Closed loop observers bundle for uncertain biotechnological models. *Journal of Process Control* 2004; **14**:765–774.

DOI: 10.1002/acs

34. Nedialkov NS, Kreinovich V, Starks SA. Interval arithmetic, affine arithmetic, Taylor series methods: why, what next? *Numerical Algorithms* 2004; **34**:325–336.
35. Ignacio Araya, Bertrand Neveu, Gilles Trombettoni. An interval extension based on occurrence grouping. *Computing* 2012; **94**(2-4):173–188.
36. Araya I, Trombettoni G. Exploiting monotonicity in interval constraint propagation. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence, AAAI*: Georgia, USA, 2010.
37. MA Sainz, Armengol J, Calm R, Herrero P, Jorba L, Vehi J. Modal interval analysis: new tools for numerical information. In *Lecture Notes in Mathematics*, Vol. 2091. Springer: Switzerland, 2014.
38. Pacini G, Bergman RN. Minmod: a computer program to calculate insulin sensitivity and pancreatic responsivity from the frequently sampled intravenous glucose tolerance test. *Computer Methods and Programs in Biomedicine* 1986; **23**(2):113–122.
39. Yellow Spring Instruments. *YSI 2300 stat plus glucose and lactate analyser: product specifications*, 2013. Available from:http://www.ysilifesciences.com/index.php?page=specs. [Accessed on August, 2012].
40. Even MS, Sandusky CB, Barnard ND, Mistry J, Sinha MK. Development of a novel elisa for human insulin using monoclonal antibodies produced in serum-free cell culture medium. *Clinical Biochemistry* 2007; **40**(1-2):98–103.
41. Erichsen L, Agbaje OF, Luzio SD, Owens DR, Hovorka R. Population and individual minimal modeling of the frequently sampled insulin-modified intravenous glucose tolerance test. *Metabolism* 2004; **53**(10):1349–1354.
42. Daniel RK. MLAB - civilized software. In *Minimal Models for Glucose and Insulin Kinetics.* Civilized Software, 2013. http://www.civilized.com/mlabexamples/glucose.htmld/.
43. Kugelberg E. Device therapy: automatic insulin-pump suspension reduces hypoglycaemia. *Nature Reviews Endocrinology* 2013; **9**:502–502.
44. Keith-Hynes P, Guerlain S, Mize B, Hughes-Karvetski C, Khan M, McElwee-Malloy M, Kovatchev BP. Dias user interface: a patient-centric interface for mobile artificial pancreas systems. *Nature Reviews Endocrinology* 2013; **4**(6):1416–1426.

DOI: 10.1002/acs