MASTER

Optimal experiment design of a new minimal model of glucose kinetics during an oral glucose tolerance test incorporating effects of recruitment of muscle capillaries Glut-4 translocation and hepatic glucose production

Almagro Frutos, S.

Award date:
2006

Link to publication
Optimal Experiment Design of a New Minimal model of Glucose Kinetics during an Oral Glucose Tolerance Test

Incorporating effects of
recruitment of muscle capillaries
Glut-4 translocation and
hepatic glucose production

by S. Almagro Frutos

Master of Science thesis

Project period: October 2008 - September 2009
Report Number: 06A / 0
Research chair: MBS-CS

Commissioned by: Prof. Dr. Ir. P.P.J. van den Bosch (TU/e)
Prof. Dr. A.J.M. Wagenmakers
(University of Birmingham, TU/e)
Dr. Ir. N.A.W van Riel (TU/e)

Supervisors: Dr. Ir. N.A.W van Riel (TU/e)

Additional Commission members:
Dr. Ir. A.A.H. Damen (TU/e)
Optimal Experiment Design of a New Minimal model of Glucose Kinetics during an Oral Glucose Tolerance Test

Incorporating effects of recruitment of muscle capillaries Glut-4 translocation and hepatic glucose production

S. Almagro Frutos, A.J.M. Wagenmakers, P.P.J. van den Bosch, N.A.W. van Riel

Abstract

Investigating glucose kinetics under physiological conditions e.g. a meal or an oral glucose tolerance test (OGTT), is extremely valuable since diabetes mellitus is a pressing public health concern. In this study an expanded version of the glucose minimal model of Bergman has been developed and verified from a physiological and identification point of view using synthetic data produced from parameters found in literature.

In succession of the minimal model presented by [Bergman et al., Am. J. Phys. 236: E667-E677, 1979] which is widely used under non-physiological conditions, different minimal models are presented for use under physiological conditions during an OGTT e.g. [Natalucci et al., Int. Conf. IEEE EMB, Cancun, Mexico, 2003]. Natalucci's et al. minimal model is expanded to include jointly the effects of capillary recruitment, Glut-4 translocation and the production and inhibition of hepatic glucose production. The expanded minimal model incorporates a physiological model of glucose appearance from gut into blood plasma and an interstitial glucose compartment which can be measured using the microdialysis technique.

The new model has been tested in all its identifiability aspects: on a priori global identifiability by using an algorithm based on differential algebra [Audoly et al., IEEE Trans Biomed., 48: 1: 55-65, 2001] and a posteriori identifiability using model to model adjustment. Also, the sensitivity of the a priori identifiable parameters in relation with the estimated parameters has been examined showing large sensitivity for two of the six a priori parameters. Parameter estimation has been performed using Maximum likelihood estimation and the worst bias is calculated showing low bias for all estimated parameters. The oral glucose ingestion dosis has been optimised to reduce the bias for the estimated parameters and an algorithm has been designed to find the optimal sample schedule. Knowing the joint interaction between these mechanism in an novel, physiologically more detailed model and identifying the new oral glucose minimal model in all its aspects and performing optimal experiment design, is a great step forward to understand and quantify different potential causes of insulin resistance in type 2 diabetes.

Keywords: minimal compartmental model, glucose kinetics, OGTT, capillary recruitment, Glut-4 translocation, saturable receptor-mediated insulin transport, microdialysis, interstitial glucose, a priori global identifiability, a posteriori identifiability, maximum likelihood estimation, optimal experiment design

Introduction

1 Diabetes Mellitus

Diabetes mellitus is a metabolic disease characterized by varying or persistent elevated glucose levels in blood plasma. Diabetes mellitus is a pressing public health concern. In the Netherlands more than a half million people suffer from diabetes. Diabetes mellitus can be divided into two types: type 1 and type 2. Diabetes type 1 is caused by a lack of insulin
production and insulin secretion by the pancreas and type 2 diabetes is caused by decreased sensitivity of body tissues for insulin. This study focuses on type 2 diabetes.

The glucose regulatory system in humans consists of releasing insulin by the pancreatic β cells when the glucose concentration in blood rises above the normal range of 3.6 to 6.1 mM. Insulin stimulates glucose uptake by skeletal muscle (and adipose tissue) and suppress glucose production by the liver (hepatic glucose production). Insulin stimulates glucose uptake by skeletal muscle via two sequential mechanisms: recruitment of capillaries surrounding muscle fibers and glucose transporter-4 (Glut-4) translocation.

The second mechanism depends on the access of glucose and insulin to skeletal muscle. Glucose uptake by human skeletal muscle is regulated by insulin concentration in the interstitium [H. Herkner 2003]. Insulin is transported from blood plasma through the endothelium to reach the interstitium. After insulin has reached the interstitium, insulin is cleared due to internalization and degradation of insulin by muscle cells: Insulin binds to the insulin receptor in the interstitium which is the initiating step in insulin signalling pathways as discussed.

Two hypotheses have been proposed for the first mechanism: capillary recruitment and insulin-mediated increase of blood flow [Bergman 2003 and references in there]. In skeletal muscles two vascular flow routes exits: one route by nutritive capillaries that surround the skeletal muscles fibres responsible for the delivery of glucose and insulin and a second route by non-nutritive capillaries that has no contact with the skeletal muscles fibres. The first hypothesis suggests that insulin controls the blood flow to skeletal muscle at rest and during exercise by varying blood flow between the nutritive capillaries and the non-nutritive capillaries. It is believed that this process is very fast, 5 minutes after increase of insulin [Vincent 2004, Iwashita 2001, Porter 1997] and at insulin concentrations lower than required for the second hypothesis [Zhang 2004]. Changes in the distribution pattern can by caused by insulin resistance of the endothelial cells. Several studies [Clerk 2006, Gudbjörnsdóttir 2005, Gudbjörnsdóttir 2003] show an increase of capillary recruitment (increased permeability-surface area product (PS)) during enhanced physiological insulin concentration and a reduced capillary recruitment (lower PS) in type 2 diabetes humans than in healthy (obese) humans. The second hypothesis suggests that insulin acts as a vasodilator (alternating the blood flow at regular intervals through different capillary groups) controlling the access of glucose and insulin to skeletal muscle. Insulin binds to the insulin receptor on the luminal endothelial cells and this activates nitric oxide synthase (NOS). In response, endothelial cells release the vasodilator nitric oxide (NO) that prevents the smooth muscles cells of the precapillary arteriole to contract. This effect is more apparent at higher insulin concentration (pharmacologic concentrations) and after several hours. Impaired vasodilation can by due to insulin resistance. No increase in blood flow is seen during physiological insulin concentrations in obese healthy humans and type 2 diabetic humans [Bergman 2003 and reference in there, Gudbjörnsdóttir 2005, Gudbjörnsdóttir 2003]. However, [Clerk 2006] shows an increase during physiological insulin concentrations in lean healthy humans as well as in obese healthy humans. Insulin resistance can also have a more physical-chemical reason like a highly impermeable endothelium or reduced capillary density.

Several studies have adressed the mechanism of insulin transport from blood plasma through the endothelium to reach the interstitium by measuring glucose and insulin concentrations in blood plasma and interstitium. These studies result in conflicting conclusions either diffusion transport or a saturable receptor- mediated transport. These conflicting results have been analysed here (appendix A). It is concluded that a saturable
receptor-mediated insulin transport is considered when insulin concentrations are in a physiological range (up to 600 pM) whereas a non-saturable insulin transport is considered at pharmacologic insulin concentrations (+600 pM).

The second mechanism by which insulin stimulates glucose clearance is by increased uptake by myocytes. The transport of glucose through the muscle cell membrane is facilitated by a special carrier mediated protein called Glut-4. In the early morning fasting state only a small portion of the total amount of Glut-4 is located in the outer muscle cell membrane (insulin independent). In response to the binding of insulin to the insulin receptor in the interstitium, triggering a cascade of protein phosphorylation reactions that ultimately activate the translocation of Glut-4 stored in microsomal membranes are transported to the outer muscle cell membrane. (insulin dependent). [Lauritzen 2006] shows a time delay of 10 minutes between insulin leaving the interstitium and the first translocations of Glut-4 after intravenous insulin infusion. Maximum translocation of Glut-4 was reached after 30 minutes.

Three impairments exist in the above described mechanisms for type 2 diabetic humans: Impairment of Glut-4 translocation, impairment of the recruitment of muscle capillaries, and impairment of the inhibition of hepatic glucose production [Zijlstra 2004]. The contribution of each of these impairments is not clear, since only an overall insulin sensitivity ($S_I$), the efficacy of insulin to glucose uptake by body tissues $[1/(\text{pM} \cdot \text{min})]$, is used in experiments to quantify the whole insulin resistance.

The goal of this study is to develop a method to understand and quantify different potential causes of insulin resistance in type 2 diabetes under normal physiological conditions. For this purpose a novel model is to be developed and to be verified. In general the model will incorporate the main glucose regulating mechanism to obtain descriptions of the variables involved and to enable estimation of physiologically relevant parameters using measured data of glucose profiles, insulin profiles or other profiles of processes in a human.

The model should meet all the conditions for valid use in widespread clinical use. The oral glucose minimal model must describe the mechanism by which insulin regulates glucose uptake by muscle cells or adipose tissues as best as possible. Simplification of the model leads to undermodelling of the glucose regulatory system giving erroneous estimates. Overmodelling leads to complex model which are difficult to identify. The data from clinical measurements used for the novel model must be obtained during normal life conditions, the possibilities for invasive measurements in humans are limited and the costs of the measurements must be kept under a reasonable financial budget. Furthermore, the model must provide reliable unique estimates for the model parameters which describe the effects of Glut-4 translocation, the recruitment of muscle capillaries, and impairment of the inhibition of hepatic glucose production with high precision using the measurements techniques available, in other words the model needs to be identifiable for those parameters.

The first part of this report consists of modelling glucose kinetics in humans. First the mechanisms of glucose uptake by skeletal muscle and its possible impairments were discussed in this section. Further a closer analyse was made of the mechanism of transport of insulin from blood plasma through the endothelium to reach the interstitium list in appendix D. Secondly, several minimal models modelling the rate of appearance of glucose from the gut in the blood plasma are discussed. Thirdly, the new oral glucose minimal model is presented with is modifications compared to the classical minimal model of [Bergman 1979]. Finally the new oral glucose minimal model is represented in state space and the plasma
glucose, interstitial glucose, interstitial insulin and the rate of appearance of glucose are simulated using measured insulin profiles. A different approach could have been to use a model to simulate insulin (insulin profiles) that is secreted by pancreatic \( \beta \)-cells via the liver into the portal vein. A study discussing the model is list in appendix B.

The second part of this report consists of parameter estimation to identify the unknown parameters in the glucose metabolism model as optimal as possible. First, the glucose metabolism model is tested on a priori global identifiability and a posteriori identifiability. Secondly, the sensitivity of a priori parameters in relation with the estimated parameters is examined. Thirdly, the parameter estimation algorithm used is explained and also a technique is presented to calculate a lower bound of the precision that can be achieved for each parameter when estimating the parameters. Fourthly, an algorithm is designed to optimize the glucose input doses and an algorithm is designed to find the optimal sampling scheduling. Then the results are shown. Finally, a discussion is made and suggestions for future research are given.

2 Minimal Models

\( S_I \) can be estimated in experiments with the Clamp technique, with the intravenous glucose tolerance test (IVGTT), with the meal oral glucose tolerance test (MGTT) or with the oral glucose tolerance test (OGTT). The Clamp technique consists of intravenous injecting a known insulin concentration together with an infusion of glucose to control the glucose concentration in blood plasma to a fixed value. The ratio of injected insulin and infused glucose provides a measure of insulin sensitivity. The IVGTT consists of injecting a known glucose concentration into the blood plasma. The glucose and insulin concentration in blood plasma are measured in time. The minimal model [R.N. Bergman 1979] is used to estimate parameters from which \( S_I \) is calculated. The CLAMP technique as well as the IVGTT use non-normal life conditions. The MGTT and OGTI are tests using normal life conditions. The MGTT uses an ingested meal containing glucose and the OGTI uses an ingested liquid containing glucose. The glucose and insulin profiles in blood plasma are measured. Also less surgical intervention is needed (only arterial and venous blood samples) than during a CLAMP or IVGTT. The minimal model of Bergman needs to be adapted to incorporate the rate of appearance of glucose from the gut in the blood plasma / systemic circulation of nutrient taken by mouth (\( Ra(t) \)).

In the past two mathematical models without physiological meaning have successfully been used to estimate \( Ra(t) \). The “integral equation” method is based on the assumption that \( Ra(t) \) is an anticipated version of blood plasma glucose concentrations [A. Caumo 2000]. The method calculates the areas under the measured glucose and insulin blood plasma concentrations. To properly calculate the areas, the glucose and insulin concentration have to return to basal levels. This may cause problems if the “integral equation” method is applied to humans with either impaired glucose tolerance or impaired insulin secretion, in which the return of glucose and insulin to the pretest levels requires a rather long time. Compared to the Bergman minimal model for an IVGTT, only two additional parameters need to be estimated. The “differential equation” method uses a piecewise-linear parametric model of \( Ra(t) \) [Dalla Man 5-2002]. An advantage of this method is that glucose and insulin concentrations do not have to return to basal value at the end of the test, hence, the experiment is dramatically shortened. Four additional parameters have to be estimated.
Two mathematical models with physiological meaning have been used successfully to estimate $Ra(t)$. [D.R.L. Worthington 1997] shows a physiological model which was tested on a type I diabetic humans. A one compartmental method was proposed, having the following physiological parameters: fraction content of glucose in the food normalized to glucose distribution volume, time constant of gastric emptying and the transport delay between ingestion of glucose and appearance of glucose in blood plasma. Advantages of this method are that only three additional parameters have to be estimated and the parameters are physiological.

An interesting physiological model is studied by [J.D. Elashoff 1982]. This model is used in this study. The model uses the exponential curve which follows a set of physiological and mathematical characteristics. The most important mathematical characteristic is that at time zero the fraction of glucose ingested ($DS\ [g]$) is 1.0 and at infinite time the ingested fraction is 0.0. Further, the power exponential curve ($t$) also allows to have a slow initial emptying followed by an accelerated emptying ($\beta>1$). If $\beta<1$ the opposite occurs. This phenomenon is often seen in humans who have undergone an operation. If $\beta=1$ the exponential curve is obtained. Physiologically, the model uses the inverse of the time constant of gastric emptying ($k\ \text{[1/min]}$) and $\beta$.

$$f(t) = e^{-k_{1}t^\beta} \quad (1)$$

Data of older studies obtained by measuring the contents of different ingested meals (e.g. glucose contents) and taken samples of the contents of stomach and using by different healthy humans as well as by healthy humans with operated stomach were fit to (1) using nonlinear least squares regression. This model showed high goodness of fit with the measured data.

Another study [J. Schirra 1996] performed an experiment with eight healthy non-obese males using this power exponential curve (1). The experiment used 50 or 100 grams glucose dissolved in 400 ml water and drank in 2 minutes showing $k$'s and $\beta$'s with a very high precision.

Natalucci coupled the mono compartmental model [S. Natalucci 2003] using the power exponential curve (1) with the adapted minimal model [Bergman 1979] for an OGTT. Figure 1 shows this model were $G_{\text{gut}}(t) \ [\text{t/kg}]$ is the mass of glucose in the gut and $K_{\text{abs}} \ [\text{t/min}]$ is the rate constant of glucose absorption from the gut into blood plasma. The model was tested on nine non diabetic humans using a doses of 75 grams of glucose ($DS$) dissolved in water. However, the parameters of the power exponential curve were not estimated, but the values $\beta=1.23$ and $k=0.014 \ [\text{t/min}]$ were taken from [J. Schirra 1996]. The experiment showed $S_1$ and $K_{\text{abs}}$ with low variance.

Figure 1: Mono compartmental model of the gut
New Oral Glucose Minimal Model

Since the contribution of the different potential impairments in the glucose regulatory system in type 2 diabetes, as stated in the introduction, is not clear yet, we propose to split up SI in the individual components \( S_{\text{cap}} \), \( S_{\text{glut-4}} \) and \( S_{\text{liver}} \). \( S_{\text{cap}} \) is the efficacy of insulin to stimulate insulin and glucose transport from blood plasma into interstitium by capillary recruitment. \( S_{\text{glut-4}} \) is the efficacy of insulin to stimulate glucose clearance in muscle and adipose tissue by GLUT-4 translocation. \( S_{\text{liver}} \) is the efficacy of insulin to inhibit hepatic production and stimulate glucose uptake by liver. The aim of this study was to modify the method developed by Bergman [R.N. Bergman 1979] to make this splitting possible.

[Zijlstra 2004] suggested a modification of the minimal model of Bergman. This modification consists of adding a (measured) interstitial glucose compartment to the classic minimal model, to model the control of glucose transport into the interstitium by insulin as discussed in the introduction. Adding this compartment could make the identification of the rate constant of transport of plasma glucose into the interstitial compartment possible. This extra compartment is not new; a remote (non-accessible, only exchange with the glucose blood plasma compartment) glucose compartment was already used during an IVGTT in [Cobelli 1999], during an OGTT [Dalla Man 10-2002] and references in there to avoid overestimation of \( S_G \) (the insulin independent glucose uptake) and underestimation of \( S_I \) seen in one Compartmental Minimal Models (1CMM) like the classic Bergman model. This extra compartment was also used in [Caumo 1993] and references in there to derive a time profile of endogenous hepatic glucose production during a labeled IVGTT. However, this extra compartment was not measured in all earlier models.

Another difference in the new oral glucose minimal model with the classic minimal model of Bergman is that inhibition of hepatic production and stimulation of glucose uptake by liver is regulated by plasma insulin instead of interstitial insulin as access to hepatocytes is immediate, due to a highly fenestrated endothelium [Bergman 2003]. Also, plasma insulin regulates the transport of insulin and glucose from plasma to interstitium as discussed in the introduction.

In the classic minimal model of Bergman \( S_G \) consists of the net rate of glucose produced by the liver and insulin independent glucose uptake by tissues like the brain. Unfortunately, those effects cannot be estimated independently. By using ingested labelled glucose, a distinction is made between ingested (labelled) glucose and the net rate of (non-labelled) glucose produced by the liver. By measuring the concentration of labelled glucose in blood plasma \( S_{\text{liver}} \) can be obtained as e.g. performed in [Dalla Man 10-2002].

The suggestions made by Zijlstra [Zijlstra 2004] together with the new suggestions made in this section, the previously discussed physiologic description of glucose appearance from the gut into blood plasma and with the saturable receptor-mediated insulin transport hypothesis (appendix D) resulted in the new oral glucose minimal model shown in figure 2.
This new model consists of four compartments $G(t)$ [mM], glucose plasma, $G'(t)$ [mM], interstitial glucose, $I(t)$ [pM] insulin plasma and $I'(t)$ [pM] interstitial insulin. $Rge$ [g/min] is the rate of gastric emptying, $DS$ [g] is the amount of glucose ingested, $G_{gut}(t)$ [I/g] is the mass of glucose in the gut and $k_{abs}$ [I/min] is the rate constant of glucose absorption from the gut into blood plasma.

Glucose $G(t)$ is transported at a rate proportional to $k_7$ [I/min] to the interstitium and cleared to muscle and adipose tissue at a rate proportional to $k_1$ [I/min]. Glucose plasma $G'(t)$ is also cleared to tissue in a non-insulin dependent way at a rate proportional to $p_6$ [I/min].

Insulin $I(t)$ is transported at a rate proportional to $k_2$ [I/min] to the interstitium. Insulin $I(t)$ controls the inhibition and stimulation of hepatic glucose uptake by liver ($k_6$ [I/(pM·min)]). Also, insulin controls the transport of plasma glucose ($k_8$ [I/(pM·min)]) and plasma insulin ($k_9$ [I/(pM)]) to the remote compartments. Interstitial insulin controls glucose uptake by muscles ($k_4$ [I/(pM·min)]) and is degraded at a rate proportional to $k_3$ [I/min].

Figure 3 shows the reparameterized model of figure 2 (model I). In model I, $p_2$ equals $k_3$ [I/min], $S_{i-cap}$ equals $k_2$·$k_9$ [I/(pM·min)], $p_4$ equals $k_7$·$k_8$ [I/(pM·min)], $p_5$ equals $k_5$ [I/min], $p_6$ equals $S_{i}$ [I/min], $S_{i-glut4}$ equals $k_4$·$k_1$ [I/(pM·min)] and $S_{i-liver}$ equals $k_6$ [I/(pM·min)].
Figure 3: Reparameterized new oral glucose minimal model (model 1) for glucose metabolism

The reparameterized new oral glucose minimal model 1 of figure 2 can be modified further by including ingested labelled glucose $GL(t)$ resulting in the new oral glucose minimal model 2 as shown in figure 4 (index $L$ denotes the labelled glucose version of a compartment). Remark: Another method could be to enrich blood plasma with labelled glucose by injecting intravenous labelled glucose resulting in the new oral glucose minimal model 3. The new oral glucose minimal model 2 can easily be modified to describe this experimental setup.

Figure 4: New reparameterized oral glucose minimal model (model 2) for glucose metabolism including labelled ingested glucose
The new oral glucose minimal model describes jointly the effects of Glut-4 translocation, the recruitment of muscle capillaries, and impairment of the inhibition of hepatic glucose production using existing minimal models in literature and modifying them using recent insights in the mechanism by which insulin regulates glucose uptake by muscle cells or adipose tissue. Knowing the joint interaction between these mechanisms is a great step forward to understand and quantify different potential causes of insulin resistance in type 2 diabetes. Furthermore, analyzing and verifying the new oral glucose minimal model in all its aspects (a priori and a posteriori identifiability, a priori parameter sensitivity) and performing optimal experiment design (optimal input and optimal sample schedule) is a great step forward in modelling glucose kinetics since to the authors knowledge it has never been applied all in one to a minimal model of glucose kinetics before.

Methods

4.1 Measurements Techniques

The data will be obtained in the near future at the School of Sport and Exercise Sciences at the University of Birmingham. Sophisticated techniques are used to measure the concentrations of plasma glucose and insulin and interstitial glucose during an OGTT. Glucose and insulin in blood plasma are measured by taking samples by inserting an intravenous cannula (BD Venflon™) into one of the antecubital veins. The subject’s arm is placed in a so-called ‘hot box’ in which the high temperature will arterialise the venous blood. The blood samples will be centrifuged at 3000g at 4°C for 10 min and stored at -70°C for later analysis of glucose and insulin.

Plasma glucose concentrations will be determined by the Cobas Mira Plus automated analyser (ABX Diagnostics, UK) with a coefficient of variation (CV) of 2% if measured in 10-fold. Plasma Insulin concentrations will be determined by a commercially available kit (DRG instruments GmbH, Germany) with a CV of 10%. Further, the interstitial glucose concentrations are measured by the microdialysis technique using custom made CMA107 microdialysis probe with a length of 3 cm, a MW cut off of 20000 and a coefficient of variation (CV) of 4%. Details of this technique are discussed in the next section.

4.1.1 Microdialysis

The microdialysis technique [Lönnroth 1987, Lönnroth 1997] consists of the diffusion of a water-soluble substance over a dialysis membrane into a perfusion medium. It enables measurement of the interstitial fluid. The inlet of the catheter is perfused by a pump with a perfusion fluid (water containing isotonic saline) at a rate of 0.667 μl/min. Exchange occurs between the perfusion fluid and the interstitial fluid. After equilibrium is reached, subsequent dialyses samples (20 μl) are collected at the outlet of the catheter. The collected sample is a mean measurement over the time needed to reach equilibrium. The samples are analyzed for interstitial glucose. Due to the permeability of the dialysis membrane, maximal 85-90% of interstitial glucose is recovered in the dialysis. The microdialysis technique is not suitable for interstitial insulin measurements since very low recovery (+/- 3%) is achieved yet [Sjöstrand 2005, Sjöstrand 1999]. Another drawback of this method is that data samples
suffer from slow dynamics: interstitial mean glucose concentrations are only available after reaching equilibrium in about 30 minutes.

4.2.1 Model Structure in State Space

The oral glucose minimal model can be represented in state space. The state space model structure and its state space measurement model of a dynamic system is shown in equations (2, 3) and (4), respectively, where \( x \) is the \( n \)-state vector, \( u \) the multi input vector, \( f \) are linear or nonlinear functions which describe the structure of the system parameterized by vector \( \theta_o \), \( g \) are nonlinear or linear functions which describe the output configuration parameterized by vector \( \theta_v \). \( y_m \) are the measurements of the \( v \)th output sampled at \( N_v \) discrete times \( t_{iv} \). \( y_t \) is the modeled \( v \)th output at time \( t \) or sampled at \( N_v \) discrete times \( t_{iv} \). \( e_v \) is the measurement error, assumed to be additive zero mean white noise with known variance \( \sigma_v^2(t) \) and \( M \) the number of measured outputs.

State equation:

\[
x(t, \theta_v) = f(x(t, \theta_v); u(t), \theta_v) \quad x(t_0, \theta_v) = x_0
\]  

Output equation:

\[
y_v(t, \theta_v) = g_v(x(t, \theta_v); \theta_v)
\]  

Measurement model:

\[
y_{tv}[t_{iv}] = y_{mv}[t_{iv} \theta_o] + e_{tv}[t_{iv}] \quad i = 1, 2, 3, ..., N_v \quad v = 1, 2, 3, ..., M
\]

This framework includes non-equidistant time samples, that not all outputs (blood plasma glucose and interstitial glucose) are sampled at the same time points and outputs measured with different accuracy. It should be noted that for the glucose minimal model two different types of input variables have to be discriminated. The orally ingested glucose can be manipulated by the experimenter and, therefore, is an independent input, which, in system theory, is commonly denoted by \( u(t) \). In our approach the measured insulin profile is also applied as an input to the minimal model. However, since this input is a variable which results from the response of the system to the independent input \( u(t) \), it is not independent.

The concept of 'dependent inputs' allows the opening of some of the feedback loops that connect the system of interest to the rest of the closed loop real system [van Riel 2006]. The model can focus on a smaller part of the system, as long as this subsystem is still integrated in its real system through the measured input signal. If the system contains a subsystem (state variables) which output can be measured, this subsystem (state variables) can be substituted by using the measured signal as input to the system (dependent input \( v(t) \)) as shown in (5). The number of states of the system is reduced. The drawback is a loss of predictive power, because the 'open-loop' model can only be used to simulate situations for which the dependent inputs have been measured in the real system. To solve the differential equations for the used integration step, the measured dependent input signal \( v(t) \) is linearly interpolated. If the dependent input(s) \( v(t) \) can be (accurately) measured in time, these signal can be used as forcing functions for these input(s).

\[
x(t, \theta_v) = f(x(t, \theta_v); u(t), v(t), \theta_v) \quad x(t_0, \theta_v) = x_0
\]
4.2.2 Simulation

The differential equations have been solved in Matlab by the differential equation solver Ode45. The relative error tolerances (RelTol) of the solutions were set four orders lower than the amplitudes of the outputs to $1 \cdot 10^{-4}$ and the absolute error tolerances (AbsTol) were set two orders smaller than RelTol to $1 \cdot 10^{-6}$.

4.3 Identifiability

The parameters of the new oral glucose minimal model 1, 2 and 3 are tested on a priori global identifiability and a posteriori identifiability to analyse which parameters can be estimated using the measured techniques discussed in section 4.1. A parameter can be estimated if its variance has an acceptable low value.

A priori global identifiability consists of analyzing the structure of the model to verify which parameters can be estimated using the measured techniques discussed in section 4.1 and using noise-free data without having any a priori information about the parameters. In contrast to a priori global identifiability, a posteriori identifiability verifies which parameters can be estimated by analysing the effects of the dimensions of the parameters, the effects of different input profiles and the effects of different sample schedules.

4.3.1 A Priori Global Identifiability

A priori global identifiability under noise-free conditions has been examined by using an algorithm based on differential algebra [Audoly 2001]. Briefly, the algorithm consists of first deriving one equation which is algebraic observable, in other words may only contain the model input and output variables and their derivatives and not contain any states or its derivatives. The input-output differential equation representation of a dynamic model has this property. Hence, the differential equation has to be derived from the state space representation of the model. In [Audoly 2001] this is done by eliminating all states in each equation of the state space representation by substituting state equations or their derivatives in each other, the so called characteristic set of the model, resulting in one equation without states, the so-called input-output relation of the model.

The differential equation of the model is identical to the input-output relation of the model. The differential equation is used to find the set of identifiable parameters by extracting the coefficients of the differential equation of the model. These coefficients are subtracted by the same coefficients which are substituted by a random numerical point in the parameter space for the unknown parameters. An example is shown below:

Example A:

Differential equation:

$$c1 \cdot c2 \cdot x(t) + c3/c2 \cdot x(t) = 0$$  \hspace{1cm} (a1)

Coefficients of the differential equation:

$$c1 \cdot c2, c3/c2$$  \hspace{1cm} (a2)
A random numerical point in the parameter space for the unknown parameters:

\[ c_1 = 12 \]
\[ c_2 = 4 \]
\[ c_3 = 36 \]

Coefficients of the differential equation subtracted by the same coefficients substituted by random numerical point in the parameter space for the unknown parameter:

\[ c_1 \cdot c_2 - 48 = 0 \] (a4)
\[ c_3 / c_2 - 9 = 0 \] (a5)

The obtained set of coefficients (example 1: (a4, a5)) of the differential equation, which are called exhaustive summary in [Audoly 2001], are solved by the Buchberger algorithm [Buchberger 2001] obtaining the Gröbner basis as shown in (a6). If this basis has a finite number of solutions, all parameter can be estimated properly. Else, non-identifiable parameters have to be estimated jointly, to obtain a finite number of solutions of the Gröbner basis. In example 1 C1 and C2/C3 are identifiable.

\[ 9c_2 - c_3 \]
\[ -9 + c_3 / c_2 \]
\[ c_1 - 48 / c_2 \] (a6)

The algorithm described in [Audoly 2001] assumes generic initials conditions for the state space model. Generic initials conditions are initial equations which are not restricted to a particular value. If non-generic initials conditions are used, parameters could not be observable in the differential equation leading to parameters which cannot be tested on identifiability using this algorithm. Hence, if non-generic initials conditions are used a manual check needs to be done to verify if unknown parameters are not observable when deriving the input-output relation of the model. If parameters vanish, the set of state equations of the state space representation needs to be enlarged by adding the polynomials evaluated at these initial conditions. In other words, the initial condition is filled in the state space representation of the model and this polynomial is added as extra equation in the state space representation of the model. The input-output relation must be recalculated.

[Saccomani 2002, Saccomani 2001] presented a modified algorithm based on [Audoly 2001] for dealing with those non-generic initial conditions. The algorithm is based on the lack of reachability / accessibility from the initial state. If the system is accessible from all initial states, the characteristic set of the model is valid. Else the characteristic set needs to be enlarged adding the polynomials evaluated at the initial conditions which lacks of accessibility. The accessibility property of a model is checked by using the accessibility Lie Algebra (6) where \( \partial / \partial x \) denotes the Jacobian matrix, \( A \) are the polynomials of the state space model which are multiplied by the states \( x \) and \( B \) are the polynomials of the state space model which are multiplied by the input \( u \) as defined in (7). A model is accessible from its initial conditions if the so-called accessibility rank condition holds for each initial condition [Sontag 1998]. The accessibility rank condition is satisfied when the linear span of all vectors functions that can be obtained starting with the \( A \) 's and taking any numbers of Lie brackets of them as shown in (8) has the same dimension as the state space model.
\[ [A, B] = \left( \frac{\partial}{\partial x} B \right) A - \left( \frac{\partial}{\partial x} A \right) B \]  
\[ x = A \cdot x + B \cdot u \]  
\[ [A, B, [A, [A, B]], [A, [A, [A, B]]], [A, [A, [A, B]]], \ldots] = [A B], [A B], [A B], [A B], \ldots \]  

**4.3.2 A posteriori Identifiability**

The model to model adjustment [Damen 2005] approach has been used to analyse the a posteriori identifiability using different input profiles under noise and noise-free conditions and under. The effect using when using different samples schedules can also be studied. Model to model adjustment consists of analyzing the error cost function (9) for combinations of parameters \( \hat{\theta} \) with different values in the parameter space where \( y_v(t_{iv}) \) are the measurements of the \( v \)th output sampled at \( N_v \) discrete times \( t_{iv} \), \( y_{vm}(t_{iv}, \theta) \) are the modeled \( v \)th output sampled at \( N_v \) discrete times \( t_{iv} \) using the parameters \( \theta_v \), \( z_v \) is the \( v \)th output weighting factor, \( \theta_i \) are the true parameters, \( N_v \) the number of samples taken for each output \( v \) and \( M \) the number of outputs.

\[
J(\hat{\theta}) = \frac{1}{N_v} \sum_{i=1}^{N_v} \sum_{v=1}^{M} z_v \left( y_v(t_{iv}, \theta) - y_{vm}(t_{iv}, \hat{\theta}) \right)^2 
\]

The 2 dimensional contour plot, plots the cost function (the model error) for two parameters for a range of values. Other parameters in the parameter space are kept fixed at the true parameter \( \theta \). If the combination of parameter is identifiable, the contour plot shows concentric ellipses centered at the supposed true values of the two parameters (global minima). The more narrow and steeper the concentric ellipse are, how more accurate the estimation of both parameters will be.

**4.3.3 A Priori Parameters Sensitivity**

Parameters which are not identifiable by the used model setup in combination with the used experimental setup have been set a priori. Unfortunately, parameters can vary among population. The parameter sensitivity shows if a non-identifiable parameter can be set a priori without losing significant precision of the parameters when the non-identifiable parameter varies significantly among population. Therefore the parameters sensitivities have been calculated by varying the the a priori parameters one by one over a fixed range with respect to the true known parameter, setting the value of the other a priori parameters fixed, estimating the corresponding unknown identifiable parameters and finally calculating the corresponding parameter covariance matrix \( \Delta(\theta) \) which will be explained in section 4.4. If the identifiable parameters are unbiased, the determinants of the obtained covariance matrices are an indication for the parameter sensitivity for variation of the a priori parameter. Else, if estimates were not unbiased, the worstcase bias technique, as stated in the last paragraph of section 4.4, has been used.
4.4 Parameter Estimation

Several algorithms exist to estimate parameters of a model, such as: Weight Least Squares estimation (WLS), Maximum Likelihood estimation (ML), Generalized Least Squares estimation and Bayesian estimation (MAP). In this study Maximum Likelihood estimation has been used since the noise introduced by the measurements techniques used (section 4.1) can be assumed to be white, Gaussian, zero mean and with variance $\sigma_v^{-2}$.

The ML criterion assumes that the measured $v$th output $y_v$ sampled at $N_v$ discrete times $t_{v,i}$ consists of the modelled $v$th output $y_{v,m}$ depending on true parameters $\theta$, and noise $e$ as shown in (10), which is different from (4) since the parameter vector has been split in a priori unidentifiable parameters which were kept fixed $\theta_{\text{fix}}$ and the identifiable parameters which have been estimated $\theta_{\text{est}}$.

\[ y_v(t_{v,i}) = y_{v,m}(t_{v,i}, \theta_{\text{fix}}, \theta_{\text{est}}) + e_v(t_{v,i}) \quad i = 1, 2, 3, \ldots, N_v, \quad v = 1, 2, 3, \ldots, M \]  

(10)

The probability density function of Gaussian noise is shown in (10).

\[ p_v(e_v(t_i)) = \frac{1}{\sqrt{2\pi\sigma_v^2}} e^{-\frac{e_v(t_i)^2}{2\sigma_v^2}} \]  

(11)

Combining equations (10) and (11) and assuming that the consecutive noise samples are independent, results in the likelihood function for the residue of the $v$th output with variance $\sigma_v^{-2}$ (12).

\[ L(y_v | \theta_{\text{est}}) = \left( \frac{1}{\sqrt{2\pi\sigma_v^2}} \right)^{N_v} e^{-\frac{1}{2\sigma_v^2} \sum_{i=1}^{N_v} (y_v(t_{v,i}) - y_{v,m}(t_{v,i}, \theta_{\text{fix}}, \theta_{\text{est}}))^2} \]  

(12)

The noise is assumed to be white, Gaussian, zero mean and with variance $\sigma_v^{-2}$. This assumption has been checked, as will be discussed in this section, after having estimated the parameters.

The likelihood function for the residues of all outputs $M$ is shown in (13).

\[ L(y | \theta_{\text{est}}) = \prod_{v=1}^{M} \left( \frac{1}{\sqrt{2\pi\sigma_v^2}} \right)^{N_v} e^{-\frac{1}{2\sigma_v^2} \sum_{v=1}^{M} \sum_{i=1}^{N_v} (y_v(t_{v,i}) - y_{v,m}(t_{v,i}, \theta_{\text{fix}}, \theta_{\text{est}}))^2} \]  

(13)

Taking the log of (13) results in the cost function for the ML (14) [Damen 2005 p. 47-62].

\[ J(\theta_{\text{est}}) = \sum_{v=1}^{N_v} \sum_{i=1}^{M} \frac{(y_v(t_{v,i}) - y_{v,m}(t_{v,i}, \theta_{\text{fix}}, \theta_{\text{est}}))^2}{2\sigma_v^2} \]  

(14)
The model parameters $\theta_{est}$ are estimated by minimizing (14) as shown in (18).

$$\theta_{est} = \arg \min_{\theta_{est}} J(\theta_{est})$$

(15)

Function minimization of (15) has been accomplished in Matlab using the non linear least square estimation algorithm lsqnonlin which uses the Medium-Scale Gauss-Newton Algorithm with Line search cubicpoly. The minimization of (15) stopped when the variation in cost function, between two subsequent iterations was less than TolFun or the absolute parameters value changed less than TolX. TolFun was set much lower than the amplitudes of the weighted outputs $z_{iv}$, to $1 \cdot 10^{-4}$ and TolX, a factor 1000 lower than the lowest true value of the estimated parameter set to $1 \cdot 10^{-10}$. The minimization algorithm also stops after a number of iterations is reached. The maximal number of iterations was set to 800.

After having estimated the parameters, the residues of the criterium $J(\theta_{est})$ must be analyzed to ensure that the assumption made earlier that the noise is Gaussian, zero mean with variance $\sigma_v^2$ is true. For white noise it holds that consecutive noise samples are independent. Its mean is zero and its autocorrelation function $\phi_v(t_{shift})$ (16) takes the form of a pulse at zero time shift and is zero elsewhere (Dirac pulse).

$$\phi_v(t_{shift}) = \frac{1}{N_v} \sum_{i=1}^{N_v} J_i(t_{iv}, \theta_{est}) \cdot J_i(t_{iv} + t_{shift}, \theta_{est})$$

(16)

The lower bound of the parameter covariance matrix $\Delta(\theta_{est})$ of the estimated model parameters $\theta_{est}$ is obtained (using the Cramer-Rao theorem) from the inverse of the Fisher information matrix $F(\theta_{est})$ [Walter 1997 p. 245-250] as shown in (17). If MLE qualifies as an unbiased and minimum variance estimator, the lower bound in (17) becomes an equality.

An estimator is unbiased if (18) holds where $E$ denotes the expectation. In addition, if the variance $\sigma_v^2$ is set to the inverse of the data covariance of the disturbance and $N_v \gg n_0$ (number of parameters) holds, MLE qualifies as an asymptotically unbiased and minimum variance estimator [Damen 2005], respectively.

$$\Delta(\theta_{est}) = F^{-1}(\theta_{est}) = \sum_{v=1}^{M} E_{y_{vm}|\theta_{est}} \left\{ \frac{\partial^2 \left( - \ln \left( L(y_{vm} | \theta_{fix}, \theta_{est}) \right) \right)}{\partial \theta_{est} \partial \theta_{est}^T} \right\}^{-1}$$

(17)

$$E(\theta_{est} - \theta) = 0 \text{ for } N_v \gg n_0$$

(18)
with $E$ the expected value and $n_0$ the number of estimated parameters. The variance $\sigma^2_\epsilon$ in (17) has been determined a posteriori. After having estimated the parameters, information is available concerning the variance $\sigma^2_\epsilon$. It is set to the squared 2-norm of the residues of $J(\theta_{est})$ (14).

If the parameters are biased ($E(\theta_{est} - \theta) \neq 0$) the parameter covariance matrix cannot be calculated using (17). In this case, the relative worst case bias is used to provide a measure of the (in)accuracy of the parameter estimations. The relative worst case bias as shown in (19) has been calculated for each identifiable parameter $\theta_{ei}$ in the parameter space by varying the initial values of each identifiable parameter one $\theta_{ei}$ by one between -50 to +50% of $\theta$ in steps of 10%, setting all other identifiable parameter fixed and estimating its value using the maximum likelihood estimation technique and adding for each estimation procedure $q$ a different sequence of noise to the synthetic data with a 1/SNR of 0%, 5% or 10%. The relative worst case bias is then the maximal difference found, among all $Q$ estimation procedures, between estimate $\theta_{est,q}$ and true value $\theta_{q,t}$ as shown in (20).

$$\text{Bias}_{\text{rel}}(\%)_{\text{est,q}} = \frac{|\theta_{\text{est,q}} - \theta|}{\theta} \cdot 100\% \quad q = 1, 2, 3 ..., Q$$

$$\text{Bias}_{\text{rel}}(\%)_{\text{est}} = \max_{\theta} \text{Bias}_{\text{rel}}(\%)_{\text{est,q}}$$

### 4.5 Optimal Experiment Design

Designing an optimal experiment for parameter identification consists primarily of selecting the inputs and outputs to make all parameters of interest in the model identifiable (section 4.3). Secondary, it consists of choosing a suitable parameter estimation algorithm, optimizing the shape of the input signal, choosing the optimal sampling schedule, choosing the type of sensors etc. to get the maximal information from the obtained data.

Since the choices of inputs and outputs were restricted due to physical accessibility, optimal experiment design has been focused on optimizing the glucose input doses and choosing the optimal sampling schedule discussed in section 4.5.1 and 4.5.2 respectively.

#### 4.5.1 Optimal Input Shaping

An optimal input can be determined by the D-optimality criterion [Walter 1997 p. 287-291] as applied in [Cobelli 1988] if the identifiable parameters are unbiased. D-optimality criterion consists of minimizing the determinant of the parameter covariance matrix, which is equivalent to maximizing the determinant of the Fisher information matrix (17) (unbiased and minimum variance estimator) for the input signal $u(t)$ using a conventional sample schedule that is intuitively used by experts. The optimal input criterion for each $v$th output is shown in (21).
If the identifiable parameters are biased, (17) cannot be used to estimate the variance of the parameters. In this case, the relative worst case bias technique has been used as measure of parameter accuracy, as discussed in the last paragraph of section 4.4.

### 4.5.2 Optimal Sample Schedule

The parameter covariance matrix $\Delta(\theta_{est})$ also depends on the used sample schedule according to the inverse of the Fisher information matrix (17).

The optimal sample schedule for each output is determined by maximizing the determinant of the Fisher matrix (17) (D-optimality) (unbiased and minimum variance estimator) for each sample taken ($t_v$) if the identifiable parameters are unbiased. The optimal sample schedule criterion for each $v$th output is shown in (22).

$$
\text{MIN}_{t_v} \left[ \det[\Delta(\theta_{est}, t_v)] \right] = \text{MAX}_{t_v} \left[ \det\left( \frac{N_v}{2\sigma_v^2} \sum_{v=1}^{\infty} \frac{1}{N_v} \sum_{i=1}^{N_v} \frac{\partial^2 \ln(\mathcal{L}(y_m|\theta_{est}, t_v, \theta_{est}, t_v))}{\partial \theta_{est} \partial \theta_{est}^T} \right) \right]
$$

If the parameters are biased the Fisher information matrix cannot be used. In this case the relative worst case bias, as stated in the last paragraph of section 4.4, has been used to provide the variance of the parameter estimations.

Several algorithms are based on starting with a conventional sample schedule (the sample schedule which is used normally) where each individual sample is optimized by comparing it with its neighbouring samples in the conventional sample schedule. If by taking a neighbouring sample, the determinant of the Fisher matrix is increased, this new sample is exchanged with the old one [Mori 1979], [Landaw 1982], [Xianjin 2000].
The algorithm developed in this study starts at the first sample of the conventional sample schedule searching for the first sample within a fixed horizon of samples that increases the precision. This sample is exchanged with the old one. This process is repeated until the maximal duration of the experiment is achieved. If after finishing, the optimal sample schedule contains too many samples than ethical/financial possible, the fixed horizon of samples is increased. Changing the horizon of samples is stopped when the optimal sample schedule contains the desired number of samples.

The search horizon was taken arbitrary to reduce the number of samples taken: twice the time needed to measure interstitial glucose. The optimal samples have been selected in the range (domain) of the conventional samples

The outputs do not have to be sampled at the same time points. The samples are separated by at least the time needed to take a measurement of the output. Measurement of $G(t)$ takes much less than one minute and measurement of $G'(t)$ takes thirty minutes.

### Results

#### 5.1.1 Model Structure in State Space

The new oral glucose minimal model 1, valid for insulin concentrations in a physiologic range (up to 600 pM), is a multi input multi output (MIMO) nonlinear dynamic system consisting of a four state-vector (26-29) having one independent input, one dependent input and two outputs. The independent input is $DS$ [g], the dependent input $I(t)$ [pM] and the outputs (30-31) are $G(t,\theta_0)$ [mM] and $G'(t,\theta_0)$ [mM] a mean over each 30 minutes as stated in section 4.1.2. The dependent input could become an independent input if simulated using the $\beta$-cell insulin secretion models which are list in Appendix B. The outputs are sampled at discrete time $t_i$ with $N_G$, $N_C$ samples, respectively, as shown in (32, 33) were $G_m[t_i,\theta_{pt},\theta_{ext}]$ is the modelled quantized glucose output and $G'_m[t_i,\theta_{pt},\theta_{ext}]$ is the modelled quantized mean interstitial glucose output over each 30 minutes. Due to measurement restrictions, the minimum time intervals of samples of $G(t,\theta_0)$ and $G'(t,\theta_0)$ were taken as 1 minute and 30 minutes, respectively. $e_v$ is the measurement error, assumed to be additive zero mean white noise with known variance $\sigma_v^2$ for output $v$. The remaining parameters are $G_b$ [mM] which is the basal glucose concentration, $I_b$ [pM] the basal insulin concentration, $\sigma_i^2$ the variance of the measurement noise and $\sigma_2^2$ the variance of the measurement noise.

$$
\begin{align*}
\dot{G}(t) &= -[S_{l-liver} + p4] \cdot [G(t) - G_b] \cdot [I(t) - I_b] + p5 + S_G \cdot G(t) - \frac{Kabs \cdot G_{cut}(t)}{V} \\
G(0) &= G_b \\
\dot{I}(t) &= -p2 \cdot I'(t) + S_{l-carp} \cdot [I(t) - I_b] \\
I'(0) &= 0 \\
\dot{G'}(t) &= -S_{l-glut-4} \cdot I'(t) \cdot G'(t) + p4 \cdot [G(t) - G_b] \cdot [I(t) - I_b] \\
G'(0) &= G'_b
\end{align*}
$$

(26)
\[ G_{\text{gu}}(t) = DS \cdot k \cdot \beta \cdot e^{(-k \cdot t)} - Kabs \cdot G_{\text{gu}}(t) \]
\[ G_{\text{gu}}(0) = 0 \] (29)

\[ y_1(t, P) = G(t, \theta) \] (30)

\[ y_2(t, P) = G'(t, \theta) \] (31)

\[ y_1[t_i] = G_m[t_i, \theta_{\text{fit}}, \theta_{\text{est}}] + e_i[t_i] \ i = 1, 2, 3, \ldots, N_G \] (32)

\[ y_2[t_j] = G_m'[t_j, \theta_{\text{fit}}, \theta_{\text{est}}] + e_i[t_j] \ j = 1, 2, 3, \ldots, N_G \] (33)

### 5.1.2 Simulation

Synthetic datasets are produced by simulating the model states space representation in Matlab using parameters found in literature for healthy non-obese humans and measured insulin input profile from [Wagenmakers 2006]. Sixth datasets were produced. Details of the datasets and plots used are shown in appendix C. Plots of the simulated glucose and insulin plasma and interstitial glucose data are also found in appendix C.

The parameters \( k \), \( Kabs \), \( \beta \), and \( DS \) related by the gut were estimated in [Natalucci 2003] and [Schirra 1996] as stated in section 3, the parameters \( p_5 \) and \( S_1\text{-liver} \) related by the liver were estimated by the two compartment minimal model (using a non-accessible interstitial glucose compartment, but having similar behaviour as an accessible interstitial glucose compartment) [Dalla Man 2002] and [Caumo 1993], the parameters \( p_2, p_6 \) and \( V \) were estimated by the (improved) hot IVGTT two compartment minimal model [Toffolo 2003], the parameter \( k_7 \) was estimated by [Regittinig 2003] using high frequently sampled ISF (openflow microperfusion) which uses a mathematical model [Steil 1996] to describe transendothelial transport of glucose form blood plasma to remote interstitium and finally \( k_2 \) was estimated by experimental data of [Sjöstrand 2005] using low frequently sampled ISF (microdialysis) and a mathematical model [Steil 1996] to describe transendothelial transport of insulin form blood plasma to remote interstitium. Mean values and variances are shown in table 1. No a priori information of the parameters \( k_8 \), \( k_9 \) and \( S_1\text{-glut4} \) were found in literature. Parameters \( k_8 \) and \( k_9 \) have been, to the authors knowledge, never been estimated. Since the product of \( k_7/k_2, k_8/k_9 \) and \( I(t) \) must be in the range of the transport from blood plasma to interstitium \( k_7/k_8 \), these parameters were first settled roughly, proportional to the inverse of the concentration of insulin to maintain the same range of transport. Secondly, parameters \( k_8 \) and \( k_9 \) were adjusted by trial and error producing a possible dataset of glucose kinetics. \( S_1\text{-glut4} \) was totally unknown and was set by the trial and error method just discussed.

### Table 1: Overview of the parameter value used in the new oral glucose minimal model 1 to simulate the sixth datasets (appendix C)

<table>
<thead>
<tr>
<th>Kabs</th>
<th>Beta</th>
<th>( k )</th>
<th>( \nu )</th>
<th>( k_9 )</th>
<th>( p_2 )</th>
<th>( S_1\text{-cap} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( 2.89 \cdot 10^{-3} )</td>
<td>1.23</td>
<td>0.014</td>
<td>1.386</td>
<td>( \pm 2.0 \cdot 10^{-1} )</td>
<td>0.113</td>
<td>( k_2 \cdot k_9 )</td>
</tr>
<tr>
<td>( \pm 0.02 \cdot 10^{-3} )</td>
<td>( \pm 0.09 )</td>
<td>( \pm 0.001 )</td>
<td>( \pm 3.7 \cdot 10^{-2} )</td>
<td>( (1/0.21)^{1/11} )</td>
<td>( \pm 0.023 )</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>( S_G )</th>
<th>( k_2 )</th>
<th>( k_8 )</th>
<th>( S_1\text{-glut4} )</th>
<th>( k_7 )</th>
<th>( S_1\text{-liver} )</th>
<th>( p_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0018</td>
<td>0.0395</td>
<td>3.0 \cdot 10^{-2}</td>
<td>(0.063 \cdot 10^{-7})</td>
<td>1.57 \cdot 10^{-5}</td>
<td>( k_7 \cdot k_8 )</td>
<td></td>
</tr>
<tr>
<td>( \pm 0.006 )</td>
<td>( \pm 0.008 )</td>
<td>( \pm 3.0 \cdot 10^{-2} )</td>
<td>( (1/0.21)^{1/11} )</td>
<td>( \pm 1.08 \cdot 10^{-5} )</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1 Conversion factor between [ml/min 1/\( \mu \text{U} \)] to [1/min 1/pM]: 1/7.0625. 2 Conversion factor between [dl/mg] to [1/min 1/mM]: 180.16 1/mM. 3 Ratio between distribution volume of blood plasma and interstitial fluid.
5.2 Identifiability

In the new oral glucose minimal model 1 two outputs are measured namely, measurement of plasma glucose $G(t)$ and interstitial glucose $G'(t)$. A restriction exits for measurements of interstitial glucose $G'(t)$, as only a mean measurement is available over each 30 minutes $(G'(t))$ as stated in section 4.1.2. This restriction is taken into consideration by calculating the mean of $G_m(t)$ over 30 minutes using the parameters $\theta$, in the oral minimal glucose model. The mean is calculated by summing up the values of $G_m(t)$ with intervals set to one-fifth of the smallest experimental time constant used in the model and dividing it by the number of intervals.

5.2.1 A Priori Global Identifiability

A priori global identifiability is done for the new oral glucose minimal models 1, 2 and 3. First the manual check is performed to verify if unknown parameters are not observable when dealing with non-generic initial conditions. Fortunately all parameters are observable. Table 2 shows the identifiable parameters.

<table>
<thead>
<tr>
<th>Identifiable parameter sets</th>
</tr>
</thead>
<tbody>
<tr>
<td>model 1 Ib, Gb, kabs, v, p4, and Sicap or Siglut4</td>
</tr>
<tr>
<td>model 2 Ib, Gb, kabs, v, p4, SiLiver or p5 and Sicap or Siglut4</td>
</tr>
<tr>
<td>model 3 Ib, Gb, kabs, v, p4, SiLiver or p5, Beta or k and Sicap or Siglut4</td>
</tr>
</tbody>
</table>

Table 2: Overview of the a priori global identifiability parameters for models 1, 2 and 3

A remark can be made about the identifiability of $S_1$-cap. $S_1$-cap can be identified separately if the interstitial insulin $G'(t)$ profile is measured and $p_2$ is known a priori. Unfortunately, measurements of interstitial insulin $G'(t)$ are not available. However, the interstitial insulin profile can be estimated after estimating $S_1$-glut4. In this case the variance of $S_1$-cap depends on a summation of two parts. One part is its own variance when estimating it from the measured plasma insulin profile and estimated interstitial insulin. The other part is non linear and depends on the variance of $S_1$-glut4.

5.2.2 A Posteriori Identifiability

The cost function of the new oral glucose minimal model 1 is evaluated for the plasma glucose output signal and the interstitial glucose output signal. The cost function is shown in (34).

$$J(\theta) = \frac{1}{N_G} \sum_{i=1}^{N_G} (G(t_i) - G_m(t_i, \theta))^2 + \frac{1}{N_G} \sum_{i=1}^{N_G} (G'(t_i) - G_m'(t_i, \theta))^2$$  \hspace{1cm} (34)

For each combination of the two dimensions in the parameter space, the contour plot of the error function is plot for model 1 (DS=75 gram), 2 (DS=72 gram, labelled ingested glucose 3
gram) and 3 (DS=75 gram, enrichment labelled glucose 3%) with variations of the parameter \( \theta \) of +/- 50 % using steps of 5%. The contour lines for \( J(\theta) \leq 10 \) are plot at 0.01 distance from each other. For \( J(\theta) > 10 \leq 200 \), the contour lines are plot at 5.00 distance from each other. The contour plots also include estimation results indicated by red circles. (The true value is indicated with a red plus-sign.) The a posteriori identifiability is examined for the first and the second data set described in section 5.2. / appendix C with added white noise having a \( 1/\text{SNR} \) of 0 % and 5%. Plots, not all, are shown in Appendix D.

Appendix D shows for model 1 using dataset 1, that the contourplots for all possible combinations of parameters \( kab, p, S_r\text{-glut4} \) and \( G_b \) are narrow and steep centered at the supposed true values of the two parameters (global minima). In contrast, parameter \( I_b \) shows for the same combinations of parameters in almost all case heavily stretched ellipses indicating poor-identifiable parameters. Parameter \( v \) shows stretched ellipses for the same combinations with \( kab \) and \( S_r\text{-glut4} \). \( S_r\text{-liver} \) and all other parameters not included in appendix D.1 or D.2 show heavily stretched ellipses for combinations with \( kab, p, S_r\text{-glut4} \). The same holds when a different insulin profile (dataset 2) is used.

Model 2 is examined by the model 2 model adjustment to know whether ingesting labelled glucose using the same experimental setup improves identifiability of \( S_r\text{-liver} \). However, \( S_r\text{-liver} \) show heavily stretched ellipses for combinations with \( kab, p, S_r\text{-glut4} \) and \( G_b \). Although model 3 shows less stretched ellipses than model 2 for combinations with \( kab, p, S_r\text{-glut4} \) and \( G_b, S_r\text{-liver} \) remains a poor-identifiable parameter.

Although a priori global identifiability shows identifiability of \( \beta \) and \( k \) in all possible combinations with \( kab, p, S_r\text{-glut4} \) and \( G_b \) in model 3, stretched ellipses are shown for all possible combination in appendix D.3.

In all cases that no noise is added, parameters are estimated exactly (unbiased parameters). By adding a \( 1/\text{SNR} \) of 5% white noise to the output signal, bias was introduced in the estimation of parameters. However, parameter estimations are all close to the exact value (not shown in appendix D).

### 5.2.3 A Priori Parameters Sensitivity

The a priori parameters sensitivity is calculated using the first and the second data set described in section 5.1.2 / appendix C with added white noise having a \( 1/\text{SNR} \) of 0%, 5% and 10%, respectively. The a priori parameters are varied one by one by +/- 5, 10, 20 and 30 %. The initial values of the identifiable parameters used in the estimation algorithm are set to those which showed largest bias in the relative worst case bias test in the case of a \( 1/\text{SNR} \) of 0% white noise (section 5.3). The results shown in appendix E shows the best possible variance for each identifiable parameter when the a priori parameters varies form the true value used.

Appendix E shows that for all estimated parameters, \( I_b, G_b, \beta \) and \( k \) are highly sensitivity to variations of the true value among population. Parameter \( p_2 \) only shows highly sensitivity for parameter \( S_r\text{-glut4} \). The remaining parameters \( p_5, S_c \) and \( S_r\text{-liver} \) are insensible. An exception occurs for \( S_r\text{-liver} \) which shows a quite high sensitivity when white noise is added with a \( 1/\text{SNR} \) of 10%.
5.3 Parameter Estimation

The cost function for ML is shown in (35) where $\sigma_G^2, \sigma_C^2$ are the variance of measurement errors in plasma glucose $G(t)$ and interstitial glucose $G'(t)$.

$$J(\theta, i) = \frac{1}{2\sigma_G^2} \sum_i (G(t_i) - G_m(t_i, \theta_{\text{fix}}, \theta_{\text{est}}))^2 + \frac{1}{2\sigma_C^2} \sum_i (G'(t_i) - G'_m(t_i, \theta_{\text{fix}}, \theta_{\text{est}}))^2$$  

(35)

A priori information is available about the quality of the measurements of glucose and insulin plasma and interstitial glucose specified by the manufacturer of the measurement equipment or the lab where the samples are analyzed as stated in section 4.1.

The relative worst case bias is first calculated using the first and the second data set described in section 5.1.2 / appendix C with added white noise having a $1/\text{SNR}$ of 0%, (2.5%), 5% and 10%, respectively. The initial values of the identifiable parameters used in the estimation algorithm are varied between $\pm 50\%$ using steps of 10%. The results shown in table 3, shows the relative worst case bias for each identifiable parameter.

<table>
<thead>
<tr>
<th>$1/\text{SNR}$ (%)</th>
<th>Insulin profile 1</th>
<th>Insulin profile 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$k_{\text{abs}}$</td>
<td>$v$</td>
</tr>
<tr>
<td>0</td>
<td>3.31</td>
<td>3.76</td>
</tr>
<tr>
<td>5</td>
<td>1.18</td>
<td>2.64</td>
</tr>
<tr>
<td>10</td>
<td>2.40</td>
<td>6.26</td>
</tr>
</tbody>
</table>

Table 3: Overview of the worst case bias of all identifiable parameters using synthetic data for two different insulin profiles (data sets 1 and 2, respectively, appendix b) with added white noise having a $1/\text{SNR}$ of 0%, (2.5%), 5% and 10%

Table 3 shows that all estimated parameters suffer from bias in all cases. All estimated parameters shows a low variance in all cases. (The initial values of the identifiable parameters in the case of a $1/\text{SNR}$ of 0% white noise which showed largest relative worst case bias are for the first dataset $1.4 \cdot k_{\text{abs}}, 1.4 \cdot v, 1.4 \cdot P_4, S_4$, for the second dataset $0.5 \cdot k_{\text{abs}}, 0.5 \cdot v, 0.5 \cdot P_4, 0.5 \cdot S_4$.)

Secondly, the relative worst case bias is calculated using the third, fourth, fifth and sixth data set described in section 5.1.2/ appendix C with added white noise having a $1/\text{SNR}$ of 0%, 5% and 10%, respectively. The results are shown for datasets third and fourth in table 3 and for datasets fifth and sixth in table 4.
Table 4: Overview of the worst case bias of all identifiable parameters using synthetic data for two different insulin profiles and two different sample schedules (data sets 3, 4, 5 and 6, respectively, appendix b) with added white noise having a 1/SNR of 0, 5 and 10%.

Table 4 shows for all estimated parameters a higher variance for each estimated parameter than by using the first and second dataset, but still acceptable. However, table 4 shows a lower variance for all estimated parameters in the case of a 1/SNR of 5% and 10% than by using the third and fourth dataset but still higher than by using the first and second dataset.

5.4 Optimal Experiment Design

5.4.1 Optimal Input Shaping

The optimal dosis was calculated by varying the input dose $DS$ in the range of 1 gram to 100 gram in steps of 1 gram using the first and the second data set described in 5.1.2/ appendix C with added white noise having a 1/SNR of 0%, 5% and 10%, respectively. The initial values of the identifiable parameters used in the estimation algorithm were set to those which showed largest bias in the worst biased test in the case of a 1/SNR of 0% white noise (section 5.3).

If the relative worst bias of each estimated parameter was lower than 10% and did not increase for an increase in glucose dose, this minimum glucose input was said to be optimal. The results are shown in table 5.

Table 5: Optimal dosis using synthetic data for two different insulin profiles (data sets 1, 2, respectively, appendix B) with added white noise having a 1/SNR of 0, 5 and 10%.
6 Discussion and Suggestion

The goal of this study was to investigate and validate a new oral glucose minimal model describing jointly the effects of Glut-4 translocation, the recruitment of muscle capillaries and the effects of impairments of the inhibition of hepatic glucose production using existing minimal models in literature and modifying them using recent insights in the mechanism by which insulin regulates glucose uptake by muscle cells or adipose tissues. The new oral glucose minimal model was to be identified in all its aspects (a priori and a posteriori identifiability, a priori parameter sensitivity and to perform optimal experiment design).

Three oral glucose minimal models have been developed. The first model was developed to estimate the rate constant of glucose absorption from the gut into blood plasma ($k_{abs}$), to estimate the efficacy of insulin to stimulate its own transport from blood plasma into interstitium by capillary recruitment ($S_{cap}$), to estimate the efficacy of insulin to stimulate glucose transport from blood plasma into interstitium ($p_4$), to estimate the efficacy of interstitial insulin to stimulate glucose clearance into muscle by Glut-4 translocation ($S_{glut}4$) and finally, to estimate the plasma glucose distribution volume per unit of body weight ($v$). The second model was developed to estimate the previous parameters, but also to estimate the efficacy of insulin to inhibit hepatic production and stimulate glucose uptake by liver ($S_{liver}$) by including ingested labelled glucose. The third model was an alternative for the second model since model 3 enriches blood plasma with labelled glucose through injecting labelled glucose instead of ingesting labelled glucose.

After developing the three oral glucose minimal models, a priori global identifiability analysis proved that with the models structure used, these estimations were possible. However, more information was acquired when choosing model 3, since model 3 could also estimate $\beta$, the variable of the power exponential curve denoting the velocity of emptying of the gut. This was obvious since extra data was available to estimate $v$ by the infusion of labelled glucose.

A posteriori identifiability analysis showed, for all possible combinations of parameters $k_{abs}$, $p_4$, $S_{glut}4$ and $Gb$, contour plots with narrow ellipses that were steep centered at the supposed true values of the parameters (global minima) using two different insulin profiles. Since $Gb$ could be measured, this parameter was set a priori in order to not influence negatively the parameter estimation of the other unknown parameters. Parameter $v$ showed stretched ellipses when estimated together with parameters $k_{abs}$ or $S_{glut}4$ indicating low accuracy and less reliability as was seen when the relative worst bias was calculated. A problem occurs if parameter $S_{liver}$ was estimated since all three models showed heavily stretched ellipses for combinations of $k_{abs}$, $p_4$, and $S_{glut}4$. Heavily stretched ellipses indicate non-identifiable parameters. Parameter $S_{liver}$ was non-identifiable because $p_4 >> S_{liver}$ in (26).

Further, the a priori parameters sensitivity was calculated using two different insulin profiles. All estimated parameters, $lb$, $Gb$, $\beta$ and $k$ showed highly sensitivity to variations of the true value among population. Since $lb$ and $Gb$ could be measured, this was not a problem. On the other hand, high a priori parameter sensitivity of parameters $\beta$ and $k$ was a real problem. A priori global identifiability showed identifiability of either $\beta$ or $k$, however, analyzing the a posteriori identifiability rejected this possibility. Good estimates of $k_{abs}$ and $k$ are available in literature [Schirra 1996], however, study [J.D. Elashoff 1982] shows that humans who have undergone an operation in the stomach can be modelled by changing parameter $\beta$. Hence, variations of $\beta$ among populations exist. $\beta$ or $k$ could be estimated by measuring the content of glucose in the gut or by using high(er) concentrations of labelled glucose. Parameter $p_2$ showed only highly sensitivity for parameter $S_{glut}4$. Estimates of $p_2$ are
available in literature e.g. [Toffolo 2003]. The bias of $S_{g\text{lut4}}$ could be lowered by using the open-flow microperfusion technique [Bodenlenz 2005] instead of the microdialysis technique for measurements of interstitial glucose $G_m(t)$. The open-flow microperfusion technique makes faster sampling of interstitial glucose $G_m(t)$ possible than the microdialysis technique. The remaining parameters $p_5$, $S_G$ and $S_{r\text{liver}}$ were insensible for variation of the a priori identifiable parameters. An exception occurred for $S_{r\text{liver}}$ that showed a quite high sensitivity when noise was added with a 1/SNR of 10% to the output signals. However, a maximal 1/SNR of 5% was present in the measured outputs using the microdialysis technique.

A remark could be made about the identifiability of $S_{r\text{cap}}$. $S_{r\text{cap}}$ could be identified separately if the interstitial insulin $G'(t)$ profile was measured and $p_2$ was known a priori. Unfortunately, measurements of interstitial insulin $G'(t)$ were not available. However, the interstitial insulin profile could be estimated after estimating $S_{g\text{lut4}}$. In this case the variance of $S_{r\text{cap}}$ depends on a summation of two parts. One part is its own variance after estimating it from both the measured plasma insulin profile and the estimated interstitial insulin. The other part is non-linear and depends on the variance of $S_{g\text{lut4}}$.

The studied identifiability agreed with the found relative worst case bias. A relative worst case bias of maximal 6.5% was achieved for all estimated parameters using a 1/SNR of 5% added white noise. If a 1/SNR of 10% white noise was added, a maximal relative worst case bias of 13% was achieved. If no noise was added, a maximal relative worst case bias of 3.76% was achieved indicating bad working of the minimalization algoritm used to minimize (15). Important to note is that measurements of interstitial glucose $G'(t)$ had been undermodelled.

The interstitial glucose $G'(t)$ samples were modelled by calculating the mean of $G_m(t)$ over 30 minutes by summing up the values of $G_m(t)$ with intervals set to one-fifth of the smallest experimental time constant used in the model and dividing it by the number of intervals. In fact, subsequent samples of interstitial glucose $G_m(t)$ are influenced by each other since the samples are collected at the outlet of the catheter without stopping the inlet of new interstitial glucose, causing mixing of subsequent interstitial glucose samples. This effect was not taken into account since the behaviour of mixing is still unknown.

The optimal dosis was calculated by varying the input doses $DS$ in the range of 1 gram to 100 gram in steps of 1 gram using two different insulin profiles and using sequences of added white noise with a 1/SNR of 0%, 5% and 10%. A complication occurred when $DS$ was varied. If $DS$ vary, this also implies a different insulin profile. However, the same insulin profile was used during all variations of $DS$ since an insulin profile of glucose ingestion other than 75 or 100 gram was not available. It is not common to have such an OGTT. This complication set doubts to the obtained results. This minimum glucose input was said to be optimal if the relative worst bias of each estimated parameter was lower than 10% and did not increase for an increase in glucose doses. For each 1/SNR a different optimum glucose input was obtained. For a 1/SNR of 0% of added white noise, all glucose doses were optimal. For a 1/SNR of 5% of added white noise an optimal glucose doses of 33 gram was obtained and for a 1/SNR of 10% of added white noise, 78 gram was obtained. It could be concluded that the doses of glucose should be as high as possible.

Due to limited time issues, the optimal sample schedule algorithm was not applied. Instead of this, a different test was applied: The relative worst case bias was calculated for two extra
different sample schedules (appendix C). The first extra sample schedule (the conventional sample schedule) is used as standard in many studies [Sjostrand 2005], [Wagenmakers 2006]. For the second extra sample schedule, a new sample schedule was developed which takes samples of glucose and insulin plasma during events when most changes take places in the insulin profile. This is often, during the first 60 minutes after glucose ingestion. This second extra sample schedule showed a lower variance for all estimated parameters in the case of a 1/SNR of 0% and 5% of added white noise than using the first extra sample schedule. This is obvious, because the second extra sample schedule has samples of glucose and insulin plasma during events when the most changes takes places in the insulin profile. Hence, it is very interested to find the optimal sample schedule to obtain the lowest possible relative worst bias for the estimated parameters. In both cases, the relative worst case bias was higher than the best relative worst case bias calculated in section 5.3 because fewer samples are taken in both extra sample schedules. It could be discussed if an optimal samples schedule could be found, since great variety exits among measured glucose profiles.

A remark could be made about the optimal sample schedule algorithm developed here to calculate the optimal sample schedule. The optimal sample schedule could use the Fisher information matrix as indication how optimal a samples schedule is. At the start of the algorithm a sample schedule with many samples is available which showed, as in section 5.3, low relative worst case bias. However, as the algorithm progress, the current optimum sample schedule has fewer samples than the initial sample schedule, introducing more bias in the estimation of parameters. Hence, the conditions of Fisher information matrix are not satisfied, and the relative worst case bias technique must be used instead. A drawback of the relative worst case bias technique is the processing time which is extremely large. But if the search of the optimum sample schedule progress well, the sample schedule should always show low bias and the Fisher information matrix could be used. A dilemma that must be investigated, before using the optimal sample schedule algorithm.

A different remark can be made about the choice of first calculating the optimal input doses before searching for the optimal sample schedule or quite the reverse. Since the optimal samples is selected in the range (domain) of the conventional samples, it is obvious to calculate first the optimal input doses for the conventional samples and then to reduce this set of samples obtaining the optimal sample schedule.

7 Conclusion

The goal of this study was to investigate possible impairments of insulin resistance in type 2 diabetes. Model 1 could be used to investigate whether there are impairments in Glut-4 translocation and impairments in the recruitment of muscle capillaries during physiological insulin concentrations that. The model did not take into account insulin production by the pancreas, instead of this the insulin concentration profile in blood plasma wasmeasured which insulin was produced by the pancreas. Model 1 could not discriminate impairment of hepatic glucose production.

The following parameters can be estimated using model 2 and measuring glucose blood plasma, insulin blood plasma, basal glucose level, basal insulin level and interstitial glucose using the microdialysis technique: The rate constant of glucose appearance from the gut into blood plasma, the efficacy of insulin to stimulate its own transport from blood plasma into interstitium by capillary recruitment, the efficacy of insulin to stimulate glucose transport from blood plasma into interstitium, the efficacy of interstitial insulin to stimulate glucose
clearance into muscle by Glut-4 translocation and the plasma glucose distribution volume per unit of body weight. This was the first time that these parameters were estimated jointly to understand and quantify different potential causes (mechanism of Glut-4 translocation and recruitment of muscle capillaries) of insulin resistance in type 2 diabetes during an OGTT using a physiological model of glucose appearance in blood plasma. This model refines earlier developed models.

Using a minimum glucose doses of 78 gram during an OGTT and having a 1/SNR of 10%, parameters were estimated with a maximum variance of ±13% if each minute a sample of insulin and glucose blood plasma and each 30 minutes a sample of interstitial glucose were taken during the first 240 minutes after glucose ingestion. If the 1/SNR is 5%, parameters were estimated with a maximum variance of ±6.5%. An exception holds for the parameter which denotes the efficacy of insulin to stimulate its own transport from blood plasma into interstitium by capillary recruitment which can be slight higher. This is due to the fact that this parameter also depends in a still unknown non-linear way on the precision of the efficacy of interstitial insulin to stimulate glucose clearance into muscle by Glut-4 translocation. If fewer samples are taken, it is advised to take samples when most dynamics takes places in glucose kinetics or to use the optimal sample schedule algorithm developed in this study.

All a priori identifiable parameters used in model 1 except beta and k are insensitive for the estimated parameters when having a 1/SNR of 0%, 5% and 10% of white noise and using model 1 and the experimental setup described in this study. Varying the true value of all the a priori identifiable parameters (except beta and k) found in literature with ±30% in steps of 5% resulted in a maximum variance of ±12% in the estimation of the identifiable parameters. On the other hand beta and k showed a maximum variance of respectively 241% and 195%. Although, having this high sensitivity for the a priori identifiable parameters beta and k, the description of glucose appearance with model 1 is better than earlier developed models found in literature. The earlier developed models were based on interpolating glucose appearance in blood plasma instead of using physiological parameters to model glucose appearance in blood plasma.
References


Modelling the insulin transport from blood plasma through the endothelium to reach the interstitium.

Several studies have examined the mechanism of insulin transport from blood plasma through the endothelium to reach the interstitium, resulting in conflicting outcomes: a saturable receptor-mediated or diffusion transport. A saturable receptor-mediated transport is possible since [R.S. Bar 1978] found receptors on cultured human endothelial cells obtained from human umbilical veins and [M.L. Peacock 1982] found receptors on cultured and isolated bovine endothelial cells obtained from pulmonary and systemic vessels.

Some in vivo studies have measured insulin concentrations in the interstitium using microdialysis techniques [Lonnroth 1987, Lonnroth 1997] in human skeletal muscle of conscious healthy lean humans, with no medication in (semi) supine position during an OGTT [H. Herkner 2003, M. Sjöstrand 2005] and during a CLAMP [M. Sjöstrand 1999, P.E. Jansson 1993]. Other studies have measured insulin concentration using the peripheral lymphatic cannulation technique [Castillo 1991] in hindlimb lymph of conscious healthy humans with a wide range of obesity and with no medication in (semi) supine position during a CLAMP [C. Castillo 1994] and with anaesthetized healthy dogs during a CLAMP [M. Hamilton-Wessler 2002, G. M. Steil 1996]. Another study has measured insulin concentration in thoracic duct lymph with conscious healthy dogs during a CLAMP [Y. Yang 1989]. (In all lymph based studies, lymph fluid concentration is increased by massaging or saline injection.) No difference exists between the two different measurement sites, since [Y.J. Yang 1989, C. Castillo 1994] showed strong correlation and direct proportionality between insulin concentration in the hindlimb lymph near the human skeletal muscle and glucose uptake by human skeletal muscle. Only a delay time exists between insulin in the interstitium and insulin in hindlimb lymph due to the transport time needed between both places. From this, it can be assumed that insulin concentration in interstitium and hindlimb lymph also shows strong correlation and direct proportionality.

Saturable receptor-mediated transport of insulin can be shown by an increase of insulin gradient (or decrease of insulin ratio) between blood plasma and interstitium/lymph at higher insulin concentration. After all, an increase of insulin gradient (or decrease of insulin ratio) implies a decrease in insulin transport indicating a saturable receptor-mediated transport of insulin. But since changes in clearance of insulin by muscle cells affect the gradient, it is critical to assess this parameter if an increase in the gradient is to be interpreted as a decrease in transport. However, studies [Y.J. Yang 1989, C. Castillo 1994, H. Herkner 2003] show a strong correlation between glucose uptake and insulin concentrations in interstitium/lymph concluding that no saturation occurs in clearance of insulin by muscle cells. Hence, analyzing this gradient is significant for the mechanism of insulin transport through the endothelium.

After analyzing the discussed gradient in experimental data of several in vivo studies an interesting conclusion is drawn. The studies using insulin concentrations at physiological range (up to 600 pM in blood plasma) [Y.J. Yang 1989, H. Herkner 2003, C. Castillo 1994, M. Sjöstrand 2005] show a decreased gradient indicating saturable receptor-mediated transport. [P.E. Jansson 1993] shows at insulin concentrations of 3500 pM a decreased gradient indicating saturable receptor-mediated transport. The studies [G. M. Steil 1996, M. Hamilton-Wessler 2002] using insulin concentrations at pharmacologic range (more than 5000 pM in blood plasma) show an increased gradient indicating diffusion transport. As a unification of both conclusions, a saturable receptor-mediated insulin transport is considered under physiological conditions whereas a non-saturable insulin transport is considered at pharmacologic insulin concentrations.

In vitro study [cultured bovine aortic endothelial cells with the use of dual chamber, not exposed to a flow] [L. King 1985] at physiological range concludes a receptor-mediated transport of insulin through the endothelium. Another study in vitro (endothelial cells from rat hearts) [F. Brunner 1998] at physiological range concludes a diffusion process. However, [M.A. Vincent 2005] shows a different behavior of the endothelium of rats compared with the endothelium of humans. The insulin
concentration ratio between interstitium and blood plasma is around 80% in contrast to humans 40-50% [M.A. Vincent 2005, M. Sjöstrand 1999] or dogs 35-50% [M.A. Vincent 2005]. Hence, the endothelium in rats is not a good representation of the endothelium in humans. In vitro study (cultured endothelial cells in a hollow fiber apparatus and continuously exposed to a flow) [F. Salvetti 2002] at physiological range shows a decreased gradient at higher insulin concentrations. Since these studies are all in vitro, the conclusions drawn from them are taken less serious than in vivo studies. However, the conclusions are in agreement with in vivo conclusions.

Hence, the classic minimal model of Bergman [Bergman 1979] is only valid using insulin concentrations at physiological range. This is always the case during an OGTT.

Another interesting case to study, are possible delays. A significant delay about 20 minutes in the activation phase of transporting insulin from blood plasma to interstitium is seen in vivo studies [M. Sjöstrand 2005, M. Sjöstrand 2002, C. Castillo 1994, Y.J. Yang 1989, E. Jansson 1993] and in vitro study [F. Salvetti 2002]. The cause of this delay is not known yet. No delay is seen in the deactivation phase of transporting insulin from blood plasma to interstitium [Y.J. Yang 1989]. A small delay is seen between glucose uptake and lymph insulin for insulin binding to its cognate cell receptor [Y.J. Yang 1989].
Modelling insulin that is secreted by pancreatic β-cells via the liver into the portal vein.

The ability to control glucose production and utilization depends on insulin that is secreted by pancreatic β-cells via the liver, where insulin is decreased (extracted) by approximately 50%, into the portal vein. When the concentration of glucose in plasma increases rapidly, insulin is secreted by pancreatic β-cells in two phases: the first and second phase. The first 10 minutes (first phase) is a fast rise of insulin secretion. The next 20 minutes (second phase) is a gradually rise of insulin secretion which is directly related to the degree and duration of the stimulus [Caumo 2004]. Between the 70s and 80's several β-cell insulin secretion models were developed based on measurement of plasma insulin concentrations or measurement of c-peptide. These models can be added to the new oral glucose minimal model discussed in section 3 to estimate the plasma insulin profile instead of measuring the plasma insulin profile. Adding one of these models into the new oral glucose minimal model discussed in section 3 is also a closer step into a future automated portable continuous glucose monitoring system. First, models based on measurements of plasma insulin concentrations are given and secondly, models based on c-peptide measurements are given.

Toffolo proposed eight models tested on conscious healthy lean dogs by an IVGT [Toffolo 1980] using the minimal model [Bergman 1979]. The models assume that clearance of insulin is of the first order, the initial insulin peak represents a bolus loaded into insulin plasma after the glucose injection and the rate of the secondary rise in insulin is determined by the concentration of glucose in plasma above a specific threshold value. This model has the disadvantage that parameter estimation has to be done in two steps. First the insulin concentration is used as input data to estimate the parameters of the minimal model [Bergman 1979], secondly the recorded glucose data is used as input for the β-cell insulin secretion model. However, by the single-step fitting process [Zheng 2005] this handicap is resolved.

This proposed minimal model by Toffolo in 1980 is tested on healthy lean and obese humans by an IVGT [Bergman 1981] showing strong correlation with [Toffolo 1980]. This study also shows that lean lower tolerance humans is related to 77% diminished second phase responsivity (S2 remained unchanged) and that obese lower tolerance humans 60% diminished S2 (second phase remained unchanged).

Zheng introduced a modification of Toffolo's model [Zheng 2005] by the assumption that the insulin decay rate is not always a first-order process showing improved fitting (97.7%) than Toffolo's model in 1980. He also added a mathematical function for describing the endogenous insulin infusion rate. The model is shown in (b1) where $\gamma$ [pmol·time/min] is the rate of pancreatic release of insulin after the glucose injection, $h$ [mM] is the pancreatic threshold value, $t$ [min] is the time, $n$ [(pmol)·time/min] is the plasma insulin decay rate constant, $u(t)$ [mU/mg·min] is the endogenous insulin infusion rate, $m$ stands for the $m$-th order processes and $k$ [pmol] is the initial plasma insulin concentration after the glucose bolus injection. Due to the assumption that the insulin decay rate is not always a first-order process equation is modified by (b2).

\[
\begin{align*}
I(t) &= \begin{cases} 
\gamma \cdot [G(t) - h] \cdot t - n \cdot [I(t) - I_b]^n + u(t)/V & \text{if } G(t) > h \\
-n \cdot [I(t) - I_b]^n + u(t)/V & \text{if } G(t) \leq h
\end{cases} \\
I(0) &= k + I_b \quad (\text{b1})
\end{align*}
\]

\[
\begin{align*}
\dot{I}(t) &= -p2 \cdot I'(t) + p2 \cdot S_i \cdot [I(t) - I_b]^n \\
I(0) &= 0 \quad (\text{b2})
\end{align*}
\]
Steil introduced a new model [Steil 2003] which assumes a proportional component that reacts to the difference between plasma and basal glucose \( P \), an integral component that adjusts basal delivery in proportion to hypo/hyperglycemia \( h \) and a derivative component that responds to the rate of change in glucose plasma \( D \). The model is test during a CLAMP on healthy humans showing good fitness. Furthermore, this model is especially designed for closed loop systems linking a glucose sensor with an insulin pomp (an automated portable continuous glucose monitoring system) that includes stability and improved frequency characteristics/response. The model is shown in (b3, b4) where \( SR(t) \) is the \( \beta \)-cell insulin secretion rate, \( n \) \( [1/min] \) is the reciprocal of the insulin plasma decay rate constant, \( K_p \) \( [pmol \cdot 1/min \cdot 1/M] \) is the rate of \( \beta \)-cell insulin secretion, \( T_i \) \( [min] \) is the rate of proportional to integral release and \( T_o \) \( [Min] \) is the ratio of derivative to proportional release.

\[
I(t) = -n \cdot I(t) + \frac{SR(t)}{V}
\]

\( SR(t) = P(t) + I(t) + D(t) \) \( \text{SR(t) \geq 0} \)

\[
P(t) = K_p \cdot (G(t) - G_b)
\]

\[
I(t) = K_p \cdot (G(t) - G_b)/T_i \quad I(0) = I_b
\]

\[
D(t) = K_p \cdot T_o \cdot G(t)
\]

The test reveals that impaired glucose tolerance humans have a defect in the second phase as earlier stated.

Other models are based on the measurement of c-peptide [Eaton 1980] that is secreted in equal amounts as insulin with advantages that c-peptide is hardly extracted by the liver and its kinetics are linear. A disadvantage, from a qualitative standpoint, of models based on c-peptide is that the measurement of c-peptide is worse than the measurement of plasma insulin concentrations. However, these models can be modified in such a way that it can be used with measurements of plasma insulin concentrations.

[Toffolo 1995] introduces a c-peptide minimal model for an IVGTT which was extended by Breda to an oral c-peptide minimal model [Breda 2001]. The model assumes that \( \beta \)-cell insulin secretion depends on glucose plasma concentrations (static component, delayed production of new insulin) and on the rate of change of glucose plasma (dynamic component, secretion of stored insulin). This model is comparable with Toffolo’s model in 1980 that use the same static component, the same order (1\(^{st}\)) for clearance of insulin, but with no static delay and no dynamic component. The introduction of a dynamic component into the model of \( \beta \)-cell insulin secretion gives the model more physiological meaning. The model is shown in (b4, b5) where \( SR_s(t) \) \( [pM/min] \) is the static \( \beta \)-cell insulin response, \( SR_d(t) \) \( [pM] \) is the dynamic \( \beta \)-cell insulin response and \( 1/a \) \( [min] \) is the delay time in the static response.

\[
SR(t) = SR_s(t) + SR_d(t)
\]

\[
SR_s(t) = \alpha[SR_s(t) - \gamma \cdot [G(t) - h]]
\]

\[
SR_s(0) = 0
\]

\[
SR_d(t) = \begin{cases} 
K_p \cdot G(t) & \text{if } G(t) > 0 \\
0 & \text{if } G(t) \leq 0 
\end{cases}
\]
[Breda 2002] test reveals that impaired glucose tolerance humans have a diminished $S_b$, but an unaltered static and dynamic component as Bergman [Bergman 1981] found in obese lower tolerance humans. This test also reveals a delay between β-cell insulin secretion and changes of glucose concentration in plasma. Several other papers show similar results, e.g. [Ehrmann 2002, Cretti 2001]. Other publications have been published having likewise models [Mari 2002].

Both Steil's model and Breda's model are compared in [Steil 2003] using Bergman's minimal model [Bergman 1979] during a CLAMP on healthy humans. Both models fit the β-cell insulin secretion profile with low residuals. Also, comparable parameters between both models are not statistical different. However, Steil's model has better performance than Breda's model, since Steil's model has parameters with lower variations, more stability in close loop systems and adjusts better insulin needs to varying $S_b$ and endogenous glucose appearance. One drawback is that Steil's model is tested by a CLAMP and not during physiological circumstances like Breda's model that is tested during an OGTT. Since Steil's model represents very well the β-cell insulin secretion behaviour and has interesting closed loop properties for future application of the model into an automated portable continuous glucose monitoring system, this model is used.
C.1 Insulin profiles

<table>
<thead>
<tr>
<th>(min)</th>
<th>Profile 1</th>
<th>Profile 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>I(t) [pM]</td>
<td>G(t) [mM]</td>
</tr>
<tr>
<td>-15</td>
<td>53.0</td>
<td>6.16</td>
</tr>
<tr>
<td>0</td>
<td>45.9</td>
<td>6.55</td>
</tr>
<tr>
<td>15</td>
<td>86.2</td>
<td>6.66</td>
</tr>
<tr>
<td>30</td>
<td>134</td>
<td>7.87</td>
</tr>
<tr>
<td>45</td>
<td>145</td>
<td>8.31</td>
</tr>
<tr>
<td>60</td>
<td>165</td>
<td>8.91</td>
</tr>
<tr>
<td>90</td>
<td>174</td>
<td>8.97</td>
</tr>
<tr>
<td>120</td>
<td>169</td>
<td>9.52</td>
</tr>
<tr>
<td>180</td>
<td>196</td>
<td>8.86</td>
</tr>
<tr>
<td>240</td>
<td>162</td>
<td>8.91</td>
</tr>
</tbody>
</table>

Table C.1: Two insulin and glucose profiles [Wagenmakers 2006] measured during an OGTT (75 gram glucose)

C.2 Samples Schedules

<table>
<thead>
<tr>
<th>(min)</th>
<th>Sample schedules</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0 1 2 3 4 5 6 7 8 9 10 ... 240</td>
</tr>
<tr>
<td>2</td>
<td>0 10 20 25 30 35 40 45 50 55 60 65 70 80 90 120 180 240</td>
</tr>
<tr>
<td>3</td>
<td>0 15 30 45 60 90 120 180 240</td>
</tr>
</tbody>
</table>

Table C.2: Sample schedule 1, 2 and 3

C.3 Datasets

Dataset 1 is generated using insulin profile 1 and sample schedule 1.
Dataset 2 is generated using insulin profile 2 and sample schedule 1.
Dataset 3 is generated using insulin profile 1 and sample schedule 2.
Dataset 4 is generated using insulin profile 2 and sample schedule 2.
Dataset 5 is generated using insulin profile 1 and sample schedule 3.
Dataset 6 is generated using insulin profile 2 and sample schedule 3.

Interstitial glucose $G'(t)$ is measured in all previous datasets by taking the mean over each 30 minutes by taking the first sample at the 30th minute and the last sample at 240th minute.

1 Subject no: 5 Visit 6 / Day 15 [Wagenmakers 2006]
2 Subject no: 8 Visit 6 / Day 15 [Wagenmakers 2006]
3 Conversion factor between [μU / l] to [pM]: 7.0625.
4 Conversion factor between [mg/dl] to [mM]: 1/180.16 1/mM

C.4 Simulated model
Insulin profile 1

Figure c.1: Simulated glucose plasma, insulin plasma, interstitial glucose, interstitial insulin and the rate of appearance using profile 1, sample schedule 1 and true value shown in table 1.
Insulin profile 2

Figure C.2: Simulated glucose plasma, insulin plasma, interstitial glucose, interstitial insulin and the rate of appearance using profile 1, sample schedule 2 and true value shown in table 1
Plots of model to model adjustment

Model 1
Figure D.1: Plots of model to model adjustment of model 1 using dataset 1 (appendix c) with no added white noise
Figure D.2: Plots of model to model adjustment of model 1 using dataset 2 (appendix c) with no added white noise
Figure D.3: Plots of model to model adjustment of model 2 using dataset 2 (appendix C) with no added white noise
Model 3
Figure D.4: Plots of model to model adjustment of model 3 using dataset 3 (appendix C) with no added white noise
A priori parameter sensitivity for kabs

<table>
<thead>
<tr>
<th>1/SNR (%)</th>
<th>profile 1</th>
<th>profile 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Ib</td>
<td>Gb</td>
</tr>
<tr>
<td>0</td>
<td>40.0</td>
<td>72.5</td>
</tr>
<tr>
<td>5</td>
<td>40.0</td>
<td>131.7</td>
</tr>
<tr>
<td>10</td>
<td>40.0</td>
<td>73.3</td>
</tr>
</tbody>
</table>

A priori parameter sensitivity for v

<table>
<thead>
<tr>
<th>1/SNR (%)</th>
<th>profile 1</th>
<th>profile 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Ib</td>
<td>Gb</td>
</tr>
<tr>
<td>0</td>
<td>40.0</td>
<td>70.6</td>
</tr>
<tr>
<td>5</td>
<td>49.0</td>
<td>84.6</td>
</tr>
<tr>
<td>10</td>
<td>43.1</td>
<td>94.2</td>
</tr>
</tbody>
</table>

A priori parameter sensitivity for k

<table>
<thead>
<tr>
<th>1/SNR (%)</th>
<th>profile 1</th>
<th>profile 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>p2</td>
<td>p5</td>
</tr>
<tr>
<td>0</td>
<td>6.12</td>
<td>1.00</td>
</tr>
<tr>
<td>5</td>
<td>20.3</td>
<td>3.21</td>
</tr>
<tr>
<td>10</td>
<td>4.27</td>
<td>8.3</td>
</tr>
</tbody>
</table>

A priori parameter sensitivity for p4

<table>
<thead>
<tr>
<th>1/SNR (%)</th>
<th>profile 1</th>
<th>profile 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Ib</td>
<td>Gb</td>
</tr>
<tr>
<td>0</td>
<td>50.0</td>
<td>50.0</td>
</tr>
<tr>
<td>5</td>
<td>40.0</td>
<td>394.7</td>
</tr>
<tr>
<td>10</td>
<td>40.0</td>
<td>63.0</td>
</tr>
</tbody>
</table>

A priori parameter sensitivity for Si-Glut4

<table>
<thead>
<tr>
<th>1/SNR (%)</th>
<th>profile 1</th>
<th>profile 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>p2</td>
<td>p5</td>
</tr>
<tr>
<td>0</td>
<td>2.13</td>
<td>0.789</td>
</tr>
<tr>
<td>5</td>
<td>6.89</td>
<td>3.32</td>
</tr>
<tr>
<td>10</td>
<td>4.35</td>
<td>8.78</td>
</tr>
<tr>
<td>1/SNR (%)</td>
<td>profile 1</td>
<td>profile 2</td>
</tr>
<tr>
<td>-----------</td>
<td>-----------</td>
<td>-----------</td>
</tr>
<tr>
<td></td>
<td>Ib</td>
<td>Gb</td>
</tr>
<tr>
<td>0</td>
<td>50.0</td>
<td>50.0</td>
</tr>
<tr>
<td>5</td>
<td>48.2</td>
<td>946.4</td>
</tr>
<tr>
<td>10</td>
<td>48.7</td>
<td>770.4</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>1/SNR (%)</th>
<th>profile 1</th>
<th>profile 2</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>p2</td>
<td>p5</td>
</tr>
<tr>
<td>0</td>
<td>34.0</td>
<td>0.135</td>
</tr>
<tr>
<td>5</td>
<td>49.7</td>
<td>3.98</td>
</tr>
<tr>
<td>10</td>
<td>42.58</td>
<td>15.6</td>
</tr>
</tbody>
</table>

Table E.1: A priori parameter sensitivity for kabs, v, p4 and Si-glut4 using synthetic data and using two different insulin profiles (data sets 1, 2, respectively, appendix c) with added white noise having a 1/SNR of 0, 5 and 10%
European Research activities
On
(Closed loop) artificial pancreas and automatic insulin delivery

Consulted sources:
EMBase (ScienceDirect), MEDLINE (ScienceDirect), Elsevier BIOBASE
(ScienceDirect), Pubmed, Pascal (Pica), Web of Science, Elsevier Scopus, Inspec
(webspirs)

Principal Author(s): Hovorka, R. Chassin, L.J.

Literature:


Website: http://www.iph.cam.ac.uk/diabetes/

----------------------------------------------------------------------------------------

Inst. für Diabetes-Technologie, Universität Ulm, Helmholtzstr. 20, D-89081 Ulm, Germany:

Principal Author(s): Freckmann, G. B.

Literature:


More publications available at http://www.idt-ulm.de/framet.htm
Website: http://www.idt-ulm.de/frames.htm


Principal Author(s): Reach G.

Literature:

Choleau, C. and G. Reach, “Continuous glucose monitoring: Different systems, different ambitions,” *Diabetes and Metabolism*, Vol. 29(2003), No. 2 II.


Website: http://diabete.hotel-dieu.com

Syc, d'Endocrinologie Diabetologie, Hop. Universitaires de Strasbourg. Hopital Civil. 1 place de l'Hopital. 67091 Strasbourg Cedex, France.

Principal Author(s): Kessler, L.

Literature:


Website: http://www-ulpmcd.u-strasbg.fr/

¹ Department of Endocrinology, University Hospital, Strasbourg, France
² Department of Surgery, University Hospital, Geneva, Switzerland
³ Department of Endocrinology, University Hospital, Grenoble, France
⁴ Department of Medical Information, University Hospital, Strasbourg, France
⁵ Department of Endocrinology, University Hospital, Besancon, France
⁶ Department of Urology, Hospices Civils, Lyon, France
⁷ Department of Transplantation, University Hospital, Strasbourg, France
⁸ Department of Medical Information, Hospices Civils, Lyon, France
⁹ Service d'Endocrinologie et de Diabetologie, Hopitaux Universitaires, 1, Place de l'Hopital, 67091 Strasbourg Cedex, France

**Department of Endocrinology. Lapeyronie Hospital. F34295 Montpellier Cedex 5, France**

**Principal Author(s): Renard, E.**

**Literature:**


Renard, E., "Insulin therapy by insulin pump: Continuous or conventional self-blood glucose monitoring?," *Diabetes and Metabolism*, Vol. 29(2003), No. 2 11.


Website: http://irb.chu-montpellier.fr/fr/equipes_diabete.html
(http://www.chu-montpellier.fr/GB/rechclin2/act-inal-endocrin2.html?id=61&LANG=GB)

**Istituto per Ricerche di Dinamica dei Sistemi e di Bioingegneria. Consiglio Nazionale delle Ricerche. Univ. Padova. 35100 Padova. Italy**
Principal Author(s): Cobelli C.

Literature:
(no newer literature found)


Website: http://www.dei.unipd.it/wdyn/?IDsezione=172,
http://www.dei.unipd.it/wdyn/?IDsezione=313&IDgruppo_pass=32&preview

Istituto di Patologia Speciale Medica, University of Perugia, Via F. dal Pozzo, 06100 Perugia, Italy
(Internal Medicine and Endocrine and Metabolic Sciences)

Principal Author(s): Reboldi G.P

Literature:


Website: http://bioingweb.dimisem.med.unipg.it/default.htm,
http://bioingweb.dimisem.med.unipg.it/scuole/endocrinologia/Default.htm

=""
OLDER LITERATURE

Diabetic Hospital Steno Diabetes Center, Gentofte, Denmark

Principal Author(s):

Literature:


Metab. Unit, City Hosp. Vienna-Lainz, Vienna, Austria

Principal Author(s): Irsigler K.; Kritz H.

Literature:


Principal Author(s): Klein, J C; Slama, G.

Literature:


Vuk Vrhovac Institute for Diabetes, Endocrinology and Metabolic Diseases, School of Medicine, University of Zagreb, YU-41000 Zagreb, Yugoslavia

Principal Author(s): R savnik-Oblak, M., Granic, M.

Literature:


**Principal Author(s):** Battermann, P., Gyaram, H., Wahl, K., et al

**Literature:**


**Clin. Mal. Metab. Endocrinennes, Hop. Saint Eloi, Montpellier, France**

**Principal Author(s):** Mirouze, J., Selam, J.L., Chi Pham, T.

**Literature:**

Matlab code

D.1 Basic files

D.1.1

gluc_param_new.m

%---
% Oral Minimal Model of glucose metabolism
%---
% This script implements all parts of the Oral Minimal Model of glucose
% absorption and the time course of plasma insulin.
%
% History
% 26 March 06, Sebastian Ahlsone Ekma, Loo
% 04 May 04, Natalia et al., Loo
% 04 May 04, Natalia et al., Loo
%

close all; clear all;
plt=0;

%---
% Inner model parameters:
%---

Ib = 7.0625*(7.5+6.5)/2; % [mM] insulin concentration in plasma
Gb = 0.055*(112+119)/2; % [mM] glucose concentration in plasma
Glb=3.38; % [mM] baseline plasma glucose concentration

end

%---
% Inner model parameters:
%---

Ib = 7.0625*(6.4+4.5)/2; % [mM] insulin concentration in plasma
Gb = 0.055*(112+93)/2; % [mM] glucose concentration in plasma
Glb=3.38; % [mM] baseline plasma glucose concentration

end

%---
% Inner model parameters:
%---

Ib = 7.0625*(7.8+7.6)/2; % [mM] insulin concentration in plasma
Gb = 0.055*(165+169)/2; % [mM] glucose concentration in plasma
Glb=3.38; % [mM] baseline plasma glucose concentration in interrum

end

% Known unknown model parameters
% (Mean values obtained from other studies on intact people)
k8 = 3e-2; // Pore water pressure
p2 = 0.113; // Density of saturated water
DS = 75/10; // Saturated density
Kabs = 2.89e-2; // Bulk modulus of soil
Beta = 1.23; // Bonding modulus
k = 0.014; // Field capacity
V = 2.497e-1; // Inundation field
p5 = 0.018; // Saturated pressure
p6 = 0.00763*10; // Saturated pressure
SICap = 2*e-3*0.0395*(1/0.21)*e3; // Saturated capillary pressure
Siliver = 1.16e-5*1e4*(1/7.0625); // Soil volume
Siglut4 = 1.0e-2; // Soil volume
p4 = k7*k8; // Pore water pressure
Coeff = 3/75; // Q
p_true = [Ib, Gb, p2, DS, Kabs, Beta, k, V, p4, p5, p6, SICap,...
Siliver, Siglut4, Coeff];
Fixed initial conditions:
L0(1) = Gb; [mM] glucose concentration at birth.
L0(2) = 0; [mM] labeled water concentration.
L0(3) = 0; [mM] insulin concentration in PP.
L0(4) = Gf; [mM] glucagon concentration in PP.
L0(5) = 0; [mM] glucose concentration in the PP.

Output: insulin concentration in plasma [mM]; assumed to be known at each time sample from linear interpolation of its measured sample.

Time and amplitude data:
1 Subject: 1 Male 18 y.
1 Blood glucose data:
t_insu1in1 = [0 15 30 45 60 90 120 180 240];
c_insu1in1 = 7.0625*[(7.5+6.5)/2 12.2 19 20.5 23.3 24.6 23.9 27.7 23];
t_c_insu1in1 = [t_insu1in1' c_insu1in1'];

2 Blood glucose data:
t_insu1in2 = [0 15 30 45 60 90 120 180 240];
c_insu1in2 = 7.0625*[(6.4+4.5)/2 14.8 22.2 27.7 46.8 81 46 7.7];
t_c_insu1in2 = [t_insu1in2' c_insu1in2'];

3 Interstitial glucose data:
t_insu1in1 = [0 15 30 45 60 90 120 180 240];
c_insu1in1 = 7.0625*[(7.8+7.6)/2 13.4 19.7 20.2 21.5 29.6 27.5 18.8 14.2];
t_c_insu1in1 = [t_insu1in1' c_insu1in1'];

4 Interstitial glucose data:
t_insu1in2 = [0 15 30 45 60 90 120 180 240];
c_insu1in2 = 7.0625*[(7.8+7.6)/2 13.4 19.7 20.2 21.5 29.6 27.5 18.8 14.2];
t_c_insu1in2 = [t_insu1in2' c_insu1in2'];

5 Append 2 norms of the errors: e1 and e2.
$a=1; \text{ Start out with the initial or baseline concentration (1st output)}$

$b=1; \text{ Output would not the first or baseline plasma concentration (2nd output)}$

$c=1; \text{ Output would not the first or baseline plasma concentration (3rd output)}$

$\sigma_1=0.025; \text{ Variance of baseline plasma glucose concentration output}$

$\sigma_2=0.05; \text{ Variance of initial plasma glucose concentration output}$

$\sigma_3=1; \text{ Variance of 1st hour plasma glucose concentration output}$

$\sigma_4=0.05; \text{ Variance of 2nd hour plasma glucose concentration output}$

$\sigma_5=0.025; \text{ Variance of 3rd hour plasma glucose concentration output}$

$\sigma_6=0.05; \text{ Variance of 4th hour plasma glucose concentration output}$


```matlab
% Make plots at plt 1
plt=1;
figure; plot(t_c_insulin(:,1), t_c_insulin(:,2), 'r'); hold on
plot([t_c_insulin(1,1) t_c_insulin(end,1)], [Ib Ib], 'k--', 'LineWidth',1.5)
xlabel('Time (hr)'); ylabel('C Glucose (mM)'); title('C Glucose vs time at plt 1')
end
```

$t_{exp}=15; \text{ Time to end study}$
D.1.2

```
function dOut = gluc_ode_new(t,lin,t_c_insulin,p)

% gluc_ode_new: 5th order Minimal Model of glucose dynamics
% using the parameters from the gluc_ode.m.

% Function:
% dOut = gluc_ode_new(t,lin,t_c_insulin,p).

% Inputs:
% Lin: state values at previous time step: (1x k) = [G(k),GL(k-d),
% GL(k-2),y(k-3)]; dOut(k) = [column k] of matrix
% Output:
% dOut: new state derivatives (dx(k),dy(k),dGL(k))

% Define:
% 1. March 06, Silvia Albaam 1.
% 2. May 06, Natal van der Ruyt.
% 3. June 06, Natal van der Ruyt.

% defining the new way to get state in a vector form in present
idG = 1; glucose concetration [mM], first column of vector
idGL = 2; plasma glucose concentration [mM], second column of vector
idX = 3; first partial derivative - vector [mM], third column of vector
idY = 4; interstitial glucose concentration [mM], fourth column of vector
idGut = 5; the rate of glucose to the gut [1/min], fifth column of vector

% (p1, p2) parameters:
Ib = p(1); [mM] baseline plasma concentraion of glucose
Gb = p(2); [mM] baseline plasma concentration in plasma
```

* Known/unknown model parameters
* Glucose from blood plasma to interstitia by epithelial recruitment
\begin{verbatim}
dG = -((Sliver/10000) + (p4))*(Lin(idG)-Gb)*(u-Ib)+p5-(p6/10)*Beta*(Lin(idG)-Gb)+(Kabs*Lin(idGgut))/V;
dGL = 0;
dX = -p2*Lin(idX)+ (Slcap/10000) *(u-Ib)^2;
dY = -Slglut4*Lin(idY)*Lin(idX)+ (p4/10)*(Lin(idG)-Gb)*(u-Ib);
dGgut = (DS*10)*k*Beta*exp(-k*t)*Beta-Kabs*Lin(idGgut);
dLout = [dG; dGL; dX; dY; dGgut];
\end{verbatim}
D.1.3
gluc_sim_new.m

function [c_glucose, c_glucose_labeled, cInterstitial_insulin,...
c_interstitial_glucose, c_Ggut, c_Ggut_labeled]=gluc_sim_new (L0,...
t_c_insulin,p,plt,TITLE)

% Initial Minimal Model of glucose system

% This function makes a simulation of the Initial Minimal Model of glucose
% system using the parameters in the main function of the program.

function:

[c_glucose, c_glucose_labeled, c_interstitial_glucose, c_Ggut, c_Ggut_labeled]=gluc_sim_new (L0,...
t_c_insulin,p,plt,TITLE)

% Initial Minimal Model of glucose system

% This function makes a simulation of the Initial Minimal Model of glucose
% system using the parameters in the main function of the program.

% Model parameters:

% L0 [mM] is the insulin concentration in plasma.
% t_c_insulin [min] is the time course of the insulin injected.
% p [mM] is the plasma insulin concentration.
% plt [mM] is the plasma glucose concentration.
% TITLE is the title of the graph.
ode_options =odeset('RelTol',1e-6,'AbsTol',1e-12);
[t,L] = ode45(@gluc_ode_new,[0 t_c_insulin(:,1)],LO,ode_options,
t_c_insulin,p);

% create plots and labels
plot(1:plt,plt==1)
figure;
Ib=p(1);
subplot(321);plot(t_c_insulin(:,1),t_c_insulin(:,2),'.','LineWidth',1.5)
hold;
 subplot(322);plot(t_c_insulin(1,t_c_insulin(end)),[Ib Ib],'.','LineWidth',1.5)
ylabel('')
% Create plots

subplot(322); plot(t_c_insulin(:,1), c_interstitial_insulin, '-');
xlabel('Time (min)'); ylabel('Insulin (mU/mL)');
Title(TITLE);

Gb=p(2);
subplot(323); plot(t_c_insulin(:,1), c_glucose, '-');
hold on;
plot([t_c_insulin(1,1) t_c_insulin(end,1)], [Gb Gb], 'r');% Gb
xlabel('Time (min)'); ylabel('Glucose (mg/dL)');
Title(TITLE);

subplot(324); plot(t_c_insulin(:,1), c_glucose_labeled, '-');
xlabel('Time (min)'); ylabel('Labeled Glucose (mg/dL)');
Title(TITLE);

subplot(325); plot(t_c_insulin(:,1), c_interstitial_glucose, '-');
xlabel('Time (min)'); ylabel('Interstitial Glucose (mg/dL)');
Title(TITLE);

subplot(326); plot(t_c_insulin(:,1), c_Ggut, '-');
xlabel('Time (min)'); ylabel('Gut Glucose (mg/dL)');
Title(TITLE);
end
function error = obj_fn_new(p_var, p_fix, LO, datal, data2, data3,...
  t_c_insulin1, t_c_insulin2, a, b, c, sigma1, sigma2, sigma3,...
  t_exp, Nparameters, NoMean)

%% Objective function for Nelder-Mead minimization
%%
%% This function is used as objective function for NAGNONLIN.
%%
%% Function:
%% error = obj_fn_new(p_var, p_fix, LO, datal, data2, data3,...
%%                      t_c_insulin1, t_c_insulin2, a, b, c, sigma1, sigma2, sigma3,...
%%                      t_exp, Nparameters, NoMean) ;

%% Inputs:
%% p_var: Parameter space which will be optimized.
%% p_fix: Parameter space which is fixed.
%% LO: Fixed initial conditions, i.e. initial concentration of
%%     plasma, L(u) [mg] in the initial condition.
%% V(t) [mg] time evolution of concentration,
%% t_c_insulin[1,2] [min] time of injection period 1 and 2.
%% t_exp: time of all experiments in seconds.
%% Nparameters: number of parameters.
%% NoMean: boolean for mean calculation.
%%
%% Output: error: output weighting factor 1: cost,
%%        weights: output weighting factor 2: error,
%%        cost: cost function defined, the parameter which is used in the model to
%%               monitor the retardation between.

%% History
%% D.I.4

90
p_estimating = [p_fix(1), p_fix(2), p_fix(3), p_fix(4), p_var(1), ...
p_fix(5), p_fix(6), p_var(2), p_var(3), p_fix(7), ...
p_fix(8), p_fix(9), p_fix(10), p_var(4), p_fix(11)];

% Simulating the Oral Minimal Model of glucose dynamics using the
% estimation
% All temporal parameter values are the interpolated insulin concentration
% samples for data.

plt=0; % make No plots
[c_glucose_estimating, c_glucose_labeled_estimating,...
c_interstitial_insulin_estimating, c_interstitial_glucose_estimating,...
c_Ggut_estimating] = gluc_sim_new(lo, t_c_insulin1,
p_estimating, plt, ...
'...');

% Simulating the Oral Minimal Model of glucose dynamics using the
% estimation
% All temporal parameter values are the interpolated insulin concentration
% samples for data.

if NoMean==0
  " Interpolating data. 
  t_insulin2_interpl = t_c_insulin2(1,1):1:t_c_insulin2(end,1);" [min]
c_insulin2_interpl=interpl(t_c_insulin2(:,1),t_c_insulin2(:,2),...
t_insulin2_interpl);
t_c_insulin2_interpl(:,1)=t_insulin2_interpl;
t_c_insulin2_interpl(:,2)=c_insulin2_interpl;

[c_glucose_estimating2_interpl, c_glucose_labeled_estimating2_interpl,...
c_interstitial_insulin_estimating2_interpl,...
c_interstitial_glucose_estimating2_interpl,...
c_Ggut_estimating2_interpl] = gluc_sim_new(lo, t_c_insulin2_interpl,
...p_estimating, plt, '...');

  " Calculating the mean of the interpolated data at the data2
  samples. 
  G1b=LO(4);
c_interstitial_glucose_estimating2_mean(1,1)=G1b;
i=2;
for r=t_c_insulin2(2:end,1)'
c_interstitial_glucose_estimating2_mean(i,1)=...
sum(c_interstitial_glucose_estimating2_interpl(r+1-t_exp:r))/t_exp;
i=i+1;
end"
else
  [c_glucose_estimating2, c_glucose_labeled_estimating2, ...
   c_interstitial_insulin_estimating2, ...
   c_interstitial_glucose_estimating2, c_Ggut_estimating2] = ...
   gluc_sim_new(L0, t_c_insulin2, p_estimating, plt, ...
   '"."');
end

function l_error = get_error(error1, error2, error3, NoMean, ...
  error1 = c_glucose_estimating-data1;
  if NoMean == 0
    error2 = c_interstitial_glucose_estimating2_mean-data2;
    else
    error2 = c_interstitial_glucose_estimating2-data2;
  end
  error3 = c_glucose_labeled_estimating-data3;
  ...
  if length(error1) < length(error2)
    for q=1:(length(error1)-length(error2))
      error2_new(length(error2)+q) = 0;
      data2_new(length(data2)+q) = 1;
    end
    error1_new = error1;
    data1_new = data1;
    error3_new = error3;
    data3_new = data3;
  else
    for q=1:(length(error2)-length(error1))
      error1_new(length(error1)+q) = 0;
      data1_new(length(data1)+q) = 1;
      error3_new(length(error3)+q) = 0;
      data3_new(length(data3)+q) = 0;
    end
  end

end

92
error2_new=error2;
da2_new=data2;
end

ererror=1e4*error1_new+1e4*error2_new+125*1e4*error3_new;

figure(100)
subplot(411);
plot(t_c_insulin1(:,1),c_interstitial_insulin_estimating,'-',
xlabel(''),
ylabel(''),
Title({'i
drawnow;pause(.3);

subplot(412);
plot(t_c_insulin1(:,1),c_glucose_estimating,'-',t_c_insulin1(:,1),
xlabel(''),
ylabel(''),
Title({'1
drawnow;pause(.3);

subplot(413);
plot(t_c_insulin1(:,1),c_glucose_labeled_estimating,'-',t_c_insulin1(:,1),
xlabel(''),
ylabel(''),
Title({'!
drawnow;pause(.3);

subplot(414);
if NoMean==0
plot(t_c_insulin2(:,1),c_interstitial_glucose_estimating2_mean,'-',
     t_c_insulin2(:,1),data2,' '); 
else
plot(t_c_insulin2(:,1),c_interstitial_glucose_estimating2,'-',
     t_c_insulin2(:,1),data2,' '); 
end
xlabel(''),
ylabel('');
D.2 model2model adjustment

D.2.1

model2model_bias_new.m

This script identifies a posteriori the Dual Minimal Model of glucose
excursions under the model-to-model adjustment approach for each:
stimulation
of the experimental protocol (p).

History:
22 March 06, Salvador Almagro Frutos, TUI

Structures all parameters of the 2D or all-time sample of plasma
insulin.
clear 1;
NoMean=0; "Calculate the mean over the multiple of interstitial glucose
samples, if set to 1.
gluc_param2_new;
Nparameters=length(p_true);

% define data:
data1_new=data1;
data2_new=data2;
data3_new=data3;

% initial parameter space
rr=[-0.5 -0.45 -0.4 -0.35 -0.3 -0.25 -0.2 -0.15 -0.1 0 0.1 0.15 0.2 0.25 0.3 0.35 0.4 0.45 0.5];
sr=[0.01 0.1 0.2 0.3 0.4];
par=[0.0 0.1 0.2 0.3 0.4 0.5];
pname = ['1'; '2'; '3'; '4'; '5'; '6'; '7'; '8'; '9'; '10'; ...'; '11'; '12'; '13'; '14'; '15'; '16'; '17'; '18'; '19'; '20'];

Model: model parameters approach - each parameter $(P_1, P_2)$ of
be
$14$-dimensional parameter set $p$.

$p_{11} p_{12} p_{13}; P_{21} p_{22} p_{23}; \ldots; P_{14} p_{14} p_{15}$
$a_{P_{11}} p_{11} p_{12}; P_{21} a_{P_{21}} p_{22} p_{23}; \ldots; P_{14} a_{P_{14}} p_{14} p_{15}$
$sP_{11} p_{11} p_{12}; P_{21} sP_{21} p_{22} p_{23}; \ldots; P_{14} sP_{14} p_{14} p_{15}$

Initialize the $\epsilon$-estimator (see Eq. 12)
for $qq=[5, 8, 9, 13, 14]$ and $q' = \{\text{parameters of } 1\} =$

$\text{estimator}(1) = qq;$

Initialize the $\epsilon$-estimator (see Eq. 12)
for $qq=[5, 8, 9, 13, 14]$ and $q' = \{\text{parameters of } 2\} =$
bb=1;

if $yy>qq$

$\text{estimator}(2) = yy;$

Save all other parameters of the $14$-dimensional parameter
space.

$p_{\text{fix}} = []$
for $zz=1:Nparameters$

\begin{align*}
&zz = \text{estimator}(1), \\
&zz = \text{estimator}(2)
\end{align*}
$p_{\text{fix}} = [p_{\text{fix}} p_{\text{true}}(zz)];$
\end{align*}
end
end

for $ii=1:length(rr)$
for $jj=1:length(rr)$

Initialisation at the $N$-dimensional parameter space which
will
be estimated.

initial_start_value_parameters = [p_{true} (estimator(1))
p_{true} (estimator(2))];
p_{init} = [p_{true} (estimator(1)) + rr(ii) * p_{true} (estimator(1))]
\ldots
p_{true} (estimator(2)) + rr(jj) * p_{true} (estimator(2))];
```matlab
[resnorm] = gluc_omm_mls2_new(data1, data2, data3, L0, t_c_insulin1, ...
    t_c_insulin2, a, b, c, sigma1, sigma2, sigma3, p_true, p_fix, p_init, ...
    t_exp, Nparameters, estimator, NoMean, plt);

% Saving the squared error of the error function for residuals
DataZ(ii, jj) = resnorm;

for both true parameters, a purpose of Phi, plt
if rr(ii) == 0 & rr(jj) == 0
    resnormTrue = resnorm;
end

clear i; j;

[p_model, L0_model, c_glucose_model, c_glucose_labeled, ...
    c_interstitial_insulin_model, c_interstitial_glucose_model, ...
    c_Ggut_model, resnorm, residual, varp, stdp] = gluc_omm_mls_bias_new...
    (data1_new, data2_new, data3_new, L0, ...
    t_c_insulin1, t_c_insulin2, a, b, c, sigma1, sigma2, sigma3, ...
    p_fix, p_init, t_exp, Nparameters, estimator, NoMean, plt);

ptruel = p_true(qq)
ptrue2 = p_true(yy)
estimation(bb, :) = [p_model(qq) p_model(yy)]
bb = bb + 1;
end

DataY(jj) = p_true(estimator(2)) + rr(jj) * p_true(estimator(2));
end

DataX(ii) = p_true(estimator(1)) + rr(ii) * p_true(estimator(1));
end

clear i; j;

Minn = min(min(Data2));
```
Maxx=max(max(DataZ))
Contourlines=50
((min(min(DataZ))); 0.01: (max(max(DataZ))))';
Contourlines>200
Contourlines=200;
display...
end

figure(333);
contour(DataX,DataY,DataZ,Contourlines);
hold ;
plot (p_true(estimator(1)),p_true(estimator(2)),',
hold
plot(estimation(:,1),estimation(:,2),',
xlabel(['',pname(estimator(1),:))]);
ylabel(['',pname(estimator(2),:))]);
Title('');
colormap [1];
colorbar;
hold 

savename=['',pname(estimator(1),:) pname(estimator(2),:)]
saveas(gcf, savename, 'break');
savename=['',pname(estimator(1),:) pname(estimator(2),:)]
saveas(gcf, savename, '');
savename=['',pname(estimator(1),:) pname(estimator(2),:)]
saveas(gcf, savename, 'break');
close(figure(100));
end

end

98
D.2.2

gluc_param2_new.m

% This script calculates all parameters of the Oral Minimal Model of glucose metabolism
% given the time sample and the parameters.

% Author:
% V. E. March 6th, Salvador Alvarado-Fuentes, DF.
% 3-8 May 84, Natal van de Vel, DN.
% 17-June 84, Natal van de Vel, DN.

close fig;
% make plots fit plt
plt=1;

% Oral minimal model parameters:
% Values from: 1988:

if 1
% Subject no: 1 Visit 1: Day 1
Ib = 7.0625*(7.5+6.5)/2; [1.0] baseline glucose concentration in plasma
Gb = 0.055*(112+119)/2; [1.0] baseline glucose concentration in plasma
Glb=3.38; [1.0] fasting glucose concentration in plasma
end

if 0
% Subject no: 1 Visit 1: Day 1
Ib = 7.0625*(7.8+7.6)/2; [1.0] baseline glucose concentration in plasma
Gb = 0.055*(112+93)/2; [1.0] baseline glucose concentration in plasma
Glb=3.38; [1.0] fasting glucose concentration in plasma
end

% Known unknown model parameters:
% (Mean) du,t+1 = a*du,t + b*du,t-1 + c*t + d*du,t-1 + e*du,t-1 + f*du,t-1 + g*du,t-1
% Annual International Conference on the IIEE Engineering in Medicine
% 24-26 November, Cancun, Mexico, 1988:
% Natalien, S., and F. De March, F. Radzevicius, C. De March, F. Moreira,
% S. Farbman.
% "Absorption and Insulin Sensitivity in a Two-Stage Tolerance Test,"
% Proceedings of the 25th Annual International Conference on the
k8 = 3e-2

p2 = 0.113

DS = 75/10

Kabs = 2.89e-2

Beta = 1.23

k = 0.014

V = 2.497e-1

p5 = 0.018

p6 = 0.00763*10

k7 = 0.063*(1/0.21)

SIcap = 2e-3*0.0395*(1/0.21)*le3

Silver = 1.11e-5*le4*(1/7.0625)

SIglut4 = 1.0e-2

Coeff = 3/75

p_true = [Gb, Gb, p2, DS, Kabs, Beta, k, V, p4, p5, p6, SIcap, Silver, SIglut4, Coeff]
% Input insulin concentrations in plasma [ml]; supposed to be shown if
% Simulation time sample from 1 to 5; or a long time in measured
% sample.

% Time and amplitude samples of insulin data:
% 1 Subject no. 1 Visit 1, 2
% t: blood glucose data
c: insulin data

% t_c: insulin = [t_inulin1' c_inulin1'];
end

% t_c: insulin = [t_inulin1' c_inulin1'];
end

% Subject no. 1 Visit 2
% t: blood glucose data

% t_c: insulin = [t_inulin2' c_inulin2'];
end

% Subject no. 2 Visit 1
% t: blood glucose data

% t_c: insulin = [t_inulin2' c_inulin2'];
end

% Subject no. 2 Visit 2
% t: blood glucose data

% t_c: insulin = [t_inulin2' c_inulin2'];
end

% Output weighting factor of plasma glucose concentration flat
a=1;
% Output weighting factor of interstitial glucose concentration flat
b=1;
% Output weighting factor of plasma glucose concentration flat
end output
c=1;
% Output weighting factor of interstitial glucose concentration flat
end output

% Variance of plasma glucose concentration flat output
sigma1=1;
% Variance of interstitial glucose concentration flat output
sigma2=1;
% Variance of plasma glucose concentration flat output
sigma3=1;
%Plots
figure; plot(t_c_insulin1(:,1), t_c_insulin1(:,2), 'r'); hold
plot([t_c_insulin1(1,1) t_c_insulin1(end,1)], [Ib Ib], 'r--', 'LineWidth', 1.5)
xlabel('Time (min)'); ylabel('Insulin (mU/L)'); title('Insulin Profile')
end

%Simulating the 2nd Interval Data: Choose what's using the TRUE
14-dimensional parameter space and the interpolated insulin concentration
examples for data1.

%sample internal [p1] at time

if NoMean==0
% Interpolating data...
end

% calculating the mean of the interpolated data at the data2
samples

else
end

102
=gluc_sim_new (L0, t_c_insulin2, p_true, plt, )
end

% data vector of the oral Minimal Model of glucose kinetics using the true
% 15-dimensional parameter space with interpolated insulin
% concentration samples.

data1 = c_glucose1;
data2 = c_interstitial_glucose_mean;
data3 = c_glucose_labeled1;

t_exp = 15;
NoMean = 0
if NoMean == 0
data2 = c_interstitial_glucose_mean;
else
data2 = c_interstitial_glucose2;
end

data2 = c_interstitial_glucose mean;
D.2.3

`gluc_omm_mls2_new.m`

```matlab
function [resnorm]=gluc_omm_mls2_new(data1,data2,data3,L0,t_c_insulin1,...
    t_c_insulin2,a,b,c,sigma1,sigma2,sigma3,p,p_fix,p_init,t_exp,...
    NParameters,estimator,NonMean,plt)

... MATLAB code ...
```

104
for i: ww=estimator(i) compute the i-th estimated parameter and save it at the correct position in the defined 14-dimensional parameter space

if WW==estimator(1)
    offset=offset+1;
    p_model=[p_model p_init(i)];
else
    offset=offset+1;
    p_model=[p_model p_fix(ww-offset)];
end

end

simulate the first Minimal Model of glucose kinetics using the 14-dimensional parameter space and the predicted insulin concentration samples for each
[c_glucose1,c_glucose_labeled1,c_interstitial_insulin1,...
c_interstitial_glucose1,c_Gut1,c_Gut_labeled1]....
=gluc_sim_new(L0,t_c_insulin1,p_model,plt,"111111")

if NoMean==0
    [interp1=p_output t_c_insulin2_interpl=t_c_insulin2(1,1):t_c_insulin2(end,1);]
c_insulin2_interpl=interp1(t_c_insulin2(:,1),t_c_insulin2(:,2),... t_insulin2_interpl);
t_c_insulin2_interpl(:,1)=t_c_insulin2_interpl;
t_c_insulin2_interpl(:,2)=c_insulin2_interpl;
c_interstitial_glucose_interpl, c_interstitial_insulin_interpl, c_Ggut_interpl, c_Ggut_labeled_interpl,...
=gluc_sim_new (L0, t_c_insulin2_interpl, p_model, plt, ', ');

% calculating the mean of the interpolated data2 at the data2 samples
Glb=L0(4);
c_interstitial_glucose_mean(1,1)=Glb;
i=2;
for r=t_c_insulin2(2:end,1)
  c_interstitial_glucose_mean(i,1) =
  sum(c_interstitial_glucose_interpl(r+1-t_exp:r))/t_exp;
i=i+1;
end

else
  simulation samples:
  [c_glucose2,c_glucose_labeled2,c_interstitial_insulin2,...
c_interstitial_glucose2,c_Ggut2,c_Ggut_labeled2]....
  =gluc_sim_new (L0, t_c_insulin2, p_model, plt, ', ');
end

% error function resoalts:
error1=c_glucose1-data1;

if NoMean==0
  error2=c_interstitial_glucose_mean-data2;
else
  error2=c_interstitial_glucose2-data2;
end

error3=c_glucose_labeled1-data3;

% total model error

% if datasets have different range of samples, the dimensions of error
% and data are not equal except for the limiting values:
error1_new=error1;
error2_new=error2;
error3_new=error3;
data1_new=data1;
data2_new=data2;
data3_new=data3;

if length(error1)>length(error2)
  for q=1:(length(error1)-length(error2))
    error2_new(length(error2)+q)=0;
data2_new(length(data2)+q)=1;
  end
end
%f

if error2_new is already not
end

error1_new=error1;
data1_new=data1;
error3_new=error3;
data3_new=data3;

else
for q=1:(length(error2)-length(error1))
    error1_new(length(error1)+q)=0;
data1_new(length(data1)+q)=1;
    error3_new(length(error3)+q)=0;
data3_new(length(data3)+q)=1;
end

error2_new=error2;
data2_new=data2;
end

\textbf{Squared} 2 \textbf{Norm of the Error Function Residuals}

\begin{align*}
\text{resnorm1} &= 1 \times (1/\text{length}(\text{error1})) \times \text{sum}((\text{error1}_{\text{new}})^2); \\
\text{resnorm2} &= 1 \times (1/\text{length}(\text{error2})) \times \text{sum}((\text{error2}_{\text{new}})^2); \\
\text{resnorm3} &= 125 \times 1 \times (1/\text{length}(\text{error3})) \times \text{sum}((\text{error3}_{\text{new}})^2); \\
\end{align*}

\textbf{Total Squared} 2 \textbf{Norm of the Error Function Residuals}

\begin{align*}
\text{resnorm} &= \text{resnorm1} + \text{resnorm2} + \text{resnorm3}; \\
\end{align*}

107
D.2.4

```matlab
function [p_model,LO_model,c_glucose_model,c_glucose_labeled_model,...
  c_interstitial_insulin_model,c_interstitial_glucose_model,...
  c_Gut_model,resnorm,residual,varp,stdp]=gluc_omm_mls_bias_new(data1,...
  data2,data3,LO,t_c_insulin1,t_c_insulin2,a,b,c,sigma1,sigma2,sigma3,...
  p_fix,p_init,t_exp,Nparameters,estimator,NoMean,plt)
```

These are the functions to calculate different models and parameters based on the input data. The function `gluc_omm_mls_bias_new` takes multiple arguments, including data sets, models, and parameters, to compute various outputs such as `p_model`, `LO_model`, etc. The output includes model parameters, residual sum of squares, and other metrics relevant to the model fitting process. The function is designed to handle datasets for different physiological models, including glucose, insulin, and other parameters related to glucose metabolism.
options = optimset('Display','iter', 'MaxFunEvals', le-20, ...
                          'MaxIter', le-10, ...);

[p_estimated, resnorm, residual, exitflag, output, lambda, jacobian] =
lsqnonlin(@obj_fn_bias_new, p_init, lb, ub, options, p_fix, L0, data1, data2, ...
                  data3, t_c_insulin1, t_c_insulin2, a, b, c, sigmal, sigma2, sigma3, t_exp, ...)
                  Nparameters, estimator, NoMean);  %_init, lb, ub, options

if exitflag==0
STOP=1;
% Accuracy:
% The model outputs the mean of the predictions:
varp = resnorm*inv(jacobian'*jacobian)/length(t_c_insulin(:,1));

stdp = sqrt(diag(varp)); the current realistic is the square root of
the variance

% 1-dimensional parameter space:

% No, est, fix, fix, est, fix, est, est, fix, est, est, est, ...
% estimator, estimator, ...

p_model = []; offset=0;

for ww=1:Nparameters
    if ww==estimator(1) % Add 1st estimated parameter and save it
        offset=offset+1;
        p_model=[p_model p_estimated(1)];
    else
        ww==estimator(2) % Add 2nd estimated parameter and
    offset=offset+1;
        p_model=[p_model p_estimated(2)];
    else
        if the other parameters are unestimated
    offset=offset+1;
        p_model=[p_model p_fix(ww-offset)];
    end
end

L0_model = [p_fix(2),0,0,0];

% make plots with new 1-dimensional parameter space: plot(L
[c_glucose_model,c_glucose_labeled_model,c_interstitial_insulin_model,
.. c_interstitial_glucose_model,c_Ggut_model]= gluc_sim_new (L0,...
t_c_insulin,p_model,plt,'- ');}
D.2.5

obj_fn_bias_new.m

function error = obj_fn_bias_new(p_var, p_fix, L0, data1, data2, data3,...
    t_c_insulin1, t_c_insulin2, a, b, c, sigma1, sigma2, sigma3,...
    t_exp, Nparameters, estimator, NoMean)

pvar=p_var

% obj_fn_bias_new
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
% This function is used to calculate the cost function in the DE!
estimating the interstitial glucose concentration

\[ p_{estimating} = []; \]
offset=0;

for \( \text{ww}=1: \text{Nparameters} \)
    if \( \text{ww}==\text{estimator}(1) \)
        offset=offset+1;
        \[ p_{estimating}=[p_{estimating} \text{p\_var}(1)]; \]
    elseif \( \text{ww}==\text{estimator}(2) \)
        offset=offset+1;
        \[ p_{estimating}=[p_{estimating} \text{p\_var}(2)]; \]
    else
        \[ p_{estimating}=[p_{estimating} \text{p\_fix} \text{(ww-offset)}]; \]
    end
end

%Simulating the \text{t\_c\_insulin1} and \text{t\_c\_insulin2} concentrations
%Simulating the \text{p\_estimating} and \text{t\_c\_insulin1} concentrations
%Simulating the \text{interstitial glucose concentration}

plt=0; % make \text{plt} plot
\[ \text{c\_interstitial\_insulin\_estimating, c\_interstitial\_glucose\_estimating, ...} \]
\[ \text{c\_Ggut\_estimating} = \text{gluc\_sim\_new}([0, \text{t\_c\_insulin1,} \]
\[ \text{p\_estimating, plt, ...} \]

%Simulating the \text{t\_c\_insulin1} and \text{t\_c\_insulin2} concentrations
%Simulating the \text{p\_estimating} and \text{t\_c\_insulin1} concentrations
%Simulating the \text{interstitial glucose concentration}

if \text{NoMean}=0

% Interpolation ...
\[ \text{t\_insulin2\_interpl} = \text{t\_c\_insulin2}(1,1):1: \text{t\_c\_insulin2}(\text{end},1); \]
\[ \text{c\_insulin2\_interpl} = \text{interpl}([\text{t\_c\_insulin2}(:,1), \text{t\_c\_insulin2}(:,2), ...}, \]
\[ \text{t\_insulin2\_interpl}); \]
\[ \text{t\_c\_insulin2\_interpl}(::,1) = \text{t\_insulin2\_interpl}; \]
\[ \text{t\_c\_insulin2\_interpl}(::,2) = \text{c\_insulin2\_interpl}; \]

112
[c_glucose_estimating2_interpl, c_glucose_labeled_estimating2_interpl, ...
  c_interstitial_insulin_estimating2_interpl, ...,
  c_interstitial_glucose_estimating2_interpl, ...
  c_Ggut_estimating2_interpl] = gluc_sim_new(LO, t_c_insulin2_interpl,
  ...
  p_estimating, plt, ' ', ' ');

% calculating the mean of the interstitial data at the data sampling points;
Glb=LO(4);
c_interstitial_glucose_estimating2_mean(i,1)=Glb;
i=2;
for r=t_c_insulin2(2:end,1)'
c_interstitial_glucose_estimating2_mean(i,1)=...
  sum(c_interstitial_glucose_estimating2_interpl(r+1-t_exp:r))/t_exp;
i=i+1;
end
else
[c_glucose_estimating2, c_glucose_labeled_estimating2, ...
  c_interstitial_insulin_estimating2, ...,
  c_interstitial_glucose_estimating2, c_Ggut_estimating2]=...
  gluc_sim_new(LO, t_c_insulin2, p_estimating, plt, ...
    ' ', ' ');
end

% SYNTH IN: objective function should return the model error:
error1 = c_glucose_estimating-data1;
if NoMean==0
  error2= c_interstitial_glucose_estimating2_mean-data2;
else
  error2= c_interstitial_glucose_estimating2-data2;
end

error3=c_glucose_labeled_estimating-data3;

Total model error:

if datasets have different number of samples, the dimensions of errorX
and error are not equal required to:
error1_new=error1;
error2_new=error2;
error3_new=error3;
data1_new=data1;
data2_new=data2;
data3_new=data3;

if length(error1)>length(error2)
  for q=1:length(error1)-length(error2)
    error2_new(length(error2)+q)=0;
  end
end

113
data2_new(length(data2)+q)=1;  
error1_new(length(error1)+q)=0;
data1_new(length(data1)+q)=1;  
error3_new(length(error3)+q)=0;
data3_new(length(data3)+q)=1;  
end

else  
for q=1:(length(error2)-length(error1))  
error1_new(length(error1)+q)=0;
data1_new(length(data1)+q)=1;  
error3_new(length(error3)+q)=0;
data3_new(length(data3)+q)=1;  
end

end

error=le4*error1+1*le4*error2+125*le4*error3;

elem(figure(100))

subplot(411)
plot(t_c_insulin1(:,l),c_interstitial_insulin_estimating,'l');
xlabel('');
ylabel('');
Title({'
...'
});
drawnow;pause(.3);

subplot(412)
plot(t_c_insulin1(:,l),c_glucose_estimating,'l',t_c_insulin1(:,l),...
data1,'l');
exlabel('');
ylabel('');
Title({'
...'
});
drawnow;pause(.3);

subplot(413)
plot(t_c_insulin1(:,l),c_glucose_labeled_estimating,'l',...
t_c_insulin1(:,l),data3,'l');
xlabel('');
ylabel('');
Title({'Title({',
      '},', '}
     );
drawnow;pause(.3);

figure(1100)
subplot(414);
if NoMean=0
    plot(t_c_insulin2(:,1),c_interstitial_glucose_estimating2_mean,’-‘,...
         t_c_insulin2(:,1),data2,’-‘);
else
    plot(t_c_insulin2(:,1),c_interstitial_glucose_estimating2,’-‘,...
         t_c_insulin2(:,1),data2,’-‘);
end
xlabel('');
ylabel('');
Title({'Title({',
      '},', '}
     );
drawnow;pause(.3);
end
D.3 worstcase bias

\textbf{worstcase\_bias\_ss.m}

\begin{verbatim}
function \{WorstCase_Bias, estimation\} = worstcase_bias_SS()

%     
% This script calculates the worst-case impact on a Model of glucose
%     
% History
% 25 05 05, Salvador Alonso Frutos, First

% Function:
% \{WorstCase_Bias\} = worstcase_bias_SS
% Inputs: function: none
% Outputs: function:
% WorstCase_Bias: Estimated worst-case impact on the identifiable parameters

clear

gluc_param_new;
Nparameters=length(p_true);
NoMean=0; % Calculates the mean over 1000 samples of unidentifiable glucose
samples=1; if NoMean==0

diary %
% Set the name of the file to save the estimated parameters:
filename='unknown_parameters.dat';

p = [1b Gb p2 DS Beta k p5 p6 Sicap SILiver Coeff];
% Given optimal input
DS=75/10; % Real input, lumps unknown
% Given optimal input
p_init = [Kabs V p4 Sglut4];

% sample schedules with constant glucose inputs to final

end
\end{verbatim}
c_insulin1_interpl = interp1(t_c_insulin1(:,1), t_c_insulin1(:,2), ...
    t_insulin1_interpl);
t_c_insulin1_interpl(:,1) = t_insulin1_interpl;
t_c_insulin1_interpl(:,2) = c_insulin1_interpl;

t_insulin1 = 1:l: t_c_insulin1(end, 1)
    c_insulin1_interpl = interp1(t_c_insulin1(:,1), t_c_insulin1(:,2), ...
    t_insulin1_interpl);  
    t_c_insulin1_interpl(:,1) = t_insulin1_interpl;
    t_c_insulin1_interpl(:,2) = c_insulin1_interpl;

% simulation using data
[c_glucose_model, c_glucose_labeled_model, ... 
  c_interstitial_insulin_model, ... 
  c_interstitial_glucose_model, c_Ggut_model] = ... 
  gluc_sim_new(LO, t_c_insulin1, p_true, plt, ... 
  'amount');

if NoMean == 0
% interpolation of data
    t_insulin2_interpl = t_c_insulin2(1, 1):1:t_c_insulin2(end, 1);
    c_insulin2_interpl = interp1(t_c_insulin2(:,1), t_c_insulin2(:,2), ...
    t_insulin2_interpl);
    t_c_insulin2_interpl(:,1) = t_insulin2_interpl;
    t_c_insulin2_interpl(:,2) = c_insulin2_interpl;

% interpolation of data
[c_glucose_model2_interpl, c_glucose_labeled_model2_interpl, ... 
  c_interstitial_insulin_model2_interpl, ... 
  c_interstitial_glucose_model2_interpl, ... 
  c_Ggut_model2_interpl, c_Ggut_model2_labeled_interpl] = ... 
  gluc_sim_new(LO, t_c_insulin2_interpl, p_true, plt, ... 
  'amount');

% calculation the mean of the interpolation data of the data
Glb = LO(4);
    c_interstitial_glucose_mean(1, 1) = Glb;
    i = 2;
    for r = t_c_insulin2(2:end, 1)' 
        c_interstitial_glucose_mean(i, 1) = ...
        sum(c_interstitial_glucose_model2_interpl(r+1-t_exp:r))/t_exp;
        i = i + 1;
    end
else
% simulation using data
[c_glucose_model2, c_glucose_labeled_model2, ... 
  c_interstitial_insulin_model2, ...
c_interstitial_glucose_model12,c_Ggut_model12,c_Ggut_labeled_model12]
=gluc_sim_new (L0,t_c_insulin2,p_true,plt,' ');
end

% adding to c

data1_new=t_insulin1;
data1_new=c_glucose_model;
data1_new=c_interstitial_glucose_model;
data1_new=c_interstitial_glucose_mean;
data1_new=c_interstitial_glucose_model12;
data1_new=c_interstitial_glucose_mode12;
data1_new=c_interstitial_glucose_labeled_model;
data1_new=c_Ggut_model;
data1_new=resnorm;
data1_new=residual;
data1_new=varp..
data1_new=stdp
=gluc_omm_mls_new(data1_new,LO,t_c_insulin1,t_c_insulin2,a,...
b,c,sigma1,sigma2,sigma3,p_fix,p_init,t_exp,Nparameters,NoMean,plt);
estimator
p_Model=p_model(estimator) ;

p_init_copy=p_init;
b=1;
estimation(1,:)=[-0.5 -0.4 -0.3 -0.2 -0.1 0.1 0.2 0.3 0.4 0.5 ];
estimation1(1,:)=estimation1(1,:);
estimation2(1,:)=estimation2(1,:);
estimation3(1,:)=estimation3(1,:);
estimation4(1,:)=estimation4(1,:);
for kk=1:length(p_init)
l1=0;
for zz=estimation(1,:);
   p_init=p_init_copy;
l1=l1+1;

   estimator=find(p_init(kk)==p_true);
estimator1=find(p_init(1)==p_true);
estimator2=find(p_init(2)==p_true);
estimator3=find(p_init(3)==p_true);
estimator4=find(p_init(4)==p_true);

   p_True=p_true(estimator);
p_init(kk)=pp_init(kk) + zz*p_init(kk)
end
[p_model,LO_model,c_glucose_model,c_glucose_labeled_model,...
c_interstitial_insulin_model,...
c_interstitial_glucose_model,...
c_Ggut_model,resnorm,residual,varp...,
,estimator]=gluc_omm_mls_new(data1_new,LO,...
t_c_insulin1,t_c_insulin2,a,...
b,c,sigma1,sigma2,sigma3,p_fix,p_init,t_exp,Nparameters,NoMean,plt);
estimator
p_Model=p_model(estimator) ;
estimation(kk+1, ll) = p_model(estimator);
estimation1(kk+1, ll) = p_model(estimator1);
estimation2(kk+1, ll) = p_model(estimator2);
estimation3(kk+1, ll) = p_model(estimator3);
estimation4(kk+1, ll) = p_model(estimator4);
WorstCase_Bias(b, :) = max(abs(p_true(estimator) - estimation(kk+1, :)));
b = b + 1;

TARGET = fopen('filename', 'w');
fprintf(TARGET, ...
', estimation1');
fprintf(TARGET, ...
', estimation2');
fprintf(TARGET, ...
', estimation3');
fprintf(TARGET, ...
', estimation4');
fclose(TARGET);
D.4 A priori parameter sensitivity

\textbf{param_sens\_new.m}

\begin{verbatim}
function [varp_matrix]=param_sens_new()

% General Minimal Model of glucose level (JU)
% This function analyzes the a priori parameters used
% knowing the estimation procedure together with the optimal input.
% It varies the a priori parameter \( \gamma, \delta, \varepsilon, \eta, \zeta \) and \( \theta^* = 1 \)
% 1)
% 2)
% History
% 07/July/06, Salvatore Giacinto, Ilc
% Function:
% [varp_matrix]=param_sens_new
% Inputs function:
% \( \gamma, \delta, \varepsilon, \eta, \zeta, \theta^* = 1 \)
% Outputs function:
% varp_matrix: Characterization of the different variations of the a priori parameters by \( \gamma, \delta, \varepsilon, \eta, \zeta, \theta^* = 1 \)

% Initialization ...

clear;

\textbf{gluc\_param\_new;}
\textbf{Nparameters=length(p\_true);}
\textbf{NoMean=0; \textbf{Calculates the mean and standard of interstitial glucose Samples at NoMean=0}}

\% sets the name of the file ...
\textbf{filename='diary.txt';}
\textbf{diary \textbf{('filename');}}

\%p = [Ib, Gb, p2 DS Beta k p5 p6 S1 Silver Coef];
\%2 unknown parameters
\textbf{p\_fix = [Ib \textbf{Gb p2 DS Beta k p5 p6 S1 Silver Coef]};}
\% using optimal input
\textbf{DS=75/10; \textbf{DS\_opt} \textbf{ Optimal \textbf{}}}
\% unknown model parameters
\textbf{p\_init = [1.4*Kabs 1.4*V 1.5*p4 1.5*S1glut4];}
\% sample schedule with a frequency of 5 minutes for data ...
\textbf{t\_insulin\_interpl = t\_c\_insulin1(l,1):t\_c\_insulin1(end,1); \textbf{[min]}}
\end{verbatim}
c_insulin1_interpl=interp1(t_c_insulin1(:,1),t_c_insulin1(:,2),...
  t_insulin1_interpl);

  t_c_insulin1_interpl(:,1)=t_insulin1_interpl;
 t_c_insulin1_interpl(:,2)=c_insulin1_interpl;

  simulation using data:
[c_glucose_model_interpl,c_glucose_labeled_model_interpl,...
  c_interstitial_insulin_model_interpl,...
  c_interstitial_glucose_model_interpl,c_Ggut_model_interpl]=...
  gluc_sim_new (L0,t_c_insulin1_interpl, p_true,plt,...
  ...);

if NoMean==0
  interpolation data2:
  t_insulin2_interpl=t_c_insulin2(1,1):t_c_insulin2(end,1);
  c_insulin2_interpl=interp1(t_c_insulin2(:,1),t_c_insulin2(:,2),...
  t_insulin2_interpl);
  t_c_insulin2_interpl(:,1)=t_insulin2_interpl;
  t_c_insulin2_interpl(:,2)=c_insulin2_interpl;

  simulation using data2:
[c_glucose_model2_interpl,c_glucose_labeled_model2_interpl,...
  c_interstitial_insulin_model2_interpl,...
  c_interstitial_glucose_model2_interpl,...
  c_Ggut_model2_interpl,c_Ggut_model2_labeled_interpl]=...
  =gluc_sim_new (L0,t_c_insulin2_interpl,p_true,plt,'...');

  calculating the mean of the first simulated data of the data2
  sample:
  Glb=L0(4);
  c_interstitial_glucose_mean(1,1)=Gl;b;
  i=2;
  for r=t_c_insulin2(2:end,1),'
    c_interstitial_glucose_mean(:,1)=...
    sum(c_interstitial_glucose_model2_interpl(r+1-t_exp:r))/t_exp;
    i=i+1;
  end
else
  simulation using data:
[c_glucose_model2,c_glucose_labeled_model2,...
  c_interstitial_insulin_model22,...
  c_interstitial_glucose_model2,c_Ggut_model2,c_Ggut_labeled_model22]...
  =gluc_sim_new (L0,t_c_insulin2,p_true,plt,'...');
end

  writing files:
  datat_new=t_insulin1_interpl;
  data1_new=c_glucose_model_interpl;
  data1_new=data1_new+0.05 * randn(length(data1_new),1);,.
  if NoMean==0
    data2_new=c_interstitial_glucose_mean;
  else
    data2_new=data2_new+0.10 * randn(length(data2_new),1);,.
  

121
else
  data2_new=c_interstitial_glucose_model2;
  data2_new=data2_new+0.10 * randn(length(data2_new),1);
end

data3_new=c_glucose_labeled_model_interpl;

t=1:

Calculating the estimate of the true mean at various points of the
prior parameter by 0.1, -0.1, 0.1, -0.1, 0.1, 0.2, and 0.3.

p_fix_copy=p_fix;
estimation(1,:)=[-0.3 -0.2 0.05 0.1 0.2 0.3];
estimation1(1,:)=estimation(1,:);
estimation2(1,:)=estimation(1,:);
estimation3(1,:)=estimation(1,:);
estimation4(1,:)=estimation(1,:);
for kk=1:length(p_fix)
  ll=0;
  p_fix=p_fix_copy;
  for zz=estimation(1,:)
    p_init_saved = [Kabs V p4 SIglut4];
estimator1=find(p_init_saved(1)==p_true)
estimator2=find(p_init_saved(2)==p_true)
estimator3=find(p_init_saved(3)==p_true)
estimator4=find(p_init_saved(4)==p_true)
    ll=ll+1;
    p_fix(kk)=p_fix(kk)+ zz*p_fix(kk)
    [p_model,LO_model,c_glucose_model,c_glucose_labeled_model,c_interstitial_insulin_model,...
    c_interstitial_glucose_model,c_gut_model,resnorm,residual,varp,...
    stdp]=gluc_omm_mls_new(data1_new,data2_new,data3_new,LO,...
    t_c_insulin1_interpl,t_c_insulin2,a,b,c,sigma1,sigma2,sigma3,...
    p_fix,p_init,t_exp,Nparameters,NoMean,plt);
    estimation1(kk+1,ll)=p_model(estimator1);
estimation2(kk+1,ll)=p_model(estimator2);
estimation3(kk+1,ll)=p_model(estimator3);
estimation4(kk+1,ll)=p_model(estimator4);
    varp_matrix(kk,ll)=det(varp)
  end
end

TARGET=fopen(filename,');
fprintf(TARGET,...
  ',estimation1');
fprintf(TARGET,...
fprintf(TARGET, ... 'estimation2');
fprintf(TARGET, ...
'estimation3');
fprintf(TARGET, ...
'estimation4');
fclose(TARGET);
varp_matrix=[p_fix' varp_matrix]
D.5 Optimal Input

**opt_input_new.m**

```matlab
function [DS_T_opt,DS_matrix]=opt_input_new()

% Nominal Model of the true system
% Use these as initial values for the current optimisation
% Also, use them as the true values for the validation

% History
% 10 July 06, Salvador sheet notes, like
% % Function:
% [DS_T_opt,DS_matrix] opt_input_new

% Input function:
% ---

% Output function:
% ---

% GLUC_PARAM new;
Nparameters=length(p_true);
NoMean=0;
% As calculate the mean over all samples at interstitial glucose
% samples:
plt=0;
% make a plot

% Sets the name of the file to store the estimated parameters:
filename='diary';
diary = diary(filename);

% p_est = [lb, ub, pc1, pc2, pc3, pc4, pc5, pc6, pc7, pc8];
sliver, slgut, coeff];
% p_init = [1.4*Kabs 1.4*V 1.5*p4 1.5*SIglut4];
% A sample schedules with a frequency of 1 minute for data
% t_insulin1_interpl = t_c_insulin1(1,1):t_c_insulin1(end,1);
t_c_insulin1_interpl=interpl(t_c_insulin1(:,1),t_c_insulin1(:,2),...
    t_insulin1_interpl);
    t_c_insulin1_interpl(:,1)=t_insulin1_interpl;
    t_c_insulin1_interpl(:,2)=c_insulin1_interpl;

% Also, the with the same procedure
% t_insulin2_interpl = t_c_insulin2(1,1):t_c_insulin2(end,1);
t_c_insulin2_interpl=interpl(t_c_insulin2(:,1),t_c_insulin2(:,2),...
t_insulin2_interpl;
t_c_insulin2_interpl(:,1)=t_insulin2_interpl;
t_c_insulin2_interpl(:,2)=c_insulin2_interpl;

% end initialization

% different values for DS, time vector (tvec) and DS
DS_vector=[1:1:100]/10;      % values for test 15 (see differential equations)
kkz=0;

for z=1:length(DS_vector)
    kkz=kkz+1
    p_init_saved = [Kabs V p4 SIglut4];
    estimator1=find(p_init_saved(1)==p_true)
    estimator2=find(p_init_saved(2)==p_true)
    estimator3=find(p_init_saved(3)==p_true)
    estimator4=find(p_init_saved(4)==p_true)

    change DS
    DS=DS_vector(z)
    p_true(4)=DS;
    p_fix = [Ib Gb p2 DS Beta k p5 p6 SIcap SILiver Coeff];

    [c_glucose_model_interpl,c_glucose_labeled_model_interpl, ...
     c_interstitial_insulin_model_interpl,.....
     c_interstitial_glucose_model_interpl,c_Ggut_model_interpl]=...
     gluc_sim_new(L0,t_c_insulin2_interpl, p_true,plt,...
     'NoMean==0');

    i=2;
    % simulation using data

    [c_glucose_model2_interpl,c_glucose_labeled_model2_interpl, ...
     c_interstitial_insulin_model2_interpl, ...
     c_interstitial_glucose_model2_interpl, ...
     c_Ggut_model2_interpl,c_Ggut_model2_labeled_interpl]=....
     gluc_sim_new (L0,t_c_insulin2_interpl2,p_true,plt, 'NoMean==0');

    GIb=L0(4);
    c_interstitial_glucose_mean(1,1)=GIb;

    for r=t_c_insulin2(2:end,1)
        i=2;
        c_interstitial_glucose_mean(i,1)=...
        sum(c_interstitial_glucose_model2_interpl(r+1-t_exp:r))/t_exp;
i=i+1;
end
else
    % simulation using data
    [c_glucose_mode12, c_glucose_labeled_mode12, ...
     c_interstitial_insulin_mode12, ...
     c_interstitial_glucose_mode12, c_Ggut_mode12, c_Ggut_labeled_mode12]...
    =gluc_sim_new (LO,t_c_insulin2,p_true,plt,' ');
end

% adding of noise
data1_new=t_insulin1_interpl;
data1_new=data1_new+0.025 * randn(length(data1_new),1);
end
data2_new=c_interstitial_glucose_mean;
data2_new=data2_new+0.05 * randn(length(data2_new),1);
else
data2_new=c_interstitial_glucose_model2;
data2_new=data2_new+0.05 * randn(length(data2_new),1);
end
data3_new=c_glucose_labeled_model_interpl;

% plot of data with noise
i: plt=1
figure;
Gb=p_true(2);
subplot (311); plot (datat_new,datal_new, ...);
hold on;
plot (t_c_insulin1_interpl(:,1),c_glucose_model_interpl, ...);
hold on;
plot ( [datat_new(1) datat_new(end)], [Gb Gb], ...);
xlabel('
' ...
'
' ...);
ylabel('
' ...
'
' ...);
Title( ...
'
' ...);

if NoMean==0
subplot(312); plot(t_c_insulin2(:,1), data2_new, ...);
else
subplot(322); plot(t_c_insulin2(:,1), data2_new, ...);
end
xlabel('\\text{Title}'); ylabel...

\begin{verbatim}
xlabel('', '');
ylabel('', '');

Title...

subplot(311); plot(datat_new, data3_new, 'r', 2);
hold;
plot(t_c_insulin1_interpl(:,1), c_glucose_labeled_model_interpl, 'r', 2);
hold;
plot([datat_new(1) datat_new(end)], [Gb Gb], ...

xLabel('', '');
yLabel('', '');

Title...

\end{verbatim}

end

\begin{verbatim}
\text{estimating the model parameters:}
[p_model, L0_model, c_glucose_model, c_glucose_labeled_model,...
c_interstitial_insulin_model,...
c_interstitial_glucose_model, c_Gut_model, resnorm, residual, varp,...
stdp]=gluc_omm_mls_new(datat_new, data2_new, data3_new, L0,...
t_c_insulin1_interpl, t_c_insulin2, a, b, c, sigma1, sigma2, sigma3, p_fix,...
p_init, t_exp, Nparameters, NoMean, plt);

if z==1
 varp_old=varp
 DS_T_opt=DS_vector(z)
else
    if det(varp)<det(varp_old)
        varp_old=varp;
        DS_T_opt=DS_vector(z)
    else
        DS_T_opt=DS_vector(z)
    end
end

DS_matrix(z,:)=[DS, det(varp)];

estimation1(kkz)=p_model(estimator1);
estimation2(kkz)=p_model(estimator2);
estimation3(kkz)=p_model(estimator3);
estimation4(kkz)=p_model(estimator4);

end

TARGET=fopen(filename, 'w');
fprintf(TARGET, '  
  estimation1');
fprintf(TARGET, '  
  estimation2');
fprintf(TARGET, '  
  estimation3');
fprintf(TARGET, '  
  estimation4');
fclose(TARGET);
D.6 Optimal Sample Schedule

opt_sample_schedule_new.m

function [opt_SS_data1, opt_SS_data2, opt_data1, opt_data2, ...
, opt_data3] = opt_sample_schedule_new()

% Initialization
clear
NoMean=0; % Calculates the mean of the samples at the end of the glucose
Max_Exp_Dur=240; % The maximum duration of the experiment can take
gluc_param2_new;
Nparameters=length(p_true);

% Using optimal sample
DS=75/10

\[ p_{\text{fix}} = \{10 \text{ Gp2 DS Beta k p5 p6 SIcap SIilver Coeff}\}; \]
\[ p_{\text{init}} = \{1.4*Kabs 1.4*V 1.4*p4 1.4*SIglut4\}; \]

Calculating model using a couple of loops with a frequency of 1 minute

\[ \text{plt}=0; \]

\[ \text{t}\_\text{insulin1}\_\text{interp1} = \]
\[ \text{t}\_\text{c}\_\text{insulin1}(1,1):1:\text{Max}\_\text{Exp}\_\text{Dur}; \]
\[ \text{c}\_\text{insulin1}\_\text{interp1}\_\text{interp1}(\text{t}\_\text{c}\_\text{insulin1}(,1),\text{t}\_\text{c}\_\text{insulin1}(,2),...\]
\[ \text{t}\_\text{insulin1}\_\text{interp1}); \]
\[ \text{t}\_\text{c}\_\text{insulin1}\_\text{interp1}(1,1)=\text{t}\_\text{insulin1}\_\text{interp1}; \]
\[ \text{t}\_\text{c}\_\text{insulin1}\_\text{interp1}(1,2)=\text{c}\_\text{insulin1}\_\text{interp1}; \]

\[ \text{c}\_\text{glucose}\_\text{model}\_\text{interp1},\text{c}\_\text{glucose}\_\text{labeled}\_\text{model}\_\text{interp1},...\]
\[ \text{c}\_\text{interstitial}\_\text{insulin}\_\text{model}\_\text{interp1},....\]
\[ \text{c}\_\text{interstitial}\_\text{glucose}\_\text{model}\_\text{interp1},\text{c}\_\text{Ggut}\_\text{model}\_\text{interp1}] =...\]
\[ \text{gluc}\_\text{sim}\_\text{new}(\text{L0},\text{t}\_\text{c}\_\text{insulin1}\_\text{interp1},\text{p}_{\text{true}},\text{plt},...\]
\[ \ldots); \]

\[ \text{t}\_\text{insulin2}\_\text{interp1} = \]
\[ \text{t}\_\text{c}\_\text{insulin2}(1,1):1:\text{Max}\_\text{Exp}\_\text{Dur}; \]
\[ \text{c}\_\text{insulin2}\_\text{interp1}=\text{interp1}(\text{t}\_\text{c}\_\text{insulin2}(,1),\text{t}\_\text{c}\_\text{insulin2}(,2),...\]
\[ \text{t}\_\text{insulin2}\_\text{interp1}); \]
\[ \text{t}\_\text{c}\_\text{insulin2}\_\text{interp1}(1,1)=\text{t}\_\text{insulin2}\_\text{interp1}; \]
\[ \text{t}\_\text{c}\_\text{insulin2}\_\text{interp1}(1,2)=\text{c}\_\text{insulin2}\_\text{interp1}; \]

\[ \text{c}\_\text{interstitial}\_\text{glucose}\_\text{model}\_\text{interp1},\text{c}\_\text{Ggut}\_\text{model}\_\text{interp1},\text{c}\_\text{Ggut}\_\text{labeled}\_\text{model}\_\text{interp1}]...\]
\[ =\text{gluc}\_\text{sim}\_\text{new}(\text{L0},\text{t}\_\text{c}\_\text{insulin2}\_\text{interp1},\text{p}_{\text{true}},\text{plt},' ...'); \]

\[ \text{datat}\_\text{new}\_\text{t}\_\text{insulin1}\_\text{interp1}; \]
\[ \text{datat}\_\text{new}\_\text{c}\_\text{glucose}\_\text{model}\_\text{interp1}(' \text{measure}\_\text{interp1},' \text{linear}'); \]
\[ \text{datat}\_\text{new}\_\text{c}\_\text{interstitial}\_\text{glucose}\_\text{model}\_\text{interp1}(' \text{measure}\_\text{interp1},' \text{linear}'); \]

129
data2_new=c_interstitial_glucose_model2;

Data2_new shows the interstitial glucose level, measured...

endvalue=le6;

for q=1:t_exp:endvalue
    for z=[aa]
        qz=q+z
    
    % Calculation of the optimal schedule...
    % Algorithm based on data, with a prespecified sample schedule where each individual sample is optimally approximated within the
    % Right-hand part shows a visual representation of the sampled schedule...
}

datatl_using=opt_SS_data1;
datatl_usinggg=opt_SS_data1'
datatl_using=opt_datal;
data3_using=opt_data3;
datatl_using_old=datatl_using';

if datatl_using(q/t_exp+1)>Max_Exp_Dur
    display('The duration is exceeded in the search for...
    
    % The optimal sample schedule is not feasible...
\begin{verbatim}
r=r+1;
if numel(aa)==0
    display('Bad initial guess. Please specify another initial guess.
which

end

opt_SS_data2=0
opt_data2=0
return

end

datal_using_old=datal_using';
end a patient

datal_using(q/t_exp+1)=datal_new(opt_SS_data(q/t_exp+1));

while not end of new sample

data3_using(q/t_exp+1)=data3_new(opt_SS_data(q/t_exp+1));

end

datacl_using_old=datacl_using';
end

datacl_using(q/t_exp+1)=t_c_insulin_interpl(opt_SS_data(q/t_exp+1),
2);

while not end of new sample

datacl_using_new=datacl_using';
end

datacl_c_using=[datacl1_using datacl_using]

if (q/t_exp+2)<=length(datal_using)
    pause
    ww=1;
    while (q/t_exp+1+ww)<=length(datal_using)-1
        datal_using(q/t_exp+1+ww)= datal_using(q/t_exp+1+ww)+t_exp
        datacl_using(q/t_exp+1+ww)= datacl_using(q/t_exp+1+ww)+t_exp
        data3_using(q/t_exp+1+ww)= data3_using(q/t_exp+1+ww)+t_exp
        datacl_using(q/t_exp+1+ww)+ww=ww+1;
    end
end

datacl_using=datacl_using(1:length(datal_using)-1)
datacl_using=datacl USING(1:length(datacl_using)-1)
data3_using=datacl USING(1:length(data3_using)-1)
data3_c_using=datacl USING(1:length(data3 USING)-1)

end
end

end

end

end

131
\end{verbatim}
stdp]=gluc_omm_mis_new(datal_using, data2, data3_using, L0, data1_c_using, ...
t_c_insulin2,a,b,c, sigma1, sigma2, sigma3, p_fix, p_init, t_exp, ...
Nparameters, NoMean, plt);

if z==aa(1)
  varp_old=varp; % reset the initial matrix
  optimumDET=det(varp)
  opt_SS_data1=datat1_using
  opt_data1=datat1_using
  SAVED=1 % save the data of the current SS
  if det(varp)<det(varp_old)
    varp_old=varp;
    optimumDET=det(varp)
    optimumSample=qz
    opt_SS_data1=datat1_using
    opt_data1=datat1_using
    SAVED=1 % save the data of the current SS
  else
    det(varp) % display the current SS
    NOTSAVED=1 % display the current SS
  end
end
end

if 0
% calculating the optimal sample schedule for input 2
% initialising the optimal sample schedule
% selecting the optimal schedule with a nearest-neighbor sample schedule where
% each individual sample is its closest sample in the given sample schedule.
opt_data2=data2;
opt_SS_data2=t_c_insulin2(:,1);
aa=t_exp:t_exp:5*t_exp;
endvalue=le6;
r=l;

for q=1*t_exp:t_exp:endvalue
for z=[aa]
    qz=q+z
end

for q=1*t_exp:t_exp:endvalue
for z=[aa]
    datat2_using=opt_SS_data2;
data2_using=opt_data2;

    for z=[aa]
        qz=q+z
    end

    SumOfData=0;
    if NoMean==0
        for pp=1:1:t_exp
            SumOfData=SumOfData+data2_new(opt_SS_data2(q/t_exp)+z+1-(pp));
        end
        Meandata=1/t_exp*SumOfData;
        data2_using(q/t_exp+1)=Meandata; saving data at new sample.
    else
        data2_using(q/t_exp+1)=data2_new(opt_SS_data2(q/t_exp)+z);
    end

    datat2_using_old=datat2_using;
data2_using(q/t_exp+1)=opt_SS_data2(q/t_exp)+z;

    if datat2_using(q/t_exp+1)>=Max_Exp_Dur
        display('Max_Exp_Dur exceeded by the experiment');
        aat_expit_exp:(5-r)*t_exp; the length of the horizon must ...
        at least 5 at the first level input;
        r=r+1;
    if numel(aa)==0
        display('All data not estimated in this for loop');
        opt_SS_data1=0
        opt_SS_data3=0
        opt_data1=0
D.7 White Noise test

wnoisetest_new.m

function [datacov,mean_data1,mean_data2,mean_data3]=...
    wnoisetest_new(data1,data2,data3,residual)

%%
%% Time Minimal Model of glucose kinetic pool
%% This function analyzes the properties of data used on residuals left
%% during the estimation procedure.
%%
%%
%% History
%% 07 July 06, Salvador Almagro Brunet, TUM
%%
%% Function:
%% [datacov,mean_data1,mean_data2] = wnoisetest(data1,data2,residual)
%%
%% Inputs:
%% data1: data of interstitial insulin [pM]
%% data2: data of interstitial glucose [mM]
%% data3: data of labeled plasma glucose [mM].
%% residual: the value of the residual of the estimation algorithm.
%%
%% Outputs:
%% datacov: crosscorrelation of the data.
%% mean_data1: calculating the mean of the data. Must be zero.
%% mean_data2: calculating the mean of the data. Must be zero.
%% mean_data3: calculating the mean of the data. Must be zero.

% TEST: calculating the mean. Must be zero.
mean_data1=mean(data1);
mean_data2=mean(data2);
mean_data3=mean(data3);

% TEST: Statistical properties: gaussian distribution with zero mean
pp=-1:0.01:1;
yy=hist(residual,pp);
figure;
plot(pp,yy);title('');xlabel('');ylabel('');

% normalizing residuals
residual=residual./max(residual);

% TEST: crosscorrelation. Must be the form of a pulse at zero
% timeshift:
for m=0:(length(residual)-round(0.4*length(residual)))
    sum=0;
for n=1:(round(0.4*length(residual)))
    part=residual(n)*residual(n+m);
    sum=sum+part;
end

datacov(m+1)=(1/(round(0.4*length(residual)))).*sum;
end

figure
plot((0:(length(residual)-round(0.4*length(residual)))),datacov);
title('');
xlabel('');
ylabel('');
D.8 Worst case bias $S_{\text{cap}}$

D.8.1

\texttt{worstcase\_bias\_steil.m}

```matlab
%%
% Oral Minimal Model of glucose kinetics.
% This script estimates the identifiable parameters in Steil's
% Model of insulin kinetics using Maximum Least
% Squares Estimation
%%

% History
% 28 July-06, Salvador Almagro Frutos, TFC
% 2%

% Initializes all parameters of Steil's Model of insulin kinetics
% and all time samples of plasma insulin.

clear
gluc_param_new;
Nparameters=length(p_true);
NoMean=0; % Calculates the mean over $t_{\text{exp}}$ minutes of interstitial glucose
% Samples if NoMean=0

diary
% Sets the names of the mat files containing the estimated parameters:
filename='...';

% make plots if plt=1
plt=1;

% NoMean==0
  % Interpolating data2
  t_insulin2_interpl = t_c_insulin2(1,1):t_c_insulin2(end,1);
  c_insulin2_interpl=interpl(t_c_insulin2(:,1),t_c_insulin2(:,2),...
                             t_insulin2_interpl);
  t_c_insulin2_interpl(:,1)=t_insulin2_interpl;
  t_c_insulin2_interpl(:,2)=c_insulin2_interpl;

  % simulation using data2
  [c_glucose_mode12_interpl,c_glucose_labeled_mode12_interpl,...
   c_interstitial_insulin_mode12_interpl,...
   c_interstitial_glucose_mode12_interpl,...
   c_Ggut_mode12_interpl,c_Ggut_mode12_labeled_interpl,...
   gluc_sim_new (LO,t_c_insulin2_interpl,p_true,plt,'...');
```

137
% calculating the mean of the interpolated data at the data2 samples
GIb=LO(4);
c_interstitial_glucose_mean(1,1)=GIb;
i=2;
for r=t_c_insulin2(2:end,1)';
c_interstitial_glucose_mean(i,1)=...
sum(c_interstitial_glucose_model2_interpl(r+1-t_exp:r))/t_exp;
i=i+1;
end
else
% simulation using data2
[c_glucose_model2,c_glucose_labeled_model2,...
c_interstitial_insulin_model2,...
c_interstitial_glucose_model2,c_Ggut_model2,c_Ggut_labeled_model2]...
=gluc_sim_new (LO,t_c_insulin2,p_true,p1t,'');
end

% adding noise
if NoMean==0
data2_new=c_interstitial_insulin_model2_interpl;
%data2_new=data2_new*0.10 * randn(size(data2_new),1); %+8% RUTS
else
data2_new=c_interstitial_insulin_model2;
%data2_new=data2_new*0.10 * randn(size(data2_new),1); %+8% RUTS
end

% make plots i = plt-1
plt=1;

stell_param;
Nparameters=length(p_true);
% End initialization

% 2-dimensional parameter space
%p: = [1b Sfcap p2];
% known parameters:
p_fix = [1b p2];
% unknown model parameters:
p_init = [Sfcap]

p_init_copy=p_init;
estimation(1,:)=[-0.5 -0.4 -0.3 -0.2 -0.1 0.1 0.2 0.3 0.4 0.5 ];
estimation1(1,:)=estimation(1,:);

for kk=1:length(p_init)
  ll=0;
  for zz=estimation(1,:)
    p_init=p_init_copy;
    ll=ll+1;
    estimator=find(p_init(kk)==p_true);
  end
  % code for estimating
end
estimator1 = find(p_init(l) == p_true);
P_True = p_true(estimator)
P_init(kk) = p_init(kk) + zz * p_init(kk)

Estimation of parameters:

[p_model, L0_model, c_interstitial_insulin_model, resnorm, residual, varp, ...]

stdp = gluc_omm_mls_steil(data2_new, L0, t_c_insulin2_interpl, a, sigmal, ...
 p_fix, p_init, Nparameters, plt);

p_Model = p_model(estimator)

estimation(kk+1, ll) = p_model(estimator);
estimation1(kk+1, ll) = p_model(estimator1);
end

TARGET = fopen(filename, 'w');
fprintf(TARGET, ...
'... ... ...
... ... ...
... ... ...
... ... ...
... ... ...
... ... ...
', estimation1);
fclose(TARGET);
D.8.2

steil_param.m

%%% % Steil's Model of insulin kinetics.
%%% This script initializes all parameters of Steil's Model of insulin kinetics and the time samples of plasma insulin.
%%% %
%%% % History
%%% 28-July-06, Salvador Almagro Frutos, Ute
%%% 29-Jan-04, Natal van Riel, TU/e
%%% 17-Jun-03, Natal van Riel, TU/e
%%% %

%model parameters:
%Values from: ????????
Ib = Ib; % [pM] baseline insulin concentration in plasma
Sicap = Sicap; % [1/min] trans-endothelial insulin transport parameter
p2 = p2; % [1/min] total fractional insulin clearance from the interstitial fluid
p_true = [Ib, Sicap, p2]; % 3-dimensional parameter space

% Fixed initial conditions
clear I;
L0(1) = 0; % [pM] insulin concentration in interstitium

% Input insulin concentration in plasma [pM]; assumed to be known at each simulation time sample from linear interpolation of its measured samples

% Time and amplitude samples of plasma insulin:
% -> blood glucose data
t_insulin1 = t_c_insulin1(:,1);
c_insulin1 = t_c_insulin1(:,2);
t_c_insulin1(:,1) = t_insulin1';
t_c_insulin1(:,2) = c_insulin1';

% squared 2-norm of the error-function residuals
a=1; % output weighting factor of plasma glucose concentration (lit output)
sigma=1; % variance of plasma glucose concentration (lit output)

% Plots
% make plots if plt=1
if plt==1;
% Plots
figure; plot(t_c_insulin1(:,1), t_c_insulin1(:,2), 'r'); hold

% Baseline level:
plot(t_c_insulin1(1,1), t_c_insulin1(end,1), [Ib Ib], 'r'); ...
1.5)
xlabel('Time (min)'); ylabel('Insulin (mU/L)'); title('
end

141
D.8.3

steil_sim.m

function [c_interstitial_insulin]=steil_sim
(L0,t_c_insulin,p,plt,TITLE)

%%
%% Steil's Model of insulin kinetics.
%% This function makes a simulation of the Steil’s Model of insulin kinetics using the parameters defined in steil_param.m.
%%
%% Function:
% c_interstitial_insulin=steil_sim (L0,t_c_insulin,t_c_insulin2,p,...
% plt,TITLE)
%%
%% Inputs function:
% L0: Fixed initial conditions, [0] - [pM] insulin concentration in interstitium.
% t_c_insulin: [pM] [min] Time and amplitude samples matrix of plasma insulin 1 - blood glucose data 2 - interstitial glucose data
% p: = [Sticap, p2] 2 dimensional parameter space
% plt: plots are made in this function if plt=1.
% TITLE: set the title name if plots are made in this function (plt=1).
%%
%% Outputs function:
% c_interstitial_insulin: [pM] Interstitial insulin concentration.
%%
%%
%% This model parameters:
% Sticap [1/min] transendothelial insulin transport parameter
% p2 [1/min] total fractional insulin clearance from the interstitial fluid

% Differential equations solver: ode45s
ode_options =odeset('RelTol',1e-6,'AbsTol',1e-12); % parameters 2-end
ode_options =odeset('RelTol',1e-6,'AbsTol',1e-6);

[t,L] = ode45(@steil_ode,t_c_insulin(:,1),L0,ode_options,
t_c_insulin,p);

% Output:
% c_interstitial_insulin=L(:,1); % [pM] Interstitial insulin concentration
% make plots if plt=1

142
plt==1

figure (333);
Ib=p(1);
subplot(221); plot(t_c_insulin(:,1),t_c_insulin(:,2), ;' ,
', h'; ,2);
hold ';
%Baseline level:
plot([t_c_insulin(1) t_c_insulin(end)], [Ib Ib], ;',
', 1.5)
ylabel('');
xlabel('');
Title('');

subplot(222); plot(t_c_insulin(:,1), c_interstitial_insulin,
', ' , , h'; ,2)
xlabel(''); ylable...
', ('); Title(TITLE);

end
D.8.4

steil_ode.m

function dLout = steil_ode(t, Lin, t_c_insulin, p)

%%
%STEIL_ODE ODE's of STEIL'S Model of insulin kinetics.
%%This function sets STEIL_ODE ODE's of STEIL'S Model of insulin
%kinetics using the parameters defined in steil_param.m.
%%
%
%Function:
%dLout = steil_ode(t, Lin, t_c_insulin, p).
%
%Inputs function:
%Lin: state values at previous time sample L(k-1) - [X1k H1k]
%t_c_insulin: c_insulin(k) input signal at sample k) OR matrix
%p: = [SLcap, p2]; 1 2 dimensional parameter space
%
%Outputs function:
%dLout: new state derivatives dL(k) = [dX(k)];
%
%%
%%
%History
%28-Jul-06, Salvador Almago Frutos, TU/e
%29-Jan-04, Natal van Riel, TU/e
%17-Jun-04, Natal van Riel, TU/e
%%

% defining the row were each state in statevector Lin is present
idX = 1; %interstitial insulin concentration [pM]

% model parameters:
Ib = p(1); %[pM] baseline insulin concentration in plasma
SLcap = p(2); %[1/min] transendotheilal insulin transport parameter
p2 = p(3); %[1/min] total fractional insulin clearance from the
%interstitial fluid

calculate c_insulin by interpolation of t_c_insulin
u = interp1(t_c_insulin(:,1), t_c_insulin(:,2), t);

%ode's

dX = SLcap/1e3*(u-Ib)^2-p2*Lin(idX);

dLout = [dX];
glucl_omm_mls_steil.m

function [p_model,LO_model,c_interstitial_insulin_model,...
resnorm,residual,vep,stdp]=gluc_omm_mls_steil...
(datal,LO,t_c_insulinl,a,sigma,p_fix,...
p_init,Nparameters,plt)

%%
%STEIL_MLS Maximum Least Squares Estimation of Steil's Model of
%insulin kinetics
%%
%This function estimates the parameters using Maximum Least Squares
%Estimation of Steil's Model of insulin kinetics using the
%parameters defined in steil_param.m.
%%
%Function:
%[p_model,LO_model,c_interstitial_insulin_model,...
resnorm,residual,vep,stdp]=gluc_omm_mls_steil...
(datal,LO,t_c_insulinl,a,sigma,p_fix,p_init,Nparameters,plt)

%Inputs function:
%data: data of interstitial insulin [pM]
%LO: Fixed initial conditions. [3(t)] - [pM] insulin concentration in
%interstitium.
%t_c_insulin: [pM] [min] Time and amplitude samples matrix of plasma
%insulin 1 - blood glucose data.
%p: ([Slep, p2]): 2 dimensional parameter space
%p_fixed: parameter space which is fixed.
%p_init: initialisation of parameter space which will be estimated.
%windowinterval [min] of data 2.
%Nparameters: number of dimensional parameter space p.
%plt: plots are made in this function if plt=1.

%Outputs function:
%p_model: ([Slep, p2]): 2 dimensional parameter space
%c_interstitial_insulin_model: [pM] interstitial insulin concentration.
%resnorm: squared 2-norm of the error function residual.
%residual: the value of the residual of the estimation algorithm.
%varp: variance
%stdp: The standard deviation is the square root of the variance.
%%

%History
%28-July 06, Salvador Almagro Fuentes, Tübingen
%29-Jan-04, Natal van Riel, Tübingen
%17-Jun-03, Natal van Riel, Tübingen

% lower (lb) and upper (ub) bounds, so that the solution of the
% estimated
% parameters is always in the space the parameter is ub.
lb = zeros(size(p_init)); lb(3) = 0;
ub = zeros(size(p_init)); ub(3) = 1;
%
% provides the function specific details for the options values for
% lsqnonlin
options = optimset('tolfun',1e-6,'reltol',1e-20,...
'lsqnonlin','Jacobian','on');

% LSQNONLIN: objective function should return the model error
[p_estimated,resnorm,residual,exitflag,output,lambda,jacobian] =
    lsqnonlin(@obj_fn_steil,p_init,lb,ub,options,p_fix,LO,datal,...
    t_c_insulin1,a,sigma,Nparameters);

if exitflag==0
    STOP=1;
end
%
% Accuracy:
% resnorm returns the jacobian as a sparse matrix
varp = resnorm*inv(jacobian'*jacobian)/length(t_c_insulin1(:,1));
% variance
stdp = sqrt(diag(varp));
%
[p_model] = [p_fix(1) p_estimated(1) p_fix(2)];
LO_model = [LO(1)];
%
% make plots with new 2 dimensional parameter space if plt=1
[c_interstitial_insulin_model]= steil_sim (LO,...
    t_c_insulin1,p_model,plt,'');
function error = obj_fn_Steil(p_var, p_fix, LO, datal,t_c_insulinl,...
,a,sigma,Nparameters)

%%
%Objective function Maximum Least Squares Estimation of Steil's
%Model of insulin kinetics.
%%
%%This function is used as objective function for LSQNONLIN.
%%
%Function:
% obj_fn_steil(p_var, p_fix, LO, datal, t_c_insulin...
% Nparameters)
% Inputs:
% p_var: parameter space which will be estimated
% p_fix: parameter space which is fixed.
% LO: Fixed initial conditions, LO[1] = [PM] insulin concentration in ...
% interstitium.
% datal: data of interstitial insulin [PM].
% t_c_insulin: [min] time and amplitude sampled matrix of plasma
% insulin.
% a: output weighting factor of the output.
% sigma: variance of 1st output.
% Nparameters: number of dimensional parameter space p.
% estimator: defines the parameter which is used in the model to model
% adjustment identification.
% Outputs:
% error: minimization variable of MEC model error
%%
%History
% 28 July-06, Salvador Almasre Frutos, TUE
% 29 Jan-04, Natal van Riel, TUE
% 17 Jun 03, Natal van Riel, TUE

p_estimating=[p_fix(1) p_var(1) p_fix(2)];

%estimating 2-dimensional parameter space:
%p_true = [16,Ste002];

%Simulating Steil model of insulin kinetics using the estimating
%2-dimensional parameter space and the interpolated insulin
%concentration
%samples for datal.

plt=0; % make NO plots
[c_interstitial_insulin_estimating]=...
steil_sim(LG, t_c_insulin1, p_estimating, 0, ...
'. . .');

error1 = c_interstitial_insulin_estimating-datal;

error=error1;

p_var

plot(1:N, c_interstitial_insulin_estimating,' ',1:N,datal,'');
xlabel(''); ylabel(''); Title({''); . . .; . . .; drawnow;pause(.3); end