Università degli Studi di Padova

Laurea Magistrale in Ingegneria dell'Automazione

Modeling SMS driven conversion of ceramide to sphingomyelin reveals the existence of a positive feedback mechanism

A systems theoretic modeling approach

Caterina Thomaseth

Supervisors: Ch.mo Prof. Augusto Ferrante

Junior-Prof. Nicole Radde

University of Padova

Department of Information Engineering

University of Stuttgart

Institute for Systems Theory and Automatic Control

A.A. 2012/2013

11 December 2012

Abstract

Here we present a minimal mathematical model for the Sphingomyelin synthase 1 (SMS1) driven conversion of ceramide to sphingomyelin based on chemical reaction kinetics. We demonstrate, via sampling-based parameter estimation and mathematical analysis, that this model is not able to qualitatively reproduce experimental measurements on lipid compositions after altering SMS1 activities. We conclude that a positive feedback mechanism is required from the products to the reactants of the reaction, which in fact exists *in vivo* via protein kinase D and the ceramide transfer protein CERT. Accordingly, a modified model that comprises this feedback mechanism was able to reproduce experimental findings.

Sommario

In questa tesi presentiamo un modello matematico minimo per la conversione di un ceramide in sfingomielina catalizzata dall'enzima sfingomielina sintasi 1 (SMS1) basato sulle leggi della cinetica chimica. Viene dimostrato, utilizzando tecniche di sampling per la stima parametrica e metodi di analisi matematica, che questo modello non è in grado di riprodurre qualitativamente delle misure sperimentali sulla composizioni dei lipidi in seguito ad alterazione dell'attività enzimatica di SMS1. Concludiamo quindi che è necessario considerare un meccanismo di feedback positivo fra i prodotti e i reagenti della reazione, che esiste effettivamente *in vivo* tramite la proteina chinasi D e la proteina di trasporto di ceramide CERT. Di conseguenza, proponiamo un secondo modello modificato in modo da comprendere questo meccanismo di feedback, che risulta essere in grado di spiegare i risultati sperimentali.

Contents

List of Figures

List of Tables

Introduction

The process of secretion of proteins in mammalian cells is one of the most highly controlled processes of living beings, since it underlies the regulation of a lot of biochemical functions throughout the entire organism. A detailed understanding of the secretory pathway and of the underlying regulatory network is the basis for targeted intervention and is thus highly relevant for pharmaceutical applications.

The use of formal mathematical models to describe complex biochemical reaction networks is an important approach to study the properties of such biological systems, and to be able to simulate the effects of external intervention.

The main focus of this thesis is the systematic study of the functioning of a small but relevant subsystem of the secretion regulatory pathway at the trans-Golgi network: the enzymatic conversion of ceramide to sphingomyelin driven by the catalysing enzyme sphingomyelin synthase 1 (SMS1). In particular the aim is to propose an ordinary differential equation (ODE) model to formally describe the biochemical reactions under study and to partially validate the model by fitting it to a given experimental dataset, using sampling-based statistical approaches for parameter estimation. Discrepancies encountered between simulated model predictions and experimental observations have led to the formulation of a final model, which considers a particular positive feedback mechanism between two reactants of the reaction. Using mathematical analysis we demonstrate that the proposed model including such feedback regulation is sufficient to qualitatively reproduce experimental measurements on lipid compositions after altering SMS1 activities. This theoretical result is supported by an improved statistical model fit.

In Chapter 1 we introduce some general biological notions concerning the secretion of proteins. We give a brief idea of the underlying regulatory network and we highlight in particular the role of the reaction of interest in the regulation of secretion: the SMS1 driven conversion of ceramide into sphingomyelin at the trans-Golgi network. Moreover we present how chemical reactions can be modelled with ordinary differential equation systems. Chapter 2 presents in more detail the reaction of interest, and the related parameterized ODE system. From some experimental findings we can show that, considering the reaction in isolation, the simple model is not able to capture the presented experimental data, and that there is the need to develop a modified model that takes into account a positive feedback regulation between two reactants of the reaction. We conclude this Chapter with a mathematical proof of this theory, rejecting analytically the first model, while showing that the introduction of a feedback control ensures a qualitative explanation of the experimental findings. These theoretical expectations were tested against experimental data by applying, for parameter estimation, the statistical inference approach of the maximum likelihood estimation, and a sampling-based Bayesian approach, whose theory and results are presented in Chapters 3 and 4, respectively. These results confirm our theoretical investigations supporting the hypothesis of the feedback regulation, while rejecting the model without feedback. The sampling method provides also interesting results concerning model predictions and further information about the distribution of parameters, through the marginal posterior density functions.

Chapter 1

Biological context

In this Chapter we want to give a general overview of the biological context underlying this thesis, focusing on the description of the process of secretion of proteins in mammalian organisms and on the explanation of the functioning of the enzymatic reaction that metabolises sphingomyelin from ceramide, driven by the enzyme sphingomyelin synthase 1, that takes place at the trans-Golgi network. In particular we want to highlight the connection between the biochemical reaction of interest and the control of the secretory pathway.

Afterwards we want to briefly present the basic concepts of the mathematical modelling of cellular biochemical reactions by means of ordinary differential equation systems, that will be used in our subsequent analysis of Chapter 2.

1.1 Biological background

1.1.1 Regulation of secretion of proteins in mammalian cells

Most human cells secrete proteins. Secretory proteins include many hormones, enzymes, toxins, and antimicrobial peptides and they are synthesized in the endoplasmic reticulum (ER). When they are assembled and folded correctly they are transported to the Golgi apparatus by means of special vesicles. Passing through the cisternae of the Golgi apparatus these proteins are further elaborated. In particular at the trans-Golgi network proteins to be secreted are sorted and segregated from lysosomal enzymes. When they are ready for secretion, secretory proteins leave the Golgi apparatus, packed in specialized vesicles, to be transported towards the cellular membrane. Finally the vesicle membrane fuses with the cell membrane and so the proteins leave the cell. This last process of fusion of the vesicle with the plasma membrane and the following release of its contents is called *exocytosis*. More detailed informations about the process of secretion can be found in [5] and [27, Chap. 8]. In Figure 1.1 the main steps of the secretory pathway are graphically represented.

This complex secretory process is highly controlled and regulated at different stages within mammalian cells. In particular we mention an important regulation mechanism, based on the interdependence of protein kinase D (PKD) and of the ceramide transport protein CERT, which influences the formation of secretory vesicles at the trans-Golgi network (TGN). In fact PKD has been identified as a crucial regulator of the secretory transport at the TGN [9]. Recruitment and activation of PKD at the TGN is regulated by binding with the lipid diacylglycerol (DAG) [3], a pool of which is produced by sphingomyelin synthase (SMS) from ceramide (Cer) and phosphatidylcholine (PC) taking place at the TGN. The non-vesicular transfer of ceramide from the endoplasmic reticulum (ER) to the Golgi complex is mediated by the ceramide transport protein CERT [19, 20, 21, 22, 23]. Moreover CERT is critical for PKD activation and for PKD-dependent protein transport at the plasma membrane. Thus the interaction between PKD and CERT has a key role for the maintenance of Golgi membrane integrity and secretory transport [9].

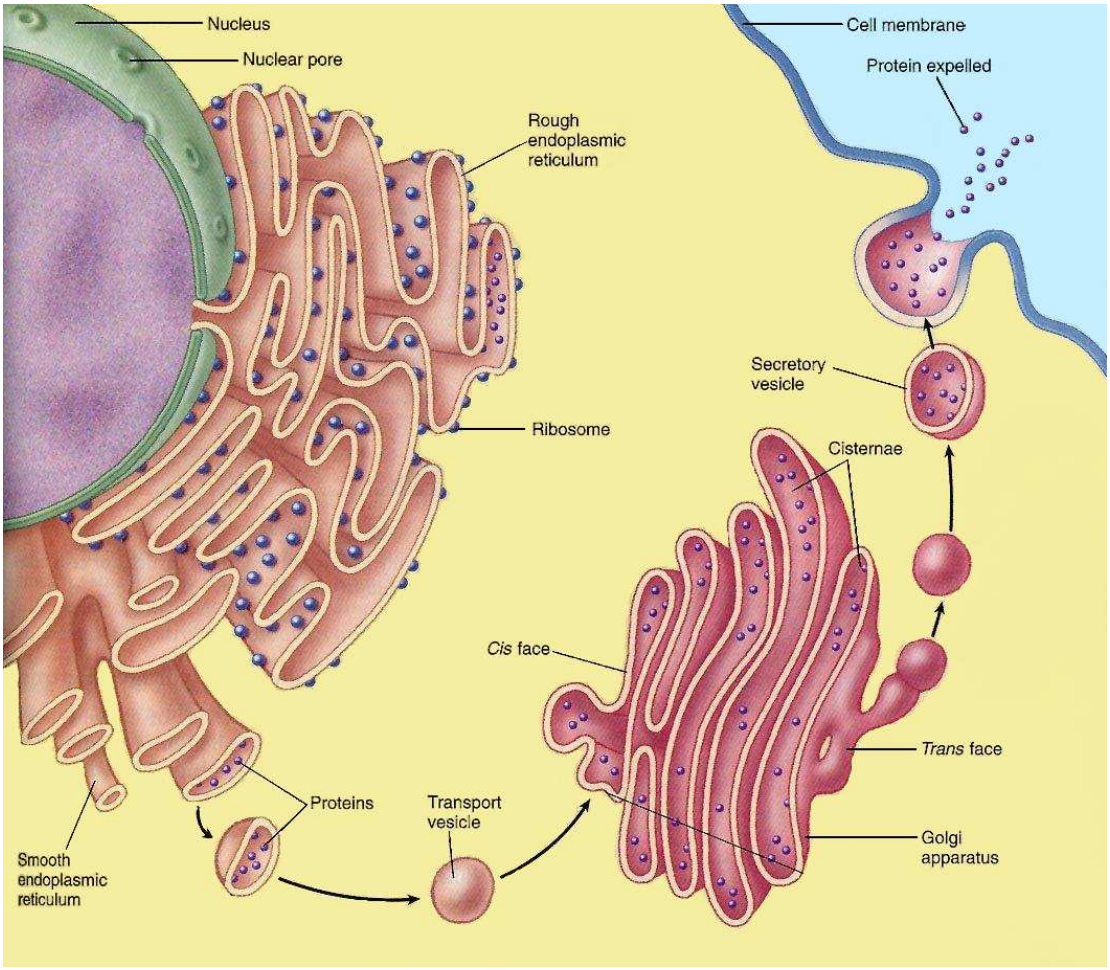

Figure 1.1: Representation of the secretory pathway within human cells: secretory proteins are synthesized in the ER, transported to the Golgi apparatus, where they are post-modified and sorted, and finally, packed in specialized vesicles, transported to the plasma membrane. Copyright ©The McGraw-Hill Companies, Inc.

1.1.2 Sphingomyelin synthase 1: SMS1 driven conversion of ceramide into sphingomyelin at the TGN

Due to the facts described at the end of the previous Subsection, we maintain that an interesting subsystem of the secretion regulatory network to analyse is the metabolic conversion of ceramide (Cer) into sphingomyelin (SM) catalysed by the enzyme sphingomyelin synthase (SMS) at the TGN, yielding diacylglycerol (DAG) as a side product [33, 37].

Each organism capable of SM production displays a multiplicity of SM synthase (SMS) genes. The mammalian genome contains two SMS isoforms, named SMS1 and SMS2. SMS1 and SMS2 are co-expressed in a wide range of human cell types and the corresponding enzymes reside in organelles where SMS synthesis is known to occur: SMS1 is localised to the Golgi, while SMS2 resides primarily at the plasma membrane [25, 44]. Moreover they operate as the key Golgi- and plasma membrane-associated SM synthases, respectively [39].

Now we want to present in detail the enzymatic reaction taking place at the trans-Golgi network, referring in this way specifically to the enzyme SMS1. In fact, in the study of the regulation of protein trafficking and secretion, we have prevalent interest in the reaction taking place at the Golgi apparatus, and in this work we will not consider the reaction localised at the cell membrane.

The enzyme "*Sphingomyelin Synthase 1*" (SMS1, UniProt identifier: Q86VZ5) consists in a transmembrane protein, with sequence length of 419 amino acids, and molecular mass of 49,208 kDa. SMS1 is an integral membrane protein of the trans-Golgi membrane [25, 39]. It is supposed that SMS1 possesses six transmembrane domains and that both the carboxy terminus and the amino terminus face the cytoplasmic side of the trans-Golgi membrane [25]. Instead the potential catalytic amino acids of SMS1 are probably oriented towards the lumen side, the exoplasmic leaflet, of the trans-Golgi membrane.

SM synthesis is mediated by a *phosphatidylcholine:ceramide cholinephosphotransferase*, i.e. SM synthase 1, and the reaction takes place in the lumen of the trans-Golgi [25]. It consists in an enzyme that catalyses the transfer of a phosphocholine head group from phosphatidylcholine (PC) to ceramide, thus generating SM and DAG [25, 38]. Moreover SMS1 is also able to catalyse the reverse reaction at the trans-Golgi membrane, namely the formation of PC and ceramide from SM and DAG. For this reason SMS1, rather than functioning strictly as SM synthase, is a bi-directional transferase capable of using PC or SM as phosphocholine donors to produce PC or SM, and the specific direction depends on the relative concentrations of DAG and ceramide as phosphocholine acceptors present in the membrane, respectively [25]. Some studies provide also evidence that SMS1 represents

a major SM synthase activity in mammalian cells, compared to SMS2, with a critical role in cell growth [38, 39].

The putative reaction mechanism of the SM synthesis catalysed by SMS1 proceeds through the following steps, as outlined in Figure 1.2 [25, 38]:

1. binding of a two-chain choline phospholipid, PC or SM, to a single binding site of the enzyme SMS1;
2. the phosphocholine head group of the donor is transferred to a conserved histidine residue in the enzyme's active site;
3. formation of DAG or ceramide according to the used phosphocholine donor, and release of the produced DAG or ceramide, while the head group stays bound to the enzyme;
4. the phosphocholine head group is transferred to the phosphocholine acceptor bound to the enzyme, ceramide or DAG, forming SM or PC;
5. release of the synthesised SM or PC from the active site of the enzyme to allow another round of catalysis.

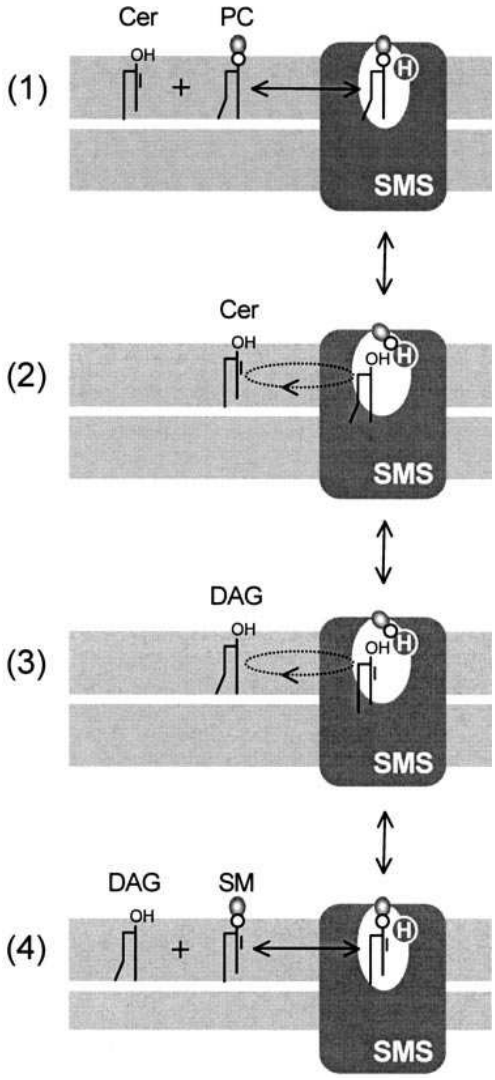

Figure 1.2: Putative reaction mechanism of SMS1-mediated SM synthesis (Figure 3 in [38]).

As already highlighted all steps in this reaction mechanism are reversible, thus satisfying the experimental observation that SM and DAG can also be converted to PC and ceramide.

1.2 Modelling cells as systems

The main goal of the quite recent field of research named "*Systems Biology*" is the systematic study of complex biological systems using the precise mathematical structure of *Systems Theory*, and at the same time cooperating with the advanced experimental knowledge and results of *Biology*.

In order to study complex relations between biochemical networks of reactions and to describe them in a mathematical way, it is important to structure the problem in a simple way. Some basic hypotheses and simplifications are needed to allow quantitative understandings and realistic predictions of cellular processes and of the underlying regulatory mechanisms. For more details about this whole Section we refer to standard texts such as [1, 2] and [32].

1.2.1 Biochemical reactions

The dynamics of intracellular processes, such as signal transduction or metabolic pathways, are often described by homogeneous systems of chemical reactions, named *chemical reaction networks* (CRN). In this simplified modelling approach, the cellular system is considered as a homogeneous system, and concentration gradients or spatial differences are ignored.

Some examples of such chemical reaction equations are:

- degradation of molecules, $\mathrm{A} \longrightarrow \emptyset$
- dimerization (reversible), $2\,\mathrm{A} \rightleftharpoons \mathrm{A}_2$
- activation (e.g. phosphorylation), $\mathrm{A} \rightleftharpoons \mathrm{A}^*$
- more complex reversible reactions, $2\,\mathrm{A} + \mathrm{B} \rightleftharpoons \mathrm{C}$

where A, B, C represent the molecular concentrations of three different reactants. In general the single arrow indicates that the reaction can go only one way, while the double arrow symbol indicates that the reaction is reversible.

1.2.2 ODE models based on chemical reaction kinetics

In order to define the dynamics of cellular reactions, translating chemical reaction systems into ordinary differential equation (ODE) models, we have to determine the velocity of each chemical reaction, named *reaction rate*. In this way we can express the conversion rate at which a particular reactant's concentration changes, $\frac{d}{dt}[A(t)]$, where $[A(t)]$ is the concentration of the molecular species A as a function of time.

The "*law of mass action*" (LMA) is a standard way to assign reaction rates to chemical reactions, allowing the construction of differential equation systems. ODE models are deterministic models, and they are appropriate to describe the behaviour of an average cell or a cell population, rather than a single cell. Formally, the law of mass action states that the rate at which a reaction takes place is proportional to the product of the concentrations of the molecular species participating in the reaction. The factor of proportionality is called *reaction rate constant*.

To make some examples we present the reaction rates relative to some simple chemical reactions (with the cursive capital letters in the rate equations indicating the concentrations of the reactants):

$$\mathrm{A} \xrightarrow{k} \qquad \text{Rate: } v = kA \tag{1.1a}$$

$$\mathrm{A} + \mathrm{B} \xrightarrow{k} \qquad \text{Rate: } v = kAB \tag{1.1b}$$

$$2\,\mathrm{A} + \mathrm{B} \xrightarrow{k} \qquad \text{Rate: } v = kA^2B \tag{1.1c}$$

where we employ the law of mass action. We can define all quantities v also with the concept of flux, in the sense of velocity at which the molecular mass of the reactant changes in a time unit. Applying these rules, and considering all fluxes contributing to the conversion rate of each specific reactant, we are able to translate complex chemical reaction systems into parametric ODE systems.

For example, for a generic reversible reaction such as $2\,\mathrm{A} + \mathrm{B} \underset{k_-}{\overset{k_+}{\rightleftharpoons}} \mathrm{C}$ the corresponding ODE model reads:

$$\dot{A}(t) = -2k_+A^2B + 2k_-C \tag{1.2a}$$

$$\dot{B}(t) = -k_+A^2B + k_-C \tag{1.2b}$$

$$\dot{C}(t) = k_+A^2B - k_-C. \tag{1.2c}$$

1.2.3 Michaelis-Menten enzyme kinetics

Now we briefly present one of the most used models for enzymatic reactions, proposed by Michaelis and Menten (1913), which describes the conversion of a substrate S into a product P, via an enzyme catalyst E. For details we refer to the texts [1, 2, 32].

In this reaction the substrate S reacts with the enzyme E, and they bind to form an intermediate complex C, that can reversibly dissociate to form again S and E. Finally the complex C decays into a product P and the original enzyme E. The considered chemical reaction is:

$$\mathrm{S} + \mathrm{E} \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} \mathrm{C} \xrightarrow{k_2} \mathrm{P} + \mathrm{E} \tag{1.3}$$

The reaction rate constants are defined as k_1, k_{-1} and k_2. Instead of formulating the complete ODE model corresponding to this chemical system, we are going to simplify the equations through some hypotheses in order to express the reactions in a single differential equation, the so-called *Michaelis-Menten* (MM) *enzyme kinetics*.

We assume that the reaction $\mathrm{C} \xrightarrow{k_2} \mathrm{P} + \mathrm{E}$ is slow compared to the time scale of the reversible reaction $\mathrm{S} + \mathrm{E} \underset{k_{-1}}{\overset{k_1}{\rightleftharpoons}} \mathrm{C}$. This assumption leads to the so-called "Quasi-steady state approximation" (QSSA). This means that the fast reversible reaction is always in equilibrium (with slowly changing substrate concentration). Considering the total amount of the enzyme E_T, that represents the sum of the free molecules E and the ones bound to the complex at steady state C_S, and solving the differential equation for the product P with respect to the substrate concentration S, we obtain the desired MM equation:

$$\dot{P} = k_2 \cdot C_S = k_2 \cdot \frac{E_T \cdot S}{K_m + S}. \tag{1.4}$$

The constant K_m is called Michaelis-Menten constant and it equals:

$$K_m = \frac{k_- + k_2}{k_+}. \tag{1.5}$$

To conclude this topic we highlight the fact that for small concentrations of the substrate S the MM kinetics provides a linear relation between the synthesis rate of the product P and the substrate level. Instead at high concentrations of the substrate, in particular for $S \gg K_m$, this relation becomes a constant, i.e. the synthesis rate is no more influenced by differences in the levels of the substrate and the enzymes are limiting.

Chapter 2

SMS1 reaction system

The enzyme sphingomyelin synthase (SMS) uses ceramide and phosphatidylcholine as substrates in the reaction that synthesizes sphingomyelin and diacylglycerol, as described in detail in Chapter 1, Section 1.1, from a biological point of view.

The concentrations of SMS1 and SMS2, the two isoforms of mammalian SMS, play a fundamental role in the control of SM, DAG and ceramide levels within human cells, and is thus strongly related with a lot of important cellular functions, as it will be described in Section 2.1.

The main focus of this Chapter is to present a mathematical dynamical model (ODEs) of the subsystem of the secretion regulatory pathway catalysed by SMS1 at the trans-Golgi network, based on chemical reaction kinetics. The aims are to investigate the role of SMS1 activity, to understand the underlying relations between educts and products of this reaction and to describe as good as possible experimental data taken from biological literature.

2.1 Impact of SMS on multiple cellular functions

Sphingomyelin (SM) is the most abundant sphingolipid species in mammalian cells, comprising 5-15% of total phospholipids [39], and it represents an important component of cellular membranes.

As already described in Chapter 1, SM sythase (SMS) is a class of enzymes that catalyse the reaction that produces SM from ceramide, by transferring a phosphocholine head group from phosphatidylcholine (PC) to ceramide, with the additional production of diacylglycerol (DAG) as byproduct, an important bioactive lipid. Therefore this enzyme plays a central role in the metabolism of sphingolipids and glycerolipids reacting together, which are involved also in different important cellular processes [38, 39]. In nature there exist two known isoforms of human SMS, named SMS1 and SMS2. They reside in different organelles, where SM synthesis in known to occur: SMS1 is located on the *cis*-medial aspect of the Golgi apparatus, and SMS2 on the plasma membrane [25].

Several biological findings show that the concentrations of these two isoforms of SMS, expressed in all major human tissues [25], are fundamental for the control of the endogenous levels of the lipids taking part in this reaction, in particular ceramide, SM and DAG, at the Golgi apparatus and at the plasma membrane, respectively [8, 39, 42]. The artificial manipulation of both enzymes can in principle influence the metabolism of all four bioactive lipids participating in the reaction, and thus we assume that SMS behaves as an important potential regulator of many cellular processes linked to these particular lipids.

Ceramide is a proapoptotic factor and has antimitogenic properties while DAG behaves as a mitogenic factor, i.e. that induces mitosis, the process in cell division in eukaryotes in which the nucleus divides to produce two new nuclei, each having the same number and type of chromosomes as the original. Moreover DAG, gathered in subcellular pools at the Golgi apparatus, binds with protein kinase D (PKD) mediating in this way its recruitment at the trans-Golgi, where, once activated, it efficiently regulates the formation of Golgi-derived secretory vesicles that are specifically destined to the cell surface [3], a process that is essential also for cell growth.

Finally SM has a high-packing density when accumulating in the plasma membrane, and a high affinity with cholesterol, contributing in this way to the barrier function of the membrane. Cooperating with cholesterol and glycosphingolipids, SM has a strong ability to form lipid "rafts" in the plasma membrane [36] [1], which are known to have an important role as a platform for signal transduction and protein sorting and trafficking in cell membranes.

Some interesting publications, which we are going to illustrate, maintain that SMS is involved, via regulation of endogenous levels of those lipids, in the regulation of multiple biological cellular functions, such as: signal transduction, functional modulation of cell membrane structure, in particular of plasma membrane lipid rafts [28], cell proliferation, differentiation, apoptosis [8], cell growth and survival [39], PKD recruitment at the TGN and activation [39, 42], which in turn is tightly related with regulation of secretion [3, 9].

Since in this study we are specifically interested only in the reaction that takes place at the trans-Golgi apparatus, where SMS1 is located, from this point forward we will refer mainly to the experiments and results relative to this SMS isoform, avoiding to mention the information concerning the other enzyme SMS2.

In the study of Tafesse et al. [39] human cervical carcinoma HeLa cells underwent RNA interference, in order to specifically deplete SMS1 and SMS2 expression. Their analysis focused to grasp the effects of these manipulations on the Golgi- and plasma membrane-associated SM synthase activities, SM production levels, overall lipid composition and cell growth. After 7 days of siRNA (small interfering RNA) treatment, the SM synthase activity in SMS1-depleted cells was reduced by 80% respect to control cells. The impact of this SMS1 depletion on the total lipid composition of HeLa cells was a 20% reduction in SM levels compared with controls and a 1.8-fold increase in ceramide levels. The decrease in SM levels seemed to the authors quite minor compared with the strongly reduced enzyme SMS activity. They state later that this is due to a growth arrest of the cell, which is accompanied by a general down-regulation of phospholipid synthesis. The effect on the

[1] The "rafts" are cholesterol- and SM-enriched membrane regions, also known as liquid-ordered domains [36], which are known to have an effect on multiple signaling pathways.

levels of PC, cholesterol and DAG was not significant. The last impact of SMS silencing underlined in [39] by the authors was the connection between SMS and growth in HeLa cells. In fact a growth arrest occurred in cells treated with SMS1 siRNAs within 3 days after transfection[2], regardless of the culture conditions. A 2-fold increase in cells undergoing apoptosis was also observed.

Ding et al. [8] proposed some experimental results that demonstrate that SMS1 and SMS2 are key factors in the control of endogenous cellular SM and DAG levels and furthermore that there exists an important relationship between SMS activity and cell apoptosis. The experimental findings about the effects of SMS1-expression modulation on the lipid composition of the cells will be described in detail in Section 2.3, since we will use these experimental data for the validation of the model and for parameter estimation. To evaluate the role of SMS in apoptosis, the authors in [8] applied SMS1 and SMS2 gene overexpression and silencing techniques to CHO cells and THP-1-derived macrophages, respectively. The overexpression led to an increase in the SMS1 activity and significantly higher intracellular SM, DAG and ceramide levels with respect to controls. CHO cells overexpressing SMS1 were more likely to undergo lysis mediated by lysenin, proving SM enrichment of the plasma membrane. Then they showed an incrementation of plasma membrane lipid rafts and of tumor necrosis factor-α-induced apoptosis, compared with wild-type CHO cells. On the other hand, SMS1 siRNA was used to knock down SMS1 activity in human macrophages. This led to significantly reduced intracellular and plasma membrane SM levels, reduced DAG and ceramide levels, and finally a decreased rate of LPS-mediated macrophage apoptosis, compared with the control case. Change in the differences in cellular PC levels was not significant in both cases of overexpression and silencing. Finally the authors suppose that both SMS1 and SMS2, regulating SM and DAG levels, could contribute to change lipid rafts on the plasma membrane and thus affect protein kinase C (PKC) activity in certain disease states [15, 26].

[2]Transfection is the process of deliberately introducing nucleic acids into cells. The term is used notably for non-viral methods in eukaryotic cells. Genetic material (such as plasmid DNA or siRNA constructs), or even proteins such as antibodies, may be transfected.

As last work we consider the one of Villani et al. [42]. In this study the authors investigated the role of the enzymes SMS1 and SMS2 on the regulation of DAG by modulating their expression. In particular they inquired into the possibility that SMSs could modulate subcellular pools of DAG, once acute activation of the enzymes is induced. Their experimental results showed that regulation of SMS affected the formation of DAG at the TGN, and SMS knockdown reduced the recruitment of the DAG-binding protein PKD at the Golgi apparatus. These findings proved that both enzymes are able to regulate the formation of DAG in HeLa cells, that this pool of DAG is biologically active and directly implicate SMS1 and SMS2 as regulators of DAG-binding proteins at the Golgi apparatus.

All of these findings demonstrate that an important relationship exists between SMS activity and cell membrane SM concentrations and thus between SMS activity and cellular functions. In particular manipulation of SMS1 cellular levels, exclusively located at the trans-Golgi apparatus, influences the secretion of proteins through the regulation of local lipid pools (DAG) and their consequent effects on protein kinase D (PKD) and ceramide transfer protein (CERT) [41].

2.2 Chemical reaction and ODE model

2.2.1 Chemical reaction

As starting point for the construction of a dynamic mathematical model that describes the conversion of ceramide (Cer) into sphingomyelin (SM) in dependence of the activity of the enzyme SMS1, we consider the following chemical reaction:

$$\mathrm{Cer} + \mathrm{PC} \underset{\mathbf{p_2},\mathrm{SMS1}}{\overset{\mathbf{p_1},\mathrm{SMS1}}{\rightleftharpoons}} \mathrm{DAG} + \mathrm{SM} \tag{2.1}$$

The SMS1 driven reaction is reversible and this fact is represented by the double arrow between the two substrates (Cer and PC) and the two products (SM and DAG), where the parameters $\mathbf{p_1}$ and $\mathbf{p_2}$ indicate the forward and backward reaction rate constants. It is however known that effectively, *in vivo*, the net reaction is always to the right, since SM is

constantly removed from the Golgi apparatus to form the vesicles that transport secretory proteins to the plasma membrane [41].

To complete the graphical representation of this chemical reaction we need to take into account some further biological knowledge. In fact this is not a closed subsystem and there are some in- and outflows that have to be further considered. Thus there is no mass conservation of the four reactants participating in the reversible reaction, and the net flux of the entire reaction does not constantly equal zero, i.e. the difference between the two unidirectional fluxes of the reactions does not vanish.

Describing the reaction using mass action kinetic, lipids' concentrations reach a steady state value, and the corresponding derivatives are zero. However the effective *in vivo* situation is a "dynamic" equilibrium, due to these fluxes that are constantly carried into and out from the system [41].

First of all there is an influx $\mathbf{C_{in}}$ of ceramide, that represents the quantity of this sphingolipid that is produced at the endoplasmic reticulum (ER) and transported, through the ceramide transfer protein (CERT), to the Golgi apparatus, where the reaction occurs. For now we consider this flux as a constant value, postponing the consideration that the influx $\mathbf{C_{in}}$ could be a function of other chemical reactants.

Then we consider for every reactant, except for PC, a simple linear degradation factor, that takes into account both the degradation and the outflux due to transportation, like in the case of SM.

Finally we assume the concentration of phosphatidylcholine (PC) to be constant, according to some experimental results taken from the literature, which show that the concentration of PC does not significantly change if the value of SMS1 is manipulated [8, 39, 42]. This assumption can be interpreted as a dominant regulation of PC by other chemical pathways, which balance the effect of SMS1, or by a large overall pool of PC, which does not notice changes of small fractions [41].

Taking all these effects into account, the chemical reaction of interest can be represented like in Figure 2.1.

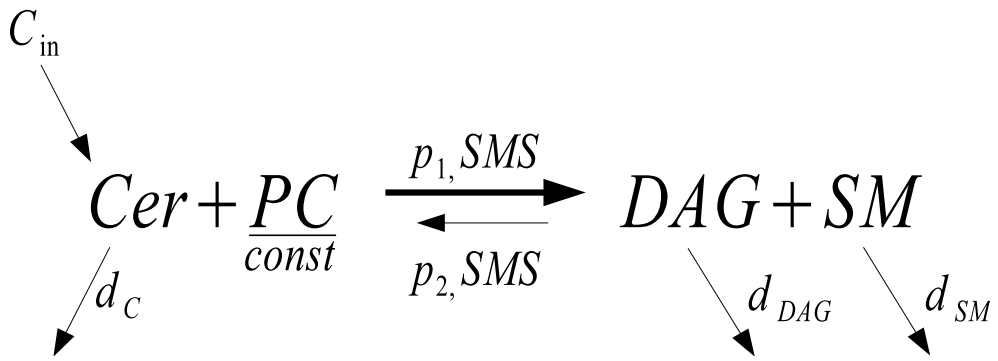

Figure 2.1: SMS1 driven conversion of ceramide to sphingomyelin.

2.2.2 Parameterized differential equation model

Referring to the chemical reaction just described, we now want to develop a dynamical model for this process of the form $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \theta)$, using mass action kinetics to translate chemical reaction systems into ordinary differential equations systems, as described in the introductory Chapter 1, Section 1.2. In this way we obtain a *deterministic* mathematical model for the biochemical system in the form of differential equations that represent the first derivatives w.r.t. time of the concentrations of the reactants taking part in the reaction.

To describe the two enzymatic forward and backward reactions with two substrates we use Michaelis-Menten enzyme kinetics, presented in Chapter 1, Section 1.2. Our theoretical analysis about the dynamical ODE system are, however, independent of the exact choice of the functions modelling these fluxes, and they could be represented with two generic fluxes $v_1(SMS1, Cer, PC)$ for the forward reaction, and $v_2(SMS1, DAG, SM)$ for the backward one, expressions that we will consider in the last Section 2.4 of this Chapter. To simplify the notation we define the concentration of ceramide with the initials C, and moreover we write simply SMS to express the total concentration of the enzyme $SMS1$.

This model will be of course a function of a certain number of parameters, namely: the influx of ceramide (C_{in}), the reaction rate constants of the degradation effects (d_C, d_{DAG} and d_{SM}) and the two reaction rate and two Michaelis-Menten constants of the forward and backward enzymatic reactions (p_1, p_2, k_1 and k_2).

The developed parametric differential equation system, for the lipids and proteins reacting together, is presented as follows:

$$\dot{C} = C_{in} - d_C\, C - p_1\, SMS\, \frac{C \cdot PC}{C \cdot PC + k_1} + p_2\, SMS\, \frac{DAG \cdot SM}{DAG \cdot SM + k_2} \tag{2.2a}$$

$$\dot{DAG} = -d_{DAG}\, DAG + p_1\, SMS\, \frac{C \cdot PC}{C \cdot PC + k_1} - p_2\, SMS\, \frac{DAG \cdot SM}{DAG \cdot SM + k_2} \tag{2.2b}$$

$$\dot{SM} = -d_{SM}\, SM + p_1\, SMS\, \frac{C \cdot PC}{C \cdot PC + k_1} - p_2\, SMS\, \frac{DAG \cdot SM}{DAG \cdot SM + k_2} \tag{2.2c}$$

$$\dot{PC} = 0 \tag{2.2d}$$

$$\dot{SMS} = 0. \tag{2.2e}$$

We present now a simplified and more compact version of this ODE model, that will be considered for the future analysis and considerations about the system. This notation is also used in MATLAB for the computational simulations (see Appendix A).

First of all we set $SMS1 = u$, because it is the input (or control) of our system, which is defined *a priori* to simulate different experiments. It expresses the "activity" of the enzyme SMS1 compared to a reference situation, in the sense that its value is 1 in the control case, and it can vary taking larger or smaller values, in dependence to which experiment is considered. More precisely $u > 1$ in the case of SMS1 overexpression, and $u < 1$ in the case of silencing (knockdown) of the enzyme, in the sense that consequently to these manipulations the cellular SMS1 activity is subject to a u-fold increase or decrease.

We consider only the first three differential equations, considering u and PC among the other parameters, given that their value remains constant by hypothesis. We redefine also the state variables as $\mathbf{x} = (x_1, x_2, x_3) = (C, DAG, SM)$, and the three corresponding degradation rates as d_1, d_2 and d_3. In this way the system can be rewritten as follows:

$$\dot{x}_1 = f_1(\mathbf{x}) = C_{in} - d_1 \cdot x_1 - p_1\, u\, \frac{x_1 \cdot PC}{x_1 \cdot PC + k_1} + p_2\, u\, \frac{x_2 \cdot x_3}{x_2 \cdot x_3 + k_2} \tag{2.3a}$$

$$\dot{x}_2 = f_2(\mathbf{x}) = -d_2 \cdot x_2 + p_1\, u\, \frac{x_1 \cdot PC}{x_1 \cdot PC + k_1} - p_2\, u\, \frac{x_2 \cdot x_3}{x_2 \cdot x_3 + k_2} \tag{2.3b}$$

$$\dot{x}_3 = f_3(\mathbf{x}) = -d_3 \cdot x_3 + p_1\, u\, \frac{x_1 \cdot PC}{x_1 \cdot PC + k_1} - p_2\, u\, \frac{x_2 \cdot x_3}{x_2 \cdot x_3 + k_2}. \tag{2.3c}$$

Our model is thus of the form $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \theta)$, with the state: $\mathbf{x} = (x_1, x_2, x_3) \in \mathbb{R}^3_+$, belonging to the positive orthant of $\mathbb{R}^3$, referring to the fact that concentrations are positive quantities, and the parameter being a 10-dimensional vector:

$$\theta = (C_{in}, p_1, p_2, d_1, d_2, d_3, k_1, k_2, u, PC) \in \mathbb{R}^{10}_+ \tag{2.4}$$

also made up of all positive quantities. The first 8 parameters $(C_{in}, p_1, p_2, d_1, d_2, d_3, k_1, k_2)$ will have to be estimated from experimental data taken from biological literature, that will be described in the next Section 2.3, while u and PC have constant values defined by the experimental conditions.

To complete this Section we list the units of measurement of the variables and parameters of the system. Since we will use non-dimensional values for all lipid concentrations and for the input u by normalizing to some control measurement values, we assume the variables x_1, x_2 and x_3 and the two constants u and PC to be dimensionless. Consequently k_1 and k_2 are dimensionless too, while $C_{in}, p_1, p_2, d_1, d_2$ and d_3 are $[\text{min}^{-1}]$ [3].

2.3 Experimental data from literature

The aim of the deterministic mathematical model just presented is to predict, for specific values of the 8 parameters, $\theta = \theta_0$, real data of the concentrations of the four reactants participating in the reaction, and in this way to describe in a formal way the dynamics of the reaction. The first step is thus to gather experimental datasets of the concentrations of the reactants at particular instants of time, measured through biochemical experiments that reproduce the enzymatic reaction in specific cells expressing human SMS1. The second step is to estimate a specific value for the parameter vector θ, such that the model can explain as good as possible the given data obtained under the specific experimental conditions, and finally predict other datasets under different experimental conditions.

[3] We could generalize defining the unit of measurement of all these parameters as $[\text{time}^{-1}]$. In fact we are interested in the steady state values of the concentrations so we do not need a specific unit for the time, and we could let it not specified.

For this purpose we use some experimental datasets provided in [8]. In this article the authors explain the effects of manipulation of both SMS isoforms, using gene overexpression and knockdown techniques, on the levels within the cell of the following lipids: ceramide (Cer), phosphatidylcholine (PC), diacylglycerol (DAG) and sphingomyelin (SM).

We will consider only the experiments concerning the manipulation of SMS1 activity, and its influence on the metabolism of all four lipids of the reaction. That's because, as already underlined in Section 2.1, we consider only the reaction that takes place at the trans-Golgi apparatus, where this specific SMS isoform (SMS1) is located, while SMS2 is located at the plasma membrane which is beyond the scope of this thesis.

2.3.1 Overexpression

Overexpression was carried out in genetically modified CHO (Chinese hamster ovary) cells, that stably express human SMS1 and SMS2 [8]. This procedure increased SMS1 and SMS2 mRNA levels, and this fact resulted in an incrementation of cellular SMS activity, compared with control CHO cells. In particular SMS1 overexpression resulted in a 2.2-fold increase in cellular SMS activity, compared with wild type cells (Fig. 1B in [8]). Enzymatic assays were used to measure cellular lipid levels, whose absolute values are reported in Table 2.1 (Table 1 in [8]).

Table 2.1: Lipid concentrations in CHO cells overexpressing SMS1.

Exp.#	Type of cell	u	Cer	PC	DAG	SM
1.	Control	1	0.96 ± 0.09	294 ± 23	2.45 ± 0.51	25 ± 3
2.	Overexpressing SMS1	2.2	1.38 ± 0.08^a	283 ± 37	3.31 ± 0.66^a	31 ± 3^a

The units of measurement of all quantities are *nmol/mg protein*, and the values of the concentrations of four lipids are given as mean ± standard deviation (SD), and derived from five repeated experiments. The meaning of the superscript ^{a} in the second line of Table 2.1 is that differences between these data and the values of the control experiment

in the same column are statistically significant ($P < 0.01$ by ANOVA and $P < 0.05$ by Student-Newman-Keuls test).

As results in Table 2.1 show, cells in which SMS1 was overexpressed contained significantly higher SM and DAG levels, as it was expected. Not expected instead was the significant increase of ceramide levels, compared with controls. Finally PC levels showed no significant changes between the control experiment and the one with overexpression. This aspect was already considered in our mathematical ODE model, by defining the concentration of phosphatidylcholine as constant.

From data it was not possible for the authors in [8] to understand which particular SM was increased by overexpression, representing SM levels reported in Table 2.1, the total amount of SM present in all cellular membranes, including ER, Golgi apparatus and also plasma membrane. With some further considerations the authors explain that SMS overexpression increases the lipid rafts, or raft-like domains, in the plasma membrane. This means that SMS overexpression has an important effect on SM levels specifically in the plasma membrane, where signal transduction begins, because it changes the overall cell membrane structure [8].

2.3.2 Silencing

To further evaluate the relationship between SMS activity and cellular lipid levels, the authors of [8] used SMS1 and SMS2 small interfering RNA (siRNA) to knock down the respective mRNA in THP-1-derived macrophages, that are human macrophages. They explain that one reason to use a different type of cells w.r.t. the overexpression experiment is that hamster SMS1 and SMS2 cDNA sequences are not known, and so the siRNA approach to knock down these two enzymes in CHO cells would not yet be plausible.

This procedure caused a significant decrease in cellular SMS activity, in particular SMS1 siRNA reduced it by 23% compared with control cells (Fig. 4A in [8]). With enzymatic assays they measured cellular levels for the same four lipids of interest, whose absolute concentrations are shown in Table 2.2, as in the case of overexpression (Table 2 in [8]).

Table 2.2: Lipid concentrations in SMS1 knockdown macrophages.

Exp.#	Type of cell	u	Cer	PC	DAG	SM
3.	Control	1	0.77 ± 0.06	320 ± 33	2.26 ± 0.12	44 ± 5
4.	SMS1 siRNA	0.77	0.73 ± 0.09	339 ± 19	1.71 ± 0.23^a	35 ± 4^a

From these data it results that total intracellular SM and DAG levels were significantly decreased in cells that had been transfected with SMS1 siRNA, compared with wild type cells. Instead no significant change was measured in ceramide cellular levels, as for PC levels. Like in Table 2.1 concerning the experiment of overexpression, the units of measurement of all quantities are *nmol/mg protein*, and the values of the concentrations of lipids are given as mean ± SD, and are the average of five experiments. Also the meaning of the superscript ^{a} is the same, indicating statistically significant data compared with control in the same column ($P < 0.01$ by ANOVA and $P < 0.05$ by Student-Newman-Keuls test).

2.3.3 Normalization of the two datasets

All the data relating to the four experiments (two controls, one overexpression and one silencing) will be the starting point for the parameter estimation of the model.

In order to use these experimental data for this purpose, we interpret the measurements as dynamic equilibrium states (steady states) $\bar{\mathbf{x}}(u) = (\bar{x}_1(u), \bar{x}_2(u), \bar{x}_3(u))$, functions of the input u. In this situation the concentrations and hence the reaction fluxes are constant, i.e. $\dot{x}_i = f_i(\bar{\mathbf{x}}) = 0, \forall\, i = 1, 2, 3$. Nevertheless we have to consider that those datasets are obtained from two different types of cells, Chinese hamster ovary cells and human macrophages, respectively. In the two datasets concerning the two control experiments (the first and the third) in the two different kinds of cells, one can notice at once that the absolute concentrations of all four reactants are quite different. Therefore the two datasets given in Tables 2.1 and 2.2 are not consistent, being measured in two different cellular systems. For this reason we cannot compare directly the absolute concentration values in

the parameter estimation procedure. To make our analysis possible and to be able to test hypotheses across cell lines, we normalize the concentrations of every reactant, i.e. mean and standard deviation, w.r.t. the corresponding value of the control experiment, namely the mean value measured in the wild type cell. We do that separately for the two different cells, so that all concentrations (the mean values) in the first and third experiment will be equal to 1, with normalized SD, and the ones in the second and forth experiment will be normalized w.r.t. the control absolute levels. In Table 2.3 we present all these values, which we will use for parameter estimation and further analyses. For the relative PC levels, we consider only the normalized mean values and not the standard deviations, because in the model we define PC as a constant parameter, as we do also for u. In fact for each experiment we define the specific value of PC, and we do not consider it as a measurement, like the other three lipids with statistical description, to be used for parameter estimation. Every lipid normalized concentration reported in Table 2.3 is thus dimensionless, being normalized w.r.t. control levels, and it justifies the consideration made in Subsection 2.2.2 about the units of measurement of the state variables x_i.

Table 2.3: Lipid concentrations for all 4 experiments normalised w.r.t. control levels in wild type cells, relative to the same cell, expressed as mean value ± SD.

Exp.#	Type of cell	u	Cer	PC	DAG	SM
1.	Control	1	1 ± 0.09	1	1 ± 0.21	1 ± 0.12
2.	Overexpressing SMS1	2.2	1.44 ± 0.08^a	0.96	1.35 ± 0.27^a	1.24 ± 0.12^a
3.	Control	1	1 ± 0.08	1	1 ± 0.05	1 ± 0.11
4.	SMS1 siRNA	0.77	0.95 ± 0.12	1.06	0.76 ± 0.10^a	0.8 ± 0.09^a

In this way from now on we consider relative changes of lipid levels in response to manipulation of SMS1 activity (overexpression and knockdown), and we analyse the behaviour of the model in a qualitative way. In fact this normalization has the effect that both quantitative datasets relative to the two different cell types are transformed into one rather semi quantitative dataset to test our hypothesis.

2.4 Analysis of the ODE system

In this Section we want to make some simple considerations about the model before dealing with the estimation of parameters and the problem of identification of parameters, that will be discussed in the next Chapters.

We will prove that the hypothesis of a simple constant ceramide influx C_{in} is not adequate to explain the qualitative trend of steady state levels in response to different SMS activities.

For this reason we will present a second revised model, in which the constant ceramide influx is replaced by an influx that is a function of another reactant (i.e. diacylglycerol), taking the form of a feedback regulation term $C_{in}(DAG) = f(DAG)$ between a product and an educt. We will explain why a feedback of this form is the next logical extension of the first model, supporting this choice with effective biological findings about the regulation through diacylglycerol of the transport of ceramide at the TGN. We want to show theoretically that this second modified model can better reproduce biochemical data in a qualitative way.

2.4.1 Ceramide influx: feedback regulation?

Changes in the activity of the enzyme SMS1 produce the alteration of the fluxes of the reversible reaction (2.1), and consequently of the steady state concentrations of the four reactants participating in the reaction. From experimental data, given in Tables 2.1 and 2.2, we can notice that the increase and decrease of SMS1 activity (u), produce a significant increase and decrease of SM and DAG cellular levels, respectively. This fact let us presume that SMS1 overexpression increases the net flux of the reversible reaction to the right side. Considering this statement as true, and under the hypothesis of constant C_{in}, we conclude that ceramide level should decrease after SMS1 is overexpressed, and should increase after the enzyme is knocked down, while we already know that PC level remains almost constant following to u-level alteration [41]. In fact, if we consider the biochemical reaction, represented in Figure 2.1, in isolation, the changes of steady state concentrations

of lipids SM and DAG after SMS manipulation should have opposite sign with respect to the change of steady state ceramide level. This statement is also proven in Section 2.4 in a formal mathematical way.

But if we consider the real trend of ceramide levels at steady state after overexpression and silencing, looking at Table 2.3, we notice a completely different behaviour. In fact in the case of overexpression the quantity $\bar{x}_1(u)$ (steady state concentration of ceramide) significantly increases, while in the case of SMS1 knockdown, it slightly decreases, although not marked as significant in [8].

In Table 2.4 we summarize the real trend of lipid concentrations at the equilibrium after SMS manipulation (**Data**), together with the expected trend given by the model (2.3) (**Expected**), highlighting with double arrows the contradiction in the ceramide's trends.

Table 2.4: Trend of lipid levels compared to controls after overexpression and silencing: comparison between trend of experimental data and what is expected from the ODE model with C_{in} constant.

	u	$\bar{x}_1(u)$ (Cer)	$\bar{x}_2(u)$ (DAG)	$\bar{x}_3(u)$ (SM)	PC
Overexpression: **Data**	SMS1↑	⇑	↑	↑	–
Overexpression: **Expected**	SMS1↑	⇓	↑	↑	–
Silencing: **Data**	SMS1↓	– (⇓)	↓	↓	–
Silencing: **Expected**	SMS1↓	⇑	↓	↓	–

In this way we can already underline that the first model $\dot{\mathbf{x}} = \mathbf{f_1}(\mathbf{x}, \theta)$, defined by the equations (2.3), fails to explain experimental findings qualitatively and we need to extend our hypothesis, modifying the structure of the differential equation system.

From biological knowledge we know that there exists an indirect feedback regulation from DAG to the transport of ceramide to the TGN, via protein kinase D and the ceramide transfer protein CERT (see the graphical scheme in [41]). Therefore we want to describe this phenomenon in our dynamic mathematical model, trying to change the equations

in order to explain qualitatively better the experimental results of the changes in lipid composition in response to SMS perturbations, given in Table 2.4. In particular we assume that the velocity of transport of ceramide to the TGN, where the reaction takes place, represented by the influx C_{in}, is a function of the DAG concentration, becoming in this way a term of the form $C_{in}(DAG)$. Moreover we assume that this regulation takes place without this last reactant being consumed, and hence we do not have to change or add some extra terms in the other differential equations of system (2.3).

In this sense we describe the indirect regulation from DAG to ceramide levels at the TGN in a very simplified way. In fact the term $C_{in}(DAG)$ summarizes all biochemical reactions that occur in the pathway between DAG and Cer with a unique effective direct regulation term [41].

One has to be aware of the strong simplification adopted in this context, but we will show that it is sufficient to explain qualitatively the experimental results of the changes in lipid levels at the TGN consequently to SMS1 manipulation.

2.4.2 Choice of the feedback function $C_{in}(DAG) = f(DAG)$

The question that arises now is which function $f(DAG)$ should be chosen to represent this ceramide influx at the TGN, in order to explain experimental data in a proper way.

Unfortunately this indirect feedback effect that binds DAG levels at the TGN to the transport of ceramide via CERT is not yet well understood in detail. Moreover this process could be differently regulated depending on the specific cellular system. What is known is that DAG at the TGN can influence the transport of ceramide from the endoplasmic reticulum to the Golgi apparatus both in a positive and in a negative way, trough different chemical pathways, and for this reason it is not so clear how to represent this effect by a mathematical function of the concentration of diacylglycerol, x_2 in our model (2.3).

We try to approximate the relation between DAG levels at steady state and the influx C_{in}, by comparing the outputs of the model (2.3) with the available lipid measurements for the different experimental conditions, as it is shown in Table 2.4. From these data we

can understand that the feedback regulation function should be *monotonically increasing*.

First of all it can be observed that the concentration of ceramide at steady state $\bar{x}_1(u)$ is monotonically increasing with C_{in}, ensured by the effect of mass balance considering the system in isolation in a situation of dynamic equilibrium. Afterwards we can make the following considerations: as depicted in the first panel of Figure 2.2, we fix a point in the plane (DAG, C_{in}) that refers to the control experiment, $(\bar{x}_2^{control} = 1, C_{in}^{control})$. In the case of SMS1 overexpression the DAG level increases significantly. This means that the net flux increases to the right, with consequently more production of SM from ceramide. As we already explained, without the feedback regulation there should be a consequent decrease in the steady state level of ceramide, but the experimental results show an opposite behaviour. For this reason we assume that an increased DAG concentration should also increase the influx $C_{in}^{overexpr}$ w.r.t. the control experiment (second panel of Figure 2.2).

In the opposite case of SMS1 silencing, the DAG level decreases significantly compared with the control level, due to a smaller net flux of the reversible reaction to the right direction. In the hypothesis of constant influx C_{in}, ceramide levels at equilibrium should increase, since it is less consumed by the reaction. But also in this case experimental data give controversial results, since when SMS1 is knocked down the ceramide level at steady state slightly decreases. This fact means that there should be a decreased influx of ceramide $C_{in}^{silencing}$ at the TGN with respect to the control case (third panel of Figure 2.2).

All these considerations bring us to assume that the function $C_{in}(DAG)$ is monotonically increasing, and, in a first attempt, the simplest approximation of such a function that can be chosen is a linear one, as shown also in the last panel of Figure 2.2, of the form:

$$C_{in}(DAG) = f(DAG) = a \cdot DAG \quad a > 0. \tag{2.5}$$

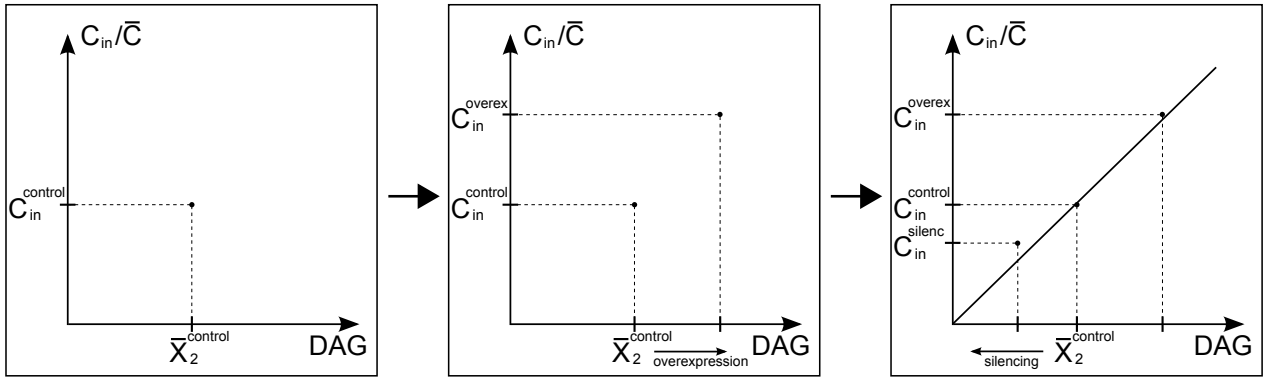

Figure 2.2: Steps for the choice of a linear feedback function $C_{in}(DAG)$.

The second modified model $\dot{\mathbf{x}} = \mathbf{f_2}(\mathbf{x}, \theta)$, with this new feedback function, reads as follows (using the same notation of model (2.3)):

$$\dot{x_1} = f_1(\mathbf{x}) = a \cdot x_2 - d_1 \cdot x_1 - p_1\, u\, \frac{x_1 \cdot PC}{x_1 \cdot PC + k_1} + p_2\, u\, \frac{x_2 \cdot x_3}{x_2 \cdot x_3 + k_2} \tag{2.6a}$$

$$\dot{x_2} = f_2(\mathbf{x}) = -d_2 \cdot x_2 + p_1\, u\, \frac{x_1 \cdot PC}{x_1 \cdot PC + k_1} - p_2\, u\, \frac{x_2 \cdot x_3}{x_2 \cdot x_3 + k_2} \tag{2.6b}$$

$$\dot{x_3} = f_3(\mathbf{x}) = -d_3 \cdot x_3 + p_1\, u\, \frac{x_1 \cdot PC}{x_1 \cdot PC + k_1} - p_2\, u\, \frac{x_2 \cdot x_3}{x_2 \cdot x_3 + k_2}. \tag{2.6c}$$

We could have added in equation (2.5) also a constant term, writing in this way $C_{in}(DAG) = a \cdot DAG + b$, avoiding to have for the new system a fixed point in the origin, $\bar{\mathbf{x}} = (0, 0, 0)$, and moreover having one more degree of freedom. But using the parametrization of equation (2.5), we have in the model (2.6) the same number of parameters of the first model (2.3), so that we can compare directly the goodness of the data fits for the two models, without using other criteria of model comparison.

A short explanation is also needed for reasons about why we chose a linear approximation of the feedback function. For example we could have chosen also functions of higher orders, having no clear idea of how this feedback effect acts biologically on the regulation of the transport of ceramide. With MATLAB we carried out a maximum likelihood parameter estimation, method explained in Chapter 3, in order to estimate the three different influxes for the three different experimental conditions: $C_{in}^{control}, C_{in}^{overexpr}$ and $C_{in}^{silencing}$. After plotting the three obtained C_{in} as function of the three respective

normalized DAG concentrations (the three normalized DAG measurements reported in Table 2.3), which are represented in Figure 2.3 with three bold points (**Data**), we performed a linear and quadratic interpolation of the three points, passing through the origin, to compare which function could better represent the three available points. These two different interpolations are also depicted in Figure 2.3, with a continuous straight line for the linear interpolation and a dotted line for the quadratic one. As one can observe in the figure the linear approximation is sufficient to represent the feedback function $C_{in}(DAG)$.

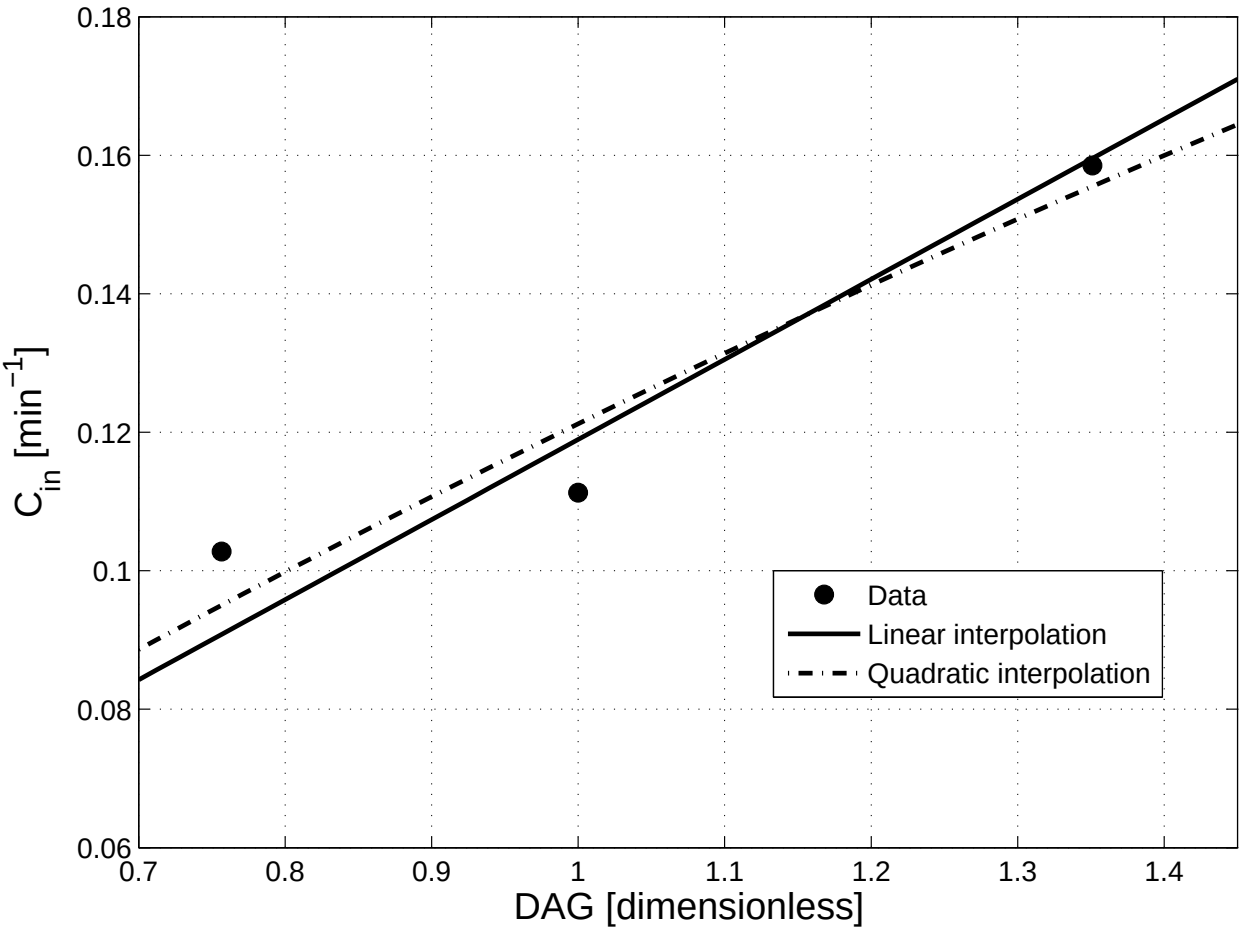

Figure 2.3: Linear (continuous line) and quadratic (dotted line) approximations of the feedback function $C_{in}(DAG)$.

2.4.3 Steady state analysis with different SMS concentrations

In this Subsection we want to prove mathematically that the original model (2.3), with constant ceramide influx, can never be able to explain the experimental results given in Table 2.3, especially the trends of steady states of lipid levels in response to a variation of the value of the input u, which represents the activity of the enzyme SMS1. Instead the second modified model (2.6) in principle can explain the experimental findings for particular values of some parameters.

To prove these facts it is sufficient to consider the differential equations of the two ODE models at the equilibrium, and make some elementary considerations. Moreover these theoretical results are independent of the choice of the particular form for the fluxes of the two forward and backward enzymatic reactions, and can be proven employing, instead of the Michaelis-Menten terms considered in the two models (2.3) and (2.6), two generic fluxes $v_1(u, x_1, PC)$ and $v_2(u, x_2, x_3)$, in both ODE systems.

These analytical results will be confirmed also by the simulated results that follow the estimation of parameters with the methods of maximum likelihood and of sampling, described in the next Chapters 3 and 4.

Theorem 2.4.1:

Given the ODE model (2.3)*, of the form* $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \theta)$*, where all* $f_i(\mathbf{x}, \theta)$ *are continuous in* $\mathbf{x} \in \mathbb{R}^3_+$, $\forall\, i = 1, 2, 3$*, and assuming that the hypotheses of the* Implicit function theorem *are satisfied, there exist the following relations between the partial derivatives of the three states* x_1, x_2 *and* x_3 *at equilibrium w.r.t. the input* u*:*

$$\frac{\partial \bar{x}_1(u)}{\partial u} = -\frac{d_2}{d_1} \cdot \frac{\partial \bar{x}_2(u)}{\partial u} \tag{2.7a}$$

$$\frac{\partial \bar{x}_3(u)}{\partial u} = \frac{d_2}{d_3} \cdot \frac{\partial \bar{x}_2(u)}{\partial u}. \tag{2.7b}$$

This means that the variation of $\bar{x}_1(u)$ *when* u *is varied, has always opposite sign compared to the corresponding variations of the other two steady states* $\bar{x}_2$ *and* $\bar{x}_3$*.*

Theorem 2.4.2:

Given the ODE model (2.6)*, of the form* $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \theta)$*, where all* $f_i(\mathbf{x}, \theta)$ *are continuous in* $\mathbf{x} \in \mathbb{R}^3_+$, $\forall\, i = 1, 2, 3$*, and assuming that the hypotheses of the* Implicit function theorem *are satisfied, there exist the following relations between the partial derivatives of the three states* x_1, x_2 *and* x_3 *at equilibrium w.r.t. the input* u*:*

$$\frac{\partial \bar{x}_1(u)}{\partial u} = \frac{1}{d_1} \cdot (a - d_2) \cdot \frac{\partial \bar{x}_2(u)}{\partial u} \tag{2.8a}$$

$$\frac{\partial \bar{x}_3(u)}{\partial u} = \frac{d_2}{d_3} \cdot \frac{\partial \bar{x}_2(u)}{\partial u}. \tag{2.8b}$$

This means that, if $a > d_2$*, the variation of* $\bar{x}_1(u)$ *when* u *is varied has the same sign as the corresponding variations of the other two steady states* $\bar{x}_2$ *and* $\bar{x}_3$*, and so the three partial derivatives* $\partial \bar{x}_i / \partial u$ *all have the same sign.*

Now we prove both theorems together, since the procedure is the same.

Proof. To apply the *Implicit function theorem* (IFT) we consider that for both dynamical models:

$$f : \mathbb{R}^3_+ \times \mathbb{R}^8_+ \to \mathbb{R}^3_+ \in C^1 : (\mathbf{x} \in \mathbb{R}^3_+, \theta \in \mathbb{R}^8_+) \mapsto \mathbf{x} \in \mathbb{R}^3_+. \tag{2.9}$$

If the hypotheses of the IFT are satisfied, it follows that there exist neighbourhoods $U(\theta_0) \subseteq \mathbb{R}^8_+$ and $V(\bar{\mathbf{x}}(\theta_0) \coloneqq \bar{\mathbf{x}}_0) \subseteq \mathbb{R}^3_+$ and a unique and smooth function $\bar{\mathbf{x}} : U \to V : \theta \mapsto \bar{\mathbf{x}}(\theta)$, that represents the steady state of the system as a continuous function of parameters. For this reason we can consider the partial derivative of every variable at steady state $\bar{x}_i(\theta)$ w.r.t. the parameter u, the input of the model.

If we consider the first model (2.3) at steady state we obtain:

$$\begin{aligned} 0 &= C_{in} - d_1 \cdot \bar{x}_1 - v_1(u, x_1, PC) + v_2(u, x_2, x_3) \\ 0 &= -d_2 \cdot \bar{x}_2 + v_1(u, x_1, PC) - v_2(u, x_2, x_3) \\ 0 &= -d_3 \cdot \bar{x}_3 + v_1(u, x_1, PC) - v_2(u, x_2, x_3). \end{aligned}$$

We notice that the term $v_1(u, x_1, PC) - v_2(u, x_2, x_3)$ is the same for all three equations, and thus we can obtain the following equalities:

$$C_{in} - d_1 \cdot \bar{x}_1 = d_2 \cdot \bar{x}_2 = d_3 \cdot \bar{x}_3. \tag{2.11}$$

Considering the first equality we obtain:

$$\bar{x}_1 = \frac{1}{d_1} \cdot (C_{in} - d_2 \cdot \bar{x}_2),$$

and thus if we differentiate with respect to u, we easily obtain the relation of equation (2.7a). If we consider the second equality instead we obtain:

$$\bar{x}_3 = \frac{d_2}{d_3} \bar{x}_2,$$

from which the second relation (2.7b) derives.

For the second modified model (2.6) one can do the same considerations just presented above for the first model. The model at steady state reads:

$$\begin{aligned} 0 &= a \cdot \bar{x}_2 - d_1 \cdot \bar{x}_1 - v_1(u, x_1, PC) + v_2(u, x_2, x_3) \\ 0 &= -d_2 \cdot \bar{x}_2 + v_1(u, x_1, PC) - v_2(u, x_2, x_3) \\ 0 &= -d_3 \cdot \bar{x}_3 + v_1(u, x_1, PC) - v_2(u, x_2, x_3) \end{aligned}$$

and in this case the following relations hold:

$$a \cdot \bar{x}_2 - d_1 \cdot \bar{x}_1 = d_2 \cdot \bar{x}_2 = d_3 \cdot \bar{x}_3. \tag{2.13}$$

As before we obtain the following relations between the steady states:

$$\begin{aligned} \bar{x}_1 &= \frac{1}{d_1} \cdot (a - d_2) \cdot \bar{x}_2; \\ \bar{x}_3 &= \frac{d_2}{d_3} \bar{x}_2; \end{aligned}$$

from which we derive equations (2.8a) and (2.8b). With all parameters and variables being positive quantities, the considerations about the signs of this partial derivatives follow easily.

□

Chapter 3

MLE-based statistical inference approach for parameter estimation

To describe mathematically the dynamics and the equilibrium state of the chemical reaction of interest we presented in the previous Chapter 2 two different deterministic dynamical ODE models based on chemical reaction kinetics: the first one, given by the differential equations (2.3), considering a constant ceramide influx C_{in}, and the second one, model (2.6), taking into account a positive feedback regulation between the level of DAG and the influx of ceramide at the TGN.

As already underlined both systems have the same number of parameters, represented compactly by the vector θ, whose values have to be estimated with particular techniques, in order to explain the considered experimental dataset and to respect some criteria of optimality and goodness of fit.

The main content of this Chapter is the description of the results obtained for both ODE models using a particular statistical method for the estimation of parameters, the Maximum Likelihood Estimation (MLE). This statistical approach provides a specific value for the parameter vector $\hat{\theta}_{MLE}$ as result of an optimization problem. A comparison of the results of the simulations obtained for the two ODE models with the estimated parameter value confirms the analytical result of Section 2.4 that asserts that the feedback term is necessary to explain qualitatively the experimental findings of Section 2.3.

3.1 Maximum likelihood parameter estimation

As first topic of this Chapter we want to present a brief overview of the concept of maximum likelihood estimation of parameters, introducing both the formal statistical definition and the practical optimization problem that has to be implemented, describing some related questions and problems that arise in the search of the optimal solution.

3.1.1 Statistical definition

We want to present here only the basic concepts and definitions concerning the statistical method of maximum likelihood for parameter estimation. For further and more specific details about the theory we refer to standard statistical texts, such as [35].

Suppose that we have a random vector $\mathbf{y} \in \mathbb{R}^q$ distributed with unknown probability density function that belongs to the parameterized family $\{p_{\mathbf{y}}(y|\theta), \theta \in \Theta\}$. We consider the observation $y_i \in \mathbb{R}^n$, $i = 1, ..., N$, where n is the number of measured outputs, and i representing the index of the N observed experiments. In this way we set $q = n \cdot N$, and the vector $\mathbf{y}$ is simply the sequence of all observations y_i.

The likelihood function of the set of observations $y_0 = \{y_i,\, i = 1, ..., N\} \in \mathbb{R}^q$ is the function $L_{y_0} : \Theta \to \mathbb{R}_+$ defined by:

$$L_{y_0}(\theta) = p_{\mathbf{y}}(y_0|\theta). \tag{3.1}$$

The "maximum likelihood principle", introduced by Gauss in 1856 and subsequently popularized by R.A. Fisher, suggests to take as estimate of θ, referring to the observed data y_0, the vector $\hat{\theta} \in \Theta$ that maximizes $L_{y_0}(\theta)$:

$$L_{y_0}(\hat{\theta}_{MLE}) = \max_{\theta \in \Theta} L_{y_0}(\theta); \tag{3.2}$$

that means:

$$\hat{\theta}_{MLE} = \arg\max_{\theta \in \Theta} L_{y_0}(\theta); \tag{3.3}$$

assuming implicitly that the maximum exists. In this way the value of the vector $\hat{\theta}_{MLE}$ is the one that maximizes the probability to see "a posteriori" the observation y_0.

3.1.2 Prior distribution over parameters

As concerns the practical resolution of this optimization problem the main question that arises is where the solution $\hat{\theta}_{MLE}$ has to be searched in the parameter space. In a general framework we expect that the desired result should be a global one, but most of the times finding the global maximum is a very difficult and complex problem. This occurs especially if the dimension p of the given parameter vector θ is large and in some cases if the likelihood function is a very irregular function, with many local maxima and minima or with stiffness properties.

Moreover in a biological framework the considered parameters represent reaction rate constants, specific fluxes of concentrations of reactants or other particular parameters, like for example the Michaelis-Menten parameter. Therefore estimated parameter values should be compatible with their biological meaning, e.g. half-lives, synthesis rates, diffusion rates, and a partial knowledge of the biochemical context under study can be useful to set some constraints for parameters, e.g. at least positivity.

For these reasons to implement the optimization problem of interest the solution can not be easily searched in the entire space $\mathbb{R}^p$ and we need constraints for our problem. In this sense we need to impose bounds for each parameter that has to be estimated, and it would be reasonable to set these bounds in a region where we expect that the solution should lie.

This can be interpreted as an *a priori* information about the distribution of parameters, and in a statistical framework this information can be expressed by a probability density function (pdf) $p(\theta)$, that represents the a priori knowledge about θ before having seen the data y, and for this reason it is defined **prior distribution** over parameters.

From a practical point of view imposing bounds on parameters is a useful strategy for ensuring convergence of MLE optimization algorithms by avoiding, during intermediate optimization steps, inadmissible parameter values, e.g. negative values under positivity constraints, that may either hinder recovery to the admissible parameter region or even cause failure of numerical algorithms such as integration procedures. In a probabilistic

context, imposing *hard* bounds on parameters by means of lower and upper limits, e.g. $\theta_{min} \leq \theta \leq \theta_{max}$, can be interpreted as assuming a uniform prior distribution on parameters, i.e. $\theta \sim \mathcal{U}(\theta_{min}, \theta_{max})$. If the parameter bounds are wide enough that the maximum of the likelihood function is attained inside the admissible parameter region then the bounds are not influential and the MLE estimate coincides with the maximum a posteriori (MAP) estimate, i.e. the parameter value that maximizes the posterior distribution of the parameters given the data, under the assumption of a uniform prior. This links the MLE and the Bayesian inference considered in the next Chapter.

As last consideration, we underline the fact that building quantitative dynamic models for intracellular processes is only possible for specific parts of a cell, for which we have to assume that they function autonomously and can be described in isolation. Anyway external manipulations made on the specific subsystem do not act only locally but have certainly multiple effects on other parts spread all over the cell. These effects could involve unmodeled components that are not considered in the simplified model, and there could be unexpected results that cannot be explained by the model under study [41].

It is clear how choosing model constraints and bounds for parameters has a very important meaning and at the same time it consists in a very difficult task in the construction of predictive models.

3.1.3 Constrained nonlinear optimization problem

As described above, the computation of the maximum likelihood estimate for the parameter θ consists, from a practical point of view, in a local optimization problem, being the solution searched in a subspace of the parameter space $\mathbb{R}^p$.

The choice of the bounds for each parameter is the first effective problem to be considered by the computational algorithm that solves the MLE problem, defining the knowledge about the prior distribution.

Another practical problem of the algorithm is the search of the optimum itself, i.e. the value of the parameter that maximizes the likelihood function in the considered space.

In particular, if a maximum exists, it could be not unique, or there could be many local maxima inside the constrained subspace and the algorithm could converge to a wrong solution.

Finally an important problem that has to be considered in the field of the estimation of parameters is the one of the *structural identifiability* of parameters. In a very simple way we can define the probability parameterized family $p_{\mathbf{y}}(y|\theta)$, or the parameter θ itself, to be *locally identifiable* in θ_0, if there exists an opened region Θ_0 around θ_0 where $p_{\mathbf{y}}(y|\theta_1) = p_{\mathbf{y}}(y|\theta_2)$ implies $\theta_1 = \theta_2$, $\forall\, \theta_1, \theta_2 \in \Theta_0$ [35]. It means that a parameter is *not identifiable* if the probability to see the data given the parameter, i.e. the likelihood function, takes the same value for two different parameter vectors. Written in a mathematical way if:

$$L_y(\theta_1) = L_y(\theta_2), \quad \theta_1 \neq \theta_2. \tag{3.4}$$

3.2 Statistical model

The computation of the likelihood function $L_y(\theta)$, where y is the given dataset, and the consequent possibility to estimate a specific set of parameters that maximizes it, require obviously to define the likelihood function itself, i.e. to introduce the *statistical model* that will be considered for the simulations and the estimation problem.

We report now the first considered *deterministic model*, represented by the ordinary differential equation system (2.3) for the lipid concentrations, introducing also the outputs z_i that we used in our simulations with MATLAB. We define them as the natural logarithm of the three state variables (see equations (3.6)), and the reason of this choice has to do with the form of the statistical model that will be explained just afterwards. The equations of the model and of the outputs are:

$$\begin{aligned}
\dot{x}_1 &= C_{in} - d_1 \cdot x_1 - p_1\, u\, \frac{x_1 \cdot PC}{x_1 \cdot PC + k_1} + p_2\, u\, \frac{x_2 \cdot x_3}{x_2 \cdot x_3 + k_2} \\
\dot{x}_2 &= -d_2 \cdot x_2 + p_1\, u\, \frac{x_1 \cdot PC}{x_1 \cdot PC + k_1} - p_2\, u\, \frac{x_2 \cdot x_3}{x_2 \cdot x_3 + k_2} \\
\dot{x}_3 &= -d_3 \cdot x_3 + p_1\, u\, \frac{x_1 \cdot PC}{x_1 \cdot PC + k_1} - p_2\, u\, \frac{x_2 \cdot x_3}{x_2 \cdot x_3 + k_2}
\end{aligned}$$

$$z_1 = \log x_1 \tag{3.6a}$$

$$z_2 = \log x_2 \tag{3.6b}$$

$$z_3 = \log x_3. \tag{3.6c}$$

The same output equations apply also to the second ODE model (2.6), which differs only by parameter C_{in} being replaced with the term $a \cdot x_2$.

To introduce the chosen statistical model, we assume that the observed data, that are the three steady state lipid concentrations for the three different experimental conditions, defined as $\bar{y}_{i,u}$, are given by the steady state concentrations predicted by the model as function of the parameter θ multiplied by the errors $\eta_{i,u}$. The index i represents the three different state variables x_1, x_2 and x_3 of the ODE systems, i.e. the normalized concentrations of ceramide, diacylglycerol and sphingomyelin, while u represents the different values of the input for the three different experimental conditions, control, overexpression and silencing, respectively. The equations for the measurements are:

$$\bar{y}_{i,u} = \bar{x}_{i,u}(\theta) \cdot \eta_{i,u} \quad i = 1, 2, 3, \ u = 1, 2.2, 0.77. \tag{3.7}$$

The choice of this particular relation between data and measurement errors arises from the fact that biological concentration measurements are strictly nonnegative, such that an additive measurement noise model may not adequately describe experimental data. Moreover measurement errors increase often with the measure itself, as with the constant coefficient of variation model in which (additive) measurement noise has a standard deviation proportional to the measure, e.g. $y = x + x \cdot \epsilon = x \cdot (1 + \epsilon)$ with ϵ zero mean random noise with standard deviation equal to the coefficient of variation. A convenient probabilistic representation of measurements that combines both nonnegativity and increase of noise with measurement levels, is given by the log-normal distribution, described in the next Subsection.

As already presented in the equations (3.6) after the ODE model above, we define the outputs of the model to be the natural logarithm of the state variables, and this relation

is valid in particular for the equilibrium situation:

$$\mathbf{z}(\theta) = \log(\mathbf{x}(\theta)) \Longleftrightarrow \bar{\mathbf{z}}(\theta) = \log(\bar{\mathbf{x}}(\theta)). \tag{3.8}$$

3.2.1 Log-normal distribution error model

The properties of the stochastic model are determined in our case by the definition of the statistical error model, i.e. the distribution of the measurement errors $\eta_{i,u}$, for each variable and experimental condition. For our simulations we chose a *log-normally distributed* measurement noise.

In probability theory, a log-normal distribution is a continuous probability distribution of a random variable whose natural logarithm is normally distributed. It means that if Y is a log-normally distributed random variable, then X = log(Y) has a normal distribution. The log-normal distribution is a distribution of a random variable that assumes only positive real values, [12, pp. 578] and [29].

In our study it means that the natural logarithm of the errors $\eta_{i,u}$ is normally distributed, and we can summarize this fact with the following equation:

$$\eta_{i,u} \sim \log \mathcal{N}(0, \sigma_{i,u}^2) \Longleftrightarrow \log(\eta_{i,u}) \sim \mathcal{N}(0, \sigma_{i,u}^2). \tag{3.9}$$

The normal distribution at the right side is characterized by mean zero and variance $\sigma_{i,u}^2$. The values of the standard deviations $\sigma_{i,u}$ that we considered in our simulations are the ones listed in Table 2.3 together with the observed data $\bar{y}_{i,u}$, that had been normalized w.r.t. the control values.

If we consider equation (3.7), that expresses the observed data as function of the simulated state variables and of the measurement noise, and we take the natural logarithm of both sides we obtain the relation:

$$\tilde{y}_{i,u} = \log(\bar{y}_{i,u}) = \log(\bar{x}_{i,u}(\theta)) + \log(\eta_{i,u}) = \bar{z}_{i,u}(\theta) + \log(\eta_{i,u}). \tag{3.10}$$

In this way we obtain a linear relation between the logarithm of the data $\tilde{y}_{i,u}$, the outputs of our model at steady state $\bar{z}_{i,u}(\theta)$ and the logarithm of the errors. This relation allows

us to find a specific expression of our stochastic model, given by the probability density function $p(y|\theta)$, i.e. the likelihood function, that describes the stochastic data generation process given the parameters of the model θ.

In fact from the equations (3.9) and (3.10) we derive that also the natural logarithm of data is normally distributed, with mean $\bar{z}_{i,u}(\theta)$ and same variance of the errors $\sigma^2_{i,u}$:

$$\tilde{y}_{i,u} \sim \mathcal{N}(\bar{z}_{i,u}(\theta), \sigma^2_{i,u}). \tag{3.11}$$

Assuming all measurement errors to be independent and log-normally distributed, as defined in equation (3.9), we obtain the following expression for the likelihood function:

$$L_{\tilde{y}}(\theta) = p(\tilde{y}|\theta) = \prod_{u,i} p(\tilde{y}_{i,u}|\theta) = \prod_{u,i} \frac{1}{\sigma_{i,u}\sqrt{2\pi}} \exp\left[-\frac{1}{2}\left(\frac{\tilde{y}_{i,u} - \bar{z}_{i,u}(\theta)}{\sigma_{i,u}}\right)^2\right]. \tag{3.12}$$

We underline again the fact that the considered distribution is relative to the *logarithm* of the dataset, otherwise we should have used in (3.12) the expression of the pdf of a log-normal distribution, but in general the normal distribution has a lot of more interesting properties and we prefer to consider the given expression (3.12).

3.3 Simulations and results on simulated data

In this Section we report some details about the simulations concerning the maximum likelihood parameter estimation (MLE), and the results relative to both ODE systems (2.3) and (2.6) without and with the feedback regulation term. This analysis regarding the MLE was carried out mainly to investigate a first model fit for both ODE systems, to check the hypothesis of the feedback control effect and to determine the prior support region of the parameters to be used also for the sampling analysis that will be described in the next Chapter 4.

All simulations were carried out with the numerical computing environment MATLAB, with the additional utilization of the toolboxes SBPD, SBTOOLBOX2 , which offer specifically a powerful environment in which to build models of biological systems, and finally MCMCSTAT. This last toolbox was used exclusively for the simulations described in the

next Chapter 4 to carry out the sampling from the posterior distribution with a Markov Chain Monte Carlo method.

For more information all specific details about the program and the code written in MATLAB are reported in the Appendix A.

3.3.1 Estimation of parameters in logarithmic scale

For all simulations concerning the estimation of parameters, we applied a logarithmic transformation to base 10 to the parameter vector, so that the objective function $L_y(\theta)$, that was originally function of the vector θ, becomes function of the new vector $\psi = \log_{10}\theta$. This means formally that θ in the dynamic models is replaced by $10^{\log_{10}\theta} = 10^{\psi}$. Practically in the two ODE systems we defined the previous parameter vectors, θ_1 for the first model and θ_2 for the second one, to equal:

$$\theta_1 = 10^{\psi_1} = (C_{in}, p_1, p_2, d_1, d_2, d_3, k_1, k_2) \tag{3.13a}$$

$$\theta_2 = 10^{\psi_2} = (a, p_1, p_2, d_1, d_2, d_3, k_1, k_2) \tag{3.13b}$$

so that the new parameters to estimate are:

$$\psi_1 = \log_{10}\theta_1 \tag{3.14a}$$

$$\psi_2 = \log_{10}\theta_2. \tag{3.14b}$$

If we define as $\hat{\theta}$ the estimated solution of the optimization problem (without active constraints):

$$\hat{\theta} = \arg\max_{\theta} L_y(\theta) \tag{3.15}$$

then the following optimality condition holds:

$$\nabla_\theta L_y(\theta)|_{\theta=\hat{\theta}} = 0. \tag{3.16}$$

If we suppose that $\hat{\theta} > 0$, then we can state that the optimal solution does not change with the logarithmic transformation, obtaining:

$$\hat{\psi} = \widehat{\log_{10}\theta} = \log_{10}\hat{\theta} = \arg\max_{\log_{10}\theta} L_y(10^{\log_{10}\theta}). \tag{3.17}$$

In fact with the logarithmic transformation the condition of positivity of (3.15) is fulfilled and, moreover, the following optimality condition for the new estimated parameter vector (3.17) holds:

$$\nabla_{\log_{10}\theta} L_y(10^{\log_{10}\theta})\big|_{\log_{10}\theta=\widehat{\log_{10}\theta}} = \log 10 \cdot \mathrm{diag}(\theta)\nabla_\theta L_y(\theta)|_{\theta=\hat{\theta}} = 0 \tag{3.18}$$

where

$$\mathrm{diag}(\theta)\nabla_\theta L_y(\theta) = [\theta_1 \frac{\partial L_y(\theta)}{\partial \theta_1}, \theta_2 \frac{\partial L_y(\theta)}{\partial \theta_2}, \ldots]^T \tag{3.19}$$

being $\theta_i = 10^{\log_{10}\theta_i}$. We can notice in particular that, with the considered transformation, the two optimality conditions (3.16) and (3.18) are equivalent, holding $\hat{\theta} > 0$.

In this way the elements of the gradient (3.18) are given by:

$$\frac{\partial L_y(10^{\log_{10}\theta})}{\partial \log_{10}\theta_i} = \log 10 \cdot \frac{\partial L_y(\theta)}{\partial \theta_i}\theta_i = \log 10 \cdot \frac{\partial L_y(\theta)}{\partial \theta_i/\theta_i} \tag{3.20}$$

and in principle the logarithmic transformation of parameters causes the only fact that the derivative of the objective function $L_y(\theta)$ with respect to the components of the vector θ becomes scaled by the value of the corresponding component and in some way it is replaced by the derivative of $L_y(\theta)$ w.r.t. relative changes of parameters.

Such non-linear transformation of parameters has a positive effect on the properties of convergence of the optimization algorithm towards the optimum (maximum or minimum) solution, because it reduces the problems with parameter scaling, especially if the final parameter values are very distant from the initial conditions of the algorithm. At the same time the transformation into logarithmic scale implicitly guarantees the constraint that all parameters remain positive.

The details of how these transformations were implemented in the model in the MATLAB code are reported in Appendix A.

In the next subsection we will explain the choice of the prior distribution of parameters $p(\psi) = p(\log_{10}\theta)$, and we will assume a log-uniform prior distribution for the model parameter θ, i.e. the log-transformed random variable ψ is assumed to be uniformly distributed.

3.3.2 Choice of bounds for parameters

To solve with MATLAB the constrained optimization problem of determining a maximum likelihood estimate $\hat{\theta}_{MLE}$ for the parameter vector θ, we employed the MATLAB function `fmincon` which attempts to find a constrained *minimum* of a scalar multivariable function starting at an initial estimate.

As optimization algorithm was chosen `OPTIONSfmincon.Algorithm='interior-point'`, the tolerance on the constraint violation and the termination tolerance on the function value were set to `OPTIONSfmincon.TolCon = 1e-6` and `OPTIONSfmincon.TolFun = 1e-6`. The objective function given as input to the optimization algorithm was minus the logarithm of the likelihood function, i.e. $-\log L_{\tilde{y}}(\theta)$, having to find the *maximum* of the likelihood, and since the logarithm does not change the optimal solution being a monotonically increasing function.

The main issue of this analysis was the choice of the constraints for the 8 parameters that had to be estimated. In fact we had no information at all about some possible values of the constant rates coming from biological knowledge and we had even no indicative awareness about the order of magnitude of such parameters. In the statistical framework this means that there wasn't any knowledge about the prior distribution of parameters.

In order to find the optimal solution, we made multiple attempts to detect where (in the parameter space) the objective function assumed larger values. Some useful information during these trials was given by the fact that, for some of the 8 parameters, the final values were at one of the edges imposed by the constraints on the parameters. Thus in the following attempts the specific bounds were expanded or moved to the right or left direction in accordance with where the previous respective values had been estimated.

We decided to choose a width of the intervals of all parameters of 4 units, that in a logarithmic scale is equivalent to four orders or magnitude. Since we had no information about those values a priori, as already explained, we consider it a reasonable choice to start with. After multiple attempts to find a good estimate of the parameter $\hat{\psi}_{MLE}$, we set the intervals to be approximately symmetric around the final values. These bounds will be

kept fixed also for the simulations concerning the sampling from the posterior distribution, described in Chapter 4. This procedure is a sort of *empirical Bayes* method, in which the prior distribution is estimated from the data. Of course the boundaries relative to two ODE systems resulted very different for almost the totality of all 8 parameters. All the details and specific results will be explained in the two next Subsections, for the model without and with feedback term, respectively.

3.3.3 Model without feedback

As first point of our simulations, we carried out the maximum likelihood estimation for the first model, given by the differential equation system (2.3), with outputs (3.6). We underline again the fact that in the implementation with MATLAB we effectively defined the parameters in logarithmic scale, as defined in equation (3.14). After many attempts to define the bounds for each parameter $\theta_i \in \mathbb{R}^8_+$, we determined boundaries for each $\psi_i \in \mathbb{R}^8$ that cover intervals of four units around the MLE parameters $\hat{\psi}_{MLE}$, that, considering the real model parameter $\theta = 10^{\psi}$, are equivalent to intervals of four orders of magnitude. We calculated a specific set of values defining the maximum likelihood estimate $\hat{\psi}_{MLE} = \log_{10} \hat{\theta}_{MLE}$ by maximization of the likelihood function $L_{\tilde{y}}(\psi) = L_{\tilde{y}}(\log_{10} \theta)$.

In Table 3.1 are reported the MLE values of the parameters and the respective bounds that define the support region of the log-uniform prior distribution. The log likelihood value calculated for this specific maximum likelihood estimate, which is a measure of the overall data fit quality, is $\log L_{\tilde{y}}(\hat{\theta}_{MLE}) = 7.3$.

Afterwards we simulated the first ODE model (2.3) with the specific parameter values reported in Table 3.1 (to be more precise $\hat{\theta}_{MLE} = 10^{\hat{\psi}_{MLE}}$), considering especially the equilibrium situation. The main interest is to compare the experimental dataset with the steady state lipid concentrations predicted by the particular differential equations model $\dot{\mathbf{x}} = \mathbf{f_1}(\mathbf{x}, \hat{\theta}_{MLE})$.

Table 3.1: MLE parameters for the first ODE model (2.3) without feedback and respective prior support regions.

Parameter $\hat{\psi}$	$\hat{\psi}_{MLE}$	Prior support region
$\log_{10}\hat{C}_{in}$	3.1015	[1,5]
$\log_{10}\hat{p}_1$	-1.9265	[-4,0]
$\log_{10}\hat{p}_2$	1.3463	[-1,3]
$\log_{10}\hat{d}_1$	3.0591	[1,5]
$\log_{10}\hat{d}_2$	-2.0617	[-4,0]
$\log_{10}\hat{d}_3$	-2.0513	[-4,0]
$\log_{10}\hat{k}_1$	-4.0847	[-6,-2]
$\log_{10}\hat{k}_2$	3.7859	[2,6]

In Figure 3.1 we can see the simulated trajectories of the lipid concentrations for ceramide, DAG and SM. Starting from random initial conditions, the model $\dot{\mathbf{x}} = \mathbf{f_1}(\mathbf{x}, \hat{\theta}_{MLE})$ was simulated for the three different experimental conditions, i.e. for the three different values of the input $u = 1, 2.2, 0.7$, that are represented in the first, second and third panels of Figure 3.1, respectively. We can at first notice how the simulated trajectories, and consequently the steady states, of ceramide remain always constant for all three experimental conditions, as though the SMS1 activity manipulations, obtained through overexpression and silencing, had no effect on the concentration of this lipid. As already mentioned, and even mathematically proved at the end of Section 2.4, we affirm that the considered ODE model is not able to fit the qualitative changes of the steady state levels of ceramide in response to the alteration of the activity of the enzyme, and for this reason the predicted steady state concentration is simply the mean between all measurements, because the model prediction in response to SMS1 manipulation would have an opposite behaviour respect to the effective trend of data. Instead the simulated steady states of DAG and SM appear to be more sensitive to the changes of the amount of SMS1, significantly increasing and decreasing w.r.t. the control level in the cases of overexpression and silencing, respectively.

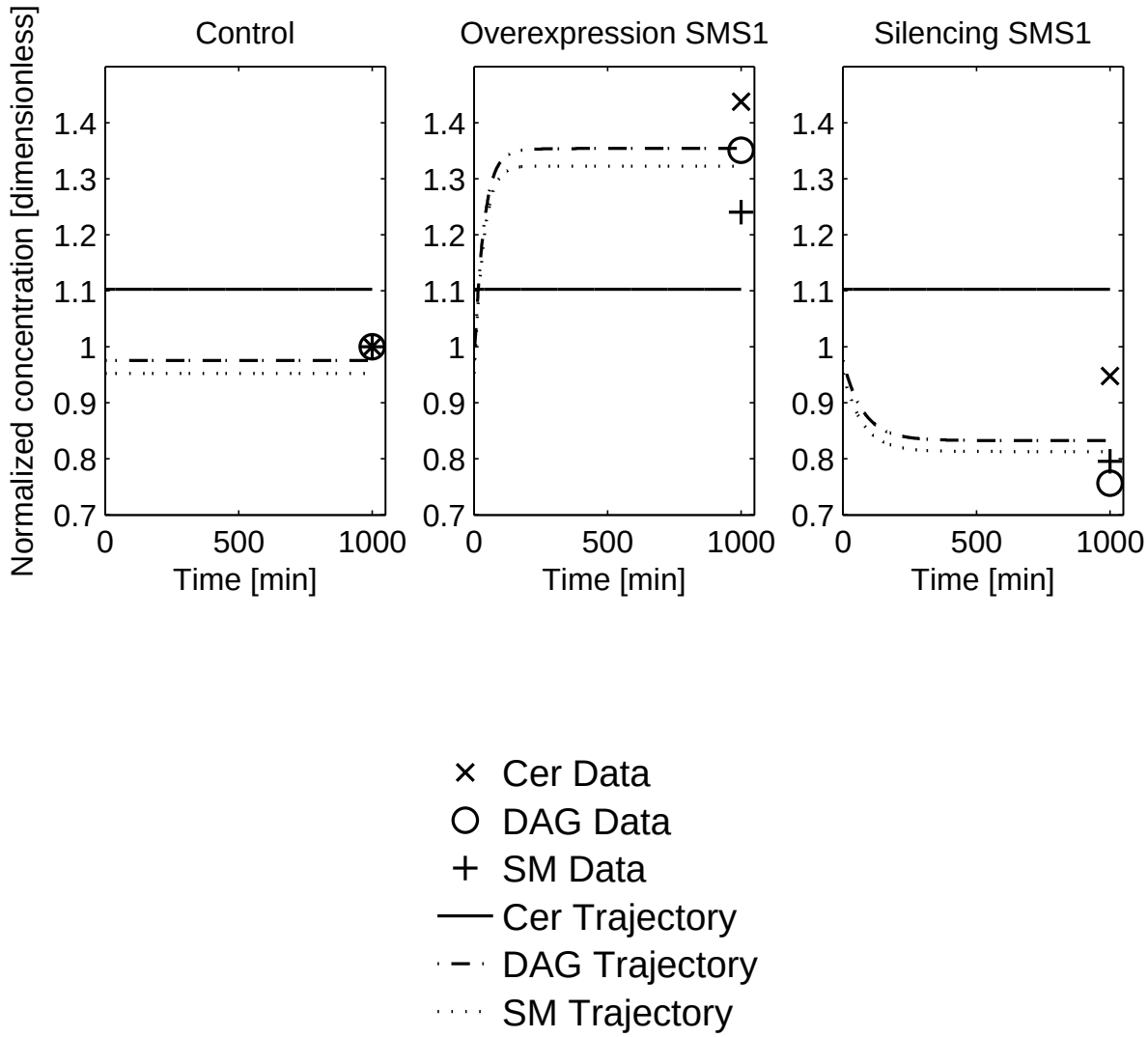

Figure 3.1: Trajectories of lipid concentrations obtained with the ODE model with constant ceramide influx C_{in} simulated with the MLE parameters of Table 3.1, plotted together with the experimental data at steady state for the three different experimental conditions.

These results are reached by a high turnover rate for ceramide in comparison to its conversion in the SMS1 driven enzymatic reaction, i.e. ceramide has a high production ($\hat{C}_{in} \simeq 1.26 \cdot 10^3$) and high degradation rate ($\hat{d}_1 \simeq 1.15 \cdot 10^3$) compared to the forward rate constant for the reversible reaction ($\hat{p}_1 \simeq 1.18 \cdot 10^{-2}$). SMS1 manipulations thus effect ceramide steady state levels only marginally. On the other side the degradation rates of

DAG and SM are almost the same ($\hat{d}_2 \simeq 0.87 \cdot 10^{-2}, \hat{d}_3 \simeq 0.89 \cdot 10^{-2}$) and much slower that the one of ceramide. We can also notice that these last two parameters are of the same order of magnitude as the forward rate constant of the reversible reaction, and for this reason their steady state levels can be highly regulated by the activity of SMS1. The two Michaelis-Menten constants differ in several orders of magnitude ($\hat{k}_1 \simeq 8.23 \cdot 10^{-5}$ and $\hat{k}_2 \simeq 6.11 \cdot 10^{3}$) but we can maintain that the estimates of these parameters are not reliable, since the model does not depend linearly on them, and thus we do not pay too much attention on the significance of these values [41].

We can notice that the concentrations of DAG at steady state, predicted by the model for this particular MLE parameter vector, are always slightly higher than the SM level for all experimental conditions. The experimental data instead show a different behaviour, with DAG slightly above SM in the overexpression experiment, and on the contrary slightly beneath SM in the silencing experiment. Since the same amounts of DAG and SM are produced or consumed by the enzymatic reversible reaction, and moreover there is no significant difference in the estimated degradation rate constants d_2 and d_3, the model cannot explain at the same time both effects just presented and the trend of the increase and decrease for a single reactant is the same for all experiments. Anyway the experimental data of DAG and SM are still well captured by the model.

3.3.4 Model with feedback

We report now the same kind of results described in the previous Subsection obtained this time for the second ODE model (2.6) that takes into account the positive feedback regulation from the concentration of DAG on the influx of ceramide at the TGN.

In the implemented differential equation for the concentration of ceramide, with all parameters expressed in logarithmic scale, the constant term $10^{\log_{10} C_{in}}$ is replaced by the term $10^{\log_{10} a} \cdot x_2$.

Table 3.2 reports the MLE values for all 8 parameters, and the respective bounds, that define the support regions for the log uniform prior distributions. The 8 real estimated

model parameters are given by $\hat{\theta}_{MLE} = 10^{\hat{\psi}_{MLE}}$. The log likelihood value calculated for this specific maximum likelihood estimate is in this case $\log L_{\tilde{y}}(\hat{\theta}_{MLE}) = 14.3$, which is twice the one obtained for the first model (2.3). This result is already an interesting information about the fact that the second modified model can better fit the given dataset.

Table 3.2: MLE parameters for the second ODE model (2.6) with feedback and respective prior support regions.

Parameter $\hat{\psi}$	$\hat{\psi}_{MLE}$	Prior support region
$\log_{10} \hat{a}$	2.9391	[1,5]
$\log_{10} \hat{p}_1$	1.5896	[0,4]
$\log_{10} \hat{p}_2$	3.4195	[1,5]
$\log_{10} \hat{d}_1$	2.9116	[1,5]
$\log_{10} \hat{d}_2$	1.0208	[-1,3]
$\log_{10} \hat{d}_3$	1.0302	[-1,3]
$\log_{10} \hat{k}_1$	0.0974	[-3,1]
$\log_{10} \hat{k}_2$	2.5266	[0,4]

One can immediately notice that the maximum likelihood estimates are in this case very different from those obtained in the first model, reported in Table 3.1, and thus also all prior support intervals. While the synthesis and degradation rates of ceramide are of the same order of magnitude than before ($\hat{a} \simeq 0.87 \cdot 10^3$ and $\hat{d}_1 \simeq 0.82 \cdot 10^3$), the SMS1 driven forward and backward reaction rate constants p_1 and p_2 are several orders of magnitude larger in this second model ($\hat{p}_1 \simeq 0.39 \cdot 10^2$ and $\hat{p}_2 \simeq 2.63 \cdot 10^3$). As a consequence the ceramide level at equilibrium is much more influenced by manipulations of the input u than with the first model. Also the degradation rate constants of DAG and SM are larger than before but always very similar ($\hat{d}_2 \simeq 0.1 \cdot 10^2$ and $\hat{d}_3 \simeq 0.11 \cdot 10^2$) and finally the Michaelis-Menten constants are different in the modified model ($\hat{k}_1 \simeq 1.25$ and $\hat{k}_2 \simeq 3.36 \cdot 10^2$) [41].

To see effectively the change of the quality of the data fit for this second revised model, we simulated it with the estimated set of parameter values of Table 3.2. The resulting trajectories for the three lipid concentrations in the three different experimental conditions are shown in Figure 3.2.

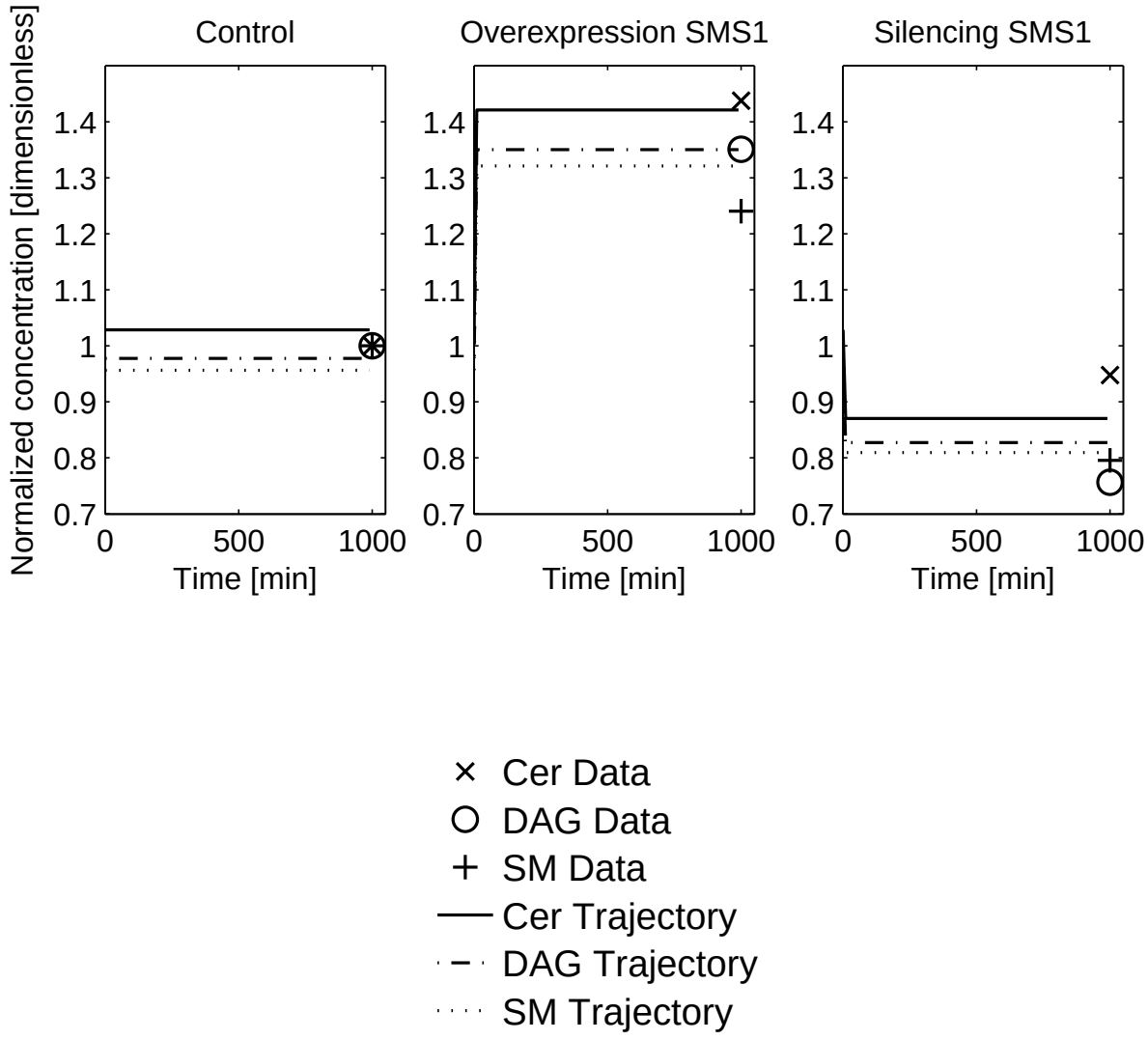

Figure 3.2: Trajectories of lipid concentrations obtained with the second ODE model with the feedback regulation simulated with the MLE parameters of Table 3.2, plotted together with the experimental data at steady state for the three different experimental conditions.

One can immediately observe that in this case the steady state ceramide levels predicted by this second model can qualitatively follow the real trend of the experimental data in response to manipulations of the input u. As before the steady state level of DAG is always above the level of SM for all three experiments, observing only the experimental finding relative to the overexpression experiment, and not the one of silencing. The explanation is the same of that described for the first ODE model.

3.3.5 Comparison of the results of the two models

The results obtained with the simulations of the two ODE models (2.3) and (2.6) for the particular values of the parameter vectors estimated with the method of maximum likelihood estimation $\hat{\theta}_{MLE} = 10^{\hat{\psi}_{MLE}}$ confirm the theoretical investigations of the steady state analysis, described in Section 2.4, and moreover showed a good fit quality.

Since the qualitative behaviour of ceramide steady state levels cannot be captured by the first model (2.3) in all three experiments, the most likely parameters are those that leave ceramide levels constant across all three experiments, as can be seen in Figure 3.1.

Instead, with the introduction of the hypothesis of the feedback regulation from DAG to ceramide, the second model (2.6) is able to qualitatively capture the changes of ceramide endogenous levels in response to SMS1 manipulation.

Besides this fundamental outcome, that corroborates our biological hypothesis and also our formal mathematical results, these statistical simulations were interesting to detect the prior support regions of the parameter distribution, that will be used for the Bayesian estimation approach of the sampling from the posterior distribution described in the next Chapter. Considering intervals of four orders of magnitude we maintain that in this way we can evaluate a wide range of possible values for parameters and this is also interesting for the reason that we do not have any a priori information about the values of such parameters.

Chapter 4

Sampling-based Bayesian approach for parameter estimation

In addition to the statistical inference approach of MLE presented in the previous Chapter, we used also another method for parameter estimation, that consists in a sampling-based statistical Bayesian approach.

This technique for parameter estimation has the big advantage that it provides not only best point estimates (e.g. MAP: maximum a posteriori probability estimation) but also the entire a posteriori distribution of parameters, allowing the assessment of confidence intervals for parameters and of model predictions.

In this Chapter we will present a general overview of the idea of Bayesian learning, introducing the concept of sampling from the posterior distribution, and a particular method to realize it: the Metropolis-Hastings algorithm from the class of the Markov chain Monte Carlo sampling methods. We applied this statistical technique to the two considered ODE parametric models (2.3) and (2.6) to carry out the parameter estimation, and also in this case we found that the model with feedback can fit the experimental data in the right way, while this does not hold for the first ODE model with constant ceramide influx.

Besides the results concerning the data fit, we will present also interesting information about the predictions of the two models and about the marginal posterior distributions, quantities that supply a first understanding of the distribution and the identification of the single parameters.

4.1 Introduction: Bayesian learning

In this introductory Section we want to present the basic general concepts of the statistical method of Bayesian learning. For more details about the theory we refer to standard books such as [12, 43].

In general *statistical inference* is concerned with drawing conclusions about unobserved quantities starting from the knowledge of experimental numerical data.

In particular *Bayesian inference* is the process of fitting a stochastic model to a dataset and summarizing the result by a probability distribution on the model parameters θ and on unobserved quantities, such as predictions for new observations z [12, Chap. 1].

These probability models that characterize Bayesian conclusions about the quantities of interest, θ and z, are conditional on the observed data y. This feature of conditioning on observed data distinguishes Bayesian inference from common approaches to statistical inference that aim to estimate θ (or z) over the distribution of possible y values conditional on the true unknown value of θ, as in the case of the maximum likelihood estimation (MLE).

In a Bayesian framework we need stochastic models for quantities that we observe and for quantities about we wish to learn. Data y and parameter θ are interpreted as random variables, with respective probability density functions often referred to as *data distribution* (or *sampling distribution*) $p(y|\theta)$ and *prior distribution* $p(\theta)$. These two densities describe the stochastic data generation process given model parameter θ and the a priori knowledge about θ itself before having observed the data. For given θ, $p(y|\theta)$ is a probability distribution over all possible observations y. Instead for given data y, and as a function of the unknown parameter θ, it expresses how probable it is to observe the data for each value of the model parameter, and in this context it is called the *likelihood function* $p(y|\theta) = L_y(\theta)$, as described in Section 3.1.

In this way we can obtain a model providing the *joint probability distribution* for θ and y, whose density function is given by the product of the two previous densities:

$$p(\theta, y) = p(\theta) \cdot p(y|\theta) = p(\theta) \cdot L_y(\theta). \tag{4.1}$$

4.1.1 Posterior distribution

The main function of interest in a Bayesian framework is the *posterior distribution*, which is the distribution over the parameter θ conditioned on the observed value of the data y. By applying the basic property of conditional probability known as **Bayes' rule**, we obtain the following expression for the posterior density function:

$$p(\theta|y) = \frac{p(\theta) \cdot L_y(\theta)}{p(y)}, \tag{4.2}$$

where $p(y)$ is the *marginal likelihood* given by:

$$p(y) = \int p(y|\theta)\, p(\theta) d\theta. \tag{4.3}$$

Often one can express equation (4.2) by omitting the factor $p(y)$, which does not depend on θ and, given fixed y, can thus be considered as a constant, albeit unknown. In this way we obtained an unnormalized posterior distribution, that reads:

$$p(\theta|y) \propto p(\theta) \cdot L_y(\theta). \tag{4.4}$$

This last expression of the posterior distribution will be, in particular, the starting point for the resolving algorithm for parameter estimation, described in the following Section 4.2.

We can notice that using Bayes' rule and given a specific stochastic model, the posterior distribution (4.4) is influenced by data y only through the likelihood function $L_y(\theta)$. In this regard the chosen statistical model, which comprises the deterministic structure of the system and the stochastic nature of parameters and measurement errors, plays a fundamental role in the analysis of experimental data and determines results and conclusions [12, Chap. 1].

4.1.2 Model prediction

In a Bayesian learning approach also unknown observable quantities are often important objects of interest, like in the case of predictions for new outputs of the model that have not yet been observed.

Following a similar logic like the one used to obtain the posterior distribution, we can introduce the density function of the unknown but observable data y as:

$$p(y) = \int p(y, \theta) d\theta = \int p(\theta)\, p(y|\theta) d\theta = \int p(\theta)\, L_y(\theta) d\theta. \tag{4.5}$$

Previously we had defined this quantity as marginal distribution of y, but a more informative name is *prior predictive distribution*, indicating that it is a distribution over an unknown observable quantity (prediction) that is not conditional on previous observations.

Introducing the knowledge of the observed data y, we can predict a new unknown observable quantity z, generated from the same process, through the definition of the *posterior predictive distribution* $p(z|y)$, posterior because it is conditional on the observed data y, and predictive because it is a prediction for the observable z [12, Chap. 1].

4.2 Sampling from the posterior distribution

4.2.1 Monte Carlo integration

Before dealing with the concept of sampling from the posterior distribution $p(\theta|y)$, we want to present the *Monte Carlo method* for numerical integration. The main application of this method is the computation of complex integrals of multivariate functions, such as expected values of the form:

$$E_p[f(x)] = \int_{\mathbb{R}^n} f(x) p(x)\, dx \quad x \sim p(x), \quad x \in \mathbb{R}^n \tag{4.6}$$

approximating this integral through a sample mean.

In fact, if $\{x_t,\, t = 1, ..., N\}$ are N independent and identically distributed realizations of the random process x, following the distribution p, then, for the *strong law of large numbers*, the sample average converges almost surely towards the expected value (4.6). Formally this reads:

$$Pr\left(\lim_{N\to\infty} \bar{f}_N(x) = \lim_{N\to\infty} \frac{1}{N} \sum_{t=1}^{N} f(x_t) = E_p[f(x)] \right) = 1 \tag{4.7}$$

where

$$x_t \sim p(x) \quad i.i.d. \quad \forall\, t = 1, ..., N, \quad x_t \in \mathbb{R}^n. \tag{4.8}$$

In this way it is possible to numerically calculate multidimensional integrals that are usually not solvable in an analytical way, because of the high dimension of the problem, and this represents one of the most important and common issues in Bayesian statistics.

The question that arises now concerns the simulation of the samples x_t from the random distribution of interest $p(x)$, as indicated in equation (4.8). There are particular cases in which it is possible to sample directly from the desired density function, e.g. from uniform or normal distributions. If the direct sampling from $p(x)$ is not possible (or not computationally efficient), we have to consider other strategies, like the Markov chain Monte Carlo methods, described in the next Subsection, which use computer simulation of Markov chains in the parameter space.

4.2.2 Markov chain Monte Carlo methods: MCMC

Markov chain Monte Carlo (MCMC) methods are a class of algorithms for sampling from probability distributions based on constructing a Markov chain whose equilibrium distribution is the desired one. In a Bayesian approach the Markov chain is defined in such a way that the posterior distribution $p(\theta|y)$, in the given statistical inference problem, is the asymptotic distribution, and each state of the chain represents a sample of the parameter vector, named θ_k. The samples are drawn iteratively from approximate distributions, with the distribution of the sampled θ_k given all previous drawn values depending only on the last drawn value θ_{k-1}. Hence they represent draws form a Markov chain, but lose the property of independence between them. The key to the method's success, however, is not the Markovian property but rather the fact that, under suitable hypotheses, the generated chain $\{\theta_k\}$ has as asymptotic distribution the target posterior distribution $p(\theta|y)$ [12, Chap. 11].

This allows to use ergodic sample averages:

$$\hat{J} = \bar{f}_{N_s}(\theta) = \frac{1}{N_s} \sum_{k=1}^{N_s} f(\theta_k) \quad \theta_k \sim p(\theta|y), \quad \theta_k \in \mathbb{R}^p \tag{4.9}$$

to approximate desired posterior expectations:

$$J = E[f(\theta)] = \int_{\mathbb{R}^p} f(\theta) p(\theta|y)\, d\theta \quad \theta \sim p(\theta|y), \quad \theta \in \mathbb{R}^p \tag{4.10}$$

as described in the previous Subsection. In fact in a Bayesian framework the posterior distribution $p(\theta|y)$ contains all relevant information on the unknown parameter θ given the observed data y, and all statistical inference can be deduced from the posterior distribution, typically by evaluating integrals of the form (4.10). For example, point estimate for the parameter can be given by the posterior mean, i.e. $f(\theta) = \theta$, or prediction for unobserved data z is based on the posterior predictive distribution $p(z|y) = \int p(z, \theta|y) d\theta$, for which we obtain $f(\theta) = p(z|\theta)$ (see also equation (4.18) in Section 4.4). Another quantity of interest could be the *marginal posterior distribution* for each single parameter composing the vector $\theta \in \mathbb{R}^p$:

$$p(\theta_i|y) = \int ... \int p(\theta|y) d\theta_1 ... d\theta_{i-1} d\theta_{i+1} ... d\theta_p, \tag{4.11}$$

that can be calculated by means of Monte Carlo estimation too.

Hence the art of MCMC simulation is to set up a Markov process whose stationary distribution is the desired posterior $p(\theta|y)$, and run the simulation long enough that the distribution of the current draws is close enough to this stationary distribution. Practically one cannot however prove convergence, but only diagnose a negative result on non convergence [12, Chap. 11] and [31].

Several standard approaches to define such Markov chains exist, e.g. the Gibbs sampler or the Metropolis-Hastings algorithm, which is described in the next Subsection. Using these algorithms it is possible to implement posterior simulation in essentially any problem which allows pointwise evaluation of the prior distribution $p(\theta)$ and of the likelihood function $L_y(\theta)$ [12, Chap. 11] and [31]. Once the simulation algorithm has been implemented, and the samples generated, then it is absolutely necessary to check the convergence of the simulated sequence [12, Chap. 11]. We discuss how to check convergence in Section 4.3.

4.2.3 Metropolis-Hastings (MH) algorithm

There are cases in many statistical models in which the complete conditional posterior distributions $p(\theta_j|\theta_i, i \neq j, y)$ assume the expression of some known distributions from which direct sampling is possible, allowing efficient random variate generation. In these cases the *Gibbs sampler* represents an interesting simulation algorithm. But there are also many important applications in which this is not the case, requiring alternative MCMC algorithms.

The most general Markov chain simulation method is the *Metropolis-Hastings* (MH) algorithm [14, 24], of which the Gibbs sampler and the Metropolis scheme [30] are special cases. The algorithm proceeds as follows [12, Chap. 11] and [31]:

1. initialise θ_0, for which $p(\theta_0|y) > 0$, from a starting distribution $p_0(\theta)$;

2. at time $k \geq 1$, generate a proposal θ^* from some *proposal distribution* (or *jumping distribution*) $J_k(\theta^*|\theta_{k-1})$, where θ_{k-1} is the current state of the Markov chain (the choice of the proposal distribution $J_k(.)$ is discussed later);

3. compute the ratio
$$r = \frac{p(\theta^*|y)}{p(\theta_{k-1}|y)} \cdot \frac{J_k(\theta_{k-1}|\theta^*)}{J_k(\theta^*|\theta_{k-1})} \tag{4.12}$$
(N.B. the ratio r is always defined, since a jump from θ_{k-1} to θ^* can only occur if both $p(\theta_{k-1}|y)$ and $J_k(\theta^*|\theta_{k-1})$ are greater than zero);

4. compute the acceptance factor $\alpha(\theta_{k-1}, \theta^*) = \min\{1, r\}$ and set
$$\theta_k = \begin{cases} \theta^*, & \text{with probability } \alpha(\theta_{k-1}, \theta^*) \\ \theta_{k-1}, & \text{otherwise} \end{cases} \tag{4.13}$$

5. increase k by one unit and return to point 2.

The *Metropolis* algorithm [30] is a special case of the MH, in which the proposal distribution is a *symmetric* function, satisfying the condition $J_k(\theta_a|\theta_b) \equiv J_k(\theta_b|\theta_a)$, $\forall\, \theta_a, \theta_b$ and k. In this case the acceptance factor becomes:

$$\alpha(\theta_{k-1}, \theta^*) = \min\left\{1, \frac{p(\theta^*|y)}{p(\theta_{k-1}|y)}\right\}. \tag{4.14}$$

The acceptance/rejection rule of the Metropolis algorithm can be stated as follows: if the candidate θ^* at time k has probability $p(\theta^*|y) > p(\theta_{k-1}|y)$, then $\alpha(\theta_{k-1}, \theta^*) = 1$, i.e. all parameters with posterior probability greater than the one of θ_{k-1} are accepted. Otherwise if $p(\theta^*|y) < p(\theta_{k-1}|y)$, then θ^* is accepted with probability $\alpha(\theta_{k-1}, \theta^*) < 1$. In this way there can be also jumps towards regions with smaller posterior density, that otherwise would never be explored.

Both for the basic Metropolis algorithm and for the more general Metropolis-Hastings, some regularity conditions are requested to be able to prove the convergence of the Markov chain to the target posterior distribution, showing first that the simulated sequence is a Markov chain with a unique stationary distribution and second that this stationary distribution equals the target distribution. These properties requested for the Markov chain are: *ergodicity* (which comprises the properties of irreducibility, aperiodicity and recurrence), and *reversibility* respect to the distribution $p(\theta|y)$, that implies the property of invariance w.r.t. the same posterior distribution.

Under these mild regularity conditions the MCMC estimator (4.9) is also proved to be asymptotically unbiased and normally distributed.

We notice that to solve the MH algorithm it is necessary to be able to calculate the ratio r in (4.12) for all parameters (θ, θ^*), and to draw a sample θ^* from the proposal distribution $J_k(\theta^*|\theta)$, for all θ and k. The first point is ensured by the fact that in (4.12) we have a ratio of the posterior distributions calculated for θ^* and for θ_{k-1}. In fact in general the normalization factor $p(y)$ in the equation (4.2) is extremely difficult to compute, and in this way the presented algorithm can generate samples from $p(\theta|y)$ without knowing this constant of proportionality. It requires only that a function proportional to the desired density $p(\theta|y)$ be calculable, represented in this context by the product of the prior and of

the likelihood, as highlighted by equation (4.4). Finally step 4 requires the generation of uniform distributed random numbers [12, Chap. 11].

The choice of the proposal distribution $J_k(\theta^*|\theta_{k-1})$ is essentially arbitrary, and subject only to some technical constraints. The ideal jumping rule would be to sample the proposal θ^* from the target distribution, i.e. to have $J(\theta^*|\theta) \equiv p(\theta^*|y)$, $\forall\, \theta$. In this way the ratio r in (4.12) is always 1, and the draws θ_k are a sequence of independent samples from $p(\theta|y)$ [12, Chap. 11]. Since usually this algorithm is applied to problems for which direct sampling is not possible, some good properties for the proposal distribution are necessary and the success of the MCMC methods depends in general on how well the proposal distribution fits the target distribution. Allowing asymmetric jumping rules (in the case of MH), for example, could be useful in increasing the speed of the random walk. Other interesting and useful ideas to define an efficient jumping rule are presented in [11] and in [12, Chap. 11].

In the next Subsection we present an efficient variation of the MH simulation algorithm.

4.2.4 DRAM: Delaying Rejection Adaptive Metropolis

An efficient variation of the MCMC method consists in the delaying rejection adaptive Metropolis (called DRAM). The authors in [18] propose some strategies to combine efficiently two powerful ideas appeared in the literature about MCMC: adaptive Metropolis samplers [16, 17] and delaying rejection [40].

Delaying rejection (DR) is a strategy to modify the standard MH algorithm that is proved to outperform the original method in the Peskun absolute efficiency ordering [40]. This means that, using the DR, we obtain MCMC estimators (4.9) that have a smaller asymptotic variance for every function f, whose expectation relative to $p(\theta|y)$ we want to estimate [18]. The basic idea of this method is that, upon rejection happened in a MH step, instead of advancing time and saving the current position, a second stage move is proposed. The probability of the second proposal to be accepted is computed so that reversibility of the Markov chain relative to the target distribution is preserved. Such a process of delaying

rejection can be iterated for a fixed or random number of stages. Moreover DR allows partial adaptation of the proposal within each iteration of the simulation, since the higher stage proposals can depend on the candidates so far proposed and rejected. DR can be considered as a way of combining different proposals for MH: in order to better explore the parameter space, global moves are tried first and local moves follow later. Specific details about the DR algorithm are given in [18].

The *adaptive Metropolis* (AM) algorithm can be considered as a global adaptive strategy that in the DRAM method is combined with the local adaptive strategy provided by the DR. The main intuition behind this method is that updating the Metropolis jumping rule during the simulation can improve the value of the *acceptance rate*, i.e. the proportion of jumps that are accepted. There are in fact some proposed optimal values for the acceptance rate (see [11] and [12, Chap. 11]), relating to the specific proposal distribution being used, that can be better obtained with some adaptive simulation algorithms. In particular in the AM approach on-line tuning, that means modifying while the simulation is running, the Metropolis proposal distribution can be based on the past history of the chain. Due to this adaptation, the chain looses its Markovian and reversibility properties. Anyway the authors in [17] show that, under some regularity conditions on the adaptation scheme of the proposal, the AM preserves the desired stationary distribution [18]. More details and theory about the implementation of this method are presented in [16, 17, 18].

The intuition behind adaptive strategies is to learn from the information obtained during the simulation, and to tune the proposal to work in a more efficient way. There are in general plenty of strategies of combining AM or MH together with the DR approach, as highlighted in [18]. The authors show how a successful combination of the two algorithms, that modify the standard MH sampler, outperforms the original simple methods: the adaptation AM enhances the efficiency of the delaying rejection algorithm in cases where good candidates for the proposal distributions are not available. On the other hand the DR provides a systematic remedy for cases where the adaptation has difficulties to get started [18]. In their work the authors also prove the ergodicity of the combined approach and demonstrate with some test examples the efficiency of the method.

4.3 Convergence test

As already mentioned, the use of the MCMC estimator (4.9) requires the verification of two conditions related to convergence. First of all, the Markov chain has to converge asymptotically, i.e. for $N_s \to \infty$, to the desired posterior distribution $p(\theta|y)$. Second, even if this theoretical convergence is established, we need a convergence test to assess when to stop simulations in the practical implementation of the algorithm. The regularity conditions necessary to ensure the convergence to the target posterior distribution, mentioned also in Section 4.2 in the description of the MH algorithm, are: ergodicity, which includes irreducibility, aperiodicity and recurrence, and invariance, for which reversibility represents a sufficient condition. These properties have to be verified in the implementation of Gibbs or MH algorithms, especially when choosing the proposal distribution.

Practically more important than establishing theoretical convergence, is to recognize practical convergence. In fact a critical issue when using MCMC methods is how to determine when it is safe to stop sampling and use the samples to estimate characteristics of the distribution of interest. Practically we have to judge when sufficiently many transitions N_s have been simulated to obtain ergodic averages $\hat{J}$ of equation (4.9) close to the desired posterior expectation J, equation (4.10). Several formal convergence tests have been proposed in the recent literature [7]. For example Gelman and Rubin [10] propose to consider several independent parallel runs of the MCMC simulations. Convergence is then diagnosed if the differences of $\hat{J}$ across the parallel runs are within a reasonable range. Another famous convergence test was proposed by Geweke [13] and it is presented in the following Subsection. To better assess convergence of iterative simulation, it is recommended to compare different independently simulated sequences (at least two), with starting points drawn from an overdispersed distribution [12, Chap. 11].

4.3.1 Geweke test

In his work [13], Geweke recommends the use of methods from spectral analysis to assess convergence of MCMC sampler when the intent of the analysis is to estimate the mean

of some function f of the parameter θ. The sequence $f(\theta_k)$ computed with the simulated samples can be regarded as a time series.

The Geweke method rests on the assumption that the nature of the MCMC process and of the function f imply the existence of a spectral density $S_f(\omega)$ for this time series, that has no discontinuities at the frequency 0, i.e. there exists the finite value $S_f(0)$. If this assumption holds, then for the MCMC estimator (4.9) ($\bar{f}_{N_s}(\theta)$) of the expectation $E[f(\theta)]$ given by equation (4.10), based on N_s iterations of the algorithm, the asymptotic variance is $S_f(0)/N_s$. The square root of this asymptotic variance can be used to estimate the standard error of the mean. Geweke's convergence diagnostic after N_s draws of the MCMC sampler is calculated by taking the difference between the means $\bar{f}_A(\theta)$, based on the first n_A iterations, and $\bar{f}_B(\theta)$, based on the last n_B iterations:

$$\bar{f}_A(\theta) = \frac{1}{N_A}\sum_{k=1}^{N_A} f(\theta_k) \tag{4.15a}$$

$$\bar{f}_B(\theta) = \frac{1}{N_B}\sum_{k=n^*}^{N_s} f(\theta_k) \tag{4.15b}$$

where $1 < n_A < n^* < N_s$ and $n_B = N_s - n^* + 1$, and dividing by the asymptotic standard error of the difference, computed from spectral density estimates $\hat{S}_f^A(0)$ and $\hat{S}_f^B(0)$ for the two different pieces of the sequence. If the ratios n_A/N_s and n_B/N_s are fixed, with

$$\frac{n_A + n_B}{N_s} < 1, \tag{4.16}$$

and if the sequence $f(\theta_k)$ is stationary, then by the central limit theorem, the distribution of this diagnostic tends to a standard normal distribution as N_s tends to ∞:

$$Z_{N_s} = \frac{\bar{f}_A(\theta) - \bar{f}_B(\theta)}{\left(\frac{\hat{S}_f^A(0)}{N_A} + \frac{\hat{S}_f^B(0)}{N_B}\right)^{1/2}} \xrightarrow[N_s\to\infty]{d} \mathcal{N}(0,1) \tag{4.17}$$

Thus the value Z_{N_s} can be used to test the null hypothesis of $\bar{f}_A(\theta) = \bar{f}_B(\theta)$ and if this is rejected then it indicates that the chain has not converged yet. Geweke suggests taking $n_A = N_s/10$ and $n_B = N_s/2$ [4, 7].

4.4 Results on simulations: quality of data fit

With the support of the MATLAB toolbox MCMCSTAT, we estimated the model parameter θ by sampling from the posterior distribution $p(\theta|y)$, as explained in the first Sections 4.1, 4.2 and 4.3 of this Chapter.

In this Section we present and compare the results for both ODE models (2.3) and (2.6) obtained with the considered Bayesian approach, in particular the simulated trends of the steady state lipid levels in response to the different experimental conditions defined by the activity of the enzyme SMS1 (value of the input u).

Having a wide set of estimated parameters θ_k, $k = 1, ..., N_s$, obtained by sampling from the posterior distribution inside the prior support region (the orders of magnitude are 10^5 samples for the first model and 10^6 for the second one), an interesting analysis concerns the predictive power of the two models, evaluated in terms of their ability to fit the experimental data.

The idea of prediction of a model falls within the concept of making inference about an unknown observable quantity of interest, and in our practical case it consists in the computation of the distribution over all possible outputs of the model, i.e. the steady state solutions $\bar{z}_{i,u}, i = 1, 2, 3, u = 1, 2.2, 0.77$, given the knowledge of the dataset y. This concept can be formally expressed by the *posterior predictive distribution* $p(\bar{z}_{i,u}|\tilde{y})$, already presented in Section 4.1, where we consider the logarithm of the measurements, as explained in Section 3.2 about the statistical model:

$$p(\bar{z}_{i,u}|\tilde{y}) = \int p(\bar{z}_{i,u}, \theta|\tilde{y})\, d\theta \qquad \text{Marginalization} \tag{4.18a}$$

$$= \int p(\bar{z}_{i,u}|\theta, \tilde{y})\, p(\theta|\tilde{y})\, d\theta \qquad \text{Factorization of the joint distribution} \tag{4.18b}$$

$$= \int p(\bar{z}_{i,u}|\theta)\, p(\theta|\tilde{y})\, d\theta \qquad \bar{z}_{i,u} \text{ is independent of } \tilde{y} \text{ given } \theta \tag{4.18c}$$

Also in this context concerning the sampling from the posterior distribution, for all simulations the estimation of parameter was carried out in logarithmic scale with a log-uniform bounded prior distribution $p(\theta)$, spanning over four orders of magnitude, like in the previous case of the MLE, explained in Section 3.3. In this way the model parameter is expressed

as $\theta = 10^{\psi}$ and what the sampling algorithm estimates is a set of possible values for the log-transformed parameter $\psi = \log_{10}\theta$.

As concerns the bounds defining the prior distributions over all parameters we considered the interval values reported in the Tables 3.1 and 3.2, for the two dynamical models (2.3) and (2.6), respectively. Such support regions were centred approximately around the values of the maximum likelihood estimates $\hat{\psi}_{MLE}$.

4.4.1 Bayesian estimation results of the model without feedback

Considering the differential equation model (2.3), that describes the biochemical system using the hypothesis of constant ceramide influx at the TGN, represented by the parameter C_{in}, we carried out the sampling from the posterior distribution $p(\theta|\tilde{y})$ to estimate a set of possible values for the model parameter $\theta = (C_{in}, p_1, p_2, d_1, d_2, d_3, k_1, k_2) \in \mathbb{R}^8_+$, and afterwards we used these parameters for predictions of the model.

From a practical point of view we carried out the estimation of the log-transformed parameter $\psi = \log_{10}\theta$, inside the prior support regions listed in Table 3.1. To do this we employed the algorithm offered by the MATLAB function `mcmcrun`, choosing the "DRAM" sampling method, from the `mcmcstat` toolbox (see Appendix A).

Before performing the main run, a warm-up/tuning of the covariance based proposal distribution was carried out by simulating a sample of size 10^4, while in the subsequent main run we generated a chain of size of $8{\cdot}10^6$. Moreover to speed up the MCMC simulation and to have more samples of the parameter to analyse, two independent Markov chains of the same dimension were started in parallel. The acceptance rate was almost 58% for both chains. To test the convergence of the two Markov chains to the desired posterior distribution we used the Geweke method implemented inside the `mcmcstat` toolbox. In the considered simulation both chains and also the resulting merged chain obtained by concatenation of the two (consisting of 1.6 million samples for the parameter vector) passed the convergence test with a p-value of at least 0.9 in each sub-dimension, attesting the occurred convergence. Further details about the simulation can be obtained directly in the

MATLAB code reported in Appendix A.

In Figure 4.1 we can observe the trajectories of the three lipid concentrations relative to the three experiments, generated with 1000 different MCMC estimated values of the model parameter vector θ extracted from the posterior distribution $p(\theta|\tilde{y})$, according to the considered bounded prior distribution. We highlight with a continuous line, together with the new presented trajectories, also that simulated using the MLE parameter vector $\hat{\theta}_{MLE}$. We can notice that all trajectories simulated using the samples θ_k are spread in a region around the MLE generated trajectories for each lipid and each experiment, and moreover also in this context the general trends of such trajectories are similar to those relative to the maximum likelihood estimate.

The most interesting result is that, considering a single estimated value θ_k for the parameter vector, the respective simulated ceramide levels are constant and assume the same value for each experiment (see the first line of plots), as it happens also with the the maximum likelihood estimation applied to the first ODE model without feedback (see Section 3.3). This means that even the estimation by sampling from the posterior distribution $p(\theta|\tilde{y})$ gives as most likely resulting parameters those that leave the ceramide levels constant across all three experiments, since the considered model cannot capture the qualitative behaviour of ceramide at steady state in response to changes of the enzyme activity.

Figure 4.2 represents the posterior predictive distributions of the outputs of the model at steady state $p(\bar{z}_{i,u}|\tilde{y})$, $i = 1, 2, 3$ and $u = 1, 2.2, 0.77$, relative to the model (2.3) without feedback, estimated from the MCMC samples using equation (4.18). This calculated density of the predicted steady state solutions is represented with continuous dark grey lines, that have a characteristic bell-shape centred approximately around the maximum likelihood estimates. The maximum likelihood estimates are marked with a black vertical straight line, while the normalized experimental data are plotted with vertical grey lines with the respective normalized standard deviations marked also in grey with dotted lines.

The panels of Figure 4.2 are transposed compared to that of Figure 4.1 for better comparison. From this Figure we can observe the same results, already highlighted by

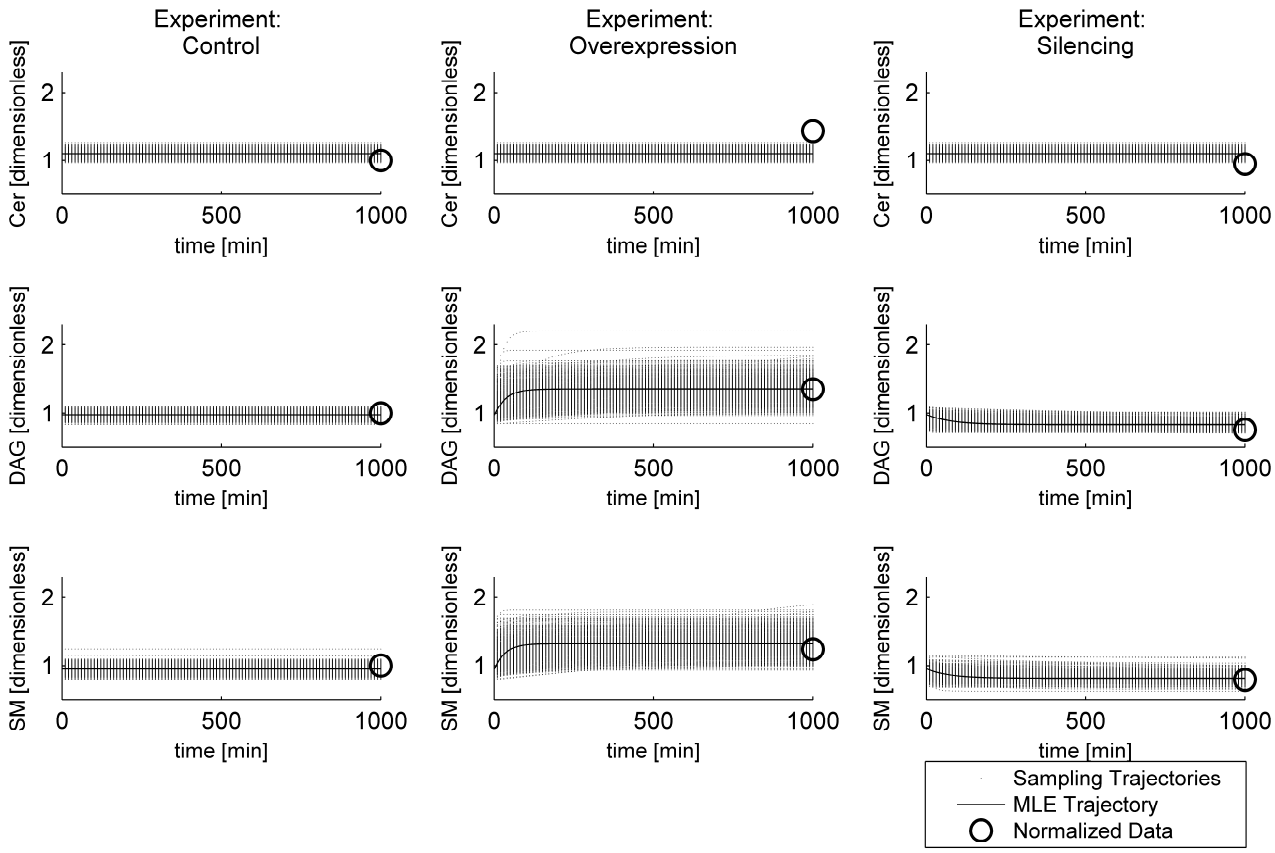

Figure 4.1: Trajectories of the concentrations of ceramide, DAG and SM obtained with the ODE model (2.3) with constant ceramide influx C_{in} simulated with the MCMC samples drawn from the posterior distribution and with the MLE parameters of Table 3.1, plotted together with the experimental data at steady state for the three different experimental conditions.

the previous Figure 4.1, that confirm also the hypothesis of our theoretical investigations described in Section 2.4. In fact we see in the first column of plots how the steady state levels of ceramide predicted by the model are constant across all the three experiments, since the qualitative behaviour of ceramide for different experimental conditions cannot be described by this model in the right way. The two panels of the Figure relative to the posterior predictive distribution of ceramide steady states in the cases of overexpression and silencing are identified with two black exclamation marks, because these are the cases

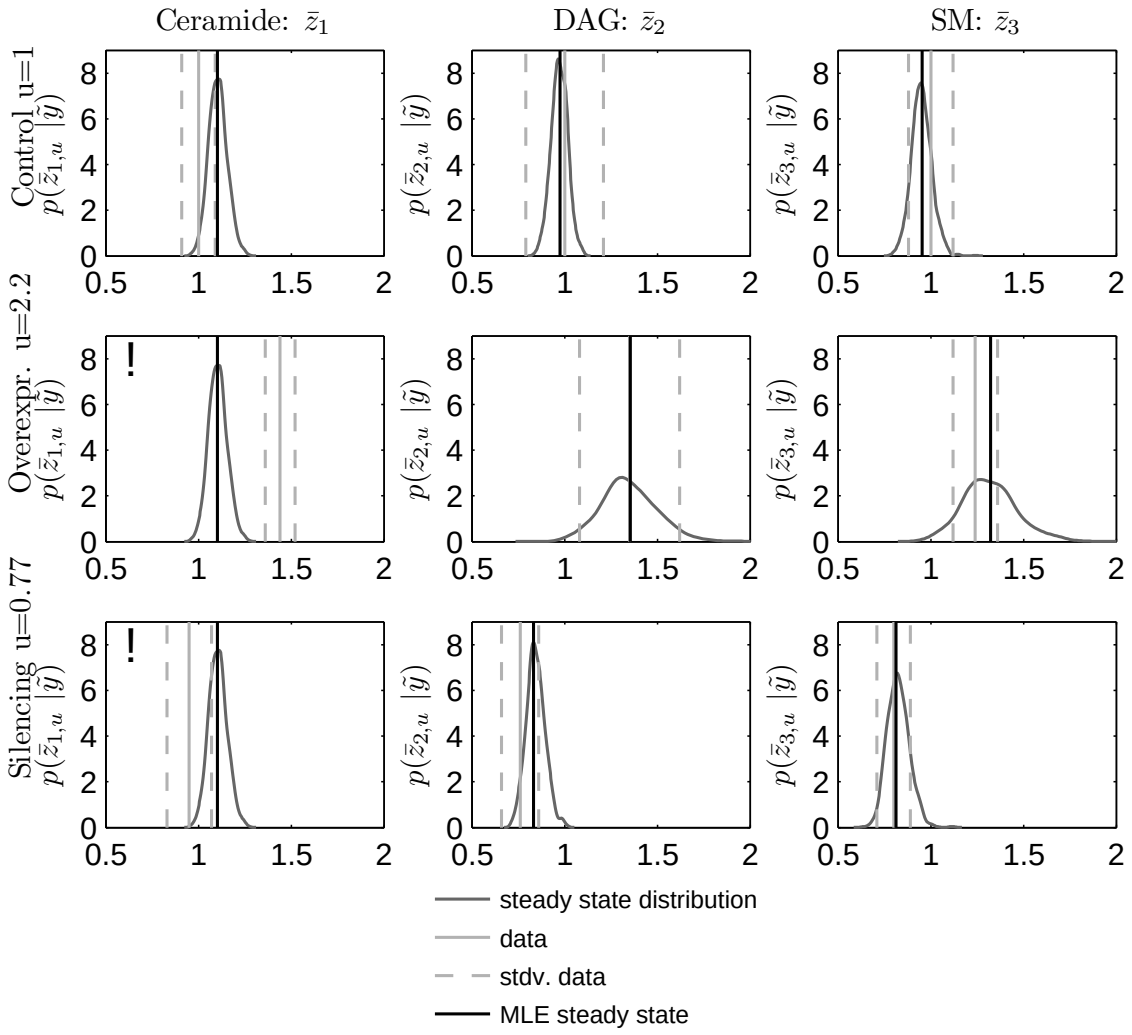

Figure 4.2: Posterior predictive distributions of the steady state levels of ceramide, DAG and SM for model (2.3).

in which the model fails to explain the experimental data relative to the changes of the concentrations of ceramide in response to SMS1 manipulations. Instead the predicted steady state levels of DAG and SM are more sensitive to the changes of the SMS1 activity, showing a significant increase or decrease in the overexpression and silencing experiments, respectively, compared to the control case. Moreover the posterior predictive distributions of the steady state levels of DAG and SM are estimated to be very similar in each experimental condition, capturing quite well the experimental results.

Finally one can notice that the variances of the model predictions are rather small, especially in the case of ceramide. The largest variances can be seen in the overexpression case for DAG and SM. This fact can be a hint that this first model is wrong or nor flexible enough, such that it cannot well capture all experimental data.

4.4.2 Bayesian estimation results of the model with feedback

In this Subsection we report the results of the MCMC sampling relatively to the modified ODE system (2.6), that considers the feedback regulation term $C_{in}(DAG) = a \cdot DAG$ in the place of the simple constant ceramide influx C_{in}. Considering this model we carried out the sampling from the posterior distribution $p(\theta|\tilde{y})$ to estimate a set of possible values for the model parameter $\theta = (a, p_1, p_2, d_1, d_2, d_3, k_1, k_2) \in \mathbb{R}^8_+$, and consequently used these parameters for predictions of the model. The technical details for the estimation and the employed algorithm are the same described in the previous Subsection.

A warm-up/tuning of the covariance based proposal distribution was performed by simulating a sample of size 10^4, as in the case of the first model, while in the subsequent main run we generated a larger chain of size of $3 \cdot 10^6$, having encountered more difficulties to reach the convergence to the desired posterior distribution. Also in this case two Markov chains of the same dimension were started in parallel. The acceptance rate was about 32% for both chains. As regards the convergence test, in the considered simulation both chains and also the resulting merged chain obtained by concatenation of the two (consisting of 6 million samples for the parameter vector) passed the convergence test with a p-value of at least 0.8 in each sub-dimension, attesting the occurred convergence. Further details about the simulation can be obtained directly in the MATLAB code reported in Appendix A.

Figure 4.3 shows the trajectories of the three lipid concentrations relative to the three experiments, generated with 1000 different MCMC estimated values of the model parameter vector θ extracted from the posterior distribution $p(\theta|\tilde{y})$, according to the considered log-uniform bounded prior distribution, whose boundaries are listed in Table 3.2.

Also in this case all trajectories simulated using the samples θ_k are spread in a region

around the trajectories obtained with the MLE, following the same trend of these last, with different densities or width of the trajectory distributions depending on the particular lipid or experiment.

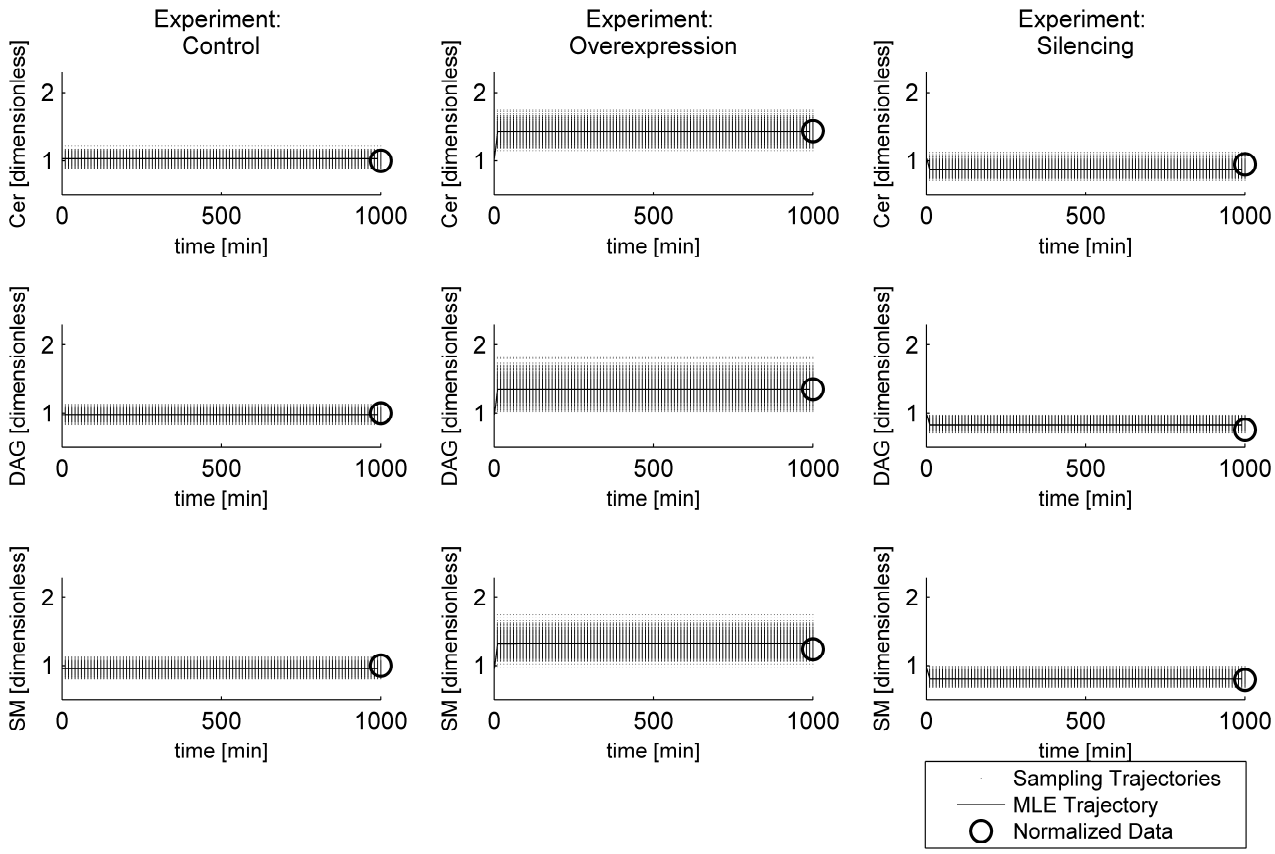

Figure 4.3: Trajectories of the concentrations of ceramide, DAG and SM obtained with the ODE model (2.6) with feedback regulation $C_{in}(DAG)$ simulated with the MCMC samples drawn from the posterior distribution and with the MLE parameters of Table 3.2, plotted together with the experimental data at steady state for the three different experimental conditions.

As most interesting result we can immediately notice how in this second case there is a very different behaviour of the simulated trajectories of ceramide across the three different experiments. Unlike the constant trend of the trajectories of ceramide predicted by the first model without feedback (see Figure 4.1), in this case we observe that such trajectories

simulated with the modified model are more sensitive to the manipulation of SMS1 activity compared with the original model and they follow in a qualitative way the trend of the experimental data.

Figure 4.4 shows the posterior predictive distributions of the simulated steady state levels $p(\bar{z}_{i,u}|\tilde{y})$.

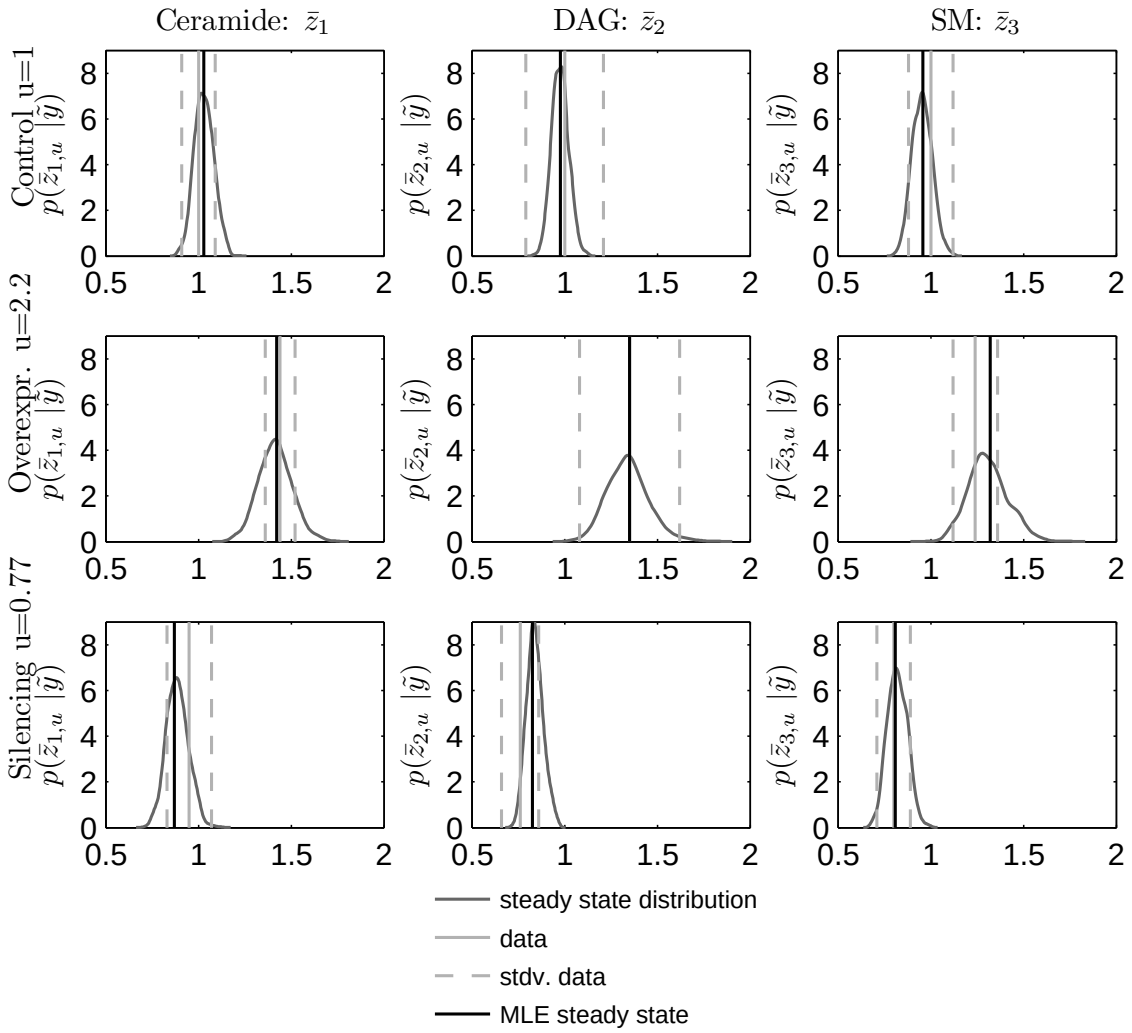

Figure 4.4: Posterior predictive distributions of the steady state levels of ceramide, DAG and SM for model (2.6).

While the distributions relative to DAG and SM look very similar to those of Figure 4.2, even though the MLE and hence the prior support regions are very different from those of

model (2.3), the main difference, as already underlined, can be seen in the ceramide steady state levels. These are in fact now more sensitive to the changes of the enzyme activity, and ceramide shows an increase after SMS1 overexpression and a less pronounced decrease after silencing. This is in accordance with the data and thus confirms the hypothesis that the model including the feedback regulation term is qualitatively able to capture the experimental findings.

4.4.3 Marginal parameter distribution

In this Subsection we present the results concerning the computation of the marginal posterior distributions for each single parameter θ_i relative to both ODE models (2.3) and (2.6), calculated using the formula (4.11). In practice the computed marginal distributions are all relative to the components of the log-transformed parameter $\psi = \log_{10}\theta$. We highlight another time the fact that we chose the boundaries for the log-uniform prior distributions taking intervals of 4 orders of magnitude centred around the values of the maximum likelihood estimated parameter vector $\hat{\psi}_{MLE}$. This can be easily noticed in the two following Figures 4.5 and 4.6, which represent the considered marginal posterior distributions for the two different ODE models. Figure 4.5 shows the marginal distributions $p(\psi_i|\tilde{y})$ relative to the first ODE model (2.3), with:

$$\psi_i \in \{\log_{10} C_{in}, \log_{10} p_1, \log_{10} p_2, \log_{10} d_1, \log_{10} d_2, \log_{10} d_3, \log_{10} k_1, \log_{10} k_2\}.$$

These distributions were generated by kernel density estimation from the obtained MCMC samples, and they are plotted in the Figure together with the maximum likelihood estimates, marked with dark grey vertical lines, and with the 5% and 95% percentiles, marked with grey dotted lines. We can observe that the marginals of the ceramide production, C_{in}, of the ceramide degradation rate, d_1, and of the forward Michaelis-Menten constant, k_1, are almost uniform over the intervals of the prior distribution. This result highlights the fact that data provide little information about these parameters and demonstrates that within the given prior support no particular model parametrization better explains the

data. The marginals of the other 6 parameters are slightly more informative, although all parameters remain only vaguely determined over several orders of magnitude.

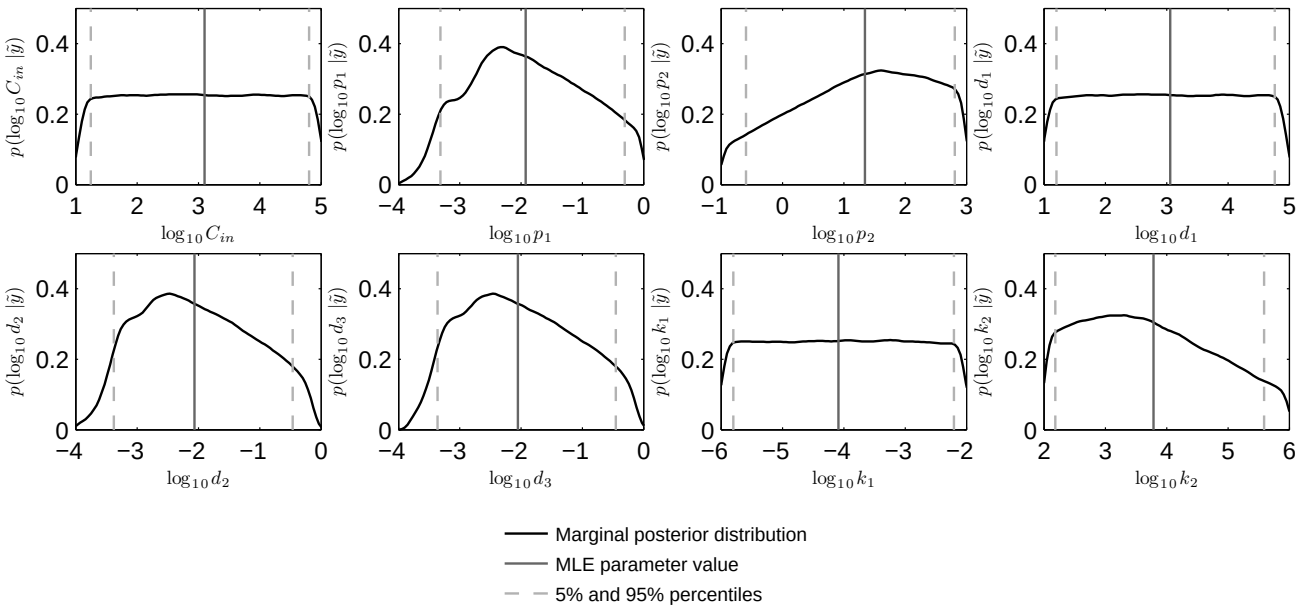

Figure 4.5: 1D Marginals of log-transformed model parameters estimated by Monte Carlo integration from MCMC sampling relative to the ODE model (2.3).

As regards the modified ODE model (2.6), the relative parameters' marginal posterior distributions are plotted in Figure 4.6. We notice that in this case we can derive more information about the distribution of parameters, since the variances of the marginal distributions are slightly smaller, in particular for the parameters a and d_1. Nevertheless, also in this case concerning the revised model, parameters are not determined in a precise way, and this fact suggests that a lot of different parametrizations could reproduce the good fit quality.

4.4.4 Comparison of the results of the two models

Summarizing the results of the two previous Subsections, now we want to make a comparison between the features of the two ODE models. We can maintain that, overall, the results obtained with the parameter estimation by sampling from the posterior distribution

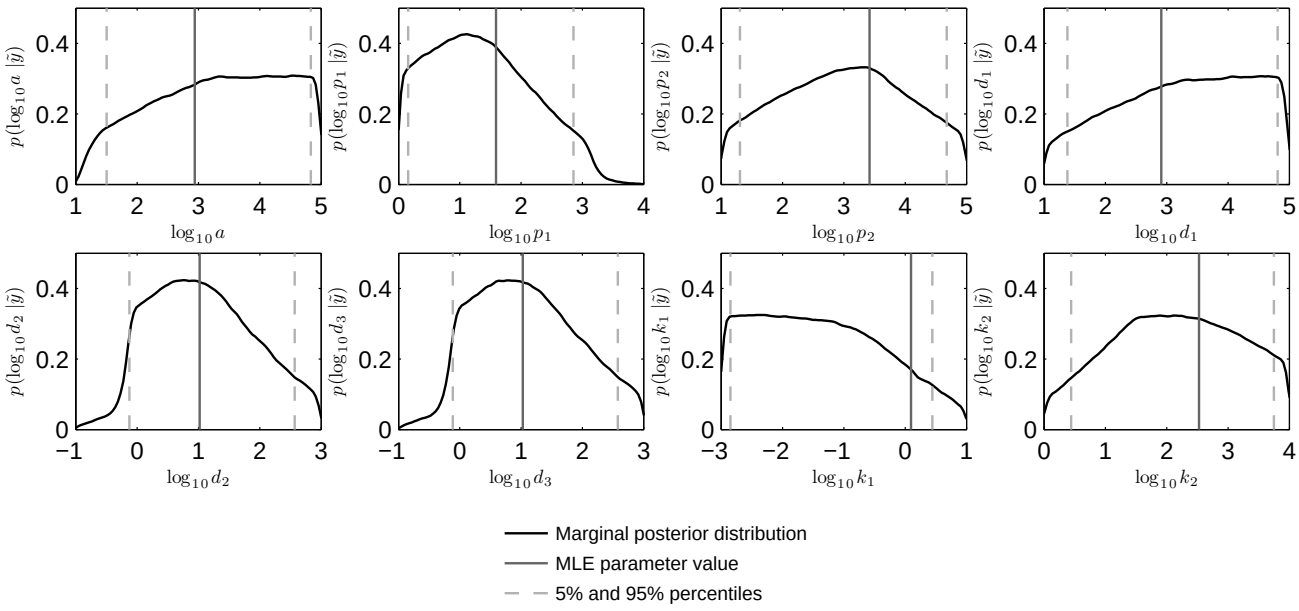

Figure 4.6: 1D Marginals of log-transformed model parameters estimated by Monte Carlo integration from MCMC sampling relative to the ODE model (2.6).

$p(\theta|\tilde{y})$ confirm our theoretical analysis described in Section 2.4, concerning the qualitative changes of the steady state lipid composition in response to SMS1 manipulation, and also the results of the simulations of the MLE, presented in Section 3.3.

First of all we can affirm, in general, that the first model (2.3) with constant ceramide influx C_{in} is not able to capture qualitatively the experimentally observed changes of the steady state ceramide concentrations following manipulations of the activity of the enzyme SMS1 that drives the considered reaction. In fact all parameter estimations produce as most probable value of the estimated parameter vector the one that leaves the concentration of ceramide constant across all experiments, since the model would predict changes of such steady state levels in the opposite direction with respect to that observed in the experimental results. Besides the results of this Bayesian analysis, the mathematical investigation of Section 2.4 proves that, regarding the chemical reaction in isolation, i.e. without considering the feedback term, the changes in lipid steady state levels of both sides of the reaction must have opposite signs in response to changes in SMS1 activity, and

this outcome holds independently of the exact kinetics modelling the reversible enzymatic reaction. In this way both analytical and statistical results qualitatively reject model (2.3).

Instead the modified differential equation system (2.6) that includes the positive feedback regulation from DAG level to ceramide presents a highly improved data fit quality. In fact in this case the changes in ceramide levels in response to SMS1 manipulations can qualitatively be captured for a wide range of estimated parameter values.

Even though the marginal posterior distributions of the single components of the model parameter are not so informative also in the considered revised model, and the parameters remain vaguely determined over several orders of magnitude, we obtain anyway a significant improvement of the data fit, and the predictions of the model can capture the qualitative behaviour of the experimental findings.

Conclusions

Summary of results and discussion

The aim of this thesis was to build a dynamic mathematical model, based on chemical reaction kinetics, in order to describe the reversible metabolic conversion of ceramide (Cer) and phosphatidylcholine (PC) into sphingomyelin (SM) and diacylglycerol (DAG), catalysed by the enzyme sphingomyelinsynthase 1 (SMS1). As experimental dataset to be used for model parameter estimation, we considered lipid concentrations at steady state, measured under different experimental conditions in which the activity of SMS1 was altered. In response to these SMS1 manipulations, changes in lipid composition were observed and we aimed at describing qualitatively these results with our mathematical model. We proved that a simple model that considers the reversible reaction in isolation fails to explain the considered experimental findings. In particular, changes of ceramide levels at steady state in response to SMS1 overexpression and silencing could not be captured by the model. Consequently to these results, and based on biological knowledge, we modified the first ODE model by considering a positive feedback regulation from DAG to ceramide, and thus modelling the influx of this last lipid at the TGN as an increasing linear function of the concentration of DAG. The validity of this choice to improve our dynamical model was demonstrated both in a theoretical way, using the hypothesis of the *Implicit function theorem*, and with statistical inference approaches. In fact, using MLE- and sampling-based statistical methods, we showed that a simple linear feedback term was sufficient to explain the observed data qualitatively, with a significant improvement of the quality of fit.

We underline the fact that, even though we used dynamic models (differential equation systems) to describe the SMS1 driven reaction, what influences the parameter estimation is only the equilibrium situation of the system, since all measurements are steady state lipid concentrations. This means to consider the system at steady state, i.e. $\mathbf{f}(\bar{\mathbf{x}}, \theta) = 0$.

From the obtained results we maintain that feedback regulation might be an essential feature of the SMS1 driven conversion of ceramide into SM, and that the effects caused *in vivo* by this feedback control may not be explained by a model that considers the reversible reaction in isolation. In this study we have motivated the existence of such a feedback regulation with an indirect influence of DAG on the efficiency of ceramide transport to the trans-Golgi network, regulated by protein kinase D (PKD) and by the ceramide transport protein CERT [41]. Anyway we cannot affirm with certainty that this biochemical pathway has the principal contribution to the studied feedback regulation. In fact, there could be other unknown causes that underlie this phenomenon, and moreover the effects on ceramide levels in response to modulation of the activity of SMS could be differently regulated depending on the specific cellular context. From a biological point of view, a precise knowledge about this reaction taking place at the trans-Golgi network is not yet available, and further research should be conducted. For this reason we have to pay attention about the conclusions that we draw from our results, even if we maintain that further investigation in the direction of the feedback regulation should be supported. Besides these considerations concerning the biological fundamentals of this thesis, we have to be careful about the meaning that we attribute to the obtained results. In fact, our study clearly shows that model errors can have a drastic effect on parameter estimation. For example, as we described in detail in Sections 3.3 and 4.4 concerning the results of parameter estimation, we can notice that the estimated parameters for the two ODE models, which differ formally only for the single term C_{in}, show differences for several orders of magnitude. Moreover we can notice that the choice of model parameter boundaries is extremely influential on the predictive power of the ODE model, and in general it is an important aspect in the construction of mathematical models describing biochemical cellular reaction networks. These facts put parameter values estimated from experimental

data and simple models much into question, and moreover justify the use of statistical sampling methods, which also provide information about uncertainties due to modelling errors.

Future work

As already underlined, from a biological point of view it would be fundamental to further investigate the functioning of the complex secretion regulatory network, in particular the dynamic relations between the lipids involved in the SM synthesis reaction, in order to bring interesting improvements and future possible developments of this study.

Additional investigation of the revealed feedback mechanism could be developed and supported by other experimental datasets. In particular, from a modelling point of view, time series data of lipid concentrations would bring much more information for estimating model parameters, improving in this way also the investigation of parameter bounds. Finally it would be interesting to consider different cellular systems, to have a general overview of the problem and to understand possible similarities and differences.

Acknowledgements

At the end of this work and of my master studies, I would like to write some acknowledgements to a few important people that have supported me in many ways during these years spent as a student at the University of Padova and, relatively to the last passed year, at the University of Stuttgart. I hope to express me in a proper and comprehensible way, since English is not my mother language. Surely, if I would write in Italian, I could better put my feelings into words. First of all I want to thank my supervisor of the University of Stuttgart, Nicole Radde, who gave me the opportunity to make my whole thesis at the Institute for Systems Theory and Automatic Control (IST), directed by Professor Frank Allgöwer. She taught me a lot of new interesting things, especially in the field of Systems Biology, and she tutored me for the entire duration of my thesis. Together with her, I have to mention Patrick Weber, research assistant at the IST, who also supervised me in relevant measure, providing me helpful advices for my thesis, and also his own MATLAB code as model for my simulations. It was a real pleasure and an enriching experience to work with both of them! Then I want to thank some Professors of the Department of Information Engineering of the University of Padova: my supervisor Professor Augusto Ferrante, Professor Sandro Zampieri, who encouraged my choice to make my thesis at the Insitute IST of the University of Stuttgart, and finally Professor Giovanni Marchesini, who supervised my bachelor thesis. He made arise in me the passion for the Systems Theory and always gave me precious advices for my university career. Departing from the academic world, I would like to thank two people that have always been present in my life, my parents. Without them I would not be the person that I am. In 25 years of life they always gave me

an excellent education. Thanks to them I have always had the opportunity to grow up in a healthy way, to study, to cultivate my passions and hobbies, to have fun, to travel, and finally to spend my last year in Stuttgart, choice that somehow signed my future. Grazie mamma e papà! A special thanks goes also to my little sister, Eleonora. Grazie Ele for our funny jokes and for our awesome tie! As last but not least, I have to thank another special person for his constant support, and to be able to patiently understand and "stand" me also in my irrational moments. Grazie Checco! To conclude I want to thank all my awesome friends, from Padova, but also from many countries around the world, for always being there when there is the need, even if we do not live in the same place, and for sharing with me so many special moments. Thanks everybody! Grazie a tutti!

Appendix A

Programming with Matlab

All basic numerical computations have been performed using the software MATLAB, version R2010b (32 bit). For the definition of the model and the management of the experimental data the toolboxes SBPD and SBTOOLBOX2 [34] have been used, which offer specifically a powerful environment in which to build models of biological systems. For the numerical integration of the differential equation systems (2.3) and (2.6) the toolbox SBTOOLBOX2 employs the particular integrator `CVODE` from SUNDIALS[1], which is a solver for stiff and non-stiff ordinary differential equation (ODE) systems (initial value problem) given in explicit form [6]. As options for the absolute and relative error tolerances of the MEX integrator in the case of the first model we set to at least `options.abstol=1e-12` and `options.reltol=1e-12`. Instead for the second ODE model, having difficulties to carry out the integration of the ODE system, we used less strict constraints (`options.abstol=1e-6` and `options.reltol=1e-6`), and moreover, to avoid the error `CV_TOO_MUCH_WORK` that occurred during the simulation with MATLAB caused by stiffness problems, we increased the number of maximum internal steps to `options.maxnumsteps=1000000`.

[1] `https://computation.llnl.gov/casc/sundials/`

A.1 Project internal structure

To employ the toolboxes SBPD and SBTOOLBOX2 we need a precise structure of the project, in which we define the model, the experimental data (measurements) and the different experiments. In practice our project consists of two folders for the simulations relative to the two different ODE models (2.3) and (2.6). Each project-folder must contain in particular two specific subfolders: (1) subfolder "`models`", in which we put the files with the definition of the ordinary differential equation models (e.g. `modelName.txt`) (2) subfolder "`experiments`", which contains other subfolders, with particular files describing the experiments (e.g. `experiment1.exp` contained in the folder `Experiment1`).

In this Section we describe in particular all details relative to the first ODE model without the positive feedback term, since, from the informations already given in the text about some simulations' details, one can easily obtain the code for the simulations concerning the second ODE model with feedback. In Table A.1 are presented the specific subfolders and files contained in the project-folder relative to the simulations for the first ODE model.

Table A.1: Project folder: `SMS1_Project_1`.

Folder	Contained subfolders and files
`models`	`SMS1_model_noFB.txt`
`experiments`	`Experiment1` → `experiment1.exp`
	`Experiment2` → `experiment2.exp`
	`Experiment3` → `experiment3.exp`
	`Experiment4` → `experiment4.exp`

A.1.1 ODE model and experiments

The code written in the file `SMS1_model_noFB.txt`, defining all properties of the ODE model without feedback is:

```
********** MODEL NAME
SMS1 Reaction system 1
********** MODEL NOTES
First model, where Cin = constant.
All variables are dimensionless (normalized) and parameters have physical units.
PC and u are considered as parameters, and their values are specified
for each experiment. There are 10 parameters, but only 8 (without u and PC)
are estimated.
State variables:
x1 = Cer; x2 = DAG; x3 = SM
Parameters:
Cin, p1, p2, d1 = dC, d2 = dDAG, d3 = dSM, k1, k2, u = SMS, PC
Parameters are estimated in logarithmic scale.
The outputs are defined as logarithm of the state variables,
because we use a log-normal distribution error model.
********** MODEL STATES
d/dt(x1) = s1 - s4*x1 - s2*u*x1*PC/(x1*PC+s7) + s3*u*x2*x3/(x2*x3+s8)
d/dt(x2) = - s5*x2 + s2*u*x1*PC/(x1*PC+s7) - s3*u*x2*x3/(x2*x3+s8)
d/dt(x3) = - s6*x3 + s2*u*x1*PC/(x1*PC+s7) - s3*u*x2*x3/(x2*x3+s8)
x1(0) = 1
x2(0) = 1
x3(0) = 1
********** MODEL PARAMETERS
Cin = 1
p1 = 1
p2 = 1
d1 = 1
d2 = 1
d3 = 1
```

```
k1 = 1
k2 = 1
u = 1
PC = 1
********** MODEL VARIABLES
Cer = log(x1)
DAG = log(x2)
SM = log(x3)
s1 = 10^Cin
s2 = 10^p1
s3 = 10^p2
s4 = 10^d1
s5 = 10^d2
s6 = 10^d3
s7 = 10^k1
s8 = 10^k2
********** MODEL REACTIONS
********** MODEL FUNCTIONS
********** MODEL EVENTS
********** MODEL MATLAB FUNCTIONS
```

We report also as example the code contained in the file `experiment1.exp` relative to the description of the first experiment, in which we considered the values taken from [8]:

```
********** EXPERIMENT NAME
Experiment 1 for SMS1 Reaction system 1
********** EXPERIMENT NOTES
The input u expresses the SMS1 activity (% of WT-control).
PC is considered constant in the model (dPC/dt = 0), the value set for this
experiment is the one given in the table, considering only the mean without
```

```
SD, normalized to the value of the control experiment.
In this experiment there is no SMS1 overexpression or knockdown.
********** EXPERIMENT INITIAL PARAMETER AND STATE SETTINGS
u = 1
PC = 294/294
********** EXPERIMENT PARAMETER CHANGES
********** EXPERIMENT STATE CHANGE
```

A.1.2 Structure of the main script

We present here an itemized scheme of the steps constituting the main MATLAB script for the simulations.

- Enable the parallel language features in the MATLAB language (e.g. parfor) by creating a special job on a pool of workers for parallel computation, using the MATLAB function `matlabpool`.
- Create a new SBmodel from the text file with extension `.txt` containing the description of the model, using the MATLAB function
  `model = SBmodel('SMS1_model_noFB.txt')`.
- Define random initial conditions for the state variables $x_i(0)$, $i = 1, 2, 3$.
- Convert the SBmodel to a high performance "Matlab EXecutable C-code" (MEX) model and link it with the CVODE integrator from SUNDIALS, using the MATLAB function `SBPDmakeMEXmodel`.
- Prepare a special experiment project structure, named `expStruct`, containing the description and the features of each of the four experiments.
- Read all experimental data from the excel file containing the normalised mean values and standard deviations for the three lipids concentrations Cer, DAG and SM, using the function `BioDataImport`.

- Using the instruction `RelDataCruncherAdv(model,expStruct)`, process the data for a MEX based parameter estimation.
- Translate data in logarithmic scale and enhance the time vector, with the function `PEenhanceTvector`.
- Define the lower and upper bounds for each of the parameters to be estimated.
- Start the constrained optimization to maximize the likelihood function and thus find the MLE parameter vector $\hat{\theta}_{MLE}$, giving minus the logarithm of the likelihood as input of the optimizing function `fmincon`.
- Merge the SBmodel with all 4 defined *in silico* experiments using the function `SBmergemodexp`, and simulate it with the estimated MLE parameter vector $\hat{\theta}_{MLE}$ using the function `SBPDsimulate`, in order to obtain the relative trajectories of the three state variables, i.e. the three lipid concentrations of Cer, DAG and SM.
- Run the burn-in sampling, defining `dram` as adaptation method, and afterwards the sampling main run with two parallel Markov chains, using the function `mcmcrun` from the MATLAB toolbox MCMCSTAT.
- Chain merging and convergence analysis using the function `geweke`, always offered by the toolbox MCMCSTAT.
- Computation of posterior predictive distributions of the model steady states and of the one dimensional marginals of the parameter posterior distribution, using the MATLAB function `ksdensity`.

A.1.3 Matlab functions

In Table A.2 we report a list of all self-written functions and of the most important functions already implemented in MATLAB that we used in our code. For each method we specify the usage code and a brief description of the specific function.

[2]http://helios.fmi.fi/ lainema/mcmc/mcmcrun.html

Table A.2: MATLAB functions.

Method	Usage and Specific function
`BioDataImport`	`[stdv MuN] = BioDataImport('*.xls')` → imports *.xls data and produces *.csv files for each experiment and calculates standard deviation cell variable following four diffent rules.
`RelDataCruncher`	`[timevector, ISvalues, IPvalues, stdv0, expmodel, Iparametervector]= RelDataCruncher(SBmodel, expStruct)` → preprocesses the data for a MEX based parameter estimation.
`PErelativeHTspeed`	→ forms objective function (likelihood) to be optimized
`fmincon`	`[x,fval,exitflag,output,lambda,grad,hessian] = fmincon(fun,x0,A,b,Aeq,beq,lb,ub,nonlcon,options)` → finds the minimum of constrained nonlinear multivariable function.
`mcmcrun`	`[results,chain,s2chain,sschain] = mcmcmrun(model,data,params,options)` → MATLAB function for the MCMC run. The user provides her own Matlab function to calculate the "sum-of-squares" function for the likelihood part, e.g. a function that calculates minus twice the log likelihood [2].
`geweke`	`[z,p] = geweke(chain,a,b)` Geweke's MCMC convergence diagnostic. Test for equality of the means of the first a% (default 10%) and last b% (50%) of a Markov chain.
`ksdensity`	`[f,xi] = ksdensity(x)` → computes a probability density estimate of the sample in the vector x. `f` is the vector of density values evaluated at the points in `xi`. The estimate is based on a normal kernel function, using a window parameter (width) that is a function of the number of points in `x`.

Bibliography

[1] U. Alon. 2006. An Introduction to Systems Biology - Design Principles of Biological Circuits. *Mathematical and Computational Biology Series.* Chapman & Hall/CRC.

[2] D.J. Barnes, and D. Chu. 2010. Introduction to Modeling for Biosciences. Springer.

[3] C.L. Baron, and V. Malhotra. 2002. Role of Diacylglycerol in PKD Recruitment to the TGN and Protein Transport to the Plasma Membrane. *Science* **295:**325-328.

[4] S.P. Brooks, and G.O. Roberts. 1998. Assessing Convergence of Markov Chain Monte Carlo Algorithms. *Statistics and Computing.* **8:**319-335.

[5] T.L. Burgess, and R.B. Kelly. 1987. Constitutive and Regulated Secretion of Proteins. *Ann. Rev. Cell Biol.* **3:**243-293.

[6] S.D. Cohen, and A.C. Hindmarsh. 1996. CVODE, A Stiff/Nonstiff ODE solver in C. *Computers in Physics.* **10:**138-143.

[7] M.K. Cowles, and B.P. Carlin. 1996. Markov Chain Monte Carlo Convergence Diagnostics: A Comparative Review. *Journal of the American Statistical Association.* **91(434):**883-904.

[8] T. Ding, Z. Li, T. Hailemariam, S. Mukherjee, F. R. Maxfield, M.-P. Wu, and X.-C. Jiang. 2008. SMS overexpression and knockdown: impact on cellular sphingomyelin and diacylglycerol metabolism, and cell apoptosis. *J. Lipid Res.* **49:**376-385.

[9] T. Fugmann, A. Hausser, P. Schffler, S. Schmid, K. Pfizenmaier, and M.A. Olayioye. 2007. Regulation of secretory transport by protein kinase D-mediated phosphorylation of the ceramide transfer protein. *J. Cell Biol.* **178:**15-22.

[10] A. Gelman, and D. Rubin. 1992. lnference from Iterative Simulation Using Multiple Sequences. *Statistical Science.* **7(4):**457-472.

[11] A. Gelman, G.O. Roberts, and W.R. Gilks. 1996. Efficient Metropolis jumping Rules. *Bayesian Statistics.* **5:**599-607.

[12] A. Gelman, J.B. Carlin, H.S. Stern, and D.B. Rubin. 2004. Bayesian data analysis. *Texts in Statistical Science.* Chapman & Hall/CRC. 2nd edition.

[13] J. Geweke. 1992. Evaluating the accuracy of sampling-based approaches to the calculation of posterior moments. *Bayesian Statistics 4* (J.M. Bernardo, J. 0. Berger, A. P. Dawid and A. F. M. Smith, eds.) 169-193. Oxford University Press.

[14] W.R. Gilks, S. Richardson, D. Spiegelhalter. 1996. Markov Chain Monte Carlo in Practice. Chapman & Hall/CRC.

[15] E.M. Griner, and M.G. Kazanietz. 2007. Protein kinase C and other diacylglycerol effectors in cancer. *Nat. Rev. Cancer.* **7:**281-294.

[16] H. Haario, E. Saksman, and J. Tamminen. 1999. Adaptive proposal distribution for random walk Metropolis algorithm. *Comp. Stat.* **14:**375-395.

[17] H. Haario, E. Saksman, and J. Tamminen. 2001. An adaptive Metropolis algorithm. *Bernoulli.* **7:**223-242. doi: 10.1007/s11222-006-9438-0.

[18] H. Haario, M. Laine, A. Mira, and E. Saksman. 2006. DRAM: Efficient adaptive MCMC. *Stat. and Comput.* **16:**339-354. doi: 10.1007/s11222-006-9438-0.

[19] K. Hanada, K. Kumagai, S. Yasuda, Y. Miura, M. Kawano, M. Fukasawa, and M. Nishijima. 2003. Molecular machinery for non-vesicular trafficking of ceramide. *Nature.* **426:**803-809.

[20] K. Hanada. 2006. Discovery of the molecular machinery CERT for endoplasmic reticulum-to-Golgi trafficking of ceramide. *Molecular and Cellular Biochemistry.* **286(1-2):**23-31.

[21] K. Hanada, K. Kumagai, N. Tomishige, and M. Kawano. 2007. CERT and intracellular trafficking of ceramide. *BBA - Molecular and Cell Biology of Lipids.* **1771(6):**644-653.

[22] K. Hanada, K. Kumagai, N. Tomishige, and T. Yamaji. 2009. CERT-mediated trafficking of ceramide. *BBA - Molecular and Cell Biology of Lipids.* **1791(7):**684691.

[23] K. Hanada. 2010. Intracellular trafficking of ceramide by ceramide transfer protein. *Proceedings of the Japan Academy, Ser. B, Physical and Biological Sciences.* **86(4):**426437.

[24] W.K. Hastings. 1970. Monte Carlo Sampling Methods Using Markov Chains and Their Applications. *Biometrika.* **57(1)**:97-109.

[25] K. Huitema, J. van den Dikkenberg, J. FHM Brouwers, and J. CM Holthuis. 2004. Identification of a family of animal sphingomyelin synthases. *The EMBO J.* **23:**33-44.

[26] E. Ikonen, and S. Vaini. 2005. Lipid microdomains and insulin resistance: is there a connection?. *Sci. STKE.* **268:**1-3.

[27] G. Karp. 2005. Molekulare Zellbiologie. Springer.

[28] Z. Li, T.K. Hailemariam, H. Zhou, Y. Li, D.C. Duckworth, D.A. Peake, Y. Zhang, M.-S. Kuo, G. Cao, and X.-C. Jiang. 2007. Inhibition of sphingomyelin synthase (SMS) affects intracellular sphingomyelin accumulation and plasma membrane lipid organization. *Biochim. Biophys. Acta.* **1771(9):**11861194.

[29] E. Limpert, W.A. Stahel, and M. Abbt. 2001. Log-normal Distributions across the Sciences: Keys and Clues. *BioScience.* **51(5):**341-352.

[30] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, and E. Teller. 1953. Equations of State Calculations by Fast Computing Machines. *J. Chem. Phys.* **21(6):**1087-1092.

[31] P. Müller. Monte Carlo Methods and Bayesian Computation: MCMC. Available at `http://www.math.utexas.edu/users/pmueller/class/422/mcmc-tutorial.pdf`

[32] J.D. Murray. 2002. Mathematical Biology - I. An introduction. *Interdisciplinary Applied Mathematics.* Vol. 17. Springer. 3rd edition.

[33] E. Sarri, A. Sicart, F. Lzaro-Diguez, and G. Egea. 2011. Phospholipid Synthesis Participates in the Regulation of Diacylglycerol Required for Membrane Trafficking at the Golgi Complex. *J. Biol. Chem.* **286:**2863228643.

[34] H. Schmidt, and M. Jirstrand. 2006. Systems Biology Toolbox for MATLAB: a computational platform for research in systems biology. *Bioinf.* **22(4):**514-515. doi: 10.1093/bioinformatics/bti799.

[35] G.A.F. Seber, and C.J. Wild. 1989. Nonlinear regression. John Wiley & Sons, Inc.

[36] K. Simons, and E. Ikonen. 1997. Functional rafts in cell membranes. *Nature.* **387:**569-572.

[37] M. Subathra, A. Qureshi, and C. Luberto. 2011. Sphingomyelin Synthases Regulate Protein Trafficking and Secretion. *PLoS ONE.* **6(9)**:e23644.

[38] F.G. Tafesse, P. Ternes, and J. CM Holthuis. 2006. The Multigenic sphingomyelin synthase family. *J. Biol. Chem.* **281(40):**29421-29425.

[39] F.G. Tafesse, K. Huitema, M. Hermansson, S. van der Poel, J. van den Dikkenberg, A. Uphoff, P. Somerharju, and J.C.M. Holthuis. 2007. Both sphingomyelin synthases SMS1 and SMS2 are required for sphingomyelin homeostasis and growth in human HeLa cells. *J. Biol. Chem.* **282(24):**17537-17547.

[40] L. Tierney, and A. Mira. 1999. Some adaptive Monte Carlo methods for Bayesian inference. *Statistics in Medicine*. **18:**1507-2515.

[41] C. Thomaseth, P. Weber, T. Hamm, K. Kashima, and N. Radde. 2012. Modeling SMS driven conversion of ceramide to sphingomyelin reveals the existence of a positive feedback mechanism. Submitted to *Journal of Theoretical Biology*.

[42] M. Villani, M. Subathra, Y.-B. Im, Y. Choi, P. Signorelli, M. Del Poeta, and C. Luberto. 2008. Sphingomyelin synthases regulate production of diacylglycerol at the Golgi. *Biochem. J.* **414:**31-41.

[43] D. Wilkinson. 2006. Stochastic Modelling for Systems Biology. *Mathematical and Computational Biology series*: Volume 11. Chapman & Hall/CRC. 1st edition.

[44] C. Yeang, T. Ding, W.J. Chirico, and X.-C. Jiang. 2011. Subcellular Targeting Domains of Sphingomyelin Synthase 1 and 2. *Nutrition & Metabolism.* **8:**89. doi: 10.1186/1743-7075-8-89