Model Analysis and Parameter Estimation in Biochemical Reaction Networks

Thesis by:

Attila Gábor

supervisor: **Prof. Katalin M. Hangos**
co-supervisor: **Prof. Julio R. Banga**

DOI:10.18136/PE.2017.639

Thesis submitted to
University of Pannonia

in partial fulfilment of the requirements for the degree of

Doctor of Philosophy
in Computer Science

Doctoral School of Information Sciences and Technology
H-8200, Egyetem utca 10.
Veszprém, Hungary

January, 2017

Model Analysis and Parameter Estimation in Biochemical Reaction Networks

Értekezés doktori (PhD) fokozat elnyerése érdekében.

Írta: Gábor Attila

Készült a Pannon Egyetem Informatikai Tudományok Doktori Iskolája keretében.

Témavezető: Prof. Hangos Katalin
Elfogadásra javaslom (igen / nem)

(aláírás)

A jelölt a doktori szigorlaton ..... %-ot ért el

Veszprém .....

a Szigorlati Bizottság elnöke

Az értekezést bírálóként elfogadásra javaslom:

Bíráló neve: ..... (igen / nem)

(aláírás)

Bíráló neve: ..... (igen / nem)

(aláírás)

A jelölt az értekezés nyílvános vitáján .....%-ot ért el

Veszprém .....

a Bíráló Bizottság elnöke

A doktori (PhD) oklevél minősítése .....

.....
Az EDT elnöke

Acknowledgements

First of all, I would like to express my gratitude to Prof. Katalin M. Hangos and to Prof. Julio R. Banga. I would like to thank for Prof. Hangos for hosting me during my MSc and in the first year of the PhD at the MTA-SZTAKI, for her guidance and continuous help since my undergraduate studies to the very end of my PhD. I also would like to thank to Prof. Banga for his continuous support and for the excellent work conditions in his research lab in the last three years of my PhD at the IIM-CSIC in Vigo.

I also would like to thank to Prof. Gábor Szederkényi for his continuous help in our joint work and for Dr. Eva Balsa-Canto and Dr. Ralf Hannemann-Tamás for our inspiring discussions.

Further, I am thankful for the friendly and inspiring environment for the members of the Process Control Research Group at MTA SZTAKI (Zoltan Tuza, Janos Rudan), for the members of the Bio-Process Engineering group at IIM-CSIC (David Henriques, Alex, Irene, David Penas, Ana, Gundian, Miriam, Carlos, Luis, Manuel) and for Gabri, Ledi, Giulia, Jose and Maria.

Thanks for my parents and Szimi for their patience and continuous support.

I also acknowledge the generous founding from the EU FP7 projects "NICHE", ITN Grant number: 289384.

Abstract

Kinetic models are central in systems biology to describe and analyse metabolic, generic and signalling networks. Kinetic models provide a way to summarize and precisely formulate the current knowledge about the dynamics of biological systems in terms of differential equations. Therefore computational tools for the analysis and calibration of these type of models are of great interest.

In this thesis, first the concepts of chemical reaction network theory are extended for biochemical reactions. A complex-reaction graph is defined for the network, in which the nodes are complexes and the edges represent reactions with multiple kinetics. Then, it is shown that the system of dynamic equations of the bio-CRNs can be formulated such that it has a close relationship to this reaction graph. Further, an algorithm is presented to find a network (a realization) to a given kinetic equation system.

The proposed form of the model equations let us formulate optimization problems to find dynamically equivalent realizations, i.e. multiple networks which can be described by the same kinetic equations. Further, it is shown by the linear conjugacy theorem, that if the scaling of the state variables is allowed, structurally different further realizations of the same kinetic system can be found.

In the third part of this work a model reduction method is proposed for large scale kinetic networks. The original mixed integer nonlinear optimization problem is approximated by a finite sequence of mixed integer quadratic optimization problems, which is much cheaper to solve by existing methods. The reduction method sequentially eliminates reactions from the network, such that the trajectories of some important species do not change, i.e. the reduction error in each step is minimized. Further, the kinetic rate parameters are simultaneously tuned in given bounds to guarantee the best fit between the original and the reduced model.

In the last part of the thesis the calibration of kinetic models to experimental data is considered. Here a global optimization method is proposed together with regularization techniques. We illustrate by seven case studies of increasing complexity, how the presented method overcome the non-convex nature of these calibration problems and results in faster and more reliable convergence than traditional alternatives. Further, the calibrated models are evaluated by out of sample cross-validation, showing that the regularized estimations have better predictive value.

Abstracto

Los modelos cinéticos son centrales en la biología de sistemas para describir y analizar redes metabólicas, redes de señalización y redes genéricas. Los modelos cinéticos proveen una forma de resumir y formular de manera precisa, el actual conocimiento sobre la dinámica de los sistemas biológicos en términos de ecuaciones diferenciales. Por lo tanto, herramientas computacionales para el análisis y calibración de este tipo de modelos son de gran interés.
En esta tesis, primero, los conceptos de la teoría de redes de reacciones químicas son extendidos para reacciones bioquímicas. Un grafo complejo-reacción es definido para la red, donde los nodos representan complejos y las aristas representan reacciones con cinéticas múltiples. Luego, es mostrado que el sistema de ecuaciones dinámicas de las redes de reacciones bioquímicas (bio-CRNs) puede ser formulado de tal forma que tenga una relación cercana a esta grafo de reacción. Mas adelante, un algoritmo es presentado para encontrar una red (una realización) dado un sistema de ecuaciones cinéticas.
La forma propuesta de las ecuaciones del modelo nos permiten formular problemas de optimización para encontrar realizaciones dinámicas equivalentes, p. ej. múltiples redes que pudieran ser descritas pot las mismas ecuaciones cinéticas. Además, es mostrado, por el teorema de conjugación lineal , que si el escalamiento de las variables de estado es permitido, posteriores realizaciones de diferentes estructuras del mismo sistema cinético pueden ser encontradas.
En la tercera parte de este trabajo, un método de reducción de modelos es propuesto para redes cinéticas de gran escala. El problema original de optimización no lineal mixto de enteros, es aproximado con una secuencia finita de problemas de optimización cuadráticos mixtos de enteros. El método de reducción elimina secuencialmente reacciones de la red, de tal forma que las trayectorias de algunas especies importantes no cambien, p. ej. en cada paso el error de reducción es minimizado. Además, los parámetros de las tasas cinéticas son simultáneamente ajustados a ciertos límites.
En la parte final de la tesis la calibración de modelos cinéticos a datos experimentales es considerada. Aquí, un método de optimización global es propuesto junto con técnicas de regularización. Ilustramos con 7 casos de estudio de gran complejidad, cómo es que el método presentado se sobrepone a la naturaleza no convexa de esstos problemas de calibración y resulta en una convergencia más rápida y confiable que las de alternativas tradicionales. Además, los modelos calibrados son evaluados por evaluación cruzada dejando uno fuera, mostrando que las estimaciones regularizadas tienen un mejor valor predictivo.

Absztrakt

A kinetikai modellek központi szerepet töltenek be a rendszer-biológiában, ahol például metabólikus hálózatok vagy jelátviteli hálózatok leírására alkalmazzák őket. Ezen hálózatokhoz közönséges differenciál egyenleteket rendelhetünk, melynek analízise áll a dolgozat középpontjában.

A dolgozat első felében a kémiai reakcióhálózatok elméletét terjesztem ki biokémiai reakciókat tartalmazó hálózatokra. Értelmezem a komplex-reakció gráfot, majd a hálózat dinamikai egyenleteit olyan formában fogalmazom meg, mely jól tükrözi a hálózat struktúráját. Ennek segítségével egy algoritmust fejlesztek, amely kinetikai egyenletekből kiindulva, az egyenletekkel ekvivalens hálózatot realizál.

Az egyenletek alakja lehetővé teszi, hogy optimalizációs eszközökkel dinamikusan ekvivalens halózatokat keressünk. Ezek olyan struktúrálisan, gráfjukban különböző hálózatok, melyeket ugyanazon dinamikai egyenletekkel írhatunk le. Optimalizációs problémaként fogalmazom meg, hogyan lehet sűrű, ritka és reverzibilis reakciógráfot találni egy adott dinamikához. Továbbá bevezetem a dinamikusan konjugált hálózatok fogalmát, mely dinamikában hasonló, de struktúrában változatosabb hálózatok keresését teszi lehetővé.

A dolgozat harmadik részében kinetikai hálózatmodellek egyszerűsítésére dolgozok ki optimalizációs módszert. Az algoritmus a kiindulási hálózat reakcióinak számát lépésenként csökkenti, oly módon, hogy bizonyos választott komponensek koncentráció-trajektóriái közel maradjanak az eredeti hálózatbeli trajektóriáikhoz. Ez a feladat kevert típusú, nemlineáris optimalizációs problémaként fogalmazható meg, melyet egy véges számú lépést tartalmazó kevert, kvadratikus optimalizációs problémával közelítek meg. Ez a közelítés lehetővé teszi nagy hálózatok egyszerűsítését a számítási igény alacsonyan tartása mellett.

A dolgozat ötödik fejezetében biológiai modellek paramétereinek becslésére dolgozok ki eljárást. Dinamikus modellek esetén a paraméterbecslés egy nem-konvex, nem-lineáris és gyakran rosszul kondícionált optimalizációs probléma megfogalmazására vezet. Ennek megoldására egyrészt egy meglévő globális optimalizációs algoritmus továbbfejlesztését javaslom, továbbá regularizációs technikákat vizsgálok meg a kondicionáltság javítására. Hét esettanulmányon mutatom be a kidolgozott módszer működését. A vizsgált esetekben a választott optimalizáló algoritmus gyorsabban és megbízhatóbban konvergál a globális minimumhoz, mint a gyakorlatban használt más algoritmusok. Továbbá a regularizacióval kapott kalibrált modellek jobban teljesítenek a modellek validációja során.

Contents

List of Figures

List of Tables

List of Notations

Acronyms

AMS	Advanced Multi-Start
BBG	Biomass Batch Growth Model
bio-CRN	Biochemical reaction network
BP	Balancing Principle
DMP	Modified Discrepancy Principle
DP	Discrepancy Principle
eSS	enhanced Scatter Search
FHN	FitzHugh-Nagumo Model
FIM	Fisher Information Matrix
GCV	Generalized Cross Validation
GOsc	Goodwin's Oscillator Model
HBP	Hardened Balancing Principle
LASSO	Least Absolute Shrinkage and Selection Operator
LCC	L-curve method based on maximal curvature detection
LOO	Leave One Out cross validation
MAL-CRN	Chemical reaction network obeying the mass action law
MER	Monotone Error Rule
MINLP	Mixed Integer Non-Linear Program
MIQP	Mixed Integer Quadratic Program
NLS	Nonlinear Least Squares
ODE	Ordinary Differential Equation
ODE	Ordinary Differential Equation
QO	Quasi Optimality
RGCV	Robust Generalized Cross Validation
SMS	Simple Multi-Start
SRGCV	Strong Robust Generalized Cross Validation
TMP	Transformed Discrepancy Principle
TSMP	Three-step Metabolic Pathway Model

Greek Symbols

η_i	stoichiometric composition of the i-th complex, equivalent to the i-th column of Y
ν	Stoichiometric coefficient
$\Phi(\cdot,\cdot)$	Least squares functional

Ψ	Monomial function
$\rho^{(i,j)}$	Reaction vector corresponding to $(C_i, C_j) \in \mathcal{R}$
φ	Vector of kinetics

Roman Symbols

ℓ	Number of connected components in the graph (linkage classes)
$\tilde{A}_k$	Kirchhoff matrix
A_{k}	Generalized Kirchhoff matrix
C	Chemical complex
D	Denominator function of the reaction kinetics
g	Reaction kinetics function
I_p	set of index pairs of the non-negative elements of A_k
$I_{n\times n}$	N dimensional identity matrix
k	Principal reaction rate coefficient
l	number of connected components in the graph (linkage classes)
M	coefficient matrix
$n_{\mathcal{I}}$	Number of important species in model reduction
P	Kinetic weighting function
r	Reaction rate function
x	Concentration vector, state vector
X_i	Chemical species
Y	Complex composition matrix
d	deficiency of the network
m	Number of complexes
N	Stoichiometric matrix
n	Number of species
S	stoichiometric subspace

Other Symbols

$\mathbb{R}$	Real number
$\mathbb{R}_{>0}$	Positive real number
$\mathbb{R}_{\geq 0}$	Non-negative real number
$\mathcal{G}_i$	Biochemical kinetics in complex C_i
$\mathcal{I}$	Set of indices of important species in model reduction

Chapter 1

Introduction

1.1 Motivation

Mathematical modelling is the central element in quantitative approaches to molecular and cell biology. The possible uses of quantitative modelling of cellular processes go far beyond explanatory and predictive studies [1, 2]. They provide a way to understand complex biosystems [3, 4] and have given rise to systems biology as a new way of thinking in biological research [5]. Models in systems biology vary in their degree of network complexity and accuracy of representation [6]. Dynamic (i.e. kinetic) models offer the greatest degree of flexibility and accuracy to explain how physiological properties arise from the underlying complex biochemical phenomena.

The use of kinetic models to understand the function of biological systems has already been successfully illustrated in many biological systems, including signalling, metabolic and genetic regulatory networks [7–15]. Further, dynamic model-based approaches have also been used to identify possible ways of intervention or (re-)design, such as in optimal experimental design [16–20], metabolic engineering [21] and synthetic biology [22, 23]. Other recent efforts have been focused on scaling-up, i.e. on the development and exploitation of large-scale (genome-scale) kinetic models [24], and ultimately, whole-cell models [25, 26].

Although nonlinear dynamical models have become the most common approach in systems biology, they have received relatively little attention in the statistical literature, especially when compared with other model types [27]. As a consequence, the area can be regarded as one of the most fertile fields for modern statistics [28]: it offers many opportunities, but also many important challenges [29].

Many computational methods have been developed for chemical reaction networks (CRNs), which are described by polynomial ordinary differential equations (ODEs). The general goal of this work is to further generalize these modeling, network realization, model analysis, complexity reduction and parameter estimation methods for biochemical reaction networks. This model class is more frequently used in biochemical research, however it is more complex than CRNs, since it allows a more complicated dynamics that is described by ODEs with rational function right hand side.

1.2 Chemical reaction networks

The fundamentals of chemical reaction network theory (CRNT) were established back in the 1970s by [30] and [31]. This theory aims to consider complex reaction systems from a general perspective, linking the structure of the reaction network with its possible dynamics (such as unstable and/or multiple steady states, oscillations or chaotic behaviour). CRNT uses powerful descriptors, like the reaction graph and its properties.

Later, the notion of dynamic equivalence has appeared and it became apparent that possibly several equivalent reaction kinetic schemes, i.e. reaction network realizations can be constructed to a given dynamic ODE model by using optimization, see e.g. [32, 33]. These realizations offer the possibility to analyse or ensure advantageous dynamic property to a CRN if a suitable realization with desirable structural properties (e.g. zero deficiency and weak reversibility) is found.

It is worth highlighting that although Horn and Jackson [30] already mentioned complex reactions in biological systems as one of the general situations they wanted to embrace with their theory, its applications in biology did not appear until Bailey [34] rescued it, highlighting its potential for the analysis of biochemical networks without calibrating the model with experimental data ("complex biology with no parameters"). In other words, CRNT can be used to characterize kinetic models (multi-stability, oscillations, etc.) without knowing the precise values of the kinetic parameters.

Several important applications regarding the characterization of the dynamics of biochemical reaction networks have appeared since then, including the works of e.g. [35–43].

Furthermore, the theory has also been applied to the identification, i.e. parameter and model structure estimation, of biological systems. For example, Craciun and Pantea [44] used CRNT to show that, given a (mass action) reaction network and its dynamic equations, it might be impossible to identify its rate constants uniquely (even with perfect measurements of all species). Furthermore, they also concluded that, given the dynamics, it might be impossible to identify the reaction network uniquely.

In [32, 33] and [45], CRNT principles were used to pinpoint inherent limitations in the inference of biological networks. These works show that, in addition to the obstacles identified by [46] (lack of data and deficiencies in the inference algorithms), there are fundamental problems related to the uniqueness and distinguishability of these networks. Further, these problems are present even for the utopian case of fully observed networks with noiseless measurements.

Despite of the above mentioned works in the area of biochemical reaction networks (bio-CRNs), no systematic attempt has been made to construct and analyse the structure of bio-CRNs, to characterize their canonical structure as a subset of positive rational ODEs [47], and to link their structural properties to the dynamic properties of the underlying biochemical system.

Therefore, our general aim in Chapter 2 is to extend the well-known formalism of the chemical reaction networks obeying the mass action law (MAL-CRN) for general biochemical networks where the reaction rate functions often account for more complex mechanism than the simple mass action law, such as the Michaelis-Menten kinetics, Hill kinetics, activating or inhibiting mechanisms. A canonical decomposition of these reaction rate functions being in rational function form lets us define the main structural elements of the reaction network and the complex – reaction graph. The MAL-CRN case then becomes a special case of this biochemical reaction network.

This way the basic structural (i.e. parameter-independent) properties of a bio-CRN structure are easy to define and understand, that include reversibility, weak reversibility, and deficiency. The ordinary differential equations describing the dynamics of the biochem-

ical reaction network (i.e. the concentrations of the species in the network) can be stated in a form that is similar to the MAL-CRN formalism. This new form of the equations let us easily analyse certain properties of the network, for example to prove the non-negativity of the solutions. Furthermore, the proposed structure allows to algorithmically infer a network from the ODEs, and determine alternative bio-CRN structures that are dynamically equivalent to the original network.

1.2.1 Introduction to MAL-CRNs

Here we shortly introduce the notions and important results on mass action chemical reaction networks (MAL-CRNs). The classical theory of reaction networks assumes a closed thermodynamic system with constant physico-chemical properties under isothermal and isobaric conditions. The applicability of this model class is surprisingly wide: besides the description of purely chemical mechanisms, CRNs can be effectively used to model processes of living (i.e. cell) environments [48], compartmental models [49] or general nonnegative systems with possible application domains completely outside of (bio)chemistry [50, 51].

The chemical species X_i, $i = 1, \dots n$ participate in chemical reactions of the form

$$\sum_{k=1}^{n} \nu_{ki} X_k \xrightarrow{k_{ij}} \sum_{k=1}^{n} \mu_{kj} X_k \ . \tag{1.1}$$

Here the *stoichiometric coefficients* of the *reactants* are denoted by $\nu_{1i}, \dots, \nu_{ni}$ and of the *products* by $\mu_{1j}, \dots, \mu_{nj}$. These coefficients are integer values and quantify the amount of molecules consumed and produced in a reaction step. The non-negative integer linear combinations of the species, for example $C_i = \sum_{k=1}^{n} \nu_{ki} X_k$ and $C_j = \sum_{k=1}^{n} \mu_{kj} X_k$ in the two sides of the reactions (1.1), are called the *complexes* and are denoted by $C_1, \dots C_m$. The coefficients of the complexes are stored in the complex composition matrix $Y \in \mathbb{R}^{n \times m}$ such that $[Y]_{ij} = \nu_{ij}$, i.e. the i-th column of the matrix stores the stoichiometric coefficients in the i-th complex C_i, for $i = 1 \dots m$. To simplify the notation, we often use η_i to refer to the i-th column of Y.

As the stoichiometric coefficients are non-negative, it may happen that for a certain complex C_0 all of them are equal to zero, i.e. the corresponding column in Y is the zero vector. Such a complex C_0 is called the *zero complex*, and it can be used to describe the case when the system is not closed but has in-/outflow from/to the environment, as it is a usual situation in biochemical models.

Based on a list of reactions one is interested in the calculation of the concentrations of the species X_i, which are denoted by x_i for $i = 1, \dots n$. Then the variable x is the n-dimensional, non-negative state vector.

Traditionally the reactions are explained by the picture of molecular collisions, which gives rise to the *mass action law*. According to the mass action law, the intensity of the reaction is proportional to the concentrations of the molecules participating in the reaction step. Formally, this means that the reaction rate function is defined for the reaction $C_i \xrightarrow{k_{ij}} C_j$ as

$$r_{ij}(x) = k_{ij} \cdot \prod_{l=1}^{n} x_l^{\nu_{li}} \ . \tag{1.2}$$

The reaction rate function $r_{ij}(\cdot) : \mathbb{R}^n_{\geq 0} \mapsto \mathbb{R}_{\geq 0}$ is usually measured in units $\left[\frac{mol}{s}\right]$ and shows how many moles of a reactant X_k with stoichiometry $\nu_{ki} = 1$ is used, or how many moles of a product X_l with $\mu_{lj} = 1$ is produced by the reaction in one second.

Based on the above notions the MAL-CRN is uniquely described by the following sets

I. $\mathcal{S} = \{X_1, \ldots, X_n\}$ is the set of species or chemical substances.

II. $\mathcal{C} = \{C_1, \ldots, C_m\}$ is the set of complexes.

III. The set of chemical reactions is

$$\mathcal{R} = \{(C_i, C_j) \mid C_i, C_j \in \mathcal{C}, \text{ and } C_i \text{ is transformed to } C_j\}.$$

The above three sets determine the mass action system, thus we introduce the sort notation for a MAL-CRN: $\Sigma_{\text{MAL}} = (\mathcal{S}, \mathcal{C}, \mathcal{R})$.

Reaction vectors and stoichiometric subspace. The reaction vectors $\rho^{(i,j)} = \eta_j - \eta_i$ are defined for each reaction $(C_i, C_j) \in \mathcal{R}$. They show the net production of the species in the corresponding reaction steps. When arranged in a matrix as columns, these reaction vectors form the stoichiometric subspace of the reaction network:

$$S = \text{span}\{\rho^{(i,j)} = \eta_j - \eta_i \mid (C_i, C_j) \in \mathcal{R}\} \ . \tag{1.3}$$

The rank of S is often referred as the rank of the stoichiometric subspace and denoted by s, i.e. $s = \text{rank}(S)$.

Reaction graph of MAL-CRNs

A directed, weighted reaction graph [52] is constructed from the set of complexes and reactions. The reaction graph $\mathbb{G} = (\mathbb{V}, \mathbb{E})$ consists of a finite non-empty set of nodes, which represents the complexes: $\mathbb{V} = \{C_1, \ldots C_m\}$ and a finite set of directed edges $\mathbb{E}$, which represents the reactions: $\mathbb{E} = \{(C_i, C_j) \mid (C_i, C_j) \in \mathcal{R}, i, j = 1 \ldots m\}$. The reaction rate coefficients are assigned to the edges as weights.

Reversibility. A reaction graph is called *reversible*, if whenever the reaction $C_i \xrightarrow{k_{ij}} C_j$ exists, then a reverse reaction $C_j \xrightarrow{k_{ji}} C_i$ is also present. A reaction graph is called *weakly reversible*, if whenever complex C_j is reachable from complex C_i on a directed path, then there is a directed path from C_j to C_i, too. When the reaction graph is (weakly) reversible, then the reaction network is also called so.

Linkage classes. A set of connected complexes is called a linkage class of the network, if the complexes in the set are connected to each other, but there is no connection to complex outside of the set. The number of linkage classes of the network is denoted by l.

Deficiency. The *deficiency* d of a reaction network [53] is defined as:

$$d = m - l - s \tag{1.4}$$

where m is the number of complexes, l is the number of linkage classes and s is the rank of the stoichiometric subspace.

Dynamic equations of MAL-CRNs

Following [53], the dynamic equations of the MAL-CRNs can be written as

$$\frac{dx}{dt} = Y \tilde{A}_k \Psi(x) \ , \tag{1.5}$$

where Y is the complex composition matrix, $\tilde{A}_k \in \mathbb{R}^{m\times m}$ is a column conservation matrix– also known as the Kirchhoff matrix–, which contains the reaction rate coefficients as

$$[\tilde{A}_k]_{ij} = \begin{cases} k_{ji} & \text{if } i \neq j \\ \sum_{l=1, l\neq i}^{n} -k_{il} & \text{if } i = j \end{cases} , \tag{1.6}$$

and $\Psi : \mathbb{R}^n \to \mathbb{R}^m$ is a monomial vector function defined as

$$[\Psi(x)]_i = \prod_{l=1}^{n} x_l^{[Y]_{li}} \quad \text{for} \quad i = 1 \dots m \ . \tag{1.7}$$

As we can see from the above, the dynamic equations of mass action law systems consist of polynomial functions. When a system of ODEs can be written in the form (1.5), it is called a *kinetic polynomial system*.

Complex balanced network Whenever for the steady state solution x^* of (1.5) fulfils the condition

$$\tilde{A}_k \Psi(x^*) = 0, \tag{1.8}$$

the corresponding MAL-CRN is called complex balanced realization at this steady state.

Detailed balanced network The network is called detailed balanced at the steady state x^*, if (i) the network is reversible and (ii) the reversible reaction rates are equal in the steady state, i.e.

$$(C_i, C_j) \in \mathcal{R} \Leftrightarrow (C_j, C_i) \in \mathcal{R} \quad \text{and} \quad r_{ij}(x^*) = r_{ji}(x^*) \quad \text{for} \quad i, j = 1 \dots, m \tag{1.9}$$

1.2.2 Properties of MAL-CRNs

Many results have been obtained about MAL-CRNs based on the above formalism, for which a good review can be found in [51] and [54]. Here we briefly highlight some important results, which illustrates the power of the above framework for the analysis of nonlinear dynamical systems.

Non-negativity of the solution All the solutions of (1.5) corresponding to non-negative initial conditions remain non-negative independently on the reaction rate parameters.

Properties of detailed and complex balanced steady states [54] If a CRN is complex balanced (detailed balanced) at any positive x^* then it is complex balanced (detailed balanced) at all other positive equilibrium points. Further, if a CRN is complex balanced then it is weakly reversible.

Finally, a CRN is complex balanced for any positive values of reaction rate coefficients if and only if it is weakly reversible, and the deficiency of the system is zero.

Stability of the solution Important stability results about the solutions of the MAL-CRNs were formulated in [53, Theorem 6.1.1], also known as the *the zero deficiency theorem*.

Theorem 1.2.1. *For zero deficiency MAL-CRNs,*

(i) if the network is not weakly reversible, then the dynamics (1.5) *cannot admit a positive steady state.*

(ii) *if the network is not weakly reversible, then the dynamics* (1.5) *cannot admit a strictly positive cyclic composition trajectory.*

(iii) *if the network is weakly reversible, then for each stoichiometric compatibility class, the dynamics* (1.5) *has exactly one positive steady state, which is asymptotically stable.*

Example 1-1 (Simple mass action chemical reaction network)

Let us consider the simple MAL-CRN in Figure 1.1. This reaction network contains four chemical species $\{X_1,\ X_2,\ X_3,\ X_4\}$ in three complexes ($C_1 = X_1 + 2X_2,\ C_2 = X_1 + X_3,\ C_3 = X_4$). The nodes of the reaction graph representing the complexes are connected by five directed, which represents the reactions. Each edge is weighted by the reaction rate coefficient of the corresponding reaction.

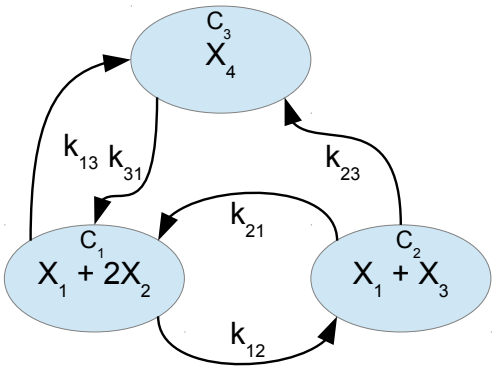


Figure 1.1: Reaction graph of a simple MAL-CRN

The matrices and the non-linear vector function that characterizes the realization is as follows

$$Y = \begin{bmatrix} 1 & 1 & 0 \\ 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad , \quad A_{\mathrm{k}} = \begin{bmatrix} -(k_{12}+k_{13}) & k_{21} & k_{31} \\ k_{12} & -(k_{21}+k_{23}) & 0 \\ k_{13} & k_{23} & -k_{31} \end{bmatrix} \quad , \quad \Psi(x) = \begin{bmatrix} x_1 x_2^2 \\ x_1 x_3 \\ x_4 \end{bmatrix}.$$

The above elements – based on (1.5) – define the following differential equation model

$$\dot{x}_1 = -k_{13}x_1x_2^2 - k_{23}x_1x_3 \tag{1.10}$$

$$\dot{x}_2 = -2k_{13}x_1x_2^2 - 2k_{12}x_1x_2^2 + 2k_{21}x_1x_3 + 2k_{31}x_4 \tag{1.11}$$

$$\dot{x}_3 = -k_{23}x_1x_3 - k_{21}x_1x_3 + k_{12}x_1x_2^2 \tag{1.12}$$

$$\dot{x}_4 = -k_{31}x_4 + k_{13}x_1x_2^2 + k_{23}x_1x_3 \ . \tag{1.13}$$

It is easy to see, that the realization is weakly reversible and there is one linkage class. The stoichiometric subspace is spanned by the following reaction vectors:

$$s_1 = \begin{pmatrix} 1 \\ 2 \\ 0 \\ 0 \end{pmatrix} - \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 2 \\ -1 \\ 0 \end{pmatrix}, s_2 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 4 \end{pmatrix} - \begin{pmatrix} 1 \\ 2 \\ 0 \\ 0 \end{pmatrix} = \begin{pmatrix} -1 \\ -2 \\ 0 \\ 4 \end{pmatrix}, s_3 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 4 \end{pmatrix} - \begin{pmatrix} 1 \\ 0 \\ 1 \\ 0 \end{pmatrix} = \begin{pmatrix} -1 \\ 0 \\ -1 \\ 4 \end{pmatrix}$$

Therefore the rank of the stoichiometric subspace is

$$s = \operatorname{rank}\left[\begin{pmatrix} 0 & -1 & -1 \\ 2 & -2 & 0 \\ -1 & 0 & -1 \\ 0 & 4 & 4 \end{pmatrix}\right] = 2,$$

and the deficiency is $d = 3 - 1 - 2 = 0$. Therefore this network is a deficiency zero, weakly reversible network and according to Theorem 1.2.1 there is precisely one, globally stable, positive steady state solution of the equations (1.10)-(1.13). ■

1.3 Problem statements and goals

The previous section summarizes the notions of mass action chemical reaction networks and introduces the basic principles. In the first part of the thesis, we extend the above MAL-CRN formalism for biochemical networks, then we develop computational methods to solve realization, model verification, model reduction and parameter estimation problems for this class of dynamic models. In the following part of this section we shortly summarize the above mentioned computational problems. A more detailed discussions of each problem can be found in the introductions of the corresponding chapters of the thesis.

Formulation of biochemical reaction networks The mathematical models that describe biochemical systems are often very similar to the equations of mass action chemical reaction networks. In both cases, the model equations account for the change of a compound in a unit time due to reactions that consume and/or produce the chemical species. Therefore we can expect to represent chemical and biochemical networks in a common framework.

The power of chemical reaction network theory is to analyze complex nonlinear dynamic systems based on the corresponding reaction graph and its structural properties. Thus it is desirable to (1) define similar, precise notions for biochemical reactions networks, for example what is a complex and how to handle different reaction kinetics, and (2) to find the (mathematical) properties of the equations that represent biochemical networks. These studies are presented in the first part of the thesis.

Verification of biochemical models Model verification is a process in which the model is tested against certain, predefined requirements. Verification tools often accompany modeling toolboxes, such as COPASI [55], however, these are often restricted to syntactical checking of the models. Further, for example the Biomodels database [56] requires that the uploaded models are connected to published, reproducible results. After a model is submitted, it is tested whether the simulation results reproduce the published curves or not.

Instead of synthactical or qualitative checking, here we will focus on checking the models from the physico-chemical point of view. For example, the non-negativity of the solutions of the dynamic equations is expected in case these equations represent chemical concentrations. Using the notions of biochemical networks, we derive a set of procedures which guarantee that (1) solutions of the the mathematical model are non-negative, (2) the reactions are plausible chemical reactions and (3) the mass conservation is fulfilled.

Network realization. The canonical realization algorithm [57] finds a network representation of kinetic equations. It turns out that not only chemical systems, but other positive systems, for example electrical circuit models, can be transformed to a MAL-CRN. The condition, that a network can be associated to an ordinary differential equation model is

closely connected to the essential non-negative property of the equations. We show, that essential non-negativity is also a property of biochemical models, which leads to the goal, to generalize the algorithm by Hárs and co-authors for biochemical network inference, i.e. to find biochemical network realizations of equations, which have rational polynomial right hand side functions.

It is known from chemical reaction networks, that multiple network give rise to the same dynamics. These networks are dynamically equivalent realizations. Writing the biochemical equations in a similar form that of MAL-CRNs allow us to formulate linear optimization problems to find dynamically equivalent networks.

Linearly conjugated networks. Johnston and Siegel [58] showed that the linear scaling of the state variables in CRNs generalizes the dynamic equivalence and preserves the stability properties of the dynamics. This linear conjugacy theorem allowed them to find network realizations for systems, which are structurally different from all known dynamic equivalent realizations. We will develop a similar theorem and computational tool for biochemical reaction networks.

Model complexity reduction. The mathematical models of reaction kinetic systems are most often too large and detailed for dynamic analysis or parameter estimation purposes as they are usually constructed based on detailed kinetic studies. There are a number of extensively studied areas where there are detailed models of chemically reacting systems available. These include biochemical systems, such as signal transduction pathway modeling and reacting flow or catalytic reaction systems. These models are used for both model analysis (stability analysis and the investigation of nonlinear dynamic properties, such as oscillations or chaotic behavior), and for dynamic predictions (simulation). Because of the huge number of species and/or chemical reactions present in the detailed reaction kinetic mechanism of these systems, the need has arisen for developing a simplified or reduced mechanisms that can accurately describe the dynamics of the system.

Here we will propose a method to obtain a simplified model, which preserves the interpretation of the model components. The algorithm presented in the study eliminates reactions from the reaction network, thus it reduces the network complexity in a way that the dynamics of some selected species are not changing.

This general problem can be formulated as a nonlinear mixed integer optimization problem, which is hard to solve in case of many reactions. Using the structure of the chemical reaction networks, we will approximate the above optimization problem, such that the procedure remains computationally affordable for large size networks.

Parameter estimation. One of the main challenges in modeling biochemical processes is the calibration of dynamic models, also known as the parameter estimation problem. Parameter estimation aims to find the unknown, constant model parameters such that the calibrated model gives the best fit to a set of experimental data. Then the calibrated model can be used to predict the behaviour of the biochemical process for unmeasured conditions or to use it for experimental design.

The estimation procedure can be broken down to the following tasks: (1) model simulation, (2) minimization of the cost function via optimization, which optionally requires (3) gradient calculation and (4) analysis of the results. Depending on the particular model, the procedure can be computationally demanding, therefore efficient implementation and tailored methods are needed to calibrate medium and large scale models. Our goal is to develop a pipeline for the calibration of biochemical models, which show good scaling properties and further the procedure can incorporate a priory information about the parameters via regularization.

Structure of the thesis. The thesis consists of 6 chapters. The structure of the thesis is as follows. In **Chapter 2** the biochemical reaction networks are introduced and the model verification of such systems are considered. In **Chapter 3** the realization algorithm is presented, and the dynamically equivalent and linearly conjugated networks are described. The model reduction in kinetic systems is described in **Chapter 4**. **Chapter 5** considers the robust calibration of kinetic models. Finally, **Chapter 6** contains the overall conclusions and summarizes the new scientific results and possible future work.

Chapter 2

Biochemical reaction networks

In this chapter, first a modeling framework for biochemical systems is proposed. In Subsection 1.2 we showed a powerful tool to analyze kinetic polynomial systems by the chemical reaction network theory. Here, we construct a similar framework for biochemical models, which we will call biochemical reaction networks (bio-CRNs).

We show that this modeling framework is able to describe a large set of biochemical systems. Further, mass action systems turn to be a special case of this system class and many important notions previously defined only for mass action systems can be easily extended. In more details, similarly to the chemical reaction network theory considering mass action systems, here one can also associate a complex-reaction graph to the dynamical equations. Then, based on the structure of the reaction graph valuable information can be obtained about the dynamics.

Many of the medium and large-scale kinetic models in systems biology show problems when the space of parameters is explored. For example, dynamic simulations for certain parameter values result in negative concentrations –suggesting that mass-balance may not be correct–, or simply blow-up. Therefore, careful checks should be performed before the use of a published model. This is routinely done in large biochemical model bases (see e.g. [56]), but these checks cannot detect every deficiency that may arise from the many different uses (simulation, parameter estimation, experiment design, etc.) of these models.

There are computational tools (see e.g. [55] or [59]), which help the user to avoid making modeling mistakes, for example, by offering predefined rate-functions, tracking the variables, or supporting measurement units and their consistency. These tools serve mostly for syntactic checking purposes. Furthermore, some tools [60] can also check fundamental model properties, such as mass balance, the existence of admissible steady states, or the characteristics of the dynamic behavior near a steady state, among others.

Despite the above efforts to ensure the acceptable quality of a biochemical model, it is easy to find in the literature such models that do not possess very basic properties, like positivity. This is usually the consequence of model simplification based on assumptions [61]. However these assumptions are sometimes forgotten or not known explicitly.

Therefore, our aim was to formulate simple syntactical and semantic criteria of biochemical origin that ensure the plausibility of the studied model and the positivity (more precisely, non-negativity) of its solution. Similar ideas of model checking appear in [62] and [63].

In the derivation of the verification rules, partially the basic properties of the biochemical reaction networks are used. The definition of the biochemical reaction rate functions helps us to find implausible reaction rate functions in models. Further, as we will see, the plausibility of the reaction rates guarantees the essential non-negativity of the model equa-

tions, which further implies that the model trajectories remain non-negative independently of the model parameters.

In addition to the model verification procedures, we aim at localizing the reaction, or set of reactions, that cause a particular problem (for example, possible negative solutions), and at giving advice on how to correct them.

2.1 Mathematical formulation of biochemical networks

The basic notions of mass action chemical reactions networks (MAL-CRNs) were introduced in Section 1.2.1. The goal here is to extend the notions of MAL-CRNs for a larger systems class, in which the kinetic rate laws are not restricted by the monomial rate expressions of the mass action law, but they are stated as rational functions.

2.1.1 Basic notations

As in the case of MAL-CRNs, here we also assume a closed thermodynamic system with constant physico-chemical properties under isothermal and isobaric conditions. The biochemical species X_i, $i = 1, \dots n$ participate in biochemical reactions of the form

$$\sum_{k=1}^{n} \nu_{ki} X_k \xrightarrow{r_{ijl}} \sum_{k=1}^{n} \mu_{kj} X_k \ , \tag{2.1}$$

where $\nu_{1i}, \dots, \nu_{ni}$ and $\mu_{1j}, \dots, \mu_{nj}$ are stoichiometric coefficients as in the MAL-CRN case. Further, we recall the notions of complexes $C_1, \dots C_m$, complex composition matrix $Y \in \mathbb{R}^{n \times m}$ and the notion of zero complex C_0 from section 1.2.1. The concentrations of the chemical species are also stored in the n-dimensional, non-negative state vector x.

One of the main difference between the bio-CRNs and the MAL-CRNs is the definition of the reaction rate function $r_{ijk}(\cdot) : \overline{\mathbb{R}}_+^n \mapsto \overline{\mathbb{R}}_+$, which is often a non-linear function of the concentration vector. In general it is possible that the species in a complex give rise to multiple reaction paths, thus multiple reaction rate functions can be assigned to each complex. We will categorize these reaction rates based on their mathematical formulations. Using the notations of complexes and kinetics, the reaction (2.1) can be equivalently written as $C_i \xrightarrow{k_{ijl}, g_{il}} C_j$. Here the reaction rate r_{ijl} of the reaction is decomposed as

$$r_{ijl}(x) = k_{ijl} \cdot g_{il}(x), \tag{2.2}$$

where $k_{ijl} \in \mathbb{R}_0^+$ is a constant, non-negative principal reaction rate coefficient and $g_{il}(\cdot)$ is a function of the species concentrations, called the kinetics of the reaction.

Traditionally the reactions are explained by the picture of molecular collisions, which gives rise to the *mass action law*. According to the mass action law, the intensity of the reaction is proportional to the concentrations of the molecules participating in the reaction step. This results in the following form for the mass action kinetics: $g_{il}(x) = \prod_{k=1}^{n} x_k^{\nu_{ki}}$.

However, in biological applications the reaction rates are often not limited to mass action kinetics and therefore the kinetics are not a monomial function of the concentration vector. This is partially a consequence of simplified reaction pathways, for example the Michaelis Menten kinetics ($\frac{x_i}{K+x_i}$) can be derived from a series of enzymatic reactions [64].

Here we assume that for each complex containing some species, a finite set of biochemical kinetics $\mathcal{G}_i = \{G_1, G_2, \dots G_{d_i}\}$ can be determined. Each of these kinetics defines a relationship among the species of the complex, for example $G_1 =$ `'Mass action'`, $G_2 =$

'`Michelis Menten kinetics`', G_3 = '`Hill kinetics`' etc. With this notation, the mathematical function $g_{il}(x)$ is associated with the type of kinetics G_l of the complex C_i.

Based on the above notions we can define the following four sets to uniquely describe a reaction network.

I. $\mathcal{S} = \{X_1, \dots X_n\}$ is the set of species or chemical substances.
II. $\mathcal{C} = \{C_1 \dots C_m\}$ is the set of complexes.
III. $\mathcal{G} = \cup_{i=1}^{m} \mathcal{G}_i$ is the set of reaction rates (kinetics).
IV. The set of biochemical reactions is

$$\mathcal{R} = \{(C_i, C_j, G_l) \mid C_i, C_j \in \mathcal{C}, G_l \in \mathcal{G}_i \text{ and } C_i \text{ is transformed to } C_j \text{ by the kinetics } G_l\}.$$

Since many kinetic biochemical ODE models can be represented in this form, we call these models ***biochemical reaction networks*** (bio-CRNs). The set of species, complexes and reactions with the kinetics uniquely determines the biochemical reaction network which is denoted by $\Sigma = (\mathcal{S}, \mathcal{C}, \mathcal{G}, \mathcal{R})$.

In the special case, when the bio-CRN contains only mass-action type kinetics, also called as mass action law chemical reaction networks (MAL-CRN), the set of kinetics becomes meaningless and the traditional $\Sigma_{\text{MAL}} = (\mathcal{S}, \mathcal{C}, \mathcal{R})$ is obtained.

2.1.2 Reaction graph of the biochemical network

The set of complexes and the set of reactions can be represented by a directed, weighted graph. The reaction graph $\mathbb{G} = (\mathbb{V}; \mathbb{E})$ consists of a finite non-empty set of vertices, which represents the complexes: $\mathbb{V} = \{C_1, \dots C_m\}$ and a finite set of directed edges $\mathbb{E}$, which represents the reactions. The directed edges representing reactions are defined by triplets of the form $e_{(ijl)} = (C_i, C_j, G_l)$ for $i, j = 1, \dots, m$, $i \neq j$, $l = 1, \dots, d_i$, where i, j and l are the indices of the source complex, product complex and the kinetics of the reaction, respectively. The principal reaction rate coefficients and the kinetic functions are assigned as weights to the edges.

Note that, there might be multiple directed edges with different kinetics converting C_i to C_j for any $i, j = 1, \dots, m$, $i \neq j$, which is a fundamental difference from the reaction graph of mass action systems, where such multiple edges are not allowed.

(Weak) Reversibility. A reaction network is called *reversible*, if whenever the reaction $C_i \xrightarrow{k, g_{il}} C_j$ with *any kinetics* g_{il} exists, then a reverse reaction $C_j \xrightarrow{k', g_{jl'}} C_i$ with any other kinetics $g_{jl'}$ is also present in the network. A reaction network is called *weakly reversible*, if whenever complex C_j is reachable from complex C_i on a directed path in the reaction graph, then there exists a directed path from C_j to C_i, too.

Deficiency. The notion of the *deficiency* [53] of a reaction kinetic system is built on the set of *reaction vectors* ($\rho^{(l,k)}$) forming the *stoichiometric subspace* S that is defined as

$$S = \text{span}\{\rho^{(l,k)} = \eta_j - \eta_i \mid (C_i, C_j, G_l) \in \mathbb{E} \text{ for any } l \in \{1, \dots d_i\}\} \tag{2.3}$$

where η_i denotes the ith column of Y. The *deficiency* d of a reaction network is defined as:

$$d = m - \ell - s \tag{2.4}$$

where m is the number of complexes, ℓ is the number of linkage classes and s is the rank of the stoichiometric subspace, i.e. $s = \text{rank}(S)$.

Feinberg [53] proved important properties of the solutions of kinetic systems with mass action law related to the existence, uniqueness and stability of equilibria based on the deficiency and weak reversibility of the network, particularly in the Deficiency Zero and Deficiency One Theorems. With biochemical kinetics, one can apply those points of the Deficiency Zero Theorem 1.2.1 that correspond to arbitrary (not necessarily mass action) kinetics. For example, when the deficiency of the network is zero, but the network is not weakly reversible, there is no strictly positive steady state solution, and there cannot be exist a cyclic trajectory in which all states remain positive [53].

2.1.3 Biochemical reaction rates

The species in each complex may react in different ways, which is described by the concept of kinetics. Therefore a set of kinetics $\mathcal{G}_i = \{G_1, \dots G_{d_i}\}$ was introduced in the previous section for each complex and a function $g_{il}(\cdot)$ was associated to each kinetics. In this section we further characterize the form of these kinetic functions.

Kinetic functions determine the rate of the reactions (2.2) based on the concentrations of a set of species. For each reaction, the species are classified as dominant species, modifiers or neutral species. Neutral species do not participate in the reaction, i.e. their stoichiometric coefficients are zero. The dominant species are the source species, the concentration of which strongly effects the reaction rate, i.e. if any of the dominant species concentration is zero, the reaction rate is also zero. On the other hand, the species are called modifiers of the reaction if their presence may increase (activators) or decrease (inhibitors) the reaction rate, but they are not required for that given reaction – the reaction rate is non-zero even though the modifier concentration is zero.

The complexes are formed by the dominant species. Therefore, if any of the species has zero concentration in a complex all the corresponding reaction rate functions are also zero. Further, we take the following assumptions:

(KA1) any kinetic function $g_{il}(\cdot)$ is always non-negative,
(KA2) any kinetic function $g_{il}(\cdot)$ is zero if and only if the concentration of any of its source species is zero.

To be consistent with assumption (KA1) the reversible reactions are represented by two reactions.

The reaction rate functions are stated in polynomial or rational polynomial form in most kinetic biochemical ODE models. Thus, from now on, we assume that the reaction rate can be written as a ratio of two functions as

$$g_{il}(x) = \frac{\Psi_i(x)}{D_{il}(x)}, \tag{2.5}$$

where $\Psi_i(x)$ is a monomial function

$$\Psi_i(x) = \prod_{k=1}^{n} x_k^{\nu_{ki}} \tag{2.6}$$

and $D_{il}(x)$ is a positive polynomial function of the concentration vector. To make the decomposition (2.2) unique, $g_{il}(x)$ must not contain any linear scaling factor. Therefore the zero-order term in the denominator polynomial is fixed to 1, i.e. $D_{il}(x)$ is written as

$$D_{il}(x) = 1 + \sum \alpha_{m_1,m_2,\dots m_n} x_1^{m_1} x_2^{m_2} \dots x_n^{m_n} \quad , \tag{2.7}$$

where $\alpha \in \mathbb{R}_{\geq 0}$ and $m_1, m_2, \ldots, m_n$ are non-negative integers. Similar decomposition of the reaction rates was used for example in [63] for the verification of biochemical dynamic models and in [65] for model reduction.

2.1.4 Differential equations of the biochemical models

Reactions transform initial substances to products and we are interested in the concentrations of these substances in time. Balance equations are formulated, which account for the production and consumptions of the species in the following way:

$$\dot{x} = \sum_{r_{ijk} \in \mathcal{R}} r_{ijk}(x)(Y_{\cdot,j} - Y_{\cdot,i}) = \sum_{r_{ijk} \in \mathcal{R}} k_{ijk} g_{ik}(x)(Y_{\cdot,j} - Y_{\cdot,i}), \tag{2.8}$$

where $Y_{\cdot,j}$ denotes the j-th column of the complex composition matrix Y, i.e. the vector of stoichiometric coefficients of species in complex C_j.

The mass balance equations form a dynamic, ODE model which describes the concentration trajectories of the species. This systems of ODEs can be transformed to have a special structure, which let us read out the structure of the reaction graph from the equations. Motivated by the ODE structure of MAL-CRNs [53], we define the dynamic model of bio-CRN into the following form

$$\dot{x} = Y \cdot A_k \cdot P(x) \cdot \Psi(x), \tag{2.9}$$

where $x \in \mathbb{R}^n$ is the concentration vector of the species, Y is the complex composition matrix, A_k is the generalized Kirchhoff matrix, $P(.)$ is the kinetic weighting function and $\Psi(.)$ is a monomial vector function. In the following part we go through each of the terms introduced.

The generalized Kirchhoff matrix $A_\mathrm{k} \in \mathbb{R}^{m \times r}$ stores the principal reaction rate coefficients and it is a column conservation matrix. When only one kinetics is associated to each complex, for example in mass action networks, A_k is a square matrix, see (1.6). The off-diagonal elements of the matrix are reaction rate coefficients and the diagonal elements contains the corresponding negative columns sums.

In general, more than one kinetics can be associated to each complex. In this case the number of kinetics of each complex C_i is denoted by d_i. Further, let κ be the sum of kinetics in all complexes: $\kappa = \sum_{i=1}^{m} d_i$. Then the generalized Kirchhoff matrix $A_k \in \mathbb{R}^{m \times \kappa}$ can be written as the concatenation of m blocks

$$A_k = \left[A_k^{(1)} \ldots A_k^{(i)} \ldots A_k^{(m)} \right], \tag{2.10}$$

where the size of block $A_k^{(i)}$ is $m \times d_i$ for $i = 1 \ldots m$. The precise structure of each block is as follows. The j-th row of the block $A_k^{(i)}$ $(i \neq j)$ contains the principal reaction rate coefficients of the reactions from the complex C_i to complex C_j with kinetics g_{il} for $l = 1 \ldots d_i$. The elements of i-th row of the i-th block contains the negative sum of the column elements.

For later reference, let us define formally the generalized Kirchhoff matrix similar to (1.6)

$$A_{\mathrm{k},i,z_j+l} = \begin{cases} k_{jil} & \text{if } i \neq j, \\ -\sum_{o=1, o \neq i}^{m} k_{jo,l} & \text{if } i = j \end{cases} \quad \text{for } i = 1 \ldots m, j = 1 \ldots m \text{ and } l = 1 \ldots d_i. \tag{2.11}$$

Here we introduced the index variable

$$z_j = \sum_{k=1}^{j-1} d_k \quad \text{for} \quad j = 1 \dots m, \tag{2.12}$$

which denotes the sum of the number of kinetics in the first $j-1$ complexes, i.e. the number of columns of the first $j-1$ blocks in (2.10). By definition $z_1 = 0$.

Defining the generalized Kirchhoff matrix for bio-CRNs in the above way has the following advantages. The matrix has a close relationship with the complex-reaction graph: the non-zero elements correspond to the edges of the reaction graph. The locations of the non-zero elements will be referred as the *structure of the A_k matrix*. Let us take $A_k^{(i)}$ as an example. An empty row in this block, e.g. row j, indicates no reaction taking place from complex C_i to complex C_j. Consequently an empty block indicates no reaction from complex C_i to any other complexes. Further, the number of columns of a block tells the number of possible kinetics of the corresponding complex.

The kinetic weighting function $P : \mathbb{R}^n \to \mathbb{R}^{\kappa \times m}$ contains the denominator terms of the reaction rate functions (c.f. (2.2)) arranged in a matrix form

$$P(x) = \begin{bmatrix} P^{(1)}(x) & 0_{d_1} & \dots & 0_{d_1} \\ 0_{d_2} & P^{(2)}(x) & \dots & 0_{d_2} \\ & \vdots & & \\ 0_{d_m} & 0_{d_m} & \dots & P^{(m)}(x) \end{bmatrix}. \tag{2.13}$$

Here each block $P^{(i)}$ (for $i = 1 \dots m$) is of size $d_i \times 1$ and 0_{d_i} denotes a d_i dimensional zero vector. Each block $P^{(i)}$ contains the denominators of the kinetics g_{i1}, g_{i2}, $\dots g_{id_i}$, for example $\left[P^{(i)}\right]_l = \frac{1}{D_{il}(x)}$.

Note that for MAL-CRNs the reaction rate function (1.2) do not have denominator, thus the P is the m-dimensional identity matrix.

The monomial function $\Psi : \mathbb{R}^n \to \mathbb{R}^m$ is a vector function such that each element is a monomial

$$\Psi_i(x) = \prod_{j=1}^{n} x_j^{Y_{ji}} \quad \text{for} \quad i = 1 \dots n \ . \tag{2.14}$$

Note, that the monomial function has a close relationship with the complex composition matrix. Further, the elements of the monomial functions appear in the nominator of the reaction kinetics (2.5).

From the above, it is clear that the bio-CRN Σ can be equally well characterized either by $\Sigma = (\mathcal{S}, \mathcal{C}, \mathcal{G}, \mathcal{R})$ or by $\Sigma = (Y, A_k, P)$. While the former is used in the analysis of the network, the latter is more suitable for computational purposes. Equation (2.9) is called the *normal form* of the dynamic equations.

The vector of kinetics is defined as the product

$$P(x) \cdot \Psi(x) = [g_{11}(x),\, g_{12}(x),\, \dots g_{md_m}(x)]^T := \varphi(x) \ . \tag{2.15}$$

This vector simply collects all the reaction rates without the principal reaction rate coefficients.

2.1.5 Non-negativity of the solutions

The dynamic variables x of any biochemical model are species concentrations, which are naturally non-negative. In what follows we show that all the solutions started from the non-negative orthant remain non-negative for all time, independently of the numerical values of the reaction rate coefficients.

The analysis is based on the notion of essentially non-negative functions [51, 66–68]. A function $f = [f_1 \dots f_n]^T : [0,\infty)^n \mapsto \mathbb{R}^n$ is called essentially non-negative if, for all $i = 1, \dots, n$ and $x \in [0,\infty)^n$, $f_i(x) \geq 0$, whenever $x_i = 0$. Let us consider an autonomous non-linear system

$$\dot{x}(t) = f(x(t)), \quad x(0) = x_0, \quad t \in [0, t_f) \ . \tag{2.16}$$

Haddad [66] showed that for a locally Lipschitz f function, the non-negative orthant $\overline{\mathbb{R}}^n_+$ is invariant under the system dynamics (2.16) (i.e. starting from a non-negative initial condition, all the state variables in x remain non-negative for all time) if and only if f is essentially non-negative. For example, it is shown in [51] and in [69] that MAL-CRNs are essentially non-negative, and therefore all the solutions are non-negative.

Similar results can be obtained for the bio-CRNs as a consequence of the properties (KA1)-(KA3) of the kinetic functions as shown in Appendix A.2.

2.1.6 Examples

Here we present an example, which considers a biochemical network in the presented mathematical structure. A similar example can be found in Appendix A.1 which shows that the MAL-CRNs form a special case of the bio-CRNs.

Example 2-1 (An example with rational kinetics)

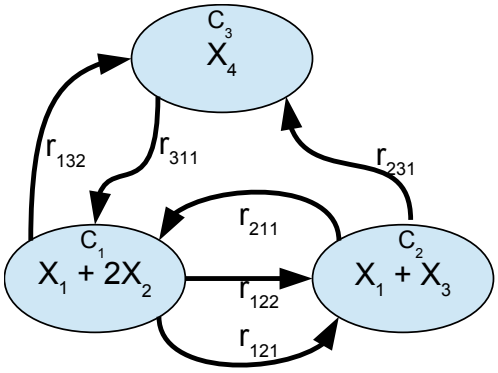


Figure 2.1: Reaction graph of a simple bio-CRN.

Consider the biochemical network in Figure 2.1. There are three complexes:

$$C_1 = X_1 + 2X_2, \quad C_2 = X_1 + X_3, \text{ and } \quad C_3 = X_4,$$

which are connected by six irreversible reaction steps. The reaction rate functions are given as follows

$$r_{121} = k_1 \frac{x_1 x_2^2}{1 + K_{11} x_1}\,, \qquad r_{122} = k_2 \frac{x_1 x_2^2}{1 + K_{21} x_1 + K_{22} x_1 x_2}\,, \qquad r_{211} = k_3 \frac{x_1 x_3}{1 + K_{31} x_3}$$
$$r_{132} = k_4 \frac{x_1 x_2^2}{1 + K_{41} x_4}\,, \qquad r_{311} = k_5 x_4\,, \qquad r_{231} = k_6 \frac{x_1 x_3}{1 + K_{31} x_3}.$$

The stoichiometric coefficients of the species can be read out from the complexes and thus the complex composition matrix and the corresponding monomial vector function are

$$Y = \begin{bmatrix} 1 & 1 & 0 \\ 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}\,, \quad \Psi(x) = \begin{bmatrix} x_1 x_2^2 \\ x_1 x_3 \\ x_4 \end{bmatrix}. \tag{2.17}$$

The kinetics (2.5) corresponding to the reaction rate functions are

$$g_{11} = \frac{x_1 x_2^2}{1 + K_{11} x_1}\,, \qquad g_{12} = \frac{x_1 x_2^2}{1 + K_{21} x_1 + K_{22} x_1 x_2}\,, \qquad g_{21} = \frac{x_1 x_3}{1 + K_{31} x_3}$$
$$g_{13} = \frac{x_1 x_2^2}{1 + K_{41} x_4}\,, \qquad g_{31} = x_4\,.$$

Note that three kinetics are corresponding to complex C_1 and one-one kinetics to complexes C_2 and C_3. Further, the kinetics of the reactions rate r_{211} and r_{231} are the same. The kinetic weighting function (2.13) contains the denominator terms of the reaction kinetics. There are three complexes, thus according to (2.13) $P^{(1)}$ contains the denominators of g_{11}, g_{12} and g_{13}, $P^{(2)}$ is the denominator of g_{21} and $P^{(3)}$ is the denominator of g_{31}, i.e. 1 for this case. This results in the following kinetic weighting function:

$$P(x) = \begin{bmatrix} \frac{1}{1+K_{11}x_1} & 0 & 0 \\ \frac{1}{1+K_{21}x_1+K_{22}x_1x_2} & 0 & 0 \\ \frac{1}{1+K_{41}x_4} & 0 & 0 \\ 0 & \frac{1}{1+K_{31}x_3} & 0 \\ 0 & 0 & 1 \end{bmatrix}\,.$$

From the reaction graph and reaction rates we can read out the principal reaction rate coefficients and the generalized Kirchhoff matrix can be built. The generalized Kirchhoff matrix (2.10) has three blocks corresponding to the three source complexes. The block $A_k^{(1)}$ is made up by the rate coefficients of the reactions starting from complex C_1. For example, reaction r_{121} transforms substances from complex C_1 to C_2 by kinetics g_{11}, thus $[A_k^{(1)}]_{21} = k_1$. There is no reaction from complex C_1 to complex C_3 with kinetics g_{11}, therefore $[A_k^{(1)}]_{31} = 0$. The negative column sum is written in $[A_k^{(1)}]_{11}$. There is a parallel reaction from C_1 to C_2 with kinetics g_{12} with coefficients k_2, thus $[A_k^{(1)}]_{22} = k_2$. The last reaction from complex C_1 transforms the material to complex C_3 by the third kinetics, g_{13}, therefore the corresponding reaction rate goes in $[A_k^{(1)}]_{33}$. Both reactions from complex C_2 has the same kinetics, thus the second block is a simple column vector $A_k^{(2)} = [k_3,\, -k_3 - k_6,\, k_6]^T$.

There is only one reaction starting from complex C_3. It points to complex C_1, thus $A_k^{(3)} = [k_5,\, 0,\, -k_5]^T$. The concatenation of the blocks results in the following generalized Kirchhoff matrix of the bio-CRN example reads as:

$$A_{\mathrm{k}} = \begin{bmatrix} -k_1 & -k_2 & -k_4 & k_3 & k_5 \\ k_1 & k_2 & 0 & -(k_3+k_6) & 0 \\ 0 & 0 & k_4 & k_6 & -k_5 \end{bmatrix} \ .$$

The product $f(x) = YA_kP(x)\Psi(x)$ results in the kinetic systems of ordinary differential equations:

$$\begin{aligned}
\dot{x}_1 =& k_5x_4 - k_4\frac{x_1x_2^2}{1+K_{41}x_4} - k_6\frac{x_1x_3}{1+K_{31}x_3} \\
\dot{x}_2 =& -2k_1\frac{x_1x_2^2}{1+K_{11}x_1} - 2k_2\frac{x_1x_2^2}{1+K_{21}x_1+K_{22}x_1x_2} - 2k_4\frac{x_1x_2^2}{1+K_{41}x_4} + \\
& + 2k_3\frac{x_1x_3}{1+K_{31}x_3} + 2k_5x_4 \\
\dot{x}_3 =& k_1\frac{x_1x_2^2}{1+K_{11}x_1} + k_2\frac{x_1x_2^2}{1+K_{21}x_1+K_{22}x_1x_2} - (k_3+k_6)\frac{x_1x_3}{1+K_{31}x_3} \\
\dot{x}_4 =& k_4\frac{x_1x_2^2}{1+K_{41}x_4} + k_6\frac{x_1x_3}{1+K_{31}x_3} - k_5x_4 \ .
\end{aligned}$$

■

2.2 Model verification of biochemical models

In this section we present criteria of biochemical origin that ensure the plausibility of the studied models and the positivity (more precisely, non-negativity) of their solutions. Further, we aim at localizing non-plausible reaction rate functions of the models. We have to note, that similar techniques are available scattered in the literature and computational tools. For example, COPASI [55] checks the plausibility of the reaction rate functions, which we call here the kinetic property of the reaction rates.

2.2.1 Plausible biochemical reaction rates

Because of the chemical meaning of the reaction rates, the reaction rate functions should posses the following properties.

1. *Rate positivity.* As the elementary reaction steps are irreversible and the reaction rate is defined as the rate of the consumption (decrease) of the reactant concentrations, the inequality $r_{ijk}(x) \geq 0$ should be fulfilled over the entire domain of the reaction rate function.
2. *Kinetic dependence.* The reaction rate function should be zero if and only if at least one of the species concentration in the source complex is zero, i.e. r_{ijk} is said to be *kinetic with respect to the species* in the source complex ($X_k \in C_i$) if

$$r_{ijk}(x_l = 0) = 0 \quad \text{for all} \quad l = \{1, \ldots, n | X_l \in C_i\} \ . \tag{2.18}$$

The reaction rate with the above two properties will be termed plausible. These conditions are also in agreement with the assumptions (KA1 and KA2) for the bio-CRN reaction rate

functions in Subsection 2.1.3.

2.2.2 Plausibility of some common biochemical reaction rate functions

Only a limited number of rate function types are usually present in biochemical reaction systems, that are characterized by a functional form and the values of its parameters [70]. A few of the most important ones are analysed for plausibility below.

(i) *Mass action kinetics*
This is the simplest reaction rate function form $r_{\mathrm{MA},i} = k_i \cdot \prod_{l=1}^{n} x_l^{\nu_{li}}$ where $k_i > 0$ is the reaction rate constant, and the reactant complex is $C_i = \sum_{l=1}^{n} \nu_{li} X_l$. It is easy to see that $r_{\mathrm{MA},i}$ is kinetic in each of the species in complex C_i.

(ii) *Michaelis-Menten kinetics*
Recall, that elementary reaction steps are irreversible, then the rate function is in the form

$$r_{\mathrm{MM},i} = k_i \cdot \frac{x_i}{(K_i + x_i)} \tag{2.19}$$

where where $k_i > 0$ and $K_i > 0$ are constant parameters, and the reactant complex $C_i = X_i$. This reaction rate function is kinetic in X_i.

(iii) *Constant level reactions*
Here the rate function is simply a constant, i.e. $r_{\mathrm{C},i} = k_i^M$, where $k_i^M > 0$ is a constant. This rate function does not have kinetic dependence on any specie, thus no reactant species can be associated to this reaction. Consequently *it is not a plausible reaction rate function*, whenever it is a consuming reaction. Note that, when this reaction stands for model input it always occurs with positive sign in the balance equations.

Correcting non-plausible reaction rates. There is unfortunately no general way of correcting non-plausible reaction rates. However, in some cases, such rates can be made plausible. An example of this case is, when a constant level type reaction rate function is present in the kinetic equation of the species X_i with negative sign. Then we can multiply the rate function with x_i that will make this rate function kinetic in X_i.

2.2.3 Mass conservation

The conservation equations are constructed for species that are either reactants or products of the chemical reactions in the form

$$\frac{dx_k}{dt} = -\sum_{i=1}^{m} \nu_{ki} r_i + \sum_{i=1}^{m} \mu_{ki} r_i = \sum_{i=1}^{m} s_{ki} r_i, \tag{2.20}$$

[71] defined the *mass conservative reaction set* as a set of the reactions having the mass conservative property below. Let us define w_i as the molecular weight of the specie X_i with strictly positive value. If reaction $C_l \to C_j$ is present in the network, the following can be written:

$$\sum_{i=1}^{n} Y_{il} w_i = \sum_{i=1}^{n} Y_{ij} w_i = c_s \tag{2.21}$$

where $c_s > 0$ is a constant weighted column-sum. Let us define vector $w \in \mathbb{R}_+^n$ as a row vector composed of the molecular weights. Now, (2.21) can be rewritten as $w \cdot \eta_l = w \cdot \eta_j = c_s$

where η_l refers to the lth column of matrix Y, or with other words, the composition vector of complex C_l. Finally, it can be said that a reaction is mass conservative if the following holds:

$$w \cdot \rho^{(l,j)} = 0 \ , \tag{2.22}$$

where $\rho^{(l,j)} = \eta^{(l)} - \eta^{(j)}$ is the *reaction vector* and w is strictly positive. It should be noted that a given CRN is called mass conservative if all of its reactions are in the mass conservative reaction set and a common w can be determined. A common way to test this condition is to collect the reaction vectors in a matrix form S, similarly as defined in (2.3) and test if there exists a strictly positive vector m such that $m^T S = 0$, i.e. there is a positive vector in the left kernel of S (see [72], [73] and for efficient computation methods [74]).

Mass conservation in open systems. As we have seen before, kinetic models are originally constructed based on the conservation of the masses of species assuming *closed systems* and isothermal conditions. The introduction of the zero complex enabled to construct a CRN model for an open system but then the condition of (2.21) is violated [71].

In addition to the mass conservative reactions, the CRN model of open systems has (i) *input terms*, that have positive sign and may depend on externally set concentrations and/or mass flow of certain non-conserved specie, i.e. these are *positive constants*, and (ii) *output terms*, that are linear in one conserved specie, have negative sign and appear only in the dynamic equation of that particular specie. Typically, the dilution due to cell growth can represented by such an output term. Therefore, *all of the input and output terms should be set to zero when checking the conservation property*: this form of the dynamic model will be called the *truncated model*.

To check the mass conservation, a *truncated stoichiometric matrix* $\tilde{Y} \in \mathbb{R}^{n \times m}$ can be constructed from stoichiometric coefficients of the truncated model by associating a column $\tilde{\eta}^{(i)}$ to each complex C_i. The truncated biochemical model has the conservation property, if there exists a strictly positive vector w such that (2.22) holds for all of the truncated reaction vectors $\tilde{\rho}^{(l,j)} = \tilde{\eta}^{(l)} - \tilde{\eta}^{(j)}$.

2.2.4 Plausible model structure

The model structure is said to be plausible, when the stoichiometric constants in the conservation equations (2.20) are consistent with the reactants and products of the reactions, i.e. ν_{ki} is strictly positive if reaction r_i consumes the species X_k and μ_{ki} is strictly positive if X_k is a product of the reaction r_i. The stoichiometric coefficient of a reaction which neither consumes nor produces a species should be zero in the corresponding balance equation.

2.2.5 Verification of some biochemical models

Here we present several examples, where we used the above presented tools to localize potential modeling pitfalls. First, the working pipeline is illustrated by the analysis of a bioreactor model.

Example 2-2 (A continuous flow stirred tank bio-reactor)

A homogeneous, continuous flow stirred tank bioreactor (CFSTR) serves as an tutorial example to illustrate the verification steps. depicted in Figure 2.2. The reaction network consists of three species (A, B, C), their concentrations are denoted by x_A, x_B and x_C, respectively.

There are two elementary reactions: a two substrate, one product reversible Michaelis-Menten kinetics (2.23) and a non-plausible (see subsection 2.2.2 case (iii)) constant reaction (2.24). The zero complex denoted by a "0" in the reaction graph corresponds to the environment of the system. The reactor feed contains the substrates A and B with x_A^f and x_B^f constant concentrations, respectively. These inflows are represented by the pseudo-reactions in Equations (2.25). The output stream that contains all species is represented by the pseudo-reactions in Equations (2.26).

The reaction rate functions and the ODEs of the CFSTR are

$$r_1 = V_f \frac{\frac{x_A}{K_{x_A}}\frac{x_B}{K_{x_B}}}{1 + \frac{x_A}{K_{x_A}} + \frac{x_B}{K_{x_B}}} - V_r \frac{\frac{x_C}{K_{x_C}}}{1 + \frac{x_C}{K_{x_C}}} \tag{2.23}$$

$$r_2 = K_d \tag{2.24}$$

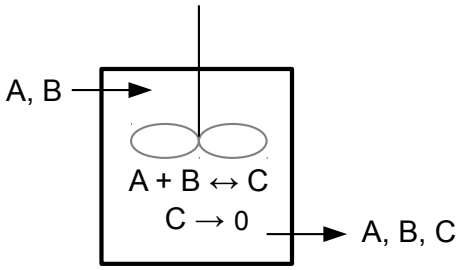
$$r_3 = \zeta x_A^f; \qquad r_4 = \zeta x_B^f \tag{2.25}$$

$$r_5 = \zeta x_A; \qquad r_6 = \zeta x_B; \qquad r_7 = \zeta x_C \tag{2.26}$$

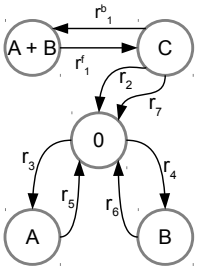
$$\frac{dx_A}{dt} = -r_1 + r_3 - r_5 \tag{2.27}$$

$$\frac{dx_B}{dt} = -r_1 + r_4 - r_6 \tag{2.28}$$

$$\frac{dx_C}{dt} = r_1 - r_2 - r_7 \ . \tag{2.29}$$



(a) The reactor has a constant inflow containing species A and B. Species A and B form C in a reversible reaction. Species C is degrading, which is represented by the reaction to the zero complex. The continuous outflow of the reactor contains all the three species.



(b) The reaction graph associated to the CFSTR. The inflow and the outflow are represented by pseudo reactions from/to the zero complex.

Figure 2.2: Continuous flow stirred tank reactor (CFSTR) and its reaction graph representation.

Splitting the reversible reactions The first correction to this example is by representing each reversible reaction by two irreversible elementary steps. In this example Equation (2.23) is separated to a forward and a backward reactions

$$r_1^f = V_f \frac{\frac{x_A}{K_{x_A}}\frac{x_B}{K_{x_B}}}{1 + \frac{x_A}{K_{x_A}} + \frac{x_B}{K_{x_B}}} \text{ and } r_1^b = V_r \frac{\frac{x_C}{K_{x_C}}}{1 + \frac{x_C}{K_{x_C}}} \ . \tag{2.30}$$

Simultaneously the differential equations are updated to

$$\frac{dx_A}{dt} = -(r_1^f - r_1^b) + r_3 - r_5 \tag{2.31}$$
$$\frac{dx_B}{dt} = -(r_1^f - r_1^b) + r_4 - r_6 \tag{2.32}$$
$$\frac{dx_C}{dt} = (r_1^f - r_1^b) - r_2 - r_7 \ . \tag{2.33}$$

Checking model positivity by the kinetic property Next, the stoichiometric matrix (S) is constructed based on the ODEs. Whenever the v_{ij} element of S is negative, i.e. reaction r_j consumes the species x_i, r_j must be kinetic with respect to x_i. This can be checked by substituting zeros for the species x_i in the rate function and evaluating it; the result must be zero. In our example the model Eqs. (2.31)-(2.33) give rise to the stoichiometric matrix

$$S = \begin{bmatrix} -1 & 1 & 0 & 1 & 0 & -1 & 0 & 0 \\ -1 & 1 & 0 & 0 & 1 & 0 & -1 & 0 \\ 1 & -1 & -1 & 0 & 0 & 0 & 0 & -1 \end{bmatrix} \tag{2.34}$$

and the irreversible reaction vector $R = [r_1^f, r_1^b, r_2, r_3, r_4, r_5, r_6, r_7]^T$. Considering the location of the negative entries of S reaction r_1^f and r_5 must be kinetic to species A, r_1^f and r_6 with respect to B and r_1^b, r_2 and r_7 with respect to C.

By substituting zeros for the reactant species in the rate functions –e.g. $r_1^f(x_A = 0)$, $r_5(x_A = 0)$, $r_1^f(x_B = 0)$ etc. – the plausible ones give zeros. At this point reaction r_2 is found to be non-kinetic to the species C, and thus it is a non-plausible reaction. We may correct the rate function by multiplying with its reactant species concentration: $r_2^* = K_d x_C$. This reaction can be regarded as a model output, too.

Component mass conservation The truncated model without the input reactions (Equation (2.25)) and the output reactions (Equation (2.26)) and the corrected r_2^*) is represented by the first two columns of S. This sub–matrix is rank deficient and has the $m = [1\,1\,2]^T$ strictly positive vector in the left nullspace indicating the mass conservation law. ■

Biomodels Database [56] is a collection of published and curated biochemical models. We selected models from the database, which are related to the *Escherichia coli* bacterium. These models are rather different by their nature, for example, BIOMD296 considers chemical interactions between populations, BIOMD067 presents a gene-metabolic oscillator, BIOMD66 focuses on the threonine-synthesis pathway, and BIOMD51 is a model for the central carbon metabolism of the bacteria. Table 2.1 contains the unique identifiers of the considered models in the database. Further details about each model can be obtained through the database.

Among the 11 considered *E. coli* models some contain non-plausible reactions. The number of species, the number of reactions and the computation time of the algorithm is also included in the following columns. The 5th column contains the non-plausible reaction, while the last column shows whether the truncated model admits mass conservation. In Appendix B the verification of the central carbon metabolism model BIOMD051 is also presented in details.

Table 2.1: Verified models. The models were obtained from the BioModels database. The model ID, the number of dynamic equations, number of reactions, the required computation time on a Dell Precision T5500 (Intel Xeon 2.4GHz) and a short description of the problem found are depicted in the columns.

BioModel ID	No. of species	No. of reactions	Time [s]	Non-plausible reaction	Mass conservation
BIOMD296	4	10	man.*	plausible	no
BIOMD413	5	9	0.3	plausible	no
BIOMD200	22	46	2.3	plausible	yes
BIOMD217	12	22	23	plausible	yes
BIOMD051	18	62	5	reaction vMURSYNTH is not kinetic w.r.t. species CF6P	no
BIOMD066	11	10	man.*	reaction vATPASE is not kinetic w.r.t. species ATP	yes
BIOMD012	6	12	0.8	plausible	no
BIOMD067	7	16	0.6	plausible	no
BIOMD221	8	22	1.9	reaction vSYN is not kinetic w.r.t. species AKG	no
BIOMD222	8	22	1.9	reaction vSYN is not kinetic w.r.t. species AKG	no
BIOMD065	8	16	0.5	plausible	no

*the separation of some reaction rate function needed manual manipulation

2.3 Summary

In this chapter the traditional mass-action chemical reaction network formalism was extended for mathematical models with biochemical reactions. The main purpose of this work was to formulate the bio-CRNs balance equations such that the equations have an intrinsic relationship to the complex-reaction graph.

The form of the reaction rate functions was confined to rational functions that have monomials in the nominator and contain positive polynomials in the denominator. Based on the mathematical form of these rate functions we introduced the notion of reaction kinetics. Unlike in the MAL-CRN case, here the species in each complex can have multiple kinetics. Thus, it is possible to describe, for example, when both a Michaelis-Menten and a MAL kinetic function are assigned to a source complex. This results in that the biochemical reaction graph may contain parallel edges, independent reactions, which is not allowed in the traditional MAL-CRNs. This led us to revise the MAL-CRN reaction graph definition and its properties, such as the reversibility, weak reversibility and deficiency.

The non-negativity of state-variables is a fundamental feature of any biological model in which the state variables repsresent concentrations. We showed that the dynamics of bio-CRN models are essentially non-negative. Therefore, independently of the numerical values of the model parameters, the solutions of the dynamic equations remain in the non-negative orthan.

In the second half of this chapter we focused on the verification of biochemical models. We developed simple syntactical checks, both reaction rate level and model structure level criteria, which can detect potential pitfalls or implicit modeling assumptions. We note, that similar approaches exist scattered in literature and in computer programs, such as checking the plausability of the reaction rates or mass conservation in closed systems. Our contribution here are the formulation and implementation of these procedures in a integrated framework, and extension of the mass conservation checking for open systems.

Chapter 3

Realization and computation of equivalent and conjugated networks

Biochemical reaction networks, presented in the previous chapter, provide a way to analyse the dynamics of non-linear ordinary differential equations by means of the structure of the associated reaction graph. Therefore it is our interest to find a bio-CRN for a given system of ordinary differential equations. When there is at least one such a bio-CRN, then the system of ordinary differential equations describe a *kinetic system* and the corresponding bio-CRN is called a *realization* of the equations.

Our first goal is to find a graph representation for a kinetic system. Hence, we determine necessary and sufficient conditions for the dynamics such that a bio-CRN representation can be determined. These are the kinetic realizability conditions. Then, we will present an algorithm, which can construct the network.

Similarly to the MAL-CRNs, presented in Section 1.2.1, here we also observe that the reaction graph is not unique, i.e. more than one reaction network can be associated to the kinetic system. This phenomena is known as the dynamic equivalence of the networks or – from the realizations points of view– the confoundability of kinetic systems [44].

Dynamically equivalent networks might have different graph properties. For example, the number of edges, reversibility, weak reversibility or deficiency are realization dependent properties. Therefore it is of our further interest to find networks with given graph properties. Numerical optimization methods are presented which can be used to find sparse, dense and reversible dynamically equivalent networks.

Johnston and Siegel [58] defined the linear conjugacy of MAL-CRNs. Two MAL-CRNs are called linearly conjugated if their trajectories (solutions) can be mapped into each other by a positive linear transformation, i.e. by scaling the state variables. It is well-known that for example, stability condition is preserved under linear state transformation. Therefore, linearly conjugated networks share stability properties. In the last part of this chapter we extend the linear conjugacy theorem for bio-CRNs.

3.1 Kinetic realizations of ODEs with rational function right-hand sides

Hárs and Tóth [57] considered the inverse problem of the chemical reaction networks. They aimed at finding a MAL-CRN realization for a system of ordinary differential equations, which has polynomials in their right hand side functions. Apart from the necessary and

sufficient conditions, their constructive proof resulted in an algorithm –the canonical realization algorithm–, which can determine the elements of the corresponding MAL-CRN.

Here we follow a similar approach for biochemical reaction networks. The main difference is that the right hand side functions of the ODEs contain rational rate expressions instead of simple polynomials.

3.1.1 Kinetic realizability conditions

The following theorem gives necessary and sufficient conditions for a system of autonomous differential equations such that a bio-CRN can be associated to the dynamics.

Theorem 3.1.1 (Kinetic realizability)**.** *Given an autonomous system of ordinary equations*

$$\dot{x}(t) = f(x(t)), \quad x(0) = x_0, \quad t[0, t_f), \tag{3.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}^n$, $x_0 \geq 0$ and $x \in \mathbb{R}^n$. Then, there exists a bio-CRN of n species with the dynamics (3.1) *if and only if for each $i = 1, \ldots n$ (I) the i-th component of f, $f_i(x)$ is a linear combination of biochemical reaction rate functions(see Equations* (2.2) *and* (2.5)*) and (II) the function*

$$f_i([x_1, x_2, \ldots, x_{i-1}, 0, x_{i+1} \ldots x_n])$$

is a non-negative *linear combination of the biochemical reaction rate functions.*

The proof of the theorem follows similar path, that was presented by Chellaboina [51] for the kinetic realizability of polynomial systems.

Proof of Sufficiency. According to condition (I), the right hand side function is a linear combination of biochemical reaction rate functions. Thus each term in equation $f_i(x)$ has either the form

$$a_i \frac{x_1^{p_1} x_2^{p_2} \ldots x_i^{p_i} \ldots x_n^{p_n}}{D(x)}, \tag{3.2}$$

where the exponents $p_j \geq 0$ for $j = 1 \ldots n$ and specially $p_i > 0$, or

$$b_i \frac{x_1^{q_1} x_2^{q_2} \ldots x_{i-1}^{q_{i-i}} x_{i+1}^{q_{i+i}} \ldots x_n^{q_n}}{D(x)}, \tag{3.3}$$

where the $b_i > 0$ and the x_i is missing from the nominator, i.e. $q_i \equiv 0$. Let

$$\operatorname{sign}(x) = \begin{cases} 1 & \text{if } x > 0 \\ 0 & \text{if } x = 0 \\ -1 & \text{if } x < 0 \end{cases}.$$

Consider the following reaction

$$\sum_{j=1}^{n} p_j X_j \xrightarrow{\left(|a_i|, \frac{\sum_{j=1}^{n} x_j^{p_j}}{D(x)}\right)} (p_i + \operatorname{sign}(a_i))\, X_i + \sum_{j=1, j\neq i}^{n} p_j X_j, \tag{3.4}$$

where $\sum_{j=1}^{n} p_j X_j$ is the source complex, $(p_i + \operatorname{sign}(a_i))\, X_i + \sum_{j=1, j\neq i}^{n} p_j X_j$ is the product complex, $|a_i|$ is the principal reaction rate coefficient and $\frac{\sum_{j=1}^{n} x_j^{p_j}}{D(x)}$ is the reaction kinetics.

This reaction results the term (3.2) in f_i, but does not contribute to any other f_j, $j = 1, \dots n$, $j \neq i$, since the stoichiometric coefficients for any other X_j are the same in the reactant and in the product side. Further, the reaction

$$\sum_{j=1}^{n} q_j X_j \xrightarrow{\left(b_i, \frac{\sum_{j=1}^{n} x_j^{q_j}}{D(x)}\right)} X_i + \sum_{j=1, j\neq i}^{n} q_j X_j, \tag{3.5}$$

contributes (3.3) in f_i, but nothing for the other f_j for similar reason.

In conclusion, we can associate a reaction (either (3.4) or (3.5)) with each term in f, which proves sufficiency. □

Proof of Necessity. Let $o = 1, \dots n$ and recall Equation (2.8)

$$f_o(x) = \sum_{r_{ijl} \in \mathcal{R}} r_{ijl}(x) \left([\eta_j]_o - [\eta_i]_o\right) \quad .$$

Here η_j is the j-th column of the complex composition matrix Y, and $[\eta_j]_o$ is its o-th entry, i.e. $[\eta_j]_o = Y_{jo}$. Inserting the reaction rate function (2.2) and (2.5) into the above equation yields

$$f_o(x) = \sum_{\mathcal{R}} \frac{k_{ijl}}{D_{il}(x)} x^{\eta_i} \left([\eta_j]_o - [\eta_i]_o\right). \tag{3.6}$$

Here, to simplify the notation we used the following convention: if $a, b \in \mathbb{R}^n$ then $a^b = \prod_{i=1}^{n} a_i^{b_i}$. We see from Equation (3.6) that the differential equations of a biochemical reaction network is a linear combination of biochemical reaction rates, therefore the first part (I) is done.

Regarding (II), we have to show that if $x_o = 0$, then (3.6) is a *non-negative* linear combination of the rate functions. Let $x_o = 0$. If $[\eta_i]_o > 0$, then the monomial x^{η_i} is zero and therefore the corresponding terms in the summation disappear. On the other hand, if $[\eta_i]_o = 0$, then the 0^0 should be further discussed. This term is treated in the following way:

$$\begin{aligned}\frac{k_{ijl}}{D_{il}(x)} x^{\eta_i} \left([\eta_j]_o - [\eta_i]_o\right) &= \lim_{x_o \to 0^+} \frac{k_{ijl}}{D_{il}(x)} x^{[\eta_i]_1} \dots x^{[\eta_i]_{o-1}} x_o^0 x^{[\eta_i]_{o+1}} \dots x^{[\eta_i]_n} [\eta_j]_o = \\ &= \frac{k_{ijl}}{D_{il}([x_1, \dots, x_{o-1}, 0, x_{o-1} \dots x_n]^T)} x^{[\eta_i]_1} \dots x^{[\eta_i]_{o-1}} 1 x^{[\eta_i]_{o+1}} \dots x^{[\eta_i]_n} [\eta_j]_o.\end{aligned}$$

Let $\overline{\mathcal{R}} = \{\text{terms in which } [\eta_i]_o = 0\}$ and so

$$f_o([x_1, x_2, \dots, x_{o-1}, 0, x_{o+1} \dots x_n]) = \sum_{\overline{\mathcal{R}}} \frac{[\eta_j]_o k_{ijl}}{D_{il}(x)} x^{[\eta_i]_1} x^{[\eta_i]_2} \dots x^{[\eta_i]_{o-1}} x^{[\eta_i]_{o+1}} \dots x^{[\eta_i]_n},$$

which is indeed a *non-negative* linear combination of elementary biochemical reaction rate functions, thus we finished with (II). □

3.1.2 Canonical realization algorithm

If the conditions of the kinetic realizability are fulfilled, the first part of the proof of kinetic realizability suggests an easy way to construct a bio-CRN. The goal is to find the components of the bio-CRNs, i.e. the sets of species, complexes, reactions and kinetics. We summarized the full procedure by Algorithm 1.

The algorithm processes each equation one-by-one. It requires the list of rational function terms on the right hand side of the differential equations. First, the source complex is determined based on the exponents of the monomials in the nominator. Then, a product complex is assigned to the reaction based on (3.4) or (3.5). In case of processing the i-th equation, the stoichiometric coefficients of the species in the product and source complex are the same for all the species, except the species X_i. Thus, the inferred reaction does not contribute to any other than the i-th equation. Finally, the kinetics is determined from the functional form of the rational term.

Algorithm 1 can also be seen as an extension of the canonical realization algorithm [51, 57], which is applicable only for polynomial differential equations.

It is known that the canonical realization algorithm generally results in a large number of complexes. Further, the inferred biochemical reaction network does not fulfill thermodynamic constraints and almost never fulfills the mass conservation due to the way the product complex is constructed for each reaction.

The only purpose of the presented algorithm is to generate *a* biochemical reaction network for kinetic systems. The above observations also motivate our work presented in the following part of this chapter, i.e. to find other realizations with desired properties based on numerical optimization.

Example 3-1 (Canonical realization of a biochemical reaction network)

Consider the following system of ordinary differential equations

$$\dot{x}_1 = -k_{21}x_1^3 - \frac{k_{22}x_1^3}{1+K_2x_1} + 3k_{11}x_2^3 + \frac{3k_{12}x_2^3}{1+K_1x_2}$$
$$\dot{x}_2 = k_{21}x_1^3 + \frac{k_{22}x_1^3}{1+K_2x_1} - 3k_{11}x_2^3 - \frac{3k_{12}x_2^3}{1+K_1x_2}.$$

It is easy to check that the kinetic realizability conditions hold for this system, i.e. (i) f_1 and f_2 are linear combinations of biochemical reaction rate functions and (ii) $f_1([0,x_2]^T)$ and $f_1([x_1,0]^t)$ are positive linear combinations of biochemical reaction rate functions.

We applied Algorithm 1 and summarized each step in Table 3.1. Each row of the table corresponds to the identification of a reaction step, i.e. the algorithm finds the source and product complexes, determines the principal reaction rate coefficient and reaction kinetics.

In conclusion, Algorithm 1 found the following set of complexes

$$\mathcal{C} = \{3X_1,\ 2X_1,\ 3X_2,\ X_1+3X_2,\ 3X_1+X_2,\ 3X_2\}$$

and identified the set of kinetics

$$g_{11} = x_1^3 \qquad g_{12} = \frac{x_1^3}{1+K_2x_1}$$
$$g_{31} = x_2^3 \qquad g_{32} = \frac{x_2^3}{1+K_1x_2}\ .$$

Algorithm 1 Algorithm for realization of biochemical reaction network from ODEs.

Require: a set of ODEs of n variables; each f_i is a linear combination of elementary reaction rate functions, i.e.
$f_i(x) = \sum_{l=1}^{m} \left(c_l \frac{\prod_{j=1}^{n} x_j^{\nu_{jl}}}{D_l(x)} \right) = \sum_{l=1}^{m} c_l g_l(x);$

$\mathcal{R} = \emptyset$; # set of reactions
$\mathcal{C} = \emptyset$; # set of complexes
$\mathcal{S} = \{X_1 \dots X_n\}$; # set of species
for $i = 1$ **to** n **do**
  for each $c_l g_l(x)$ **do**
    $C_s = \sum \nu_{jl} X_j$; # source complex
    if $C_s \notin \mathcal{C}$ **then**
      add C_s to $\mathcal{C}$;
    end if
    if $\nu_{il} \geq 1$ **then**
      $C_p = (\nu_{il} + \text{sign}(c_i))\, X_i + \sum_{j=1, j\neq i}^{n} \nu_{jl} X_j$; # product complex
      $r = |c_i| \frac{\sum_{j=1}^{n} x_j^{\nu_{jl}}}{D_l(x)}$; # reaction rate
      $g = \frac{\sum_{j=1}^{n} x_j^{\nu_{jl}}}{D_l(x)}$; # kinetics
    else
      $C_p = X_i + \sum_{j=1, j\neq i}^{n} \nu_{jl} X_j$; # product complex
      $r = c_i \frac{\sum_{j=1}^{n} x_j^{\nu_{jl}}}{D_l(x)}$; # reaction rate
      $g = \frac{\sum_{j=1}^{n} x_j^{\nu_{jl}}}{D_l(x)}$; # kinetics
    end if
    if $C_p \notin \mathcal{C}$ **then**
      add C_p to $\mathcal{C}$;
    end if
    add reaction (C_s, C_p, g) to $\mathcal{R}$;
  end for
end for

Table 3.1: Detailed procedure of the simple realization example 3-1. Each line of the table corresponds to a realized reaction. The first column contains the right hand side component of the ODEs, the second column shows a reaction term of the right hand side function. The third column contains the assigned reaction: the source and the product complexes are written on the sides of the arrow, the principal reaction rate coefficient and the reaction kinetic function is written above the arrow. The found new complexes are in the last column.

Equation	Term	Realized reaction	New complexes
$f_1(x)$	$-k_{21}x_1^3$	$3X_1 \xrightarrow{\lvert -k_{21}\rvert \left(\frac{x_1^3}{1}\right)} (3+\text{sign}(-k_{21}))X_1$	$C_1 = 3X_1$, $C_2 = 2X_1$
$f_1(x)$	$-\frac{k_{22}x_1^3}{1+K_2x_1}$	$3X_1 \xrightarrow{\lvert -k_{22}\rvert \left(\frac{x_1^3}{1+K_2x_1}\right)} (3+\text{sign}(-k_{22}))X_1$	-
$f_1(x)$	$3k_{11}x_2^3$	$3X_2 \xrightarrow{\lvert 3k_{11}\rvert \left(\frac{x_2^3}{1}\right)} X_1 + 3X_2$	$C_3 = 3X_2$, $C_4 = X_1 + 3X_2$
$f_1(x)$	$\frac{3k_{12}x_2^3}{1+K_1x_2}$	$3X_2 \xrightarrow{\lvert 3k_{12}\rvert \left(\frac{x_2^3}{1+K_1x_2}\right)} X_1 + 3X_2$	-
$f_2(x)$	$k_{21}x_1^3$	$3X_1 \xrightarrow{\lvert k_{21}\rvert \left(\frac{x_1^3}{1}\right)} X_2 + 3X_1$	$C_5 = X_2 + 3X_1$
$f_2(x)$	$\frac{k_{22}x_1^3}{1+K_2x_1}$	$3X_1 \xrightarrow{\lvert k_{22}\rvert \left(\frac{x_1^3}{1+K_2x_1}\right)} X_2 + 3X_1$	-
$f_2(x)$	$-3k_{11}x_2^3$	$3X_2 \xrightarrow{\lvert -3k_{11}\rvert \left(\frac{x_2^3}{1}\right)} (3+\text{sign}(-3k_{11}))X_2$	$C_6 = 2X_2$
$f_2(x)$	$-\frac{3k_{12}x_2^3}{1+K_1x_2}$	$3X_2 \xrightarrow{\lvert -3k_{12}\rvert \left(\frac{x_2^3}{1+K_1x_2}\right)} (3+\text{sign}(-3k_{12}))X_2$	-

We can see from the kinetics that only complex C_1 and C_3 are source complexes, each of them with two types of kinetics. The corresponding reaction graph of the bio-CRN can be seen in Figure 3.1.

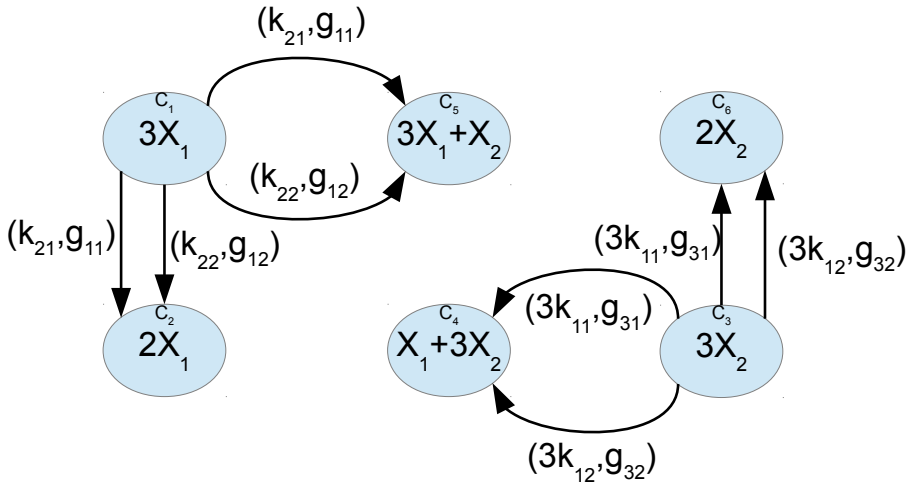


Figure 3.1: The reaction graph of the example constructed by the canonical realization algorithm.

■

3.2 Dynamically equivalent biochemical reaction networks

It has been known that different reaction graph structures in the MAL-CRNs may lead to the same kinetic differential equations. In other words, the reaction graph structure corresponding to a given kinetic ODE system is non-unique. This phenomenon is called macro-equivalence, dynamical equivalence or confoundability [30, 32, 44, 58]. Since the mass action case is a special case of the generalized kinetic description, dynamical equivalence necessarily emerges for the bio-CRN structure, too.

3.2.1 Dynamical equivalence in bio-CRNs

When the kinetic realizability condition is fulfilled, the dynamic equations can be written as linear combinations of kinetic reaction rate functions. In this case, Equations (3.1) can be written using matrix notations as

$$f(x) = \underbrace{Y A_k}_{M} \underbrace{P(x)\Psi(x)}_{\varphi(x)} = M\varphi(x) \tag{3.7}$$

where $\Sigma = (Y,\ A_k,\ P(x))$ is a *realization of the equations. AG: The subsection was rewritten according to these suggestions.*. From now on, we assume that this factorization is fixed and given, for example produced by the Realization Algorithm 1. The coefficient matrix $M \in \mathbb{R}^{n\times\kappa}$ contains the coefficients of the linear combinations and φ is a vector of kinetic functions (2.15).

It is important to note, that the above factorization might be not unique in general and the realization algorithm provide only one of the possible factorizations.

We further assume that the components of the kinetic vector $\varphi(x)$ are linearly independent, i.e.

$$\sum_{i=1}^{\kappa} \alpha_i \varphi_i(x) = 0, \text{ for all } x > 0 \text{ if and only if } \alpha_i = 0 \text{ for } \forall i = 1, \dots \kappa. \tag{3.8}$$

From the network point of view, this assumption guarantees that parallel edges between two complexes represent independent kinetics.

The realization can be seen as a process for finding a complex composition matrix Y, generalized Kirchhoff matrix A_k and a rate weighting function $P(.)$ such that

$$M\varphi(x) = YA_kP(x)\Psi(x) \quad \text{for all} \quad x \in \mathbb{R}^n_+. \tag{3.9}$$

The bio-CRN $\Sigma = (Y, A_\mathrm{k}, P)$ is called a *realization* of the dynamic equations, further the model defined by these dynamic equations is called a *kinetic model* or *kinetic system*.

The realization is non-unique if there exist multiple bio-CRNs, for example $\Sigma_1 = (Y^1,\ A^1_\mathrm{k},\ P^1)$ and $\Sigma_2 = (Y^2,\ A^2_\mathrm{k},\ P^2)$ for which

$$M\varphi(x) = Y^1A^1_\mathrm{k}P^1(x)\Psi^1(x) = Y^2A^2_\mathrm{k}P^2(x)\Psi^2(x) \quad \text{for } \forall x \in \mathbb{R}^n_+ \ . \tag{3.10}$$

Let introduce a shared complex composition matrix Y, which stores the stoichiometric coefficients of the complexes of *both* networks and let P be the common rate weighting function, which also contains the reaction kinetics of both networks by forming the *union* of the two corresponding sets. In this case the generalized Kirchhoff matrix of each network will contain some zero rows and columns corresponding to the other network kinetics and complexes. Further, let $\Psi_i(x) = \prod_{j=1}^m x_j^{Y_{ji}}$, i.e. it contains the monomial functions of both networks.

Using the above conventions, dynamic equivalence of two networks can be stated as

$$YA^1_\mathrm{k}P(x)\Psi(x) = YA^2_\mathrm{k}P(x)\Psi(x). \tag{3.11}$$

Therefore once a global complex composition matrix Y and kinetic weighting function P are fixed, all the possible networks can be generated by suitable changes in the generalized Kirchhoff matrix.

Note that $P(x)\Psi(x) = \varphi(x)$, therefore, when the complexes and reaction kinetics are fixed, the search for dynamically equivalent realizations can be seen as a matrix factorization problem

$$M = YA_k \ , \tag{3.12}$$

where M and Y are given and we are looking for the matrix A_k, which must fulfill certain condition (column conservation and structural properties) to be a generalized Kirchoff matrix, see (2.11).

If two networks are dynamically equivalent, then the affine combination of their generalized Kirchhoff matrix also results in a dynamically equivalent realization. Therefore if there are two dynamically equivalent realizations, then there are infinitely many of them. To show this, let $0 \geq a \geq 1$ and $A^3_\mathrm{k} = aA^1_\mathrm{k} + (1-a)A^2_\mathrm{k}$, then

$$YA^3_\mathrm{k}\varphi(x) = YaA^1_\mathrm{k}\varphi(x) + Y(1-a)A^2_\mathrm{k}\varphi(x) = YA^1_\mathrm{k}\varphi(x) \ , \tag{3.13}$$

where in the second step we used (3.11).

3.2.2 Sparse and dense realizations

When the dynamic system has more than one realizations, it has infinitely many. Some realizations show structural differences, i.e. the corresponding un-weighted reaction graphs are different, while others are different only in the numerical weights of the graph. The reaction network with the least possible number of edges is called the *sparse realization*. Further, the realization with the most edges is a *dense realization*.

Some interesting properties of these special realizations were shown in [75] for MAL-CRNs. These results are adapted here for biochemical reaction networks.

Let $\Sigma = (Y, A_\mathrm{k}, P)$ be a kinetic system with fixed complexes and kinetics, and let $\Sigma^s = (Y, A^s_\mathrm{k}, P)$ and $\Sigma^d = (Y, A^d_\mathrm{k}, P)$ be the dynamically equivalent sparse and dense realizations, respectively.

Proposition 3.2.1. *The un-weighted graph of any dynamically equivalent realization of* Σ *is a sub-graph of the dense realization* Σ^d*.*

Proof. Let $M = YA_k$ and assume that (Y, A_k, P) is a dense realization of Σ and thus, A_k has the most number of positive entries among the possible solutions of $M = YA_\mathrm{k}$. Further assume that A'_k is also a valid generalized Kirchhoff matrix solution of $M = YA'_\mathrm{k}$, but it has an edge, which is not in A_k. Formally, there is $i \leq n, j \leq m, l \leq d_j$, $i \neq j$ such that $A_{\mathrm{k},i,z_j+l} = 0$ but $A'_{\mathrm{k},i,z_j+l} > 0$. Then, it follows from (3.13) that $A''_\mathrm{k} = \frac{1}{2}A'_\mathrm{k} + \frac{1}{2}A_\mathrm{k}$ is also a valid dynamically equivalent realization of Σ, but A''_k has more positive elements than A_k, which is a contradiction. □

Proposition 3.2.2. *The dense realization* Σ^d *is structurally unique.*

Proof. This can be seen based on the previous proof. □

Proposition 3.2.3. *The realization of the kinetic system is structurally unique if and only if the sparse and the dense realizations are structurally identical.*

Proof. ($\Rightarrow$) If the sparse and the dense realizations are structurally identical, then all the realizations are structurally the same, since the dense realization is structurally unique. Thus any realization is structurally unique. ($\Leftarrow$) If the structure of the realization is unique then the dense and the sparse realizations are trivially identical. □

A simple implication of the above propositions is that the dense realization contains all the possible reactions.

Example 3-2 (Example for dynamically equivalent realizations)

Consider the networks depicted in Figure 3.2A with species $\mathcal{S} = \{X_1, X_2, X_3\}$ and complexes $C_1 = 3X_2$, $C_2 = 3X_1$ and $C_3 = 2X_1 + X_2$. The complex composition matrix and the corresponding monomial vector are

$$Y = \begin{bmatrix} 0 & 3 & 2 \\ 3 & 0 & 1 \end{bmatrix} \quad , \quad \Psi(x) = \begin{bmatrix} x_2^3 \\ x_1^3 \\ x_1^2 x_2 \end{bmatrix} \quad .$$

The species in each complex can react with two different kinetics ($d_1 = d_2 = d_3 = 2$):

$$g_{1\,1} = x_2^3, \qquad g_{1\,2} = \frac{x_2^3}{1 + K_1 x_2}, \qquad g_{2\,1} = x_1^3,$$

$$g_{2\,2} = \frac{x_1^3}{1 + K_2 x_1}, \qquad g_{3\,1} = x_1^2 x_2, \qquad g_{3\,2} = \frac{x_1^2 x_2}{1 + K_3 x_1 x_2} \quad ,$$

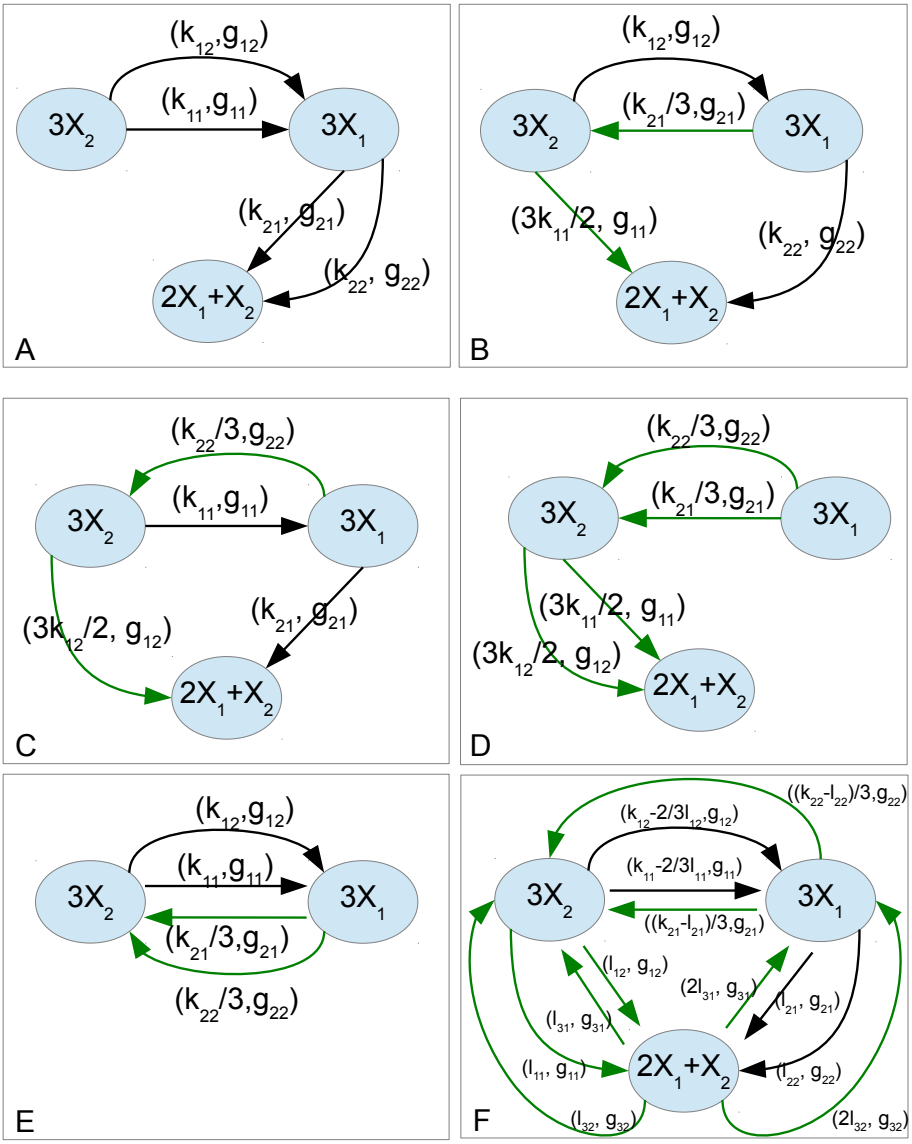


Figure 3.2: Reaction graphs of dynamically equivalent biochemical reaction networks. The differences from the network A are depicted with green edges.

where the first index indicates the source complex and the second index identifies the kinetics. The vector of kinetics can be written as the product of the monomial vector and the rate weighting matrix

$$\varphi(x) = P(x)\Psi(x) = \begin{bmatrix} 1 & 0 & 0 \\ \frac{1}{1+K_1x_2} & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \frac{1}{1+K_2x_1} & 0 \\ 0 & 0 & 1 \\ 0 & 0 & \frac{1}{1+K_3x_1x_2} \end{bmatrix} \cdot \begin{bmatrix} x_2^3 \\ x_1^3 \\ x_1^2x_2 \end{bmatrix} = \begin{bmatrix} x_2^3 \\ \frac{x_2^3}{1+K_1x_2} \\ x_1^3 \\ \frac{x_1^3}{1+K_2x_1} \\ x_1^2x_2 \\ \frac{x_1^2x_2}{1+K_3x_1x_2} \end{bmatrix} .$$

Here, for illustrative purpose, we keep the symbolic values of reaction rate parameters ($k_{12},\ k_{11},\ k_{21}$ and k_{22}) and we present six A_{k}^i matrices $i \in \{A, \dots F\}$, which correspond to six dynamically equivalent networks depicted in Figure 3.2:

$$A_{\mathrm{k}}^A = \begin{bmatrix} -k_{11} & -k_{12} & 0 & 0 & 0 & 0 \\ k_{11} & k_{12} & -k_{21} & -k_{22} & 0 & 0 \\ 0 & 0 & k_{21} & k_{22} & 0 & 0 \end{bmatrix}, \qquad A_{\mathrm{k}}^B = \begin{bmatrix} -\frac{3}{2}k_{11} & -k_{12} & \frac{1}{3}k_{21} & 0 & 0 & 0 \\ 0 & k_{12} & -\frac{1}{3}k_{21} & -k_{22} & 0 & 0 \\ \frac{3}{2}k_{11} & 0 & 0 & k_{22} & 0 & 0 \end{bmatrix}$$

$$A_{\mathrm{k}}^C = \begin{bmatrix} -k_{11} & -\frac{3}{2}k_{12} & 0 & \frac{1}{3}k_{22} & 0 & 0 \\ k_{11} & 0 & -k_{21} & -\frac{1}{3}k_{22} & 0 & 0 \\ 0 & \frac{3}{2}k_{12} & k_{21} & 0 & 0 & 0 \end{bmatrix}, \quad A_{\mathrm{k}}^D = \begin{bmatrix} -\frac{3}{2}k_{11} & -\frac{1}{2}3k_{12} & \frac{1}{3}k_{21} & \frac{1}{3}k_{22} & 0 & 0 \\ 0 & 0 & -\frac{1}{3}k_{21} & -\frac{1}{3}k_{22} & 0 & 0 \\ \frac{3}{2}k_{11} & \frac{3}{2}k_{12} & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$A_{\mathrm{k}}^E = \begin{bmatrix} -k_{11} & -k_{12} & \frac{k_{21}}{3} & \frac{k_{22}}{3} & 0 & 0 \\ k_{11} & k_{12} & -\frac{k_{21}}{3} & -\frac{k_{22}}{3} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix},$$

$$A_{\mathrm{k}}^F = \begin{bmatrix} (-k_{11} - \frac{l_{11}}{3}) & (-k_{12} - \frac{l_{12}}{3}) & \frac{1}{3}(k_{21} - l_{21}) & \frac{1}{3}(k_{22} - l_{22}) & l_{31} & l_{32} \\ (k_{11} - \frac{2l_{11}}{3}) & (k_{12} - \frac{2l_{12}}{3}) & (-\frac{k_{21}}{3} - \frac{2l_{21}}{3}) & (-\frac{k_{22}}{3} - \frac{2l_{22}}{3}) & 2l_{31} & 2l_{32} \\ l_{11} & l_{12} & l_{21} & l_{22} & -3l_{31} & -3l_{32} \end{bmatrix}.$$

One can easily check that all networks ($\Sigma^i = (Y, A_k^i, P)$ for $i = A, B, C, D, E, F$) give rise to the same dynamic equations (2.9), which can be written as

$$\begin{aligned} \dot{x}_1 &= -k_{21}x_1^3 - \frac{k_{22}x_1^3}{1 + K_2x_1} + 3k_{11}x_2^3 + \frac{3k_{12}x_2^3}{1 + K_1x_2} \\ \dot{x}_2 &= k_{21}x_1^3 + \frac{k_{22}x_1^3}{1 + K_2x_1} - 3k_{11}x_2^3 - \frac{3k_{12}x_2^3}{1 + K_1x_2} \end{aligned}.$$

We can see in Figure 3.2 that the structure of the networks are rather different. The networks A-E contain only 4 reactions, which is the minimum number of reactions that can represent the network and therefore they are all sparse realizations. In network E, the complex $2X_1 + X_2$ is isolated, i.e. it does not have incoming or outgoing reactions, therefore that complex is not shown.

Network F shows dense realizations, which contain all the possible reactions. The dense realization is *structurally unique* and in this case it contains all the possible edges. However, there are continuum many dense realizations, with the following conditions on the weights: $l_{11} < \frac{3k_{11}}{2}$, $l_{12} < \frac{3k_{12}}{2}$, $l_{21} < k_{21}$, $l_{22} < k_{22}$, $l_{31} > 0$, $l_{32} > 0$. These conditions guarantee that the reaction rate coefficients are positive and matrix A_{k}^F is a proper generalized Kirchhoff matrix.

Further note that the networks A, B, C and D are neither reversible nor weakly reversible, but networks E and F are reversible realizations. ■

3.2.3 Optimization methods for the computation of realizations with preferred properties

In this section, we consider the computation of dynamically equivalent realizations for kinetic systems. We are seeking for special realizations, which have some preferred properties, such as a dense, sparse or reversible graph structure. This task is formulated as a mixed-integer linear optimization problem [76].

A general mixed integer linear programming problem (MILP) can be stated as

$$\begin{aligned} \underset{x}{\text{minimize}}\ & c^T x && (3.14)\\ \text{subject to}\ & A^{\text{eq}} x = b^{\text{eq}} && (3.15)\\ & B^{\text{ineq}} x \leq b^{\text{ineq}} && (3.16)\\ & x^{\text{lb}} \leq x \leq x^{\text{ub}} && (3.17)\\ & x_j \text{ is integer for some } j \in \{1, \dots k\} && (3.18) \end{aligned}$$

where x is the k dimensional decision vector. Some elements of the decision vector are real variables, while the others can take only integer values. When there is no integer variables included in the decision vector, the problem is called Linear Programming (LP) problem. The optimization problem is constituted by a linear objective function (3.14), linear equality constraints (3.15), linear inequality constraints (3.16) and bound constraints (3.17).

There are many recent results on the computation of equivalent realizations for MAL-CRNs based on MILP problems. For example, in [32] the optimization problem for the computation of sparse and dense realizations are stated as mixed integer linear optimization problem, which can be efficiently solved even for hundreds of chemical species. In [33] the procedure is adapted for finding complex and detailed balance realizations. In [54] the problem of finding equivalent realizations with minimum or maximum number of complexes is considered. In this section we show, how the MILP procedure can be adapted for the biochemical reaction network case.

Given a realization invariant coefficient matrix $M \in \mathbb{R}^{m\times\kappa}$, the set of complexes $Y \in \mathbb{R}^{n\times m}$ and the number of reaction kinetics in each complex $(d_1, \dots d_m)$, such that the total number of kinetics $\kappa = \sum_{i=1}^{m} d_i$. The goal is to find valid generalized Kirchhoff matrix or matrices (1.6), which fulfils the following matrix equation:

$$M = Y A_{\text{k}}. \tag{3.19}$$

Comparing to the existing optimization methods in [32, 33] and [54], the main difference here is the properties of the matrix A_{k}. In the MAL-CRN case A_{k} is a square column conservation matrix with negative diagonal elements. But A_{k} is typically rectangular in the bio-CRN case, and the location of the negative elements depends on the number of kinetics in each complex.

The entries of the A_{k} are written in the following way utilizing the index variable $z_i = \sum_{k=1}^{i-1} d_k$, for $i = 1 \dots m$ ($z_1 = 0$ and $z_m + d_m = \kappa$) as in (2.12):

$$A_{\text{k}}(a) = \begin{pmatrix} -a_{1,1} & \dots & -a_{1,d_1} & a_{1,z_2+1} & \dots & a_{1,z_2+1} & \dots & a_{1,z_m+d_m} \\ a_{2,1} & \dots & a_{2,d_1} & -a_{2,z_2+1} & \dots & -a_{2,z_2+d_2} & \dots & a_{2,z_m+d_m} \\ & & & & \vdots & & & \\ a_{m-1,1} & \dots & & & & & & a_{m-1,z_m+d_m} \\ a_{m,1} & \dots & & & & & & -a_{m,z_m+d_m} \end{pmatrix}. \tag{3.20}$$

With this explicit notation of the negative elements we can restrict the $m \times \kappa$ decision variables (a_{ij}) of the optimization problem to the non-negative orthant.

Column conservation constraint. The column conservation property of A_{k} can be expressed as κ number of equations as

$$1_m^T A_{\text{k},(\cdot,i)} = 0 \quad \text{for } i = 1 \dots \kappa\ , \tag{3.21}$$

where 1_m is the m dimensional one vector and $A_{(\cdot,i)}$ is the ith column of the matrix.

Counting the number of reactions. The non-zero elements of the Kirchhoff matrix, which do not have the negative sign, i.e. a_{i,z_j+l} for $i = 1 \dots m$, $j = 1 \dots m$, $l = 1 \dots d_i$, $i \neq j$, define the reactions in the network. Let us introduce $(m-1) \times \kappa$ binary decision variables w_{ij} for each non-negative entry of the matrix A_{k}. A binary variable equals to 1 if and only if the corresponding entry of matrix A_k is larger than zero (in practice, larger than a small threshold value ϵ):

$$a_{ij} \geq \epsilon \iff w_{ij} = 1 \quad \text{for} \quad (i,j) \in I_p \ , \tag{3.22}$$

where $I_p = \{(i,j) \mid i = 1 \dots, m;\ j = 1 \dots \kappa;\ j \neq z_i + 1, \dots z_i + d_i\}$, i.e. it is the set of index pairs of the non-negative elements of A_k in (3.20). Furthermore, let a^{ub} be a practical upper bound for the elements of the matrix:

$$0 \leq a_{ij} \leq a^{\mathrm{ub}} \text{ for } (i,j) \in I_p \ . \tag{3.23}$$

Then the relation (3.22) can be expressed in terms of linear inequalities in the following way:

$$0 \leq a_{ij} - \epsilon w_{ij} \text{ for } (i,j) \in I_p \tag{3.24}$$

$$0 \leq -a_{ij} + a^{\mathrm{ub}} w_{ij} \text{ for } (i,j) \in I_p \tag{3.25}$$

Finding sparse realization. Combining the above linear constraints (3.21), (3.23), (3.24) and (3.25) with an objective function that counts the number of reactions we can formulate a mixed integer linear optimization problem that finds a sparse realization:

$$\underset{w,a}{\text{minimize}} \sum_{i=1}^{m} \sum_{j=1}^{\kappa} w_{ij} \tag{3.26}$$

$$\text{subject to: } M - Y A_{\mathrm{k}}(a) = 0 \tag{3.27}$$

$$1_m^T A_{(\cdot,i)} = 0 \text{ for } i = 1 \dots \kappa \tag{3.28}$$

$$0 \leq a_{ij} \leq a^{\mathrm{ub}} \text{ for } (i,j) \in I_p \tag{3.29}$$

$$0 \leq a_{ij} - \epsilon w_{ij} \text{ for } (i,j) \in I_p \tag{3.30}$$

$$0 \leq -a_{ij} + a^{\mathrm{ub}} w_{ij} \text{ for } (i,j) \in I_p \tag{3.31}$$

$$w_{ij} \text{ are binary variables, for } (i,j) \in I_p \tag{3.32}$$

where (3.26) defines the goal of finding the sparse realization by minimizing the number of edges, (3.27) defines the constraints ensuring the dynamic equivalence, (3.28) ensures the column conservation property of the Kirchhoff matrix, (3.29) bounds the principal reaction rate coefficients, and (3.30) - (3.32) define the binary variables associated to the reactions, i.e. $w_{ij} = 1 \iff a_{ij} > \epsilon$.

Finding dense realization. The realization that contains the most number of edges can be found by solving the optimization problem (3.26)-(3.32), but changing the sign of the objective function (3.26) to negative.

Remarks.

- One can provide further equality constrains to incorporate a-priory knowledge about existing reactions or to exclude possible reactions.

- Although the dense realization is structurally unique, i.e. the positions of zero and non-zero elements of A_k are well determined, the numerical values of the entries may vary depending on the numerical algorithm used to solve (3.26) - (3.32).

Reversible realization. The network is reversible if whenever there is an edge from complex C_i to complex C_j, there is also an edge from C_j to C_i. Using the block structure of A_k in (2.10), this means the following: whenever there is a nonzero element in the j-th row in $A_k^{(i)}$ (there is a reaction from complex C_i to C_j) there must be a positive element in the i-th row in $A_k^{(j)}$.

Formally, the corresponding condition can be expressed in terms of the parametrized Kirchhoff matrix (3.20) using the index notation (2.12) as

$$\sum_{l=z_j+1}^{z_j+d_j} a_{il} \geq \epsilon_2 \Leftrightarrow \sum_{l=z_i+1}^{z_i+d_i} a_{jl} \geq \epsilon_2 \text{ for } i = 1 \dots m, j = i+1 \dots m \ . \tag{3.33}$$

Here $\epsilon_2 > \epsilon$ is a small suitable threshold value. We further define $\frac{m\times(m-1)}{2}$ binary variables δ_{ij}, $i, j = 1 \dots m,\ i > j$. These binary variables take the value 1 if the corresponding interaction between the complexes is reversible and 0 otherwise. Then, the logical relationship (3.33) can be expressed in terms of inequality constraints as

$$\begin{aligned}
0 \leq & (\epsilon_2 - \epsilon) - \sum_{l=z_j+1}^{z_j+d_j} a_{il} + (a^{ub} - \epsilon_2)\delta_{ij} \\
0 \leq & (\epsilon_2 - \epsilon) - \sum_{l=z_i+1}^{z_i+d_i} a_{jl} + (a^{ub} - \epsilon_2)\delta_{ij} \\
0 \leq & \sum_{l=z_j+1}^{z_j+d_j} a_{il} - \epsilon_2\delta_{ij} \\
0 \leq & \sum_{l=z_i+1}^{z_i+d_i} a_{jl} - \epsilon_2\delta_{ij} \quad \text{for} \quad i = 1, \dots m,\ j = 2, \dots m, j > i.
\end{aligned} \tag{3.34}$$

Note that the condition of reversible realization is formulated as a set of constraints. Using these constraints together with (3.26)-(3.32) results in a MILP problem, which finds a sparse reversible realization. Similarly, by changing the sign of the objective function, we can find dense reversible realization.

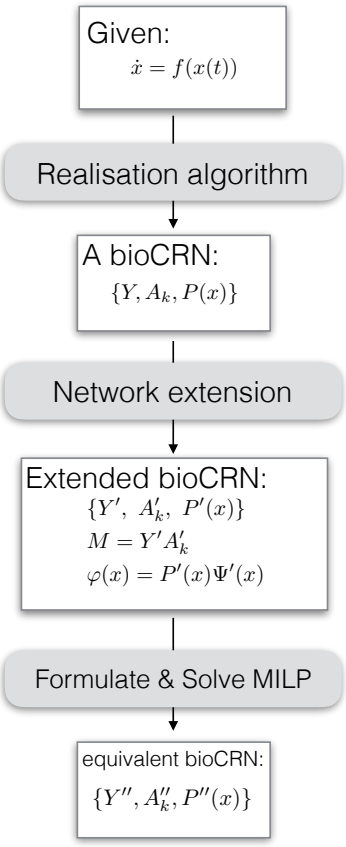


Figure 3.3: Main steps in finding the dynamically equivalent realization of a kinetic system.

Summary The whole procedure of finding dynamically equivalent realizations are summarized in Figure 3.3. First, the realization algorithm is used to form a biochemical reaction network from the kinetic equations. The algorithm generates a set of reaction kinetics and Kirchhoff matrix. In the next step, the vector of reaction kinetics can be (optionally) extended. In that case we have to include zero columns in the Kirchhoff matrix. Finally, the linear optimization problem is formulated, which can aim for finding

the sparse, dense or reversible equivalent realization.

3.3 Conjugated biochemical reaction networks

Johnston and Siegel [58] introduced the notion of linearly conjugated networks for mass action systems, which can be seen as an extension of dynamic equivalence. Two networks are said to be linearly conjugated if the solutions of their dynamic equations can be transformed into each other by a positive linear transformation, i.e. changing the units of the states and a possible reordering of the state variables. Since the dynamics of the two networks are only different by some scaling factors, they share qualitative properties. Thus, if there is a network with known properties, for example the known number of equilibrium points and the stability properties of these equilibria, these properties are transferred to all of its linearly conjugated networks.

The above formalism of the biochemical reaction networks let us extend the linear conjugacy concept to biochemical reaction networks, i.e. networks which do not solely obey to the mass action rule.

Definition 1. *Given a set of dynamic equations by the coefficient matrix M and by the kinetic vector $\varphi(x)$, such that $\dot{x} = M\varphi(x)$. The solution of the equations with initial conditions $x(t_0) = x_0$ is denoted by $x(t; x_0, t_0)$. Further given a bio-CRN $\Sigma = (Y, A_k, P)$ with solution $\hat{x}(t; \hat{x}_0, t_0)$. This bio-CRN is called a* linearly conjugated realization *of the dynamics if there is a linear, bijective function $h(.) : \mathbb{R}^n_+ \to \mathbb{R}^n_+$ such that*

$$h(x(t; x_0, t_0)) = \hat{x}(t; h(x_0), t_0) \quad \forall x_0 \in \mathbb{R}^n_+ \ . \tag{3.35}$$

Lemma 3.3.1. *[58, Lemma 3.1] A linear, bijective function $h(.) : \mathbb{R}^n_+ \to \mathbb{R}^n_+$ is at most a positive scaling and possible reordering of the state variables.*

Due to this lemma, we can conclude that the original kinetic system (3.7) and the linearly conjugated bio-CRN share local stability conditions.

It is easy to show that if two networks are linearly conjugated realizations of a dynamic system, then there is a linear mapping transforming their solutions to each other.

Lemma 3.3.2. *Given two biochemical reaction networks $\Sigma' = (Y', A'_k, P')$ and $\Sigma'' = (Y'', A''_k, P'')$, which are linearly conjugated realizations of a dynamic system with solution $x(t; x_0, t_0)$ with linear bijective mappings h' and h'' respectively. Then Σ'' is linearly conjugated to Σ' with linear mapping $h = h'' \circ h'^{-1}$ and thus*

$$h(x'(t; x'_0, t_0)) = x''(t; h(x'_0), t_0) \quad \forall x'_0 \in \mathbb{R}^n_+, \tag{3.36}$$

where $x'(t; x'_0, t_0)$ and $x''(t; x''_0, t_0)$ are the solutions of Σ' and Σ'' respectively.

Proof. First, Σ' is a linearly conjugated realization, so

$$h'(x(t; x_0, t_0)) = x'(t; h'(x_0), t_0) \quad \forall x_0 \in \mathbb{R}^n_+,$$

and since h' is bijective, there is an inverse such that

$$x(t; h'^{-1}(x'_0), t_0)) = h'^{-1}(x'(t; x'_0, t_0)) \quad \forall x'_0 \in \mathbb{R}^n_+.$$

Now using that Σ'' is also a linearly conjugated realization

$$h''(x(t; x_0, t_0)) = x''(t; h''(x_0), t_0)$$

leads to

$$x''(t; h''(h'^{-1}(x'_0)), t_0) = h''(h'^{-1}(x'(t; x'_0, t_0))) \quad \forall x'_0 \in \mathbb{R}^n_+.$$

□

Similarly to [58, Lemma 3.2] we can also conclude that the two linearly conjugated networks share the same local stability conditions. Lemma 3.3.1 showed that $h(.)$ can be at most a positive scaling and reordering of the state variables. From now one, we set aside the option to reorder the states, and we will focus on the special case, in which $h(.)$ can be written as $h(x) = Tx$, where T is a positive diagonal matrix.

We need the following three short lemmas to proceed to the linear conjugated biochemical reaction networks. The first lemma considers the positive linear scaling of a generalized Kirchhoff matrix.

Lemma 3.3.3. *Let $A_k \in \mathbb{R}^{m\times\kappa}$ be a generalized Kirchhoff matrix corresponding to a network of m complexes and d_1, $d_2 \dots d_m$ kinetics ($\kappa = \sum_{i=1}^{m} d_i$) of the respective complexes. Further, let $\bar{H}$ be a positive diagonal matrix $\bar{H} = diag(c)$, where $c \in \mathbb{R}^{\kappa}_+$. Then, $A'_k = A_k\bar{H}$ is also a generalized Kirchhoff matrix with the same structure (same locations of the non-zero elements) as A_k.*

Proof. The i-th column of A_k is multiplied by c_i, which do not alter the location of the zero and non-zero elements of the column and the column-sum remains zero. □

Since the generalized Kirchhoff matrix has a strong relationship to the reaction graph, this lemma means that a linear scaling can change the weights on the reaction graph, but neither can an existing reaction disappear, nor can a non-existing reaction appear. The following lemma considers the linear transformation of the variables in the kinetic weighting function $P(.)$ as defined in (2.13).

Lemma 3.3.4. *Let $P : \mathbb{R}^n \to \mathbb{R}^{\kappa\times m}$ be a kinetic weighting function corresponding to a network of m complexes and d_1, $d_2 \dots d_m$ kinetics ($\kappa = \sum_{i=1}^{m} d_i$) corresponding to the complexes. Further, let T be a positive diagonal matrix $T = diag(c)$, where $c \in \mathbb{R}^n_+$. Then $\hat{P}(x) = P(Tx)$ is also a kinetic weighting function, further, the non-zero elements of matrix $\hat{P}(x)$ locate in the same places as in $P(x)$, thus they are structurally similar.*

Proof. The invariance of the structure (the position of the non-zero entries) under a linear scaling is straightforward from the construction of P, see (2.13). However, the coefficients of the polynomials in the denominators are transformed. Recall that, the l-th element of the i-th block in (2.13) is $\left[P^{(i)}\right]_l(x) = \frac{1}{D_{il}(x)}$, where $D_{il}(x)$ is a polynomial with positive coefficients ($\alpha_{m_1,m_2,\dots m_n} > 0$) and leading coefficient 1, i.e. $D_{il}(x) = 1 + \sum \alpha_{m_1,m_2,\dots m_n} x_1^{m_1} x_2^{m_2} \dots x_n^{m_n}$. The linear scaling alters the elements as

$$\begin{aligned} D_{il}(Tx) &= 1 + \sum \alpha_{m_1,m_2,\dots m_n} c_1^{m_1} x_1^{m_1} c_2^{m_2} x_2^{m_2} \dots c_n^{m_n} x_n^{m_n} \\ &= 1 + \sum \hat{\alpha}_{m_1,m_2,\dots m_n} x_1^{m_1} x_2^{m_2} \dots x_n^{m_n} \\ &= \hat{D}_{il}(x), \end{aligned}$$

where the new coefficients are $\hat{\alpha}_{m_1,m_2,\dots m_n} = \alpha_{m_1,m_2,\dots m_n} c_1^{m_1} c_2^{m_2} \dots c_n^{m_n}$. Since $\hat{D}_{il}(x)$ is also a positive polynomial with leading coefficient 1, $\left[\hat{P}^i\right]_l(x) = \frac{1}{\hat{D}_{il}(x)}$ fulfils the requirements to be a proper weighting matrix element. □

We can see that the linear transformation of the variables of the kinetic weighting function means a transformation in the constant parameters of the biochemical reaction rates. We can also note that the originally zero elements in P remain zero, therefore the number of kinetics of the complexes remains unchanged under the transformation. Further, in case of MAL kinetics, i.e. when the denominator terms are constant ones, the linear transformation does not transform the kinetics.

The range of the kinetic weighting function is a subset of the $\mathbb{R}^{\kappa\times m}$ matrices with a special block structure. In the third lemma we construct a diagonal matrix, which can eliminate the effect of a scaling of such matrices.

Lemma 3.3.5. *Let $P : \mathbb{R}^n \to \mathbb{R}^{\kappa\times m}$ be a kinetic weighting function corresponding to a network of m complexes and $d_1, d_2 \dots d_m$ kinetics ($\kappa = \sum_{i=1}^{m} d_i$). Let $\bar{H}$ be a positive diagonal matrix $\bar{H} = diag(c)$, where $c \in \mathbb{R}^m_+$ and let the block-diagonal matrix $\hat{H}$ constructed as*

$$\hat{H} = \begin{pmatrix} \left[diag(\frac{1}{c_1})\right]_{d_1} & 0_{d_1\times d_2} & \dots \\ 0_{d_2\times d_1} & \left[diag(\frac{1}{c_2})\right]_{d_2} & \dots \\ \vdots & \ddots & \dots \\ 0_{d_m\times d_1} & \dots & \left[diag(\frac{1}{c_m})\right]_{d_m} \end{pmatrix}, \tag{3.37}$$

where $\left[diag(\frac{1}{c_i})\right]_{d_i}$ denotes a diagonal block of size $d_i \times d_i$, which contains the constant $\frac{1}{c_i}$ in its diagonal, and $0_{d_i\times d_j}$ denotes a block of zeros of size $d_i \times d_j$. As above, d_i is the number of kinetics in complex C_i. Then $P(x) = \hat{H}P(x)\bar{H}$ for all $x \in \mathbb{R}^n_+$.

Proof. Multiplying the P in (2.13) from the right by a diagonal matrix $\bar{H}$ scales each block: $P^{(i)} \to c_i P^{(i)}$ for $i = 1 \dots m$. It is easy to see, that the above defined $\hat{H}$ removes the scaling factors of the blocks. □

Therefore, a linear scaling from the right side of the weighting matrix $P(x)$ can be propagated to the left side as $P(x)\bar{H} = \hat{H}^{-1}\hat{H}P(x)\bar{H} = \hat{H}^{-1}P(x)$.

The following theorem is the extension for the biochemical reaction networks of the linear conjugacy theorem presented by [77, Theorem 2]. The original statement of the theorem for mass action systems can be found in [58].

Theorem 3.3.6. *Given a kinetic system (3.7) by the coefficient matrix M and by the kinetic vector $\varphi(x)$. Suppose that there is an A_b generalized Kirchhoff matrix, and a positive diagonal matrix $T = \mathrm{diag}(c)$, $c \in \mathbb{R}^n_+$ such that*

$$M = T \cdot Y \cdot A_b \tag{3.38}$$

and the kinetic vector can be decomposed to the kinetic weighting matrix and monomial vector function as

$$\varphi(x) = P(x)\Psi(x). \tag{3.39}$$

Then $\Sigma = (Y, A'_k, P')$ is a linearly conjugated realization of the kinetic system with generalized Kirchhoff matrix

$$A'_k = A_b H^{-1} \tag{3.40}$$

and kinetic weighting function

$$P'(x) = P(Tx), \tag{3.41}$$

where $H \in \mathbb{R}^{\kappa\times\kappa}$ is a positive, block-diagonal (invertible) matrix as

$$H = \begin{pmatrix} \left[\mathrm{diag}(\frac{1}{\Psi_1(c)})\right]_{d_1} & 0_{d_1\times d_2} & \dots \\ 0_{d_2\times d_1} & \left[\mathrm{diag}(\frac{1}{\Psi_2(c)})\right]_{d_2} & \dots \\ \vdots & \ddots & \dots \\ 0_{d_m\times d_1} & \dots & \left[\mathrm{diag}(\frac{1}{\Psi_m(c)})\right]_{d_m} \end{pmatrix}, \tag{3.42}$$

where $\left[diag(\frac{1}{\Psi_i(c)})\right]_{d_i}$ denotes a diagonal block of size $d_i \times d_i$, which contains the constant $\frac{1}{\Psi_i(c)}$ in its diagonal and $0_{d_i\times d_j}$ denotes a block of zeros of size $d_i \times d_j$ and c is the diagonal elements of T.

Proof. Let $x(t; x_0, t_0)$ be the solution for an initial condition $x(t_0) = x_0$ of the kinetic system. Further, let the linearly conjugated solution $\hat{x}(t; \hat{x}_0, t_0) = T^{-1}x(t; x_0, t_0)$ and thus $x(t; x_0, t_0) = T\hat{x}(t; \hat{x}_0, t_0)$ and for the initial condition $\hat{x}_0 = \hat{x}(t_0) = T^{-1}x(t_0) = T^{-1}x_0$. Using the dynamic equations (3.7) we can write

$$\begin{aligned} \frac{d\hat{x}(t; \hat{x}_0, t_0)}{dt} &= T^{-1}\frac{dx(t; x_0, t_0)}{dt} \\ &= T^{-1}M\varphi(x(t; x_0, t_0)) \\ &= T^{-1}MP(x(t; x_0, t_0))\Psi(x(t; x_0, t_0)) \\ &= T^{-1}MP(T\hat{x}(t; \hat{x}_0, t_0))\Psi(T\hat{x}(t; \hat{x}_0, t_0)) \end{aligned} \tag{3.43}$$

Note the following relationship

$$\begin{aligned} [\Psi(T\hat{x})]_i &= \prod_{k=1}^{n}(c_k\hat{x}_k)^{Y_{ki}} \\ &= \prod_{k=1}^{n} c_k^{Y_{ki}} \prod_{k=1}^{n} \hat{x}_k^{Y_{ki}} \\ &= [\Psi(c)]_i[\Psi(T\hat{x}]_i \quad \text{for} \quad i = 1, 2, \dots m. \end{aligned}$$

Using the notation $\Psi_T = \mathrm{diag}(\Psi(c))$ leads to $\Psi(T\hat{x}) = \Psi_T\Psi(T\hat{x})$. Insert this and the κ dimensional unity matrix $I_\kappa = H^{-1}H$ into (3.43), where H is given in (3.42). The result reads as

$$\frac{d\hat{x}(t; \hat{x}_0, t_0)}{dt} = T^{-1}MH^{-1}HP(T\hat{x}(t; \hat{x}_0, t_0))\Psi_T\Psi(\hat{x}(t; \hat{x}_0, t_0)).$$

Lemma 3.3.5 brings $HP(T\hat{x})\Psi_T = P(T\hat{x})$. Further, according to Lemma 3.3.4 $P(T\hat{x}) = P'(\hat{x})$ is a kinetic weighting function. Now, using (3.38) results in

$$\begin{aligned} \frac{d\hat{x}(t; \hat{x}_0, t_0)}{dt} &= T^{-1}TYA_bH^{-1}P'(\hat{x}(t; \hat{x}_0, t_0))\Psi(\hat{x}(t; \hat{x}_0, t_0)) \\ &= YA_bH^{-1}P'(x(t; \hat{x}_0, t_0))\Psi(\hat{x}(t; \hat{x}_0, t_0)) \\ &= YA'_kP'(\hat{x}(t; \hat{x}_0, t_0))\Psi(\hat{x}(t; \hat{x}_0, t_0)). \end{aligned}$$

Thus $\hat{x}(t; \hat{x}_0, t_0)$ is the solution of the dynamics corresponding to the bioCRN $\Sigma = (Y, A'_k, P')$.

The matrix A'_k (3.40) is a generalized Kirchhoff matrix as follows from Lemma 3.3.3 and $P'(x)$ (3.41) is a kinetic weighting function as follows from Lemma 3.3.4 and Lemma 3.3.5.

Since the solutions of the kinetic system ($\dot{x} = M\varphi(x)$) can be transformed by the linear transformation T into the solutions of a bio-CRN Σ, Σ is a linearly conjugated realization to that kinetic system. □

If two networks are linearly conjugated realizations of the same kinetic system, then they are called linearly conjugated networks.

3.3.1 Computing linearly conjugated realizations for biochemical systems

The computation of linearly conjugated networks can be stated in a very similar framework, that was used for the computation of dynamically equivalent realizations in Subsection 3.2.3.

Given a kinetic system in the form of (3.7), i.e. given the coefficient matrix M and kinetic vector φ, Algorithm (1) can be used to obtain a (dynamically equivalent) realization of the dynamics. This procedure determines a possible set of complexes and reaction kinetics, therefore we can fix the complex composition matrix Y and the kinetic weighting function P. The complex composition matrix also determines the monomial function Ψ according to (2.14) and the number of kinetics corresponding to each complex partially determines the structure the matrix A_k, i.e. its size and the positions of the negative column-sum terms.

In order to find a linearly conjugated realization, according to Theorem 3.3.6, we have to find the generalized Kirchhoff matrix A_b and a diagonal matrix $T = \mathrm{diag}(c)$ such that Equation (3.38) is fulfilled. In the optimization framework, this is a non-linear equality constraints in the optimization variables (the entries of A_b and diagonal of T). However, it is easy to transform (3.38) to a linear equality constraint by rewriting as

$$T^{-1}M - YA_b = 0 \ . \tag{3.44}$$

Thus, instead of finding the diagonal matrix $T = \mathrm{diag}(c)$, we are finding its inverse $T^{-1} \doteq \tilde{T} = \mathrm{diag}(\tilde{c})$, which always exists because all the elements of vector c are strictly positive.

The new equality constraint (3.44) can be arbitrarily combined with the goal of finding sparse or dense realization. For example, a mixed integer optimization problem, which finds a sparse A_b and $\tilde{T}$ can be written as

$$\underset{w,a,\tilde{c}}{\text{minimize}} \sum_{i=1}^{m}\sum_{j=1}^{\kappa} w_{ij} \tag{3.45}$$

$$\text{subject to: } \tilde{T}(\tilde{c})M - YA_b(a) = 0 \tag{3.46}$$

$$1_m^T A_{b,(\cdot,i)} = 0 \text{ for } i = 1 \dots \kappa \tag{3.47}$$

$$0 \le a_{ij} \le a^{\mathrm{ub}} \text{ for } (i,j) \in I_p \tag{3.48}$$

$$0 \le a_{ij} - \epsilon w_{ij} \text{ for } (i,j) \in I_p \tag{3.49}$$

$$0 \le -a_{ij} + a^{\mathrm{ub}} w_{ij} \text{ for } (i,j) \in I_p \tag{3.50}$$

$$w_{ij} \text{ are binary variables, for } (i,j) \in I_p \tag{3.51}$$

$$\epsilon_2 \le c \le \tilde{c}^{\mathrm{ub}}. \tag{3.52}$$

As in Section 3.2.3, here we also assigned to each entry of A_b a binary variable w to count the number of reactions. Thus (3.45) defines the goal of finding the sparse realization by minimizing the number of reactions. Equation (3.46) defines the constraints ensuring the linear conjugacy. Here we parametrized the entries of A_b by the variables a_{ij}, similarly as we did in (3.20). Then, (3.47) ensures the column conservation property of the Kirchhoff matrix A_b, (3.29) bounds the entries of A_b, and (3.30) - (3.32) define the binary variables associated to the reactions, i.e. $w_{ij} = 1 \iff a_{ij} > \epsilon$. Finally, (3.52) bounds the diagonal elements of $\tilde{T}$.

After the optimization problem is solved by a numerical solver, we obtain A_b and $\tilde{T}$ and the linearly conjugated network is computed as follows. Inverting $\tilde{T}$ results in T and matrix H in (3.42) can be computed from the diagonal elements of T and monomial function Ψ. Finally, the generalized Kirchhoff matrix A'_k and kinetic weighting function $P'(x)$ of the linearly conjugated network are computed as (3.40) and (3.41), respectively. The linearly conjugated realization is given by the triplet (Y, A'_k, P').

Note that, as in the case of computing dynamically equivalent realizations, here one also can introduce the constraints (3.34) to ensure the reversibility of the graph structure.

3.3.2 Example

A simple example is given to illustrate the notions and tools proposed above and their use in determining linearly conjugated realizations of a given dynamic system.

Example 3-3

Consider the following dynamical ODE model adapted from [78]

$$\begin{aligned} \frac{dx_1}{dt} &= 0.05x_2 + 0.1x_1^2x_2 - \frac{0.2x_1^2}{1+2x_1} - \frac{x_1}{1+\frac{1}{2}x_1} \\ \frac{dx_2}{dt} &= 1 - 0.05x_2 - 0.1x_1^2x_2 + \frac{0.1x_1^2}{1+2x_1}. \end{aligned} \tag{3.53}$$

This set of dynamic equations gives rise to a stable limit cycle solution, as depicted in Figure 3.4 for zero initial condition.

A biochemical reaction network realization, which corresponds to the above dynamic equations, can be constructed using Algorithm 1. However, we took the realization from the [78] and adapted the kinetics of the network manually. First, we assume the complex set $\mathcal{C} = \{0,\ X_1,\ X_2,\ 2X_1,\ 2X_1 + X_2,\ 3X_1\}$, where 0 denotes the so-called zero complex, which describes the interaction with the environment. Then, for each complex $C_1, \dots C_6$, a kinetic rate function is defined as $g_{11} = 1$, $g_{21} = \frac{x_1}{1+x_1/2}$, $g_{31} = x_2$, $g_{41} = \frac{x_1^2}{1+2x_1}$, $g_{51} = x_1^2x_2$ and $g_{61} = x_1^3$, respectively. This means a constant inflow from the environment for the complexes C_1; mass action rate law for complex C_3 and C_5, and Michaelis-Menten type rate law for the complex C_2 and C_4. Then the matrix representation of the realization reads as

$$Y = \begin{pmatrix} 0 & 1 & 0 & 2 & 2 & 3 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}, \qquad A_k = \begin{pmatrix} -k_1 & k_2 & 0 & 0 & 0 & 0 \\ 0 & -k_2 & k_3 & 0 & 0 & 0 \\ k_1 & 0 & -k_3 & k_4 & 0 & 0 \\ 0 & 0 & 0 & -k_4 & 0 & 0 \\ 0 & 0 & 0 & 0 & -k_5 & 0 \\ 0 & 0 & 0 & 0 & k_5 & 0 \end{pmatrix}$$

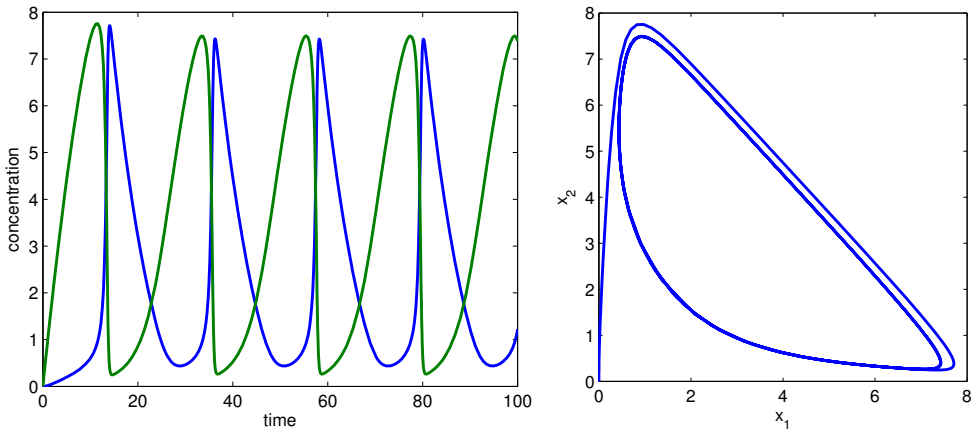


Figure 3.4: Example 3-3. Solution of the dynamic equations (3.53) with zero initial condition in time domain (blue line for x_1 and green line for x_2) and in phase space.

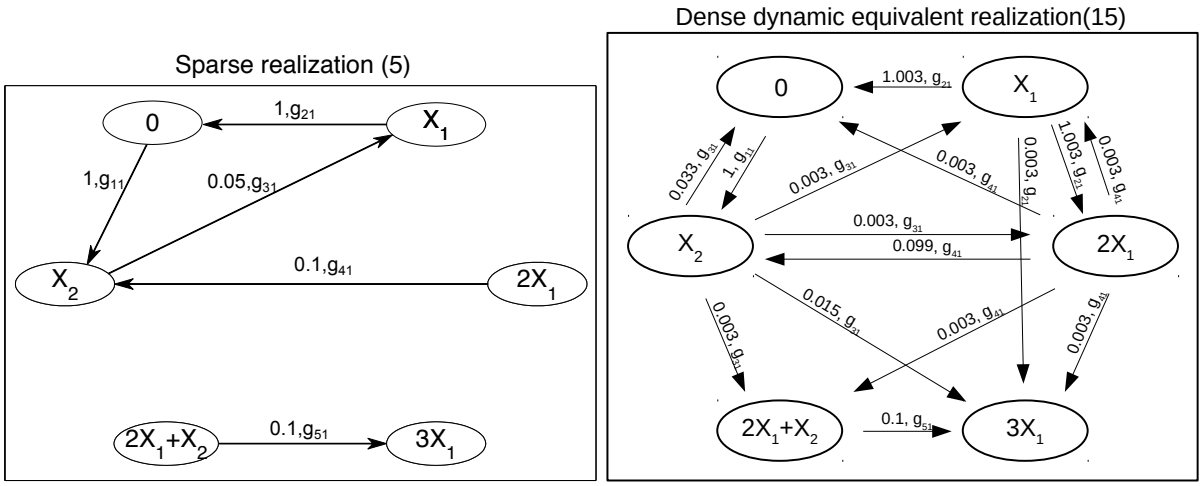
$$P(x) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{1+x_1/2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{1+2x_1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}, \qquad \Psi(x) = \begin{pmatrix} 1 \\ x_1 \\ x_2 \\ x_1^2 \\ x_1^2 x_2 \\ x_1^3 \end{pmatrix}$$

where the parameter values are $k_1 = 1$, $k_2 = 1$, $k_3 = 0.05$, $k_4 = 0.1$ and $k_5 = 0.1$. We tried to find a sparser dynamically equivalent realization using the method described in the previous section by Equations (3.26)-(3.32), but we could not find a network with fewer number of reactions. Therefore this realization contains the minimal number of reactions and thus it is a sparse realization with 5 reactions. The reaction graph is shown in Figure 3.5a.

A dynamically equivalent dense realization can be found containing 15 reactions by solving Equations (3.26)-(3.32) with a negative factor in the objective (3.26). This network is depicted in Figure 3.5b, its generalized Kirchhoff matrix is

$$A_k^{\text{dense}} = \begin{pmatrix} -1.00 & 1.003 & 0.033 & 10^{-3} & 0 & 0 \\ 0 & -1.005 & 10^{-3} & 10^{-3} & 0 & 0 \\ 1.00 & 0 & -0.051 & 0.099 & 0 & 0 \\ 0 & 10^{-3} & 10^{-3} & -0.103 & 0 & 0 \\ 0 & 0 & 10^{-3} & 10^{-3} & -0.10 & 0 \\ 0 & 10^{-3} & 0.015 & 10^{-3} & 0.10 & 0 \end{pmatrix}.$$

We showed above that the dense dynamically equivalent realization has a unique structure and further, any dynamically equivalent realization contains a *subset* of the reactions of the dense realization. This means that the reactions which are not included in the dense realization, cannot appear in any other dynamically equivalent realization.



(a) A sparse realization.

(b) A dense realization

Figure 3.5: Dynamically equivalent sparse and dense realizations for Example 3-3

Now we compute a dense linearly conjugated realization of the dynamic system (3.53). First, we solve the optimization problem (3.45)-(3.52), but with a negative sign in the objective function (3.45), which immediately results in the following intermediate Kirchhoff matrix and scaling matrix

$$A_b = \begin{pmatrix} -0.24 & 0.163 & 9.0 \cdot 10^{-3} & 10^{-3} & 10^{-3} & 0 \\ 0 & -0.165 & 10^{-3} & 10^{-3} & 10^{-3} & 0 \\ 0.24 & 0 & -0.013 & 0.015 & 10^{-3} & 0 \\ 0 & 10^{-3} & 10^{-3} & -0.027 & 10^{-3} & 0 \\ 0 & 0 & 10^{-3} & 9.000 \cdot 10^{-3} & -0.025 & 0 \\ 0 & 10^{-3} & 10^{-3} & 10^{-3} & 0.021 & 0 \end{pmatrix}$$

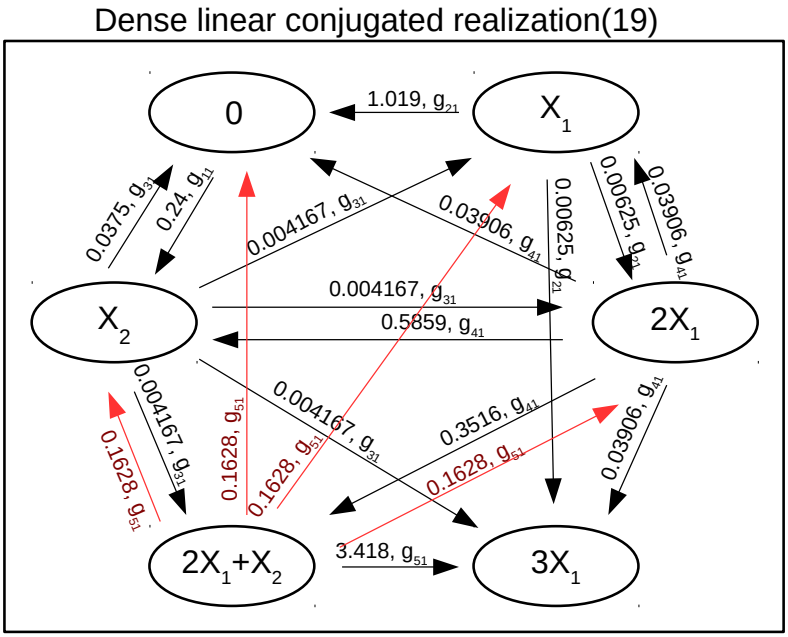
$$T(x)^{-1} = \begin{pmatrix} 0.16 & 0 \\ 0 & 0.24 \end{pmatrix}.$$

Then, following the procedure in Subsection 3.3.1, first we obtain T and then matrix H by (3.42), as

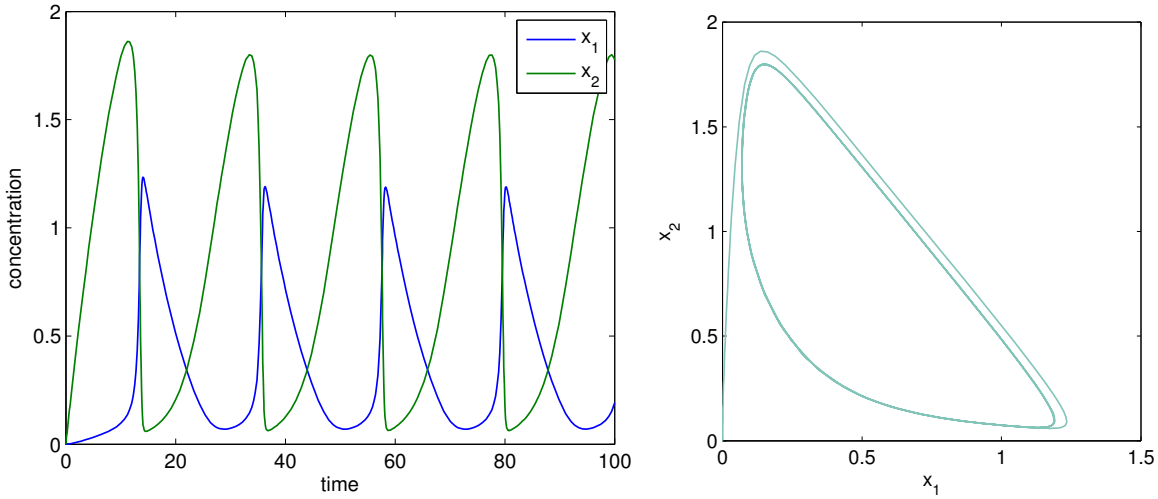
$$T = \begin{pmatrix} 6.25 & 0 \\ 0 & 4.1667 \end{pmatrix}, \qquad H = \mathrm{diag}([1,\ 0.16,\ 0.24,\ 0.0256,\ 6.144 \cdot 10^{-3},\ 4.096 \cdot 10^{-3}]^T).$$

Finally, Equations (3.40) and (3.41) result in the dense generalized Kirchhoff matrix of linearly conjugated network

$$A_k^{\mathrm{LC,dense}} = \begin{pmatrix} -0.2400 & 1.019 & 0.03750 & 0.03906 & 0.1628 & 0 \\ 0 & -1.031 & 4.167 \cdot 10^{-3} & 0.03906 & 0.1628 & 0 \\ 0.2400 & 0 & -0.05417 & 0.5859 & 0.1628 & 0 \\ 0 & 6.250 \cdot 10^{-3} & 4.167 \cdot 10^{-3} & -1.055 & 0.1628 & 0 \\ 0 & 0 & 4.167 \cdot 10^{-3} & 0.3516 & -4.069 & 0 \\ 0 & 6.250 \cdot 10^{-3} & 4.167 \cdot 10^{-3} & 0.03906 & 3.418 & 0 \end{pmatrix}$$



(a) A linearly conjugated dense realization



(b) Solution of the linearly conjugated dynamic equations

Figure 3.6: A linearly conjugated dense realization for Example 3-3 is depicted with its solution in time, and in the phase space. The highlighted reactions do not exist in the dense dynamic equivalent realization in Fig. 3.5b

and the transformed kinetic weighting function

$$P'(x) = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{1+6.25\cdot 0.5x_1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{1+6.25\cdot 2x_1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} .$$

The reaction graph of the linearly conjugated network $(Y, A_k^{\mathrm{LC,dense}}, P')$ is depicted in Figure 3.6a. Note that, we found 4 edges, i.e. 4 reactions, highlighted in red color, which do not appear in the dense dynamic equivalent realizations (see in Figure 3.5b). This means, that the reactions appearing in the linearly conjugated realizations are not necessarily form a subset of the reactions appearing in the dense dynamically equivalent realizations.

For the comparison of the dynamics of the original and the linearly conjugated network, the solutions corresponding to the zero initial condition is depicted in the time domain and in the phase space, see Figure 3.6b, where we can see the similarities with the original solutions in Figure 3.4. ■

3.4 Summary

In this chapter we considered first the realization of bio-CRNs for rational ordinary differential equations, i.e. finding a bio-CRN with the exact same dynamics as described by the given rational ordinary differential equations. After giving the necessary and sufficient conditions of realizability, which assured that a bio-CRN can be constructed, we proposed a realization algorithm (Algorithm 1). The algorithm takes the right hand side of the ODEs and finds the set of species, source and product complexes, and construct the reaction rate functions.

We found that the realization is not unique, but multiple bio-CRNs can be constructed to the dynamic equations. These networks are called dynamically equivalent realizations. We showed that, these networks can be structurally different, for example, the network with the most number of reactions is called the dense realization and a network with the lowest number of reactions is called a sparse realization. Further we showed that the dense realization is structurally unique and the graph of any other realization is a sub-graph of the dense realization. Finally, we proposed a mixed integer linear programming framework to find dynamically equivalent realizations of a kinetic system with preferred properties.

In the last part of this chapter the linear conjugacy theorem of the bio-CRNs was developed. A bio-CRN is linearly conjugated to a kinetic system if the dynamics of the bio-CRN can be mapped to the kinetic system by a linear transformation. Since the stability properties of the kinetic system remains unchanged under linear transformation, finding linearly conjugated networks can be helpful to analyse the stability conditions of non-linear systems.

Finally, we formulated linear mixed integer optimization problems to find linearly conjugated dense and sparse realizations. The tools and notions were illustrated by an example, where we showed, that linear conjugated realization may contain reactions, which do not present in the dense dynamically equivalent realization.

Chapter 4

Model reduction in biochemical reaction networks

In chemical and biochemical systems the exact chemical reactions which determine the system dynamics are often not known precisely. In these cases, mathematical models can be constructed which contain all the *possible* reactions. This approach can lead to very large and detailed reaction kinetic models, which are too detailed for dynamic analysis. Because of the huge number of species and/or chemical reactions present in the detailed reaction kinetic mechanism of these systems, the need has arisen for developing a simplified or reduced mechanisms that can accurately describe the dynamics of the system under some restricted circumstances (e.g. in isobaric or isothermal conditions).

The commonly applied approaches for obtaining simplified kinetic representations can be categorized into three main branches:

i. The use of engineering model simplification transformations [61], such as quasi-steady state or quasi-equilibrium assumptions [79].

ii. The use of general nonlinear model reduction techniques applied to reaction kinetic models, such as the balance truncation method applied to input-affine state space models [80], singular perturbation analysis [81] and method of invariant grids [82].

iii. The use of optimization methods for reducing the number of reacting species and reactions. This problem generally leads to a mixed integer nonlinear program (MINLP) problem, see e.g. [83], that present computational complexity challenges in realistic problem sizes.

The simplest models within the class of reaction kinetic systems form the sub-class of reaction kinetic networks that obey the mass action law (see Section 1.2.1) [31, 51, 69].

An early approach to obtain CRNs with reduced complexity was based on principal component analysis of the parametric sensitivity matrix of the detailed kinetic model [84]. An improved version using also concentration sensitivities was developed for the case of gas phase reactions [85]. Finally, a sophisticated combined method for constructing the minimal suitable mechanism based on combined species and reaction selective inclusion and elimination has been proposed recently [86].

A systematic model reduction method that combines the sensitivity and principal component analysis methods with variable lumping is proposed in [87]. A simultaneous adjustment of the structure of CRN and its parameters such that the qualitative dynamical properties of the system are preserved during the reduction is the basis of the complexity reduction method presented in [88] in the application area of biochemical networks.

It is clear from the above that there are numerous methods available to reduce the complexity of CRNs that all use the specialities of the problem and the model to propose a feasible solution to the inherently computationally challenging problem. However, there are certain important features of the model reduction task that received little attention but they significantly influence the mathematical problem to be solved and the properties of the solutions. Firstly, measured data are usually not available about all of the species but only about a small subset of them (the key species). Secondly, the reaction kinetic parameters, most notably the reaction rate coefficients can only be determined with an approximately 10 % of accuracy even in the best cases, and the estimated value is strongly model structure dependent. This implies that the value of these parameters is not precise, therefore the re-estimation of them can significantly improve the fit between the output of the original detailed and the reduced model.

Therefore, the overall aim of our work was to propose a robust and numerically stable method for reducing the complexity, i.e. the number of reactions of a CRN, that is also able to re-estimate the reaction rate coefficients and produce a sub-set of the original detailed reaction kinetic scheme as a result, but with suitably adjusted coefficients.

Instead of the general mixed integer nonlinear optimization (MINLP) formulation of the problem, we construct a convex mixed integer quadratic problem (MIQP) formulation for which efficient solvers exist. The effect of the problem and model parameters, as well as the tuning parameters of the proposed algorithm is also investigated on the solution procedure and on the solution properties.

4.1 Problem formulation and notations

In this chapter, we consider deterministic chemical reaction networks with *mass action kinetics* assuming constant temperature and perfect mixing of the materials [30, 53]. From a mathematical point of view, the species concentrations under the above assumptions can be described by initial value problems of parametric ordinary differential equations (ODEs)

$$\frac{dx(t)}{dt} = f(x(t), k), \quad t \in [t_0, t_f], \tag{4.1}$$

$$x(t_0) = x_0, \tag{4.2}$$

where the right-hand side function $f : \mathbb{R}^n \times \mathbb{R}^q \to \mathbb{R}^n$ in Eq. (4.1) can be easily constructed from the list and parameters of chemical reactions in the MAL-CRN.

Note that $x(t)$ implicitly depends on the initial values x_0 and the kinetic parameters k. To stress these dependencies we may also use the notation $x(t, k, x_0)$ or only $x(t, k)$, especially when we are interested in the parametric sensitivities $\partial x(t, k)/\partial k$.

To derive the differential equations describing the time-evolution of chemical species concentrations, we will apply the classical description using the stoichiometric matrix [82]. Note that, this is a slightly different approach to derive the dynamic equations then that was used in Section 1.2.1.

Let us consider the MAL-CRN $\Sigma_{\text{MAL}} = (\mathcal{S}, \mathcal{C}, \mathcal{R})$, which contains q number of reactions, i.e. the cardinality of $\mathcal{R}$ is q. The reaction rates can be written as (1.2), however for notational convenience let us change the indexing of the reaction rates and the reaction rate coefficients by using single subscripts, i.e.

$$\{r_{ij} \mid i = 1 \dots m,\ j = 1 \dots m,\ r_{ij} \in \mathcal{R}\} \Longleftrightarrow \{r_l \mid l = 1 \dots q\} \tag{4.3}$$

and

$$\{k_{ij} \mid i = 1 \dots m,\ j = 1 \dots m,\ r_{ij} \in \mathcal{R}\} \Longleftrightarrow \{k_l \mid l = 1 \dots q\}\ . \tag{4.4}$$

Thus, from what follows let $r \in \mathbb{R}^q$ be the vector of reaction rates and $k \in \mathbb{R}^q_+$ the vector of the corresponding reaction rate parameters.

According to this notation, considering n species and q reactions, the species concentrations can be described as

$$\frac{dx}{dt} = N \cdot r(x), \tag{4.5}$$

where $x \in \mathbb{R}^n$ is the species concentration vector, $r \in \mathbb{R}^q$ is the vector of reaction rates and $N \in \mathbb{R}^{n \times q}$ is the stoichiometric matrix. N_{ij} is a real (most often integer) number denoting how many atoms/molecules of species X_i is produced or consumed in the jth reaction (where a positive value corresponds to overall production and a negative value to overall consumption). Note that, Eq. (4.5) is equivalent to (1.5).

4.2 Model Reduction in MAL-CRNs

4.2.1 Objective

Suppose that in a chemical reaction network we are only interested in variables corresponding to a few species. The concentrations of these species can be relevant because e.g. they are the measurable system output. Thus we want to reduce the network such that their concentrations remain unchanged. These species are named *important* in the following and we collect the indices of the associated variables into the set

$$\mathcal{I} := \{i_1, i_2, \dots, i_{n_\mathcal{I}}\}, \tag{4.6}$$

where $i_j \in \{1, 2, \dots, n\}$, $j = 1, 2, \dots, n_\mathcal{I}$ and $n_\mathcal{I}$ is the number of important species.

Additionally, we only care about the trajectories of the important species within a limited time horizon $[t_0, t_f]$. Recall that the reactions are in the form (1.2), then, the objective of the model reduction is to

1. reduce the number of reactions, i.e. set the corresponding rate coefficient k_i to zero, while keeping the concentration functions of the important species essentially unchanged on the time horizon $[t_0, t_f]$,

2. simultaneously adjust the remaining rate coefficients to improve the fit of the important species.

Note that, the initial conditions and time-horizon are user inputs and they fundamentally influence the results of the model reduction. This means, that for different initial conditions and different time-horizons might result in different reduced models. In the following, we will consider only one initial condition, however, our approach can be easily extended for multiple initial conditions, i.e. to find a reduced model which approximates the complex model trajectories started from a finite set of initial conditions. As we will see, although the construction of the optimization problem scales linearly with the number of defined initial conditions, it will not effect the computational complexity of the optimization procedure.

We have to note that, this objective is different from the classical model reduction approach, where the input-output behaviour of a complex model is approximated with

the reduced model. In the classical approach, the reduced order model is expected to approximate the original model independently from the initial conditions or time horizon.

4.2.2 The reduced model and its error

Due to the similar structure of the CRNs, where the reduced model is structurally a subset of the original one, the reduced model is totally specified by the reduced rate coefficient vector $\tilde{k} \in \mathbb{R}^q$. Reaction l is not present in the reduced CRN, if and only if $\tilde{k}_l = 0$ holds. The states $\tilde{x}(t)$ of the reduced model drop simply out as the solution of the initial value problem

$$\frac{d\tilde{x}}{dt} = f(\tilde{x}, \tilde{k}), \qquad \tilde{x}(t_0) = x_0, \tag{4.7}$$

where f and x_0 are the same as in Eqs. (4.1) and (4.2).

Of course, the *error* of the reduced model needs to be measured. This measure has to rely on the states $\tilde{x}(t)$ of the reduced model as well as on the states $x(t)$ of the original model and can be quantified by means of some functional $\Phi(\tilde{x}, x)$. We choose the least-square functional

$$\Phi(\tilde{x}, x) := \sum_{l=0}^{N} \sum_{i \in \mathcal{I}} {w_{il}}^2 \left(\tilde{x}_i(t_l) - x_i(t_l)\right)^2, \tag{4.8}$$

where $t_0 < t_1 < \cdots < t_N$ are selected time points, and w_{il}, $i \in \mathcal{I}$, $0 \leq l \leq N$ are some weights, e.g. to take into account the magnitude of $x_i(t_l)$. Actually, the same objective function for model reduction was used by Androulakis [83].

4.2.3 A straightforward MINLP

Note that $\tilde{x}(\cdot)$ and $x(\cdot)$ in Eq. (4.8) are totally determined by means of the corresponding parameter vectors $\tilde{k}$ and k, respectively. Hence, the nonlinear function

$$\phi(\tilde{k}, k) := \Phi(\tilde{x}(\cdot, \tilde{k}), x(\cdot, k)), \tag{4.9}$$

is well-defined since, for each $t \in [t_0, t_f]$, $\tilde{x}(t, \tilde{k})$ and $x(t, k)$ are uniquely determined by means of the corresponding initial value problems in Eqs. (4.1), (4.2) and Eq. (4.7).

As already mentioned in the preceding subsection, the number of non-zeros in $\tilde{k}$ equals the number of present reactions in the reduced model. Let NNZ denote the function which returns the number of non-zeros of a real vector, i.e.:

$$\mathrm{NNZ} : \mathbb{R}^q \to \{0, 1, \ldots, q\}, \quad \mathrm{NNZ}(v) = \#\{i \,:\, v_i \neq 0\} \, (v \in \mathbb{R}^q). \tag{4.10}$$

Obviously, the first objective of the model reduction is to find a reduced parameter vector $\tilde{k} \in \mathbb{R}^q$ which minimizes $\mathrm{NNZ}(\tilde{k})$, such that the model error $\phi(\tilde{k}, k)$ is small, say $\phi(\tilde{k}, k) < \delta$, where $\delta > 0$ is the user-specified error tolerance.

In the terms of mathematical optimization, according to Androulakis [83], we want to solve the mixed-integer nonlinear program (MINLP)

$$\underset{\tilde{k} \in [\underline{k}, \overline{k}]}{\text{minimize}} \quad \mathrm{NNZ}(\tilde{k}) \tag{4.11}$$

$$\text{subject to} \quad \phi(\tilde{k}, k) \leq \delta. \tag{4.12}$$

where k is fixed to the original values, and $\underline{k}, \overline{k} \in \mathbb{R}^q$ $(0 \leq \underline{k} \leq \overline{k})$ are the lower and upper bounds on $\tilde{k}$. The simplicity of the formulation is appealing. We have a simple linear objective function in Eq. (4.11) subject to a single nonlinear and nonconvex constraint in Eq. (4.12). In general, to find the global optimum of this MINLP, a global numerical MINLP solver has to be applied for the solution. In this context, the key problem is the evaluation of the constraint in Eq. (4.12) which requires the integration of the initial value problem (4.7). This may be very time consuming, especially when the MINLP solver additionally requires first- and second derivatives.

The situation would be much better, if we could approximate the non-convex MINLP by an optimization problem class, which can be solved more easily. This is exactly what we will do in the following: we will approximate the MINLP by a finite sequence of maximal q convex mixed-integer quadratic programs, which can be solved much faster.

At first, we consider the parametric MINLP

$$\left\{ \begin{array}{ll} \underset{\tilde{k} \in [\underline{k}, \overline{k}]}{\text{minimize}} & \phi(\tilde{k}, k) \\ \text{subject to} & \text{NNZ}(\tilde{k}) \leq \tilde{q}. \end{array} \right\}, \qquad \text{(MINLP}(\tilde{q})\text{)}$$

which depends on the integer parameter $\tilde{q} \in \{1, 2, \ldots, q\}$. We realize, that by solving MINLP$(\tilde{q})$ for $\tilde{q} = 1, 2, \ldots, q$, the associated objective function value is monotonically decreasing. Let $\tilde{k}(\tilde{q})$ denote the optimal solution of MINLP$(\tilde{q})$. Then, for the smallest $\tilde{q}$ which satisfies $\phi(\tilde{k}(\tilde{q}), k) \leq \delta$, the corresponding solution $\tilde{k}(\tilde{q})$ is identical to the solution of MINLP (4.11), (4.12).

The benefit of the reformulation is that, if we could approximate the non-convex objective function $\phi(\tilde{k}, k)$ in MINLP$(\tilde{q})$ by a convex quadratic objective function, we would tremendously reduce the computational complexity.

Obviously, the key contributions of non-convexity in Eq. (4.8) are due to terms of type

$$(\tilde{x}_i(t_l) - x_i(t_l))^2. \tag{4.13}$$

Since $\tilde{x}_i(t_0) = x_i(t_0) = x_0$ and Eqs. (4.1) and (4.7), we have the identity

$$(\tilde{x}_i(t_l) - x_i(t_l))^2 = \left(\int_{t_0}^{t_l} [f_i(\tilde{x}(t), \tilde{k}) - f_i(x(t), k)]\, dt \right)^2. \tag{4.14}$$

We consider only the integrand of (4.14) and add the zero term $(-f_i(x(t), \tilde{k}) + f_i(x(t), \tilde{k}))$ to obtain:

$$f_i(\tilde{x}(t), \tilde{k}) - f_i(x(t), k) = \underbrace{f_i(\tilde{x}(t), \tilde{k}) - f_i(x(t), \tilde{k})}_{=: A_i(\tilde{x}(t), x(t), \tilde{k})} + \underbrace{f_i(x(t), \tilde{k}) - f_i(x(t), k)}_{=: B_i(x(t), \tilde{k}, k)}. \tag{4.15}$$

Note, since $f(x, k)$ in Eq. (4.1) depends only linearly on k, we have

$$f_i(x(t), k) = \sum_{j=1}^{q} b_{ij}(x(t)) k_j, \tag{4.16}$$

where

$$b_{ij}(x) := \frac{\partial f_i}{\partial k_j}(x, k) \tag{4.17}$$

is independent from k. Hence we have the identity

$$B_i(x(t), \tilde{k}, k) = \sum_{j=1}^{q} b_{ij}(x(t))(\tilde{k}_j - k_j), \tag{4.18}$$

which is linear in k. We emphasize that Eq. (4.18) is an exact identity. On the other hand, if we assume that $\tilde{x}(t)$ is close to $x(t)$, we have by means of the continuity of f that

$$A_i(\tilde{x}(t), x(t), \tilde{k}) \approx 0 \ . \tag{4.19}$$

Surely we may assume that $\tilde{x}_i(t) \approx x_i(t)$ for $i \in \mathcal{I}$, i.e. the important species are not affected to much by the reduction. However, there is in general no justification why for non-important species ($i \notin \mathcal{I}$) $\tilde{x}_i(t)$ should be close to $x_i(t)$. Then, equation (4.19) is no longer valid and the reduction algorithm may fail. One solution of preceding problem is to enlarge the set $\mathcal{I}$ by the indexes of species which are indeed not important, but indispensable for a correct simulation of the important species. Turányi [89] calls these kind of species *necessary species* and further proposes an algorithm to identify them.

Loosely speaking, these necessary species have a strong influence on the term $A_i(\tilde{x}(t), x(t), \tilde{k})$ in Eq. (4.15). However, we found a way to deal implicitly with necessary species without applying Turányi's algorithm. If a species x_l is necessary for some index $l \notin \mathcal{I}$, then in general for some $i \in \mathcal{I}$ the absolute value of the sensitivity

$$\frac{\partial f_i(x(t), k)}{\partial x_l} \tag{4.20}$$

is relatively high. The corresponding change of f_i might be approximated to first-order by

$$\frac{\partial f_i(x(t), k)}{\partial x_l}(\tilde{x}_l(t) - x_l(t)).$$

But $(\tilde{x}_l(t) - x_l(t))$ may be approximated by first-order Taylor series expansion:

$$(\tilde{x}_l(t) - x_l(t)) \approx \frac{\partial x_l(t)}{\partial k}(\tilde{k} - k).$$

On the other hand, if a species $x_l(t)$ is not necessary at all, then the absolute value of the corresponding sensitivity in Eq. (4.20) is relatively small for all $i \in \mathcal{I}$. This motivates us to approximate $A_i(\tilde{x}(t), x(t), \tilde{k})$ in Eq. (4.15) by the linearization in $(\tilde{k} - k)$

$$A_i(\tilde{x}(t), x(t), \tilde{k}) \approx \sum_{j=1}^{q} \underbrace{\frac{\partial f_i(x(t), k)}{\partial x} \frac{\partial x(t)}{\partial k_j}}_{=: \tilde{a}_{ij}(t)} (\tilde{k}_j - k_j). \tag{4.21}$$

We are aware, that this is an heuristic approach and some problems may arise when $(\tilde{k} - k)$ is so large that the linearization is not valid anymore. However, since we are always able to compare the reduced model with the original one, we may ignore this possible complications. Finally we collect the two alternative approximations in Eqs. (4.19) and

(4.21) and approximate the term $A_i(\tilde{x}(t), x(t), \tilde{k})$ in Eq. (4.15) by

$$G_i(\tilde{x}(t), x(t), \tilde{k}) \approx \sum_{j=1}^{q} \sigma\, \tilde{a}_{ij}(t)(\tilde{k}_j - k_j), \quad \sigma \in \{0, 1\}, \tag{4.22}$$

to yield

$$f_i(\tilde{x}(t), \tilde{k}) - f_i(x(t), k) \approx \sum_{j=1}^{q} \underbrace{(\sigma\, \tilde{a}_{ij}(t) + b_{ij}(t))}_{=:\, \tilde{g}_{ij}(t)} (\tilde{k}_j - k_j), \tag{4.23}$$

where $\sigma = 0$ means that we are relying on Eq. (4.19), while $\sigma = 1$ refers to Eq. (4.21). In general, both choices of σ are possible and may be used for the subsequently introduced reduction method. However, if the partial Jacobian $(\partial f_i/\partial k_j)$, $i \in \mathcal{I}$, $j \in \{1, \ldots, q\}$, is very sparse, the choice $\sigma = 0$ may produce only poor approximations of $(f_i(\tilde{x}(t), \tilde{k}) - f_i(x(t), k))$, so that then $\sigma = 1$ should be chosen.

Inserting Eqs. (4.14) and (4.23) into Eq. (4.8), finally yields the approximate objective functional

$$\begin{aligned}
\tilde{\phi}(\tilde{k}, k) :=& \sum_{l=1}^{N} \sum_{i \in \mathcal{I}} {w_{il}}^2 \left(\int_{t_0}^{t_l} \sum_{j=1}^{q} \tilde{g}_{ij}(t)(\tilde{k}_j - k_j)\, dt \right)^2 \\
=& \sum_{l=1}^{N} \sum_{i \in \mathcal{I}} \left(w_{il} \int_{t_0}^{t_l} \sum_{j=1}^{q} \tilde{g}_{ij}(t)(\tilde{k}_j - k_j)\, dt \right)^2 \\
=& \sum_{l=1}^{N} \sum_{i \in \mathcal{I}} \left(\sum_{j=1}^{q} \left(w_{il} \underbrace{\int_{t_0}^{t_l} \tilde{g}_{ij}(t))\, dt}_{=:\, \tilde{c}_{ilj}} \right) (\tilde{k}_j - k_j) \right)^2 \\
=& \sum_{l=1}^{N} \sum_{i \in \mathcal{I}} \left(w_{il} \sum_{j=1}^{q} \tilde{c}_{ilj}(\tilde{k}_j - k_j) \right)^2 .
\end{aligned} \tag{4.24}$$

Setting $\tilde{C}_{il} := (\tilde{c}_{il1}, \ldots, \tilde{c}_{ilm})$, we have

$$\begin{aligned}
\tilde{\phi}(\tilde{k}, k) =& \sum_{l=1}^{N} \sum_{i \in \mathcal{I}} \left(w_{il} \tilde{C}_{il} (\tilde{k} - k) \right)^2 = \sum_{l=1}^{N} \sum_{i \in \mathcal{I}} (\tilde{k} - k)^T \left(w_{il} \tilde{C}_{il} \right)^T \left(w_{il} \tilde{C}_{il} \right) (\tilde{k} - k) \\
=& (\tilde{k} - k)^T \left(\sum_{l=1}^{N} \sum_{i \in \mathcal{I}} \left(w_{il} \tilde{C}_{il} \right)^T \left(w_{il} \tilde{C}_{il} \right) \right) (\tilde{k} - k),
\end{aligned} \tag{4.25}$$

which is obviously quadratic in $\tilde{k} - k$ and therewith quadratic in $\tilde{k}$. We may further compute the integrals by trapezoidal sums, say on the interval $[t_{l-1}, t_l]$ $(1 \le l \le N)$:

$$\int_{t_{l-1}}^{t_l} \tilde{g}_{ij}(t)\, dt = \frac{\tilde{g}_{ij}(t_{l-1}) + \tilde{g}_{ij}(t_l)}{2} (t_l - t_{l-1}), \tag{4.26}$$

where the "=" shall be interpreted from a numerical point of view, i.e. we assume that the grid $t_0 < t_1 < \cdots < t_N$ is sufficiently fine to accurately compute the integrals in Eq. (4.26). Hence, we have

$$\tilde{c}_{i\bar{l}j} = \frac{1}{2}\sum_{l=1}^{\bar{l}}(\tilde{g}_{ij}(t_l) + \tilde{g}_{ij}(t_{l-1}))(t_l - t_{l-1}), \tag{4.27}$$

and setting for notational convenience

$$\tilde{G}_i(t) := (\tilde{g}_{i1}(t), \ldots, \tilde{g}_{im}(t)) \tag{4.28}$$

yields

$$\tilde{C}_{i\bar{l}} = \frac{1}{2}\sum_{l=1}^{\bar{l}}\left(\tilde{G}_i(t_l) + \tilde{G}_i(t_{l-1})\right)(t_l - t_{l-1}), \tag{4.29}$$

and

$$\begin{aligned}\tilde{\phi}(\tilde{k}, k) =&(\tilde{k} - k)^T \left[\sum_{\bar{l}=1}^{N}\sum_{i\in\mathcal{I}}\left(\frac{w_{il}}{2}\sum_{l=1}^{\bar{l}}\left(\tilde{G}_i(t_l) + \tilde{G}_i(t_{l-1})\right)(t_l - t_{l-1})\right)^T\right. \\ &\left.\left(\frac{w_{il}}{2}\sum_{l=1}^{\bar{l}}\left(\tilde{G}_i(t_l) + \tilde{G}_i(t_{l-1})\right)(t_l - t_{l-1})\right)\right](\tilde{k} - k) \\ =&(\tilde{k} - k)^T \underbrace{\left(\sum_{j=0}^{N}\sum_{i\in\mathcal{I}}\left(w_{il}^*\tilde{G}_i(t_l)\right)^T\left(w_{il}^*\tilde{G}_i(t_l)\right)\right)}_{=:\,H}(\tilde{k} - k),\end{aligned} \tag{4.30}$$

where w_{il}^*, $i \in \mathcal{I}$, $0 \le l \le N$ are weighting factors, depending on the original weights w_{il} and the length of the intervals $[t_{l-1}, t_l]$. The $q \times q$-matrix H in (4.30) is positive semidefinite by construction.

4.2.4 Relationship to sensitivity analysis

We expand the definition of $\tilde{G}_i$ in Eq. (4.30):

$$\tilde{G}_i(t) = \sigma\,\frac{\partial f_i(x(t), k)}{\partial x}\frac{\partial x(t)}{\partial k} + \frac{\partial f_i(x(t), k)}{\partial k}, \quad \sigma \in \{0, 1\}. \tag{4.31}$$

Now we see how to relate the factor $\sigma \in \{0, 1\}$ to the sensitivity of the change rate f_i with respect to k: for $\sigma = 0$ the function $G_i(t)$ is identical to the *partial* derivative of f_i with respect to k; for $\sigma = 1$ $G_i(t)$ can be identified with to *total* derivative of f_i with respect to k. In both cases we have to compute the partial derivatives $\partial f_i/\partial k$, either by symbolic or algorithmic differentiation. Furthermore, for the choice $\sigma = 1$, $G_i(t)$ depends on the parametric sensitivities $\partial x(t)/\partial k$. Hence, then the computation of H requires a sensitivity analysis of the CRN with the original parameter vector k. In general, these sensitivities should be computed by an efficient numerical integrator with sensitivity analysis capabilities, e.g. [90].

4.2.5 From MINLP to MIQP

In the following, we substitute in the MINLP sequence (MINLP($\tilde{q}$)) the original objective function $\phi(\tilde{k}, k)$ by the convex and quadratic approximation $\tilde{\phi}(\tilde{k}, k)$ to yield the MIQP

$$\left\{ \begin{array}{ll} \underset{\tilde{k}\in[\underline{k},\overline{k}]}{\text{minimize}} & (\tilde{k}-k)^T H(\tilde{k}-k) \\ \text{subject to} & \text{NNZ}(\tilde{k}) \leq \tilde{q}. \end{array} \right\}, \qquad \text{(MIQP}(\tilde{q})\text{)}$$

which depends on the integer parameter $\tilde{q}$. However, for practical reasons, we eliminate the NNZ-operator in MIQP($\tilde{q}$) by means of a reformulation using the binary variable vector $y \in \{0,1\}^q$ which will satisfy

$$\sum_{i=1}^{q} y_i = \text{NNZ}(\tilde{k}).$$

Therefore, we set up the equivalent MIQP

$$\begin{aligned} \underset{\tilde{k},y}{\text{minimize}} \quad & \frac{1}{2}(\tilde{k}-k)^T H(\tilde{k}-k) && (4.32)\\ \text{subjec to} \quad & y_i \in \{0,1\}, \quad i=1,\ldots,q && (4.33)\\ & \tilde{k}_i \geq 0, \quad i=1,\ldots,q && (4.34)\\ & \tilde{k}_i - \overline{k}_i\, y_i \leq 0, \quad i=1,\ldots,q && (4.35)\\ & \tilde{k}_i - \underline{k}_i\, y_i \geq 0, \quad i=1,\ldots,q && (4.36)\\ & \sum_{i=1}^{q} y_i \leq \tilde{q}. && (4.37) \end{aligned}$$

Note that the objective function in Eq. (4.32), if we recall the construction of H and Eq. (4.31), can be interpreted as minimizing the weighted quadratic deviation of the change rates. In particular, we do *not* directly minimize the deviation in the species concentration but the deviation of their time derivatives. The same idea has been successfully applied in the incremental identification of kinetic models for homogeneous reaction systems [91].

4.3 Implementation Issues

4.3.1 Scaling, regularization and pre-reduction

In order to have a robust and numerically efficient method, one should pay attention to the implementation issues, that is the subject of this section. First we have found that a direct optimization of the MIQP (4.32)–(4.37) may result in non-acceptable results, due to numerical ill-conditioning. To avoid this problems one can apply scaling, regularization and pre-reduction to significantly improve the solution quality.

To force the optimized parameters $\tilde{k}$ to be in the same order of magnitude, a *scaling* of H with the diagonal matrix $D := \text{diag}(k)$ is performed, i.e. we employ the matrix

$$H_s := D^T H D. \qquad (4.38)$$

Formally, this leads to scale the original parameter vector k to

$$k = (1, 1, \ldots, 1)^T. \tag{4.39}$$

Hence, from now on, we assume without loss of generality the validity of Eq. (4.39).

Obviously, the matrix H_s is (at least) positive semidefinite by construction. However, using finite-precision arithmetic, H_s may become indefinite. Indeed, this is the case for many case studies. To circumvent optimization with an indefinite Hessian, we compute the minimal eigenvalue $\lambda_{\min}(H_s)$ of H_s and use the *regularization*

$$H_r = H_s + \gamma I, \qquad \gamma = |\min(\lambda_{\min}(H_s), 0)|, \tag{4.40}$$

where I is the q-dimensional unit matrix. However, the minimal eigenvalue is computed numerically and due to unreliabilities in this computation, H_r may still be indefinite, or at least the numerical MIQP solver claims H_r to be indefinite. In order to help the above detailed regularization and to facilitate the subsequent optimization, a *pre-reduction* step is also performed. There we reduce the dimension of the parameter vector $\tilde{k}$ by at least $q_{\text{pre-reduce}}$ reactions. The key point of the pre-reduction is to successively drop the parameters which, if only they are individually set to zero, have the least influence on the objective function. The corresponding algorithm is shown Table 4.1.

Table 4.1: Heuristic pre-reduction algorithm to determine the set $\mathcal{J}$ of important reactions

$\mathcal{J} := \{1, 2, \ldots, q\}$
for $l = 1, \ldots, q$ **do**
  if $\lambda_{\min}(H_{\mathcal{J}}) \geq 0$ **and** $\#\mathcal{J} \leq q - q_{\text{pre-reduce}}$ **then**
    break
  else
    $j = \arg\min_{j \in \mathcal{J}} \{\Delta k(j)^T H_{\mathcal{J}} \Delta k(j) \,:\, \Delta k_i(j) = -\delta_{ij}\, k_i \quad (i \in \mathcal{J})\}$ [a]
    $\mathcal{J} = \mathcal{J} \backslash \{j\}$
  end if
end for
return $\mathcal{J}$

[a] Here, δ_{ij} denotes the Kronecker symbol, i.e. $\delta_{ij} = 0$, iff $i \neq j$ and $\delta_{ij} = 1$, iff $i = j$. In detail, $\Delta k(j)$ is a $\#\mathcal{J}$-dimensional vector with only one non-zero entry at the position of index j.

The algorithm produces a set of parameter indices $\mathcal{J}$ which is of cardinality $n_{\mathcal{J}} = \#\mathcal{J}$ less or equal than q, such that

$$\mathbb{R}^{n_{\mathcal{J}} \times n_{\mathcal{J}}} \ni H_{\mathcal{J}} := (H_r)_{ij},\ i, j \in \mathcal{J} \tag{4.41}$$

is positive semidefinite. The corresponding parameter vectors and binary variable vector are denoted by $\tilde{k}_{\mathcal{J}}, k_{\mathcal{J}}$ and $y_{\mathcal{J}}$, respectively. Then, the pre-reduced set of reactions are used as an initial CRN in the model reduction, i.e. $H, \tilde{k}, k$ and y in MIQP (4.32)–(4.37) are replaced by $H_{\mathcal{J}}, \tilde{k}_{\mathcal{J}}, k_{\mathcal{J}}$ and $y_{\mathcal{J}}$, respectively.

4.3.2 Termination condition

The model reduction was formulated as a finite sequence of MIQPs in Eqs. (4.32)–(4.37) where in each iteration step we specify the maximum number of the existing reactions

$\tilde{q}$, $\tilde{q} = 1, 2, \ldots, q$. This MIQP sequence is derived by the original sequence MINLP($\tilde{q}$), where for each $\tilde{q}$ the goal is to find a reduced parameter vector $\tilde{k}$ such that the original objective function $\phi(\tilde{k}, k)$ is small. This original objective function is substituted by the quadratic approximation $\tilde{\phi}(\tilde{k}, k)$ (see Eq. (4.30)). At first sight, it might be straightforward to terminate the iteration based on the original objective function in Eq. (4.9). However, relying on the original objective is not necessary at all. Any other measurement for the model error can be used as well. In general, the termination condition can be any user-specified condition. This condition might even test sophisticated features of the reduced model like stability, weak reversibility, etc.

In the simplest particular case the termination condition can be defined such that the average relative deviation for all the important species should be smaller than a given limit, let say 5 %:

$$\phi_{\text{model\_error}}(\tilde{k}, k) < \delta := 0.05, \tag{4.42}$$

where we decided to measure the model error by means of the function

$$\phi_{\text{model\_error}}(\tilde{k}, k) = \frac{1}{N n_{\mathcal{I}}} \sum_{l=1}^{N} \sum_{i \in \mathcal{I}} \left(\frac{|\tilde{x}_i(t_l) - x_i(t_l)|}{\tilde{w}_{il}} \right). \tag{4.43}$$

A natural choice for the weighting factors $\tilde{w}_{il}$ in Eq. (4.43) can be $\tilde{w}_{il} = x_i(t_l)$, however if the concentration of an important species approaches zero the well known problem of the relative deviation occurs: the denominator approaches zero causing high relative deviation despite of small absolute deviation. This can be handled by choosing $\tilde{w}_{il} = \max\{Tol_i,\ x_i(t_l)\}$, where the tolerance value Tol_i is arbitrarily chosen to be

$$Tol_i = 10^{-4} \max_l x_i(t_l).$$

This gives relatively smaller weights to those points where the species concentration is less than 0.01%-of the maximum.

4.3.3 Approximations and tuning knobs

The proposed reduction methods requires some tuning factors which can be divided into two groups: the parameters of the numerical solvers and the parameters of the model reduction method itself.

The first class comprises the absolute and relative tolerance of the numerical integration routines for the solution of the IVP, as well as the parameters for the CPLEX numerical MIQP solver. However, these parameters may strongly depend on the numerical solvers and we do not discuss them in detail.

The second class comprises the number N of the grid points, as well as the location the grid points, $t_0, t_1, \ldots, t_N$ and the associated weights w_{il}, $i \in \mathcal{I}$, $l = 0, \ldots, N$ in Eq. (4.8). Further the choice of $\sigma = 0$ or $\sigma = 1$ in Eq. (4.23) may affect the solution.

The overview of the model reduction pipeline is depicted in Figure 4.1. The steps, which involve tuning factors of the second class are highlighted in yellow colour, further, steps involving approximations are depicted with dashed border.

- **Selecting important and necessary species**. Necessary species are usually selected after the first run of the algorithm, after investigating the error in the trajectories of the non-important species as discussed in the paragraph after Eq. (4.19) and in the next point.

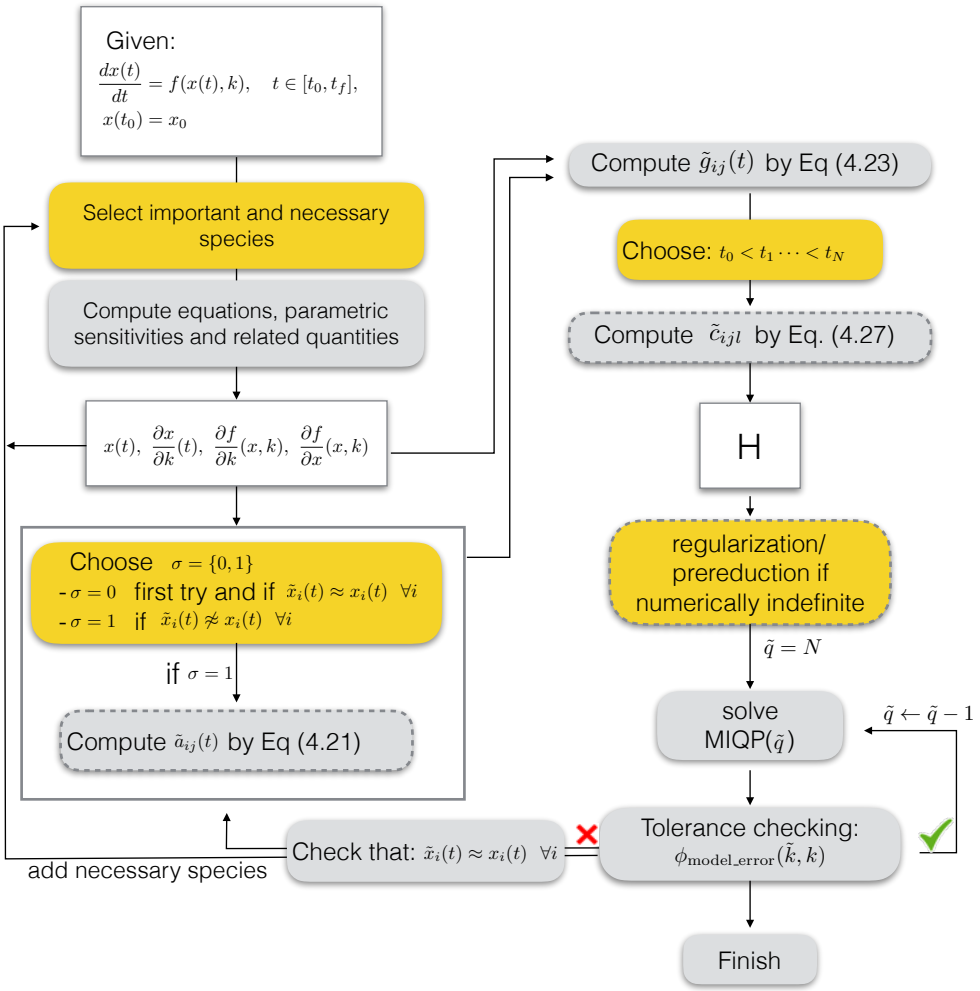


Figure 4.1: Overview of the model reduction pipeline. Yellow boxes indicate user-specific decisions, dashed borders indicate approximations. See details in Sec. 4.3.3

- **Selecting σ.** There is no guarantee that the trajectories of non-important species stay close to the original trajectories, i.e. for $j \notin \mathcal{I}$: $\tilde{x}_j \approx x_j$. However, this is an assumption for choosing $\sigma = 0$. To check how good this assumption is, one can simulate the reduced model trajectories $\tilde{x}$ and compare with the original one. When the trajectories are different, we include a first order correction via $\tilde{a}_{ij}$ in Eq. (4.21). One can calculate higher order terms to quantify the error, however, higher order approximations in the model reduction would result in non-quadratic cost function. Therefore, in case the linear correction is still poor, we recommend to consider adding necessary species.

- **Time-discretization** is chosen to approximate the integral in Eq. (4.26). This step implicitly results in the discretization of the original nonlinear dynamic system, which is known to be an open problem. It is often recommended to sample the trajectories denser whenever the states are changing and to take less samples from steady states, which approach worked for us. Wrong discretization can be usually detected in the early iterations of the reduction algorithm. In the early iterations, even though there are only a few reactions eliminated one can observe large deviation between the original trajectories and the reduced model trajectories.

Table 4.2: Reactions and rate coefficients of Hydrogen-Bromine reaction

	Reaction	Rate coefficient
(1)	$Br_2 + M \xrightarrow{k_1} 2\ Br\cdot + M$	$k_1 = 6.26 \cdot 10^5 \frac{cm^3}{mol\,s}$
(2)	$2\ Br\cdot + M \xrightarrow{k_2} Br_2 + M$	$k_2 = 1.56 \cdot 10^{15} \frac{cm^6}{mol^2\,s}$
(3)	$Br\cdot + H_2 \xrightarrow{k_3} H\cdot + HBr$	$k_3 = 2.61 \cdot 10^9 \frac{cm^3}{mol\,s}$
(4)	$H\cdot + HBr \xrightarrow{k_4} Br\cdot + H_2$	$k_4 = 1.39 \cdot 10^{13} \frac{cm^3}{mol\,s}$
(5)	$H\cdot + Br_2 \xrightarrow{k_5} Br\cdot + HBr$	$k_5 = 1.17 \cdot 10^{14} \frac{cm^3}{mol\,s}$
(6)	$Br\cdot + HBr \xrightarrow{k_6} H\cdot + Br_2$	$k_6 = 1.31 \cdot 10^4 \frac{cm^3}{mol\,s}$

- **Regularization and pre-reduction.** This issue is discussed in details in Section 4.3.1

4.4 Case Studies

The use of the proposed method are illustrated on three case studies of increasing complexity taken from the literature. The first case study, presented in the next subsection, considers a small Hydrogen-Bromine reaction network. Due to space limitations the larger networks of formaldehyde oxidation and alkane pyrolysis are presented in Appendix C.2 and C.3, respectively.

Example 4-1 (Reduction of Hydrogen-Bromine reaction network)

The Hydrogen-Bromine reaction is a well-known reaction mechanism in the literature [84, 92]. Because of the small size of this system it is easy to interpret the main idea of the method. The detailed description of the model equations (C.2-C.7) can be found in Appendix C.1.

Initialization The species in the reaction networks are Br_2, HBr, H_2, Br·, H· and M from which the molecules, namely Br_2, HBr and H_2 were selected as *important*. The rate constants corresponding to the reactions can be found in Table 4.3. The initial concentrations of species were taken from Turányi et al. [84]: $[Br_2]_0 = [H_2]_0 = 10^{-8}\, mol/cm^3$, $[M]_0 = 10^{-5}\, mol/cm^3$, the initial concentrations of the other species were considered to be zero. The time interval for model reduction is $[0,\ 1]$ second.

Computation of matrix H The matrix H is computed with the assumption of the validity of Eq. (4.19) i.e. $\sigma = 0$ in Eq. (4.23). To solve the IVP (4.1)–(4.2) the Matlab `ode15s` solver was used with $AbsTol = 10^{-19}$ and $RelTol = 10^{-13}$ absolute and relative tolerance settings. For the computation of the Hessian, $N = 100$ equidistantly sampled time points in the time interval were chosen:

$$t_l = \frac{l}{N}, \quad l = 1, \ldots, N.$$

Table 4.3: Rate coefficients of the original and the reduced models of the Hydrogen-Bromine reaction network

Rate Coefficients	**Original**	**(1*)**	**(2a*)**	**(2b*)**
k_1	$6.260 \cdot 10^5$	1.000	0.9937	1
k_2	$1.560 \cdot 10^{15}$	1.000	1.0066	1
k_3	$2.610 \cdot 10^9$	1.000	0.8542	1
k_4	$1.390 \cdot 10^{13}$	1.000	0.0	0
k_5	$1.170 \cdot 10^{14}$	1.000	1.0236	1
k_6	$1.310 \cdot 10^4$	0.0	0.0	0

In the second column are the rate coefficients of the original model, in the third column (1) are the relative rate coefficients of the reduced-by-one model, while in the last two columns (2a and 2b) are the rate coefficients for the reduced-by-two models.
*The rate constants are in the table for the reduced networks in relative units, i.e. the ratio of the estimated and the original value. For example, a value of 1.0 means that the rate coefficient did not change.

The weighting factors w_{il} in Eq. (4.8) were set to

$$w_{il} = \frac{1}{N \cdot \max(10^{-12}, x_i(t_l))}, \quad i \in \mathcal{I}, \quad l = 1, \ldots, N,$$

to reflect the relative error, where the "max"-term was introduced to avoid division by zero. Actually, apart from the "max"-term, these weights equally reflect the relative error of the important species, an approach also followed by [83].

Table 4.4: Important species concentration at the final time ($t_f = 1\,s$) in the original model and in the reduced models.

Species	**Original conc.**	**Relative deviation**		
		(1)	**(2a)**	**(2b)**
$[Br_2]$	$1.2876 \cdot 10^{-9}$	$2.2481 \cdot 10^{-5}$	0.3730	0.5140
$[H_2]$	$1.6504 \cdot 10^{-9}$	$1.9882 \cdot 10^{-5}$	0.3359	0.4652
[HBr]	$1.6699 \cdot 10^{-8}$	$-3.9301 \cdot 10^{-6}$	−0.0664	−0.0919

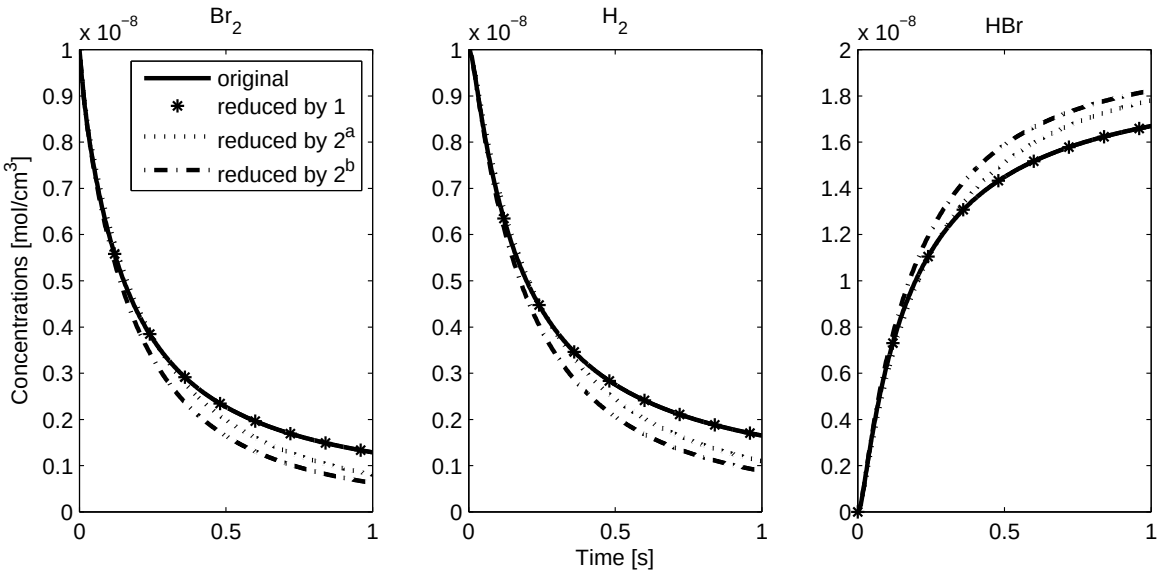


Figure 4.2: The concentrations of each important species in the original system and in the reduced systems.

Results One can see the solution of the original system of equations (C.2)–(C.7) for the important species together with the solutions of the reduced systems on Figure 4.2. The corresponding rate coefficient values can be found in Table 4.3. In the first step the algorithm omits the k_6 parameter which corresponds to the 6^{th} reaction, while the other parameter values are not changed in the first 4 digits. The resulted trajectories perfectly fit to the original solution.

If we further omit one more reaction the algorithm neglects the 4^{th} and 6^{th} rate coefficients. Using the $\underline{k}$ and $\overline{k}$ constraints in Eqs. (4.35) and (4.36) one can easily decide to let the algorithm estimate the value of non-zero parameters or not. If we want to identify the negligible parameters but want to keep the original values of the non-zero parameters then $\underline{k}_i = \overline{k}_i = 1$ should be defined. Figure 4.2 shows the result marked with dotted line when $\underline{k}_i = 0.1$ and $\overline{k}_i = 10$ and marked with dash-dotted line when $\underline{k}_i = \overline{k}_i = 1$ was chosen. The corresponding parameters are given in Table 4.3, in the columns (2^a) and (2^b) respectively.

The relative deviation of the concentration at the final time point (RD) for each important species

$$RD_i = \frac{\tilde{x_i}(t_f) - x_i(t_f)}{x_i(t_f)}, \quad i \in \mathcal{I} \tag{4.44}$$

can be found in Table 4.4, where $\tilde{x_i}$ is the concentration of the i-th important species in the reduced system and x_i is the corresponding concentration in the original system. If only less than 5 % differences is acceptable then it is clear that only the 6^{th} reaction can be omitted from the network.

■

4.5 Summary

A robust numerically stable method for reducing the complexity of large chemical reaction networks is constructed as a sequence of MIQPs where the objective function is derived from the parametric sensitivity matrix. The algorithm uses a given detailed kinetic mechanism

and measured data of the key species over a finite time horizon to determine the set of reactions as subsets of the reactions in the detailed mechanism, together with a re-estimated value of the reaction kinetic parameters. The proposed method sequentially eliminates reactions from the mechanism until the pre-specified tolerance limit in the species concentration space is reached.

The computational efficiency and numerical stability of the optimization is improved by applying a pre-reduction step, followed by suitable scaling and initial conditioning of the Hessian involved.

The proposed complexity reduction method is illustrated using three case studies taken from the reaction kinetic literature.

It is expected that the proposed method can be extended to make it suitable for eliminating the non-influential species, i.e. the species that do not change their concentration values over the time interval of interest. This is a possible direction of further work.

Chapter 5

Regularized parameter estimation in dynamic biochemical models

One of the main challenges in modeling biochemical processes is the calibration of dynamic models, also known as the parameter estimation problem. Parameter estimation aims at finding the unknown parameters of the model which give the best fit to a set of experimental data.

This problem has received considerable attention, as reviewed in [18, 93–97]. It is also frequently described as the inverse problem [98, 99], i.e., the inverse of model simulation from known parameters, considered the direct problem.

Parameter estimation is a complex task which can be approximately divided into the sub-tasks of (i) model definition, (ii) data pre-processing, (iii) computation of the estimates by numerical optimization and (iv) analysis of the resulted parameter estimates.

First, the model equations and related modelling assumptions are collected, and known and unknown parameters are listed. Although the values of model parameters are often unknown, a good guess of the possible parameter range can be found in the literature, or derived from thermodynamic assumptions, etc. However, this step is very much case study dependent.

Raw experimental data is usually processed before used for parameter estimation. Data can be transformed in order to fulfill general assumptions about experimental noise. In general, the type and magnitude of the measurement error depend on both the experimental techniques and the data processing [100–103]. Further, experimental conditions for the model simulation are also defined here, i.e. the initial conditions and inputs of the models.

The problem of finding the best fitting model parameters is usually formulated as a nonlinear optimization problem. A cost function that measures the discrepancy between model predictions and data is minimized iteratively by a selected numerical optimization algorithm. The algorithm reports the best fitting parameter vector upon convergence to the minima of the cost function. However, numerical data fitting in dynamical systems is a non-trivial endeavour, full of pitfalls (see, e.g. Chapter 4 in [104]). The problem is certainly not exclusive of systems biology: it has been extensively studied in other areas, as reviewed in [105], each one contributing with somewhat different perspectives regarding the difficulties encountered and how to surmount them.

Finally, the estimated model parameters are analyzed in order to determine the quality of the estimation. The usual procedure is to construct confidence intervals for the parameter estimates. Further, the quality of the calibrated model can also be evaluated by validation, i.e. predicting data sets that were not used in the model calibration.

In this chapter, we build up a parameter estimation framework, while we focus on the

numerical optimization part. We would like to address two key characteristics of the inverse problem which make it very hard: ill-conditioning and nonconvexity [18, 106, 107]. These concepts are intimately related with other similar notions developed independently in different communities [105]. For example, ill-conditioning can be related to the lack of identifiability arising from the model structure, and/or from information-poor data. Nonconvexity and multi-modality usually cause convergence to local solutions (local minima), which are estimation artifacts. Both are significant sources of concern that need to be properly addressed.

Non-convex optimization in parameter estimation problems. It is well-known that the typical (weighted) least-squares cost functions used in parameter estimation problems are nonlinear and nonconvex in the model parameters (see e.g. [104, 107–109]. Many efficient *local optimization algorithms* have been developed to find the solution of nonlinear least squares problems, including Gauss-Newton, Levenberg-Marquardt and trust-region methods [104]. These local methods, (and others like truncated and quasi-Newton) are especially efficient when provided with high quality first (gradient, Jacobian) and second order (Hessian) information via parametric sensitivities [110, 111]. However, in this type of problems they will likely converge to local solutions close to the initial guess of the parameters.

Multi-start local methods (i.e. performing multiple runs initiating local optimizers from a set of initial guesses distributed in the search domain) have been suggested as more robust alternatives. Typically the set of initial guesses is generated inside the parameter bounds either randomly or by a more sophisticated sampling scheme, such as Latin hypercube sampling [112]. Multi-start methods have shown good performance in certain cases, especially when high-quality first order information are used and the parameter search space is restricted to a relatively small domain [113, 114]. However, other studies [108, 115, 116] have shown that multi-start methods become inefficient as the size of the search space increases, and/or when the problem is highly multi-modal, since many of the local searches will explore the same local basins of attraction repeatedly.

Therefore, a number of researches have supported the use of global optimization as a better alternative. However, the current state of the art in global optimization for this class of problems is still somewhat unsatisfactory. Deterministic global optimization methods [117–121] can guarantee global optimality but their computationally cost increases exponentially with the number of estimated parameters. Alternatively, stochastic and metaheuristic methods [18, 96, 108, 109, 122, 123] can be used as more practical alternatives, usually obtaining adequate solutions in reasonable computation times, although at the price of no guarantees. In the context of metaheuristics, hybrids (i.e. combinations) with efficient local search methods have been particularly successful [115, 116, 122, 124–127].

Ill-conditioned optimization problems. A problem is said to be well-posed if a solution exists, the solution is unique and the solution continuously depends on the initial conditions. If any of these conditions are not fulfilled, the problem is called ill-posed or ill-conditioned.

In kinetic models we often find that, the parameter estimation has multiple solutions or the solution changes extremely upon small perturbation of the data [128]. These phenomena arise from (i) models with large number of parameters (over-parametrization), (ii) experimental data scarcity (lack of excitation) and (iii) significant measurement errors [18, 106]. As a consequence, we often obtain overfitting of such kinetic models, i.e. calibrated models with reasonable fits to the available data but poor capability for generalization (low predictive value).

Regularization methods have a rather long history in inverse problems [129] as a way to surmount ill-posedness and ill-conditioning. The regularization process introduces additional information in the estimation, usually by penalizing model complexity and/or wild

behaviour. Regularization aims at ensuring the uniqueness of the solution [130], to reduce the ill-conditioning. However, one crucial step is the proper balancing of prior knowledge and information in the data, also known as the tuning of the regularization [131].

Regularization has been mainly used in fields dealing with estimation in distributed parameter systems, such as tomography and other image reconstruction techniques. Recently, it has enjoyed wide success in machine learning [132], gaining attention from the systems identification area [133]. However, the use of regularization in systems biology has been marginal [134], especially regarding mechanistic (kinetic) nonlinear models. Bansal et. al [135] compared Tikhonov and truncated singular value decomposition regularization for the linear regression model of green fluorescent protein reporter system. Kravaris et al. [106] compared the theoretical aspects of parameter subset estimation, Tikhonov and principal component analysis based regularization, also in a linear model framework. Wang and Wang [136] presented a two stage Bregman regularization method for parameter estimations in metabolic networks. However, there is no clear conclusion from these studies, for nonlinear inverse problems, i.e. there is no general recipe for the selection of regularization method and its tuning. Further, it is known that even for linear systems, choosing a method from the plethora of existing techniques is non-trivial [131].

5.1 Parameter estimation in dynamic models

Mathematical model. Here we will consider dynamic models of biological systems described by general nonlinear differential equation. A common case is that of kinetic models. For the case of biochemical reaction networks, and under the assumption of well-mixed compartments, kinetic models describe the concentration dynamics using nonlinear deterministic ordinary differential equations. One of the most general form of these equations is given by the deterministic state-space model:

$$\frac{dx(t,\theta)}{dt} = f(t, x(t,\theta), u(t), \theta), \tag{5.1}$$

$$y(x,\theta) = g(x(t,\theta),\theta), \tag{5.2}$$

$$x(t_0) = x_0(\theta), \quad t \in [t_0,\, t_f] \ , \tag{5.3}$$

where $x \in \mathbb{R}^{N_x}$ is the state vector (often concentrations), the $f(\cdot) : \mathbb{R}^{1\times N_x \times N_u \times N_\theta} \mapsto \mathbb{R}^{N_x}$ vector function is constructed from the reaction rate functions and stimuli $u(t)$. The N_θ dimensional parameter vector θ contains the positive parameters of the reaction rate functions–for example the reaction rate coefficients, Hill exponents, dissociation constants, etc.–, but can also include the initial conditions. The observation function[1] $g(\cdot) : \mathbb{R}^{N_x \times N_\theta} \mapsto \mathbb{R}^{N_y}$ maps the state variables to the vector of observable quantities $y \in \mathbb{R}^{N_y}$, these are the signals that can be measured in the experiments. The observation functions may also directly depend on estimated parameters for example on scaling parameters. When multiple experiments in different experimental conditions are considered, typically the same model structure is assumed, but the initial conditions and stimuli are adapted to the new conditions. Note that these equations are more general, than the dynamic equations used in the previous chapters, e.g. Equation (1.5) for MAL-CRNs or Equation (2.9) for bio-CRNs.

Calibration data and error models. We consider a general case, where the data is collected in multiple experiments at discrete time points $t_i \in [t_0,\, t_f]$, thus the model outputs

[1]Note that, here g is used for the observation function, which is not related to the reaction kinetics (also denoted by g) in Chapter 2.1.

must be discretized accordingly. Let us denote the model prediction at time t_i, of the j-th observed quantity in the k-th experimental condition by y_{ijk}. Due to measurement errors the true signal value is unknown and a noise model is used to express the connection between the true value y_{ijk} and measured data $\tilde{y}_{ijk}$.

Maximum likelihood and cost function. Assuming that the measurements (which are denoted by $\tilde{y}$) are contaminated by additive normally distributed uncorrelated random measurement errors –i.e. $\tilde{y}_{ijk} = y_{ijk}(x(t_i), \theta) + \epsilon_{ijk}$ where $\epsilon_{ijk} \sim \mathcal{N}(0, \sigma^2_{ijk})$ is the random error with standard deviation σ_{ijk} and $\tilde{y}_{ijk}$ is the measured value–, the estimation of the model parameters is formulated as the maximization of the likelihood [137, 138]

$$\mathbb{L}(\tilde{y} \mid \theta) = \prod_{k=1}^{N_\mathrm{e}} \prod_{j=1}^{N_{y,k}} \prod_{i=1}^{N_{t,k,j}} \frac{1}{\sqrt{2\pi\sigma^2_{ijk}}} \exp\left(-\frac{1}{2} \frac{(y_{ijk}(x(t_i,\theta),\theta) - \tilde{y}_{ijk})^2}{\sigma^2_{ijk}} \right) \ , \tag{5.4}$$

where N_e is the number of experiments, $N_{y,k}$ is the number of observed compounds in the k-th experiment, and $N_{t,k,j}$ is the number of measurement time points of the j-th observed quantity in the k-th experiment. The total number of data points is $N_\mathrm{D} = \sum_{k=1}^{N_\mathrm{e}} \sum_{j=1}^{N_{y,k}} \sum_{i=1}^{N_{t,k,j}} 1$. The maximization of the likelihood function (5.4) is equivalent to the minimization of the weighted least squares cost function [138]

$$Q_\mathrm{LS}(\theta) = \sum_{k=1}^{N_\mathrm{e}} \sum_{j=1}^{N_{y,k}} \sum_{i=1}^{N_{t,k,j}} \left(\frac{y_{ijk}(x(t_i,\theta),\theta) - \tilde{y}_{ijk}}{\sigma_{ijk}} \right)^2 = R(\theta)^T R(\theta) \ , \tag{5.5}$$

where the residual vector $R(\cdot) : \mathbb{R}^{N_\theta} \to \mathbb{R}^{N_\mathrm{D}}$ is constructed from the squared terms by arranging them to a vector. Hence the model calibration problem can be stated as the well-known nonlinear least-squares (NLS) optimization problem:

$$\begin{aligned} \underset{\theta}{\text{minimize}} \quad & Q_\mathrm{LS}(\theta) = R(\theta)^T R(\theta) \\ \text{subject to} \quad & \theta_\mathrm{min} \leq \theta \leq \theta_\mathrm{max} \ , \\ & \frac{dx(t,\theta)}{dt} = f(u(t), x(t,\theta), \theta) \ , \\ & y(x,\theta) = g(x(t,\theta),\theta) \ , \\ & x(t_0) = x_0(\theta), \quad t \in [t_0,\, t_f] \ . \end{aligned} \tag{5.6}$$

A $\hat{\theta}$ vector that solves this optimization problem is called the *optimal parameter vector*, or the *maximum likelihood estimate* of the model parameters. However, note that (1) the uniqueness of the solution is not guaranteed. Further, the solution can be different based on the optimization algorithm and initial guesses used to solve this nonlinear optimization problem – as discussed in the introduction. These results in the ill-posedness of the calibration problem.

5.2 Global optimization

To solve the optimization problem (5.6), we have extended the enhanced scatter search (eSS) metaheuristic presented by Egea et al [125]. eSS is a meta-heuristic global optimization algorithm for general nonlinear optimization problems, which makes use of elements

of the scatter search and path re-linking metaheuristics. The algorithm switches between a stochastic global search phase (diversification) and a user specified, typically gradient based (local) method (intensification phase). By the combination of the stochastic global optimization approach with a local method, the algorithm implements new strategies to avoid sub-optimal solutions, while keeping the computational cost relatively low.

Here, we tailored the general purpose eSS to solve the nonlinear *least-squares* optimization problem (5.6) efficiently. Our extension of this method, which we will call eSS2, incorporates several methodological and numerical improvements with the aims of (i) exploiting the least-squares structure of the problem, (ii) increasing its overall robustness and efficiency, (ii) avoiding the need of tuning of search parameters by the user (a drawback of many metaheuristics). These improvements can be summarized as follows:

- *Efficient local search* after extensive comparisons of local solvers, we selected the adaptive algorithm NL2SOL [139]. This is a variant of the Gauss-Newton method that utilizes the Jacobian of the residual vector to approximate and iteratively upgrade the parameter vector. In order to increase its efficiency, we also provide it with high quality gradient information (see below), resulting in speed-ups of up to 20 times.

- *Efficient integration* of the initial value problem and its extension with parametric forward sensitivity equations using the CVODES solver [140], providing it with the Jacobian of the dynamics.

- *Fast computation*: although the global solver eSS is implemented in Matlab, the integration of the initial value problem is done in C in order to speed-up the computations up to 2 orders of magnitude.

- *Robust default tuning*: metaheuristics require the user to set a number of search parameter values which usually require a number of time-consuming initial trial runs. In the method proposed here, we have made sure that the default search parameters work well without the need of any tuning, which is an additional important advantage. These settings are given in the Appendix D.3.

5.3 Regularization

Although, we can solve the optimization problem (5.6) for kinetic models efficiently using the eSS2 method, we still face the problem of ill-posedness. As mentioned in the introduction, we often find that the solution very much depends on the initial guess of parameters and on the random numbers taken by the stochastic optimization (i.e. several runs of the optimizer give very different estimates of the parameters). This problem is inherently related to the lack of identifiability of the model parameters, and it is especially related to models with large number of parameters and scenarios in which the calibration data was collected in experiments not designed for parameter estimation –a usual scenario in biochemical models. This often leads to models with low predictive value.

Here we want to investigate the role that regularization can play regarding the calibration of nonlinear kinetic models. First of all, we need to address to question of which type of regularization should we use. Second, since kinetic models often have a fixed and rather stiff nature (as opposed to the flexibility of e.g. neural networks, as used in machine learning), it is a priori unclear if regularization can really help to avoid overfitting and enhance the predictive value of the calibrated model. Third, since most dynamic models in systems biology are severely over-parametrized, we want to explore its capabilities for systematic balancing the effective number of fitted parameters based on the available calibration data.

Fourth, we want to evaluate the impact of regularization on the convergence properties of the global optimization solvers.

In order to answer these questions, here we present a critical comparison of a wide range of regularization methods applicable to nonlinear kinetic models. We then detail a procedure with guidelines for regularization method selection and tuning. Finally, we use numerical experiments with challenging problems of increasing complexity to illustrate the usage and benefits of regularization, addressing the questions above.

5.3.1 Statement of the regularized estimation

We consider penalty type regularization techniques [129], which add a penalty $\Gamma(\theta)$ to the original objective function (5.5). This results in the following regularized optimization problem:

$$\begin{aligned} \hat{\theta}_\alpha \leftarrow \underset{\theta}{\text{minimize}} \quad & Q_{\mathrm{R}}(\theta) = Q_{\mathrm{LS}}(\theta) + \alpha\Gamma(\theta) \\ \text{subject to} \quad & \theta_{\min} \leq \theta \leq \theta_{\max} \ , \\ & \frac{dx(t,\theta)}{dt} = f(u(t), x(t,\theta), \theta) \ , \\ & y(x,\theta) = g(x(t,\theta),\theta) \ , \\ & x(t_0) = x_0(\theta), \quad t \in [t_0,\, t_f] \ . \end{aligned} \tag{5.7}$$

Here $\alpha \in \mathbb{R}_+$ is the non-negative regularization parameter and $\Gamma(\cdot) : \mathbb{R}^{\mathbb{N}_\theta} \to \mathbb{R}_+$ is the regularization penalty function. When the solution of the original problem ($\alpha = 0$) is ill-posed, one has to incorporate some a priori assumption, which makes the estimation well posed. It is assumed that the penalty function $\Gamma(\theta)$ is well conditioned and has a unique minimum. Thus, as the regularization parameter $\alpha \to \infty$ the optimization problem (5.7) is well-posed but highly biased by the a priori assumption, and when $\alpha = 0$ one obtains the original, ill-posed estimation problem. Therefore the role of the regularization parameter α is to properly balance the information of the data and the prior knowledge. However, this is a non-trivial task even for linear problems, as we will discuss below.

Determining the proper regularization parameter requires multiple solutions of the regularized optimization problem (5.7), therefore the computational efficiency is also crucial. Here we chose the Tikhonov regularization framework in order to match the form of the penalty to the least squares formalism of the objective function. In this case the least squares cost function can be simply augmented by the quadratic penalty function

$$\Gamma(\theta) = (\theta - \theta^{\mathrm{ref}})^T W^T W (\theta - \theta^{\mathrm{ref}}), \tag{5.8}$$

where $W \in \mathbb{R}^{N_\theta \times N_\theta}$ is a diagonal scaling matrix and $\theta^{\mathrm{ref}} \in \mathbb{R}^{N_\theta}$ is a reference parameter vector. In the special case, when W is the identity matrix, we call the scheme as the *non-weighted Tikhonov regularization* scheme (or shortly as Tikhonov regularization). If further, the θ^{ref} is the null-vector, the corresponding regularization scheme is often referred as *ridge regularization*.

5.3.2 Scenarios based on prior information

Kinetic models can overfit the data leading to poor generalizability. Here we propose using prior knowledge to select the most appropriate regularization method to avoid such overfit. Based on the level of confidence in this prior knowledge, we can consider three possible scenarios:

- *Worst case scenario*, where we have absolutely no prior information about the parameter values, typically resulting in very ample bounds and random initial guesses for the parameters.
- *Medium case scenario*, where there is some information about the parameters and their bounds.
- *Best case scenario*: the situation where a good guess of the parameters is at hand.

Below we will provide, for each scenario, robust recommendations regarding the regularization method to use and its tuning.

5.3.3 Prediction error in biased estimation

In order to evaluate the performance of the calibrated model, we will use cross-validation [141–143]: the calibrated model is used to predict a yet unseen set of data and the prediction error is computed. A good model should not only fit well the calibration data, but it also should predict well the validation data.

In the appendix D.1 we utilize the bias-variance decomposition of the prediction error and show when and how regularization can lead to smaller prediction error. Here we summarize the main steps and conclusions.

First the expected prediction error of a model, that was calibrated on a data set $D_{\mathcal{C}}$ and evaluated on an independent validation data set $D_{\mathcal{V}}$ is decomposed to a variance term, a bias term and a remaining term due to experimental noise in the data. We find that the variance of the prediction error decreases with larger regularization, but the bias is a monotonically increasing function of the regularization parameter. This trade-off is beneficiary as long as the following inequality holds (see the derivation in the appendix D.1):

$$\sigma^2 \frac{N_\theta - N_\theta^{\text{eff}}(\alpha)}{N_D} > \frac{\alpha}{8} ||\theta_t - \theta^{\text{ref}}||^2, \tag{5.9}$$

here σ^2 is the variance of the measurement error, N_θ is the number of model parameters, $N_\theta^{\text{eff}}(\alpha)$ is the effective number of parameters, which is a decreasing function of the regularization parameter α, N_D is the number of data points, θ_t is the true parameter vector and θ^{ref} is the reference parameter vector used in the regularization. The left hand side of the inequality represents the reduction in the variance –due to the decrease in the number of effective parameters– and the right hand side is the bias, that is introduced by the regularization, since the true parameter vector is unknown.

Therefore, *regularization increases the performance of the calibrated model when*

1. the calibration data is noisy (σ is large) and the amount of data is limited (N_D is small),
2. there are a large number of correlated parameters, and therefore the Hessian of the original problem has very small eigenvalues. In this case even a small regularization parameter can largely reduce the effective number of parameters, i.e. $N_\theta \gg N_\theta^{\text{eff}}(\alpha)$.
3. One has a good guess of the true parameters ($||\theta_t - \theta^{\text{ref}}||^2$ is small), for example from other independent experiments, previous studies or based on the biological or physico-chemical meaning of the parameters.

However, note that regularization may damage the prediction (the reduced variance is smaller than the introduced biased) if the original problem is not ill-posed, i.e. $N_\theta \approx N_\theta^{\text{eff}}(\alpha)$, α is set to a large value and the provided reference parameters are far from the true parameters.

Table 5.1: Summary of the case studies. Each column describes a calibration problem. Further details can be found in Appendix D.5 (detailed descriptions including the model differential equations).

Short Name	BBG	FHN	MAPK	GOsc	TGFB	TSMP	CHM
Description	Biomass Batch Growth	FitzHugh-Nagumo Oscillator	MAPK Signalling Pathway	Goodwin's Oscillator	TGF-β Signalling Pathway	3-Step Metabolic Pathway	Chemotaxis Signalling Pathway
Reference	[146]	[147, 148]	[149]	[150]	[113]	[108]	[151]
Implementation of dynamics	[146]	BIOMD00... 000000010*	BIOMD0000... 0000346*,[153]	[152]	[113]	[108]	BIOMD000... 00000404*
Total parameters	4	3	22	8	21	36	60
Estimated parameters	4	3	6	8	18	36	38
States	2	2	8	3	18	8	26
Observed states	2	1	2	3	16	8	7(+1)
Experiments	1	1	1	1	1	8	2
Data points	22	6	20	20	240	1344	160

* the dynamic model can be found in the Biomodels Database (http://www.ebi.ac.uk/biomodels-main)

5.3.4 Tuning the regularization

The regularization parameter balances the a priori knowledge and the information of the data, therefore plays a vital role in the regularization. When $\alpha = 0$, the regularized optimization (5.7) becomes the original problem (5.6) and the variance of the estimated parameters dominates the prediction error (D.2). While as $\alpha \to \infty$ the problem is well posed, but biased towards the reference parameter set. The goal of a tuning method is to find an optimal value for α, which minimizes the prediction (or parameter estimation) error.

The exact computation of the optimal regularization parameter is not possible, since the computation of the prediction bias-variance trade-off would require the knowledge of the true parameters. Many tuning methods (see [129, 131, 144, 145] and the references therein) have been developed based on different assumptions and approximations to compute an approximate regularization parameter value. We summarized a set of 15 tuning methods for the selection of the α parameter in Appendix D.4. These 15 tuning methods are compared based on several case studies in Section 5.4.4.

5.4 Results and discussion

5.4.1 Numerical case studies

We have considered a set of seven parameter estimations problems, which are used as numerical case studies. These case studies have been chosen as representatives of the typical problems arising in computational systems biology, i.e. partially observed nonlinear dynamic models and sparse noisy measurements. These examples include signalling and metabolic pathway models of increasing complexity. Table 5.1 contains a short summary of these case studies, with the original references and an overall view of number of estimated parameters, dynamic state variables, observables, and data. Further details, including model equations and the data sets used for model calibration and cross-validation are reported in Appendix D.5. It should be noted that in several of these examples the original references only describe the model dynamics, not the full parameter estimation problems.

In the following sections, we use these examples to illustrate the issues and pitfalls arising from the nonconvexity and ill-conditioning of the estimation problems. Next, we

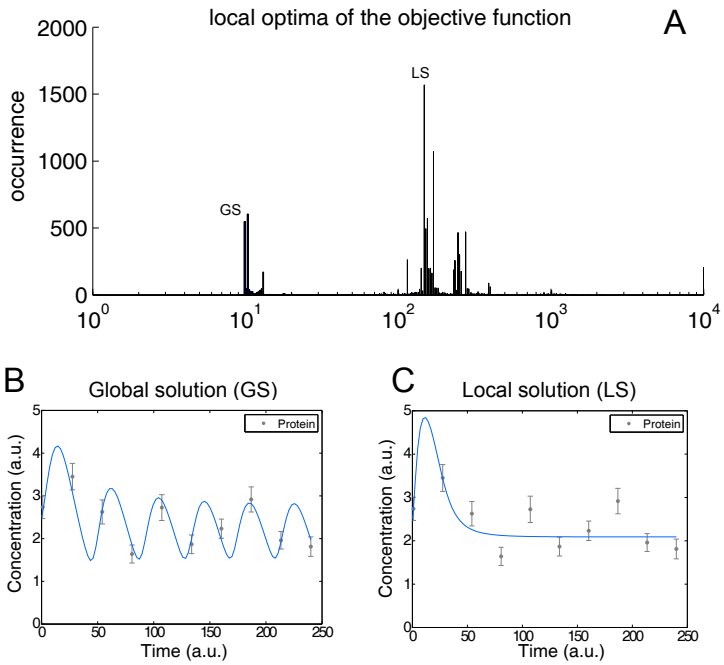


Figure 5.1: Local optima of the objective function corresponding to the GOsc case study. Figure A: distribution of the final objective function values of 10000 runs of local solver NL2SOL. Figure B: fit corresponding to the global optima (GS). Figure C: fit corresponding to the most frequently achieved local minima (LS).

use them to illustrate the key ideas behind the methods presented above, including the bias-variance trade-off, the tuning of the regularization, the effect of the quality of the prior knowledge on the regularization, and their impact on cross-validation results. For the sake of brevity, we include summarized or selected results in the main text, but detailed results illustrated by figures for all the case studies can be found in Appendix D.6.

5.4.2 Multi-modality of the optimization problem

Since the estimation problem stated above is nonconvex, multi-modality (existence of multiple local solutions) will be a key possible pitfall. As already discussed, local nonlinear least squares (NLS) algorithms will find the local minima of the objective function in the vicinity of the initial point. A characterization of the set of possible local optima can be obtained by the frequency distributions of the solutions found by a multi-start local procedure, i.e. starting local optimizations from different initial points, selected randomly in the parameter space. If the initial points cover the parameter space adequately well, the observed distribution of the local optima can be used to quantify the difficulty of the parameter estimation problem arising from multi-modality.

For example, Figure 5.1 A shows the distribution of these local minima for the Goodwin's oscillator (GOsc) case study. The distribution was obtained by solving 10000 optimization problem (of which approximately 97% converged) with the NL2SOL NLS algorithm started from randomly chosen initial guesses. These initial points were selected based on the logarithmic Latin hypercube sampling (LHS) method. The distribution of the obtained local optima is spread along several magnitudes (note the logarithmic scaling on the x-axis), with the best (lowest) objective function value of 9.8903, which is very close to the best known solution for this problem and therefore can be considered as global minimum

of the objective function. Although the local optimization was enhanced by high quality Jacobian information based on the sensitivity calculations, only 5% of the runs achieved the vicinity of the global optima.

The calibration data and the simulation results of the most frequently occurring local optima (marked as LS in the histogram; objective function value: 148.25) is shown in Figure 5.1 C. This is certainly a potential pitfall of using local optimization, which can lead to wrong conclusions about the model predictive capability. In contrast, the fit of the global solution (marked as GS in the histogram) is depicted in Figure 5.1 B, showing a good agreement between the model and the data.

We applied the same procedure to the other case studies, with the corresponding histograms shown in Figure 5.2. These histograms show that all these case studies exhibit multi-modality, but in different degree. We can see that oscillators tend to exhibit more local minima than the other types. However, case study TSMP, which does not exhibit oscillations, presents a particularly challenging histogram: none of the local searches was able to locate the vicinity of the global solution.

In summary, some of these problems could in principle be solved by a multi-start local method, especially if using high quality gradients. But this approach would fail in other cases, and we have no a priori way of distinguishing between these two groups. Therefore, we conclude that an efficient global optimization approach should be used in all cases to avoid artifices (local solutions) and ensure the best possible fit.

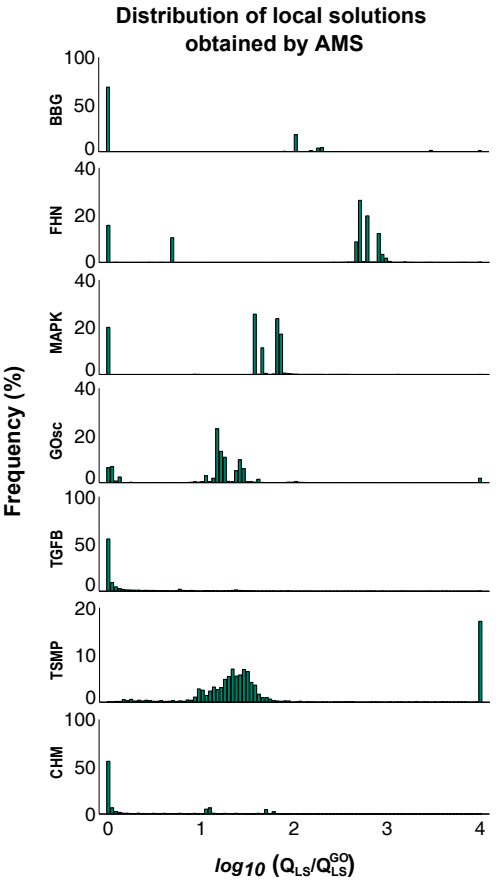


Figure 5.2: Distributions of local optima for all case studies. Each case study was solved by the AMS method and the observed frequency of the local minima is reported here. The objective function values (Q_{LS}) are scaled by the global optimum $Q_{\mathrm{LS}}^{\mathrm{GO}}$ for each case study, and the resulting ratio is reported in logarithmic scale. The height of the first bin at 0 represents the frequency of finding the vicinity of the global solution.

5.4.3 Convergence of the optimization algorithms

Once we have characterized the multi-modality of the case studies, we now illustrate the advantages of using the eSS2 global optimization method presented previously. First we consider the solutions of the non-regularized calibration problems (5.6), and then in the following subsection we will discuss the regularized estimations (5.7). The metric to be used will be based on the convergence curves, i.e. cost function values versus computation time. In order to evaluate the improvements in efficiency and robustness, we will compare the following methods for all the case studies, using a fair stopping criteria based on when a predefined computational time budget is reached:

SMS : simple multi-start (SMS) of NL2SOL with finite difference Jacobian computation.

AMS : advanced multi-start (AMS), similar to SMS, but the bounds of the feasible range

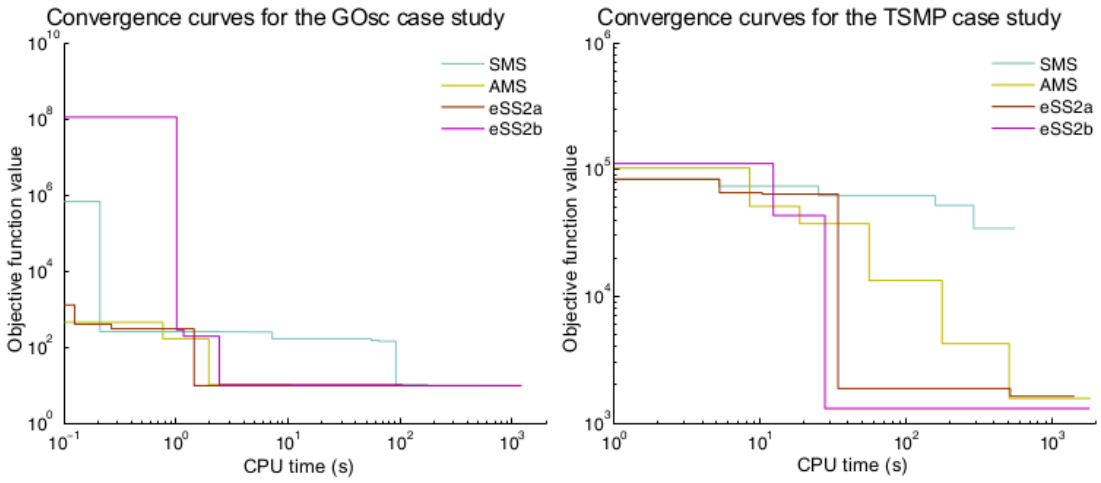


Figure 5.3: Comparison of convergence curves of selected optimization methods. The curves show the objective function versus the computation time during the minimization. Results are given for simple multi- start (SMS), advanced multi-start (AMS) and enhanced scatter search methods (eSS2a and eSS2b; see description in main text). Results are shown for two case studies: (a) GOsc and (b) TSMP.

of the parameters are transformed by the logarithmic function and then the Latin hypercube sampling method is utilized to sample initial points. This way the parameter space is better sampled, especially if the upper and lower bounds of some parameters have very different magnitudes (which is the case for all case studies). Further, NL2SOL is provided with high quality Jacobian of the residual vector.

eSS2a : the new enhanced scatter search described above, making use of NL2SOL and high quality Jacobian.

eSS2b : like eSS2a but initialized by the log Latin hypercube sampling as in AMS.

The above methods are compared based on their convergence curves (see for example Figure 5.3) and the distribution of the final cost function values reached (reported in Appendix D.6).
The empirical convergence curve depicts the current best objective function value as the optimization algorithm proceeds. An optimization method performs better than another method if a lower objective function value is reached in the same amount of computation time. Since both the multi-start and the eSS2 approaches use random numbers, the convergence curves will be different for each run. Thus we need to compare the convergence curves for several runs of each method.

Figure 5.3 shows the convergence curves for the Goodwins' oscillator case study (GOsc) and for the 3-step metabolic pathway problem (TSMP). For each method the optimization was carried out 20 times using different seeds for the random number generator, but here only the best convergence curve is shown, i.e. the run in which the best solution was reached in the shortest time by each method. Detailed results of the 20 runs can be found in Appendix D.6 for all case studies. Clearly, the simple multi-start (SMS) approach performed poorly in both cases: in GOsc, SMS needed 50 times more computation time than eSS2 to achieve the vicinity of global minimum, while in TSMP it could not find it

in the given computation time budget. The advanced multi-start (AMS) presented a performance similar to eSS2a and eSS2b for the GOsc case study, but in TSMP it was clearly outperformed by eSS2b.

Considering the results for all the case studies (see detailed convergence curves in Appendix D.6), we can conclude that the more refined version of multi-start can solve problems of small size (number of parameters) and with relatively tight bounds and good initial guesses for the parameters, but it is not reliable in more difficult situations. In contrast, the eSS2b method performed consistently well, solving all the problems in reasonable computation time using its default options (i.e. without the need of tweaking the method's search options with preliminary runs). In the remaining text we will refer to eSS2b as eSS2.

The effect of regularization on the convergence. We now consider how the penalty regularization (5.7), which changes the topology of the objective function, affects the convergence of the optimizer. We used eSS2 to solve the regularized problem for each case study, finding a narrower spread of the convergence curves. We also found improvements in the average time to reach the global solution. This benefit was especially clear in the TSMP case study, where the robustness was greatly improved (all the 20 runs of the optimization with regularization reached the global optima in 200 seconds of computation time, while only 3 runs converged using the same algorithm with the non-regularized objective function). Detailed results for all case studies are reported in Appendix D.6.

This additional beneficial effect of regularization on the convergence can be explained as follows: while the original cost function is multi-modal, the penalty term in Tikhonov regularization (5.8) is a quadratic (convex) function. Thus, in the limit $\alpha \to \infty$ the regularized objective function becomes a convex function.

Note that, the global minimum of the objective function is always larger for the regularized problem ($Q_{\mathrm{R}}(\hat{\theta}_\alpha)$ in (5.7)) than the value for the non-regularized problem ($Q_{\mathrm{LS}}(\hat{\theta})$ in (5.6)). This is because the penalty term ($\alpha\Gamma(\hat{\theta}_\alpha)$) contributes only to the objective function in (5.7). Further, the regularization avoids overfitting the data, thus the sum of squared residuals part of the objective function ($Q_{\mathrm{LS}}(\hat{\theta}_\alpha)$), is also larger than the minimum of the non-regularized solution ($Q_{\mathrm{LS}}(\hat{\theta})$).

5.4.4 Tuning the regularization and prior knowledge

Kinetic parameters of bio-models are generally unknown and vary for different cells. Thus, even if we have some prior knowledge about the parameters, it should be tested against the data. As shown later in Section 5.4.6, the predictions of the calibrated models using good prior knowledge in the regularization agree with the cross-validation data and thus generalize better.

In order to adjust the right level of the regularization, the regularization parameter (α) has to be tuned. The general pipeline is depicted in Figure 5.4. The tuning includes three steps (TS):

TS1: *a set of regularization parameter candidates are determined*: α_1, α_2, $\ldots \alpha_I$. To cover large range with few elements, typically the candidates are determined as the elements of a geometric series, i.e. $\alpha_n = \alpha_0 \cdot q^n$ for $n = 1 \ldots I$, where $\alpha_0 > 0$ and $0 < q < 1$.

TS2: *the regularized calibration problem* (5.7)-(5.8) *is solved* for each regularization parameter. This results in a set of calibrated models (candidate models), with estimated parameters denoted by $\hat{\theta}_{\alpha_1}$, $\hat{\theta}_{\alpha_2}$ $\ldots$, $\hat{\theta}_{\alpha_I}$.

TS3: *the best candidate is selected* based on a tuning method:
$\{\hat{\theta}_{\alpha_1}, \hat{\theta}_{\alpha_2}, \ldots, \hat{\theta}_{\alpha_I}\} \rightarrow \hat{\theta}_{\alpha_{opt}}$

In TS1, the range $(10^{-3} - 10^{3})$ with $I = 11$ candidates was found to be a good balance between accuracy and computational cost for all the case studies considered. In TS2, the calibration problems with different candidates can be solved parallel, since they are essentially independent optimization problems. However, when solved sequentially, the previously obtained solutions can be used to start the next optimization problem from a good initial point, and thus reduce its computational cost.

Figure 5.5A depicts trade-off between the model fit and regularization penalty for the candidates in the biomass batch growth (BBG) case study. Each cross in the figure corresponds to a calibrated model with the regularization parameter denoted by the labels next to the crosses. Larger regularization parameter results in an estimated parameter vector closer to the reference parameter vector and therefore smaller penalty, but worse fit to the calibration data.

The best way to select the optimal candidate in TS3 is cross-validation [154], but it requires an independent set of data at the time of calibration. However, in general it is unclear how the total amount of data should be divided [155] into a calibration and validation set for regularization. In case of scarce data, where the splitting is undesirable, tuning methods must be used.

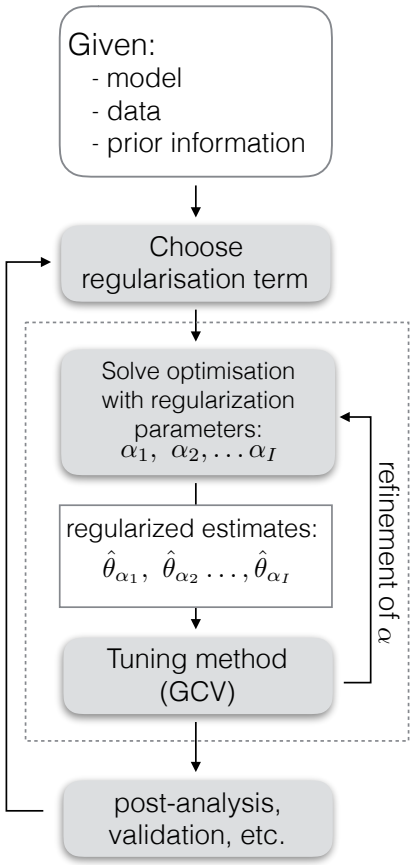


Figure 5.4: Regularized parameter estimation pipeline

We have tested 15 tuning methods on the case studies by comparing the regularization parameter selected by each tuning method with the optimal regularization parameter which minimizes the prediction error (i.e. the one with the best bias-variance trade-off, which is known for synthetic problems). The optimal regularization parameter and the regularization parameters selected by the tuning methods are reported in Appendix D.8 for each case study. We found the (robust) generalized cross validation method as the most reliable, since it identified the optimal regularization parameter reliably, outperforming the other methods.

The generalized cross-validation method does not use any further cross-validation data, but estimates the leave-one-out cross validation error of the candidate models based on the calibration data. The criteria is computed as

$$\mathrm{GCV}(\alpha_i) = \frac{\mathrm{RSS}(\alpha_i)}{N_\mathrm{D} - N_\theta^{\mathrm{eff}}(\alpha_i)}, \quad \text{for } i = 1, \ldots I \tag{5.10}$$

where $\mathrm{RSS}(\alpha_i)$ is the sum of squared normalized residuals for the candidate ($\mathrm{RSS}(\alpha_i) = R(\hat{\theta}_{\alpha_i})^T R(\hat{\theta}_{\alpha_i})$), N_D is the number of calibration data and $N_\theta^{\mathrm{eff}}(\alpha)$ is the effective number of fitted parameters in the model calibration (D.7). The $\mathrm{RSS}(\alpha)$ grows with α since larger regularization results in a worse fit to the data (see Figure 5.5A). The larger the α is, the more the fitted parameters are constrained by the reference parameter vector, thus the

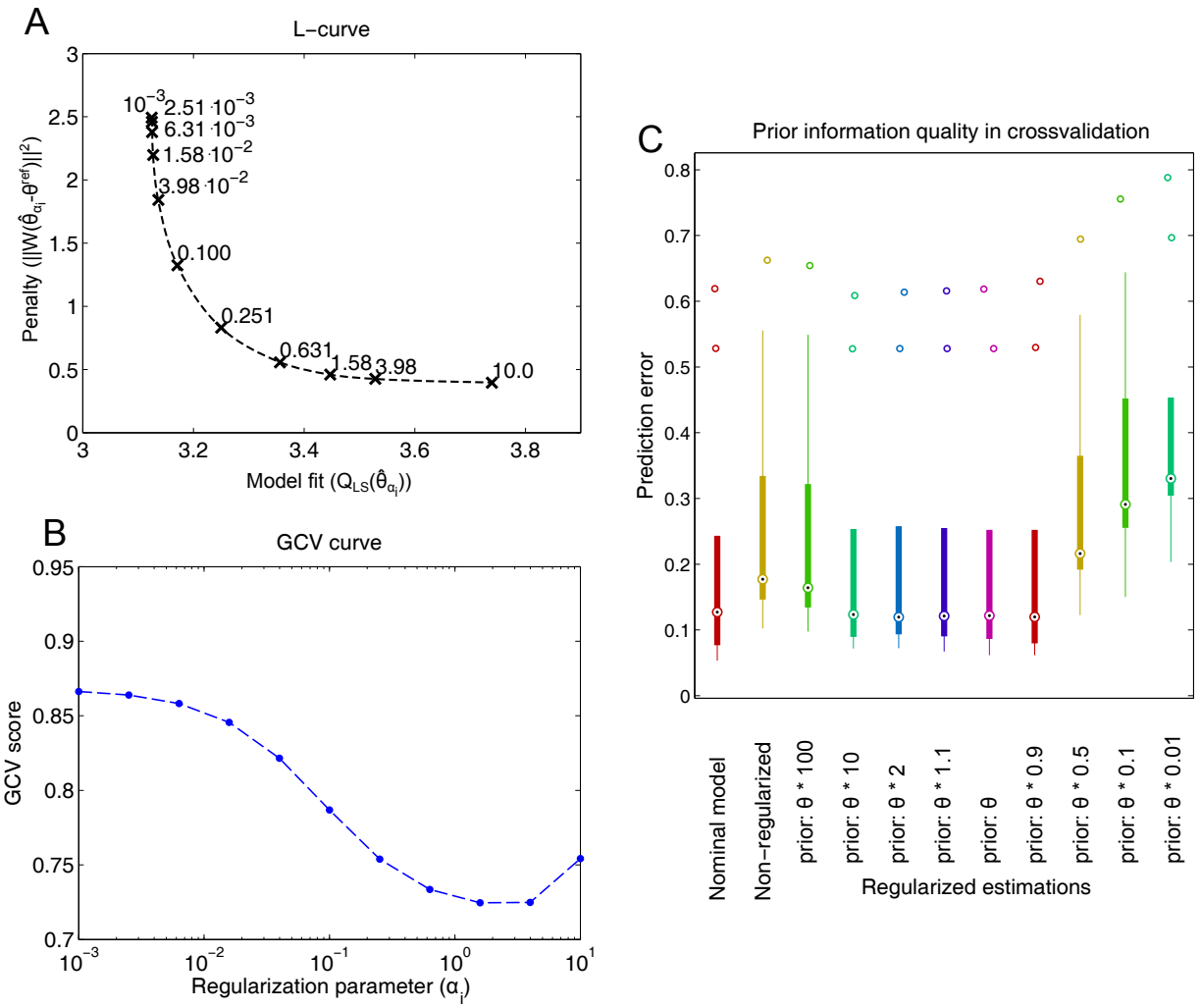


Figure 5.5: Tuning the regularization method for BBG case study. Figure A shows the trade-off between the two terms of the regularized objective function, i.e. model fit and the regularization penalty, for a set of regularization parameters (values shown close to symbols). A larger regularization parameter results in worse fit to the calibration data, small regularization parameter results in a larger penalty. Figure B compares the candidates based on the generalized cross-validation scores. A larger score indicates worse model prediction for cross-validation data. Figure C shows the normalized root mean square prediction error of calibrated model for 10 sets of cross-validation data and regularization considering different quality of the prior information (initial guess of the parameters). For a wide range of priors (initial guesses based on the reference parameter vector) the regularized estimation gives a good cross-validation error. Small priors exhibit worse predictions.

effective number of fitted parameters decreases with α (see Eq. (D.7)). The generalized cross validation error is small if the model fits the data well, while it also has a low number of effective parameters. Figure 5.5B shows the computed GCV value for the candidates in the BBG case study. It shows a minimum for the regularization parameter 1.58. Note that in cases where the amount calibration data is small, the GCV method tends to under-regularize the calibration [156], so the robust GCV (RGCV) method was found to be a better alternative.

The quality of the regularized calibration depends not only on the regularization parameter, but also on the prior knowledge of the modeller encoded by the reference parameter vector θ^{ref} and scaling matrix W. To test the robustness of the method with respect to these input information, we chose a range of reference parameter vectors and scaling matrices and solved the regularized optimization problem for each case study. In each case the generalized cross-validation score was used to select the regularization parameter. Then, the calibrated models were tested by computing predictions for cross-validation data sets. Figure 5.5C depicts the results for the BBG case study using box-plots. The first two columns show the distribution of the prediction error (normalized root mean square error) for the nominal model (known only in synthetic problems and used only for reference) and for the model calibrated without regularization. The next 9 columns in the plot show the prediction error with different quality of prior knowledge. We can see that the regularization method gives better predictions than the non-regularized for a wide range of prior quality.

5.4.5 Prediction and parameter bias-variance trade-off

Here we consider the stability of the solution of the optimization problem with respect to small perturbation in the data. Note that this numerical analysis is partially based on the bias-variance decomposition of the estimated model predictions and estimated parameters, thus it requires the knowledge of the nominal (true) parameter vector. Obviously the true model is known only for synthetic problems, but it can be used as a way to analyse the reliability of computational methods.

The experimental data is always measured with some uncertainty, which also influences the model calibration. If we could repeat the experiments, for example 10 times, taking measurements in the same conditions, we could collect 10 different datasets with slightly varying measurements –due to the random measurement error. Then each of the 10 datasets could be used to calibrate a model with and without regularization, which would result 10 slightly different calibrated models for both the non-regularized and regularized calibration procedure. Analysing the consistency of these models can reveal the sensitivity of the calibration procedure to the measurement error.

The results of this procedure for the BBG numerical case study can be seen in Figure 5.6A and 5.6B, where the nominal model predictions are shown by dashed line together with the range of the measured data depicted by error bars. In Figure 5.6A the predictions of the models, calibrated in the traditional way –without the regularization– is also shown, in contrast, the models shown in Figure 5.6B were calibrated using regularization. We can observe that the model predictions are less sensitive to the error in the data when regularization is applied, i.e. the variance of the model predictions are smaller. However, we also observe larger bias from the nominal trajectory for the regularized models, since no prior knowledge was used in this case (worst case scenario).

Figure 5.6C shows the prediction bias-variance trade-off for a range of the regularization parameter. The results are in agreement with the intuition that a lower regularization results in larger prediction variance and less bias. The mean squared error curve (the red dashed line), i.e. the sum of squared bias and variance, has the minimum for the regularization parameter $\alpha_{\text{opt}}^{\text{Pred}} \approx 0.04$, which is therefore the optimal regularization with respect to the prediction error.

Similar trends and results were obtained regarding the estimated parameters, shown in Figure 5.6D and 5.6E. Here, the distribution of the parameter estimates in the 10 regularized and 10 non-regularized calibrations are depicted by box-plots and the grey boxes show the feasible range of the parameters. The regularized calibration results in

much narrower distribution for the estimates (note the logarithmic scaling of the y-axis). The bias-variance trade-off in the estimated parameters is shown in Figure 5.6F. The optimal regularization parameter for the *minimum mean squared parameter estimation error* ($\alpha_{\text{opt}}^{\text{Param.}} \approx 0.04$) coincides with the previously obtained value for the *minimum mean square prediction error* in this case study. Although for all case studies we found that $\alpha_{\text{opt}}^{\text{Pred}}$ and $\alpha_{\text{opt}}^{\text{Param.}}$ are close to each other, they do not necessarily coincide.

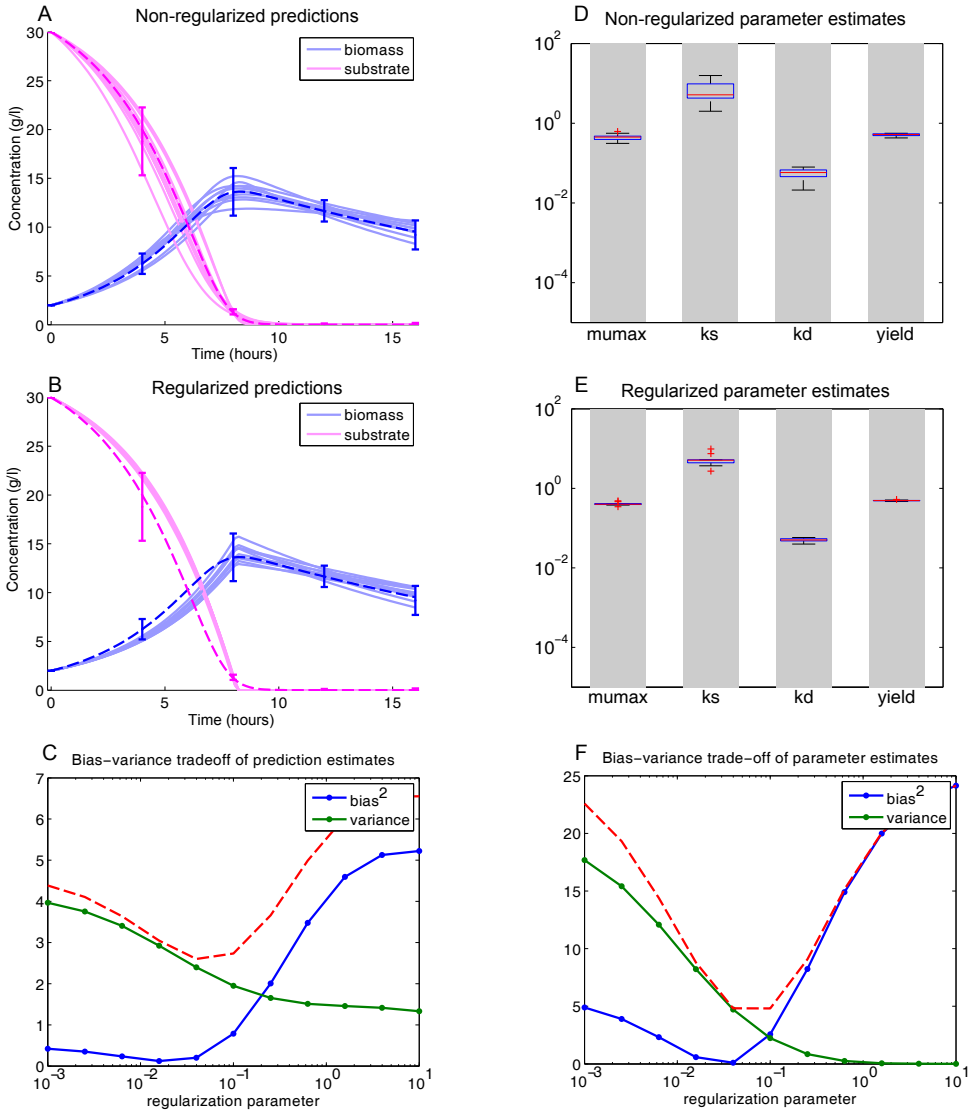


Figure 5.6: Bias-variance trade-off for the BBG case study. Figures A and B illustrate the nominal trajectory (dashed line) and the range of perturbed measurements together with predictions of calibrated models (continuous lines) without and with regularization, respectively. The distribution of the regularized predictions (in B) are narrower than in the non- regularized one (in A), but are slightly biased from the nominal trajectory. Figure C depicts the squared bias and the variance of these model predictions as a function of the regularization parameter. The mean square error (dashed line) has a minimum at 0.08. Figure D, E and F shows the results for the estimated parameters: with regularization the estimated parameters are less sensitive to perturbations in the data.

5.4.6 Ill-conditioning, cross-validation and overfitting

It is a common problem that due to the large measurement error (large noise to signal ratio) and due to data scarcity, a model with different numerical parameter values might fit the data almost equally well, which indicates identifiability problems.

A posteriori to the calibration, local analyses of the topology of the objective function can provide valuable information about the uncertainty in the estimated parameters. Particularly, the ill-conditioning of the approximated Hessian of the objective function (Eq. (D.6)) evaluated at the global optima can indicate high uncertainty in the estimated parameters [157]. Figure 5.7 shows the eigenvalues of this matrix for each case study. We can see that larger models with more parameters tend to have larger a spread in the eigenvalues, and thus larger condition number, indicating the lack of identifiability of its parameters. However, this is only a local measure of the ill-conditioning of the problem near the optima.

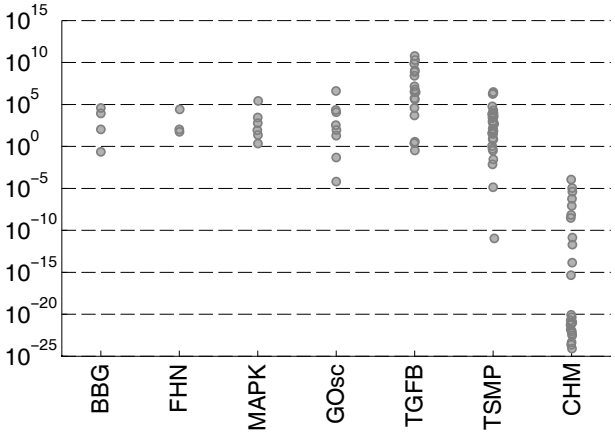


Figure 5.7: Eigenvalues of the approximated Hessian matrix for each case study. Eigenvalues are related to the identifiability of the model parameters: a large spread indicates lack of identifiability of some parameters from the given dataset.

A more sound way to measure the predictive value (generalizability) of the calibrated model is cross-validation, where a different set of data is used to asses the calibrated model. Over-fitted models will show a bad fit to cross-validation data since they fitted the noise, rather than the signal, and therefore are less generalizable. If the experimental conditions for collecting the cross-validation data are different from the calibration conditions –e.g. different stimuli levels, time-horizon etc.–, this effect will be more prominent.

Figure 5.8 shows the calibration fit (on the left) and the cross-validation (on the right) for the BBG case study (substrate measurements are not shown). The predictions of two models, one that was calibrated in the traditional way and one that was calibrated with regularization are also presented. Although there is almost no difference between the model predictions for the calibration data, the predictions for cross-validation data are rather different. The model that was calibrated without regularization predicts a slower decrease in the biomass concentration and shows large discrepancy from the cross validation data. If we compare the least-squares cost function for the two models, we find that the non-regularized model fits better the calibration data, but the regularized model generalizes better for the cross-validation data. In other words, the traditional model calibration results in overfitting, while the regularized calibration gives a more generalizable model at the expense of a slightly worse fit to the calibration data. Ideally, the cross-validation experimental scenario should be different from the calibration one in order to better assess generalizability of a model. Typically this can be achieved generating cross-validation data with experiments where the initial and boundary conditions (e.g. stimuli) of the experiments are as different as possible from those used to obtain the calibration data.

In Figure 5.9 we present similar results for the Goodwin's oscillator case study (GOsc). Here, we already see larger differences between the model predictions in the calibration data, but note that the predictions are almost identical at the time of the measurements.

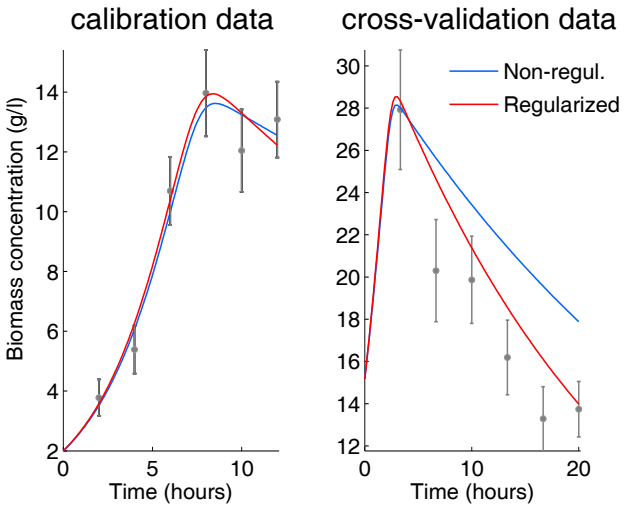


Figure 5.8: Calibration and cross-validation results for the BBG case study. Left figure shows calibration data fitted with non-regularized and regularized estimations (non-regularized, $Q_{\mathrm{LS}}(\hat{\theta}) = 3.68$ and regularized $Q_{\mathrm{LS}}(\hat{\theta}_\alpha) = 4.09$). Right figure shows cross-validation data with the predictions from the non-regularized and regularized estimations. The regularized model shows a slightly worse fit to the calibration data but much better agreement with the cross-validation data. I.e. regularization results in a more generalizable model.

Thus, for example, based on the calibration data it would be impossible to decide whether the protein concentration decreases or increases right after the beginning of the experiment. When the two models are cross-validated on an independent set of data (lower plot in Figure 5.9) we see that the regularized model is in good agreement with the new data, while the non-regularized model heavily overshoots the data in the first period of the oscillation.

Figure 5.10 summarizes our findings for all case studies regarding the generalizability of the calibrated models. Each case study was solved in the traditional, non-regularized way and with regularization assuming different level of prior knowledge (worst, medium and best case scenarios). Due to the low number of calibration data and large measurement noise, we found large variability of the predictions depending on the exact noise realization in the calibration data. Thus we repeated the calibrations with 10 calibration datasets to obtain robust results. Then, each calibrated model was cross-validated in 10 independent cross-validation data sets and the prediction error was computed. Figure 5.10 shows the distribution of these prediction errors for each case study by box-plots.

The distributions can be compared by the ob-

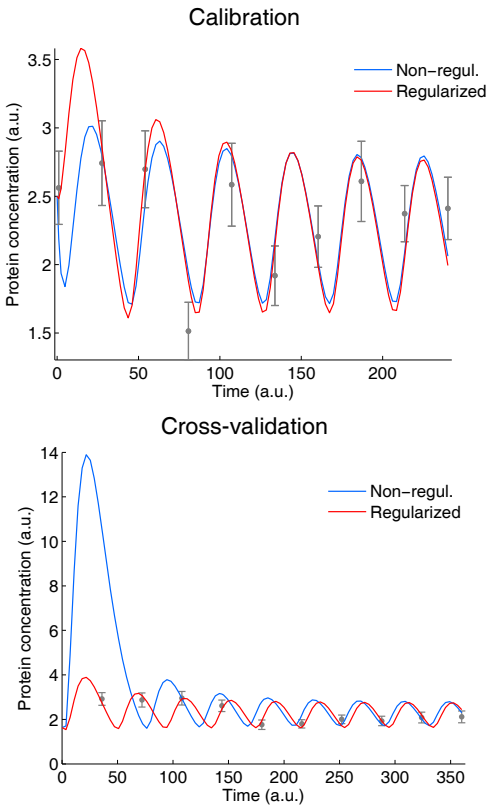


Figure 5.9: Calibration and cross-validation results for the GOsc case study. Top figure: fits to calibration data, bottom: predictions for cross-validation data.

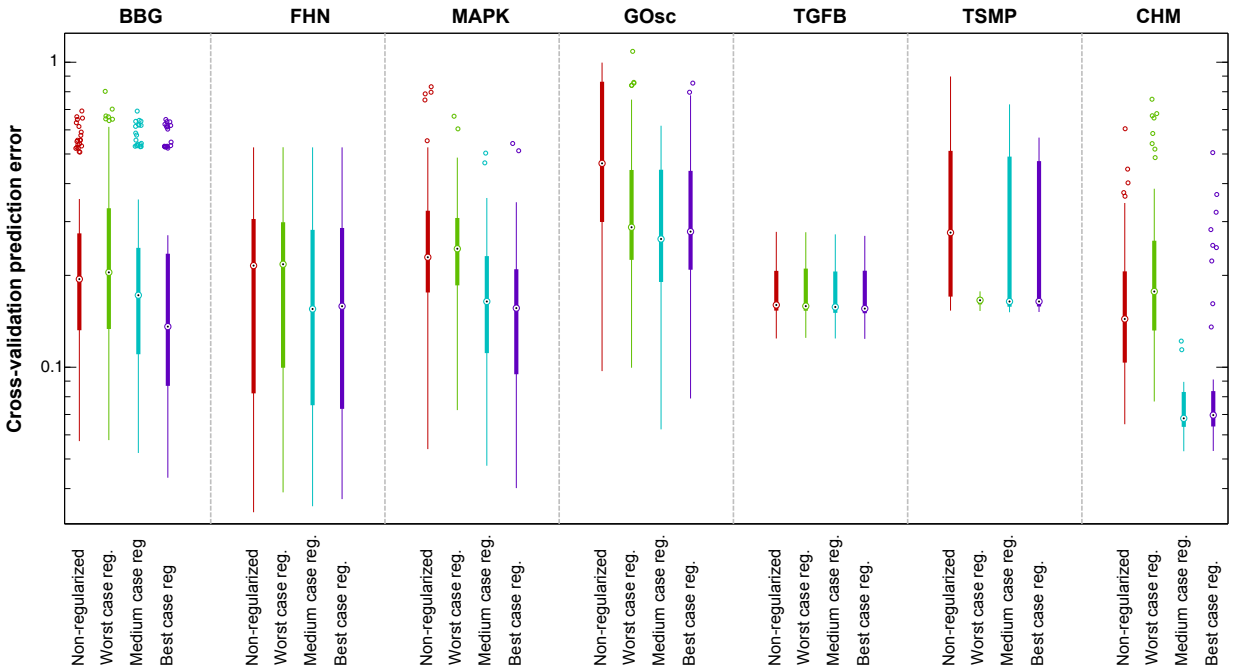


Figure 5.10: Prediction errors distribution for each case study. Prediction errors (box-plots of normalized root mean square error in log-scale) of the calibrated models with and without regularization are shown for each case study. These distributions were obtained by calibrating the models to multiple sets of calibration data (as explained in section 3.6) and cross-validating them on multiple cross-validation data sets. Most cases show the trend that better prior knowledge results in smaller cross-validation errors, i.e. regularized models are more generalizable.

served medians, which are denoted by the black dots in the box-plot. In order to check if the observed differences in the medians are significant we utilized the Wilcoxon non-parametric statistical test [158] (also known as the Mann-Whitney U test). The test results show that in the majority of the scenarios the differences in the medians are statistically significant at the 0.05 level. The exception is the FHN case study where the differences turned out to be not significant for the three scenarios. Further details of this statistical test are reported in Appendix D.7 and in Table D.4.

By comparing the medians of the distributions we see that in almost all cases the non-regularized models overfit the calibration data, i.e. the non-regularized models fit well the calibration data, but do not predict cross-validation data as well as the regularized models. In each case, the medium and the best case regularization scenarios clearly outperformed the non-regularized estimation, leading to better generalizable calibrated models. However, in two cases we observe that the worst case regularization scenario performed worse than the non-regularized case. Also note, that in case of the TGF-β pathway problem (TGFB) all scenarios gave almost identical results, meaning that the original problem is a well-posed calibration problem. However, this is generally unknown before the calibration.

In this context, it is worth mentioning that the regularization of non-mechanistic (e.g. data-driven) models –like those used in machine learning and system identification, such as e.g. neural networks– usually exhibits more dramatic benefits. The reason is that these data-driven models are by definition extremely flexible and therefore very prone to overfitting. In the case of the mechanistic kinetic models used in systems biology, in many cases they

will have a rather rigid structure despite being over-parametrized. Therefore, they might be less prone to overfitting. However, a clear exception are models exhibiting oscillatory behaviour, or models with many non-observable states.

5.4.7 Regularization schemes based on available information

Based on the above results, we recommend the following regularization procedures for the three scenarios defined previously (in Section 5.3.2):

I **Best case:** a good guess of the parameter values (θ^{guess}) is available. In this case a first order weighted Tikhonov regularization is recommended, i.e. $\theta^{\text{ref}} := \theta^{\text{guess}}$ and the weighting matrix should be initialized by the parameters too, i.e. $W = \text{diag}(1./\theta^{\text{ref}})$, where ./ is the element-wise division. In this way, parameters with different magnitudes will contribute similarly to the penalty.

II **Medium case:** a situation where a less reliable initial guess –but within one order of magnitude of the true values– is available. As in the best case scenario, the parameter guess should be used as the reference vector in the regularization penalty: $\theta^{\text{ref}} := \theta^{\text{guess}}$. However, we found, that including these values also in the weighting matrix amplified the error in the parameter estimate. Therefore, the non-weighted Tikhonov regularization is recommended.

III **Worst case:** no prior knowledge and therefore only a random guess of parameters is available. Here a two-step regularization procedure is proposed. In the first step ridge regularization is applied which results in a ridge estimate, denoted by $\hat{\theta}_{\alpha}^{\text{R1}}$. In the second step this parameter vector is used as the reference parameter vector for Tikhonov regularization, i.e. ($\theta^{\text{ref}} := \hat{\theta}_{\alpha}^{\text{R1}}$). This procedure could be repeated n-times – using the obtained regularized solution as reference parameter vector in the next step–, resulting in the n-th order Tikhonov regularization [129], but we found no practical difference after the second step.

The regularized optimization is solved for a set of regularization parameters in each scenario and depending on the amount of data at hand the generalized cross validation method (GCV) – for larger dataset– or the robust generalized cross-validation method (RGCV) – for smaller dataset– is recommended to choose the optimal candidate. A summary of this regularization scheme is illustrated in Figure D.1 in the Appendix.

Based on the results presented previously, we suggest that tuning of the regularization can be avoided in certain situations, saving considerable computation time. For scaled models where the number of data points and parameters are similar and the data has 5-10 % measurement error, our study indicates that the optimal regularization parameter will lie in the range $[0.1-10]$. For the worst case scenario, rather common in systems biology, we found that the above procedure gave smaller mean square parameter estimation error than the traditional, non-regularized estimation. Further, the optimization algorithm exhibited better convergence properties with regularization, although no significant improvements in the model predictions was observed. In the case of medium and best scenarios *regularized estimation led to both better parameter estimates and smaller cross-validation prediction error in shorter computation times.*

5.4.8 Regularization pipeline and implementation

In Figure 5.11 we outline the architecture of the resulting method and its implementation. The pre-processing step makes use of symbolic manipulation to derive the sensitivity equa-

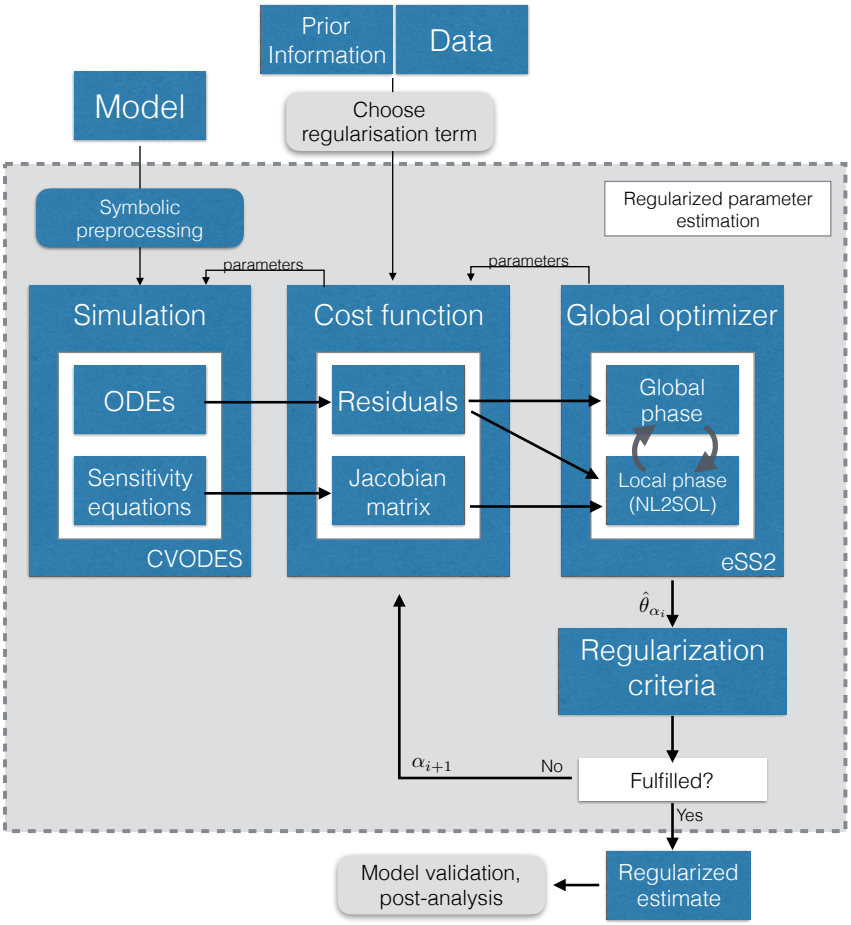


Figure 5.11: Architecture of the method: in the pre-processing phase, sensitivity equations and Jacobians (both of the residuals and of the differential equations) are derived via symbolic manipulation, generating C code which is then linked to the initial value problem (IVP) solver, CVODES. The regularization scheme is selected according to the quality of the prior knowledge, and tuned following the procedure described in section 5.4.4. Finally, global optimization with eSS2 is used to find the regularized estimate of the parameters. The resulting calibrated model can then be further evaluated using cross-validation, followed by additional post-regression and goodness-of-fit analysis.

tions and Jacobians (both of residuals and dynamics), and automatically generates C code to be compiled and linked with the CVODES solver. This procedure ensures the highest numerical efficiency and stability during the solution of the initial value problem.

The regularization scheme (term) is selected according to the quality of the prior knowledge (as described in Section 5.4.7 and illustrated in Figure D.1 in the Appendix), and the cost function is formulated. The regularized optimization problem is solved for a set of regularization parameter α_i. In each iteration the result is passed to the regularization tuning modul (regularization criteria). When the optimal regularization parameter is found, the parameter estimates are reported and the calibrated model is analysed, validated.

We evaluated the above pipeline in different scenarios:

1. we showed that parameter estimation in kinetic models leads to a non-convex optimization problem with several local minimas (in Section 5.4.2), thus we need a global optimization algorithm.

2. we compared 2 types of multiple start local methods with the global optimizer eSS2 in Section 5.4.3 and showed the improvement in the convergence to the global optima.
3. we showed that the calibration of the case studies led to solve ill-conditioned optimization problems (Section 5.4.6), thus these models are prone to overfit the data.
4. we applied regularization to condition the optimization problems and we compared 15 tuning methods (regularization criteria) based on cross-validation to select the most appropriate method (GCV), (Section 5.4.4).
5. regularization requires a reference parameter vector, which introduce bias to the estimation. We evaluated the effect of a miss-specified reference vector on the estimation in Section 5.4.4 and recommended different regularization schemes (Subsection 5.4.7), based on the quality of the prior information.

The pipeline is freely available through the AMIGO2 open source MALTAB Toolbox (https://sites.google.com/site/amigo2toolbox/download).

5.5 Summary

In this chapter we propose a new parameter estimation strategy for nonlinear dynamical models of biological systems. This strategy is especially designed to surmount the challenges arising from the non-convexity and ill-conditioning that most of these problems exhibit. The difficulties of parameter estimation problems in systems biology do not only depend on the number of parameters, but also on the structure (flexibility and nonlinearity) of the dynamic model, and the amount of information provided by the (usually scarce and noisy) available data.

Our strategy combines an efficient global optimization method with three different schemes of Tikhonov regularization, selected depending on the quality of the prior knowledge. We tested this strategy with a set of case studies of increasing complexity. The results clearly indicate that an efficient global optimization approach should always be used, even for small models, to avoid convergence to local minima. Similarly, our study illustrates how ill-conditioning and overfitting issues can damage the generalizability of the calibrated models. Overfitting was found to be especially important when models are flexible (e.g. oscillatory models), even if the number of parameters is small. Our results show how regularization can be used to avoid overfitting, leading to calibrated models with better generalizability. Finally, the use of regularization significantly improved the performance of the optimization method, resulting in faster and more stable convergence.

Chapter 6

Conclusions

6.1 New scientific contributions

The aim of this section is to summarize the main results of this work in thesis points as follows:

Thesis 1. *Bio-CRN framework and model verification (Chapter 2).* A new modeling framework was proposed for biochemical reaction networks (bio-CRNs), which can be seen as an extension of the well known mass action chemical reaction networks (MAL-CRNs). Beside of mass action kinetics, this formalism can handle reaction kinetics being in rational function form, for example Michaelis-Menten kinetics. Further, the dynamics of the reaction network was connected to a new complex–reaction graph representation, which allows multiple edges between complexes, i.e. each possible kinetics is represented by a weighted edge. The non-negativity of the dynamics of bio-CRNs was shown and further model verification steps were presented to identify potential pitfalls in existing models, such as to detect the violation of mass balance or non-plausible reaction rates.
Related publications: [159], [160]

Thesis 2. *Realizations, dynamically equivalent and linearly conjugated biochemical reaction networks (Chapter 3).* Kinetic ordinary differential equations with rational polynomials on their right hand side can be transformed into bio-CRNs, i.e. into the framework presented in Thesis 1. The necessary and sufficient conditions for network realizability were derived and a construction algorithm was presented. There might exist multiple networks corresponding to the same dynamics, which are called dynamically equivalent realizations. Optimization problems were formulated, which can find dynamically equivalent bio-CRN realizations with preferred properties, such as dense-, sparse- and reversible graph structure. Finally, the bio-CRN model class was shown to be closed to linear scaling of the state variables and linearly conjugated realization were defined. Mixed integer linear optimization problems were formulated to construct linearly conjugated networks for a given kinetic system.
Related publication: [159],[161]

Thesis 3. *Model reduction in biochemical reaction networks (Chapter 4).* A systematic and stable method was developed for kinetic, mechanistic models, which eliminates reactions from a given detailed network, such that the concentrations of some chosen, key

species remain close to the original trajectories. By exploiting the model structure, i.e. that the reaction rates are linear in the reaction rate parameters, the general mixed integer nonlinear optimization approach was approximated by a finite sequence of mixed integer quadratic optimization problems, which greatly reduces the computation cost. Further, in each reduction step the method re-estimates the remaining reaction rate parameters in the reduced model to achieve the smallest reduction error.
Related publications: [162, 163]

Thesis 4. *Regularized parameter estimation in dynamic biological models (Chapter 5).* A parameter estimation method was presented for general dynamic biological models in ordinary differential equation form. Non-convexity and ill-conditioning of these optimization problems were identified as the key challenges. Regarding the non-convexity, a hybrid optimization method was proposed, which combines existing global (enhanced scatter search) and local optimization methods (nonlinear least squares algorithm NL2SOL). In case of ill-conditioning, a regularization scheme was presented, which serves as a guideline for regularized parameter estimation depending on the quality of prior information. Several tuning methods for Tikhonov regularization were evaluated. The convergence properties of the proposed method and the improved prediction capabilities of the calibrated models were evaluated based on 7 numerical case studies. The parameter estimation pipeline was published as part of the AMIGO2 MATLAB Toolbox.
Related publication: [154, 164, 165]

6.2 Future work

The results presented in the above thesis points could be further extended in the following directions.

Biochemical reaction networks We have developed the bio-CRN framework such that the computational tools used in MAL-CRNs can be applied with reasonably small modifications. Therefore, a straightforward extension could be to formulate further optimization problems for finding realizations with minimum number of complexes such as in [75], and to find weakly reversible realizations [77, 166].

Model reduction in bio-CRNs. The model reduction method presented in Chapter 4 exploits the model structure of chemical reaction networks, i.e. that the reaction rate functions are linear in the reaction rate parameters. Therefore a reaction is eliminated from the network if and only if the reaction rate parameter is zero. A future extension of this method could be to apply it for bio-CRNs, since the reaction rates are also linear in the principal reaction rate coefficients and the reaction is also eliminated when the principal reaction rate coefficient is zero.

Bibliography

[1] Mogilner, A., Wollman, R., Marshall, W.F.: Quantitative modeling in cell biology: what is it good for? Developmental Cell **11**(3), 279–287 (2006)

[2] Epstein, J.M.: Why model? Journal of Artificial Societies and Social Simulation **11**(4), 12 (2008)

[3] Wolkenhauer, O.: Why model? Frontiers in Physiology **5**, 21 (2014)

[4] Lander, A.D.: The edges of understanding. BMC Biology **8**(1), 40 (2010)

[5] Wolkenhauer, O.: Systems biology: the reincarnation of systems theory applied in biology? Briefings in Bioinformatics **2**(3), 258–270 (2001)

[6] Stelling, J.: Mathematical models in microbial systems biology. Current Opinion in Microbiology **7**(5), 513–8 (2004)

[7] De Jong, H.: Modeling and simulation of genetic regulatory systems: a literature review. Journal of Computational Biology **9**(1), 67–103 (2002)

[8] Tyson, J.J., Chen, K.C., Novak, B.: Sniffers, buzzers, toggles and blinkers: dynamics of regulatory and signaling pathways in the cell. Current Oinion in Cell Biology **15**(2), 221–231 (2003)

[9] Kholodenko, B.N.: Cell-signalling dynamics in time and space. Molecular Cell Biology **7**(3), 165–176 (2006)

[10] Aldridge, B.B., Burke, J.M., Lauffenburger, D.A., Sorger, P.K.: Physicochemical modelling of cell signalling pathways. Nature Cell Biology **8**(11), 1195–1203 (2006)

[11] Doyle, F.J., Stelling, J.: Systems interface biology. Journal of the Royal Society, Interface **3**(10), 603–16 (2006)

[12] Schaber, J., Klipp, E.: Model-based inference of biochemical parameters and dynamic properties of microbial signal transduction networks. Current Opinion in Biotechnology **22**, 109–116 (2011)

[13] Sunnåker, M., Zamora-Sillero, E., Dechant, R., Ludwig, C., Busetto, A.G., Wagner, A., Stelling, J.: Automatic generation of predictive dynamic models reveals nuclear phosphorylation as the key msn2 control mechanism. Science Signaling **6**(277), 41–41 (2013)

[14] Link, H., Christodoulou, D., Sauer, U.: Advancing metabolic models with kinetic information. Current Opinion in Biotechnology **29**, 8–14 (2014)

[15] Le Novère, N.: Quantitative and logic modelling of molecular and gene networks. Nature Reviews Genetics **16**, 146–158 (2015)

[16] Baltes, M., Schneider, R., Sturm, C., Reuss, M.: Optimal experimental design for parameter estimation in unstructured growth models. Biotechnology Progress **10**(5), 480–488 (1994)

[17] Apgar, J.F., Toettcher, J.E., Endy, D., White, F.M., Tidor, B.: Stimulus design for model selection and validation in cell signaling. PLoS Computational Biology **4**(2), 30 (2008)

[18] Banga, J.R., Balsa-Canto, E.: Parameter estimation and optimal experimental design. Essays in Biochemistry **45**, 195–210 (2008)

[19] Bandara, S., Schlöder, J.P., Eils, R., Bock, H.G., Meyer, T.: Optimal experimental design for parameter estimation of a cell signaling model. PLoS Computational Biology **5**(11), 1000558 (2009)

[20] Hasenauer, J., Waldherr, S., Wagner, K., Allgower, F.: Parameter identification, experimental design and model falsification for biological network models using semidefinite programming. IET Systems Biology **4**(2), 119–130 (2010)

[21] Almquist, J., Cvijovic, M., Hatzimanikatis, V., Nielsen, J., Jirstrand, M.: Kinetic models in industrial biotechnology - improving cell factory performance. Metabolic Engineering **24**, 38–60 (2014)

[22] Marchisio, M.A., Stelling, J.: Computational design tools for synthetic biology. Current Opinion in Biotechnology **20**(4), 479–485 (2009)

[23] Arpino, J.A.J., Hancock, E.J., Anderson, J., Barahona, M., Stan, G.-B.V., Papachristodoulou, A., Polizzi, K.: Tuning the dials of synthetic biology. Microbiology **159**(Pt 7), 1236–1253 (2013)

[24] Stanford, N.J., Lubitz, T., Smallbone, K., Klipp, E., Mendes, P., Liebermeister, W.: Systematic construction of kinetic models from genome-scale metabolic networks. PLoS ONE **8**(11), 79195 (2013)

[25] Karr, J.R., Sanghvi, J.C., Macklin, D.N., Gutschow, M.V., Jacobs, J.M., Bolival, B., Assad-Garcia, N., Glass, J.I., Covert, M.W.: A whole-cell computational model predicts phenotype from genotype. Cell **150**(2), 389–401 (2012)

[26] Macklin, D.N., Ruggero, N.A., Covert, M.W.: The future of whole-cell modeling. Current Opinion in Biotechnology **28**, 111–115 (2014)

[27] Cedersund, G., Roll, J.: Systems biology: model based evaluation and comparison of potential explanations for given biological data. The FEBS journal **276**(4), 903–922 (2009)

[28] Stumpf, M., Balding, D.J., Girolami, M.: Handbook of Statistical Systems Biology. John Wiley & Sons, Chichester, UK (2011)

[29] Cvijovic, M., Almquist, J., Hagmar, J., Hohmann, S., Kaltenbach, H.-M., Klipp, E., Krantz, M., Mendes, P., Nelander, S., Nielsen, J., Others: Bridging the gaps in systems biology. Molecular Genetics and Genomics **289**(5), 727–734 (2014)

[30] Horn, F., Jackson, R.: General mass action kinetics. Archive for Rational Mechanics and Analysis **47**(2), 81–116 (1972)

[31] Feinberg, M.: On chemical kinetics of a certain class. Arch. Rational Mech. Anal. **46**, 1–41 (1972)

[32] Szederkényi, G.: Computing sparse and dense realizations of reaction kinetic systems. Journal of Mathematical Chemistry **47**(2), 551–568 (2010)

[33] Szederkényi, G., Hangos, K.M.: Finding complex balanced and detailed balanced realizations of chemical reaction networks. Journal of Mathematical Chemistry **49**(6), 1163–1179 (2011)

[34] Bailey, J.E.: Complex biology with no parameters. Nature Biotechnology **19**(6), 503–504 (2001)

[35] Conradi, C., Flockerzi, D., Raisch, J., Stelling, J.: Subnetwork analysis reveals dynamic features of complex (bio)chemical networks. Proceedings of the National Academy of Sciences **104**(49), 19175–19180 (2007)

[36] Craciun, G., Feinberg, M.: Multiple equilibria in complex chemical reaction networks : extensions to entrapped species models. IEE Proc.-Syst. Biol. **153**(4), 179–186 (2006)

[37] Wang, L., Sontag, E.D.: On the number of steady states in a multiple futile cycle. Journal of Mathematical Biology **57**(1), 29–52 (2008)

[38] Otero-Muras, I., Banga, J.R., Alonso, A.A.: Exploring multiplicity conditions in enzymatic reaction networks. Biotechnology Progress **25**(3), 619–631 (2009)

[39] Shinar, G., Feinberg, M.: Structural sources of robustness in biochemical reaction networks. Science **327**(5971), 1389–1391 (2010)

[40] Shinar, G., Feinberg, M.: Design principles for robust biochemical reaction networks: what works, what cannot work, and what might almost work. Mathematical Biosciences **231**(1), 39–48 (2011)

[41] Karp, R.L., Millán, M.P., Dasgupta, T., Dickenstein, A., Gunawardena, J., Pérez Millán, M.: Complex-linear invariants of biochemical networks. Journal of Theoretical Biology **311**, 130–138 (2012)

[42] van der Schaft, A., Rao, S., Jayawardhana, B.: On the mathematical structure of balanced chemical reaction networks governed by mass action kinetics. SIAM Journal on Applied Mathematics **73**(2), 953–973 (2013)

[43] Neigenfind, J., Grimbs, S., Nikoloski, Z.: On the relation between reactions and complexes of (bio)chemical reaction networks. Journal of Theoretical Biology **317**(0), 359–365 (2013)

[44] Craciun, G., Pantea., C.: Identifiability of chemical reaction networks. Journal of Mathematical Chemistry **44**, 244–259 (2008)

[45] Szederkényi, G., Banga, J.R., Alonso, A.A., Szederkenyi, G.: Inference of complex biological networks: distinguishability issues and optimization-based solutions. BMC Systems Biology **5**(1), 177 (2011)

[46] Prill, R.J., Marbach, D., Saez-Rodriguez, J., Sorger, P.K., Alexopoulos, L.G., Xue, X., Clarke, N.D., Altan-Bonnet, G., Stolovitzky, G.: Towards a rigorous assessment of systems biology models: the dream3 challenges. PloS ONE **5**(2), 9202 (2010)

[47] Nemcová, J., Schuppen, J.H.: Realization theory for rational systems: The existence of rational realizations. SIAM J. Control Optim. **48**(4), 2840–2856 (2009)

[48] Llaneras, F., Picó, J.: Stoichiometric modelling of the cell metabolism. Journal of Bioscience and Bioengineering **105**, 1–11 (2008)

[49] Otero-Muras, I., Franco-Uría, A., Alonso, A.A., Balsa-Canto, E.: Dynamic multi compartmental modelling of metal bioaccumulation in fish: Identifiability implications. Environmental Modeling and Software **25**, 344–353 (2010)

[50] Angeli, D.: A tutorial on chemical network dynamics. European Journal of Control **15**, 398–406 (2009)

[51] Chellaboina, V., Bhat, S.P., Haddad, W.M., Bernstein, D.S.: Modeling and analysis of mass-action kinetics. IEEE Control Systems Magazine **29**, 60–78 (2009)

[52] West, D.B., Others: Introduction to Graph Theory vol. 2. Prentice Hall, Upper Saddle River (2001)

[53] Feinberg, M.: Chemical reaction network structure and the stability of complex isothermal reactors – i. the deficiency zero and deficiency one theorems. Chemical Engineering Science **42**(10), 2229–2268 (1987)

[54] Szederkényi, G.: Computational Methods for the Analysis of Nonnegative Polynomial Systems. Thesis for the degree "Doctor of the Hungarian Academy of Sciences" (2011)

[55] Hoops, S., Sahle, S., Gauges, R., Lee, C., Pahle, J., Simus, N., Singhal, M., Xu, L., Mendes, P., Kummer, U.: Copasi–a complex pathway simulator. Bioinformatics **22**(24), 3067–3074 (2006)

[56] Li, C., Donizelli, M., Rodriguez, N., Dharuri, H., Endler, L., Chelliah, V., Li, L., He, E., Henry, A., Stefan, M.I., Snoep, J.L., Hucka, M., Le Novère, N., Laibe, C.: Biomodels database: An enhanced, curated and annotated resource for published quantitative kinetic models. BMC Systems Biology **4**, 92 (2010)

[57] Hárs, V., Tóth, J.: On the inverse problem of reaction kinetics. In: Farkas, M., Hatvani, L. (eds.) Qualitative Theory of Differential Equations. Coll. Math. Soc. J. Bolyai, vol. 30, pp. 363–379. North-Holland, Budapest (1981)

[58] Johnston, M.D., Siegel, D.: Linear conjugacy of chemical reaction networks. Journal of Mathematical Chemistry **49:7**(7), 1263–1282 (2011)

[59] Hucka, M.e.a.: The systems biology markup language (sbml): a medium for representation and exchange of biochemical network models. Bioinformatics **19**(4), 524–531 (2003)

[60] Schmidt, H., Jirstrand, M.: Systems biology toolbox for matlab: a computational platform for research in systems biology. Bioinformatics **22**(4), 514–5 (2006)

[61] Hangos, K.M.: Engineering model reduction and entropy-based lyapunov functions in chemical reaction kinetics. Entropy **12**(4), 772–797 (2010)

[62] Fages, F., Gay, S., Soliman, S.: Automatic curation of sbml models based on their ode semantics. Technical Report July, Inria (2012)

[63] Kaleta, C., Richter, S., Dittrich, P.: Using chemical organization theory for model checking. Bioinformatics (Oxford, England) **25**(15), 1915–22 (2009)

[64] Murray, J.D.: Mathematical Biology I: An Introduction. Springer, New York, Berlin, Heidelberg (2002)

[65] Rao, S., van der Schaft, A., van Eunen, K., Bakker, B., Jayawardhana, B.: A model reduction method for biochemical reaction networks. BMC Systems Biology **8**(52) (2014)

[66] Haddad, W.M., Chellaboina, V.: Stability and dissipativity theory for nonnegative dynamical systems: {A} unified analysis framework for biological and physiological systems. Nonlinear Analysis: Real World Applications **6**, 35–65 (2005)

[67] Haddad, W.M., Chellaboina, V.S., Hui, Q.: Nonnegative and Compartmental Dynamical Systems. Princeton University Press, USA (2010)

[68] Fleming, R.M.T., Thiele, I.: Mass conserved elementary kinetics is sufficient for the existence of a non-equilibrium steady state concentration. Journal of Theoretical Biology **314**(0), 173–181 (2012)

[69] Érdi, P., Tóth, J.: Mathematical Models of Chemical Reactions. Theory and Applications of Deterministic and Stochastic Models. Manchester University Press, Princeton University Press, Manchester, Princeton (1989)

[70] Klipp, E., Herwig, R., Kowald, A., Wierling, C., Lehrach, H.: Systems Biology in Practice. Wiley-VCH Verlag GmbH, Weinheim (2005)

[71] Hangos, K.M., Szederkényi, G.: The Effect of Conservation on the Dynamics of Chemical Reaction Networks. In: Ydstie, B.E., Maschke, B., Dochain, D. (eds.) IFAC Workshop on Thermodynamic Foundations of Mathematical Systems Theory, July 13-16, Lyon, France, pp. 30–35 (2013)

[72] Famili, I., Palsson, B.O.: The convex basis of the left null space of the stoichiometric matrix leads to the definition of metabolically meaningful pools. Biophysical Journal **85**(1), 16–26 (2003)

[73] Hangos, K.M., Szederkényi, G., G.Szederkényi: The underlying linear dynamics of some positive polynomial systems. Physics Letters A **376**(45), 3129–3134 (2012)

[74] Sauro, H.M., Ingalls, B.: Conservation analysis in biochemical networks: computational issues for software writers. Biophysical Chemistry **109**(1), 1–15 (2004)

[75] Szederkényi, G., Hangos, K.M., Péni, T.: Maximal and minimal realizations of reaction kinetic systems: computation and properties. MATCH Communications in Mathematical and in Computer Chemistry **65**(2), 309–332 (2011)

[76] Wolsey, L.A., Nemhauser, G.L.: Integer and Combinatorial Optimization, 1st edn. John Wiley & Sons, Inc, New York, USA (1988)

[77] Johnston, M.D., Siegel, D., Szederkényi, G.: Dynamical equivalence and linear conjugacy of chemical reaction networks: new results and methods. MATCH Commun. Math. Comput. Chem. **68**, 443–468 (2012). arXiv:1111.1441v1

[78] Császár, A., Jicsinszky, L., Turányi, T.: Generation of model reactions leading to limit cycle behaviour. React. Kinet. Catal. Lett **18**(1-2), 65–71 (1981)

[79] Toth, J., Li, G., Rabitz, H., Tomlin, A.S.: The effect of lumping and expanding on kinetic differential equations. SIAM Journal on Applied Mathematics **57**(6), 1531–1556 (1997)

[80] Lall, S., Marsden, J.E., Glavaski, S.: A subspace approach to balanced truncation for model reduction of nonlinear control systems. International Journal of Robust and Nonlinear Control **12**(6), 519–535 (2002)

[81] Anderson, J., Chang, Y.-c., Papachristodoulou, A.: Model decomposition and reduction tools for large-scale networks in systems biology. Automatica **47**(6), 1165–1174 (2011)

[82] Gorban, A.N., Karlin, I.V., Zinovyev., A.Y.: Invariant grids for reaction kinetics. Physica A **33**, 106–154 (2004)

[83] Androulakis, I.P.: Kinetic mechanism reduction based on an integer programming approach. AIChE Journal **46**, 361–371 (2000)

[84] Vajda, S., Valko, P., Turányi, T.: Principal component analysis of kinetic models. Int. Journal of Chemical Kinetics **17**, 55–81 (1985)

[85] Turányi, T.: Applications of sensitivity analysis to combustion chemistry. Reliability Engineering and System Safety **57**, 41–49 (1997)

[86] Nagy, T., Turányi, T.: Reduction of very large reaction mechanisms using methods based on simulation error minimization. Combustion and Flame **156**, 417–428 (2009)

[87] Huang, H., Fairweather, M., Griffiths, J.F., Tomlin, A.S., Brad, R.B.: A systematic lumping approach for the reduction of comprehensive kinetic models. Proceedings of the Combustion Institute **30**, 1309–1316 (2005)

[88] Apri, M., de Gee, M., Molenaar, J.: Complexity reduction preserving dynamical behavior of biochemical networks. Journal of Theoretical Biology **304**, 16–26 (2012)

[89] Turányi, T.: Reduction of large reaction mechanism. New Journal of Chemistry **14**, 795–803 (1990)

[90] Hannemann, R., Marquardt, W., Gendler, B., Naumann, U.: Discrete first- and second-order adjoints and automatic differentiation for the sensitivity analysis of dynamic models. In: Procedia Computer Science, vol. 1, pp. 297–305 (2010)

[91] Brendel, M., Bonvin, D., Marquardt, W.: Incremental identification of kinetic models for homogeneous reaction systems. Chemical Engineering Science **61**(16), 5404–5420 (2006)

[92] Snow, R.H.: A chemical kinetics computer program for homogeneous and free-radical systems of reactions. The Journal of Physical Chemistry **70**(9), 2780–2786 (1966)

[93] van Riel, N.: Dynamic modelling and analysis of biochemical networks: mechanism-based models and model-based experiments. Briefings in Bioinformatics **7**(4), 364–374 (2006)

[94] Jaqaman, K., Danuser, G.: Linking data to models: data regression. Molecular Cell Biology **7**(11), 813–9 (2006)

[95] Chou, I.-C., Voit, E.O.: Recent developments in parameter estimation and structure identification of biochemical and genomic systems. Mathematical Biosciences **219**(2), 57–83 (2009)

[96] Ashyraliyev, M., Fomekong-Nanfack, Y., Kaandorp, J.A., Blom, J.G.: Systems biology: parameter estimation for biochemical models. The FEBS Journal **276**(4), 886–902 (2009)

[97] Vanlier, J., Tiemann, C.A., Hilbers, P.A.J., van Riel, N.A.W.: Parameter uncertainty in biochemical models described by ordinary differential equations. Mathematical Biosciences **246**(2), 305–314 (2013)

[98] Tarantola, A.: Inverse Problem Theory and Methods for Model Parameter Estimation. SIAM, Philadelphia, PA (2005)

[99] Clermont, G., Zenker, S.: The inverse problem in mathematical biology. Mathematical Biosciences **260**, 11–15 (2015)

[100] Rocke, D.M., Durbin, B.: A model for measurement error for gene expression arrays. Journal of Computational Biology **8**(6), 557–69 (2001)

[101] Schilling, M., Maiwald, T., Bohl, S., Kollmann, M., Kreutz, C., Timmer, J., Klingmu, U.: Quantitative data generation for systems biology : the impact of randomisation , calibrators and normalisers. In: IEE Proc.-Syst. Biol., vol. 152, pp. 193–200 (2005)

[102] Huber, W., Von Heydebreck, A., Sültmann, H., Poustka, A., Vingron, M.: Variance stabilization applied to microarray data calibration and to the quantification of differential expression. Bioinformatics **18**(suppl 1), 96–104 (2002)

[103] Durbin, B.P., Hardin, J.S., Hawkins, D.M., Rocke, D.M.: A variance-stabilizing transformation for gene-expression microarray data. Bioinformatics **18**(suppl 1), 105–110 (2002)

[104] Schittkowski, K.: Numerical Data Fitting in Dynamical Systems: a Practical Introduction with Applications and Software vol. 77, pp. 1–405. Springer, Dordrecht (2002)

[105] Villaverde, A.F., Banga, J.R.: Reverse engineering and identification in systems biology: strategies, perspectives and challenges. Journal of The Royal Society Interface **11**(91), 20130505 (2014)

[106] Kravaris, C., Hahn, J., Chu, Y.: Advances and selected recent developments in state and parameter estimation. Computers & Chemical Engineering **51**, 111–123 (2013)

[107] Ljung, L., Chen, T.: Convexity issues in system identification. In: 10th IEEE International Conference on Control and Automation, pp. 1–9 (2013)

[108] Moles, C.G., Mendes, P., Banga, J.R.: Parameter estimation in biochemical pathways : A comparison of global optimization methods. Genome Research **13**, 2467–2474 (2003)

[109] Chen, W.W., Niepel, M., Sorger, P.K.: Classic and contemporary approaches to modeling biochemical reactions. Genes & Development **24**(17), 1861–1875 (2010)

[110] Guay, M., McLean, D.D.: Optimization and sensitivity analysis for multiresponse parameter estimation in systems of ordinary differential equations. Computers & Chemical Engineering **19**(12), 1271–1285 (1995)

[111] Vassiliadis, V.S., Canto, E.B., Banga, J.R.: Second-order sensitivities of general dynamic systems with application to optimal control problems. Chemical Engineering Science **54**(17), 3851–3860 (1999)

[112] McKay, M.D., Beckman, R.J., Conover, W.J.: A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. Technometrics **42**(1), 55–61 (2000)

[113] Geier, F., Fengos, G., Felizzi, F., Iber, D.: Analyzing and constraining signaling networks: Parameter estimation for the user. In: Liu, X., Betterton, M.D. (eds.) Computational Modeling of Signaling Networks. Methods in Molecular Biology, vol. 880, pp. 23–40. Humana Press, Totowa, NJ (2012)

[114] Raue, A., Schilling, M., Bachmann, J., Matteson, A., Schelke, M., Kaschek, D., Hug, S., Kreutz, C., Harms, B.D., Theis, F.J., Klingmüller, U., Timmer, J.: Lessons learned from quantitative dynamical modeling in systems biology. PLoS ONE **8**(9), 74335 (2013)

[115] Rodriguez-Fernandez, M., Egea, J.A., Banga, J.R.: Novel metaheuristic for parameter estimation in nonlinear dynamic biological systems. BMC Bioinformatics **7**, 483 (2006)

[116] Kim, K.A., Spencer, S.L., Albeck, J.G., Burke, J.M., Sorger, P.K., Gaudet, S., Kim, D.H.: Systematic calibration of a cell signaling network model. BMC Bioinformatics **11**(202) (2010)

[117] Esposito, W.R., Floudas, C.A.: Global optimization for the parameter estimation of differential-algebraic systems. Ind. Eng. Chem. Res **39**, 1291–1310 (2000)

[118] Papamichail, I., Adjiman, C.S.: Global optimization of dynamic systems. Comput. Chem. Eng **28**, 403–415 (2004)

[119] Singer, A.B., Taylor, J.W., Barton, P.I., Green Jr, W.H.: Global dynamic optimization for parameter estimation in chemical kinetics. Journal of Physical Chemistry **110**(3), 971–976 (2006)

[120] Chachuat, B., Singer, A.B., Barton, P.I.: Global methods for dynamic optimization and mixed-integer dynamic optimization. Industrial & Engineering Chemistry Research **45**(25), 8373–8392 (2006)

[121] Miró, A., Pozo, C., Guillén-Gosálbez, G., Egea, J.A., Jiménez, L.: Deterministic global optimization algorithm based on outer approximation for the parameter estimation of nonlinear dynamic biological systems. BMC Bioinformatics **13**(1), 90 (2012)

[122] Rodriguez-Fernandez, M., Mendes, P., Banga, J.R.: A hybrid approach for efficient and robust parameter estimation in biochemical pathways. Bio Systems **83**(2-3), 248–65 (2006)

[123] Sun, J., Garibaldi, J.M., Hodgman, C.: Parameter estimation using metaheuristics in systems biology: a comprehensive review. IEEE/ACM Transactions on Computational Biology and Bioinformatics **9**(1), 185–202 (2012)

[124] Balsa-Canto, E., Peifer, M., Banga, J.R., Timmer, J., Fleck, C.: Hybrid optimization method with general switching strategy for parameter estimation. BMC Systems Biology **2**(1), 26 (2008)

[125] Egea, J.A., Martí, R., Banga, J.R.: An evolutionary method for complex-process optimization. Computers & Operations Research **37**(2), 315–324 (2010)

[126] Jia, G., Stephanopoulos, G., Gunawan, R.: Incremental parameter estimation of kinetic metabolic network models. BMC Systems Biology **6**, 142 (2012)

[127] Fan, M., Kuwahara, H., Wang, X., Wang, S., Gao, X.: Parameter estimation methods for gene circuit modeling from time-series mrna data: a comparative study. Briefings in Bioinformatics, 015 (2015)

[128] Zi, Z., Klipp, E.: Constraint-based modeling and kinetic analysis of the smad dependent tgf-β signaling pathway. PLoS ONE **2**(9), 936 (2007)

[129] Engl, H.W., Hanke, M., Neubauer, A.: Regularization of Inverse Problems, pp. 1–329. Kluwer Academic Publisher, Dordrecht (1996)

[130] Schölkopf, B., Smola, A.J.: Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond. MIT press, Cambridge, MA, USA (2002)

[131] Bauer, F., Lukas, M.A.: Comparingparameter choice methods for regularization of ill-posed problems. Mathematics and Computers in Simulation **81**(9), 1795–1841 (2011)

[132] Sra, S., Nowozin, S., Wright, S.J.: Optimization for Machine Learning. MIT Press, Cambridge, Massachusetts, London, England (2012)

[133] Ljung, L., Chen, T.: What can regularization offer for estimation of dynamical systems ? In: Giri, F., Van Assche, V. (eds.) 11th IFAC International Workshop on Adaptation and Learning in Control and Signal Processing, pp. 1–8. IFAC, Caen, France (2013)

[134] Engl, H.W., Flamm, C., Kügler, P., Lu, J., Müller, S., Schuster, P., Philipp, K.: Inverse problems in systems biology. Inverse Problems **25**(12), 123014 (2009)

[135] Bansal, L., Chu, Y., Laird, C., Hahn, J.: Regularization of inverse problems to determine transcription factor profiles from fluorescent reporter systems. AIChE Journal **58**(12), 3751–3762 (2012)

[136] Wang, H., Wang, X.-C.: Parameter estimation for metabolic networks with two stage bregman regularization homotopy inversion algorithm. Journal of Theoretical Biology **343**, 199–207 (2014)

[137] Walter, E., Prorizato, L.: Identification of Parametric Models from Experimental Data. Springer, Berlin (1997)

[138] Seber, G.A.F., Wild, C.J.J.: Nonlinear Regression. John Wiley & Sons, Inc, Auckland, New Zealand (2003)

[139] Dennis, J.E., Gay, D.M., Welsch, R.E.: An adaptive nonlinear least-squares algorithm. ACM Transaction on Mathematical Software **7**(3), 348–368 (1981)

[140] Serban, R., Hindmarsh, A.C.: Cvodes: the sensitivity-enabled ode solver in sundials. In: ASME 2005 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, pp. 257–269 (2005). American Society of Mechanical Engineers

[141] Hawkins, D.M.: The problem of overfitting. Journal of Chemical Information and Computer Sciences **44**(1), 1–12 (2004)

[142] Picard, R.R., Cook, R.D.: Cross-validation of regression models. Journal of the American Statistical Association **79**(387), 575–583 (1984)

[143] Snee, R.D.: Validation of regression models : Methods and examples. Technometrics **19**(4), 415–428 (1977)

[144] Palm, R.: Numerical comparison of regularization algorithms for solving ill-posed problems. PhD thesis, University of Tartu, Estonia (2010)

[145] Kaltenbacker, B., Neubauer, A., Scherzer, O.: Iterative Regularization Methods for Nonlinear Ill-Posed Problems. Radon Series on Computational and Applied Mathematics. Walter de Gruyter, Berlin, New York (2008)

[146] Rodriguez-Fernandez, M., Kucherenko, S., Pantelides, C., Shah, N.: Optimal experimental design based on global sensitivity analysis. In: 17th European Symposium on Computer Aided Process Engineering, pp. 1–6 (2007)

[147] FitzHugh, R.: Impulses and physiological states in theoretical models of nerve membrane. Biophysical Journal **1**(6), 445–466 (1961)

[148] Nagumo, J., Arimoto, S., Yoshizawa, S.: An active pulse transmission line simulating nerve axon. Proceedings of the IRE **50**(10), 2061–2070 (1962)

[149] Kholodenko, B.N.: Negative feedback and ultrasensitivity can bring about oscillations in the mitogen-activated protein kinase cascades. European Journal of Biochemistry **267**(6), 1583–1588 (2000)

[150] Goodwin, B.C.: Oscillatory behavior in enzymatic control processes. Advances in Enzyme Regulation **3**, 425–438 (1965)

[151] Bray, D., Bourret, R.B., Simont, M.I.: Computer simulation of the phosphorylation cascade controlling bacterial chemotaxis. Molecular Biology of the Cell **4**(May), 469–482 (1993)

[152] Gonze, D., Abou-Jaoudé, W.: The goodwin model: behind the hill function. PLoS ONE **8**(8), 69573 (2013)

[153] Leander, J., Lundh, T., Jirstrand, M.: Stochastic differential equations as a tool to regularize the parameter estimation problem for continuous time dynamical systems given discrete time measurements. Mathematical Biosciences **251**, 54–62 (2014)

[154] Gábor, A., Banga, J.R.: Improved parameter estimation in kinetic models: Tuning of regularization methods. In: Mendes, P., Dada, J., Smallbone, K. (eds.) 13th Conference on Computational Methods in Systems Biology, pp. 45–60. Springer, Manchester (2014)

[155] Faraway, J.J.: Does data splitting improve prediction? Statistics and Computing **26**(1), 49–60 (2016)

[156] Lukas, M.A.: Robust GCV choice of the regularization parameter for correlated data. The Journal of Integral Equations and Applications **22**(3), 519–547 (2010)

[157] Balsa-Canto, E., Alonso, A.A., Banga, J.R.: An iterative identification procedure for dynamic modeling of biochemical networks. BMC Systems Biology **4**, 11 (2010)

[158] Hollander, M., Wolfe, D.A., Chicken, E.: Nonparametric Statistical Methods. John Wiley & Sons, Hoboken, New Jersey (2013)

[159] Gábor, A., Hangos, K., Banga, J.R., Szederkényi, G.: Reaction network realizations of rational biochemical systems and their structural properties. Journal of Mathematical Chemistry **53**(8), 1657–1686 (2015) (Impact Factor: 1.056)

[160] Gábor, A., Hangos, K.M., Szederkenyi, G., Banga, J.R.: On the verification and correction of large-scale kinetic models in system biology. Computational Methods in Systems Biology, Lecture Notes in Computer Science **8130**, 206–219 (2013)

[161] Gábor, A., Hangos, K.M., Szederkényi, G.: Linear conjugacy in biochemical reaction networks with rational reaction rates. Journal of Mathematical Chemistry **54**(8), 1658–1676 (2016) (Impact Factor: 1.056)

[162] Hannemann-Tamás, R., Gábor, A., Szederkényi, G., Hangos, K.M.: Model complexity reduction of chemical reaction networks using mixed-integer quadratic programming. Computers & Mathematics with Applications **65**(10), 1575–1595 (2012) (Impact Factor: 1.398)

[163] Hangos, K.M., Gábor, A., Szederkenyi, G.: Model reduction in bio-chemical reaction networks with michaelis-menten kinetics. In: European Control Conference, pp. 4478–4483 (2013)

[164] Gábor, A., Banga, J.R.: Robust and efficient parameter estimation in dynamic models of biological systems. BMC Systems Biology **9**(1), 74 (2015) (Impact Factor: 2.213)

[165] Balsa-Canto, E., Henriques, D., Gabor, A., Banga, J.R.: Amigo2, a toolbox for dynamic modeling, optimization and control in systems biology. Bioinformatics (2016) (Impact Factor: 5.766)

[166] Rudan, J., Szederkényi, G., Hangos, K.M., Péni, T.: Polynomial time algorithms to determine weakly reversible realizations of chemical reaction networks. Journal of Mathematical Chemistry **52**(5), 1386–1404 (2014)

[167] Chassagnole, C., Noisommit-Rizzi, N., Schmid, J.W., Mauch, K., Reuss, M.: Dynamic modeling of the central carbon metabolism of escherichia coli. Biotechnology and Bioengineering **79**(1), 53–73 (2002)

[168] Vardanyan, I.A., Sachyan, G.A., Nalbandyan, A.B.: The rate constant of the reaction HO2 + CO to CO2 + OH. International Journal of Chemical Kinetics **7**, 23–31 (1975)

[169] Ljung, L.: System Identification: Theory for User. PTR Prentice Hall, New Jersey (1987)

[170] Johansen, T.A.: On tikhonov regularization, bias and variance in nonlinear system identification. Automatica **33**(3), 441–446 (1997)

[171] Sjöberg, J., Ljung, L.: Overtraining, regularization, and searching for minimum in neural networks. In: Dugard, L., Msaad, M., Landau, I.D. (eds.) IFAC Symposia Series, Adaptive Systems in Control and Signal Processing, pp. 73–78. Pergamon Press, Oxford (1992)

[172] Gupta, N.K., Mehra, R.K.: Computational aspects of maximum likelihood estimation and reduction in sensitivity function calculations. IEEE Transactions on Automatic Control **19**(6), 774–783 (1974)

[173] Turányi, T.: Sensitivity analysis of complex kinetic systems. tools and applications. Journal of Mathematical Chemistry **5**(3), 203–248 (1990)

[174] Vajda, S., Rabitz, H.: Identifiability and distinguishability of general reaction systems. The Journal of Physical Chemistry **98**(20), 5265–5271 (1994)

[175] Li, R., Henson, M.A., Kurtz, M.J.: Selection of model parameters for off-line parameter estimation. IEEE Transactions on Control Systems Technology **12**(3), 402–412 (2004)

[176] Gutenkunst, R.N., Waterfall, J.J., Casey, F.P., Brown, K.S., Myers, C.R., Sethna, J.P.: Universally sloppy parameter sensitivities in systems biology models. PLoS Computational Biology **3**(10), 1871–78 (2007)

[177] Gay, D.M.: Usage summary for selected optimization routines. AT&T Bell Laboratories Murray Hill, NJ 07974, Computing Science Technical Report No. 153 (1990)

[178] Hansen, P.C.: Analysis of discrete ill-posed problems by means of the l-curve. SIAM Review **34**(4), 561–580 (1992)

[179] Hansen, P.C., O'Leary, D.P.: The use of the l-curve in the regularization of discrete ill-posed problems. SIAM Journal on Scientific Computing **14**(6), 1487–1503 (1993)

[180] Golub, G.H., Heath, M., Wahab, G.: Generalized cross-validation as a method for choosing a good ridge parameter. Technometrics **21**(2), 215–223 (1979)

[181] Lukas, M.A.: Strong robust generalized cross-validation ot choosing the regualrization parameter. Inverse Problems **24**(3), 34006–34021 (2008)

[182] Morozov, V.A.: Methods for Solving Incorrectly Posed Problems. Springer, New York (1984)

[183] Gfrerer, H.: An a posteriori parameter choice for ordinary and iterated tikhonov regularization of ill-posed problems leading to optimal convergence rates. Mathematics of Computation **49**(180), 507 (1987)

[184] Hämarik, U., Raus, T.: On the choice of the regularization parameter in ill-posed problems with approximately given noise level of data. Journal of Inverse and Ill-posed Problems **14**(3), 251–266 (2006)

[185] Hämarik, U., Tautenhahn, U.: On the monotone error rule for parameter choice in iterative and continuous regularization methods. BIT **41**(5), 1029–1038 (2001)

[186] Lepskii, O.: On a problem of adaptive estimation in gaussian white noise. Theory of Probability & Its Applications **35**(3), 454–466 (1991)

[187] Bauer, F.: Some considerations concerning regularization and parameter choice algorithms. Inverse Problems **23**(2), 837–858 (2007)

[188] Regińska, T.: A regularization parameter in discrete ill-posed problems. SIAM Journal on Scientific Computing **17**(3), 740–749 (1996)

[189] Brezinski, C., Rodriguez, G., Seatzu, S.: Error estimates for linear systems with applications to regularization. Numerical Algorithms **49**, 85–104 (2008)

[190] Bauer, F., Mathe, P.: Parameter choice methods using minimization schemes. Journal of Complexity **27**, 68–85 (2011)

[191] Wahba, G.: Practical approximate solutions to linear operator equations when the data are noisy. SIAM Journal on Numerical Analysis **14**(4), 651–667 (1977)

Appendix A

Biochemical reaction networks

A.1 Further Example for bio-CRNs

Here we present a structurally similar example to what was presented in Example 2-1, but with mass action kinetics.

Example A-1

Let us restrict each kinetics of the network in Example 2-1 to mass action type. Note that in the previous example there were two reactions (r_{121} and r_{122}) from complex C_1 to complex C_2 with different kinetics. In the classical MAL-CRNs parallel reactions are not allowed, thus they are represented by one edge as depicted in Figure A.1.

The three complexes

$$C_1 = X_1 + 2X_2, \quad C_2 = X_1 + X_3, \quad C_3 = X_4$$

are now connected by five irreversible reaction steps, i.e.

$$\mathcal{R} = \{(C_1, C_2),\ (C_2, C_1),\ (C_1, C_3),\ (C_3, C_1),\ (C_2, C_3)\}.$$

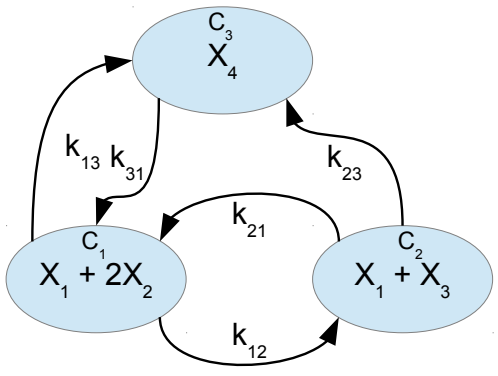


Figure A.1: Reaction graph of a simple MAL-CRN

The matrices and the non-linear vector function which characterizes the network are as follows

$$Y = \begin{bmatrix} 1 & 1 & 0 \\ 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad , \quad A_{\mathrm{k}} = \begin{bmatrix} -(k_{12}+k_{13}) & k_{21} & k_{31} \\ k_{12} & -(k_{21}+k_{23}) & 0 \\ k_{13} & k_{23} & -k_{31} \end{bmatrix} \quad , \quad \Psi(x) = \begin{bmatrix} x_1 x_2^2 \\ x_1 x_3 \\ x_4 \end{bmatrix} .$$

For mass action systems the kinetic weighting matrix is the identity matrix $P = I_{3\times 3}$. The above elements – based on (2.9) – define the following differential equation model

$$\begin{aligned} \dot{x}_1 &= -k_{13}x_1x_2^2 - k_{23}x_1x_3 \\ \dot{x}_2 &= -2k_{13}x_1x_2^2 - 2k_{12}x_1x_2^2 + 2k_{21}x_1x_3 + 2k_{31}x_4 \\ \dot{x}_3 &= -k_{23}x_1x_3 - k_{21}x_1x_3 + k_{12}x_1x_2^2 \\ \dot{x}_4 &= -k_{31}x_4 + k_{13}x_1x_2^2 + k_{23}x_1x_3 \ . \end{aligned}$$

■

A.2 Non-negativity of bio-CRNs

As all of the kinetic functions in a biochemical reaction network defined by (2.9) are locally Lipschitz and essentially non-negative, therefore the concentrations remain non-negative. The proof is as follows.

In order to prove the non-negativity of the solution, one need to check the Lipschitz condition and the essential non-negativity of the right hand side of (2.9).

It is easy to see, that the right hand sides of the ODEs are continuously differentiable, therefore they are locally Lipschitz.

To show the essential non-negativity of the right hand side functions, insert (2.15) into (2.9), for $p = 1, \dots n$, then the p-th equation reads as

$$f_p(x) = \sum_{l=1}^{m} Y_{pl} \cdot \sum_{j=1}^{\kappa} A_{\mathrm{k},lj} \cdot \varphi_j(x) \tag{A.1}$$

Rewriting the sum over all the κ kinetics into two sums: over the reactant complexes and over the kinetics in each of these complexes, one arrives to

$$f_p(x) = \sum_{l=1}^{m} Y_{pl} \sum_{j=1}^{m} \sum_{i=1}^{d_i} A_{\mathrm{k},l\,z_j+i} \cdot \varphi_{z_j+i}(x), \tag{A.2}$$

where $z_j = \sum_{k=1}^{j-1} d_k$. Using (2.15) and (2.5)

$$f_p(x) = \sum_{l=1}^{m} Y_{pl} \sum_{j=1}^{m} \sum_{i=1}^{d_i} A_{\mathrm{k},l\,z_j+i} \cdot g_{ji}(x) = \sum_{l=1}^{m} Y_{pl} \sum_{j=1}^{m} \sum_{i=1}^{d_i} A_{\mathrm{k},l\,z_j+i} \cdot \frac{\prod_{o=1}^{n} x_o^{Y_{oj}}}{D_{ji}(x)} \ . \tag{A.3}$$

From the definition of A_k we know that the coefficients $A_{\mathrm{k},l\,z_j+i}$ are negative when $l = j$ and non-negative otherwise. So we decompose the summation over j into the two cases

$$f_p(x) = \sum_{l=1}^{m} Y_{pl} \sum_{j=1, j\neq l}^{m} \sum_{i=1}^{d_i} A_{\mathrm{k},l\,z_j+i} \cdot \frac{\prod_{o=1}^{n} x_o^{Y_o,j}}{D_{ji}(x)} - \sum_{l=1}^{m} Y_{pl} \sum_{i=1}^{d_i} |A_{\mathrm{k},l\,z_j+i}| \cdot \frac{\prod_{o=1}^{n} x_o^{Y_{ol}}}{D_{li}(x)} \ . \tag{A.4}$$

Notice that the first term is always non-negative and the second term contains the factor $Y_{pl}x_p^{Y_{pl}}$. If $Y_{pl} = 0$, then $\lim_{x_p \to 0} \frac{0 \cdot x_p^0}{D(x)} = 0$. If $Y_{pl} > 0$, since the denominator term (2.7) cannot approach zero, $\lim_{x_p \to 0} \frac{Y_{pl}x_p^{Y_{pl}}}{D(x)} = 0$ and f_p is indeed essentially non-negative.

Appendix B

Model verification

B.1 Case study: Central carbon metabolism of *E. coli*

The model, published by Chassagnole and co-workers [167] describes the central carbon metabolism of the *Escherichia coli* bacteria. Although we could reproduce the simulated trajectories shown in the paper [167] with the published model (identification number of the model in the BioModels Database: BIOMD051), numerical simulations during parameter estimation tasks reported errors for certain regions of the parameter space, because negative values of the states appeared.

About the model

The metabolism is described by 48 reactions which are grouped into kinetic types: reversible and irreversible Michealis-Menten kinetics, two-substrate reversible and irreversible Michaelis-Menten kinetics, allosteric enzyme reactions, allosteric regulation, allosteric activation, ordered uni-bi mechanism, Hill kinetics, constant level reaction and reversible mass action kinetics. The mass balance equations for the 18 metabolites are in the following form

$$\frac{dx_i}{dt} = \sum_{j=1}^{48} s_{ij} r_j(x,k) - \mu x_i \quad \text{for} \quad i = 1, \dots 18 \ , \tag{B.1}$$

where x is the concentration vector of the metabolites, s_{ij} is the (i,j)-th element of the stoichiometric matrix, $r_j(x,k)$ denotes the j-th reaction rate function, which depends on the concentrations and the k rate function parameters. Finally, μ is the growth factor. The detailed equations are listed in form of tables in Figures B.1–B.4 taken from the original publication [167]. Figure B.1 shows the dynamic model equation, the reaction rate functions that appears in the right hand side of the equations are reported in Figures B.2–B.4.

Checking the rate expressions

The first criteria of a plausible model is the non-negativity of the reaction rate functions, for which the reversible reactions have to be cut into a forward and a backward reaction. The separation of the reactions are straightforward in this case study. It is also easy to see

$$\frac{dC_{\text{glc}}^{\text{extracellular}}}{dt} = D\left(C_{\text{glc}}^{\text{feed}} - C_{\text{glc}}^{\text{extracellular}}\right) + f_{\text{pulse}} - \frac{C_x r_{\text{PTS}}}{\rho_x}$$

$$\frac{dC_{\text{g6p}}}{dt} = r_{\text{PTS}} - r_{\text{PGI}} - r_{\text{G6PDH}} - r_{\text{PGM}} - \mu C_{\text{g6p}}$$

$$\frac{dC_{\text{f6p}}}{dt} = r_{\text{PGI}} - r_{\text{PFK}} + r_{\text{TKb}} + r_{\text{TA}} - 2r_{\text{MurSynth}} - \mu C_{\text{f6p}}$$

$$\frac{dC_{\text{fdp}}}{dt} = r_{\text{PFK}} - r_{\text{ALDO}} - \mu C_{\text{fdp}}$$

$$\frac{dC_{\text{gap}}}{dt} = r_{\text{ALDO}} + r_{\text{TIS}} - r_{\text{GAPDH}} + r_{\text{TKa}} + r_{\text{TKb}} - r_{\text{TA}} + r_{\text{TrpSynth}} - \mu C_{\text{gap}}$$

$$\frac{dC_{\text{dhap}}}{dt} = r_{\text{ALDO}} - r_{\text{TIS}} - r_{\text{G3PDH}} - \mu C_{\text{dhap}}$$

$$\frac{dC_{\text{pgp}}}{dt} = r_{\text{GAPDH}} - r_{\text{PGK}} - \mu C_{\text{pgp}}$$

$$\frac{dC_{\text{3pg}}}{dt} = r_{\text{PGK}} - r_{\text{PGluMu}} - r_{\text{SerSynth}} - \mu C_{\text{3pg}}$$

$$\frac{dC_{\text{2pg}}}{dt} = r_{\text{PGluMu}} - r_{\text{ENO}} - \mu C_{\text{2pg}}$$

$$\frac{dC_{\text{pep}}}{dt} = r_{\text{ENO}} - r_{\text{PK}} - r_{\text{PTS}} - r_{\text{PEPCxylase}} - r_{\text{DAHPS}} - r_{\text{Synth1}}^{\text{a}} - \mu C_{\text{pep}}$$

$$\frac{dC_{\text{pyr}}}{dt} = r_{\text{PK}} + r_{\text{PTS}} - r_{\text{PDH}} - r_{\text{Synth2}}^{\text{b}} + r_{\text{MetSynth}} + r_{\text{TrpSynth}} - \mu C_{\text{pyr}}$$

$$\frac{dC_{\text{6pg}}}{dt} = r_{\text{G6PDH}} - r_{\text{PGDH}} - \mu C_{\text{6pg}}$$

$$\frac{dC_{\text{ribu5p}}}{dt} = r_{\text{PGDH}} - r_{\text{Ru5P}} - r_{\text{R5PI}} - \mu C_{\text{ribu5p}}$$

$$\frac{dC_{\text{xyl5p}}}{dt} = r_{\text{Ru5P}} - r_{\text{TKa}} - r_{\text{TKb}} - \mu C_{\text{xyl5p}}$$

$$\frac{dC_{\text{sed7p}}}{dt} = r_{\text{TKa}} - r_{\text{TA}} - \mu C_{\text{sed7p}}$$

$$\frac{dC_{\text{rib5p}}}{dt} = r_{\text{R5PI}} - r_{\text{TKa}} - r_{\text{RPPK}} - \mu C_{\text{rib5p}}$$

$$\frac{dC_{\text{e4p}}}{dt} = r_{\text{TA}} - r_{\text{TKb}} - r_{\text{DAHPS}} - \mu C_{\text{e4p}}$$

$$\frac{dC_{\text{glp}}}{dt} = r_{\text{PGM}} - r_{\text{GlPAT}} - \mu C_{\text{glp}}$$

[a] $r_{\text{Synth1}} = r_{\text{ChoSynth}} + r_{\text{MurSynth}}$.
[b] $r_{\text{Synth2}} = r_{\text{IleSynth}} + r_{\text{AlaSynth}} + r_{\text{KivalSynth}} + r_{\text{DipimSynth}}$.

Figure B.1: Model equations of the central carbon metabolism model of *E.Coli* [167]

Phosphotransferase system:

$$r_{\mathrm{PTS}} = \frac{r_{\mathrm{PTS}}^{\max} C_{\mathrm{glc}}^{\mathrm{extracellular}} \frac{C_{\mathrm{pep}}}{C_{\mathrm{pyr}}}}{\left(K_{\mathrm{PTS,a1}} + K_{\mathrm{PTS,a2}} \frac{C_{\mathrm{pep}}}{C_{\mathrm{pyr}}} + K_{\mathrm{PTS,a3}} C_{\mathrm{glc}}^{\mathrm{extracellular}} + C_{\mathrm{glc}}^{\mathrm{extracellular}} \frac{C_{\mathrm{pep}}}{C_{\mathrm{pyr}}}\right)\left(1 + \frac{C_{\mathrm{g6p}}^{n_{PTS,g6p}}}{K_{\mathrm{PTS,g6p}}}\right)}$$

Phosphoglucoisomerase:

$$r_{\mathrm{PGI}} = \frac{r_{\mathrm{PGI}}^{\max}\left(C_{\mathrm{g6p}} - \frac{C_{\mathrm{f6p}}}{K_{\mathrm{PGI,eq}}}\right)}{K_{\mathrm{PGI,g6p}}\left(1 + \frac{C_{\mathrm{f6p}}}{K_{\mathrm{PGI,f6p}}\left(1 + \frac{C_{\mathrm{6pg}}}{K_{\mathrm{PGI,f6p,6pginh}}}\right)} + \frac{C_{\mathrm{6pg}}}{K_{\mathrm{PGI,g6p,6pginh}}}\right) + C_{\mathrm{g6p}}}$$

Phosphofructokinase:

$$r_{\mathrm{PFK}} = \frac{r_{\mathrm{PFK}}^{\max} C_{\mathrm{atp}} C_{\mathrm{f6p}}}{\left(C_{\mathrm{atp}} + K_{\mathrm{PFK,atp,s}}\left(1 + \frac{C_{\mathrm{adp}}}{K_{\mathrm{PFK,adp,c}}}\right)\right)\left(C_{\mathrm{f6p}} + K_{\mathrm{PFK,f6p,s}} \frac{A}{B}\right)\left(1 + \frac{L_{\mathrm{PFK}}}{\left(1 + C_{\mathrm{f6p}} \frac{B}{K_{\mathrm{PFK,f6p,s}} A}\right)^{n_{\mathrm{PFK}}}}\right)}$$

$$A = 1 + \frac{C_{\mathrm{pep}}}{K_{\mathrm{PFK,pep}}} + \frac{C_{\mathrm{adp}}}{K_{\mathrm{PFK,adp,b}}} + \frac{C_{\mathrm{amp}}}{K_{\mathrm{PFK,amp,b}}}$$

$$B = 1 + \frac{C_{\mathrm{adp}}}{K_{\mathrm{PFK,adp,a}}} + \frac{C_{\mathrm{amp}}}{K_{\mathrm{PFK,amp,a}}}$$

Aldolase:

$$r_{\mathrm{ALDO}} = \frac{r_{\mathrm{ALDO}}^{\max}\left(C_{\mathrm{fdp}} - \frac{C_{\mathrm{gap}} C_{\mathrm{dhap}}}{K_{\mathrm{ALDO,eq}}}\right)}{K_{\mathrm{ALDO,fdp}} + C_{\mathrm{fdp}} + \frac{K_{\mathrm{ALDO,gap}} C_{\mathrm{dhap}}}{K_{\mathrm{ALDO,eq}} V_{\mathrm{ALDO,blf}}} + \frac{K_{\mathrm{ALDO,dhap}} C_{\mathrm{gap}}}{K_{\mathrm{ALDO,eq}} V_{\mathrm{ALDO,blf}}} + \frac{C_{\mathrm{fdp}} C_{\mathrm{gap}}}{K_{\mathrm{ALDO,gap,inh}}} + \frac{C_{\mathrm{dhap}} C_{\mathrm{gap}}}{K_{\mathrm{ALDO,eq,}} V_{\mathrm{ALDO,blf}}}}$$

Triosephosphate isomerase:

$$r_{\mathrm{TIS}} = \frac{r_{\mathrm{TIS}}^{\max}\left(C_{\mathrm{dhap}} - \frac{C_{\mathrm{gap}}}{K_{\mathrm{TIS,eq}}}\right)}{K_{\mathrm{TIS,dhap}}\left(1 + \frac{C_{\mathrm{gap}}}{K_{\mathrm{TIS,gap}}}\right) + C_{\mathrm{dhap}}}$$

Glyceraldehyde 3-phosphate dehydrogenase:

$$r_{\mathrm{GAPDH}} = \frac{r_{\mathrm{GADPH}}^{\max}\left(C_{\mathrm{gap}} C_{\mathrm{nad}} - \frac{C_{\mathrm{pgp}} C_{\mathrm{nadh}}}{K_{\mathrm{GAPDH,eq}}}\right)}{\left(K_{\mathrm{GAPDH,gap}}\left(1 + \frac{C_{\mathrm{pgp}}}{K_{\mathrm{GAPDH,pgp}}}\right) + C_{\mathrm{gap}}\right)\left(K_{\mathrm{GAPDH,nad}}\left(1 + \frac{C_{\mathrm{nadh}}}{K_{\mathrm{GAPDH,nadh}}}\right) + C_{\mathrm{nad}}\right)}$$

Phosphoglycerate kinase:

$$r_{\mathrm{PGK}} = \frac{r_{\mathrm{PGK}}^{\max}\left(C_{\mathrm{adp}} C_{\mathrm{pgp}} - \frac{C_{\mathrm{atp}} C_{\mathrm{3pg}}}{K_{\mathrm{PGK,eq}}}\right)}{\left(K_{\mathrm{PGK,adp}}\left(1 + \frac{C_{\mathrm{atp}}}{K_{\mathrm{PGK,atp}}}\right) + C_{\mathrm{adp}}\right)\left(K_{\mathrm{PGK,pgp}}\left(1 + \frac{C_{\mathrm{3pg}}}{K_{\mathrm{PGK,3pg}}}\right) + C_{\mathrm{pgp}}\right)}$$

Phosphoglycerate mutase:

$$r_{\mathrm{PGluMu}} = \frac{r_{\mathrm{PGluMu}}^{\max}\left(C_{\mathrm{3pg}} - \frac{C_{\mathrm{2pg}}}{K_{\mathrm{PGluMu,eq}}}\right)}{K_{\mathrm{PGluMu,3pg}}\left(1 + \frac{C_{\mathrm{2pg}}}{K_{\mathrm{PGluMu,2pg}}}\right) + C_{\mathrm{3pg}}}$$

Enolase:

$$r_{\mathrm{ENO}} = \frac{r_{\mathrm{ENO}}^{\max}\left(C_{\mathrm{2pg}} - \frac{C_{\mathrm{pep}}}{K_{\mathrm{ENO,eq}}}\right)}{K_{\mathrm{ENO,2pg}}\left(1 + \frac{C_{\mathrm{pep}}}{K_{\mathrm{ENO,pep}}}\right) + C_{\mathrm{2pg}}}$$

Pyruvate kinase:

$$r_{\mathrm{PK}} = \frac{r_{\mathrm{PK}}^{\max} C_{\mathrm{pep}}\left(\frac{C_{\mathrm{pep}}}{K_{\mathrm{PK,pep}}} + 1\right)^{(n_{\mathrm{PK}}-1)} C_{\mathrm{adp}}}{K_{\mathrm{PK,pep}}\left(L_{\mathrm{PK}}\left(\frac{1 + \frac{C_{\mathrm{atp}}}{K_{\mathrm{PK,atp}}}}{\frac{C_{\mathrm{fdp}}}{K_{\mathrm{PK,fdp}}} + \frac{C_{\mathrm{amp}}}{K_{\mathrm{PK,amp}}} + 1}\right)^{n_{\mathrm{PK}}} + \left(\frac{C_{\mathrm{pep}}}{K_{\mathrm{PK,pep}}} + 1\right)^{n_{\mathrm{PK}}}\right)(C_{\mathrm{adp}} + K_{\mathrm{PK,adp}})}$$

Pyruvate dehydrogenase:

$$r_{\mathrm{PDH}} = \frac{r_{\mathrm{PDH}}^{\max} C_{\mathrm{pyr}}^{n_{\mathrm{PDH}}}}{K_{\mathrm{PDH,pyr}} + C_{\mathrm{pyr}}^{n_{\mathrm{PDH}}}}$$

Figure B.2: Reaction rates of the central carbon metabolism model of *E.Coli* [167]

PEP carboxylase:

$$r_{\mathrm{PEPCxylase}} = \frac{r^{\max}_{\mathrm{PEPCxylase}} C_{\mathrm{pep}} \left(1 + \left(\frac{C_{\mathrm{fdp}}}{K_{\mathrm{PEPCxylase,fdp}}}\right)^{\mathrm{n_{PEPCxylase,fdp}}}\right)}{K_{\mathrm{PEPCxylase,pep}} + C_{\mathrm{pep}}}$$

Phosphoglucomutase:

$$r_{\mathrm{PGM}} = \frac{r^{\max}_{\mathrm{PGM}} \left(C_{\mathrm{g6p}} - \frac{C_{\mathrm{g1p}}}{K_{\mathrm{PGM,eq}}}\right)}{K_{\mathrm{PGM,g6p}} \left(1 + \frac{C_{\mathrm{g1p}}}{K_{\mathrm{PGM,g1p}}}\right) + C_{\mathrm{g6p}}}$$

Glucose 1-phosphate adenyltransferase:

$$r_{\mathrm{GIPAT}} = \frac{r^{\max}_{\mathrm{GIPAT}} C_{\mathrm{g1p}} C_{\mathrm{atp}} \left(1 + \left(\frac{C_{\mathrm{fdp}}}{K_{\mathrm{G1PAT,fdp}}}\right)^{\mathrm{n_{GIPAT,fdp}}}\right)}{(K_{\mathrm{G1PAT,glp}} + C_{\mathrm{g1p}})(K_{\mathrm{G1PAT,atp}} + C_{\mathrm{atp}})}$$

Ribose phosphate pyrophosphokinase:

$$r_{\mathrm{RPPK}} = \frac{r^{\max}_{\mathrm{RPPK}} C_{\mathrm{rib5p}}}{K_{\mathrm{RPPK,rib5p}} + C_{\mathrm{rib5p}}}$$

Glycerol 3-phosphate-dehydrogenase:

$$r_{\mathrm{G3PDH}} = \frac{r^{\max}_{\mathrm{G3PDH}} C_{\mathrm{dhap}}}{K_{\mathrm{G3PDH,dhap}} + C_{\mathrm{dhap}}}$$

Serine synthesis:

$$r_{\mathrm{SerSynth}} = \frac{r^{\max}_{\mathrm{SerSynth}} C_{\mathrm{3pg}}}{K_{\mathrm{SerSynth,3pg}} + C_{\mathrm{3pg}}}$$

Mureine synthesis:

$$r_{\mathrm{MurSynth}} = r^{\max}_{\mathrm{MurSynth}}$$

DAHP synthase:

$$r_{\mathrm{DAHPS}} = \frac{r^{\max}_{\mathrm{DAHPS}} C_{\mathrm{e4p}}^{\mathrm{n_{DAHPS,e4p}}} C_{\mathrm{pep}}^{\mathrm{n_{DAHPS,pep}}}}{(K_{\mathrm{DAHPS,e4p}} + C_{\mathrm{e4p}}^{\mathrm{n_{DAHPS,e4p}}})(K_{\mathrm{DAHPS,pep}} + C_{\mathrm{pep}}^{\mathrm{n_{DAHPS,pep}}})}$$

Tryptophan synthesis:

$$r_{\mathrm{TrpSynth}} = r^{\max}_{\mathrm{TrpSynth}}$$

Methionine synthesis:

$$r_{\mathrm{MetSynth}} = r^{\max}_{\mathrm{MetSynth}}$$

Glucose-6-phosphate dehydrogenase:

$$r_{\mathrm{G6PDH}} = \frac{r^{\max}_{\mathrm{G6PDH}} C_{\mathrm{g6p}} C_{\mathrm{nadp}}}{(C_{\mathrm{g6p}} + K_{\mathrm{G6PDH,g6p}}) \left(1 + \frac{C_{\mathrm{nadph}}}{K_{\mathrm{G6PDH,nadph,g6pinh}}}\right) \left(K_{\mathrm{G6PDH,nadp}} \left(1 + \frac{C_{\mathrm{nadph}}}{K_{\mathrm{G6PDH,nadph,nadpinh}}}\right) + C_{\mathrm{nadp}}\right)}$$

6-phosphogluconate dehydrogenase:

$$r_{\mathrm{PGDH}} = \frac{r^{\max}_{\mathrm{PGDH}} C_{\mathrm{6pg}} C_{\mathrm{nadp}}}{(C_{\mathrm{6pg}} + K_{\mathrm{PGDH,6pg}}) \left(C_{\mathrm{nadp}} + K_{\mathrm{PGDH,nadp}} \left(1 + \frac{C_{\mathrm{nadph}}}{K_{\mathrm{PGDH,nadph,inh}}}\right) \left(1 + \frac{C_{\mathrm{ATP}}}{K_{\mathrm{PGDH,atp,inh}}}\right)\right)}$$

Ribulose phosphate epimerase:

$$r_{\mathrm{Ru5p}} = r^{\max}_{\mathrm{Ru5p}} \left(C_{\mathrm{ribu5p}} - \frac{C_{\mathrm{xyl5p}}}{K_{\mathrm{Ru5P,eq}}}\right)$$

Ribose phosphate isomerase:

$$r_{\mathrm{R5P1}} = r^{\max}_{\mathrm{R5P1}} \left(C_{\mathrm{ribu5p}} - \frac{C_{\mathrm{rib5p}}}{K_{\mathrm{R5P1,eq}}}\right)$$

Transketolase a:

$$r_{\mathrm{TKa}} = r^{\max}_{\mathrm{TKa}} \left(C_{\mathrm{rib5p}} C_{\mathrm{xyl5p}} - \frac{C_{\mathrm{sed7p}} C_{\mathrm{gap}}}{K_{\mathrm{TKa,eq}}}\right)$$

Transketolase b:

$$r_{\mathrm{TKb}} = r^{\max}_{\mathrm{TKb}} \left(C_{\mathrm{xyl5p}} C_{\mathrm{e4p}} - \frac{C_{\mathrm{f6p}} C_{\mathrm{gap}}}{K_{\mathrm{TKb,eq}}}\right)$$

Figure B.3: Reaction rates of the central carbon metabolism model of *E.Coli* [167]

Transaldolase:

$$r_{\mathrm{TA}} = r_{\mathrm{TA}}^{\max}\left(C_{\mathrm{gap}}C_{\mathrm{sed7p}} - \frac{C_{\mathrm{e4p}}C_{\mathrm{f6p}}}{K_{\mathrm{TA,eq}}}\right)$$

Synthesis 1:

$$r_{\mathrm{Synth1}} = \frac{r_{\mathrm{Synth1}}^{\max}C_{\mathrm{pep}}}{K_{\mathrm{Synth1,pep}} + C_{\mathrm{pep}}}$$

Synthesis 2:

$$r_{\mathrm{Synth2}} = \frac{r_{\mathrm{Synth2}}^{\max}C_{\mathrm{pyr}}}{K_{\mathrm{Synth2,pyr}} + C_{\mathrm{pyr}}}$$

Figure B.4: Reaction rates of the central carbon metabolism model of *E.Coli* [167]

that the reactions are always non-negative. The model have three constant reactions: the Mureine synthesis, the Tryptophan synthesis and the Methionine synthesis:

$$r_{\mathrm{MurSynth}} = r_{\mathrm{MurSynth}}^{\max}, \tag{B.2}$$

$$r_{\mathrm{TrpSynth}} = r_{\mathrm{TrpSynth}}^{\max} \tag{B.3}$$

$$r_{\mathrm{MetSynth}} = r_{\mathrm{MetSynth}}^{\max} \ , \tag{B.4}$$

but only r_{MurSynth} is not kinetic to its source specie, the others are input terms.

Checking the model structure

The essential non-negativity condition for the model equations (B.1) give rise to the model specific non-negativity condition

$$\frac{dx_i}{dt} = \sum_{j=1}^{48} s_{ij}r_j(x,k) - \mu x_i \geq 0 \quad \text{whenever} \quad x_i = 0, \quad \text{for all} \quad i = 1,\dots 18.$$

This condition holds for plausible reaction rate functions which have the source kinetic property according to (2.18). Furthermore, whenever a reaction is not kinetic with respect to its source species and has negative stoichiometric coefficient, depending on the numerical values of the parameters it can violate the condition and cause negative concentrations during simulations.

This is exactly what happens in some parameter domain of this *E.coli* model. From the following model equation ([167] Table I. Eq.(3)):

$$\frac{dx_{\mathrm{f6p}}}{dt} = r_{\mathrm{PGI}} - r_{\mathrm{PFK}} + r_{\mathrm{TKb}} + r_{\mathrm{TKa}} - 2r_{\mathrm{MurSynth}} - \mu C_{\mathrm{f6p}} \tag{B.5}$$

one can see that the stoichiometric coefficient of the Mureine synthesis r_{MurSynth} is negative, but it is not kinetic with respect to any metabolite. This may result in the appearance of negative concentrations and thus in a non-plausible model.

Correction of the non-plausible reaction

There are several ways to correct the non-plausible reaction. A switching function can be included, which turns off the reaction, whenever the concentration of `fp6` reach zero. This procedure does not influence the model dynamics in the plausible concentration domain, but the switching function may result in mathematical or numerical simulation issues. Alternatively, one can make the reaction source kinetic by multiplying it with x_{fp6}: $r^{\mathrm{cured}}_{\mathrm{MurSynth}} = r^{\mathrm{max}}_{\mathrm{MurSynth}} x_{\mathrm{f6p}}$. It changes the dynamics of the system, but results in a smooth, plausible reaction rate function.

Mass conservation

The truncated model is created by omitting the reactions which are either stand for inflows or outflows. We have found three linearly independent non-negative vectors for which $m_i^T S = 0$, for $i = 1, 2, 3$. This implies three moiety conservation laws, but there is no strictly positive m vector in the left kernel of S, and thus the model does not obey to the total mass conservation.

Appendix C

Model reduction

C.1 Hydrogen-Bromine reaction network

As introductory example we consider the Hydrogen-Bromine chemical reaction network according to Snow [92]. Its reactions take place in the gas phase at a temperature of about 1000 K and a pressure of about 1 bar. The chemical reactions are listed in Eqs. (HBr1)–(HBr6).

$$\mathrm{Br_2 + M \xrightarrow{k_1} 2\,Br\cdot + M} \tag{HBr1}$$

$$\mathrm{2\,Br\cdot + M \xrightarrow{k_2} Br_2 + M} \tag{HBr2}$$

$$\mathrm{Br\cdot + H_2 \xrightarrow{k_3} H\cdot + HBr} \tag{HBr3}$$

$$\mathrm{H\cdot + HBr \xrightarrow{k_4} Br\cdot + H_2} \tag{HBr4}$$

$$\mathrm{H\cdot + Br_2 \xrightarrow{k_5} Br\cdot + HBr} \tag{HBr5}$$

$$\mathrm{Br\cdot + HBr \xrightarrow{k_6} H\cdot + Br_2} \tag{HBr6}$$

In the Eqs. (HBr1)–(HBr6), the involved chemical species comprise Br_2 (Bromine molecule), Br· (Bromine radical), H_2 (Hydrogen molecule), H· (Hydrogen radical), HBr (Hydrogen bromide) and a so-called *third body*, also referred to as *inert component*, arbitrarily denoted by the symbol M. The third body M is a kind of catalyst which does not react with the other chemical species. Its only relevance is to adsorb or transfer kinetic energy from the reactant species, e.g. to split a Bromine molecule into its corresponding radicals (Eq. (HBr1). In the paper of Snow [92], Nitrogen (N_2) was the third body, but any other inert gas would do it as well.

We show all reaction rates of the chemical reaction network (HBr1)–(HBr6) in Table C.1.

Applying the above notations and rules, the stoichiometric matrix for the reaction

Table C.1: Reaction rates and rate coefficients of Hydrogen-Bromine reaction

Reaction	Reaction rate	Rate coefficient
(HBr1)	$r_1 = k_1\,[\mathrm{Br_2}]\,[\mathrm{M}]$	$k_1 = 6.26 \cdot 10^5 \frac{\mathrm{cm}^3}{\mathrm{mol\,s}}$
(HBr2)	$r_2 = k_2\,[\mathrm{Br}\cdot]^2\,[\mathrm{M}]$	$k_2 = 1.56 \cdot 10^{15} \frac{\mathrm{cm}^6}{\mathrm{mol}^2\,\mathrm{s}}$
(HBr3)	$r_3 = k_3\,[\mathrm{Br}\cdot]\,[\mathrm{H_2}]$	$k_3 = 2.61 \cdot 10^9 \frac{\mathrm{cm}^3}{\mathrm{mol\,s}}$
(HBr4)	$r_4 = k_4\,[\mathrm{H}\cdot]\,[\mathrm{HBr}]$	$k_4 = 1.39 \cdot 10^{13} \frac{\mathrm{cm}^3}{\mathrm{mol\,s}}$
(HBr5)	$r_5 = k_5\,[\mathrm{H}\cdot]\,[\mathrm{Br_2}]$	$k_5 = 1.17 \cdot 10^{14} \frac{\mathrm{cm}^3}{\mathrm{mol\,s}}$
(HBr6)	$r_6 = k_6\,[\mathrm{Br}\cdot]\,[\mathrm{HBr}]$	$k_6 = 1.31 \cdot 10^4 \frac{\mathrm{cm}^3}{\mathrm{mol\,s}}$

system (HBr1)–(HBr6) can be written as

$$N = \begin{pmatrix} -1 & 1 & 0 & 0 & -1 & 1 \\ 2 & -2 & -1 & 1 & 1 & -1 \\ 0 & 0 & -1 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 & -1 & 1 \\ 0 & 0 & 1 & -1 & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}. \tag{C.1}$$

Using N and the reaction rates in Table 4.2, we can easily write the ordinary differential equations of the Hydrogen-Bromine CRN shown in Eqs. (C.2)–(C.7).

$$\frac{d[\mathrm{Br_2}]}{dt} = -r_1 + r_2 - r_5 + r_6 \tag{C.2}$$

$$\frac{d[\mathrm{Br}\cdot]}{dt} = 2\,r_1 - 2\,r_2 - r_3 + r_4 + r_5 - r_6 \tag{C.3}$$

$$\frac{d[\mathrm{H_2}]}{dt} = -r_3 + r_4 \tag{C.4}$$

$$\frac{d[\mathrm{H}\cdot]}{dt} = r_3 - r_4 - r_5 + r_6 \tag{C.5}$$

$$\frac{d[\mathrm{HBr}]}{dt} = r_3 - r_4 + r_5 - r_6 \tag{C.6}$$

$$\frac{d[\mathrm{M}]}{dt} = 0 \tag{C.7}$$

C.1.1 Initial Values

After the construction of the ODEs, we only have to specify the initial values at the initial time t_0. For the Hydrogen-Bromine CRN we take the values of Vajda et al. [84]:

$$\begin{aligned} [\mathrm{Br_2}](t_0) &= 1 \cdot 10^{-8} \tfrac{\mathrm{mol}}{\mathrm{cm}^3}, \\ [\mathrm{Br}\cdot](t_0) &= 0, \\ [\mathrm{H_2}](t_0) &= 1 \cdot 10^{-8} \tfrac{\mathrm{mol}}{\mathrm{cm}^3}, \\ [\mathrm{H}\cdot](t_0) &= 0, \\ [\mathrm{HBr}](t_0) &= 0, \\ [\mathrm{M}](t_0) &= 1 \cdot 10^{-5} \tfrac{\mathrm{mol}}{\mathrm{cm}^3}. \end{aligned} \tag{C.8}$$

C.2 Reduction of formaldehyde oxidation reaction network

Formaldehyde oxidation in the presence of carbon-monoxide is a medium size reaction network which consists of 25 reactions listed with the corresponding rate coefficients in Table C.3. The detailed reaction network was published by Vardanyan [168] and used for model reduction by Turányi [84]. In this section the model reduction of this reaction system on two different time horizons is shown.

The species in the network are HCO, O_2, HO_2, CO, CH_2O, H_2O_2, M which is a kind of catalyst, OH, H_2O, CO_2, H, H_2, O, and finally *Destruction* which is a *sink* for reaction (6) and (7). From this list of species, 9 species (HCO, HO_2, H_2O_2, OH, H_2O, CO_2, H, H_2, O) were chosen as important. The initial conditions for the reaction network are $[O_2]_0 = 1.27 \cdot 10^{18}\ cm^{-3}$, $[CO]_0 = 2.83 \cdot 10^{18}\ cm^{-3}$, $[CH_2O]_0 = 6.77 \cdot 10^{16}\ cm^{-3}$, $[M]_0 = 7.09 \cdot 10^{18}\ cm^{-3}$ and zero for the other species, the same as in [84].

Two different time horizons were chosen for model reduction, a shorter is $[0,\ 5 \cdot 10^{-3}]$ seconds the same as presented in [84] while the longer [0, 0.1] seconds shows much more colorful dynamic behavior. In both cases the sensitivity part was included in the computation, i.e. $\sigma = 1$ in Eq. (4.23). For the solution of IVP (4.1)–(4.2), an absolute tolerance of $AbsTol = 10^{-14}$ and a relative tolerance of $RelTol = 10^{-10}$ were set.

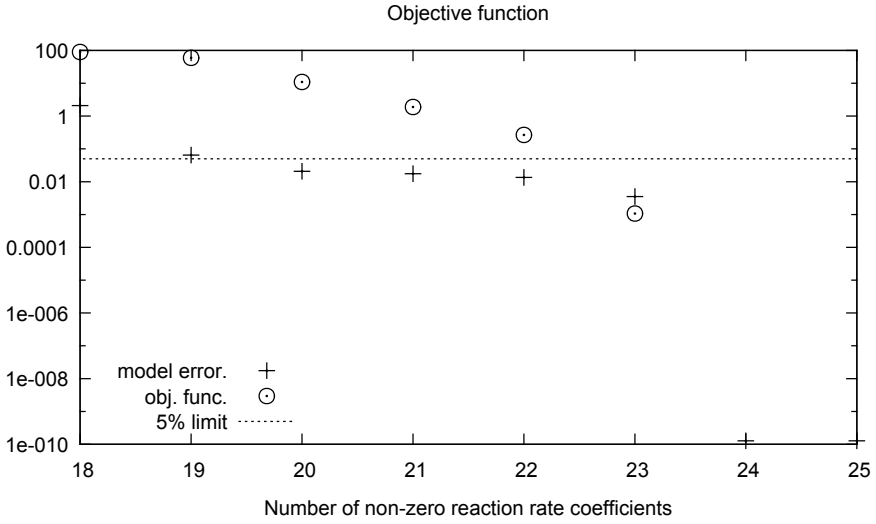


Figure C.1: Model error and objective function against the number of non-zero reaction coefficients in case of the formaldehyde reaction network in the longer time horizon

C.2.1 Model reduction in longer time horizon

In this time horizon $N = 1500$ equidistant time points were selected and the weighting factors in Eq. (4.8) are

$$w_{il} = \frac{\nu}{N \cdot \max(10^{-2}, x_i(t_l))}, \quad \nu = 10^{-10},$$

where the choice of $\nu = 10^{-10}$ in the numerator is introduced to avoid large eigenvalues of H. The choice $\nu = 1$ would result in a mathematically equivalent optimization problem. However, according to our experiences, the MIQP solver of CPLEX has computational difficulties with large eigenvalues which is the only reason for our particular choice of ν. In other respects, again apart from the "max"-term to avoid by-zero division, these weights equally reflect the relative error of the important species, an approach also followed by Androulakis [83].

The solution of the sequence of MIQPs resulted in the set of objective function value as a function of maximal number of non-zero reaction coefficients $\tilde{k}(q)$. The objective function (4.32), the model error (4.43) together with the 5 % limit can be seen in Figure C.1. One can conclude that on the specified level of acceptance the reaction network can be reduced by 5 reactions. We have depicted the important concentration trajectories which belongs to the original reaction network together with the trajectories of the reduced models in Figure C.2. The reduced model is in good agreement with the original one. The corresponding estimated parameter values can be found in Table C.3. in the column A5.

Table C.2: Relative deviation of the important species in the reduced models and in the reference paper.

Important species	Relative deviation in concentration		
	A5	B12	Turányi [84]
HCO	0	0.008	0.023
HO_2	0.054	0.006	0.024
H_2O_2	0.001	0.031	0.019
OH	0.036	0.009	0.023
H_2O	0.006	0.016	0.016
CO_2	0.004	0.029	0.017
H	0.042	0.031	0.036
H_2	0.001	0.004	0.01
O	0.001	0.009	0.004

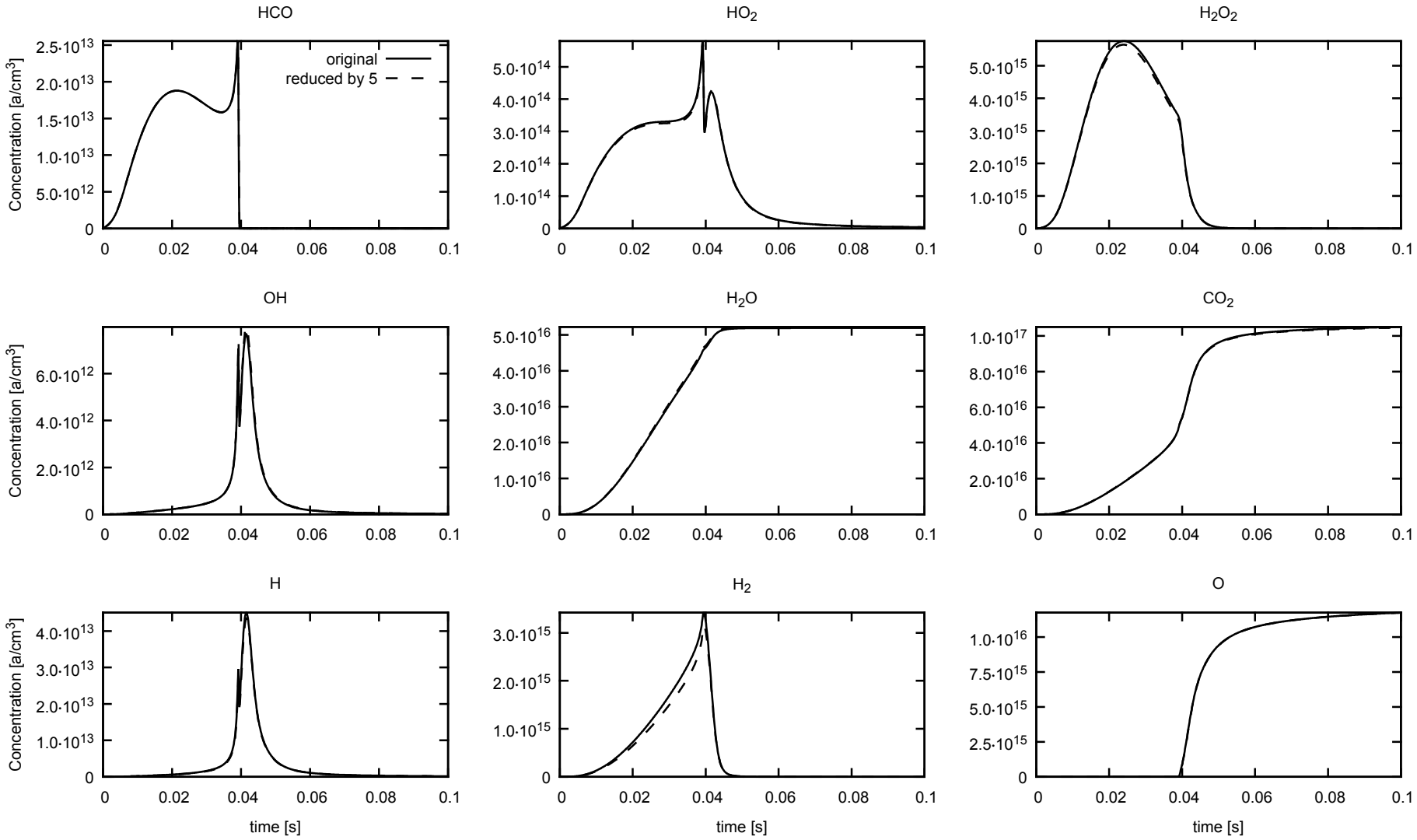


Figure C.2: The concentrations of each important species in the original system and in the reduced systems in the longer time horizon.

Table C.3: Rate coefficients of the Formaldehyde case study

	Reactions	**Original coeff.**	**A5** [0, 0.1 s]	**B12** [0, $5 \cdot 10^{-3}$ s]	[84]
(1)	$HCO + O_2 \rightarrow HO_2 + CO$	$1.00 \cdot 10^{-13}$	0.999	1.0051	1
(2)	$HO_2 + CH_2O \rightarrow H_2O_2 + HCO$	$5.70 \cdot 10^{-14}$	0.9993	1.0012	1
(3)	$H_2O_2 + M \rightarrow 2\ OH + M$	$6.66 \cdot 10^{-18}$	1.0065	0.9925	1
(4)	$OH + CH_2O \rightarrow H_2O + HCO$	$1.60 \cdot 10^{-10}$	1.0019	1.012	1
(5)	$OH + H_2O_2 \rightarrow H_2O + HO_2$	$5.10 \cdot 10^{-12}$	2.0844	0	0
(6)	$H_2O_2 \rightarrow$ Destruction	$1.05 \cdot 10^{2}$	1.0011	0.808	1
(7)	$HO_2 \rightarrow$ Destruction	$1.05 \cdot 10^{1}$	1.2011	0	0
(8)	$2\ HO_2 \rightarrow H_2O_2 + O_2$	$3.00 \cdot 10^{-12}$	1.0174	1.0398	1
(9)	$OH + CO \rightarrow CO_2 + H$	$3.30 \cdot 10^{-13}$	0.943	0.7241	1
(10)	$HO_2 + CO \rightarrow CO_2 + OH$	$1.20 \cdot 10^{-15}$	1.0316	0.9909	1
(11)	$H + CH_2O \rightarrow H_2 + HCO$	$2.70 \cdot 10^{-12}$	0.9493	0.9923	1
(12)	$H + O_2 \rightarrow OH + O$	$5.51 \cdot 10^{-14}$	1.0217	0.9354	1
(13)	$H + O_2 + M \rightarrow HO_2 + M$	$1.00 \cdot 10^{-32}$	1.0173	0	1
(14)	$HO_2 + M \rightarrow H + O_2 + M$	$4.70 \cdot 10^{-19}$	0	0	0
(15)	$O + H_2 \rightarrow OH + H$	$3.02 \cdot 10^{-13}$	1.009	0*	0
(16)	$O + CH_2O \rightarrow OH + HCO$	$1.00 \cdot 10^{-10}$	0.9986	0.9341	1
(17)	$H + H_2O_2 \rightarrow HO_2 + H_2$	$1.30 \cdot 10^{-12}$	0	0	0
(18)	$H + H_2O_2 \rightarrow H_2O + OH$	$5.90 \cdot 10^{-12}$	1.1459	0	0
(19)	$O + H_2O_2 \rightarrow OH + HO_2$	$1.00 \cdot 10^{-13}$	1.0472	0	0
(20)	$HCO \rightarrow H + CO$	$4.60 \cdot 10^{-12}$	0	0	0
(21)	$OH + H_2 \rightarrow H_2O + H$	$1.00 \cdot 10^{-11}$	0*	0*	0
(22)	$CH_2O + O_2 \rightarrow HCO + HO_2$	$2.90 \cdot 10^{-20}$	0.9934	0.9967	1
(23)	$H + HO_2 \rightarrow 2\ OH$	$5.00 \cdot 10^{-12}$	0	0	0
(24)	$H + HO_2 \rightarrow H_2O + O$	$5.00 \cdot 10^{-11}$	0.9852	0	0
(25)	$H + HO_2 \rightarrow H_2 + O_2$	$4.50 \cdot 10^{-11}$	0.9559	0	0

Rate coefficients of the original system and the relative rate coefficients in the reduced systems. In the A5 case the system was reduced by 5 reactions. In the B12 case the reduced system contains 12 reactions in the shorter time horizon. Reactions denoted by (*) were pre-reduced.

C.2.2 Model reduction in shorter time horizon

The model reduction on the shorter time horizon using principal component analysis (PCA) was presented in [84]. The author concluded that the minimal reaction network which can successfully describe the original dynamics of the important species consists of 13 reactions.

For the computation of matrix H, 100 equally distributed time points along the interval were selected. The weighting factors in Eq. (4.8) are

$$w_{il} = \frac{\tilde{w}_i}{N \cdot \max(1, x_i(t_l))},$$

where $\tilde{w}_i = 10$ for $i \in \{3, 8, 9\}$ and $\tilde{w}_i = 1$ for the other species. Again, apart from the "max"-term to avoid division by zero and apart from the important specis x_3, x_8, x_9, these

weight equally reflect the relative error of the important species, an approach also followed by Androulakis [83]. The increase of the weights for x_3, x_8, x_9 are just a result of some heuristic tuning to achieve a better reduction.

One can find the model error as a function of the number of non-zero rate coefficients in Figure C.3 according to which the model can be reduced to 12 reactions while the average relative deviations of the important species are around 1%. The estimated parameter values are in Table C.3, column B12. If we compare the two last columns of the table we find that the proposed method found the same unnecessary parameters as presented in [84] except for the 13^{th} parameter. The relative deviations for the concentrations are depicted in Table C.2 from which it can be clearly seen that the simultaneous estimation of the parameters resulted in a better fit than only omitting the unnecessary reactions using the PCA.

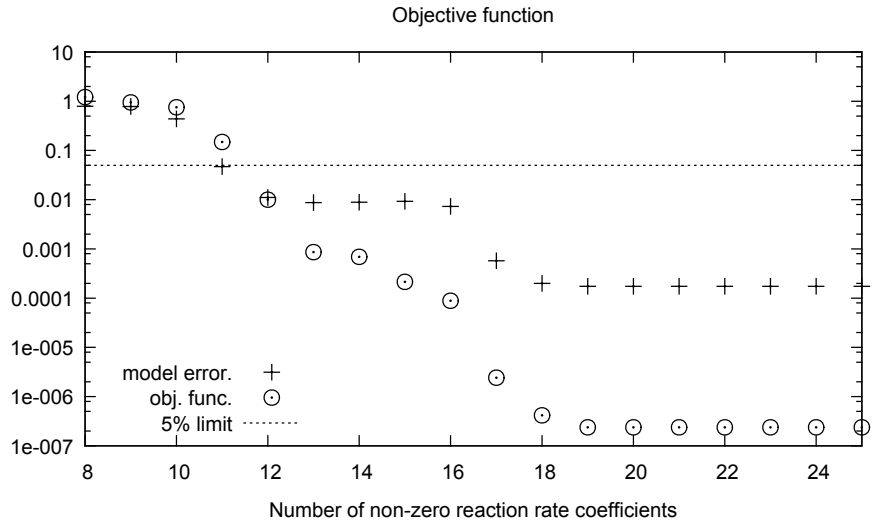


Figure C.3: Objective function and model error versus the number of non-zero reaction coefficients in the shorter time horizon.

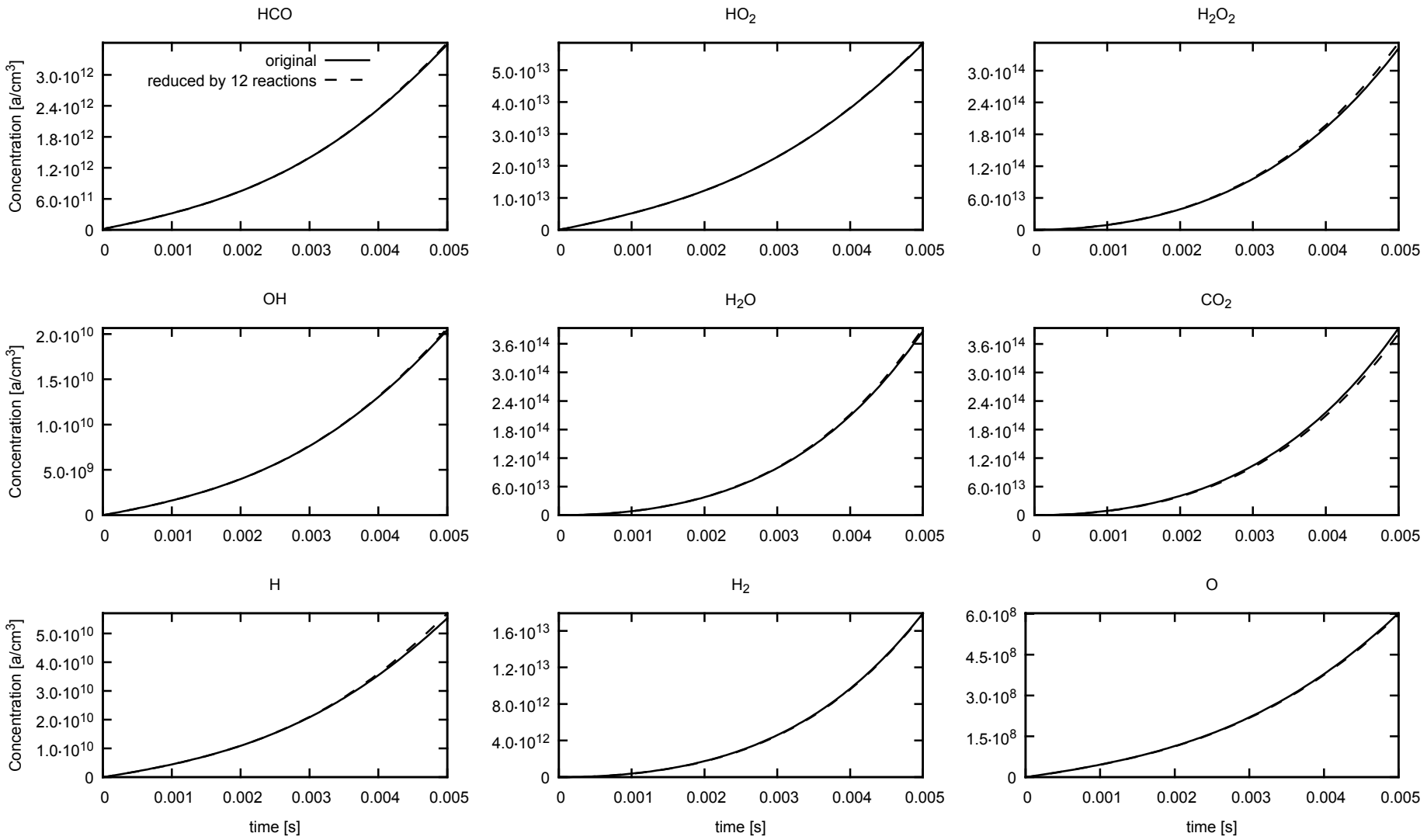


Figure C.4: The concentrations of each important species in the original system and in the reduced systems in the shorter time horizon.

C.3 Reduction of the alkane pyrolysis reaction network

The alkane pyrolysis is a large reaction network consisting of 98 reactions and 32 species. The reactions and the corresponding rate coefficients can be found in Table C.4. The reduction of this network using Principal Component Analysis was presented in [89]. The author could reduce the original 98 reactions to 38 reactions, while the concentrations of the important species did not change more than 1 %. As the author remarked the reaction network has no formation steps for five species, which resulted in that reactions 6, 74 76, 78, 82, 86, 87, 96 and 98 can be omitted because the corresponding rates are zero. These reactions are marked with (#) symbol in Table C.4.

C.3.1 Initialization

During the model reduction the same initial conditions, time horizon and selected important species were used as presented in [89] to produce comparable results. The following species were chosen as important: C_3H_8, H_2, CH_4, C_3H_6, C_2H_6, C_2H_4. The time horizon is [0, 100] seconds along which the H matrix was computed from 1000 equidistant time points using sensitivity information, i.e. $\sigma = 1$ in Eq. (4.23). The initial conditions are $1.912 \cdot 10^{-3}\, mol/dm^3$ for the Propane (C_3H_8) and zero for all the other species. The weighting factors in Eq. (4.8) are

$$w_{il} = (1 - \frac{\delta_{N0} + \delta_{Nl}}{2}) \frac{\tilde{w}_i}{N \cdot \max(1, x_i(t_l))}, \quad i \in \mathcal{I}, \quad l = 0, \ldots, N$$

where $\tilde{w}_i = 5 \cdot 10^{-4}$ for $i = 5$ and $\tilde{w}_i = 5 \cdot 10^{-3}$ for the other species. Here the factor $(1 - (\delta_{N0} + \delta_{Nl})/2)$ equals 1, for $l = 1, \ldots, N-1$ and $1/2$ for $l = 0$ or $l = N$, where δ is the Kronecker symbol. By this way, the sum over l in Eq. (4.8) becomes a trapezoidal sum.

For the solution of IVP (4.1)–(4.2), $AbsTol = 10^{-20}$ as absolute tolerance and $RelTol = 10^{-10}$ as relative tolerance were set.

C.3.2 Pre-reduction

The size of the reaction network necessitated the usage of the pre-reduction described in section 4.3.1. We have found that the optimal number of reactions which should be omitted during the pre-reduction is 57. The corresponding reaction coefficients are marked with stars (*) in Table C.4.

C.3.3 Results

In the Figure C.5 one can find the objective function and the model error as a function of the number of non-zero reaction coefficients. It suggests that if we accept less than 5 % model error then the model can be reduced to 23 reactions, which is 15 reactions less than we find in [89]. This is a remarkable result which shows the advantage of the simultaneous reduction and parameter estimation.

The trajectories of the important species in the reduced model together with the original concentrations are depicted in Figure C.6. The fitting of the concentrations' trajectories is almost perfect for all species.

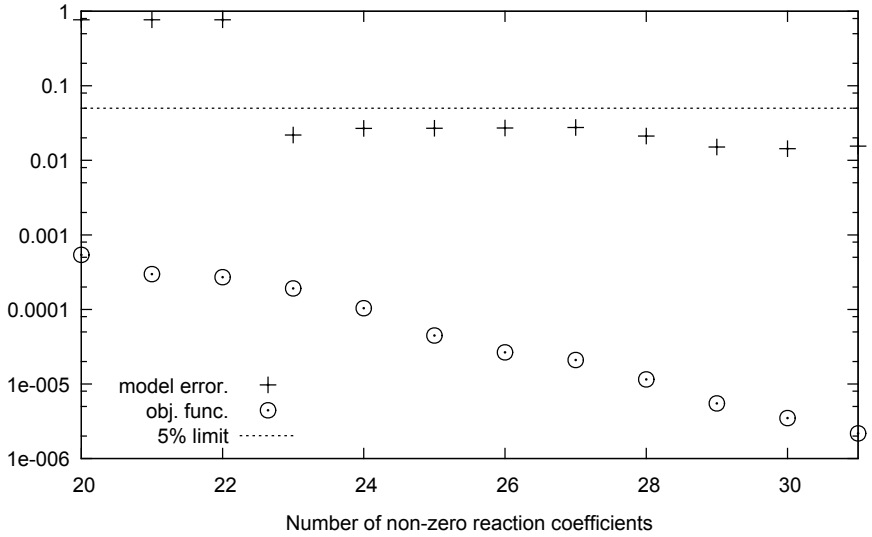


Figure C.5: The objective function value versus the number of non-zero reaction coefficients.

Table C.4: Rate Coefficients of the alkane pyrolysis case study

	Reactions	**Rate coefficients**	
		Original	Reduced
(1)	$C_3H_8 \rightarrow CH_3\cdot + C_2H_5\cdot$	$1.3203 \cdot 10^{-06}$	0.9903
(2)	$1\text{-}C_4H_8 \rightarrow CH_3\cdot + C_2H_5\cdot$	$1.5304 \cdot 10^{-04}$	0*
(3)	$1\text{-}C_5H_{10} \rightarrow C_2H_5\cdot + C_3H_5\cdot$	$7.7239 \cdot 10^{-04}$	0*
(4)	$1\text{-}C_6H_{12} \rightarrow 1\text{-}C_3H_7\cdot + C_3H_5\cdot$	$2.3342 \cdot 10^{-03}$	0*
(5)	$(C_3H_5)_2 \rightarrow C_3H_5\cdot + C_3H_5\cdot$	$3.1620 \cdot 10^{-02}$	1.0082
(6)#	1,5-heptadiene$\rightarrow C_3H_5\cdot + C_4H_7\cdot$	$9.7127 \cdot 10^{-01}$	-
(7)	$n\text{-}C_4H_{10} \rightarrow CH_3\cdot + 1\text{-}C_3H_7\cdot$	$3.6312 \cdot 10^{-06}$	0*
(8)	$n\text{-}C_4H_{10} \rightarrow C_2H_5\cdot + C_2H_5\cdot$	$2.6527 \cdot 10^{-06}$	0*
(9)	$H_2 + CH_3\cdot \rightarrow H\cdot + CH_4$	$1.4971 \cdot 10^{+06}$	0*
(10)	$H_2 + C_3H_5\cdot \rightarrow H\cdot + C_3H_6$	$1.6856 \cdot 10^{+05}$	1.138
(11)	$C_3H_8 + H\cdot \rightarrow 1\text{-}C_3H_7\cdot + H_2$	$3.1881 \cdot 10^{+08}$	0.5922
(12)	$C_3H_8 + CH_3\cdot \rightarrow 1\text{-}C_3H_7\cdot + CH_4$	$8.3505 \cdot 10^{+05}$	0.7493
(13)	$C_3H_8 + C_2H_5\cdot \rightarrow 1\text{-}C_3H_7\cdot + C_2H_6$	$1.6128 \cdot 10^{+05}$	1.5675
(14)	$C_3H_8 + 2\text{-}C_3H_7\cdot \rightarrow 1\text{-}C_3H_7\cdot + C_3H_8$	$3.5234 \cdot 10^{+04}$	1.7239
(15)	$C_3H_8 + 1\text{-}C_3H_7\cdot \rightarrow 2\text{-}C_3H_7\cdot + C_3H_8$	$8.2443 \cdot 10^{+04}$	0
(16)	$C_3H_8 + C_3H_5\cdot \rightarrow 1\text{-}C_3H_7\cdot + C_3H_6$	$2.5859 \cdot 10^{+03}$	0.8467
(17)	$C_3H_6 + H\cdot \rightarrow C_3H_5\cdot + H_2$	$1.1565 \cdot 10^{+10}$	0.5149
(18)	$C_3H_8 + H\cdot \rightarrow 2\text{-}C_3H_7\cdot + H_2$	$5.4813 \cdot 10^{+08}$	0.5551
(19)	$C_3H_8 + CH_3\cdot \rightarrow 2\text{-}C_3H_7\cdot + CH_4$	$9.7586 \cdot 10^{+05}$	1.2128
(20)	$C_3H_8 + C_2H_5\cdot \rightarrow 2\text{-}C_3H_7\cdot + C_2H_6$	$8.2443 \cdot 10^{+04}$	0

	Reactions	Rate coefficients	
		Original	Reduced
(21)	$C_3H_8 + C_3H_5\cdot \rightarrow 2\text{-}C_3H_7\cdot + C_3H_6$	$9.7813 \cdot 10^{+03}$	1.0561
(22)	$H_2 + C_2H_5\cdot \rightarrow H\cdot + C_2H_6$	$7.1206 \cdot 10^{+05}$	0*
(23)	$C_2H_6 + CH_3\cdot \rightarrow C_2H_5\cdot + CH_4$	$4.9539 \cdot 10^{+05}$	0*
(24)	$n\text{-}C_4H_{10} + H\cdot \rightarrow 1\text{-}C_4H_9\cdot + H_2$	$3.1881 \cdot 10^{+08}$	0*
(25)	$n\text{-}C_4H_{10} + CH_3\cdot \rightarrow 1\text{-}C_4H_9\cdot + CH_4$	$9.8842 \cdot 10^{+05}$	0*
(26)	$n\text{-}C_4H_{10} + C_2H_5\cdot \rightarrow 1\text{-}C_4H_9\cdot + C_2H_6$	$1.6128 \cdot 10^{+05}$	0*
(27)	$n\text{-}C_4H_{10} + C_3H_5\cdot \rightarrow 1\text{-}C_4H_9\cdot + C_3H_6$	$2.5859 \cdot 10^{+03}$	0*
(28)	$C_3H_6 + CH_3\cdot \rightarrow C_3H_5\cdot + CH_4$	$6.9894 \cdot 10^{+05}$	1.7449
(29)	$C_3H_6 + C_2H_5\cdot \rightarrow C_3H_5\cdot + C_2H_6$	$2.3810 \cdot 10^{+05}$	0*
(30)	$n\text{-}C_4H_{10} + H\cdot \rightarrow 2\text{-}C_4H_9\cdot + H_2$	$1.0937 \cdot 10^{+09}$	0*
(31)	$n\text{-}C_4H_{10} + CH_3\cdot \rightarrow 2\text{-}C_4H_9\cdot + CH_4$	$1.0723 \cdot 10^{+06}$	0*
(32)	$n\text{-}C_4H_{10} + C_2H_5\cdot \rightarrow 2\text{-}C_4H_9\cdot + C_2H_6$	$1.6450 \cdot 10^{+05}$	0*
(33)	$n\text{-}C_4H_{10} + C_3H_5\cdot \rightarrow 2\text{-}C_4H_9\cdot + C_3H_6$	$1.2885 \cdot 10^{+04}$	0*
(34)	$C_2H_5\cdot \rightarrow H\cdot + C_2H_4$	$5.7421 \cdot 10^{+02}$	0.9013
(35)	$1\text{-}C_3H_7\cdot \rightarrow CH_3\cdot + C_2H_4$	$2.5147 \cdot 10^{+04}$	1.4012
(36)	$1\text{-}C_3H_7\cdot \rightarrow H\cdot + C_3H_6$	$7.3732 \cdot 10^{+02}$	0
(37)	$2\text{-}C_3H_7\cdot \rightarrow H\cdot + C_3H_6$	$1.2185 \cdot 10^{+03}$	1.0313
(38)	$2\text{-}C_4H_9\cdot \rightarrow CH_3\cdot + C_3H_6$	$2.5889 \cdot 10^{+05}$	0*
(39)	$\text{2-methyl-1-propyl}\cdot \rightarrow CH_3\cdot + C_3H_6$	$1.6603 \cdot 10^{+05}$	0*
(40)	$\text{3-methyl-1-butyl}\cdot \rightarrow 2\text{-}C_3H_7\cdot + C_2H_4$	$3.8624 \cdot 10^{+05}$	0*
(41)	$\text{4-methyl-2-pentyl}\cdot \rightarrow 2\text{-}C_3H_7\cdot + C_3H_6$	$7.1039 \cdot 10^{+05}$	0*
(42)	$1\text{-}C_4H_9\cdot \rightarrow C_2H_5\cdot + C_2H_4$	$4.9079 \cdot 10^{+05}$	0*
(43)	$1\text{-}C_5H_{11}\cdot \rightarrow 1\text{-}C_3H_7\cdot + C_2H_4$	$7.9061 \cdot 10^{+05}$	0*
(44)	$2\text{-}C_5H_{11}\cdot \rightarrow C_2H_5\cdot + C_3H_6$	$8.1393 \cdot 10^{+04}$	0*
(45)	$\text{2-methyl-1-butyl}\cdot \rightarrow C_2H_5\cdot + C_3H_6$	$1.5978 \cdot 10^{+05}$	0*
(46)	$H\cdot + C_2H_4 \rightarrow C_2H_5\cdot$	$8.0175 \cdot 10^{+09}$	0.4892
(47)	$H\cdot + C_3H_6 \rightarrow 2\text{-}C_3H_7\cdot$	$3.7913 \cdot 10^{+09}$	0*
(48)	$H\cdot + C_3H_6 \rightarrow 1\text{-}C_3H_7\cdot$	$1.3296 \cdot 10^{+09}$	0
(49)	$CH_3\cdot + C_2H_4 \rightarrow 1\text{-}C_3H_7\cdot$	$1.0937 \cdot 10^{+06}$	0*
(50)	$CH_3\cdot + C_3H_6 \rightarrow 2\text{-}C_4H_9\cdot$	$3.3051 \cdot 10^{+06}$	0*
(51)	$CH_3\cdot + C_3H_6 \rightarrow \text{2-methyl-1-propyl}\cdot$	$1.1591 \cdot 10^{+06}$	0*
(52)	$C_2H_5\cdot + C_2H_4 \rightarrow 1\text{-}C_4H_9\cdot$	$5.8298 \cdot 10^{+05}$	0*
(53)	$C_2H_5\cdot + C_3H_6 \rightarrow 2\text{-}C_5H_11\cdot$	$3.9122 \cdot 10^{+05}$	0*
(54)	$1\text{-}C_3H_7\cdot + C_3H_6 \rightarrow 2\text{-}C_6H_13\cdot$	$8.4384 \cdot 10^{+05}$	0*
(55)	$2\text{-}C_3H_7\cdot + C_2H_4 \rightarrow \text{3-methyl-1-butyl.}$	$5.6627 \cdot 10^{+05}$	0*
(56)	$2\text{-}C_3H_7\cdot + C_3H_6 \rightarrow \text{4-methyl-2-pentyl.}$	$5.6627 \cdot 10^{+05}$	0*
(57)	$C_2H_5\cdot + C_3H_6 \rightarrow \text{2-methyl-1-butyl.}$	$1.3720 \cdot 10^{+05}$	0*
(58)	$1\text{-}C_5H_11\cdot \rightarrow 2\text{-}C_5H_11.$	$4.4305 \cdot 10^{+05}$	0*
(59)	$2\text{-}C_5H_11\cdot \rightarrow 1\text{-}C_5H_11.$	$6.8604 \cdot 10^{+04}$	0*
(60)	$1\text{-}C_6H_13\cdot \rightarrow 2\text{-}C_6H_13.$	$1.6935 \cdot 10^{+06}$	0*
(61)	$2\text{-}C_6H_13\cdot \rightarrow 1\text{-}C_6H_13.$	$2.6223 \cdot 10^{+05}$	0*
(62)	$H\cdot + 2\text{-}C_3H_7\cdot \rightarrow C_3H_8$	$1.0000 \cdot 10^{+11}$	0*
(63)	$CH_3\cdot + CH_3\cdot \rightarrow C_2H_6$	$2.5119 \cdot 10^{+10}$	0

	Reactions	Rate coefficients	
		Original	Reduced
(64)	$CH_3\cdot$ + 1-$C_3H_7\cdot \rightarrow$ n-C_4H_10	$1.9953 \cdot 10^{+10}$	0*
(65)	$CH_3\cdot$ + 2-$C_3H_7\cdot \rightarrow$ 2-methylpropane	$1.5849 \cdot 10^{+10}$	2.5848
(66)	$CH_3\cdot$ + $C_3H_5\cdot \rightarrow$ 1-C_4H_8	$1.9953 \cdot 10^{+10}$	1.4427
(67)	$C_2H_5\cdot$ + 2-$C_3H_7\cdot \rightarrow$ 2-methylbutane	$7.9433 \cdot 10^{+09}$	0*
(68)	$C_2H_5\cdot$ + $C_3H_5\cdot \rightarrow$ 1-C_5H_10	$1.0000 \cdot 10^{+10}$	1.2315
(69)	1-$C_3H_7\cdot$ + 2-$C_3H_7\cdot \rightarrow$ 2-methylpentane	$7.9433 \cdot 10^{+09}$	0
(70)	1-$C_3H_7\cdot$ + $C_3H_5\cdot \rightarrow$ 1-C_6H_12	$1.0000 \cdot 10^{+10}$	0*
(71)	2-$C_3H_7\cdot$ + 2-$C_3H_7\cdot \rightarrow$ product	$3.1623 \cdot 10^{+09}$	0
(72)	2-$C_3H_7\cdot$ + $C_3H_5\cdot \rightarrow$ product	$1.0000 \cdot 10^{+10}$	0.9624
(73)	$C_3H_5\cdot$ + $C_3H_5\cdot \rightarrow$ (C_3H_5)2	$6.3096 \cdot 10^{+09}$	1.1095
(74)#	$C_3H_5\cdot$ + $C_4H_7\cdot \rightarrow$ 1,5-heptadiene	$1.2589 \cdot 10^{+10}$	-
(75)	$CH_3\cdot$ + $C_2H_5\cdot \rightarrow$ C_3H_8	$1.9953 \cdot 10^{+10}$	0*
(76)#	$CH_3\cdot$ + $C_4H_7\cdot \rightarrow$ 2-C_5H_10	$2.5119 \cdot 10^{+10}$	-
(77)	$C_2H_5\cdot$ + $C_2H_5\cdot \rightarrow$ n-C_4H_10	$3.9811 \cdot 10^{+09}$	0*
(78)#	$C_2H_5\cdot$ + $C_4H_7\cdot \rightarrow$ 2-C_6H_12	$1.0000 \cdot 10^{+10}$	-
(79)	H$\cdot$ + 2-$C_3H_7\cdot \rightarrow$ C_3H_6 + H_2	$5.0119 \cdot 10^{+10}$	0*
(80)	$CH_3\cdot$ + 1-$C_3H_7\cdot \rightarrow$ C_3H_6 + CH_4	$1.2589 \cdot 10^{+09}$	0*
(81)	$CH_3\cdot$ + 2-$C_3H_7\cdot \rightarrow$ C_3H_6 + CH_4	$2.5119 \cdot 10^{+09}$	0
(82)#	$CH_3\cdot$ + $C_4H_7\cdot \rightarrow$ C_4H_6 + CH_4	$7.9433 \cdot 10^{+09}$	
(83)	$C_2H_5\cdot$ + 2-$C_3H_7\cdot \rightarrow$ C_3H_6 + C_2H_6	$1.5849 \cdot 10^{+09}$	0*
(84)	1-$C_3H_7\cdot$ + 2-$C_3H_7\cdot \rightarrow$ C_3H_6 + C_3H_8	$1.5849 \cdot 10^{+09}$	0*
(85)	2-$C_3H_7\cdot$ + 2-$C_3H_7\cdot \rightarrow$ C_3H_6 + C_3H_8	$1.9953 \cdot 10^{+09}$	0.6416
(86)#	2-$C_3H_7\cdot$ + $C_4H_7\cdot \rightarrow$ C_4H_6 + C_3H_8	$5.0119 \cdot 10^{+09}$	-
(87)#	$C_3H_5\cdot$ + $C_4H_7\cdot \rightarrow$ C_4H_6 + C_3H_6	$6.3096 \cdot 10^{+09}$	-
(88)	$C_3H_5\cdot$ + $C_2H_5\cdot \rightarrow$ C_2H_4 + C_3H_6	$1.2589 \cdot 10^{+09}$	0*
(89)	$C_3H_5\cdot$ + 1-$C_3H_7\cdot \rightarrow$ C_3H_6 + C_3H_6	$1.0000 \cdot 10^{+09}$	0*
(90)	$C_3H_5\cdot$ + 2-$C_3H_7\cdot \rightarrow$ C_3H_6 + C_3H_6	$1.0000 \cdot 10^{+09}$	0
(91)	$C_2H_5\cdot$ + 2-$C_3H_7\cdot \rightarrow$ C_3H_8 + C_2H_4	$1.2589 \cdot 10^{+09}$	0*
(92)	1-$C_3H_7\cdot$ + 2-$C_3H_7\cdot \rightarrow$ C_3H_8 + C_3H_6	$1.2589 \cdot 10^{+09}$	0 *
(93)	$CH_3\cdot$ + $C_2H_5\cdot \rightarrow$ C_2H_4 + CH_4	$7.9433 \cdot 10^{+08}$	0*
(94)	$C_2H_5\cdot$ + $C_2H_5\cdot \rightarrow$ C_2H_4 + C_2H_6	$5.0119 \cdot 10^{+08}$	0*
(95)	$C_2H_5\cdot$ + 2-$C_4H_9\cdot \rightarrow$ 2-C_4H_8 + C_2H_6	$1.5849 \cdot 10^{+09}$	0*
(96)#	$C_2H_5\cdot$ + $C_4H_7\cdot \rightarrow$ C_4H_6 + C_2H_6	$3.9811 \cdot 10^{+09}$	-
(97)	$C_2H_5\cdot$ + 2-$C_4H_9\cdot \rightarrow$ n-C_4H_10 + C_2H_4	$7.9433 \cdot 10^{+08}$	0*
(98)#	$C_2H_5\cdot$ + $C_4H_7\cdot \rightarrow$ C_2H_4 + olefin	$1.0000 \cdot 10^{+09}$	-

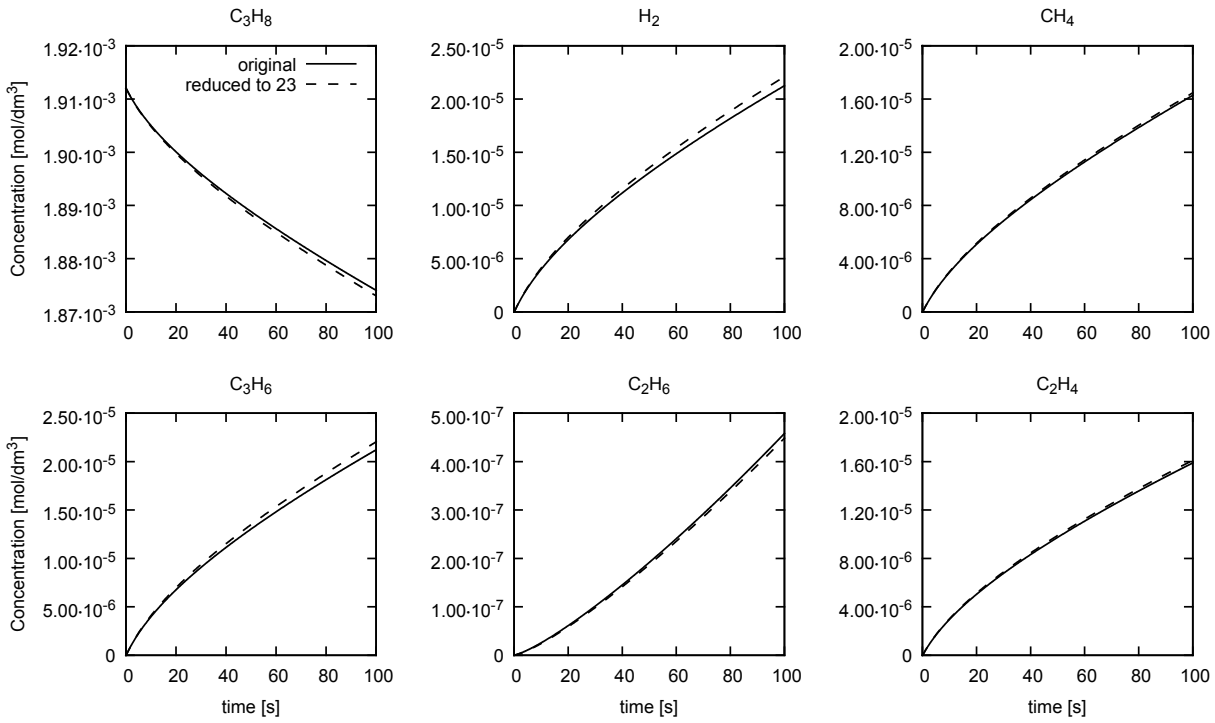


Figure C.6: The concentrations of each important species in the original system and in the reduced system.

Appendix D

Regularized parameter estimation

D.1 Bias-variance decomposition of the prediction error

In this section we utilize the bias-variance decomposition of the prediction error and show when and how regularization can lead to smaller prediction error. First, let us introduce the subscript $\mathcal{C}$ for the calibration data and the subscript $\mathcal{V}$ for the validation data. For notational simplicity we consider only one experiment and only one observable variable for both of the calibration and validation scenario, but it is straightforward to generalize for multiple experiments and observables. The expected prediction error (PE) for the validation data can be written as

$$\mathrm{PE} = \mathbb{E}_{\mathcal{V},\mathcal{C}}\left[(\tilde{y}_{\mathcal{V}} - \hat{y}_{\mathcal{V}}(\hat{\theta}_{\mathcal{C}}))^2\right] \tag{D.1}$$

where $\tilde{y}_{\mathcal{V}}$ is the validation data, $\hat{\theta}_{\mathcal{C}}$ is the estimated parameter vector based on the calibration data and $\hat{y}_{\mathcal{V}}(\hat{\theta}_{\mathcal{C}})$ is the model predictions for the validation data. The prediction error depends on the calibration data –different calibration data would result in different estimated parameters $\hat{\theta}_{\mathcal{C}}$– and also depends on the validation data. Thus the expectation is taken over the distribution of the calibration and the validation data. The measurement error in the calibration data and in the validation data is often independent, leading to the well-known (see for example [133, 169, 170]) bias-variance decomposition of expected prediction error as

$$\mathrm{PE} = \underbrace{\mathbb{E}_{\mathcal{V}}\left[\left(y_{\mathcal{V}} - \mathbb{E}_{\mathcal{C}}\left[\hat{y}_{\mathcal{V}}(\hat{\theta}_{\mathcal{C}})\right]\right)^2\right]}_{\mathrm{Bias}^2} + \underbrace{\mathbb{E}_{\mathcal{V}}\left[\left(\hat{y}_{\mathcal{V}}(\hat{\theta}_{\mathcal{C}}) - \mathbb{E}_{\mathcal{C}}\left[\hat{y}_{\mathcal{V}}(\hat{\theta}_{\mathcal{C}})\right]\right)^2\right]}_{\mathrm{Variance}} + \mathbb{E}_{\mathcal{V}}\left[\epsilon^2\right]. \tag{D.2}$$

Here, the first term corresponds to the squared bias of the calibrated model predictions from the true validation data $y_{\mathcal{V}}$, the second term is the variance of the model prediction, and the third term is the contribution of the measurement error $\mathbb{E}_{\mathcal{V}}\left[\epsilon^2\right] = \sigma^2$.

The variance term. The variance of the prediction is due to the uncertainty in the parameter estimates. This uncertainty can be especially large if the calibration data is scarce and the number of data points is close to the number of parameters. The variance

term can be expressed for unbiased estimates [171] as

$$\text{Variance} = \frac{N_\theta}{N_D}\sigma^2, \tag{D.3}$$

where N_θ is the number of estimated parameters and N_D is the number of calibration data. Each estimated parameter contributes by $\frac{\sigma^2}{N_D}$ to the prediction error, thus a model with fewer calibrated parameters would result smaller variance. For biased estimates the prediction variance becomes

$$\text{Variance} = \frac{N_\theta^{\text{eff}}(\alpha)}{N_D}\sigma^2, \tag{D.4}$$

where N_θ^{eff} is the effective number of parameters, which depends on the regularization penalty and regularization parameter α. In general, the effective number of parameters can be expressed by the second order derivatives of the objective function [170] with respect to the parameters as

$$N_\theta^{\text{eff}} = \text{trace}\left(H_{\text{LS}}(H_{\text{LS}} + \alpha H_\Gamma)^{-1} H_{\text{LS}}(H_{\text{LS}} + \alpha H_\Gamma)^{-1}\right), \tag{D.5}$$

where

$$H_{\text{LS}} = \mathbb{E}\left[\frac{\partial R(\theta)}{\partial \theta}^T \frac{\partial R(\theta)}{\partial \theta}|_{\theta=\theta_t}\right] \tag{D.6}$$

is known as the Gauss-Newton approximate of the Hessian and H_Γ is the Hessian of the regularization penalty function. Note, that (D.6) is also related to the Fisher Information matrix (FIM), which is often used in the practical identifiability and uncertainty analysis of the estimated parameters [137]. For example, the eigenvalue decomposition of the FIM can identify correlated estimated parameters and parameters with high uncertainty [157]. Small or zero eigenvalues (high condition number) indicates ill-posedness, i.e. the parameter estimation problem does not have a unique solution. This eigenvalue decomposition has been widely used in the estimation literature [122, 172–176].

In the special case of ridge regularization [171], i.e. $\Gamma(\theta) = \theta^T\theta$, the Hessian of the penalty is the identity matrix and Equation (D.5) simplifies to

$$N_\theta^{\text{eff}} = \sum_{i=1}^{N_\theta} \frac{\sigma_i^2}{(\sigma_i + \alpha)^2}, \tag{D.7}$$

where σ_i $(i = 1 \dots N_\theta)$ are the eigenvalues of H_{LS}. Note that for $\alpha = 0$ –the non-regularized case– the effective number of parameters equals to the number of model parameters and for $\alpha > 0$ –the regularized case– the effective number of parameters is less than the number of model parameters N_θ. *Thus, as the regularization parameter increases, the effective number of parameters decreases and therefore the variance term of the prediction error* (D.2) *decreases.*

The bias term. We saw above that regularization reduces the effective number of parameters, and therefore the variance of the prediction error. The cost to pay is the bias. Sjöberg and Ljung [171] derived an upper bound on the prediction bias for the non-weighted

Tikhonov regularization, i.e. when the penalty $\Gamma = (\theta - \theta^{\text{ref}})^T(\theta - \theta^{\text{ref}})$, where θ^{ref} is a reference parameter vector. It was shown that in this particular case the bias is

$$\text{Bias}^2 < \frac{\alpha}{8}||\theta_t - \theta^{\text{ref}}||^2 \tag{D.8}$$

where θ_t is the true model parameters. *Thus, the smaller the regularization parameter and the better our a priori knowledge is (expressed by the reference parameter vector), the smaller the bias that will be introduced in the estimation.*

The minimal prediction error. There is a trade-off between bias and variance. From Equations (D.2), (D.3) (D.4) and (D.8) one obtains that the reduced variance due to the regularization is larger than the introduced bias if the following inequality holds:

$$\sigma^2 \frac{N_\theta - N_\theta^{\text{eff}}(\alpha)}{N_D} > \frac{\alpha}{8}||\theta_t - \theta^{\text{ref}}||^2. \tag{D.9}$$

Therefore, *regularization generally increases the performance of the calibrated model when*

1. the calibration data is noisy (σ is large) and the amount of data is limited (N_D is small),

2. there are a large number of correlated parameters, and therefore the Hessian of the original problem has very small eigenvalues. In this case even a small regularization parameter can largely reduce the effective number of parameters, i.e. $N_\theta \gg N_\theta^{\text{eff}}(\alpha)$.

3. One has a good guess of the true parameters ($||\theta_t - \theta^{\text{ref}}||^2$ is small), for example from other independent experiments, previous studies or based on the biological or physico-chemical meaning of the parameters.

However, note that regularization may damage the prediction (the reduced variance is smaller than the introduced biased) if the original problem is not ill-posed, i.e. $N_\theta \approx N_\theta^{\text{eff}}(\alpha)$, α is set to a large value and the provided reference parameters are far from the true parameters.

D.2 Regularization schemes

Figure D.1 shows a summary of the proposed regularization schemes based on the available prior knowledge quality.

D.3 Settings of the optimization algorithms

Table D.1 shows the default settings for both the global optimization algorithm (eSS) and the local NLS algorithm (NL2SOL). Note that these values might be different from the default values of the algorithms, but the same values have been used for all the case studies (i.e. they were found to be robust settings). Interested readers can find further tuning details in eSS User's Manual and the NL2SOL User's Guide [177].

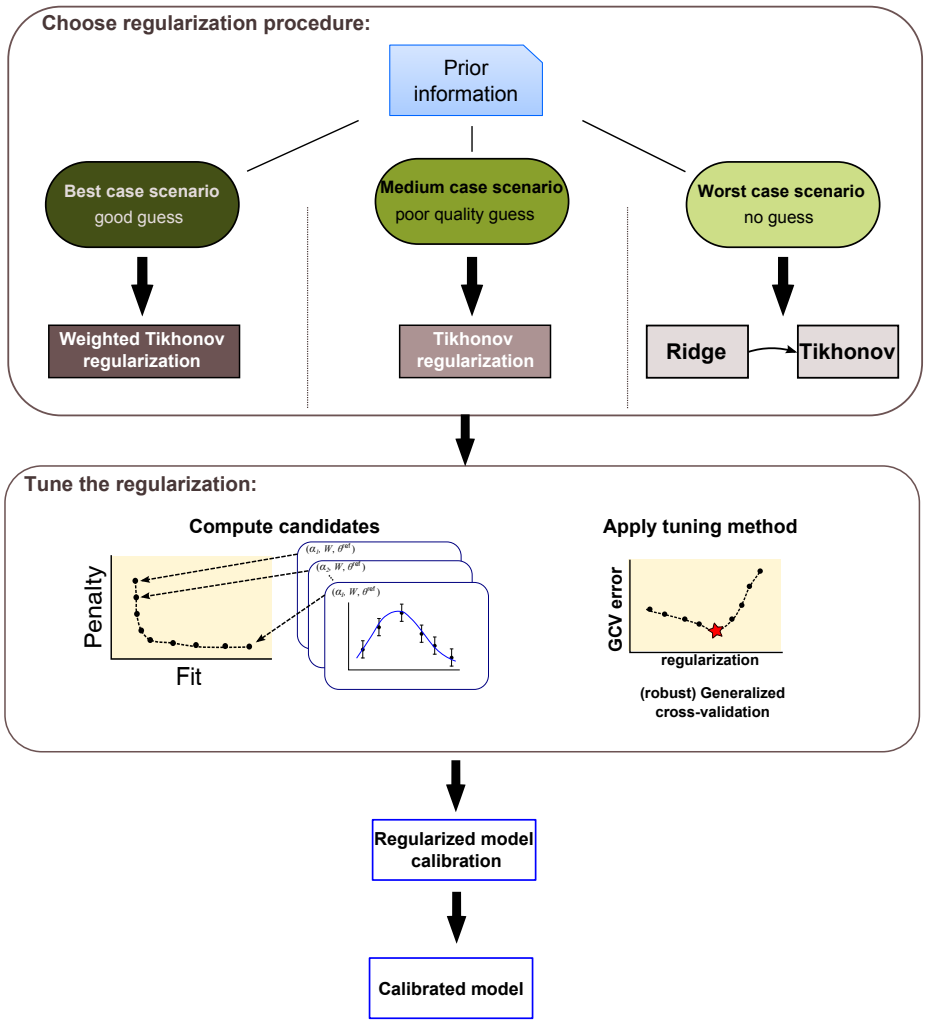


Figure D.1: Regularization scheme. Three scenarios are considered based on the quality of the available prior information: (i) best case scenario (a good guess of the parameter values is available in the literature) where a first order weighted Tikhonov regularization is recommended, (ii) medium case scenario (less reliable initial guess, but within one order of magnitude of the true values) where non-weighted Tikhonov regularization is recommended, and (iii) worst case scenario (no prior knowledge and therefore random guess of parameters) where a two-step regularization procedure is proposed. In the first step ridge regularization is applied which results the parameter vector with minimum norm, that fits the data reasonably well. In the second step this parameter vector is used as the reference parameter vector for Tikhonov regularization. In each scenario the regularized optimization is solved for a set of regularization parameter and the generalized cross validation method (GCV) is applied to choose the optimal candidate.

Optimization runs terminate when at least one stopping criteria is reached. The most frequently activated stopping criteria for the global optimization was either the allowed computation time or the allowed number of objective function evaluation. The allowed computation time and number of function evaluations can be founded for each case study in Table D.2.

Table D.1: Default optimization settings for the case studies. eSS settings: `ndiverse` is the number of diverse solutions generated, `n1` is the number of global iteration before the local algorithm is called for the first time, `n2` is the number of global iteration between consecutive calls of the local algorithm, `local_balance` influences the selection of starting point among the members of the population for initiating the local optimization, `log_var` generates the initial and new members of the population in the logarithmic scaled bounds of the parameters. NL2SOL settings: `maxfuneval` is the maximum number of function evaluation before the search terminates, `maxiter` is the maximum iteration number before termination, `tolrfun` is the relative tolerance (the algorithm terminates if the approximated global optima is within this tolerance value), `tolobjr` is the computational accuracy of the objective function and the Jacobian (which is tuned to the tolerance level of the ODE solver tolerance level).

eSS settings	**value**	**NL2SOL settings**	**value**
`ndiverse`	$10 \cdot N_\theta$	maxfuneval	300
`n1`	1	maxiter	200
`n2`	10	tolrfun	10^{-6}
`local_balance`	0.5	tolobjr	10^{-5}
`log_var`	ones$(1, N_\theta)$		

Table D.2: Case study specific stopping criteria for the optimization algorithm. CPU time is the maximum allowed computation time, NFuneval is the maximum number of objective function evaluation.

Criteria	**BBG**	**FHN**	**MAPK**	**GOsc**	**TGFB**	**TSMP**	**CHM**
`CPU Time`	60 s	120 s	60s	120 s	900 s	120 s	900 s
`NFuneval`	10 000	10 000	30 000	30 000	20 000	20 000	10 000

D.4 Regularization tuning methods

In order to find the optimal regularization parameter, α is discretized as $\alpha_1 > \alpha_2 > \cdots > \alpha_I$ and then the search for optimal regularization parameter is reduced to choose the best regularization parameter in this set (called the tuning of the regularization parameter). The optimization problem (5.7) has to be solved for each candidate, which results in the *regularization candidates*: $\hat{\theta}_{\alpha_1}$, $\hat{\theta}_{\alpha_2}$,... $\hat{\theta}_{\alpha_I}$. This is a computationally expensive task, although in an iterative framework the previously obtained solutions can be utilized to reduce the computational cost of the remaining candidates [164] .

Regularization tuning methods can be classified based on the type of information they require (error level in the data, limits for the regularization parameter, further tuning parameters) and in the way the optimal regularization parameter is selected among the candidates. In Table D.3 we shortly summarize the regularization tuning methods that we have considered and compared. Further details about each tuning method can be found in [164]. The methods considered can be classified into the following five groups:

- Discrepancy principle (DP) is based on the idea that the regularization parameter should be chosen such that the sum of residuals should be equal to the error level

of the data. For that, a good estimate of the measurement error is needed, which is often not known. Other versions of the discrepancy principle, such as the modified discrepancy principle (MDP) and the transformed discrepancy principle (TDP) are known to be less sensitive to the accuracy of the error level.

- Monotone error rule (MER) and quasi optimality criteria (QO): they use the observation that the differences between successive candidates, i.e. $||\hat{\theta}_{\alpha_i} - \hat{\theta}_{\alpha_{i+1}}||$, are large due to either large regularization or large propagated error and the difference becomes small for the optimal regularization parameter.

- Balancing (BP) and hardened balancing principle (HBP): they use all the candidates to estimate the regularization error, which is compared then to the so called approximated propagated error bound. The optimal regularization parameter is for which the two types of estimated error is minimal.

- L-curve method: proposed by Hansen et. al. [178] to display information about the candidates $\hat{\theta}_{\alpha_i}$, $i = 1 \dots I$. By plotting the two parts of the objective function (5.7): the model fit $Q_{\text{LS}}(\hat{\theta}_{\alpha_i})$ and the regularization penalty $\Gamma(\hat{\theta}_{\alpha_i})$ for $\{\alpha_1, \dots \alpha_I\}$ one obtains a discrete Pareto optimal front, which usually has an L-shape (see for example in Figure 5.5A). The horizontal part is formed by the solutions corresponding to large regularization parameters, where the regularization bias is dominating. As the regularization parameter decreases the least squares error reaches a limit that is determined by the measurement noise and the model flexibility. On the vertical part of the L-curve a small reduction in the least squares model fit error usually cause a large increase in the penalty. Intuitively, the optimal regularization parameter that balances the two types of error is located near the corner of the L-shaped curve. In [179] the corner point is defined as being the point that has the largest curvature on the L-curve (LCC).

- Generalized cross validation (GCV): an approach by Golub [180] that aims to find the regularization parameter that minimizes the leave one out (LOO) prediction error [141]. It does not require any estimate of the measurement error, but it can be sensitive if a small number of measurement data is at hand. For this reason, other variants, such as the robust (RGCV) and the strong robust generalized cross validation methods [156, 181] (SRGCV) have been developed.

Table D.3: Overview of the regularization tuning methods considered. We have indicated with a ✓sign for each method, (i) which data/information is required (residual vector, estimated kinetic model parameters or the Jacobian of the residual vector), and (ii) whether the regularization method utilizes further tuning parameters, an estimate of the measurement noise level or a limit for the maximal/minimal regularization parameter. Finally, the last three columns indicate if a computationally expensive procedure is involved, which can be an issue for large scale problems. SVD denotes singular value decomposition.

Regularization method			**Computation involves**			**Further required inputs**			**Involved computation**		
Method	**Short ID**	**Refs**	**Residuals**	**Estimated parameters**	**Jacobian**	**Tuning parameter**	**Meas. error estimate**	$\alpha_{max}/\alpha_{min}$	**Matrix inverse**	**SVD**	**Trace**
Discrepancy principle	DP	[182]	✓	-	-	✓	✓	-	-	-	-
Modified DP	MDP	[183]	✓	-	✓	✓	✓	-	✓	-	-
Transformed DP	TDP	[184]	✓	-	✓	✓	✓	-	✓	-	-
Monotone Error Rule	MER	[185]	✓	✓	✓	✓	✓	-	✓	-	-
Balancing Principle	BP	[186]	-	✓	✓	✓	✓	-	-	✓	-
Hardened Balancing	HBP	[187]	-	✓	✓	-	-	-	-	✓	-
Quasi optimality	QO	[129]	-	✓	-	-	-	✓	-	-	-
L–curve method (curvature)	LCC	[178]	✓	✓	-	-	-	✓	-	-	-
L–curve method (Reginska)	LCR	[188]	✓	✓	-	-	-	✓	-	-	-
Extrapolated Error Rule	EER	[189]	✓	-	✓	-	-	-	-	-	-
Residual Method	RM	[190]	✓	-	✓	-	-	✓	✓	-	✓
Generalized Cross-validation	GCV	[191]	✓	-	✓	-	-	-	✓	-	✓
GCV (Golub)	GCVG	[180]	✓	-	✓	-	-	-	✓	-	✓
Robust GCV	RGCV	[156]	✓	-	✓	✓	-	-	✓	-	✓
Strong RGCV	SRGCV	[181]	✓	-	✓	✓	-	-	✓	-	✓

D.5 Description of the case studies for parameter estimation

The description of all the case studies can be found on the CD supplement of the dissertation in the `Dissertation_CD_supplement.pdf` file, in Chapter 1.

D.6 Graphical results for all the case studies in parameter estimation

The graphical results of all the case studies can be found on the CD supplement of the dissertation in the `Dissertation_CD_supplement.pdf` file, in Chapter 2.

D.7 Distributions of prediction errors - statistical test

In Section 3.3 of the main text the distributions of prediction errors of non-regularized models were compared to the regularized models. Each regularization scenario (worst, medium and best case) was compared to the non-regularized case (Figure 10 in main text). In order to check that the observed differences in the medians are significant, we use here the Wilcoxon rank sum test (also known as Mann-Whitney U test). This non-parametric statistical test can be used in situations where the normal distribution of the observed data points cannot be assumed. With this test we check the null-hypothesis (H_0) that regularization does not effect the prediction error, i.e. the prediction errors of the models calibrated without regularization are distributed in the same way as those calibrated with regularization. The alternative hypothesis is that the prediction errors are not distributed in the same way. In this case based on the rank sum values we can tell which median is significantly smaller than the other (higher rank sum means higher median) and therefore we can decide if regularization significantly improved the predictions. The results are summarized in Table D.4, where we present the p-values corresponding the null-hypothesis and the result of the hypothesis tests for each regularization scenario and case study. When the null-hypothesis is rejected at the 0.05 significance level (indicated by a 1 in the H_0 column) we can conclude that the effect of regularization is significant on the model prediction performance.

The test results are mostly in agreement with the results obtained by visually comparing the medians in the box-plots in Figure 10 of the main text. However, they also allow us to see that, in the FHN case study, the observed differences in medians are statistically not significant. Further, in the case of the TGFB case study, the test shows that the best case regularization scenario significantly improves the performance even though the difference in the medians is small.

Table D.4: Comparison of non-regularized and regularized model prediction errors by the Wilcoxon rank sum test. According to the null-hypothesis (H_0) the distribution of the prediction error with regularization is similar to the distribution of the prediction error without the regularization. The alternative hypothesis is that the regularization significantly effects the prediction error. The table shows the P-value corresponding to the hypothesis test, the result of the test on 0.05 level of significance (H_0 column) and the rank sum values for the non-regularized and regularized prediction errors respectively (larger value means worst prediction error in general). In case the H_0 is rejected (indicated by 1) and the rank sum corresponding to the regularization case (the second number in the pairs) is smaller, the regularization significantly improved the prediction.

Case study	Regularization scenario								
	Worst case			Medium case			Best case		
	P-value	H_0	Rank sum	P-value	H_0	Rank sum	P-value	H_0	Rank sum
BBG	0.6	0	(9836,10264)	0.18	0	(10601, 9499)	0.0035	1	(11245, 8855)
FHN	0.95	0	(10026,10074)	0.12	0	(10680, 9420)	0.15	0	(10640, 9460)
MAPK	0.72	0	(9903,10197)	$5.5 \cdot 10^{-8}$	1	(12274, 7826)	$2.7 \cdot 10^{-10}$	1	(12634, 7466)
GOsc	$1.6 \cdot 10^{-6}$	1	(12014, 8086)	$1.6 \cdot 10^{-8}$	1	(12365, 7735)	$2.4 \cdot 10^{-7}$	1	(12165, 7935)
TGFB	0.66	0	(10232, 9868)	0.15	0	(10644, 9456)	0.043	1	(10879, 9221)
TSMP	$1.3 \cdot 10^{-16}$	1	(13437, 6663)	$6.1 \cdot 10^{-7}$	1	(12092, 8008)	$4.00 \cdot 10^{-7}$	1	(12126, 7974)
CHM	0.0021	1	(8791,11309)	$1.1 \cdot 10^{-27}$	1	(14512, 5588)	$3.8 \cdot 10^{-21}$	1	(13913, 6187)

D.8 Evaluation of tuning methods for regularization

The evaluation of tuning methods for regularization for all the case studies can be found on the CD supplement of the dissertation in the `Dissertation_CD_supplement.pdf` file, in Chapter 3.