Applied Analysis and Optimization
Volume 2, Number 3, 2018, 441–466

Yokohama Publishers
ISSN 2189-1664 Online Journal

AN INVERSE PROBLEM FRAMEWORK FOR DYNAMIC METABOLIC RESOURCE ALLOCATION PROBLEMS IN SYSTEMS BIOLOGY

MARKUS ARTHUR KÖBIS

Abstract. To fully utilize the potential of microorganisms, e. g., for the production of bio-fuels or advanced medicine, one needs to understand and simulate the metabolism of these cells in dynamically changing environments. However, mathematical models describing even simple unicellular metabolic systems usually lack sufficient information and data certainty to automatically build dynamic simulation models. We propose a description of a multitude of static and dynamic problems in systems biology within the framework of inverse problems. In particular, for the context of metabolic resource allocation problems, the concepts of dynamic enzyme-cost flux-balance-analysis (deFBA) is characterized as a form of regularization strategy. Generalizing from this point, a multitude of biologically inspired optimization principles onto the available model data can be used in the sense of inverse-problem regularization and lead to new algorithms for problems in systems biology as well as a more standardized way of communicating existing computational frameworks. A rather small benchmark model, analyzed on several stages of the model building process, will underline our approach.

1. Introduction

The simulation of metabolic networks using dynamic models nowadays embodies an important backbone for making predictions in medicine and biochemical process technology. A truly efficient computerized simulation of cell populations based on genome-wide models currently and, despite novel technical possibilities in time-resolved measurements and enhanced high-throughput technology, in the foreseeable future, faces several obstacles, (cf. [38]):

(A) Uncertainty of the given data: Neither the parameters entering the mathematical models nor in some cases even the phenomenological features those models are based on are fully quantified. This is due to multiple factors: Firstly, the measurements in the lab can never give a fully resolved picture about the inner life of the cells but have to concentrate on carefully selected samples within the set of metabolic compounds in the chemical solution. Secondly, the technical realization of these measurements is based on indirect methods that involve intermediate steps which additionally blur the results and introduce time delays that cannot be fully compensated. Thirdly, there is a natural variability in the data stemming from the biological systems

2010 *Mathematics Subject Classification.* 49N45, 65J22, 92C42.
Key words and phrases. Inverse problems, systems biology, optimal control, computer simulation.
This research was carried out in the framework of Matheon supported by the Einstein Foundation Berlin (ECMath).

being so complex that even fine-grained models cannot represent all aspects of them. Fourthly (but certainly not lastly), with a handful of exceptions, publicly available network models of metabolic systems are never complete in the sense that the entire genome information of the organisms enters the model. Even more, from a practical point of view, the metabolisms should never be viewed in isolation but should instead be seen as a part of their (animate and inanimate) surrounding environment which introduces even more uncertainty in the modeling process.

(B) Within the field of systems biology, there is a vast abundance of different techniques and modeling frameworks: Until this day, there is not *the one* agreed-upon standard on how to (i) choose and/or obtain the experimental data necessary to make predictions about the time-resolved evolution of metabolic systems, (ii) exactly define all involved physical quantities and (iii) there is no common understanding on the algorithmic treatment of certain predictive tasks. This stretches beyond just the choice of certain mathematical routines or software libraries but sometimes obscures the actual mathematical problem that is to be solved. This lack of standardization hampers a fair comparison of the computational results acquired by different techniques. Certainly, the computational framework to be used should always be based on the question what you want to know about the system, the available data respectively and there exist fixed standards concerning the statistical tests, the experimental setups, the accuracy of equipment and the reproducibility of the results. Nevertheless, a more mathematically solid definition of the underlying principles could improve practical applicability of theoretical and experimental findings. For example, the "Systems biology markup language" (SBML, [24]) for data exchange of metabolic network models had an immense impact on the research and collaboration activities within the systems biology community by improving interoperability and simplifying validation and verification of theoretical concepts along multiple computational frameworks. Similarly, protocol solutions for certain aspects of the modeling process (e. g. [54] for the generation of genome-scale network models from gene-sequencing) have largely improved the general acceptance and interchange of those models.

(C) The complexity of the biochemical processes in all lifeforms directly transfers to the computer simulation. As a result, genome-scale dynamic modeling of metabolic networks is usually only successfully done for a few applications and computational setups that usually are not fully dynamic. In lack of fully automatized/automatizable workflows, the involved models need to be hand-curated which is time-consuming and prone to mistakes making it almost impossible to give a holistic picture of the metabolic systems.

It has long been noticed [9, 11, 59] in the field of systems biology that the apt mathematical description is that of an inverse problem, [28, 55]. However, the mathematical framework is usually concerned with only selected aspects of systems biology such as network inference or parameter estimation (see below) somewhat disregarding the benefit that comes with the inverse problem formulation: That is that exactly the same theoretical techniques and software may be applied in order to

tackle the biological questions arising on different stages of the modeling process. On the other hand, in the existing literature, also the algorithmic treatment usually focuses on established mathematical tools which thwarts the big advantage that comes with working in mathematical biology which is that billions of years of natural evolution have created sophisticated self-controlled systems allowing for a multitude of optimization principles.

In recent years, several optimization-based concepts have been developed for the simulation of metabolic network models in systems biology. Among the most well-known are *flux-balance analysis* (FBA, see [42] for a review and below) where the metabolic network is reduced to a linearly constrained system optimizing biomass production of the cell; mathematically expressed in terms of a linear program (LP). FBA has had various successful applications in biotechnology and medicine, [6, 63]. FBA, however, is limited to a narrow field of applications because of its strict mathematical arrangement and hard biological simplifications and limitation to stationary problems. Because of this, researchers have created various extensions of FBA, such as (a) iterative FBA, (iFBA, [58]) which is a sequential application of FBA for (quasi-) dynamic problems (potentially with additional logical rules), (b) dynamic FBA (dFBA, [35]) where a control task or an optimal control problem is formulated based on the objectives of FBA, (c) optimal knockout analysis and flux-coupling analysis (FCA, [6]) which combines FBA with combinatorial information resulting in mixed-integer-linear programs (MILP), (d) resource balance analysis (RBA, [15]) where growth and dilution effects as well as macro-molecular assembly are taken into account, (e) conditional FBA (cFBA, [46]), dynamic enzyme-cost FBA (deFBA, [62]) and so-called models of Metabolism and macromolecular Expression (ME-models, [33]) that combine the macro-molecule production and optimal control methods in one framework.

On a genome-scale, all of these frameworks mostly stay in the linear regime (i. e. LP, MILP, potentially with quadratic constraints) for reasons of computational complexity and the availability of highly-efficient LP-solvers like CPLEX[1] or Gurobi[2]. Klipp et al. [31] proposed a general nonlinear optimal-control form which was later on generalized (e. g. [2]) and embedded into a control-theoretic frame, [43].

The cardinal justification for these optimization-based approaches is Darwin's theory of natural selection: A species that is not equipped with the (genome-encoded) ability to survive certain external conditions and surpass its competitors will eventually get extinct and not spread its inferior genetic material any further.

However, there is an ongoing dispute among researchers, whether and to what extend a solely optimization-driven framework can provide a sufficiently good picture since it is very hard to exactly predict and almost impossible to prove the 'design goals' of a metabolic system. Most of the time, multiple such design goals can be formulated, mathematically possibly providing a multitude of different solutions without any information which one is the most realistic. On the other hand, frameworks that do not rely on evolutionarily inspired optimization principles like

[1] www-01.ibm.com/software/commerce/optimization/cplex-optimizer

[2] www.gurobi.com

plain data fitting techniques oftentimes employ regularization methods that completely disregard the fact, that the objects one is dealing with are actually subject to evolutionary pressure; this way, giving up much of the biological knowledge that could be used.

The goals of this concept paper are therefore the following: By giving a very general description of a multitude of applied problems in systems biology—in terms of an inverse problem formulation—we want to provide a mathematical form and nomenclature that give practitioners some guideline for the description of the practical problems leading to a simpler communication of theoretical frameworks.

The framework will generically allow for an easier incorporation of uncertainties and robustness aspects into the computational models and construct new algorithms for those problems guided by biologically inspired optimization principles. This way, it provides a standardization of many existing frameworks which will in turn also simplify the (numerical) analysis of the actual algorithms. Systems biology problems are intricate and, as indicated above, a large variety of tools are available. Unifying some of these frameworks may also allow for a simpler interchange of models and algorithms.

Another advantage of building upon a inverse problem formulation is that it naturally encompasses the aspect of uncertainty quantification as well. In its very general form, inverse problems modeling allows for an easy incorporation of random perturbations. Within this document, we will disregard stochasticity and completely follow a set-valued analysis approach, see Section 2 below. Other approaches for including the uncertainty aspects in systems biology modeling and simulation include (biological) robustness, see [29]. Here, the focus lies on the construction of 'topological structures' that are insensitive to perturbations, or allow cardinal functions of the cell to keep intact under the influence of perturbations without a formal consideration of the input-output structure of the mathematical model. Closely related is the abstraction to logical modeling, see [7] for a mathematical introduction, where the dynamics are condensed to a purely phenomenological setting. Further approaches include the construction of robustness-measures based on sensitivities (see [60] for an example) and constraint-based frameworks that aim at a full enumeration of a multitude of possible solutions like flux-variability analysis, (FVA, [36]).

Of course, there is also the *top-down approach* in which biophysical intuition is used to infer metabolic behavior instead of the *bottom-up approach* of deducing underlying principles from detailed (but ideally semi-automatically generated) models that are beyond a full mathematical analysis. We also point out that [5, Chapter 6] covers some critical remarks on how *not* to use inverse problem theory when studying complex models in systems biology.

This article is structured as follows: In Subsection 1.1, we will give a brief overview on the basic goals of systems biology emphasizing on the mathematical modeling of metabolic networks in terms of ordinary differential equations (ODEs) and/or differential-algebraic equations (DAEs). In Section 2 we will introduce an inverse problem framework and discuss some special implications regarding mostly their regularization which stem from the biological background. Section 3 will cover

some (small-scale) examples on how the framework is already part of active biology research and how it can be used to derive new concepts before the article is summarized in Section 4.

1.1. **A Quick Synopsis of ODE/DAE and Constraint-Based Modeling in Systems Biology.** Before the formulation in terms of an inverse problem, we will provide a very quick overview on the applications we aim at in this work and their mathematical formulation. For a comprehensive survey on dynamic modeling of metabolic systems, see [21, 32].

As already outlined, the complexity of biological systems and their entanglement with their environment makes it very hard to obtain unperturbed and reliable data. In systems biology, researchers distinguish between *in vivo* conditions (that is: biological systems in their natural habitat), *in vitro* conditions (meaning: clean and partly measurable laboratory-setups), and so-called *in silico* experiments (i. e. a completely computerized solution.), see Figure 1. The ambitious vision of re-

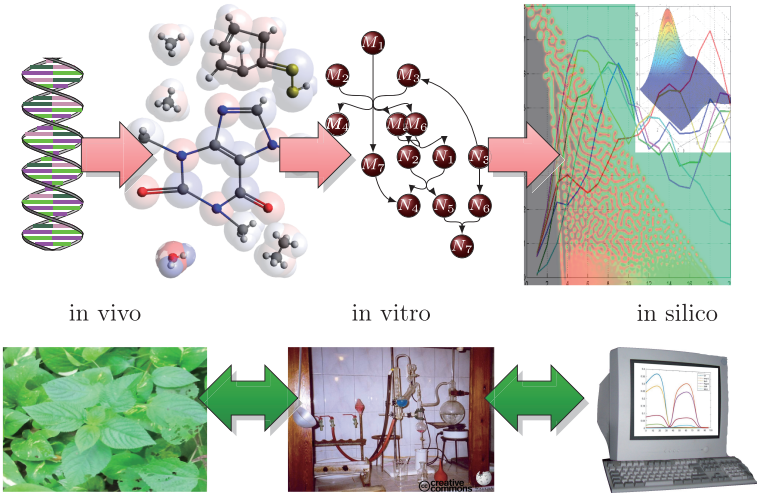

FIGURE 1. Idealistic process of systems biology knowledge acquisition, molecules designed with 'Avogadro': an open-source molecular builder and visualization tool. Version 1.20. http://avogadro.cc/, [19]

searchers in that field is that one day we might be able to fully predict the behavior of (at least simple micro-) organisms solely from genetic information. For metabolic modeling in systems biology, this means that: (i) from the genome it is possible to (ii) identify biochemical compounds that might be build within the cell and use physicochemical knowledge to (iii) construct a *metabolic network*, i. e. a hypergraph that completely describes the possible chemical interactions of these compunds, which finally allows for (iv) computer simulations.

The dynamic description of metabolic networks using an ODE or DAE framework is typically stated by means of the dynamic mass balance equations

$$\dot{\boldsymbol{y}}(t) = \frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{y}(t) = \mathbf{S} \cdot \boldsymbol{f}(t, \boldsymbol{y}(t)) \tag{1.1}$$

subject to according initial/boundary conditions.

Here, $\boldsymbol{y}\colon [t_0, t_{\text{end}}] \to \mathbb{R}^{n_{\boldsymbol{y}}}$, $t_0, t_{\text{end}} \in \mathbb{R}$, is the *state vector*, denoting the absolute amounts or relative concentrations of the involved biochemical compounds, possibly restricted to certain compartments of the cell and $\dot{(\cdot)} := \frac{\mathrm{d}}{\mathrm{d}t}(\cdot)$ denotes differentiation with respect to time t. Throughout, we will follow the convention that $n_{(\bullet)} \in \mathbb{N}$ is the dimension of a vector-valued entity $(\bullet)$. $\mathbf{S} \in \mathbb{R}^{n_{\boldsymbol{y}} \times n_{\boldsymbol{f}}}$ is the *stoichiometric matrix*, consisting of the mass ratios of the metabolites in the particular *chemical reactions* and encoding the connectivity of the network model. It is well-known [23] that the stoichiometric matrix can be expressed as the product

$$\mathbf{S} = \mathbf{K} \cdot \mathbf{Z}\,, \tag{1.2}$$

where $\mathbf{Z}$ is the incidence matrix of the network graph and $\mathbf{K}$ is the complex-stoichiometric matrix, holding all possible stoichiometric coefficients. Chemical reactions are combined in the *flux vector* $\boldsymbol{f}\colon [t_0, t_{\text{end}}] \times \mathbb{R}^{n_{\boldsymbol{y}}} \to \mathbb{R}^{n_{\boldsymbol{f}}}_{\geq 0}$. There are several biochemical justifications for this concept of ODE-modeling. Basically, it stems from an averaging idea: The actual number of molecules, their spatial distribution, and random interactions are abstracted to compound concentrations supported by the law of large numbers. If further heterogeneity of the solution is required for realistic modeling, this can be achieved by formal compartmentalization, i. e. distinguishing between chemically the same biochemical compound by using different labels. The assumption of the 'well-stirred metabolism' which is the basis of the application of the law of large numbers, is additionally founded on having almost constant temperature, since thermodynamic effects also have a heavy influence on the reaction rates.

The dynamical behavior of (1.1) is oftentimes multi-scale: Quick (e. g. allosteric) adaptations of the cells are accompanied by the rather slow building of cell plasma/membrane necessary for the growth of the cell. One way of coping with the multi-scale character is the use of the *(quasi-) steady-state assumption* ((Q)SSA). Here, it is assumed that some of the biochemical compounds adapt their concentrations infinitely fast, such that a description in terms of ODEs is no longer necessary but parts of (1.1) can instead be replaced by algebraic relations

$$\mathbf{0} = \mathbf{S}_{\mathcal{I}_{\text{int}},:} \cdot \boldsymbol{f}(t, \boldsymbol{y}(t))\,. \tag{1.3}$$

When including these algebraic constraints, the problem class shifts to a DAE [18]. Several practical studies, however, have shown limitations of the QSSA, see for example [12, 50]. A quite comprehensive (and critical) mathematical treatment of the quasi-steady-state assumption using, among others, methods from singular perturbation theory, is carried out in [48]. In a nutshell, Tikhonov's theorem gives a sufficient but not necessary condition for when this step is mathematically correct. Applying the QSSA, however, also has the additional advantage that just stating mass-balances without explicit dynamics often requires fewer parameters for the chemical reactions. Especially for quickly changing internal metabolites/smaller molecules, measuring chemical reaction rates is very complicated such that often one has no choice but to adhere to QSSA.

By the strict application of averaging techniques, it is theoretically possible to explicitly derive the flux through each edge of the metabolic network by means of

the concentrations of educts (the ingoing nodes), that is for a chemical reaction

$$\alpha_1 \cdot Y_1 + \alpha_2 \cdot Y_2 + \ldots + \alpha_{n_{\text{in}}} \cdot Y_{n_{\text{in}}} \longrightarrow X\,,$$

the according rate f_i can be expressed in terms of the concentrations

$$\{y_k(t)\}_{k=1}^{n_{\text{in}}}$$

of chemical compounds $\{Y_k\}_{k=1}^{n_{\text{in}}}$. Note that the product(s) X of the reaction are not relevant for non-reversible reactions.

In the 'standard' case of *mass-action* kinetics, this relation is given by

$$f_i \propto \prod_{k=1}^{n_{\text{in}}} y_k^{\alpha_k}\,,$$

where often the additional simplification $\alpha_k := 1$ is used. To capture more complicated effects (and that way—by hand—implementing a model reduction step), other nonlinear rate laws are also frequently used. Among the most popular are *Michaelis-Menten* kinetics and *Hill-functions*, where

$$f_i \propto \frac{y_k}{K_M + y_k} \qquad \text{or} \qquad f_i \propto \frac{y_k^{\beta_k}}{K_d^{\beta_k} + y_k^{\beta_k}}\,. \quad (K_M,\, K_d,\, \beta_k \in \mathbb{R})$$

For $y_k \to \infty$, these function level out (cf. Figure 2 for rate laws depending on only one educt y_k) such that a maximum level, an upper bound for this reaction rate can be implemented. For $\beta_k < 0$, Hill-functions are called *inhibitory* as for grow-

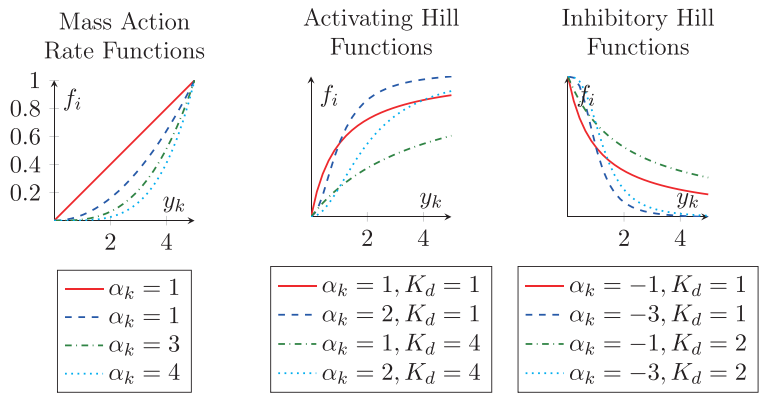

FIGURE 2. Flux profiles for different activation and inhibition rate laws

ing educt concentrations, the reaction rate gets lower. This implements chemical inhibitors, so chemical structures which prevent the reaction from actually taking place. Unfortunately for the researchers, even all this tools do not completely solve the problem of insufficient parametrization of the DAE models and techniques from control theory or exhaustive data-fitting are necessary to reason from these models at a larger scale.

An opposite approach for the construction of computational models for metabolic systems is *constraint-based modeling:* Here, solely, given constraints on the model in terms of

- stoichiometry,
- upper and lower bounds on the fluxes,
- steady-state of (most of) the metabolites, and
- thermodynamics, among others,

are formulated and one searches for all, some, or 'particularly important' solutions that remain or semi-automatic procedures to relax some of the constraints if no solution can be obtained at all. This may once again be seen as a model reduction step and a way to deal with the insufficient data situation. As constraint-based modeling can equivalently be formulated as the *feasibility approach* for the solution of inverse problems, we postpone a closer description to Section 2 below.

In Section 3 below, we will apply DAE- and constraint-based modeling techniques for a small artificial metabolic network.

2. Inverse problems: General formulation

2.1. **Setup of the Mathematical Framework.** Giving a complete mathematical framework in which any inverse problem from the literature can be cast is almost impossible. It is peculiar, almost ironic, that most introductory texts on inverse problems start by the definition of its inverse: The *direct problem*. Following [28], we will also base the representation here on the abstract formulation of the direct problem as an input-output relation of the form

$$x^{\text{meas}} := \mathcal{T}(p)\, x + \delta\,, \tag{2.1}$$

where x^{meas} is the resulting measurable (or detectible) system outcome and δ is some noise on the output. In practice, x^{meas} can be (dense or continuous) computerized data for metabolite concentrations, growth measurements of bacterial cultures or direct machine output from experiments or (discrete-valued or structural) information like genome-sequencing results, gene-regulation rules or statistical sampling data. For the purpose of a broad mathematical form, we will simply require x^{meas} and δ to be elements of a topological or Banach space $\mathbb{Y}$. The operator $\mathcal{T}\colon \mathbb{X} \to \mathbb{Y}$, possibly depending on certain parameter values $p \in \mathbb{R}^{n_p}$ provides a mathematical description of how the outcome x^{meas} can be computed once a full understanding of the system's state $x \in \mathbb{X}$ is at hand, where $\mathbb{X}$ also needs to be chosen as some topological space.

Of course, what one would like to obtain from the computational model is the system state x which should itself be part of the *set of feasible solutions* of the inverse problem $\mathbb{D} \subseteq \mathbb{X}$. So, straightforwardly, the task of the inverse problem is now the following:

(2.2) Find (some, all, 'the best') $x \in \mathbb{D}$ such that (2.1) is satisfied.

One particular aspect of this inversion process is that, formally, it is very hard to accomplish. According to the seminal work of Hadamard [17], the condition number of an inverse problem is very large or—more often still—the problem ill-conditioned or even ill-defined. In particular, this means that

(1) even smallest changes in the inputs can cause tremendous variations of the numerical results,

(2) it might be necessary to state additional, sometimes even artificial conditions in order to obtain a clear well-defined problem.

Numerical and analytical methods for coping with this ill-posedness of inverse problems, one way or the other, all boil down to the design and use of appropriate *regularization* techniques. In the following paragraph, we will shortly review the most common solution techniques for inverse problems and where the regularization enters in these cases.

2.2. **Common Solution Aspects.** Even though the field of inverse problems has been applied in various different disciplines and for even more applications, its somewhat vague definition makes it hard to give a comprehensive overview on the ingredients usually employed for the solution. Nevertheless, there are several recurrent characteristics:

(a) Guiding *optimization principles*, which in most cases means that the inverse problem is recast as an *approximation problem*. Typical questions that arise here are (i) the appropriate choice of distances to describe how far an approximation is off the given data and (ii) a compromise of the various types and elements of the available data.

Generically, this 'optimality' is understood in terms of

$$\min_{x\in\mathbb{D}} \|x^{\mathrm{meas}} - \mathcal{T}(p)x\|_*^{\gamma_\diamond}\,, \tag{2.3}$$

where $\|\cdot\|_*$ denotes a (semi-) norm on $\mathbb{Y}$ and $\gamma_\diamond \geq 0$ an appropriately chosen exponent to improve the numerical properties of the applied algorithms. More generally, approximation problems of this type might as well be stated in vector-valued fashion, see [16] for a detailed discussion of this problem class.

(b) Concept and design of *regularization techniques*. This is probably the most common part of the mathematical analysis and solution of ill-posed/inverse problems. By definition, regularization means that a system is changed in order to fulfill certain additional requirements or laws of reasoning. The most common solution aspect in practical use is the regularization by means of an additional term (the 'regularizer') $\mathcal{R}$ in the optimization principle, [55]

$$\min_{x\in\mathbb{D}} \|x^{\mathrm{meas}} - \mathcal{T}(p)x\|_*^{\gamma_\diamond} + \lambda\cdot\mathcal{R}(x;p,\mathbb{D},\mathcal{T})\,. \tag{2.4}$$

In its most general form, the regularizer may, apart from the solution vector x, depend on the parameters p but also on the feasible set $\mathbb{D}$ (e. g. when attempting to choose x very deeply inside the feasible region for robustness of the solution) and the evolution operator $\mathcal{T}$ which may provide insight into expectable smoothness of the solution. Mathematically, regularization techniques often convexify the optimization problem such that global optimization strategies can be used. In the case of *Tikhonov-regularization*, for example, one is trying to find a minimal norm approximation, i. e.

$$\mathcal{R} := \|x\|^2\,,$$

where $\|\cdot\|$ is a 'natural' norm in $\mathbb{Y}$. Other forms of this regularization include maximum-entropy and bounded-variation [1] if this norm includes

information about the weak derivatives of the solution x, for example norms in Sobolev spaces. In dynamical system simulation like ODE/DAE modeling in systems biology, dilution or damping terms are a simplified one-dimensional version of this. Tikhonov regularization is closely related to projection methods, the Moore-Penrose pseudoinverse for a solution of underdetermined linear systems is the limit case of $\lambda \to 0$ if the problem is stated in the approximation form. When studying analytical properties of the regularized solutions, the role of the *regularization parameter* $\lambda \geq 0$ is usually of central importance. The so-called *L-curve* is an indicator for a good choice of the regularization parameter, but the computational effort for its computation is often too restrictive, [28].

Of course, regularization does not only concern the objective in the optimization or approximation problem that recasts (2.2). Sometimes, also a *regularization of constraints* (constraint relaxation) is necessary, where the feasible region is extended $\mathbb{D} \to \tilde{\mathbb{D}}$. In practice, $\mathbb{D}$ can often be expressed by a set of inequalities $\mathbf{G}(x) \leq \mathbf{0}$ which can be relaxed to $\mathbf{G}(x) \leq \boldsymbol{\epsilon}$ with a vector $\boldsymbol{\epsilon} \in \mathbb{R}^{n_\mathbf{G}}_{\geq 0}$. Another possibility is to just approximately enforce the constraints using penalty techniques. Also, a restriction of the space $\mathbb{X}$, that the solutions live in has a regularizing effect. For example, requiring more smoothness or restricting x to a finite-dimensional ansatz-space may already lower the condition number of the inverse problem considerably.

Lastly, also *preprocessing*, smoothening or a prior statistical analysis (e. g. removal of outliers) in the input data x^{meas} is to be considered as regularization and sometimes an *iterative refinement of the regularization* is required as well.

In biology, systems are usually much more robust to external changes than randomly generated systems, oftentimes it is still unclear why and the field of *biological robustness* [29, 52] is vast and a recurring motif in the systems biology community. Inspired by this, in Section 2.3 below, we will argue that for biological systems, the restriction to solely mathematically motivated regularization terms falls rather short and outline some ways to include biological knowledge into the process.

(c) The construction of a *universal algorithm*. Roughly speaking, this means that the input-output structure of (2.1) is directly used to 'feed' an algorithm that 'learns' the common features of the system and that can later on be inverted to predict results for unknown inputs.

Within this category of computational tools, there has been immense progress in recent years that are typically entitled under keywords like 'artificial intelligence', '(deep) neural networks' and 'supervised learning'. Also, subspace approximations and support vector machines (SVMs) [51] are some form of universal algorithms.

But, more generally, also some plain data fitting techniques and, to a certain extend, system identification issues fall under this class and also some varieties of Monte-Carlo methods can fit into this category.

The bottlenecks of these methods are that (i) there cannot be given any guarantee that the procedures actually reproduces the systems behavior as

most of these algorithms work as black boxes and often do not use explicit knowledge, e. g. from underlying first principles, (ii) they show unpredictable behavior and sometimes do not work at all outside of the range where they were trained, and (iii) there is almost no deeper understanding or further analysis possible once the results are obtained. Concerning the last two difficulties, several recent techniques for the inversion of the input-output structure of (deep) neural networks [37, 39] have been introduced which might in the near future amend these drawbacks.

To remedy obstacle (i), the current state-of-the-art in the training phase of universal algorithms includes the use of test sets and validation sets (cross-correlation, i. e., checking against priorily unknown data) that do not explicitly feed into the parameters of the resulting algorithms but instead are used to value whether a sufficient training quality is reached. However, this reveals the most important point and demanding part of this class of solvers: It is necessary to have *sufficient data present* or, even better, techniques such that this can be generated. Unfortunately, scientific data in metabolic system analysis is often sparse and hard and expensive to acquire. In case that the data is present to a sufficient amount, the issue of *overfitting* comes into play. Overfitting describes the tendency of automatically trained systems to approximate the noise of the data rather than underlying model itself. To avoid overfitting, once again the need for apt regularization terms comes into play.

(d) *feasibility problem* formulation: The task here is to just find any (approximate) solution that fulfills all (possibly relaxed) constraints and a relaxed input-output structure (2.1) by a (reproducible and robust) procedure. Put in other words: The problem is abstracted to the set inclusion

$$\text{Find } x \in \tilde{\mathbb{D}} \text{ s.t. } x^{\text{meas}} \in \mathcal{T}x + \Delta\,, \tag{2.5}$$

where $\Delta \subseteq \mathbb{Y}$ is chose large enough to include the expected noise level. Common computational examples include feasibility problem solvers based on sequential projection methods, see [3] for a classical review on this approach. The feasibility approach in this formulation reveals the strong connection to set-optimization [26] and robustness [4]. For systems biology applications, the feasibility approach might be viewed as the most natural formulation since constraint-based modeling naturally leads to the same problem mathematical class, that is, a satisfiability problem. As indicated in (2.5), constraint relaxation is oftentimes of particular importance.

Regularization for this solution concept means that an additional objective is defined that is to be minimized over the relaxed feasible region $\tilde{\mathbb{D}}$.

$$\mathcal{R}(x; p, \mathbb{D}, \mathcal{T}) \to \min_{x \in \tilde{\mathbb{D}}}$$

The great generality of the inverse problem formulation also has its drawbacks: Apart from the model-related question which constraints should be classified as hard and which as soft constraints, it is important to notice that it is also not unique what to choose as parameter and what as input/measurable data or as part of the evolution operator $\mathcal{T}$.

2.3. **Regularization by Biology-inspired Optimization Principles.** For systems biology and bio-engineering applications, the theory of the evolution of species by environmental selection is a key for obtaining plausible optimization principles that serve as the underlying model in (2.2) or as regularization principles: Believable and superior regularization terms $\mathcal{R}(x)$ can be based on the expectation of the researcher that (on top of all the prior model assumptions and constraints) an additional minimization of a certain goal brings an evolutionary advantage of the cell culture or organism. This line of reasoning has the advantage that additional scientific expertise can be brought into the solution procedure on several levels of the modeling process but it has the drawback that on the side of mathematical analysis one might not be in the position to rely on positive mathematical properties such as strict convexity.

Typical design goals based on such evolutionary considerations can be found (by far not exhaustive) in [8, 20, 30, 34] and the references therein. We will shortly give an overview on the most common ones and review some of the examples in Section 3 below.

- *Cell efficiency:* A possible regularization term might implement a minimization a the fluxes required by the cell, that is for the dynamic model description from (1.1).

$$\mathcal{R}(x) = \|\boldsymbol{f}(\cdot, \boldsymbol{y}(\cdot))\|_\bullet^\gamma \,. \tag{2.6}$$

  Here, $\|\cdot\|_\bullet$ denotes a semi-norm and $\gamma > 0$ an exponent, that might be introduced for the same reasons as already pointed out for the approximation problem form of inverse problems (2.3). The possible evolutionary benefit from a minimization of $\mathcal{R}$ is obvious: Maintaining biochemical reactions in the cell requires energy and valuable precursors and might expose the cell to a predator. Obviously, it might be necessary to include constraints that a certain minimum of metabolic activity remains or other criteria are met, like no drop of vital compounds.

  Note that in (2.6) there is no explicit input parameter x in the right-hand side. The reason is that it depends on the choice of inverse problem formulation whether the fluxes $\boldsymbol{f}$, the states $\boldsymbol{y}$ or some other quantity is expressed by x. Note also that, if the fluxes $\boldsymbol{f}$ (without dependency on $\boldsymbol{y}$) are expressed by x, this form of cell efficiency mathematically implements a Tikhonov-type regularization.
- *Growth (a):* The probably most often used regularization term in systems biology is the maximization of biomass production, or the macro molecule assembly of the cell, respectively

$$\mathcal{R}(x) = -\|\boldsymbol{y}_{\mathcal{I}_{\text{important}}}(\cdot)\|_\bullet^\gamma \,,$$

  where $\mathcal{I}_{\text{important}}$ is an index set, collecting 'important' entries of the metabolite vector $\boldsymbol{y}$. The semi-norm $\|\cdot\|_\bullet$ might be L^2-like, including the entire time course or just measure the absolute value at the final time point. For bacteria in very good external growth conditions, there is high evidence that this is in fact (at least one of) the most important objectives [42] if one alleges that such a design goal exists at all. The evolutionary advantage is clear:

The fastest growing culture will take over the habitat by its sheer size at a certain point in time.

- *Growth (b)* Goal: Maximize flux through biomass reaction/macro molecule assembly fluxes. In this case, the regularization term could be given as
$$\mathcal{R}(x) = -\|\boldsymbol{f}_{\mathcal{I}^*_{\text{important}}}(\cdot, \boldsymbol{y}(\cdot))\|^\gamma_\bullet\,,$$
where $\mathcal{I}^*_{\text{important}}$ collects relevant fluxes from the entire flux vector $\boldsymbol{f}$. The biological reasoning follows the lines of the biomass growth.
- *Robustness (a):* On the other side of the spectrum of possible design objectives, a cell is also interested in maximizing its survival time. If the time interval for the above simulation task is not fixed, this can formally be expressed as
$$\mathcal{R}(x) = -t_{\text{end}}\,,$$
subject to minimal survival requirements (see the 'cell efficiency' goal). In some scenarios, this is indeed contrary to the aforementioned goal of e. g. maximizing the growth: If the culture is required to feed on a limited amount of available nutrients (or in the wild: is exposed to recurring droughts) or breathable air, it is not advantageous to grow as fast as possible but better to pace themselves in terms of nutrient consumption.
- *Robustness (b):* In a similar manner, cells benefit if they minimize their response times in case of sudden changes in their environment. This ranges from a quick switch of metabolic pathways when certain nutrients are depleted to the minimization of toxic intermediates and/or waste products.
- *Robustness (c):* In yet another understanding of robustness, some cells maximize their nutrient uptake to outperform competitors for the same energy source. Mathematically, this can again be expressed as
$$\mathcal{R}(x) = -\|\boldsymbol{f}_{\mathcal{I}_{\text{uptake}}}(\cdot, \boldsymbol{y}(\cdot))\|^\gamma_\bullet\,.$$
As mentioned above, in systems biology, the term 'robustness' has a very broad and somewhat blurred meaning, see [29].

As already said, this list only gives a glimpse on what additional biological intuition might provide to improve the inverse problem approach to metabolism simulation and identification problems in biological applications. Of course, also compromises between several of these design goals might be worth investigating. This leads to multi-objective optimization (see [53] for a discussion in the context of systems biology) or even set-valued formulations [26] but is beyond the scope of this paper.

3. Examples

3.1. **Description of the Benchmark.** We consider the artificial metabolic network depicted in Figure 3. In this setup, there are two nutrients N_1 and N_2 that the organism can feed on through the biochemical reactions f_1 and f_2 to first produce the internal metabolites M_1 and M_2 which can later be converted into the biomass precursor M_3 that the cell then uses to build up macromolecules. At this stage, the system needs to 'decide' whether the precursor M_3 should directly be used to create the biomass P itself or whether it should be invested to further improve the uptake reactions via the production of enzymes E_1 and E_2 both of which are

slowly degraded over time by the reactions f_8 and f_9. For simplicity, we assume all reactions to be non-invertible.

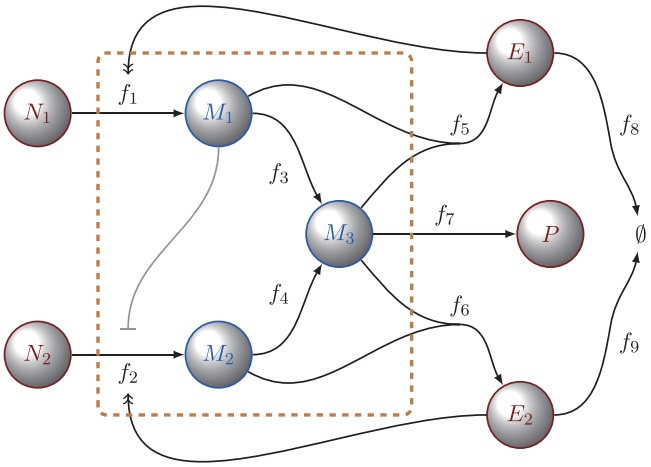

FIGURE 3. Schematic illustration of the benchmark, external metabolites in red, internal metabolites (fast components) in blue, enzymatic inhibition is indicated by a bar arrow, activation by a double arrow head. The inhibitory effect of M_1 is assumed to be unknown priorily.

In experimental biology, the effect that can be observed in this situation is the so-called *diauxic shift:* The population first 'concentrates' on just one of the nutrients until it is completely depleted. Not until this happens, the metabolic pathways for processing the 'second best' nutrient is activated. In the model, the fact that the nutrient N_1 is 'better' than N_2 is reflected by the effect of the enzyme concentration on the uptake reactions: Enzyme E_1 is about 100 times more effective in (see the 'enzyme-capacity constraints' below) catalyzing the uptake reaction $N_1 \to M_1$ than E_2 for the reaction $N_2 \to M_2$.

We distinguish eight metabolic compounds the last five of which are also subjected to given initial concentrations $\boldsymbol{y}(t_0 = 0)$. We wish to simulate the behavior of the metabolic network model over the course of two days (2x 24 hours).

$$\begin{aligned}\boldsymbol{y}(t) &= \big(y_{M_1}(t)\ y_{M_2}(t)\ y_{M_3}(t)\ y_{E_1}(t)\ y_{E_2}(t)\\ &\qquad\qquad y_P(t)\ y_{N_1}(t)\ y_{N_2}(t)\big)^\top \in \mathbb{R}^8\,,\\ \boldsymbol{y}(t_0 = 0) &= \Big(y_0^{(1)}\quad y_0^{(2)}\quad y_0^{(3)}\quad 0.05\quad 0.05\quad 0.01\quad 10\quad 10\Big)^\top\,,\\ &\text{and } t \in [0, 48]\,,\end{aligned}$$

where $y_0^{(i)}$, $i = 1, 2, 3$, need to be defined through the algebraic constraints (1.3). The first three metabolites are quickly adapting internal ones, such that in the notation of (1.3) we have:

$$\mathcal{I}_{\text{int}} = \{1, 2, 3\} = \{M_1, M_2, M_3\}\,,$$

$$\mathcal{I}_{\text{macro}} = \{4, 5, 6, 7, 8\} = \{E_1, E_2, P, N_1, N_2\}\,.$$

In the sense of constraint-based modeling techniques, the following assumptions on the model are made: The stoichiometry, i. e. the mass ratio of the metabolites in the respective reactions, is given by the following stoichiometric matrix

$$\mathbf{S} = \begin{pmatrix} 1 & 0 & -1 & 0 & -0.3 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & -0.3 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & -0.1 & -0.1 & -3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

Upper and lower bounds of the reaction rates are given by the following vectors

$$\boldsymbol{lb} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}^\top \in \mathbb{R}^9\,,$$
$$\boldsymbol{ub} = \begin{pmatrix} \infty & \infty & 1 & 1 & \frac{1}{100} & 1 & 1 & \infty & \infty \end{pmatrix}^\top \in \mathbb{R}^9\,,$$

that is for all $t \in [t_0, t_{\text{end}}]$ it holds $\boldsymbol{lb} \leq \boldsymbol{f} \leq \boldsymbol{ub}$. To include the enzymatic effects of E_1 and E_2 on the uptake reactions, we furthermore introduce the *enzyme-capacity constraints*

$$f_1(t) \leq 1000 \cdot y_{E_1}(t)\,,$$
$$f_2(t) \leq \tfrac{1}{10} \cdot y_{E_2}(t)\,.$$

The degradation reactions are dependent on the amount of available enzymes by the linear inequality relations

$$\frac{1}{20} y_{E_1}(t) \leq f_8(t) \leq \frac{1}{10} y_{E_1}(t)\,,$$
$$\frac{1}{20} y_{E_2}(t) \leq f_9(t) \leq \frac{1}{10} y_{E_2}(t)\,.$$

In the following paragraphs, we will classify existing frameworks that come across during the modeling process of metabolic networks like this one. Note again, that the descriptions we propose are not unique.

3.2. **Network Inference.** As outlined before, building up the metabolic network model from mass and or atomic stoichiometric relations of the involved metabolites can theoretically be done in a relatively straightforward manner. However, understanding activation and inhibition of reactions is not possible by simple arithmetic, unless the involved enzyme complexes are all known and explicitly included in the model which introduces more unknown constants and increases the problem size. Ways to infer interaction structures in dynamic network models are reviewed in [56]. For the sake of brevity, and since the model is artificial to start with, we will not fully work out a computational example here but note that network inference can be understood in terms of the inverse problem framework from Section 2.

Observation 3.1. The problem of network inference is an inverse problem of type (2.2) if one considers the following assignments:

- $x = \mathbf{Z}$ (from (1.2)) or $x = \mathbf{S}$, $\mathbb{X} = \mathbb{R}^{n_{\boldsymbol{y}} \times n_{\boldsymbol{f}}}$,
- $\mathcal{T} = \boldsymbol{\varphi}(\cdot; \mathbf{S})$ (as the flux of the ODE/DAE system)
- x^{meas} ... time course data of all metabolic compounds. $\mathbb{Y} = (\mathbb{R}^{n_{\boldsymbol{y}}}_{\geq 0})^N$, where N denotes the number of data points.

Typical computational methods in this field use the approximation problem formulation if sufficient data can be measured and regularization is done by sparsity enforcing p-norms with $0 < p < 1$, i. e. $0 \leq \gamma_\diamond \leq 1$ in (2.3). Strictly speaking, the feasible region $\mathbb{X}$ should only include integer-valued entities for the integer-valued ratios of atoms in the occurring molecules but (i) this is not relevant for many macro-molecules, (ii) the condition of the equations is often improved when the actual value is scaled up or down to fractional values and (iii) this (unnecessarily) increases the difficulty of the mathematical problem.

3.3. **deFBA – A Framework for Resource Allocation Problems in Systems Biology.** Based on the control framework 'dynamic FBA' [35], as a dynamic extension of the classical flux-balance analysis (FBA, see below) and 'resource balance analysis' (RBA, [15]) as a way to include the cost for macro-molecular assembly in metabolic network models, Rügen et al. [46] and Waldherr et al. [62], respectively, introduced the frameworks *conditional* FBA (cFBA) and *dynamic enzyme-cost FBA* (deFBA).

Both are optimal control frameworks, [14], that use a strongly condensed, linear model for the metabolic networks but therefore work even on the genome-scale [45].

In cFBA/deFBA, the consideration of macro-molecular assembly costs is handled by the introduction of additional ODEs, chemical reactions, respectively, describing the time evolution of the concentrations of these macro-molecules and different kinds of linear constraints on the concentrations of different metabolites and their catalysts at all times. The control variables (in the sense of classical optimal control) are the fluxes in the right-hand-sides of these dynamic equations and, from the biological side, upper bounds on their values need to be determined first from the literature. Note, that the parameter dependence of solutions of optimal control problems is an area of active research, see [13, 27] for recent results.

In its dynamic (i. e. optimal control) form, the framework can be cast as

$$
\begin{aligned}
\min_{\boldsymbol{y}(\cdot), \boldsymbol{f}(\cdot)} \int_{t_0}^{t_{\text{end}}} & (\boldsymbol{b}(t))^\top \cdot \boldsymbol{y}(t) \mathrm{d}t \\
\mathbf{0} &= \mathbf{S}_{\mathcal{I}_{\text{ind}},:} \cdot \boldsymbol{f}(t) \ \text{ for a. a. } t, \\
\dot{y}(t) &= \mathbf{S}_{\mathcal{I}_{\text{macro}},:} \cdot \boldsymbol{f}(t) \\
\boldsymbol{y}(t) &\geq \mathbf{0}\,, \\
\boldsymbol{ub} &\leq \boldsymbol{f}(t) \leq \boldsymbol{ub} \ \text{ for a. a. } t, \\
\mathbf{H}_{\boldsymbol{y}} \cdot \boldsymbol{y}(t) + \mathbf{H}_{\boldsymbol{f}} \cdot \boldsymbol{f}(t) &\leq \boldsymbol{h} \ \text{ for a. a. } t.
\end{aligned}
\tag{3.1}
$$

Here, the weight vector $\boldsymbol{b}(\cdot)$ is typically chosen as to maximize a (exponentially discounted) biomass production. Note that the QSSA in this case really substantiates

a model order reduction step: The internal metabolites completely vanish from the formulation. The positivity requirement on the compounds $\boldsymbol{y}$ is intuitively clear from their meaning as metabolite concentrations. For ODE-modeling, this is often not an important issue because the flow is mostly positivity-preserving, anyway. For the optimal control task in cFBA/deFBA, on the other hand, this introduces state constraints which complicates the numerical treatment. The mixed constraints in the last row correspond to enzyme-capacity constraints as we have seen in the above example already.

The state-of-the-art method in the context of cFBA/deFBA is to choose a constant time-step implicit time integration method (In the literature, trapezoidal rule and 1st/3rd/5th order RadauIIA collocation methods, [18] are used) and use a *full parametrization approach* of the optimal control task. This leads to a (very large and often badly scaled) linear optimization problem

$$\begin{aligned} &\min_{\boldsymbol{x}} \boldsymbol{c}^\top \cdot \boldsymbol{x}\,, \\ &\text{s.t. } \mathbf{A} \cdot \boldsymbol{x} \leq \boldsymbol{b}\,, \end{aligned} \tag{3.2}$$

which can be solved with powerful free and/or proprietary software libraries.
Pontryagin's Minimum Principle. To obtain necessary optimality conditions and circumvent the full parametrization procedure, one may introduce the *Hamiltonian*

$$\mathcal{H} := l_0 \cdot (\boldsymbol{b}(t))^\top \cdot \boldsymbol{y}(t) + (\boldsymbol{\lambda}(t))^\top \cdot \mathbf{S} \cdot \boldsymbol{f}(t)\,,$$

where the multiplier $l_0 \geq 0$ can usually be normed to $l_0 = 1$. For the *Lagrangian* $\mathcal{L}$, further multipliers $\boldsymbol{\mu}$ and $\boldsymbol{\nu}$ are introduced

$$\mathcal{L} = \mathcal{H} + (\boldsymbol{\mu}(t))^\top \cdot \boldsymbol{y}(t) + (\boldsymbol{\nu}(t))^\top \cdot \mathbf{S}_{\mathcal{I}_{\text{ind}},:} \cdot \boldsymbol{f}(t)\,.$$

Penalty Formulation: To circumvent the difficulties of mixed constraints in optimal control problems, we furthermore introduce logarithmic penalty terms

$$\boldsymbol{h} - \mathbf{H}_{\boldsymbol{y}} \cdot \boldsymbol{y} - \mathbf{H}_{\boldsymbol{f}} \cdot \boldsymbol{f} \geq 0 \;\Rightarrow\; \boldsymbol{\phi}(t, \boldsymbol{y}, \boldsymbol{f}) := -\epsilon \sum_i \ln((\boldsymbol{h} - \mathbf{H}_{\boldsymbol{y}} \cdot \boldsymbol{y} - \mathbf{H}_{\boldsymbol{f}} \cdot \boldsymbol{f})_i)$$

with a penalty parameter ϵ The *adjoint dynamics* within the Pontryagin minimum principle read

$$\dot{\boldsymbol{\lambda}}(t) = -\partial_{\boldsymbol{y}} \mathcal{L}$$

and are valid on sub-intervals of equal activity of mixed constraints, [14,61]. Adding the *local optimality* condition for $\boldsymbol{f}$:

$$\boldsymbol{f}(t) = \underset{\boldsymbol{lb} \leq \boldsymbol{u} \leq \boldsymbol{ub}}{\arg\min}\, \mathcal{H}(t, \boldsymbol{y}, \boldsymbol{u}, \boldsymbol{\lambda})\,,$$

which can either be stated by the variational formulation or from the necessary condition of local optimality

$$\mathbf{0} = \partial_{\boldsymbol{u}} \mathcal{H}\,,$$

(which only enforces optimality in the interior of the feasible set for $\boldsymbol{f}$) forms a DAE-boundary value problem from the Pontryagin principle, which can be solved by adaptive finite differences or finite elements in a more stable manner than the complete parametrization approach presented above.

Remark 3.2. The choice

$$\boldsymbol{b}(t) \equiv \begin{pmatrix} 0 & 0 & -1 & 0 & 0 \end{pmatrix}^\top$$

is a natural one as the organism has an interest in growing fast which is achieved by accumulating the biomass $P(t)$. One can, however, show that this leads to singular arcs in the profile and a non-unique solution to the optimal control problem.

Already in [62] it is pointed out that the solutions of deFBA problems oftentimes are not unique. In practical computations, this of course hampers an efficient numerical treatment. In that respect (and in light of the inverse problem viewpoint of this paper), we introduce a first regularization term on the objective vector $\boldsymbol{b}$ as

$$\boldsymbol{b} \rightsquigarrow \boldsymbol{b} + \begin{pmatrix} \epsilon & \epsilon & 0 & 0 & 0 \end{pmatrix}^\top .$$

This step can itself be interpreted as a biologically inspired regularization, since the production and maintenance of enzymes in the cell is a costly process.

Remark 3.3. In the original formulation of cFBA/deFBA, the introduction of enzymes is a modeling aspect of the framework itself. In fact, many publicly available metabolic network models concentrate on internal metabolism and many macromolecules (enzymes, ribosomes etc.) need to be added manually, anyway, if one is interested in solutions that consider enzyme-costs (or realistic cell growth, for that matter). The introduction of this macro-molecular apparatus with enzymes, ribosomes, and maybe even DNA/RNA replication costs might itself be viewed as a regularization: Taking simply the presence of proteins as control variables would in general lead to jumps in their concentrations. This is biologically not meaningful, especially the allocation of macro-molecules is a time consuming process compared to other adaptations in a cell.

In Figure 4, the results of the deFBA model with the adapted weight vector is displayed. The diauxic shift can clearly be recognized: First, the cell increases the

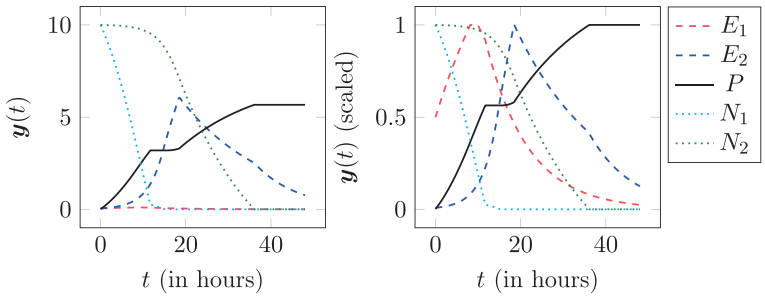

FIGURE 4. Solution of the deFBA problem, right: metabolite concentrations scaled by their maximum value over the time interval

amount of enzyme E_1 until its scarcity no longer defines an upper bound for the uptake of nutrient N_1, which then gets quickly consumed. At the same time, the cell can slowly start feeding on nutrient N_2 and to build up the necessary enzyme for that. The biomass-curve $P(t)$ first increases strongly, then almost completely saturates for a moment until sufficient E_2 is ready to fully concentrate on the consumption of the second energy source. When that is also exhausted, all reactions

come to an end; the enzyme degradation reactions are subjected to lower bounds such that these also take some time. Notice, how for this simple example, we already have more than five qualitatively completely different phases in the experiment.

Observation 3.4. The problem structure of deFBA can be understood in terms of the inverse problem framework (2.2) for the choices

- $x := \boldsymbol{f}(t)$ for $t \in [t_0, t_{\text{end}}]$,
- $\mathcal{T} = \emptyset$, $\mathbb{Y} = \emptyset$,
- $\mathbb{X} = L^\infty(\mathbb{R}^{n_{\boldsymbol{f}}})$
- $\mathbb{D} = \{\boldsymbol{f}(\cdot) \in \mathbb{X} : \boldsymbol{lb} \overset{L^2}{\leq} \boldsymbol{f} \overset{L^2}{\leq} \boldsymbol{ub}, \mathbf{S}_{\mathcal{I}_{\text{ind}},:} \cdot \boldsymbol{f} \overset{L^2}{=} \mathbf{0}, \text{ and } \mathbf{H}_{\boldsymbol{y}} \cdot \boldsymbol{y} + \mathbf{H}_{\boldsymbol{f}} \cdot \boldsymbol{f} \overset{L^2}{\leq} \boldsymbol{h}, \boldsymbol{y} \geq \mathbf{0} \text{ for } \boldsymbol{y} \text{ solving } \dot{y} = \mathbf{S}_{\mathcal{I}_{\text{macro}},:} \cdot \boldsymbol{f}\}$

where $L^\infty(\mathbb{R}^{n_{\boldsymbol{f}}})$ denotes the space of bounded functions and $\overset{L^2}{\bullet}$ means that the relation has to hold almost everywhere on $[t_0, t_{\text{end}}]$.

The solution procedure within deFBA follows the feasibility formulation approach with the bio-inspired regularization as indicated in the first line of (3.1).

3.4. **Universal Algorithms.** As outlined in Section 2 above, the recent progress in availability of powerful software libraries for artificial intelligence, usually by means of neural networks in its various forms [22, 41], has lately lead to a spike in the research activities in this area. Having a computational method like cFBA/deFBA at hand one is no longer strictly required to rely on expensive experimentally obtained time series data of metabolite concentrations to train the algorithms.

A proof-of-concept for the inference of *logical rules* in metabolic systems by means of neural networks was presented in [44]: From a simple setup of a neural network, logical rules like 'if metabolite A is present, flux B must be completely turned off.' were obtained. Such rules can be used for cross-species analysis, for verification of the cFBA/deFBA models, and also to use them in more refined models whose simulation requires less computational effort. Apart from the pure ODE/DAE-modeling (which we will address in Paragraph 3.6 below) different 'dynamic versions' of FBA have been proposed.

3.5. **Flux-Balance Analysis (FBA).** The use of linear programming for bio-engineering and systems biology purposes has a long history. However, the work of Palsson and co-workers [58], see also [42] for a review, has paved the road for the excessive and somewhat standardized use of the techniques.

In its classical form, FBA aims at finding the biochemical fluxes at a fixed point in time t^* assuming that the entire metabolism has already reached a steady-state, i. e. $\mathcal{I}_{\text{int}} = \{1, 2, \ldots, n_{\boldsymbol{y}}\}$ in (1.3). Like in cFBA/deFBA, for the fluxes we additionally assume upper and lower bounds such that the introduction of a (linear) objective like biomass flux maximization

$$\min_{\boldsymbol{f} \text{ feasible}} -\boldsymbol{f}_{\text{biomass}}$$

leads to an LP (3.2), defining the fluxes.

To obtain a dynamic version of FBA, so-called *iterative FBA* (or, not consistently sometimes also called dynamic FBA) has been introduced. As the name suggests, the fluxes are fed into an (typically explicit) algorithm for the time integration of the

biomass reaction which can be used to update metabolite concentrations and this procedure is iterated, see Figure 5. In the language of control theory, this can be

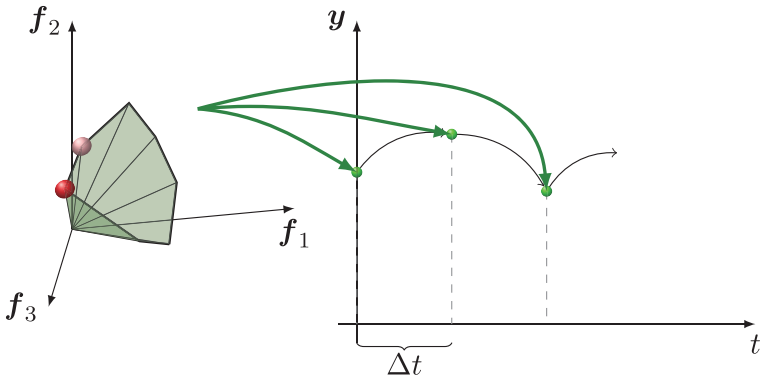

FIGURE 5. Schematic illustration of iterative versions of FBA

seen as an open loop control task. Iterative FBA can be combined with logical rules leading to so-called *regulatory FBA* (rFBA): Here, in every step before the solution of the LP, the logical rules are checked and potential knockout conditions (upper and lower bounds of certain fluxes are set to zero) are added to the constraints or existing knockouts released. Information like the rules acquired from universal algorithms (see Paragraph 3.4) or model checking algorithms can then be kept in their discrete/logical form and directly build into the semi-dynamic models.

Observation 3.5. (Iterative, regulatory) FBA can be regarded as an (iterative) application of the inverse problem framework (2.2), by means of the choices

- $x = \boldsymbol{f}(t^*)$,
- $\mathcal{T} = \emptyset$, $\mathbb{Y} = \emptyset$,
- $p = \{t^*, \langle\text{current set of knockout fluxes}\rangle\}$
- $\mathbb{X} = \mathbb{R}^{n_{\boldsymbol{f}}}$, $\mathbb{D} = \{\boldsymbol{f} : \mathbf{S} \cdot \boldsymbol{f} = \mathbf{0}, \boldsymbol{lb} \leq \boldsymbol{f} \leq \boldsymbol{ub}\}$

Using the biomass flux objective function is a bio-inspired regularization in the sense of the feasibility approach.

Remark 3.6. The bad conditioning of the inverse problem is particularly apparent for this framework: While the determination of the optimal value of an LP is usually considered a (mostly well-conditioned) standard problem in optimization and numerical mathematics, in FBA one solves for the solution *vectors*. Here, small input data changes can lead to the solution jumping from one corner of the feasible region to the next with possibly large impact on the solution which is processed further on. A simple regularization in terms of efficiency (corresponding to Tikhonov regularization again) can resolve this issue easily, but the application of linear programming solvers is no longer possible in that case.

In realistic examples, the solution space of the FBA problems is almost never unique as shown for many case studies, [25]. The success and the very existence of flux-variability analysis (FVA, i. e. a framework to enumerate all possible flux combinations that solve the LP) shows that the non-uniqueness is in fact an important issue and not just a theoretical aspect.

Remark 3.7. For iterative FBA and cFBA/deFBA, we completely neglected the evolution operator $\mathcal{T}$ and measurable data x^{meas}. This classification is obviously not the only possibility since a time evolution is at the basis of all these modeling frameworks. Nevertheless, both concepts fall into constraint-based models in systems biology such that this completely constraint-based feasibility approach seems to be the most reasonable one.

3.6. **Parameter Estimation.** As a last example, we come back to the ODE modeling of metabolic networks. If time series data (not necessarily in abundance as for universal algorithms) is available and sufficient model information/expertise to refine the optimal control model from cFBA/deFBA, the system identification procedure can be recast as a parameter estimation problem, see the monographs [47] and [10] for a general introduction and applications in systems biology and [59] for a review.

In terms of a inverse problem reformulation, this is probably the most straightforward example as it is an approximation problem to start with. For the proposed benchmark, the following ansatz functions for the flux vector were chosen

$$
\begin{aligned}
f_1 &= p_1 \cdot y_{N_1}(t) \cdot 10 \cdot \frac{y_{E_1}(t)^{p_4}}{p_3^{p_4} + y_{E_1}(t)^{p_4}}\,,\\
f_2 &= p_2 \cdot y_{N_2}(t) \cdot \frac{1}{10} \cdot \frac{y_{E_2}(t)^{p_6}}{p_5^{p_6} + y_{E_2}(t)^{p_6}} \cdot \frac{1}{p_7^{p_8} + y_{M_1}(t)^{p_8}}\,,\\
f_3 &= \frac{y_{M_1}(t)}{p_9 + y_{M_1}(t)}\,,\\
f_4 &= \frac{y_{M_2}(t)}{p_{10} + y_{M_2}(t)}\,,\\
f_5 &= \frac{1}{100} \cdot \frac{y_{M_1}(t) \cdot y_{M_3}(t)}{p_{11} + y_{M_1}(t) \cdot y_{M_3}(t)}\,,\\
f_6 &= \frac{y_{M_2}(t) \cdot y_{M_3}(t)}{p_{12} + y_{M_2}(t) \cdot y_{M_3}(t)}\,,\\
f_7 &= \frac{y_{M_3}(t)}{p_{13} + y_{M_3}(t)}\,,\\
f_8 &= \frac{1}{10} \cdot \frac{y_{E_1}(t)}{1 + y_{E_1}(t)}\,,\\
f_9 &= \frac{1}{10} \cdot \frac{y_{E_2}(t)}{1 + y_{E_2}(t)}\,,
\end{aligned}
$$

where the bounds for the fluxes (if not included through activation/inhibition terms) are explicitly built in already. The inhibition function in f_2 symbolizes the inhibitory part that could stem from a rule 'If metabolite M_2 is present, there is no need for activating flux f_2.' The mathematical problem is to find parameter values $\boldsymbol{p} \in \mathbb{R}^{13}$ such that the deviation from given data points is as small as possible.

For our experiment, the optimization was performed using the simplex-algorithm of Nelder and Mead [40], known for its robustness, with additional penalty terms to

avoid negative parameter values (log-penalty, see above) and a (bio-inspired) regularization by means of a quadratic penalty term to forbid the internal metabolites to accumulate too much. The latter regularization is closely related to the Levenberg–Marquardt method [10] for accelerating descent methods in general nonlinear programming problems.

The time integration was performed using the implicit (NDF) Matlab time integration method `ode15s.m` [49] and the given data were obtained from cFBA/deFBA with interpolated values at 20 equally distributed time instances, see Figure 6.

We obtained the solution vector

$$\boldsymbol{p} \approx \big(0.563\ 1.486\ 0.398\ 2.100\ 0.408\ 1.7623\ 0.653\ 0.681\ 1.983\ 0.783 \\ 0.121\ 0.364\ 1.828\big)^{\top}$$

and the results are also included into Figure 6.

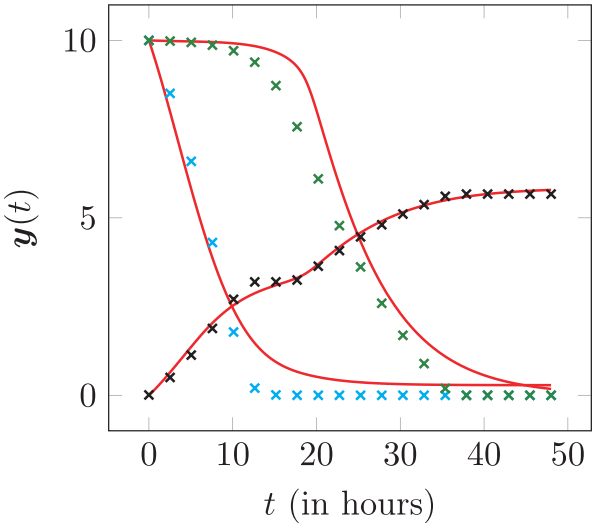

FIGURE 6. Results of the parameter estimation problem and given data points for nutrients N_1, N_2 and biomass P (ODE results in red solid line)

Observation 3.8. The parameter estimation problem can be cast as the inverse problem in the sense of (2.2) for the choices:

- $x = \boldsymbol{p}$, $\mathbb{X} = \mathbb{R}^{13}$,
- $\mathcal{T}\colon \boldsymbol{p} \to \{\boldsymbol{y}(t_i)\}_{i=1}^{20}$, such that $\dot{y}(t) = \mathbf{S}\boldsymbol{f}(t, \boldsymbol{y}(t); \boldsymbol{p}(t))$ for all $t \in [t_0, t_{\text{end}}]$.

For out experiment, we kept out the enzyme-capacity constraints for the uptake reactions, which would have introduced nonlinear constraints (and not have improved the solution).

Remark 3.9. As we use penalty techniques to ensure the positivity of the parameters, this requirement is viewed as a soft constraint. Alternatively, the classification could have included these as hard constraints $\mathbb{D} = \{\boldsymbol{p} : \boldsymbol{p} \geq \mathbf{0}\}$.

4. Conclusion

The formulation of various problems at almost all stages of the model building and simulation process in systems biology offers a multitude of benefits: It allows for a more standardized and therefore better communication and clearer representation of the identification tasks, which is crucial not at last because of the large size and complicated structure that these models show very often. Inverse problems open the possibility to use the same theoretical background and software libraries on various stages. Once such a general framework is established, it simply allows for construction of new computational frameworks by combining different techniques and regularization terms inspired by evolutionary principles.

So far, we did not mention further aspects of inverse problems in systems biology such as (inverse) bifurcation analysis which may be seen as an example of *qualitative inverse problems*, spatially distributed phenomena, explicit stochasticity in the models, the explicit consideration of thermodynamics, or chemical reaction network theory in general; to this point, we have in fact merely scratched the surface of this vast emerging topic.

Our next steps in this direction will include the use of iterative projection techniques for the solution of the feasibility problem form of these inverse problems and the exploitation of bio-inspired optimization principles on other stages. An important milestone for this field would lie in the establishment of a clear roadmap or workflow on how to uniquely classify existing frameworks but such a workflow would have to be the combined effort from various sides within the systems biology community.

References

[1] L. Ambrosio and G. Dal Maso, *A general chain rule for distributional derivatives*, Proceedings of the American Mathematical Society **108** (1990), 691–691.

[2] M. Bartl, M. Kötzing, St. Schuster, P. Li, and Chr. Kaleta, *Dynamic optimization identifies optimal programmes for pathway regulation in prokaryotes*, Nature Communications, **4(2243)** (2013).

[3] H. H. Bauschke and J. M. Borwein, *On projection algorithms for solving convex feasibility problems*, SIAM Review **38** (1996), 367–426.

[4] A. Ben-Tal, L. El Ghaoui and A. Nemirovski, *Robust Optimization*, Princeton University Press, 2009.

[5] M. Bizzarri (editor), *Systems Biology*, Springer New York, 2018.

[6] A. P. Burgard, P. Pharkya and C.D. Maranas, *Optknock: A bilevel programming framework for identifying gene knockout strategies for microbial strain optimization*, Biotechnology and Bioengineering **84** (2003), 647–657.

[7] Y. Crama and P. L. Hammer, *Boolean Functions: Theory, Algorithms, and Applications (Encyclopedia of Mathematics and its Applications)*, Cambridge University Press, 2011.

[8] G.M. de Hijas-Liste, E. Balsa-Canto, J. Ewald, M. Bartl, P. Li, J. R. Banga and Chr. Kaleta, *Optimal programs of pathway control: dissecting the influence of pathway topology and feedback inhibition on pathway regulation*, BMC Bioinformatics **16** (2015), 163 (13 pages).

[9] P. Deuflhard and E. Hairer (editors), *Numerical Treatment of Inverse Problems in Differential and Integral Equations*, Birkh 辰 user Boston, 1983.

[10] P. Deuflhard and S. Röblitz, *A Guide to Numerical Modelling in Systems Biology*, Number 12 in Texts in Computational Science and Engineering. Springer International Publishing Switzerland, 2015.

[11] H. W. Engl, Chr. Flamm, Ph. Kügler, J. Lu, St. Müller and P. Schuster, *Inverse problems in systems biology*, Inverse Problems **25(12)** (2009), 123014 (51 pages).
[12] E. H. Flach and S. D. Schnell, *Use and abuse of quasi-steady-state approximation*, IET Systems Biology **153** (2006), 187–191.
[13] P. G. Georgiev and M. Z. Nashed, *Continuous dependence of the solution of optimal control problems on parameter*, Applied Analysis and Optimiziation **2** (2018), 1–10.
[14] M. Gerdts, *Optimal Control of ODEs and DAEs*, De Gruyter, Berlin, Boston, 2011.
[15] A. Goelzer, V. Fromion and G. Scorletti, it Cell design in bacteria as a convex optimization problem, Automatica **47** (2011), 1210–1218.
[16] A. Göpfert, H. Riahi, Chr. Tammer and C. Zălinescu, *Variational Methods in Partially Ordered Spaces*, Springer-Verlag, 2003.
[17] M. Hadamard, *Les problèmes aux limites dans la théorie des équations aux dérivées partielles*, Journal de Physique Théorique et Appliquée **6** (1907), 202–241.
[18] E. Hairer and G. Wanner, *Solving Ordinary Differential Equations II Stiff and Differential-Algebraic Problems*, Springer Berlin Heidelberg, 2nd edition, 1996.
[19] M. D. Hanwell, D. E. Curtis, D. C. Lonie, T. Vandermeersch, E. Zurek and G. R. Hutchison, *Avogadro: an advanced semantic chemical editor, visualization, and analysis platform*, Journal of Cheminformatics **4** (2012), 17.
[20] R. Heinrich, St. Schuster and H.-G. Holzhütter, *Mathematical analysis of enzymic reaction systems using optimization principles*, European Journal of Biochemistry **201** (1991), 1–21.
[21] D. J. Higham, *Modeling and simulating chemical reactions*, SIAM Review, **50** (2008), 347–368.
[22] G. E. Hinton, *Learning multiple layers of representation*, Trends in Cognitive Sciences **11** (2007), 428–434.
[23] F. Horn and R. Jackson, *General mass action kinetics*, Archive for Rational Mechanics and Analysis **47** (1972), 81–116.
[24] M. Hucka, A. Finney, H. M. Sauro, H. Bolouri, J. C. Doyle, H. Kitano, A. P. Arkin, B. J. Bornstein, D. Bray, A. Cornish Bowden, A.A. Cuellar, S. Dronov, E.D. Gilles, M. Ginkel, V. Gor, I. I. Goryanin, W. J. Hedley, T. C. Hodgman, J.-H. Hofmeyr, P. J. Hunter, N. S. Juty, J. L. Kasberger, A. Kremling, U. Kummer, N. Le Novère, L. M. Loew, D. Lucio, P. Mendes, E. Minch, E. D. Mjolsness, Y. Nakayama, M.R. Nelson, P. F. Nielsen, T. Sakurada, J. C. Schaff, B. E. Shapiro, T. S. Shimizu, H. D. Spence, J. Stelling, K. Takahashi, M. Tomita, J. Wagner and J. Wang, *The systems biology markup language (SBML): A medium for representation and exchange of biochemical network models*, Bioinformatics **19** (2003), 524–531.
[25] St. M. Kelk, B. G. Olivier, L. Stougie, and F.J. Bruggeman, *Optimal flux spaces of genome-scale stoichiometric models are determined by a few subnetworks*, Scientific Reports **2** (2012).
[26] A. A. Khan, Chr. Tammer and C. Zălinescu, *Set-valued Optimization*, Springer Berlin Heidelberg, 2015.
[27] B. T. Kien and J.-C. Yao, *Semicontinuity of the solution map to a parametric optimal control problem*, Applied Analysis and Optimization **2** (2018), 93–116.
[28] A. Kirsch, *An Introduction to the Mathematical Theory of Inverse Problems*, Springer New York, 2011.
[29] H. Kitano, *Biological robustness*, Nature Reviews Genetics **5** (2004), 826–837.
[30] E. Klipp, *Timing matters*, FEBS Letters **583** (2009), 4013–4018.
[31] E. Klipp, R. Heinrich and H.-G. Holzhütter, *Prediction of temporal gene expression*, European Journal of Biochemistry **269** (2002), 5406–5413.
[32] E. Klipp, W. Liebermeister, Chr. Wierling, A. Kowald and R. Herwig, *Systems Biology*, Wiley VCH Verlag GmbH, 2nd edition, 2016.
[33] D. Ma, L. Yang, R. M. T. Fleming, I. Thiele, B. Ø. Palsson and M. A. Saunders, *Reliable and efficient solution of genome-scale models of metabolism and macromolecular expression*, Scientific Reports **7** (2017), 40863.
[34] T. R. Maarleveld, M. T. Wortel, B. G. Olivier, B. Teusink and F. J. Bruggeman, *Interplay between constraints, objectives, and optimality for genome-scale stoichiometric models*, PLOS Computational Biology **11** (2015), e1004166.

[35] R. Mahadevan, J. S. Edwards and F. J. Doyle, *Dynamic flux balance analysis of diauxic growth in escherichia coli*, Biophysical Journal **83** (2002), 1331–1340.
[36] R. Mahadevan and C. H. Schilling, *The effects of alternate optimal solutions in constraint-based genome-scale metabolic models*, Metabolic Engineering **5** (2003), 264–276.
[37] A. Mahendran and A. Vedaldi, *Understanding deep image representations by inverting them*, in: *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2015.
[38] L. Miskovic, M. Tokic, G. Fengos, and V. Hatzimanikatis, *Rites of passage: requirements and standards for building kinetic models of metabolic phenotypes*, Current Opinion in Biotechnology **36** (2015), 146–153.
[39] G. Montavon, S. Lapuschkin, A. Binder, W. Samek and K.-R. Müller, *Explaining nonlinear classification decisions with deep taylor decomposition*, Pattern Recognition **65** (2017), 211–222.
[40] J.A. Nelder and R. Mead, *A simplex method for function minimization*, The Computer Journal **7** (1965), 308–313.
[41] M.A. Nielsen, *Neural Networks and Deep Learning*, Determination Press, 2015. neuralnetworksanddeeplearning.com.
[42] J.D. Orth, I. Thiele and B. Ø. Palsson, *What is flux balance analysis?* Nature Biotechnology **28** (2010), 245–248.
[43] D. A. Oyarzún, *A control-theoretic approach to dynamic optimization of metabolic networks*, PhD thesis, Hamilton Institute, National University of Ireland Maynooth, 2010.
[44] A.-M. Reimers, *Understanding metabolic regulation and cellular resource allocation through optimization*, PhD thesis, Freie Universität, Berlin, 2017.
[45] A.-M. Reimers, H. Knoop, A. Bockmayr and R. Steuer, *Cellular trade-offs and optimal resource allocation during cyanobacterial diurnal growth*, Proceedings of the National Academy of Sciences USA **114** (2017), E6457–E6465.
[46] M. Rügen, A. Bockmayr and R. Steuer, *Elucidating temporal resource allocation and diurnal dynamics in phototrophic metabolism using conditional FBA* Scientific Reports **5** (2015), 15247 (16 pages).
[47] K. Schittkowski, *Numerical Data Fitting in Dynamical Systems*, Springer US, 2002.
[48] L. A. Segel and M. Slemrod, *The quasi-steady-state assumption: A case study in perturbation*, SIAM Review **3** (1989), 446–477.
[49] L. F. Shampine and M. W. Reichelt, *The MATLAB ODE suite*, SIAM Journal on Scientific Computing **18** (1997), 1–22.
[50] H.-S. Song and D. Ramkrishna, *When is the quasi-steady-state approximation admissible in metabolic modeling? when admissible, what models are desirable?* Industrial & Engineering Chemistry Research **48** (2009), 7976–7985.
[51] I. Steinwart and A. Christmann, *Support Vector Machines*, Springer-Verlag GmbH, 2008.
[52] J. Stelling, U. Sauer, Z. Szallasi, F. J. Doyle and J. Doyle, *Robustness of cellular functions*, Cell **118** (2004), 675–685.
[53] W.J. Sutherland, *The best solution*, Nature **435** (2005), 569.
[54] I. Thiele and B.Ø. Palsson, *A protocol for generating a high-quality genome-scale metabolic reconstruction*, Nature Protocols **5** (2010), 93–121.
[55] A. N. Tikhonov, A. Goncharsky, V. V. Stepanov and A.G. Yagola, *Numerical Methods for the Solution of Ill-Posed Problems*, Number 328 in Mathematics and Its Applications. Springer Netherlands, 1995. Originally published in Russian.
[56] M. Timme and J. Casadiego, *Revealing networks from dynamics: an introduction*, Journal of Physics A: Mathematical and Theoretical **47** (2014), 343001.
[57] N. Tsiantis, E. Balsa-Canto and J. R. Banga, *Optimality and identification of dynamic models in systems biology: an inverse optimal control framework*, Bioinformatics (2018), page bty139.
[58] A. Varma and B. Ø. Palsson, *Metabolic flux balancing: Basic concepts, scientific and practical use*, Bio/Technology **12** (1994), 994–998.
[59] A. F. Villaverde and J. R. Banga, *Reverse engineering and identification in systems biology: strategies, perspectives and challenges*, Journal of The Royal Society Interface, **11** (2013), 20130505–20130505.

[60] St. Waldherr and F. Allgöwer, *Robust stability and instability of biochemical networks with parametric uncertainty*, Automatica **47** (2011), 1139–1146.
[61] St. Waldherr and H. Lindhorst, *Optimality in cellular storage via the Pontryagin maximum principle*, IFAC-PapersOnLine **50** (2017), 9889–9895.
[62] St. Waldherr, D.A. Oyarzún and A. Bockmayr, *Dynamic optimization of metabolic networks coupled with gene expression*, Journal of Theoretical Biology **365** (2015), 469–485.
[63] H.-Q. Wu, M.-L. Cheng, J.-M. Lai, H.-H. Wu, M.-C. Chen, W.-H. Liu, W.-H. Wu, P. M.-H. Chang, C.-Y. F. Huang, A.-P. Tsou, M.-S. Shiao and F.-S. Wang, *Flux balance analysis predicts Warburg-like effects of mouse hepatocyte deficient in miR-122a*, PLOS Computational Biology **13** (2017), e1005618.

Manuscript received December 8 2018
revised December 10 2018

Markus Arthur Köbis
Department of Mathematics and Computer Science, Freie Universität Berlin, Germany
E-mail address: markus.koebis@fu-berlin.de