Luenberger observer design for a dynamic system with embedded linear program, applied to bioprocesses

Kobe De Becker, Kristel Bernaerts and Steffen Waldherr

Bio- & Chemical Systems Technology, Reactor Engineering and Safety (CREaS), KU Leuven, Celestijnenlaan 200F, 3001 Leuven (email: kobe.debecker@kuleuven.be)

Abstract: Microbial dynamics are fundamental to many processes in medicine and biotechnology. To model, estimate, and control such growth dynamics, methods of systems theory and control engineering are applied. In this paper, we use a modelling framework of dynamic constraint-based models, which appears as a system of ordinary differential equations of which the right hand side depends linearly on the optimal solution of a linear program (LP). This model describes the changes in the concentrations of extracellular metabolites and the amounts of all considered biomass components. The trajectories of the models are characterized by state-dependent switches among different optimal bases of the LP problem. The dynamics corresponding to each of these optimal bases are denoted as modes of the system.
Based on such models, we study an online estimation problem in which the state variables are to be estimated from measurements according to a linear output equation. Due to the switching nature of the trajectories, we propose to use a bank of linear Luenberger observers for the different optimal bases of the LP. The system mode is estimated by a moving average of the error norm. An observer gain for each mode is determined by solving a set of Riccati equations with a common Lyapunov matrix. Simulation studies for a toy model with two bacterial species show the feasibility of this approach; from measurements of substrate and total biomass only, the observer is capable of correctly predicting the individual biomasses of the two species during exponential growth.

Keywords: Observer design, Linear programming, Switched systems, Simultaneous stabilization, Biotechnology

1. INTRODUCTION

Measuring the complete process state of a bioprocess is in practice rarely feasible due to a lack of available sensors. These missing sensors are either too expensive or are not yet developed for on-line application [1]. Based on the available measurements and a model description of the process, an observer gives an estimate of the non-measured process states. The available measurements commonly involve extracellular variables (substrate or product amounts) or the total biomass amount.

A current challenge in observer design for bioprocesses is to design observers that are compatible with recently developed, advanced modelling techniques. Traditional bioreactor models, often referred to as *unstructured* models, describe a bioreactor by a set of ordinary differential equations (ODEs), originating from mass balancing. Unstructured models consider biomass as one overall variable. Classic observers for bioprocesses typically focus on these unstructured models. These models have a limited estimation capability as only the total biomass amount is considered, instead of the different biomass constituents [3]. This partition of biomass into multiple components is taken into account in *structured* models [2]. These models consider cells to be consisting of multiple macromolecules such as DNA, RNA, cell wall, lipids and enzymes. Especially for complex bioprocesses with multiple substrates, products and microorganisms, structured models are indispensable for accurate predictions. The purpose of this paper is to prove the feasibility of an observer design method for one specific class of structured models: the constraint based optimisation models.

Constraint based optimisation has been widely used to predict and analyse cellular behaviour, especially for microorganisms [4]. The classic method, called Flux Balance Analysis (FBA), optimises a predefined objective function to find an optimal reaction flux distribution in a (genome-scale) metabolic network structure, which is subjected to algebraic constraints based on steady state mass balance equations and reaction directionalities [5]. To include the dynamic nature of bioprocesses, FBA has been extended to dynamic Flux Balance Analysis (dFBA), in which the steady state FBA problem is coupled with ODEs acting on the reactor scale [6]. This set of ODEs describes the change of extracellular metabolite amounts and total biomass with respect to time.

This dFBA framework has been extended to take metabolic

⋆ This work has been funded by Fonds voor Wetenschappelijk Onderzoek Vlaanderen (FWO), grant number 1S97218N

enzyme costs into account [7]. This extension resulted in dynamic enzyme-cost Flux Balance Analysis (deFBA) models, which consider not only enzyme-costs, but also a varying biomass composition. The formulation of a deFBA model consists of a set of ODEs, describing the change of substrate, product and biomass component amounts. The right-hand side of the ODEs contains an embedded linear programming (LP) problem which returns the optimal reaction fluxes. The formulation employed in this paper and the accompanying numerical solution method are similar (and partially based on) the DFBAlab software package [8; 9; 10].

The applied solution method results in a linear switched system which shows state-dependent mode switches. Each of these modes is defined by the current optimal basis of the linear programming problem. Observers for linear switching systems have already been developed in case of time-dependent mode switches with a priori known switching times [11]. This method requires a complete list of possible modes, although observability of these modes is not a prerequisite. Unfortunately, for deFBA models switching times are not a priori known. Furthermore, the large number of possible modes poses an additional challenge in terms of computational efficiency. This paper has as purpose to show the feasibility of an observer design approach with a mode estimation procedure based on a bank of Luenberger observers, one for each mode.

The paper is structured into three parts: first, an overview of deFBA models is given, together with the discretisation and numerical solution method; second, the construction of the bank of Luenberger observers for deFBA models is discussed; third, the simulation results of the proposed observer for a toy model with a simple network structure are discussed.

2. DYNAMIC ENZYME-COST FLUX BALANCE ANALYSIS

2.1 Model description

This section gives an overview of the deFBA modelling framework. This framework has originally been introduced in [7]. In this work, the modelling format is changed as compared to [7] such that the deFBA problem is written as a set of ODEs with an embedded LP problem in the right-hand side. A literature-based solution method for these type of models is introduced.

deFBA is used to predict substrate, product and biomass component amount profiles, together with flux distribution profiles, based on a metabolic network representation of the microorganisms. Based on mass-balancing for each substrate, biomass component and product, a set of ODEs is written as

$$\dot{x} = S_{exch} v^*(x) + Ax + Gu. \tag{1}$$

x is a vector describing the process states, which are the substrate and product amounts x_1, and the biomass component amounts x_2, with $x = [x_1\ x_2]^{\mathrm{T}}$. $v^*(x)$ is the optimal reaction flux vector of all reactions of the metabolic network, depending on the process state via x. The *-sign indicates the optimality of the variable as it is obtained as (part of) the argument of the solution to an optimisation problem. S_{exch} is a partition of the full stoichiometric matrix S which represents the exchange reactions and biomass component production reactions. Ax describes mass transport on the reactor scale (e.g. dilution due to continuous operation), and Gu represents the influence of known inputs u.

The optimal flux vector v^* is obtained by solving a dynamic optimisation problem with objective function

$$\max_{\tilde{x}_1(\cdot),\tilde{x}_2(\cdot),v(\cdot)} \int_0^{t_p} c^T \tilde{x}_2(t)\mathrm{d}t, \tag{2}$$

with t_p the prediction horizon for the optimisation [12] and c a vector of weights. $\tilde{x}_1$ represents the substrate and product amounts, and $\tilde{x}_2$ the biomass component amounts in the optimisation problem. The objective (2) maximises the integrated sum of biomass components which has been shown to be a biologically reasonable objective [7].

Maximisation of the objective function occurs under several equality and inequality constraints. At first, the substrate, product and biomass component amounts follow the dynamic mass balance equations

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\tilde{x}_1\\ \tilde{x}_2\end{bmatrix} = S_{exch}v + A\begin{bmatrix}\tilde{x}_1\\ \tilde{x}_2\end{bmatrix},\ \begin{bmatrix}\tilde{x}_1(0)\\ \tilde{x}_2(0)\end{bmatrix} = x. \tag{3}$$

Intracellular metabolite amounts are assumed to be in steady-state. This assumption is based on the observation that the dynamics of intracellular processes are faster than those of extracellular processes and changes in biomass composition. Therefore, the model includes the equality constraint

$$0 = S_m v, \tag{4}$$

with S_m the partition of the stoichiometric matrix, only taking intracellular metabolites into account.

Second, the biomass components consist of structural compounds such as cell wall and lipids, further referred to as quota compounds, and enzymes. Enzymes have a limited catalytic capacity. A higher enzyme amount, results in a higher possible flux through the reaction that is catalysed by that enzyme. This is captured in the inequality constraint

$$H_C v \leqslant H_E \tilde{x}_2. \tag{5}$$

H_C is the capacity matrix, containing the inverse of the catalytic constants of the reactions. H_E is a filter matrix, containing the information about which enzymes catalyse which reactions.

Third, biomass composition constraints for biomass components can be added to ensure that a minimal fraction of the total biomass consists of one specific biomass component. This is written as

$$H_B \tilde{x}_2 \leqslant 0. \tag{6}$$

In addition, lower and upper boundaries on the fluxes can be imposed based on reaction directionality

$$v_{LB} \leqslant v \leqslant v_{UB}. \tag{7}$$

At last, all compound amounts should be non-negative

$$\tilde{x}_1, \tilde{x}_2 \geqslant 0,\ \forall t \in [0, t_p]. \tag{8}$$

Combining the ODEs and the dynamic optimisation problem results in the deFBA model.

Possibly, the optimisation problem (2) does not have a unique optimal solution. A unique flux vector is then found by executing a lexicographic optimisation in which multiple objective functions are hierarchically optimised. This method has been applied for dFBA models in [8; 9; 10].

2.2 Model discretisation by collocation

In order to solve the dynamic optimisation problem, it is discretised with respect to time by means of collocation. The time interval from the initial time 0 to the prediction horizon t_p is divided into N intervals, each of length h. Discretised variables for v, $\tilde{x}_1$ and $\tilde{x}_2$ are defined at each time point t_i, whereas discretised variables for the derivatives $\dot{\tilde{x}}_1$ and $\dot{\tilde{x}}_2$ are defined at the middle between two time points. To ensure continuity of the solution, discretisation equality constraints are added based on Euler discretisation of the derivatives. This results in an embedded LP problem in the right-hand side of the ODE system.

The methodology outlined in this section has been developed in the DFBAlab framework for dFBA models [8; 9; 10]. Here, the methodology is adapted for compatibility with deFBA models.

A recurrent problem with embedded optimisation problems is that they tend to become infeasible while integrating [10]. To overcome this problem, the LP problem is transformed to a Phase I LP problem. This method is explained in [8] for traditional dFBA models. For deFBA models, this method first yields a conversion of the LP problem to a standard form LP; slack variables are added to the inequality constraints and unbounded variables are written as the difference of two positive variables. In a second step, the Phase I problem is created; to each of the equality constraints a positive artificial variable a_+ is added, and another artificial variable a_- is subtracted. At last, an additional equality constraint is added that sums all artificial variables with a final artificial variable a.

Solving the resulting Phase I LP problem happens in two steps. First, the Phase I LP problem is solved for an objective function that minimises the value of a. When a^* equals zero, the initial LP problem is feasible. Otherwise, the positive value for a is integrated as a penalty function, which serves as a measure for the model infeasibility:

$$\dot{p} = a, \tag{9}$$

with p the integrated penalty function. Secondly, with the value of a constrained at its optimal value a^*, the Phase I LP problem is resolved for the discretised objective function from (2).

This procedure results in the following set of ODEs with an embedded LP problem in the right-hand side:

$$\dot{x} = S_{exch} v^*(x) + Ax + Gu \tag{10}$$

with

$$\begin{aligned} v^*(x) = \arg \max_q & \sum_{i=1}^{N} \tilde{c}_i q_i \\ \text{s.t. } & A_{eq} q = b(x) \\ & a = a^*. \end{aligned} \tag{11}$$

q is a vector containing all optimisation variables. These variables are the discretised fluxes, metabolite and biomass component amounts, amount derivatives and helper variables, together with slack variables and artificial variables. $\tilde{c}$ is a vector containing the discretised objective weights and A_{eq} is the equality constraint matrix of the Phase I LP problem. The rows of A_{eq} correspond to the deFBA constraints in discretised form. b is the right-hand side vector of the equality constraints and its values depend linearly on the values of the process states x via the initial conditions. Therefore, we can write that, in absence of inputs u in equation (3),

$$b(x) = \begin{bmatrix} \vdots \\ x \\ \vdots \end{bmatrix} = Ex, \tag{12}$$

with E a state-independent matrix of appropriate dimensions.

2.3 Model solution by integration

To avoid solving the embedded LP problem each time the right-hand side is evaluated during numerical integration, the LP problem is converted into a set of algebraic equations. At the start of the integration, the LP problem is solved once. This solution results in a set of basis and non-basis variables, q_B^* and q_{NB}^* respectively. As long as the current basis remains feasible and optimal, q^* equals

$$q^* = \begin{bmatrix} q_B^* \\ q_{NB}^* \end{bmatrix} = \begin{bmatrix} B_i^{-1} b(x) \\ 0 \end{bmatrix} = \begin{bmatrix} B_i^{-1} Ex \\ 0 \end{bmatrix}. \tag{13}$$

B_i is the current basis matrix of the embedded LP, which consists of the columns of A_{eq} corresponding to the basis variables [13]. The index i indicates the fact that different bases can be optimal at different times. Each of these bases defines a mode in which the system can be situated. A basis is no longer optimal when a basis variable becomes zero. During integration, at the time point at which such a zero crossing of one of the basis variables occurs, the LP problem is resolved, resulting in a new set of basis variables which consequently leads to entering a new mode described by a new basis matrix B_j. Mode switches are therefore state-dependent. These mode switches can in most cases be related to events on the reactor level, e.g. the depletion of a substrate within the prediction horizon.

As long as the system remains in one mode, the fluxes are linear with respect to the process states x. Therefore, the optimal flux vector is written as

$$v^*(x) = T_i q_B^* = T_i B_i^{-1} Ex \tag{14}$$

with T_i a matrix of appropriate dimensions selecting the non-zero fluxes at the current time point from q_B^*. Note that T_i is mode dependent by the index i as each mode can contain different fluxes as basis variables. Using equation (14), the governing set of ODEs (1) can be written as

$$\dot{x} = S_{exch} T_i B_i^{-1} Ex + Ax + Gu, \; x \in \mathcal{M}_i. \tag{15}$$

$\mathcal{M}_i$ is the set of states x that correspond to an optimal basis B_i. It represents a polytope in the $\mathbb{R}^n$-space, with n the number of states.

3. OBSERVER DESIGN

The challenge in designing an observer for deFBA models lies in the state-depending mode switching behaviour of the system. Mode switching introduces non-linear behaviour and can pose observability issues when the system runs into an unobservable mode. Additionally, it is not a priori known in which mode the system is situated, which possibly leads to estimation errors. This section discusses the design of an observer for the mode switching system resulting from deFBA models. The goal is to estimate the substrate, product and biomass components x based on limited extracellular metabolite amount or total biomass amount measurements. The measurements y are a linear combination of the process states

$$y = Cx, \tag{16}$$

with $C \in \mathbb{R}^{m\times n}$ the measurement matrix.

In what follows a bank of Luenberger observers will firstly be designed, which contains one observer for each detectable mode of a preliminary list of system modes. Secondly, mode estimation by a moving average error between process measurements and estimates is discussed. Lastly, the mode estimate is used in an overall Luenberger observer in order to perform state estimation for the deFBA model.

3.1 A bank of Luenberger observers

One of the challenges of the current observer design is the fact that a full list of possible system modes is not available. To circumvent this problem, a preliminary list of modes which are the most likely to be encountered is constructed by simulating the deFBA model multiple times according to the algorithm discussed in section 2.3. Each of these simulation runs starts with a different set of initial estimates for the substrate, product and biomass component amounts. All encountered modes are recorded and stored in a preliminary mode list. The variability in initial values from which the various simulations start, should be in proportion to the model certainty and the certainty with which the initial estimates for the actual observer are known. A higher certainty of these initial estimates requires a smaller variability in initial simulation values to obtain a list of most probable, encountered modes. The certainty with which these initial observer estimates are known is based on physical and biological understanding of the process of interest and is therefore heavily application dependent. The same arguments apply to the number of required simulation runs. Well chosen initial values for the initial model simulations should lead to a more complete preliminary mode list, and should therefore improve observer performance. The more modes in the preliminary mode list, the more versatile the observer dynamics can be. This more complete mode list, however, comes at a cost in terms of the required calculation effort.

All modes in the preliminary mode list are checked for detectability by the Hautus rank criterion [14]. Undetectable modes are not considered for observer design, and are, therefore, not considered in the bank of observers. Let l denote the number of detectable modes. For each of them, a Luenberger observer is designed, consisting of a simulation term that replicates the system dynamics (15) and a correction term based on the measured output. For the i-th mode, the observer equation is given by

$$\dot{\widehat{x}}_i = S_{exch}T_iB_i^{-1}E\widehat{x}_i + A\widehat{x}_i + Gu + L_i(y - C\widehat{x}_i), \tag{17}$$

where $\widehat{x}_i \in \mathbb{R}^n$ is the estimated state and L_i the observer gain for mode i.

To ensure stability of the bank of observers, gain determination for each detectable mode occurs by simultaneous stabilisation. This is achieved by finding a common Lyapunov function for the observer error dynamics with a distinct observer gain for each detectable mode, as outlined in [15].

First, a positive definite weighting matrix $Q \in \mathbb{R}^{n\times n}$ is chosen. Using Q, the observer gain L_i is designed such that the integrated estimation error

$$\int_0^{+\infty} e_i^TQe_i\mathrm{d}t \tag{18}$$

is minimised, where

$$e_i = x - \widehat{x}_i \tag{19}$$

is the estimation error of the observer for the i-th mode. The gain is determined by finding a matrix $P = P^T \in \mathbb{R}^{n\times n}$, a scalar λ, and l matrices $Y_i \in \mathbb{R}^{n\times m}$ that solve the following semidefinite program:

$$\begin{aligned} \min_{P,\lambda,Y_i} \quad & \lambda \\ \text{s.t.} \quad & 0 < P \\ & 0 < \lambda I - P \\ & 0 < (S_{exch}T_iB_i^{-1}E + A)^TP - C^TY_i^T + \ldots \\ & \quad P(S_{exch}T_iB_i^{-1}E + A) - Y_iC + Q, \ i = 1, \ldots, l. \end{aligned} \tag{20}$$

The observer gains L_i for each mode are subsequently determined as

$$L_i = P^{-1}Y_i. \tag{21}$$

The semidefinite program (20) is solved in MATLAB via YALMIP [16] using SeDuMi [17] as solver.

3.2 Mode estimation

In order to select the most fitting mode at each time point, a moving average of the estimation error between each Luenberger observer and the process measurements is defined for each mode of the observer bank. The index of the mode that results in the smallest moving average error is found as

$$\widehat{k} = \arg\min_i \ r_i \tag{22}$$

with

$$\dot{r}_i = \alpha(-r_i + \|y - C\widehat{x}_i\|). \tag{23}$$

Here, $\widehat{k}$ is the index corresponding to the mode estimate. r_i is the moving average error with respect to mode i, and α is a scalar with a predefined value.

The mode with the lowest moving average error r_i at a specific time point is considered to be the most optimal mode estimate. The parameter α influences the speed of changes in estimated modes. The larger α, the more rapidly the current error will influence the average error. This fact leads to more frequent changes in the estimated mode matrices T_i and B_i^{-1}. For large α, the presence of noise could lead to unnecessary and unwanted mode changes. On the other hand, for low α, the process could be in a non-optimal mode for a longer period.

Each mode in the preliminary list of modes is checked for detectability with respect to the measurement matrix C. In case the real life process runs into an undetectable mode, the observer will wrongly assume another (detectable) mode to be an optimal estimate, leading to an estimation error. Afterwards, if the real process returns to a detectable mode, the observer recovers in case of asymptotically stable error dynamics.

3.3 State estimation

The mode estimation $\widehat{k}$ at each time point is used for state estimation by an overall Luenberger observer. The proposed overall observer for (1) and the output defined by (16) is:

$$\dot{\widehat{x}} = S_{exch}v^*(\widehat{x}) + A\widehat{x} + Gu + L_{\widehat{k}}(y - \widehat{y}). \tag{24}$$

The estimation error is defined as the difference between the real value for the process states and the estimated values $e(t) = x(t) - \widehat{x}(t)$. The observer error dynamics are consequently written as

$$\dot{e} = S_{exch}(v^*(x) - v^*(\widehat{x})) + (A - L_{\widehat{k}}C)e. \tag{25}$$

Using equation (14) for the optimal flux vector calculation leads to the full error dynamics:

$$\begin{aligned} \dot{e} &= S_{exch}(T_k B_k^{-1} - T_{\widehat{k}} B_{\widehat{k}}^{-1})Ex + \dots \\ &(S_{exch}T_{\widehat{k}}B_{\widehat{k}}^{-1}E + A - L_{\widehat{k}}C)e, \end{aligned} \tag{26}$$

with k the index of the real system mode.

In case the correct system mode is estimated, i.e. $k = \widehat{k}$, the origin is a stable equilibrium point of the error dynamics, provided that the matrix $S_{exch}T_{\widehat{k}}B_{\widehat{k}}^{-1}E + A$ is detectable with respect to the measurements y. This statement follows directly from (26), as the first term of the right-hand side equals zero in case of a correct mode estimation. In that case the error dynamics reduce to

$$\dot{e} = (S_{exch}T_{\widehat{k}}B_{\widehat{k}}^{-1}E + A - L_{\widehat{k}}C)e. \tag{27}$$

The above equation corresponds to the error dynamics of a Luenberger observer for a linear system. When the pair $(S_{exch}T_{\widehat{k}}B_{\widehat{k}}^{-1}E + A, C)$ is observable, the observer gain matrix $L_{\widehat{k}}$ is designed by simultaneous stabilisation such that the origin is an asymptotically stable equilibrium point for the error dynamics (27).

On the other hand, when the the mode is incorrectly estimated, the first term acts as a continuous disturbance on the error dynamics. This disturbance can lead to a persistent error. This last fact stresses the importance of an accurate mode estimation, as previously discussed.

4. SIMULATION RESULTS

4.1 Toy model

This section shows the simulation results for the above mentioned observer applied to a toy model. This toy model describes the interaction between two microbial species X_1 and X_2, as commonly encountered in community and biomedical systems, such as biofilms. The underlying metabolic network describes the uptake of a substrate S. This substrate is converted to an intracellular metabolite A, which is subsequently converted to biomass of the first species X_1. Besides biomass production of X_1, a primary metabolic product P can be produced. This product serves as a substrate for the production of biomass of the second species X_2. A full overview of the metabolic network, showing metabolites and reactions, is shown in Figure 1.

A deFBA model for the toy model is constructed in accordance with the procedure outlined in section 2. The corresponding set of ODEs corresponds to the following equations:

$$\begin{aligned} \dot{S} &= -v_1(t),\ S(0) = S_0 \\ \dot{X}_1 &= v_2(t),\ X_1(0) = X_{1,0} \\ \dot{X}_2 &= v_5(t),\ X_2(0) = X_{2,0}. \end{aligned} \tag{28}$$

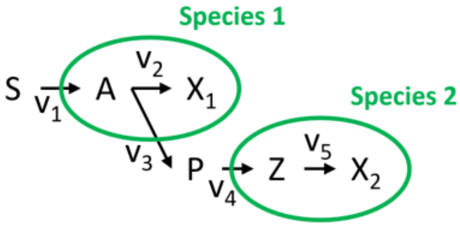

Fig. 1. A schematic overview of the metabolic network of the toy model with substrate S, intracellular metabolites A and Z, product P, and biomass components X_1 and X_2. v_i represent reaction fluxes of the respective reactions.

The amounts of the intracellular metabolites A and Z, and the product P are assumed to be in quasi steady-state. This assumption relies on the fact that, for the intracellular metabolites A and Z, intracellular dynamics are considered to be significantly faster compared to extracellular dynamics. In case of the product P, it is assumed that the production rate of species 1 equals the consumption rate of species 2. In this way, both bacterial species are described by one lumped metabolic network.

$$\begin{aligned} \dot{A} &= v_1(t) - v_2(t) - v_3(t) = 0 \\ \dot{P} &= v_3(t) - v_4(t) = 0 \\ \dot{Z} &= v_4(t) - v_5(t) = 0 \end{aligned} \tag{29}$$

The fluxes v directly depend on the state values and are obtained as the argument of a discretised optimisation problem. As objective function the integral of the sum of the total biomass amounts is taken, with as weighting factors the relative molecular weights of both biomass components:

$$\int_0^{t_f} (M_1\tilde{X}_1 + M_2\tilde{X}_2)\mathrm{d}t, \tag{30}$$

with M_1 and M_2 the relative molecular weights of X_1 and X_2 respectively.

Biomass component X_1 catalyses the uptake of substrate S. Similarly, biomass component X_2 catalyses the uptake of the product P. In accordance with equation (5), this leads to the following constraints on the fluxes v_1 and v_4:

$$v_1 \leqslant k_1X_1 \tag{31}$$

$$v_4 \leqslant k_2X_2. \tag{32}$$

No biomass composition constraints are present in the model. All reactions are irreversible and the corresponding fluxes are therefore positive. A complete overview of numerical values for all model and discretisation parameters can be found in Table 1.

4.2 Observer design

The goal of the constructed observer is to correctly predict the biomass composition i.e. the individual values for X_1 and X_2, based on noisy substrate and total biomass amount measurements. Specifically, the measurement matrix C equals

$$C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 2 \end{bmatrix}. \tag{33}$$

Gaussian white noise is added to the actual system values, obtained via a deFBA simulation, with a signal to noise

Table 1. Numerical values for model and discetisation parameters

Parameter	Value
k_1	1 h^{-1}
k_2	1 h^{-1}
M_1	1
M_2	2
t_f	10 h
N	100
h	0.1 h
S_0	4 mmol
$X_{1,0}$	10^{-2} kg
$X_{2,0}$	10^{-3} kg

Table 2. Numerical values for observer parameters

Parameter	Value
α	10
Q	$10^{-2} \times I$
$\widehat{S}_0$	7 mmol
$\widehat{X}_{1,0}$	10^{-2} kg
$\widehat{X}_{2,0}$	10^{-2} kg

Table 3. Mode overview

	Mode of the the real process?	In preliminary mode list?	Detectable?
Mode 1	Yes	Yes	Yes
Mode 2	Yes	Yes	Yes
Mode 3	Yes	Yes	No
Mode 4	Yes	Yes	No
Mode 5	No	Yes	Yes

ratio of 40 dB. A full overview of observer parameters as introduced in previous sections is given in Table 2, with $I \in \mathbb{R}^{n \times n}$ the identity matrix.

To establish the preliminary list of modes, the deFBA simulation happens three times. The three runs use half of the initial amount estimations, the actual initial amount estimations and double the initial amount estimates respectively. A total of five modes are encountered during this repeated simulation. Each mode, characterized by a unique combination of T_i and B_i^{-1}, is given a reference number. Table 3 gives an overview of the encountered modes during model simulation and observer estimation. This table also indicates three key properties of the modes: (1) Is the mode encountered during model simulation? (2) Is the mode part of the preliminary mode list? (3) Is the mode detectable? Of these five modes, four are encountered during the real model simulations. Modes 3 and 4 are not detectable and are, therefore, not part of the modes taken into account in the bank of observers. Moreover, mode 4 is described by a full zero matrix. This mode always results in a zero flux vector and represents the situation in which the substrate is completely depleted, i.e. the steady-state of a batch process.

4.3 Observer results

Figure 2 shows the state estimates and output estimates of the observer and compares them to the noiseless system values. It is observed that, after an initial transient period, both the estimated outputs and estimated states converge to the actual system values. However, from the moment the system reaches a steady-state, triggered by complete depletion of the substrate S, the state estimates for both biomass components become inaccurate. In contrast, the output estimation for the total biomass remains fairly reliable.

To explain this observation, a closer investigation of the mode estimation is required. Figure 3 shows in the top graph the modes that are encountered during the actual system simulation. The middle graph shows the estimated modes at each time point. The actual system starts in mode 1 after which it transitions to modes 2, 3 and 4 consecutively. Initially the observer mode is wrongly estimated during a transient phase. Around 1.8 h, the observer correctly estimates mode 1 to be the actual system mode. Until 6.8 h, changes in the estimated mode can be attributed to measurement noise. A smaller value for the parameter α, would have reduced the number of changes in mode estimate due to noise. However, this smaller α would also lead to a slower response in mode estimate when the actual simulation undergoes a mode switch.

After 6.8 h, the actual simulation runs into modes 3 and 4. Recall that these modes are not detectable and are therefore not considered in the observer design. Therefore, when the real system runs into these undetectable modes upon substrate depletion, it is impossible for the observer to give a correct mode estimation. The observer wrongly suggests either mode 2 or mode 5 as estimated mode. This results in less accurate state estimations for X_1 and X_2, although the output estimations remain fairly reliable with only short transient behavior upon mode switches. Also, since the substrate amount S is a direct measurement, the state estimation for S remains accurate.

The bottom part of Figure 3 shows the moving averages of the estimation error r_i. For all three modes considered in the bank of observers, the error decreases after initial transient behaviour. When the real system reaches the undetectable modes 3 and 4, the average errors increase, indicating that none of these modes is an appropriate mode estimate at this point.

5. CONCLUSION

The goal of this study was to design an observer for a set of ODEs with an embedded LP problem in the right-hand side. The system fits in the deFBA framework which is a modelling tool, used for bioprocesses, to predict substrate, product and biomass component amounts with respect to time. The embedded LP problem is the result of the discretisation of a dynamic optimisation problem which maximises biomass growth. To avoid solving this LP problem each time the right-hand side is evaluated during integration, it is converted to a set of algebraic equations, resulting in a switched system description with state-dependent switches.

A Luenberger observer has been constructed for this type of systems. Mode estimation happens by a bank of Luenberger observers, one for each detectable mode from a preliminary mode list. The optimal mode is chosen via a moving average error between estimations and measurements for each mode. The mode estimated is subsequently used in an overall Luenberger observer. The constructed observer is applied to a toy model, describing growth of two

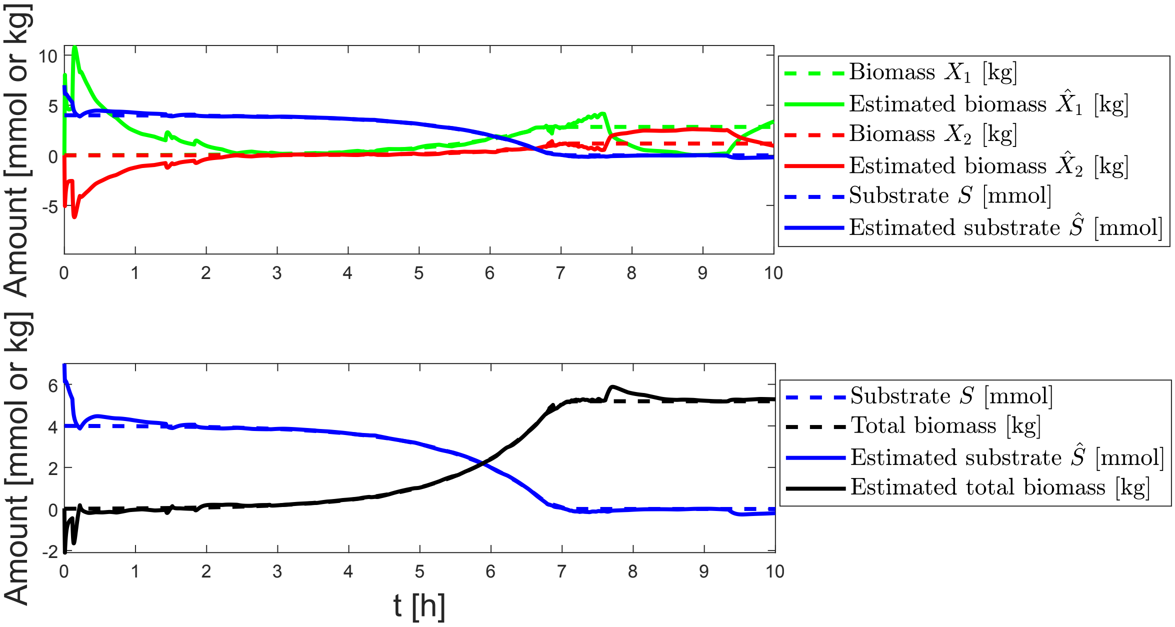

Fig. 2. (Top) State estimations (full lines) and true noiseless system values (dashed lines), (Bottom) Output estimations (full lines) and noiseless output measurements (dashed lines)

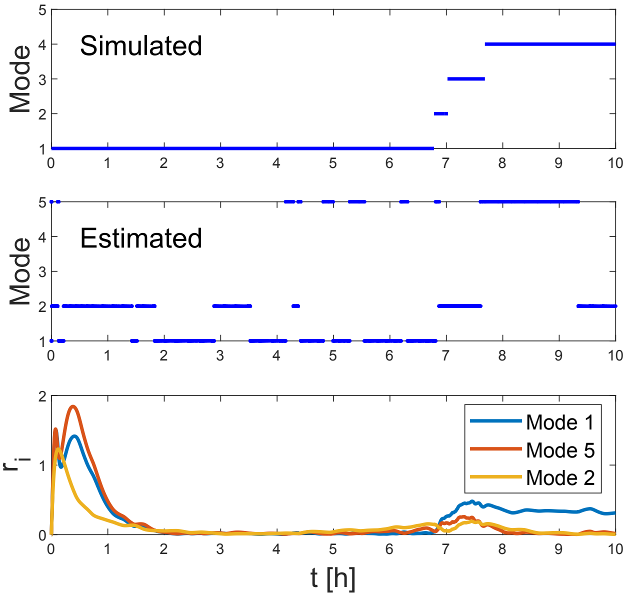

Fig. 3. (Top) Modes of the real system, (Middle) Estimated modes, (Bottom) Moving average error for observer modes

species of microorganisms. In case the real system runs into undetectable modes with respect to the measurements, the observer is not capable to accurately predict the model states. For batch processes, the steady-state value corresponds with substrate depletion. Therefore, we expect that a batch system will always run into an undetectable mode upon reaching steady-state. However, during exponential growth, a correct estimate of both biomass components is retrieved. This makes us conclude that, despite some shortcomings, the simulation results show the feasibility of the applied approach for observer design in the deFBA framework.

Future research will focus on designing mode observers based on the observable subspace of each mode such that (partly) unobservable modes can be considered during the mode estimation step. Furthermore, to make the algorithm scale better with large-scale metabolic network models, methods of fault detection will be adopted such that mode estimation should only occur whenever a fault is detected instead of evaluating for the optimal mode at each time point.

REFERENCES

[1] Bogaerts Ph. and Vande Wouwer A. (2003). Software sensors for bioprocesses. *ISA Transactions*, 42(4), 547-558.

[2] Gernaey K. V. and Lantz A. E., Tufvesson P., Woodley J. M., Sin G. (2010). Application of mechanistic models to fermentation and biocatalysis for next-generation processes. *Trends in Biotechnology.* 28(7), 346-354.

[3] Bastin G. and Dochain D. (1990). *On-line estimation and adaptive control of bioreactors.* Amsterdam: Elsevier.

[4] Reed J. L. and Palsson B. Ø. (2003). Thirteen Years of Building Constraint-Based In Silico Models of *Escherichia coli. Journal of Bacteriology.* 185(9), 2692-2699.

[5] Orth J. D., Thiele I. and Palsson B. Ø (2010). What is flux balance analysis?. *Nature Biotechnology.* 28, 245-248.

[6] Mahadevan R., Edwards J. S. and Doyle III F. J. (2002). Dynamic Flux Balance Analysis of Diauxic Growth in *Escherichia coli. Biophysical Journal.* 83(3), 1331-1340.

[7] Waldherr S., Oyarzun D. A. and Bockmayr A. (2015). Dynamic optimisation of metabolic networks coupled

with gene expression. *Journal of Theoretical Biology.* 365, 469-485.

[8] Höffner K., Harwood S. M. and Barton P. I. (2013). A reliable simulator for dynamic flux balance analysis. *Biotechnology and Bioengineering.* 110, 792-802.

[9] Harwood S. M., Höffner K. and Barton P. I. (2016). Efficient solution of ordinary differential equations with a parametric lexicographic linear program embedded. *Numerische Mathematik.* 133(4), 623-653.

[10] Gomez J. A., Höffner K. and Barton P. I. (2014). DFBAlab: a fast and reliable MATLAB code for dynamic flux balance analysis. *BMC Bioinformatics.* 15(1), 409-418.

[11] Küsters F., Trenn S. and Wirsen A. (2017). Switch-observer for switched linear systems. *IEEE 58th Annual Conference on Decision and Control (CDC).* Melbourne.

[12] Lindhorst H., Reimers A.-M. and Waldherr S. (2018). Dynamic modeling of enzyme controlled metabolic networks using a receding time horizon. *IFAC-PapersOnLine*, 51(18), 203-208.

[13] Bertsimas D. and Tsitsiklis J. (1997). *Introduction to linear optimisation.* Belmont:Athena scientific.

[14] Hautus M. L. J. (1970). Controllability and observability of linear autonomous systems. *Indagationes mathematicae (proceedings).*

[15] Alessandri A. and Coletta P. (2001). Switching observers for continuous-time and discrete-time linear systems. *Proceedings of the American Control Conference.*

[16] Löfberg J. (2004). YALMIP: A toolbox for modeling and optimisation in MATLAB. *CACSD Conference (proceedings).* Taipei, Taiwan.

[17] Sturm J. F. (1999). Using SeDuMi 1.02, a MATLAB toolbox for optimisation over symmetric cones. *optimisation methods and software*, 11(1-4), 625-653.