

# Analysis of control models based on dFBA for fed-batch bioreactors solved by interior-point methods<sup>★</sup>

Caroline S. M. Nakama<sup>\*</sup> Johannes Jäschke<sup>\*</sup>

<sup>\*</sup> Department of Chemical Engineering, Norwegian University of Science and Technology (NTNU), Trondheim, Norway (e-mail: johannes.jaschke@ntnu.no)

---

**Abstract:** Model-based control of bioprocesses based on dynamic flux balance analysis (dFBA) is an interesting strategy due to the possibility of accounting for wider ranges of cellular behaviour and operation conditions than the one based on unstructured models. This type of control models are bi-level optimization problems, since dFBA comprises of differential equations and a linear programming (LP) model. They can be solved with a nonlinear programming (NLP) solver by replacing the LP model with its first-order optimality conditions (KKT conditions) and discretizing the differential equations. When following this approach, it is important to carefully design the optimization problem considering properties of the selected solver. In this work, we show that we can formulate and solve model-based control models with dFBA for fed-batch bioprocesses by using the KKT conditions of the LP model, relaxing the inequality-multiplier complementarities that arise from these conditions and employing a line search interior point solver. We demonstrate this process on a case study that seeks to maximize growth of *Escherichia coli* on glucose and discuss fundamental steps that one should carefully evaluate, such as constraint dependence, handling of complementarities and model initialization.

*Keywords:* Bioprocess control, dynamic flux balance analysis, mathematical programming with complementarity constraints, bi-level optimization, dynamic optimization

---

## 1. INTRODUCTION

Model-based control of bioprocesses is an important strategy that can improve production efficiency and satisfaction of quality requirements (Pörtner et al., 2017). Implementation of this type of control often uses unstructured models, which are based on specific growth rate and fixed yield parameters, and are usually only valid for narrow ranges of process conditions and cellular behaviour (Jabarivelisdeh et al., 2020). Dynamic flux balance analysis (dFBA) is a modeling technique based on the metabolic network of the microorganism and can account for changes in the active metabolism of the cell. It consists of a set of ordinary differential equations (ODEs) describing the mass balance of substances external to the microorganism, and a static linear programming (LP) model that determines the flux distribution of the metabolic network (Mahadevan et al., 2002). Although cellular metabolism is subject to complex nonlinear dynamics, dFBA is able to describe a wider range of behaviours and may be seen as a compromise between mathematical complexity and model comprehensiveness, as opposed to complex kinetic models.

In the context of bioprocess model-based control, employing the dFBA model leads to a dynamic bi-level optimization model. Two methods have been mainly explored, one is the direct approach (DA), which calls an LP solver for each step calculation of the solver for the outer problem

(Jabarivelisdeh et al., 2020), while the other replaces the LP model with its first-order optimality conditions, i.e., the Karush-Kuhn-Tucker (KKT) conditions (Hjersted and Henson, 2006; Chang et al., 2016). LP solvers are efficient and, therefore, the DA approach can be computationally fast. However, integrating the solver for the outer optimization with the LP solver can potentially generate wrong results, for example, if an attempt step calculation returns an infeasible point (Ploch et al., 2020). Using the KKT conditions does not present this problem, but may be challenging to solve due to the inequality-multiplier complementarities.

When the corresponding KKT conditions substitute the LP model, the optimization problem can be discretized and nonlinear programming (NLP) solvers can be used. An issue associated with this problem formulation is that the optimization model is often ill-posed. Solvers that use active-set selection to handle inequalities, such as CONOPT (Drud, 1985), can be well suited to deal with that; however, they can become inefficient when the numbers of inequalities and degrees of freedom are large. Line search interior point solvers, e.g. IPOPT (Wächter and Biegler, 2006), can efficiently handle many degrees of freedom and inequalities, but convergence can be compromised when the model is ill-posed (Wan and Biegler, 2017). Therefore, formulating a well-posed model and understanding limitations of the NLP solver used is important.

---

<sup>\*</sup> The authors acknowledge financial support from the Norwegian Research Council through FRIPRO Project SensPATH.

Since industrial bioprocess are predominantly operated as batch and fed-batch, the effort in developing model-based control of bioprocess is directed to these types of bioreactors. Two strategies can be used, open-loop and closed-loop approaches, and both have been reported in literature (Jabarivelisdeh et al., 2020; Chang et al., 2016; Hjersted and Henson, 2006). While the optimization model resulting from the open-loop control approach can be solved off-line, a closed-loop control model is better suited to handle disturbances and model mismatch (Tebhani et al., 2008).

The contribution of this work is the analysis and discussion of the mathematical implementation of optimal control models formulated from the first-order optimality conditions of dFBA models. We consider both open and closed-loop control approaches, formulate the optimization model as a single NLP problem and solve it using IPOPT. Although there are few control applications of this approach reported in literature (Hjersted and Henson, 2006; Chang et al., 2016), not much attention is directed to how to formulate and solve these control models and the challenges they present, specifically when interior point methods are employed. We focus on these aspects and discuss how we can deal with common straightforward issues associated with them.

## 2. OPTIMAL CONTROL BASED ON DYNAMIC FLUX BALANCE ANALYSIS

In this section we show how we can formulate an optimal control model for fed-batch bioreactors using dFBA to describe the metabolism of the microorganism and the dynamics of the process. The optimization model for the process is the same for both open-loop and closed-loop strategies; the difference lies on how the model is used to control the process. In the open-loop approach, the optimization problem is solved and the trajectories obtained for the manipulated variables are implemented in the process. For the closed-loop strategy, we consider a shrinking horizon, which starts with the length of the total batch time,  $T \in \mathbb{R}$ , and is discretized into  $n_T \in \mathbb{N}$  finite elements. As the process progresses, at each time step  $t_i \in \mathbb{R}$  of length  $h_i = T/n_T$ , the optimization model is solved and the values of the manipulated variables at the current step are implemented, while the remaining future trajectories are discarded.

The optimization model is of the form

$$\max_u \Phi(x(t), u(t), v(t)) \quad (1a)$$

$$\text{s.t.} \quad \frac{dx}{dt} = f(x(t), u(t), v(t)) \quad (1b)$$

$$v_B(t) = q(x(t)) \quad (1c)$$

$$v(t) \in \underset{w}{\text{argmin}} \{-c^T w | Aw = 0, v^{\text{lb}} \leq w \leq v^{\text{ub}}\} \quad (1d)$$

where  $u(\cdot) \in \mathbb{R}^{n_U}$  is a vector with  $n_U \in \mathbb{N}$  manipulated variables,  $x(t) \in \mathbb{R}^{n_S}$  is a vector with  $n_S \in \mathbb{N}$  state variables, which usually correspond to the volume of the reactor and external concentrations,  $v(t) \in \mathbb{R}^{n_R}$  is a vector with  $n_R \in \mathbb{N}$  internal fluxes in the metabolic network, which correspond to the algebraic variables,  $v_B(t) \in \mathbb{R}^{n_B}$  is a sub-vector with  $n_B \in \mathbb{N}$  boundary

fluxes, i.e., those that exchange matter with the media,  $c \in \mathbb{R}^{n_R}$  is a weight vector for the fluxes that determine cellular growth,  $A \in \mathbb{R}^{n_M \times n_R}$  is the stoichiometry matrix of internal metabolites,  $v^{\text{lb}} \in \mathbb{R}^{n_R}$  and  $v^{\text{ub}} \in \mathbb{R}^{n_R}$  are, respectively, vectors with lower and upper bounds for the fluxes,  $\Phi : \mathbb{R}^{n_S} \times \mathbb{R}^{n_U} \times \mathbb{R}^{n_R} \rightarrow \mathbb{R}$  is the economic objective function,  $f : \mathbb{R}^{n_S} \times \mathbb{R}^{n_U} \times \mathbb{R}^{n_R} \rightarrow \mathbb{R}^{n_S}$  is a vector function representing the mass balance of the state variables, and  $q : \mathbb{R}^{n_S} \rightarrow \mathbb{R}^{n_B}$  is a vector function that governs the boundary exchanges between the microorganism and the media. The set of constraints corresponds to the dFBA model, which includes an LP model, making (5) a dynamic bi-level optimization problem.

Constraint (1d) corresponds to the flux balance analysis model (Varma and Palsson, 1994), which formulates an LP using the stoichiometry matrix of the metabolic network to characterize the fluxes and, most commonly, maximizes biomass production, i.e., growth. The metabolism of cells usually have several redundant pathways, which causes (1d) the minimum to be non-unique. To circumvent that, some methods have been proposed so that a single solution is selected (Ploch et al., 2020; Scott et al., 2018; Gomez et al., 2014). Here, we will follow the work of Ploch et al. (2020) and add a quadratic term of the form  $\frac{1}{2}w^T Q w$  to the objective function of (1d), where  $Q \in \mathbb{R}^{n_R \times n_R}$  is given by  $\epsilon I$ , being  $I \in \mathbb{R}^{n_R \times n_R}$  the identity matrix and  $\epsilon \in \mathbb{R}_{>0}$ . Since  $Q$  is positive definite, (1d) becomes a convex quadratic program (QP) and, for small enough  $\epsilon$ , the optimal solution of the QP approximates to the solution of the original LP (Ploch et al., 2020).

Since we seek to solve this problem using IPOPT, we need to reformulate the dynamic bi-level optimization model into an NLP problem, as interior point solvers cannot directly handle differential algebraic system of equations (DAEs) or LP/QP models as constraints. Considering that, we replace (1d) with its first-order optimality conditions and discretize the model using orthogonal collocation with  $n_T$  finite elements and  $n_C \in \mathbb{N}$  collocation points Biegler (2010). The problem is then described by

$$\max_u \Phi(x, u, v) \quad (2a)$$

$$\text{s.t.} \quad x(t_i) = x(t_{i-1}) + \sum_{k=1}^{n_C} h_i \omega_k f(x(t_{ik}), u(t_i), v(t_{ik})) \quad (2b)$$

$$v_B(t_{ij}) = q(x(t_{ij})) \quad (2c)$$

$$A^T \lambda_{ij} - y_{ij} + z_{ij} - c + \epsilon v(t_{ij}) = 0 \quad (2d)$$

$$Av(t_{ij}) = 0 \quad (2e)$$

$$z_{ij}(v^{\text{ub}} - v(t_{ij})) = 0 \quad (2f)$$

$$y_{ij}(v(t_{ij}) - v^{\text{lb}}) = 0 \quad (2g)$$

$$v^{\text{ub}} - v(t_{ij}) \geq 0 \quad (2h)$$

$$v(t_{ij}) - v^{\text{lb}} \geq 0 \quad (2i)$$

$$z_{ij}, y_{ij} \geq 0 \quad (2j)$$

$$\text{for } i = 1, \dots, n_T; \quad j = 1, \dots, n_C$$

where  $t_{ik} = t_{i-1} + h_i \tau_k$  with  $\tau_k \in \mathbb{R}$  corresponding to interpolation points within the finite element,  $\omega_k \in \mathbb{R}$  is a quadrature weight that depends on the method and number of collocation points,  $\lambda_{ij} \in \mathbb{R}$  is the Lagrange multiplier corresponding to the equality constraints of the

LP/QP for  $t_{ij}$ , and  $y_{ij} \in \mathbb{R}$  and  $z_{ij} \in \mathbb{R}$  are the Lagrange multipliers for the inequalities representing the lower and upper bounds for  $v(t_{ij})$  respectively. From this point onward, we drop the explicit notation for time-dependency and define, for example,  $x_i := x(t_i)$  and  $x_{ij} := x(t_{ij})$ , the latter being the value of  $x$  in the  $j^{\text{th}}$  collocation point within the  $i^{\text{th}}$  finite element.

### 2.1 Model analysis

When using line search interior point solvers, it is important to assess whether the optimization model has dependent equality constraints, since they may result in convergence issues or even failure, and address them if possible. These dependencies can be categorized into two types: global and local. In the former, the constraints are redundant and, therefore, dependent at every iteration of the solver. The latter can be an inherent problem of non-linear problems as the linearized constraints can become dependent at some iterates depending on the point being evaluated (Wan and Biegler, 2017).

Even though global dependencies do not change the optimal solution, they are associated with naive modeling and should be removed (Wan and Biegler, 2017). When working with models based on the stoichiometry matrices of metabolic networks, we can have globally dependent constraints when, for example, all metabolites involved in a closed cycle are present. Ideally we should identify these groups of metabolites and remove one of them, as any result involving the removed metabolite can be calculated from the others. The presence of global dependent constraints in this context is further discussed in the Results and discussion section demonstrated on the case study.

Employing the KKT conditions of a QP model as constraints, as in (2), results in the optimization model becoming a mathematical programming with complementarity constraints. These constraints fix the relationship between two variables by enforcing that at least one of them must be at its bound. They are usually represented as

$$0 \leq a \perp b \geq 0, \quad (3)$$

where  $a, b \in \mathbb{R}^n$  with  $n \in \mathbb{N}$ , but they can also be written as  $a_i b_i = 0$  for  $i = 1, \dots, n$  or  $a^T b = 0$  with  $a, b \geq 0$ . These type of constraints are locally dependent at any feasible point (Wan and Biegler, 2017), and interior point solvers cannot directly handle this non-smoothness, and, therefore, MPCC models are usually reformulated into a NLPs. There are essentially two approaches one can follow for to derive this reformulation. One consists in relaxing the complementarity constraints, e.g, by replacing (3) with  $a_i b_i \leq \epsilon$  for  $i = 1, \dots, n$ , and a small constant  $\epsilon \in \mathbb{R}$ , while the other uses the penalty approach, which removes  $a^T b = 0$  from the constraint set and adds a penalty term of the form  $\pi a^T b$  to the objective function, where  $\pi \in \mathbb{R}$  is a large enough positive constant (Leyffer et al., 2006).

Baumrucker et al. (2008) discuss different strategies to solve MPCCs and conclude that IPOPT performs well with both reformulation approaches and Leyffer et al. (2006) prove convergence for both methods combined with an interior point algorithm. While the relaxation approach guarantees that the complementarity is met to a certain precision, the penalty reformulation is more general and

gives more flexibility to the model. Whether one approach is better suited for employing dFBA as the control model will also be discussed in the Results and discussion section.

## 3. CASE STUDY

We base our discussion on the case study of a fed-batch process in which we wish to maximize the mass of *Escherichia coli* at the end of the cultivation time. This problem is presented in Scott et al. (2018) with an open-loop formulation to evaluate their proposed method of handling dFBA problems as DAEs or implicit ODEs. The metabolic network of the core metabolism consists of  $n_m = 72$  internal metabolites and  $n_r = 95$  fluxes (Orth et al., 2010). The dFBA model is given by

$$\frac{dV}{dt} = F \quad (4a)$$

$$\frac{dX}{dt} = v_B X - X \frac{F}{V} \quad (4b)$$

$$\frac{dS}{dt} = 0.18 v_S X - (S_F - S) \frac{F}{V} \quad (4c)$$

$$v_S = -v_{S,\max} \frac{S}{S + K_S + S^2/K_I} \quad (4d)$$

$$v \in \operatorname{argmin}_w \{-c^T w | Aw = 0, v^{\text{lb}} \leq w \leq v^{\text{ub}}\} \quad (4e)$$

where  $V \in \mathbb{R}$  is the culture volume in L,  $X \in \mathbb{R}$  is biomass concentration in g/L,  $S \in \mathbb{R}$  is glucose concentration in g/L,  $F \in \mathbb{R}$  is the feed rate in L/h and glucose concentration  $S_F \in \mathbb{R}$ ,  $v_S \in \mathbb{R}$  and  $v_B \in \mathbb{R}$  are entries of  $v$  and correspond, respectively, to the glucose uptake rate and the growth rate in mmol/(gDW·h),  $v_{S,\max} \in \mathbb{R}$  is the maximum glucose uptake rate, and  $K_S \in \mathbb{R}$  and  $K_I \in \mathbb{R}$  are affinity and inhibition constants for the substrate in g/L. Equation (4d) is a Michaelis-Menten equation describing glucose uptake taking substrate inhibition into account; this is the only boundary exchange considered.

To build the optimization model for the control problem, we discretize (4) and replace the LP problem with its KKT conditions, as done in (2), resulting in

$$\max_F \Phi = (XV)|_{t_f} \quad (5a)$$

$$\text{s.t. } V_i = V_{i-1} + \sum_{k=1}^{n_C} h_i \omega_k F_i \quad (5b)$$

$$X_i = X_{i-1} + \sum_{k=1}^{n_C} h_i \omega_k \left( v_{Bij} X_{ij} - X_{ij} \frac{F_i}{V_{ij}} \right) \quad (5c)$$

$$S_i = S_{i-1} + \sum_{k=1}^{n_C} h_i \omega_k \left( 0.18 v_{Sij} X_{ij} - (S_F - S_{ij}) \frac{F_i}{V_{ij}} \right) \quad (5d)$$

$$v_{Sij} = -v_{S,\max} \frac{S_{ij}}{S_{ij} + K_S + S_{ij}^2/K_I} \quad (5e)$$

Eq. (2d) – (2j)

for  $i = 1, \dots, n_T$ ;  $j = 1, \dots, n_C$ .

For this case study, we consider a batch time of 12 hours, and the initial values for the dynamic variables are  $V_0 = 1$  L,  $X_0 = 1$  gDW/L, and  $S_0 = 2$  g/L. For simulating the process, the parameter values are  $S_F = 100$  g/L,  $v_{S,\max} = 10$  mmol/(gDW·h),  $K_S = 1$  g/L, and  $K_I = 10$

g/L. Process simulation is also performed using IPOPT with the objective function containing only the penalty terms for the complementarities. For the control model, we introduce model mismatch by reducing  $K_I$  by 15 %, which is a small difference just enough for the open- and closed-loop formulations to not overlap.

#### 4. METHODOLOGY

For this work, we analyze two types of control problems, open-loop and closed-loop models. The former is the basis for studying the influence of globally dependent constraints on convergence of the model, while the latter model is employed to analyze the influence of the reformulation approach used to solve MPCC models with NLP solvers. Since dependent constraints result in ill-conditioned linear systems during the step calculation in line search algorithms, solvers need a way of dealing with this situation. We discuss how the regularization method used in IPOPT can influence convergence when common redundancies in models based on metabolic networks are present.

Constraints (2f)-(2j) in the optimization model (5) describe the complementarity relation between the inequalities and the corresponding Lagrange multipliers of the QP in the dFBA model (1d). We apply both reformulations discussed in Section 2.1, the penalty-term and the relaxation approaches, to turn (5) into an NLP. For that, we can define two slack variables for the inequalities,  $sl_{ij}^{up} := v^{ub} - v_{ij}$  and  $sl_{ij}^{lo} := v_{ij} - v^{lb}$ . The penalty-term approach rewrites (5) as

$$\max_F \Phi = (XV)|_{t_f} - \pi \sum_{i=1}^{i=n_T} \sum_{j=1}^{j=n_C} (sl_{ij}^{up} z_{ij} + sl_{ij}^{lo} y_{ij}) \quad (6a)$$

Eq. (5b) – (5e)

Eq. (2d), (2e), (2h) – (2j)

where  $\pi \in \mathbb{R}$  is a weight parameter for the penalty term. The relaxation approach relaxes the equality constraints (2f) and (2g) turning them into inequalities, leading to

$$\max_F \Phi = (XV)|_{t_f} \quad (7a)$$

$$sl_{ij}^{up} z_{ij} \leq \varepsilon \quad (7b)$$

$$sl_{ij}^{lo} y_{ij} \leq \varepsilon \quad (7c)$$

Eq. (5b) – (5e)

Eq. (2d), (2e), (2h) – (2j)

for  $i = 1, \dots, n_T$ ;  $j = 1, \dots, n_C$ .

where  $\varepsilon \in \mathbb{R}$  is a small threshold.

When solving the problems posed by the open-loop and the closed-loop approaches, the main difference here is that, for the closed-loop method, we repeatedly solve the optimization model in (5) at each time step with a shrinking horizon during the process, while the open-loop approach requires it to be solved only once beforehand. Because of that, closed-loop problems need to be solved fast enough and accurate enough, which depends on the process. We discuss the influence of those reformulation approaches for MPCC on these aspects for this bioprocess control problem with a closed-loop setting.

Implementation of both open-loop and closed-loop control models was done in Julia with JuMP (Dunning et al.,

2017) as mathematical programming language, and using  $n_T = 12$  finite elements and  $n_C = 3$  collocation points based on Radau roots were used. The open-loop problem was solved using the penalty-term reformulation.

#### 5. RESULTS AND DISCUSSION

In this section we present the results of both control models applied to the case study described in the previous sections. First, it is important to emphasize that an MPCC problem reformulated into an NLP is nonconvex; therefore, all results presented here are local solutions.

We begin by showing that globally dependent equality constraints can be present in dFBA models and that they can influence convergence based on the results of the open-loop control model. Then, we analyze the role of the most common reformulation approaches in handling MPCC models on the closed-loop control problem and discuss the importance of a good initialization for the variables of the optimization model in this context.

##### 5.1 Globally dependent constraints

The stoichiometry matrix for the *E. coli* core metabolism considered in the case study is  $72 \times 95$ ; however, its rank is 67. This means that 5 rows are redundant, i.e., there are 5 metabolites that any result associated to them can be derived from other metabolites. For example, adenosine triphosphate (ATP), adenosine diphosphate (ADP) and adenosine monophosphate (AMP) are well-known molecules involved in the energy cycle of a cell, being ATP responsible for providing energy to processes in the metabolism. When energy is consumed by a process, ATP loses one or two phosphate atoms and becomes ADP or AMP respectively. Since they are simultaneously present in reactions involving energy exchange, they result in linear dependent rows in the stoichiometry matrix. If the complete matrix is used in the dFBA model, those rows give rise to globally dependent constraints, which, as discussed in Section 2.1, could negatively influence convergence of IPOPT.

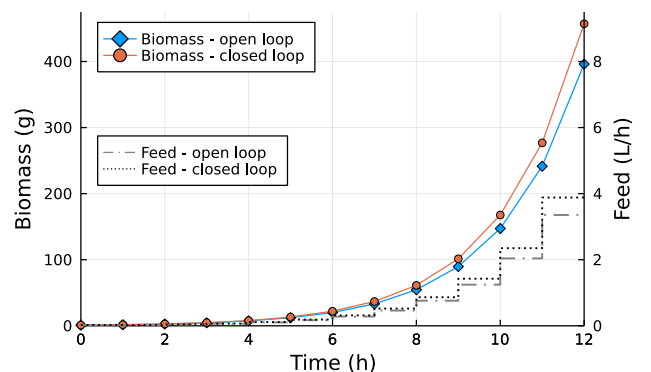


Fig. 1. Results for the open-loop and closed-loop control problems with model mismatch in  $K_I$ .

We compared the results for the open-loop control problem using the complete stoichiometry matrix,  $A$ , and the reduced full-rank matrix,  $A_r$ , after ADP and other four redundant metabolites were removed. Because the starting

point and weighting parameter  $\pi$  can influence convergence, we used 3 different initial guesses and varied  $\log_{10} \pi$  from 0 to 3 for problems built with both  $A$  and  $Ar$ . Almost every problem converged to the open-loop model results in Figure 1, except two problems formulated with the complete matrix  $A$ ; one failed to converge and the other converged to a different solution with 329 g of biomass at the end of the batch. The main difference between using  $A$  and  $Ar$  was that, when the former was employed, regularization to correct the curvature of the solution was required. These results corroborate the discussion in Wan and Biegler (2017). Primal-dual interior point methods, such as the one implemented in IPOPT, use inefficient regularization strategies to handle ill-conditioned KKT systems originated from dependent constraints that may result in early termination or poor convergence. All problems formulated with  $Ar$  required more iterations, however, they converged to a point in the solution space with positive curvature.

There are several aspects that influence convergence of non-convex NLPs to a local solution. For example, in this case, since the two problems that did not converge to the results for the open-loop model in Figure 1 were initialized with several variables at their bound, changing the initial value of IPOPT’s barrier parameter would alter the actual initial point, which converges to the same results as the other problems. The influence of the barrier parameter is further discussed in the last part of the following subsection. However, it is good practice to remove globally dependent constraints since that leads to a more robust formulation of the NLP model requiring less manual tuning.

## 5.2 MPCC reformulation

Both the penalty-term and relaxation formulations for handling the complementarity constraints from the dFBA model were able to solve the closed-loop problems and they converged essentially to the same local minimum, from which the main results are shown in Figure 1, for all weighting parameter  $\pi$  and threshold  $\varepsilon$  values considered. As expected, since we introduced a mismatch in the control model, the closed-loop MPC approach is able to reach larger biomass production at the end of the batch. The results presented here were obtained with the reduced full-rank stoichiometry matrix.

For the optimization model using the penalty-term approach for reformulating the complementarities, we considered 4 values for the weighting parameter  $\pi$ , 1, 10, 100 and 1000, and, for the model with the relaxed constraints, the values used as threshold were  $10^{-4}$ ,  $10^{-5}$ ,  $10^{-6}$ , and  $10^{-7}$ . Figure 2 presents the calculated values for the complementarities and time consumed to solve the NLP problems at each time step for both reformulation approaches.

Consider a vector  $d \in \mathbb{R}^{2 \cdot n_R \cdot n_T \cdot n_C}$  with all the complementarity expressions given by (2f) and (2g); the top plot in Figure 2 shows the maximum entry in  $d$  from the optimization model solved at each time step. Since the optimization model always converged to a local minimum, when the relaxation reformulation is used, the relaxed constraints are, as expected, always lower than the adopted threshold value. However, for the penalty-term reformulation,  $d$  only

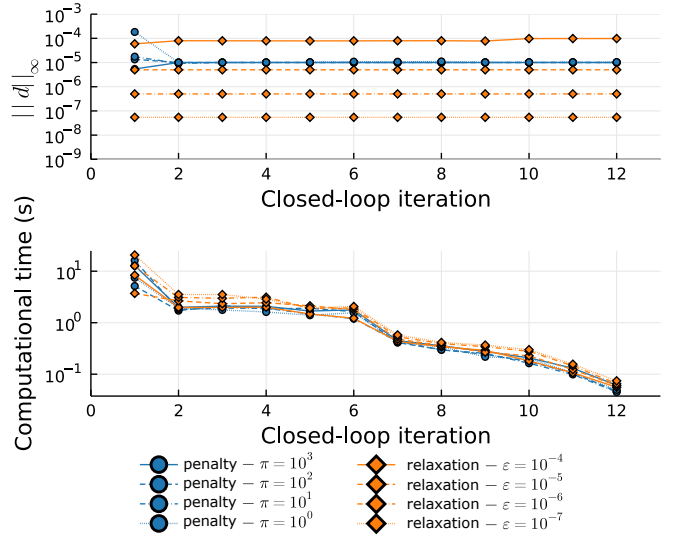


Fig. 2. Top plot: maximum complementarity value of the model that is solved at each time step. Bottom plot: run time for solving the model at each time step.

varies with the value of the weighting parameter  $\pi$  in the first optimization model and, during the process, the maximum complementarity is kept at  $10^{-5}$ .

The bottom plot of Figure 2 shows that the solving times of the models are very similar, with the relaxation approach showing a slightly larger variation with threshold  $\varepsilon$ . Since the closed-loop control model is implemented with a shrinking horizon, the optimization model solved at each time step decreases, requiring less time to be solved, as it can be seen by the trend in the plot.

These results show that both reformulation approaches can satisfactorily be used for the bioprocess control problem in this work. They resulted in the same trajectory for the manipulated variable and available biomass at the end of the batch regardless the value considered for  $\pi$  or  $\varepsilon$ . In addition, in both cases, the reactions rates that were not at a corresponding boundary had the same optimal values; while the remaining entries in  $v$  varied slightly with the complementarity tolerance, they could easily be identified. Regarding the solve time, bioprocesses commonly present slow dynamics, and, for this case, the longest run time to solve an optimization model was about 15 seconds, which is a negligible delay for an one-hour interval.

Another aspect that is worth discussing is the importance of model initialization, i.e., from which point NLP solvers start. Starting values usually have great influence on convergence, so it is important to have a good initial guess for all variables. To illustrate this, Figure 3 shows the run time for solving the optimization model at each time step without providing a starting point to the solver, which uses its default values. In every case, the solver was able to converge to the closed-loop model results shown in Figure 1; however, run time was greatly increased, especially when the relaxation reformulation was used.

For closed-loop control models, we can use the results from the previous iteration as starting point. When implementing a shrinking horizon, at each time step, the model is solved from the current time point  $t_i$  until the end of the

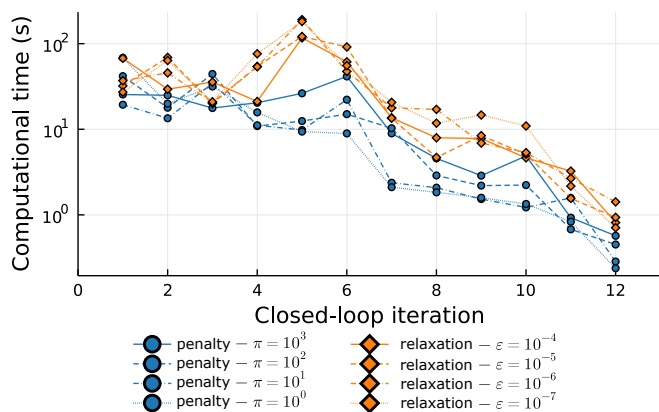


Fig. 3. Run time for solving the model at each time step without careful initialization.

batch at  $t_F$ ; the results from  $t_{i+1}$  to  $t_T$  can be saved to be used as initial guess for the optimization model in the next time step. This starting point is usually close to the solution; however, for interior point solvers, issues can arise when variables start at their bounds. To deal with variable bounds, such as  $x \geq 0$ , IPOPT uses a barrier term in the objective function of the form  $\mu \log x$ , where  $\mu \in \mathbb{R}$  is the barrier parameter, which starts at  $10^{-1}$  by default and gradually decreases as the calculation progresses. If  $x$  is set to start at 0, IPOPT changes its value to start within the barrier, so it can decrease gradually with  $\mu$ . When starting close to a solution that contains variables at their bounds, they are changed by the solver and the starting point can be altered to a "bad" initial guess. In our case study, when the penalty-term reformulation was used with  $\pi = 10^2$ , the solver failed to converge when starting from the results from the previous time step. This issue may be handled by decreasing the initial value of  $\mu$ ; for the results presented in this section,  $\mu$  was set to start at  $10^{-6}$ .

## 6. CONCLUSION AND FUTURE WORK

We have shown that model-based control of bioprocesses employing dFBA can be successfully formulated and solved by line search interior point solvers. However, it is important to understand its limitations and carefully design the problem to avoid issues that may arise. Both reformulation approaches are suitable; while the relaxation method provides better control in terms of complementarity satisfaction, the penalty-term approach can converge faster when a good initial guess is not available. For this case study, the sole goal was maximizing biomass and the active metabolism does not change during the process. For future work, we will investigate the challenges associated with changes in the active set of constraints, as well as expanding this application to genome-scale metabolic networks and exploring multi-objective dFBA formulations. As dFBA models of genome scale are extremely large, we expect the issues discussed in this paper to be more pronounced. In particular, initializing the solver, and handling ill-conditioned problems will become more difficult.

## REFERENCES

Baumrucker, B., Renfro, J.G., and Biegler, L.T. (2008). MPEC problem formulations and solution strategies

- with chemical engineering applications. *Comp. Chem. Eng.*, 32(12), 2903–2913.
- Biegler, L.T. (2010). *Nonlinear programming: concepts, algorithms, and applications to chemical processes*. SIAM.
- Chang, L., Liu, X., and Henson, M.A. (2016). Nonlinear model predictive control of fed-batch fermentations using dynamic flux balance models. *J. Process Control*, 42, 137–149.
- Drud, A. (1985). CONOPT: A grg code for large sparse dynamic nonlinear optimization problems. *Math. Program.*, 31(2), 153–191.
- Dunning, I., Huchette, J., and Lubin, M. (2017). Jump: A modeling language for mathematical optimization. *SIAM Rev.*, 59(2), 295–320.
- Gomez, J.A., Höffner, K., and Barton, P.I. (2014). Df-balab: a fast and reliable matlab code for dynamic flux balance analysis. *BMC Bioinform.*, 15(1), 1–10.
- Hjersted, J.L. and Henson, M.A. (2006). Optimization of fed-batch *saccharomyces cerevisiae* fermentation using dynamic flux balance models. *Biotechnol. Progr.*, 22(5), 1239–1248.
- Jabarivelisdeh, B., Carius, L., Findeisen, R., and Waldherr, S. (2020). Adaptive predictive control of bioprocesses with constraint-based modeling and estimation. *Comp. Chem. Eng.*, 135, 106744.
- Leyffer, S., López-Calva, G., and Nocedal, J. (2006). Interior methods for mathematical programs with complementarity constraints. *SIAM J. Optim.*, 17(1), 52–77.
- Mahadevan, R., Edwards, J.S., and Doyle III, F.J. (2002). Dynamic flux balance analysis of diauxic growth in *escherichia coli*. *Biophys. J.*, 83(3), 1331–1340.
- Orth, J.D., Fleming, R.M., and Palsson, B.Ø. (2010). Reconstruction and use of microbial metabolic networks: the core *escherichia coli* metabolic model as an educational guide. *EcoSal plus*, 4(1).
- Ploch, T., von Lieres, E., Wiechert, W., Mitsos, A., and Hannemann-Tamás, R. (2020). Simulation of differential-algebraic equation systems with optimization criteria embedded in modelica. *Comp. Chem. Eng.*, 140, 106920.
- Pörtner, R., Barradas, O.P., Frahm, B., and Hass, V.C. (2017). Advanced process and control strategies for bioreactors. In *Current Developments in Biotechnology and Bioengineering*, 463–493. Elsevier.
- Scott, F., Wilson, P., Conejeros, R., and Vassiliadis, V.S. (2018). Simulation and optimization of dynamic flux balance analysis models using an interior point method reformulation. *Comput. Chem. Eng.*, 119, 152–170.
- Tebhani, S., Dumur, D., and Hafidi, G. (2008). Open-loop optimization and trajectory tracking of a fed-batch bioreactor. *Chem. Eng. Process.: Process Intensif.*, 47(11), 1933–1941.
- Varma, A. and Palsson, B.O. (1994). Metabolic flux balancing: basic concepts, scientific and practical use. *Bio/technol.*, 12(10), 994–998.
- Wächter, A. and Biegler, L.T. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Program.*, 106(1), 25–57.
- Wan, W. and Biegler, L.T. (2017). Structured regularization for barrier nlp solvers. *Comput. Optim. Appl.*, 66(3), 401–424.