DTU Library

Economic Optimal Control of a U-loop Bioreactor using Simultaneous Collocation-based Approaches

Ritschel, Tobias K. S.; Boiroux, Dimitri; Nielsen, Marcus Krogh; Huusom, Jakob Kjøbsted; Jørgensen, Sten Bay; Jørgensen, John Bagterp

Link back to DTU Orbit

Economic Optimal Control of a U-loop Bioreactor using Simultaneous Collocation-based Approaches

Tobias K. S. Ritschel, Dimitri Boiroux, Marcus Krogh Nielsen, Jakob Kjøbsted Huusom,
Sten Bay Jørgensen, John Bagterp Jørgensen

Abstract— In this paper, we consider economic optimal control of single-cell protein (SCP) production in a U-loop reactor. The model of the U-loop reactor contains both ordinary and partial differential equations. Consequently, the optimal control problems are large-scale. The optimal operating profile for the SCP production is an unstable attractor. Therefore, we consider two simultaneous collocation-based approaches for solving the optimal control problems. We implement these two approaches in C, and we use IPOPT to solve the involved nonlinear program (NLP). Finally, we present a performance study that demonstrates the feasibility of solving economic optimal control problems that involve the U-loop reactor in real-time.

I. INTRODUCTION

The objective of optimal control is to compute an open-loop control strategy that optimizes a performance measure which either represents 1) the economics of the dynamic process or 2) the deviation of the process outputs from pre-defined setpoints. The solution of optimal control problems is relevant to model predictive control algorithms which use the moving horizon optimization principle to compute a closed-loop feedback control strategy, i.e. they solve a sequence of open-loop optimal control problems [1]. Such applications involve strict computational requirements. It is particularly challenging to meet such requirements if the process involves a large number of state variables. This is often the case when the model arises from the discretization of partial differential equations, e.g. oil reservoir fluid flow in porous media [2], [3], or from processes that involve several interconnected units, e.g. distillation columns that consist of several trays [4]. Furthermore, the solution of optimal control problems can also be used for offline analysis of process control strategies for transient operating conditions, e.g. plant startup. The computational requirements are less strict in such applications.

A. Numerical solution of optimal control problems

There exist several approaches for numerical solution of optimal control problems. Direct methods, i.e. single-shooting, multiple-shooting [5], [6], and collocation-based

approaches [4], are commonly used for real-life applications [1]. These methods transcribe the infinite-dimensional optimal control problem to a finite-dimensional nonlinear program (NLP) that can be solved using numerical optimization algorithms [7]. In single-shooting, the solution of the dynamical constraints is nested into the solution of the NLP which leads to a small-scale NLP. In collocation-based approaches, the discretized dynamical constraints are incorporated directly into the NLP leading to a large-scale problem. Multiple-shooting is a hybrid approach that attempts to combine the advantages of both single-shooting and collocation-based approaches. Two key advantages of multiple-shooting and collocation-based approaches over single-shooting are 1) that they are applicable to unstable systems and 2) that it is more straightforward to implement path constraints for these approaches. However, both of these approaches lead to large-scale NLPs.

B. Production of single-cell protein in a U-loop reactor

Methanotrophs are bacteria that grow on carbon sources such as methane or methanol which are cheap. The protein content of methanotrophs is high, and they can be used to produce single-cell protein (SCP) which can be used for animal feed. Consequently, SCP produced from methanotrophs can be used to sustain the growing human population. The U-loop reactor is a novel technology for producing SCP based on methanotrophs. However, it is nontrivial to operate the U-loop reactor, and in particular, the startup is challenging.

The dynamics of SCP production in a U-loop reactor have previously been modeled as a set of partial and ordinary differential equations [8]–[12]. Using this model, Olsen et al. [13] computed optimal operating points for steady-state operation. However, they did not consider the startup. In this work, we extend previous work on economic optimal control of the U-loop reactor [14] that involved the computation of economically optimal startup profiles. The key contributions of this work are that 1) we describe the numerical details of two collocation-based optimal control algorithms, 2) we provide details on computationally efficient C implementations of these algorithms which are based on the open-source software IPOPT 3.12.12, and 3) we present a performance study that demonstrates the feasibility of using the algorithms in closed-loop nonlinear model predictive control. A key aspect of SCP production in the U-loop reactor is that the optimal operating profile turns out to be an unstable attractor [14] (the system will diverge from the optimal trajectory if it is not controlled). This is the reason that we use collocation-
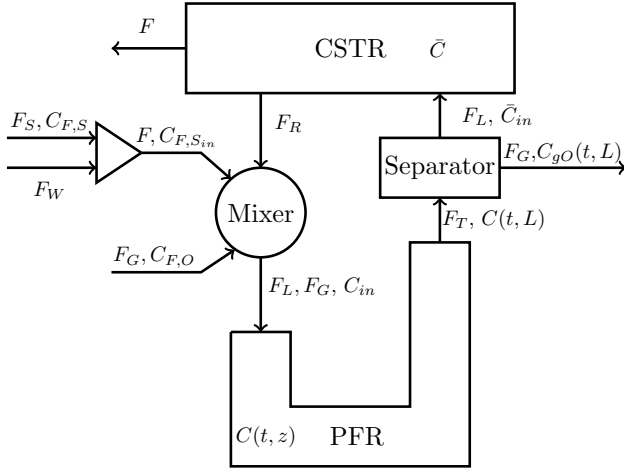
Fig. 1. A schematic of SCP production in the U-loop reactor.

based approaches. It would also be possible to use multiple-shooting.

C. Paper organization

This paper is organized as follows. In Section II, we present the model of SCP production in the U-loop reactor (using methanol as the feed). In Section III, we describe the optimal control problem that we consider as well as the two collocation-based approaches. In Section IV, we discuss the implementation of these two approaches, and in Section V, we present two numerical examples together with the performance study. Finally, we conclude this work in Section VI.

II. MATHEMATICAL MODEL

Fig. 1 shows a schematic of SCP production in the U-loop reactor [8], [9], [13], [14]. The feed substrate (methanol), feed water, feed gas (oxygen), and the recirculated mixture from the top tank are mixed together in a mixer and supplied to the inlet of the U-loop pipe. The U-loop pipe is modeled as a liquid-gas phase plug flow reactor (PFR), and the top tank is modeled as a liquid continuous stirred tank reactor (CSTR). At the end of the U-loop pipe, the gas phase is separated (ideally) from the liquid phase which enters into the top tank. Liquid (consisting of a mixture of water, biomass, substrate, and dissolved gas) is harvested from the top tank.

A. Mixing section

At the inlet of the PFR, the gas flow rate is equal to the feed gas flow rate, F_G, and the liquid flow rate, F_L, is obtained from a total static mass balance:

$$F_L = F_R + F_S + F_W. \tag{1}$$

F_R is the flow rate of recirculated liquid from the top tank through the mixer to the U-loop pipe, and F_S and F_W are the feed flow rates of substrate and water, respectively. The concentrations of biomass, X, substrate, S, dissolved oxygen in the liquid phase, O, and oxygen in the gas phase, gO, at

the inlet of the PFR are obtained using component mass balances:

$$C_{in,X} = \frac{F_R \bar{C}_X}{F_L}, \tag{2a}$$

$$C_{in,S} = \frac{F_S C_{F,S} + F_R \bar{C}_S}{F_L}, \tag{2b}$$

$$C_{in,O} = \frac{F_R \bar{C}_O}{F_L}, \tag{2c}$$

$$C_{in,gO} = C_{F,O}. \tag{2d}$$

$\bar{C}_X$, $\bar{C}_S$, and $\bar{C}_O$ are the concentrations of biomass (X), substrate (S), and oxygen (O) in the top tank, respectively.

B. The U-loop modeled as a plug-flow reactor

The phase fluxes at the inlet to the PFR, N_i, are

$$N_i(t,0) = v C_{in,i}(t), \quad i \in \{X, S, O, gO\}. \tag{3}$$

The linear velocity, v, is given by

$$v = \frac{F_L + F_G}{A}, \tag{4}$$

where A is the cross-sectional area of the U-loop pipe. The concentrations, $C_i = C_i(t, z)$, of each of the components $i \in \{X, S, O, gO\}$ along the U-loop pipe are given by the mass balances

$$\frac{\partial C_X}{\partial t} = -\frac{\partial N_X}{\partial z} + R_X, \tag{5a}$$

$$\frac{\partial C_S}{\partial t} = -\frac{\partial N_S}{\partial z} + R_S, \tag{5b}$$

$$\frac{\partial C_O}{\partial t} = -\frac{\partial N_O}{\partial z} + R_O + \frac{1}{1-\varepsilon} J_{gl,O}, \tag{5c}$$

$$\frac{\partial C_{gO}}{\partial t} = -\frac{\partial N_{gO}}{\partial z} - \frac{1}{\varepsilon} J_{gl,O}, \tag{5d}$$

for the time interval $t_0 \leq t \leq t_f$ and for $0 \leq z \leq L$. t_0 and t_f are the initial and final time, and L denotes the length of the U-loop pipe. $N_i = N_i(C_i(t, z)) = N_i(t, z)$ for $i \in \{X, S, O, gO\}$ are the fluxes of each component along the U-loop pipe, and $R_i = R_i(t, z)$ for $i \in \{X, S, O\}$ are the production rates of the components in the liquid phase. $J_{gl,O} = J_{gl,O}(t, z)$ denotes the rate at which oxygen is transferred from the gas phase to the liquid phase, and $\varepsilon = F_G/(F_G + F_L)$ is the fraction of gas in each cross section which is assumed to be constant along the U-loop pipe.

The outlet boundary conditions of the PFR are described using Danckwerts' conditions:

$$\frac{\partial C_i}{\partial z}(t, L) = 0, \quad i \in \{X, S, O, gO\}, \quad t_0 \leq t \leq t_f. \tag{6}$$

C. Gas-liquid separator

The gas and liquid are assumed to be instantaneously and perfectly separated at the outlet of the U-loop pipe, i.e. the gas phase is completely removed and the liquid phase is supplied directly to the top tank. Consequently, the concentrations at the inlet to the top tank are

$$\bar{C}_{in,i}(t) = C_i(t, L), \quad i \in \{X, S, O\}, \quad t_0 \leq t \leq t_f. \tag{7}$$

D. Model of the top tank

The liquid volume of the top tank, V, is constant, and the component mass balances for the components in the liquid phase are

$$\frac{d\bar{C}_i}{dt} = \bar{D}\left(\bar{C}_{in,i} - \bar{C}_i\right) + R_i(\bar{C}_X, \bar{C}_S, \bar{C}_O), \quad i \in \{X, S, O\},$$
(8)

for $t_0 \leq t \leq t_f$. $\bar{D} = \bar{F}/V$ is the dilution rate where $\bar{F} = F_L$. The flow rate of the product stream out of the top tank is

$$F = \bar{F} - F_R = F_L - F_R = F_S + F_W.$$
(9)

E. Convective and diffusive transport

The flux, N_i, contains a term that describes convective transport, vC_i, and one that describes diffusive transport, J_i:

$$N_i = v_i C_i + J_i, \quad i \in \{X, S, O, gO\}.$$
(10)

Fick's law is used to model the diffusive transport:

$$J_i = -D_i \frac{\partial C_i}{\partial z}, \quad i \in \{X, S, O, gO\}.$$
(11)

F. Gas-liquid transport

The rate at which oxygen is transferred from the gas phase to the liquid phase is
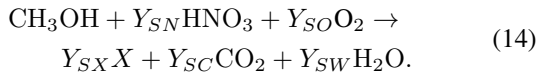
$$J_{gl,O} = (k_L a)_O(C_{sat,O} - C_O),$$
(12)

where the saturation concentration of oxygen is obtained using Henry's law and the ideal gas law:

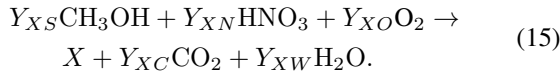$$C_{sat,O} = \frac{P_{gO}}{H_O} = \frac{RT}{M_{wO}H_O}C_{gO}.$$
(13)

Here, P_{gO} denotes the partial pressure of oxygen in the gas phase, and H_O is the Henry constant for oxygen. R is the gas constant, T denotes the temperature, and M_{wO} is the molar weight of oxygen. C_{gO} is the concentration of oxygen in the gas phase.

G. Stoichiometry and kinetics

The stoichiometric relation for the conversion of methanol (S) to biomass (X) is

$$\begin{aligned}
CH_3OH + Y_{SN}HNO_3 + Y_{SO}O_2 \rightarrow \\
Y_{SX}X + Y_{SC}CO_2 + Y_{SW}H_2O.
\end{aligned}$$
(14)

Alternatively, this relation can be formulated as

$$\begin{aligned}
Y_{XS}CH_3OH + Y_{XN}HNO_3 + Y_{XO}O_2 \rightarrow \\
X + Y_{XC}CO_2 + Y_{XW}H_2O.
\end{aligned}$$
(15)

The yield coefficients (for both formulations of the stoichiometric relation) are shown in Table I. The production rates of substrate (methanol), R_S, and oxygen dissolved in the liquid phase, R_O, are given in terms of the production rate of biomass, R_X:

$$R_S = -\gamma_S R_X, \quad \gamma_S = \frac{M_{wS}}{M_{wX}Y_{SX}},$$
(16a)

$$R_O = -\gamma_O R_X, \quad \gamma_O = \frac{M_{wO}Y_{SO}}{M_{wX}Y_{SX}}.$$
(16b)

	i	Y_{Si} [mol/mol]	Y_{Xi} [mol/mol]	M_{wi} [g/mol]	W_{Xi} [g/g]
CH_3OH	S	1.000	1.366	32.042	1.778
HNO_3	N	0.146	0.199	63.013	0.510
O_2	O	0.439	0.600	31.999	0.779
$CH_{1.8}O_{0.5}N_{0.2}$	X	0.732	1.000	24.626	1.000
CO_2	C	0.268	0.366	44.010	0.654
H_2O	W	1.415	1.933	18.015	1.414

The biomass production rate is

$$R_X = \mu(C_S, C_O)C_X,$$
(17)

where $\mu = \mu(C_S, C_O)$ is the specific growth rate which, for Methylococcus Capsulatus, is given by the Monod-Haldane expression:

$$\mu = \mu(C_S, C_O) = \mu_{max}\mu_S(C_S)\mu_O(C_O).$$
(18a)

The growth factors are

$$\mu_S(C_S) = \frac{C_S}{K_S + C_S + C_S^2/K_I},$$
(18b)

$$\mu_O(C_O) = \frac{C_O}{K_O + C_O}.$$
(18c)

H. Profit

In this work, we use the expression for the profit of the SCP production in the U-loop reactor described by Drejer et al. [14]. The profit is the difference between the value of the produced SCP (biomass) and the cost of the raw materials, and we assume that the two most expensive raw materials are the feed methanol (substrate) and oxygen, i.e. the cost of other raw materials is not included in the expression for the profit. Consequently, the profit is

$$\begin{aligned}
\phi = \int_{t_0}^{t_f} \left(p_X F(t)\bar{C}_X(t) - p_S F_S(t) - p_O F_G(t)\right) dt \\
+ p_X \left(\bar{C}_X(t_f)V + \int_0^L C_X(t_f, z)A\,dz\right) \\
- p_X \left(\bar{C}_X(t_0)V + \int_0^L C_X(t_0, z)A\,dz\right),
\end{aligned}$$
(19)

where p_X is the unit value of SCP, p_S is the unit cost of methanol (substrate), and p_O is the unit cost of oxygen.

III. OPTIMAL CONTROL

We consider optimal control problems in the form

$$\min_{[x(t)]_{t_0}^{t_f}, \{u_k\}_{k=0}^{N-1}} \phi = \phi([x(t); u(t); d(t)]_{t_0}^{t_f}),$$
(20a)

subject to

$$x(t_0) = \hat{x}_0,$$
(20b)

$$\dot{x}(t) = f(x(t), u(t), d(t)), \quad t \in [t_0, t_f],$$
(20c)

$$u(t) = u_k, \quad t \in [t_k, t_{k+1}[, \quad k = 0, \ldots, N-1,$$
(20d)

$$d(t) = \hat{d}_k, \quad t \in [t_k, t_{k+1}[, \quad k = 0, \ldots, N-1,$$
(20e)

where the objective function is in Bolza form (as is the expression for the profit (19)):

$$\phi = \phi([x(t); u(t); d(t)]_{t_0}^{t_f})$$
$$= \int_{t_0}^{t_f} \Phi(x(t), u(t), d(t))dt + \hat{\Phi}(x(t_f), u(t_f), d(t_f)).$$
$$(21)$$

Φ is the stage-cost, and $\hat{\Phi}$ is the cost-to-go. The optimal control problem (20) contains dependent decision variables (the state variables), $[x(t)]_{t_0}^{t_f}$, and independent decision variables (the manipulated inputs), $\{u_k\}_{k=0}^{N-1}$. (20b) is an initial condition for the differential equations (20c) which is obtained from the model of the U-loop reactor by discretizing the involved partial differential equations using a finite-volume discretization. (20d)-(20e) are zero-order-hold (ZOH) discretizations of the manipulated inputs, $u(t)$, and the disturbance variables, $d(t)$.

A. Temporal discretization

The dynamical system in (20c) is stiff. Therefore, we use an implicit method to temporally discretize the system. In this work, we use Euler's implicit method. In the k'th control interval, we discretize the dynamical system using N_k time steps, i.e.

$$x_{k,n+1} - x_{k,n} = f(x_{k,n+1}, u_k, \hat{d}_k)\Delta t_{k,n}, \qquad (22)$$

for $n = 0, \ldots, N_k - 1$. We formulate the discretized differential equations in residual form:

$$R_{k,n} = R_{k,n}(x_{k,n+1}, x_{k,n}, u_k, \hat{d}_k)$$
$$= x_{k,n+1} - x_{k,n} - f(x_{k,n+1}, u_k, \hat{d}_k)\Delta t_{k,n}$$
$$= 0. \qquad (23)$$

At the boundaries of the control intervals, we enforce continuity through

$$x_{0,0} = \hat{x}_0, \qquad (24a)$$
$$x_{k,0} = x_{k-1,N_{k-1}}, \qquad k = 1, \ldots, N - 1. \qquad (24b)$$

B. The simultaneous approach

We transcribe the infinite-dimensional optimal control problem (20) to a finite-dimensional NLP using the temporal discretization based on Euler's implicit method (23)-(24):

$$\min_{\{\{x_{k,n}\}_{n=0}^{N_k}\}_{k=0}^{N-1}, \{u_k\}_{k=0}^{N-1}} \Psi, \qquad (25a)$$

subject to

$$x_{0,0} = \hat{x}_0, \qquad (25b)$$
$$x_{k,0} = x_{k-1,N_{k-1}}, \quad k = 1, \ldots, N - 1, \qquad (25c)$$
$$R_{k,n}(x_{k,n+1}, x_{k,n}, u_k, \hat{d}_k) = 0,$$
$$\qquad n = 0, \ldots, N_k - 1, \quad k = 0, \ldots, N - 1. \qquad (25d)$$

The objective function in (25a) is based on a discretization of the objective function (21) using the right rectangle rule:

$$\Psi = \Psi(\{\{x_{k,n}\}_{n=0}^{N_k}\}_{k=0}^{N-1}, \{u_k\}_{k=0}^{N-1}, \{\hat{d}_k\}_{k=0}^{N-1})$$
$$= \sum_{k=0}^{N-1} \sum_{n=0}^{N_k-1} \Phi(x_{k,n+1}, u_k, \hat{d}_k)\Delta t_{k,n}$$
$$+ \hat{\Phi}(x_{N-1,N_{N-1}}, u_{N-1}, \hat{d}_{N-1}). \qquad (26)$$

We use the right rectangle rule to be consistent with the discretization of the differential equations using Euler's implicit method.

We consider two approaches for solving the NLP (25). In the first approach, we directly incorporate the continuity constraints (25b)-(25c) into the NLP as actual constraints. We refer to this as the standard approach. In the second approach, we use the continuity constraints to eliminate the states at the beginning of every control interval, $\{x_{k,0}\}_{k=0}^{N-1}$. This leads to a reduced number of decision variables and constraints in the NLP. We refer to this as the reduced approach. A possible advantage of the standard approach is that the constraints are more loosely coupled which may lead to faster convergence (i.e. fewer iterations) when solving the NLP, whereas the advantage of the reduced approach is that the computational cost per iteration in the numerical solution of the NLP is lower (due to the lower number of decision variables and constraints).

In both approaches, we solve an NLP in the form

$$\min_{s} \quad \Psi(s, \hat{d}), \qquad (27a)$$
$$\text{subject to} \quad R(s, \hat{x}_0, \hat{d}) = 0, \qquad (27b)$$

where $\hat{d} = [\hat{d}_0; \cdots; \hat{d}_{N-1}]$. In the standard approach, the decision variables are

$$s = [s_0; \cdots; s_{N-1}], \qquad (28a)$$
$$s_k = [x_{k,0}; \cdots; x_{k,N_k}; u_k], \quad k = 0, \ldots, N - 1, \qquad (28b)$$

and the residual equations (27b) include the continuity constraints (25b)-(25c). In the reduced approach, the decision variables are

$$s = [s_0; \cdots; s_{N-1}], \qquad (29a)$$
$$s_k = [x_{k,1}; \cdots; x_{k,N_k}; u_k], \qquad (29b)$$

and the residual equations (27b) do not include the continuity constraints.

The total number of decision variables in the standard approach is $(N_t+N)n_x+Nn_u$ where $N_t = \sum_{k=0}^{N-1} N_k$ is the total number of time steps, n_x is the dimension of the state vector, and n_u is the dimension of the manipulated inputs. The total number of residual equations is $(N_t + N)n_x$. The total number of decision variables in the reduced approach is $N_t n_x + Nn_u$, and the total number of residual equations is $N_t n_x$. Consequently, in the reduced approach, the NLP involves Nn_x fewer decision variables and constraints than in the standard approach. This difference can be considerable if N_t and N are of comparable size and if n_x is significantly larger than n_u.

IV. Implementation details

We implement the two collocation-based approaches described in Section III in C, and we use IPOPT 3.12.12 [15] to solve the NLP (27). Apart from providing the objective function, $\Psi(s, \hat{d})$, and the residual function, $R(s, \hat{x}_0, \hat{d})$, we provide IPOPT with 1) the gradient of the objective function, $\frac{\partial \Psi}{\partial s}$, 2) the Jacobian of the residual function, $\frac{\partial R}{\partial s}$, and 3) the Hessian of the Lagrangian, $\nabla^2_{ss} L(s, \hat{d})$, where the Lagrangian is $L(s, \hat{d}) = \sigma \Psi(s, \hat{d}) + \lambda^T R(s, \hat{x}_0, \hat{d})$ [16, Sec. 9]. IPOPT provides both the scaling factor σ and the vector of Lagrange multipliers λ. The Jacobian matrix of the residual function and the Hessian of the Lagrangian are large and sparse, and we provide the exact sparsity patterns of these two matrices (i.e. we do not represent any zero entries unnecessarily).

IPOPT requires third party software related to the solution of linear systems of equations. We use the MA57 routine from HSL [17] as well as METIS 4.0.3 [18]. Furthermore, we compare the computational efficiency of using the OpenBLAS 0.2.20 linear algebra software and Netlib's implementation of BLAS (downloaded using IPOPT). In both cases, we use LAPACK 3.4.2 (also downloaded using IPOPT).

We compile the C code (including IPOPT) using gcc 5.4.0, and we carry out the computations presented in this paper on a 64-bit workstation which uses the operating system Ubuntu 16.04 LTS. The workstation has 15.6 GB RAM and four cores, each of which contains two Intel Core i7 3.60 GHz processors (eight processors in total). The four cores share a level 3 cache of 8192 KB, and the level 2 and level 1 cache of each core have 256 KB and 64 KB (32 KB instruction cache and 32 KB data cache), respectively.

V. Numerical examples

In this section, we present a performance study of the solution of two optimal control problems. The objective in both examples is to maximize the profit (19) over a control and prediction horizon of 30 h consisting of $N = 600$ control intervals with $N_k = 1$ time steps per control interval. The first optimal control problem is the optimal startup problem considered by Drejer et al. [14]. The second optimal control problem is the one that would be solved next in a closed-loop nonlinear model predictive control algorithm. The purpose of the second problem is to test the computational efficiency of warm starting.

In the first problem, at the initial time, the U-loop reactor contains 0.1 kg/m^3 biomass in the top tank and in the U-loop pipe and no substrate or oxygen. We consider two initial guesses, one that is constant in time, and one in which the states are obtained from a linear interpolation of the states in the first initial guess and the initial condition (the manipulated inputs are identical in the two initial guesses). Fig. 2 and Fig. 3 show the optimal states and the manipulated inputs (together with the corresponding initial guesses and bounds). The initial guesses and the bounds on the concentrations of biomass, substrate, and dissolved oxygen in the U-loop pipe are the same as for the concentrations in the top tank. The

Approach	Standard	Reduced
Number of decision variables	102600	52200
Number of residual equations	100800	50400

initial guess of the oxygen in the gas phase of the U-loop pipe is 2.0 kg/m^3, and the upper and lower bounds are 6.0 kg/m^3 and 0.0 kg/m^3, respectively.

For the second problem, we construct an initial guess, $s^{\text{ws}} = [s_0^{\text{ws}}; \cdots ; s_{N-1}^{\text{ws}}]$, by $s_k^{\text{ws}} = s_{k+1}^*$ for $k = 0, \ldots, N-2$ and $s_{N-1}^{\text{ws}} = s_{N-2}^{\text{ws}}$ where $s^* = [s_0^*; \cdots ; s_{N-1}^*]$ is the solution to the first optimal control problem. We construct initial guesses of the Lagrange multipliers in a similar manner. The initial condition in the second problem is the optimal states at the end of the first control interval ($k = 0$) in the first optimal control problem, i.e. $\hat{x}_0 = x_{0, N_0}^*$. The bounds are identical in the two problems.

Table II shows the number of decision variables and residual equations in the two approaches, and Table III shows various key performance indicators for the solution of the two optimal control problems. For the standard approach, it is better to use the interpolated initial guess whereas for the reduced approach, the interpolated initial guess leads to slow convergence. The use of warm starting leads to a significant reduction in the number of iterations and the computation time for both approaches, and we are able to solve the optimal control problem in 28.5 s which is less than the length of the control intervals (3 min). This demonstrates that it is feasible to solve the optimal control problems in real-time. Finally, the use of OpenBLAS generally improves the computational performance, and in some cases, it leads to a significant reduction in the number of iterations in IPOPT.

VI. Conclusion

In this work, we consider economic optimal control of SCP production in a U-loop reactor. The dynamical model of the U-loop reactor consists of both partial and ordinary differential equations. We use a finite-volume approach to discretize the partial differential equations. This leads to a set of ordinary differential equations with 83 states. Consequently, the optimal control problems are large-scale. We describe two collocation-based approaches for solving these large-scale optimal control problems. We implement the two approaches in C, and we use IPOPT 3.12.12 to solve the involved NLP (which involves up to around 100,000 decision variables). Furthermore, we present a performance study which demonstrates that it is feasible to solve optimal control problems that involve a U-loop reactor in real-time. Future work involves 1) the use of fully implicit Runge-Kutta (FIRK) methods instead of Euler's implicit method in the collocation-based approaches and 2) implementation of economic nonlinear model predictive control for SCP production in a U-loop reactor.
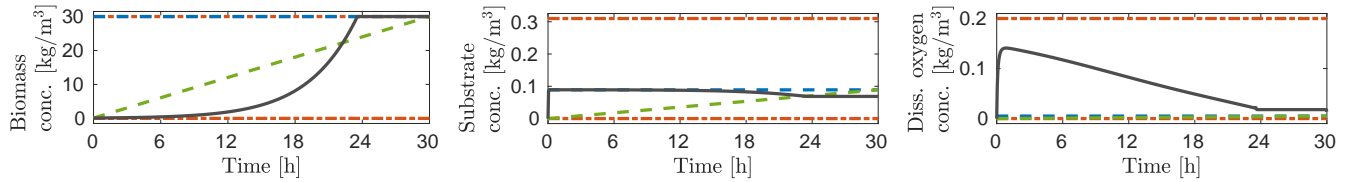
Fig. 2. Constant initial guesses (blue dashed), interpolated initial guesses (green dashed), and optimal values (black solid) of the concentrations of biomass, substrate, and dissolved oxygen in the top tank together with their bounds (red dash-dotted). The constant initial guess of the biomass concentration coincides with the corresponding optimal bound.
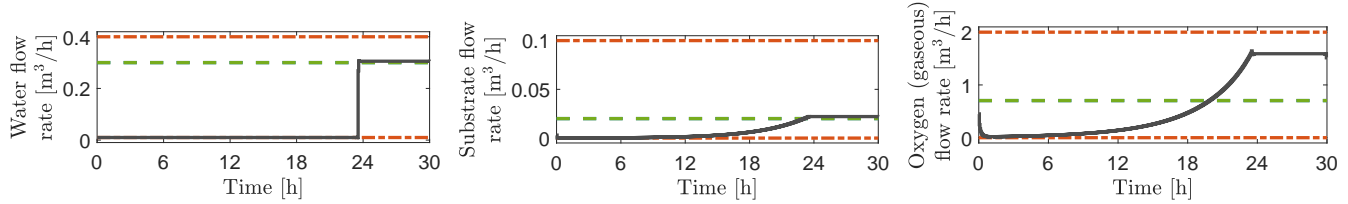

Fig. 3. Initial guesses (green dashed) and optimal values (black solid) of the manipulated inputs (water, substrate, and oxygen feed flow rates) together with their bounds (red dash-dotted). The manipulated inputs are identical in the constant and the interpolated initial guesses.

TABLE III

KEY PERFORMANCE INDICATORS OF THE STANDARD (STD.) AND REDUCED (RED.) COLLOCATION-BASED APPROACHES USING DIFFERENT STARTING GUESSES AND LINEAR ALGEBRA SOFTWARE.

Initial guess	Constant		Interpolated		Warm started	
Approach	Std.	Red.	Std.	Red.	Std.	Red.
Netlib BLAS						
Iterations	328	317	159	624	60	24
Func. eval.	329	318	232	647	71	25
Con. eval.	329	318	232	647	71	25
Grad. eval.	329	318	88	617	61	25
Jac. eval.	329	318	161	629	61	25
Hess. eval.	328	317	159	624	60	24
CPU time (s)	302.3	268.9	164.4	1113.9	71.9	31.4
OpenBLAS						
Iterations	330	299	145	426	29	24
Func. eval.	331	300	152	445	30	25
Con. eval.	331	300	152	445	30	25
Grad. eval.	331	300	126	419	30	25
Jac. eval.	331	300	147	430	30	25
Hess.eval.	330	299	145	426	29	24
CPU time (s)	284.7	238.6	199.9	703.3	36.9	28.5

REFERENCES

[1] T. Binder, L. Blank, H. G. Bock, R. Bulirsch, W. Dahmen, M. Diehl, T. Kronseder, W. Marquardt, J. P. Schlöder, and O. von Stryk, "Introduction to model based optimization of chemical processes on moving horizons," in *Online Optimization of Large Scale Systems*. Springer-Verlag Berlin Heidelberg, 2001, pp. 295–339.

[2] S. Kameswaran, L. T. Biegler, and G. H. Staus, "Dynamic optimization for the core-flooding problem in reservoir engineering," *Computers & chemical engineering*, vol. 29, no. 8, pp. 1787–1800, 2005.

[3] T. A. N. Heirung, M. R. Wartmann, J. D. Jansen, B. E. Ydstie, and B. A. Foss, "Optimization of the water-flooding process in a small 2D horizontal oil reservoir by direct transcription," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 10 863–10 868, 2011.

[4] L. T. Biegler, "An overview of simultaneous strategies for dynamic optimization," *Chemical Engineering and Processing: Process Intensification*, vol. 46, no. 11, pp. 1043–1053, 2007.

[5] H. G. Bock and K. J. Plitt, "A multiple shooting algorithm for direct solution of optimal control problems," *IFAC Proceedings Volumes*, vol. 17, no. 2, pp. 1603–1608, 1984.

[6] A. Schäfer, P. Kühl, M. Diehl, J. Schlöder, and H. G. Bock, "Fast reduced multiple shooting methods for nonlinear model predictive control," *Chemical Engineering and Processing: Process Intensification*, vol. 46, no. 11, pp. 1200–1214, 2007.

[7] J. Nocedal and S. J. Wright, *Numerical optimization*, 2nd ed. Springer Science & Business Media, 2006.

[8] D. F. Olsen, J. B. Jørgensen, J. Villadsen, and S. B. Jørgensen, "Modeling and simulation of single cell protein production," *IFAC Proceedings Volumes*, vol. 43, no. 6, pp. 502–507, 2010.

[9] O. A. Prado-Rubio, J. B. Jørgensen, and S. B. Jørgensen, "Systematic model analysis for single cell protein (SCP) production in a U-loop reactor," *Computer Aided Chemical Engineering*, vol. 28, pp. 319–324, 2010.

[10] A. M. Al Taweel, Q. Shah, and B. Aufderheide, "Effect of mixing on microorganism growth in loop bioreactors," *International Journal of Chemical Engineering*, no. Article ID 984827, 2012.

[11] M. Wu, J. K. Huusom, K. V. Gernaey, and U. Krühne, "Modelling and simulation of a U-loop reactor for single cell protein production," *Computer Aided Chemical Engineering*, vol. 38, pp. 1287–1292, 2016.

[12] L. A. H. Petersen, J. Villadsen, S. B. Jørgensen, and K. V. Gernaey, "Mixing and mass transfer in a pilot scale U-loop bioreactor," *Biotechnology and Bioengineering*, vol. 114, no. 2, pp. 344–354, 2017.

[13] D. F. Olsen, J. B. Jørgensen, J. Villadsen, and S. B. Jørgensen, "Optimal operating points for SCP production in the U-loop reactor," *IFAC Proceedings Volumes*, vol. 43, no. 5, pp. 499–504, 2010.

[14] A. Drejer, T. Ritschel, S. B. Jørgensen, and J. B. Jørgensen, "Economic optimizing control for single-cell protein production in a U-loop reactor," *Computer Aided Chemical Engineering*, vol. 40, pp. 1759–1764, 2017.

[15] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.

[16] A. Wächter, "Short tutorial: getting started with Ipopt in 90 minutes," in *Combinatorial Scientific Computing*, ser. Dagstuhl Seminar Proceedings, U. Naumann, O. Schenk, H. D. Simon, and S. Toledo, Eds., no. 09061. Dagstuhl, Germany: Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, Germany, 2009. [Online]. Available: http://drops.dagstuhl.de/opus/volltexte/2009/2089

[17] HSL, "A collection of Fortran codes for large scale scientific computation," www.hsl.rl.ac.uk, accessed February 2nd 2019.

[18] G. Karypis and V. Kumar, "A fast and high quality multilevel scheme for partitioning irregular graphs," *SIAM Journal on Scientific Computing*, vol. 20, no. 1, pp. 359–392, 1998.