

Bi-objective dynamic optimization of a nonlinear time-delay system in microbial batch process

Chongyang Liu · Zhaohua Gong · Kok Lay Teo · Ryan Loxton · Enmin Feng

Received: date / Accepted: date

Abstract In this paper, we propose a bi-objective dynamic optimization model involving a nonlinear time-delay system to optimize the 1,3-propanediol (1,3-PD) production in a microbial batch process, where the productivity of 1,3-PD and the consumption rate of glycerol are taken as the two objectives. The initial concentrations of biomass and glycerol, and the terminal time of the process are the decision variables. By a time-scaling transformation, we first transform the problem to the one with fixed terminal time but involving a new system with variable time-delay. The normalized normal constraint method is then used to convert the resulting problem into a sequence of single-objective dynamic optimization problems. A gradient-based optimization method incorporating the constraint transcription technique is developed to solve each of these single-objective dynamic optimization problems. Finally, numerical results are provided to demonstrate the effectiveness of the proposed solution method.

C. Liu (✉)

School of Mathematics and Information Science, Shandong Institute of Business and Technology, Yantai, China

E-mail: liu_chongyang@yahoo.com

Z. Gong

School of Mathematics and Information Science, Shandong Institute of Business and Technology, Yantai, China

E-mail: zhaohuagong@163.com

K.L. Teo

Department of Mathematics and Statistics, Curtin University, Perth, Australia

E-mail: k.l.teo@curtin.edu.au

R. Loxton

Department of Mathematics and Statistics, Curtin University, Perth, Australia

E-mail: r.loxton@curtin.edu.au

E. Feng

School of Mathematical Sciences, Dalian University of Technology, Dalian, China

E-mail: emfeng@dlut.edu.cn

Keywords Nonlinear time-delay system · Bi-objective optimization · Dynamic optimization · normalized normal constraint · Batch process

Mathematics Subject Classification (2000) 49M37 · 93C23 · 90C29

1 Introduction

Maximization of productivity and minimization of consumption are two top priorities for any process industry. There is no exception in the case of fermentation, where substrate consumption is expected to be minimized while the target product productivity is expected to be maximized. This gives rise to two conflicting objectives. Thus, it is a bi-objective optimization problem in optimizing 1,3-propanediol (1,3-PD) microbial production process.

1,3-PD is a bulk chemical, which is used in the manufacture of polymers, cosmetics, foods, lubricants, and medicines [27]. In recent years, 1,3-PD microbial production is particularly attractive in that the process does not generate toxic byproducts [1]. Glycerol is a prospective feedstock for 1,3-PD production. Among microorganisms synthesizing 1,3-PD, *Klebsiella pneumoniae* (*K. pneumoniae*) is commonly used because of its high yield and productivity [14]. Glycerol fermentation by *K. pneumoniae* is a complex bio-process containing time-delays and subject to multiple inhibitions of substrate and products [3,22]. Compared with other operation modes, 1,3-PD batch production can obtain the highest molar yield 1,3-PD to glycerol [7]. For 1,3-PD batch production, extensive studies have been carried out. A kinetic model for substrate consumption and product formation is established in [23]. An enzyme-catalytic kinetic model is proposed in [20]. Based on these mathematical models, a parameter identification problem is discussed in [25]. A robust dynamic optimization problem is investigated in [2]. Recently, a nonlinear time-delay system is proposed to formulate the batch process [11]. For this system, the corresponding dynamic optimization problem is discussed in [24]. Although the results obtained are interesting, only one objective is involved in the above identification and dynamic optimization problems and thus they all are single-objective dynamic optimization problems.

In this paper, we propose a bi-objective optimization model involving a nonlinear time-delay system [11] to optimize 1,3-PD batch production, where both the productivity of 1,3-PD and the consumption rate of glycerol are taken as objective functions. The initial concentrations of biomass and glycerol, and the terminal time of the process are taken as the decision variables. By a time-scaling transformation [12], we transform the bi-objective dynamic optimization problem into an equivalent one with fixed terminal time but involving a new system with variable time-delay. For this bi-objective optimization problem, or a general multi-objective optimization problem (MOOP), there is a set of optimal solutions that are equally good, which is known as the Pareto set. By the way, numerical methods for generating the Pareto set of a MOOP can be broadly classified into two categories [9]. The first category, which is known as scalarization methods, transforms a MOOP into a sequence

of parametric single-objective optimization problems, e.g., convex weighted sum (CWS) [26], normalized normal constraint (NNC) [15,16], normal boundary intersection [5] and adaptive methods [6]. The second category, which is referred as vectorization methods, generates the Pareto set directly from the multi-objective formulation, e.g., particle swarm optimization [4] and genetic algorithm [10]. Note that, for the scalarization methods, gradient-based deterministic optimization routes can be combined with to find optimal solutions for large-scale and highly constrained MOOPs in a fast and efficient way [13]. In this paper, we convert the transformed problem into a sequence of single-objective dynamic optimization problems by using the NNC method. The main reasons to choose the NNC method are: (i) it is independent of the objective scales; and (ii) it can generate well-distributed points on the Pareto set. We then develop a new gradient-based single-objective solver incorporating the constraint transcription technique [21] to solve each of these resulting single-objective problems. On this basis, the NNC method in conjunction with the new gradient-based single-objective solver is used to solve the bi-objective dynamic optimization problem. Finally, numerical results are provided to demonstrate the effectiveness of the proposed approach.

2 Dynamic Model

Based on the previous work [11], mass balance relationships for biomass, substrate, and products in the process can be expressed as the following nonlinear time-delay system:

$$\begin{cases} \dot{x}_1(t) = \mu(x(t))x_1(t-h), \\ \dot{x}_2(t) = -q_2(x(t))x_1(t-h), \\ \dot{x}_\ell(t) = q_\ell(x(t))x_1(t-h), \quad \ell = 3, 4, 5, \end{cases} \quad t \in (0, t_f], \quad (1)$$

with

$$x(t) = \phi(t, \zeta), \quad t \leq 0, \quad (2)$$

where t denotes the process time; $t_f > 0$ is the terminal time of the process; $x(t) := (x_1(t), x_2(t), x_3(t), x_4(t), x_5(t))^\top \in R^5$ is the state vector whose components are, respectively, the concentrations of biomass, glycerol, 1,3-PD, acetate and ethanol at time t in the reactor; $h = 0.26$ is a delay argument; $\zeta := (\zeta_1, \zeta_2, \zeta_3, \zeta_4, \zeta_5)^\top$ is the initial state; and $\phi : R \times R^5 \rightarrow R^5$ is a given initial function. Moreover, $\mu(x(t))$ is the specific growth rate of cells, $q_2(x(t))$ is the specific consumption rate of substrate, and $q_\ell(x(t))$, $\ell = 3, 4, 5$, are the specific formation rates of products. These quantities are expressed by the following equations [23]:

$$\begin{aligned} \mu(x(t)) &:= \frac{\mu_m x_2(t)}{x_2(t) + k_1} \prod_{\ell=2}^5 \left(1 - \frac{x_\ell(t)}{x_\ell^*}\right), \\ q_\ell(x(t)) &:= m_\ell + Y_\ell \mu(x(t)), \quad \ell = 2, 3, 4, 5, \end{aligned}$$

Table 1 The kinetic parameters and critical concentrations in the system (1) [11].

μ_m	k_1	m_2	m_3	m_4	m_5	Y_2	Y_3	Y_4	Y_5
0.994	0.368	3.24	-3.679	-0.491	7.309	11.75	76	35.54	14.78
x_{*1}	x_{*2}	x_{*3}	x_{*4}	x_{*5}	x_1^*	x_2^*	x_3^*	x_4^*	x_5^*
0.01	0	0	0	0	6	2039	939.5	1026	360.9

where μ_m , k_1 , m_ℓ and Y_ℓ are kinetic parameters; and x_ℓ^* are critical concentrations for cell growth. The values of these kinetic parameters and critical concentrations are listed in Table 1.

The terminal time in system (1) and (2) is a decision variable. Define

$$\mathcal{T} := \{\tau \in R : a_1 \leq \tau \leq b_1\},$$

where a_1 and b_1 are the lower and upper bounds for the terminal time, respectively. Any $t_f \in \mathcal{T}$ is called an *admissible terminal time*. In addition, $\zeta_\ell = 0$, $\ell = 3, 4, 5$, since there is no product being entered at the initial time point. Let $u := (\zeta_1, \zeta_2)^\top$ and define

$$\mathcal{U} := \{v \in R^2 : c_i \leq v_i \leq d_i, i = 1, 2\},$$

where c_i and d_i are real numbers such that $c_i \leq d_i$. Any $u \in \mathcal{U}$ is called an *admissible initial concentration vector of biomass and glycerol*.

For system (1) and (2), there exists a unique absolutely continuous solution $x(\cdot|u)$ corresponding to each $u \in \mathcal{U}$ on $[0, \infty]$ [8]. Furthermore, there exist critical concentrations of biomass, glycerol and products. Outside these critical concentrations, the cells will cease to grow. Thus, it is biologically meaningful to restrict the concentrations of biomass, glycerol, and products within a set defined by

$$x(t|u) \in W := \prod_{\ell=1}^5 [x_{*\ell}, x_\ell^*], t \in [0, t_f], \quad (3)$$

where $x_{*\ell}$ and x_ℓ^* are the critical concentrations for cell growth (as used in the formula $\mu(x(t))$). These values are also listed in Table 1.

3 Bi-objective dynamic optimization problem

For the batch production governed by system (1) and (2), the aims are to maximize the productivity of the target product and, at the same time, to minimize the consumption rate of the substrate. Thus, there are two conflicting objectives.

(i) Maximizing the productivity of 1,3-PD:

$$\max \frac{x_3(t_f|u)}{t_f}.$$

(ii) Minimizing the consumption rate of glycerol:

$$\min \frac{u_2 - x_2(t_f|u)}{t_f},$$

where u_2 is the initial concentration of glycerol.

Let

$$J(u, t_f) := \left(-\frac{x_3(t_f|u)}{t_f}, \frac{u_2 - x_2(t_f|u)}{t_f} \right)^\top \quad (4)$$

be the objective vector to be minimized. Then, incorporating constraint (3), we present the following bi-objective dynamic optimization problem:

$$\begin{aligned} \text{(BP)} \quad & \min J(u, t_f) \\ & \text{s.t. } x(t|u) \in W, \quad t \in [0, t_f], \\ & (u, t_f) \in \mathcal{U} \times \mathcal{T}. \end{aligned}$$

For (BP), there are three non-standard features: (i) the terminal time is free instead of fixed; (ii) the objective function is not a scalar but a vector; and (iii) constraint (3) is a continuous state inequality constraint (i.e., it must be satisfied at an infinite number of points in the time interval).

To circumvent the first difficulty, we apply a time-scaling transformation [12] from $[0, t_f]$ to $[0, 1]$ as given below:

$$t = t(s) = t_f s, \quad (5)$$

where $s \in [0, 1]$ is a new time variable. Clearly, $s = 0$ corresponds to $t = 0$, and $s = 1$ corresponds to $t = t_f$. Then, system (1) is transformed into an equivalent form given below:

$$\begin{cases} \dot{\tilde{x}}_1(t) = t_f \tilde{\mu}(\tilde{x}(s)) \tilde{x}_1(s - t_f^{-1}h), \\ \dot{\tilde{x}}_2(t) = -t_f \tilde{q}_2(\tilde{x}(s)) \tilde{x}_1(s - t_f^{-1}h), \\ \dot{\tilde{x}}_\ell(t) = t_f \tilde{q}_\ell(\tilde{x}(s)) \tilde{x}_1(s - t_f^{-1}h), \quad \ell = 3, 4, 5, \end{cases} \quad s \in (0, 1], \quad (6)$$

where

$$\begin{aligned} \tilde{x}(s) &:= x(t_f s), \\ \tilde{\mu}(\tilde{x}(s)) &:= \mu(x(t_f s)), \\ \tilde{q}_\ell(\tilde{x}(s)) &:= q_\ell(x(t_f s)), \quad \ell = 2, 3, 4, 5. \end{aligned}$$

The initial condition (2) becomes

$$\tilde{x}(s) = \phi(t_f s, \zeta), \quad s \leq 0. \quad (7)$$

Let $\tilde{x}(\cdot|u, t_f)$ denote the solution of the system (6) and (7) corresponding to each $(u, t_f) \in \mathcal{U} \times \mathcal{T}$ on $[0, 1]$. Then, under the time-scaling transformation (5), constraint (3) becomes

$$\tilde{x}(s|u, t_f) \in W, \quad s \in [0, 1], \quad (8)$$

and the objective vector (4) turns into

$$\tilde{J}(u, t_f) := \left(-\frac{\tilde{x}_3(1|u, t_f)}{t_f}, \frac{u_2 - \tilde{x}_2(1|u, t_f)}{t_f} \right)^\top. \quad (9)$$

Therefore, (BP) can be stated equivalently as the following bi-objective dynamic optimization problem:

$$\begin{aligned} \text{(EP)} \quad & \min \tilde{J}(u, t_f) \\ & \text{s.t. } \tilde{x}(s|u, t_f) \in W, \quad s \in [0, 1], \\ & (u, t_f) \in \mathcal{U} \times \mathcal{T}. \end{aligned}$$

Obviously, (EP) is a bi-objective dynamic optimization problem with fixed terminal time but involving a new system with variable time-delay.

4 A solution method

In this section, a numerical solution method combining the NNC method [16] with a gradient-based single-objective solver is developed to solve (EP).

4.1 Normalized normal constraint

To apply the NNC method, we require two anchor points, denoted by $\tilde{J}_1(u^{1*}, t_f^{1*})$ and $\tilde{J}_2(u^{2*}, t_f^{2*})$. They are obtained from solving the following (EPU_j), $j = 1, 2$, respectively.

$$\begin{aligned} \text{(EPU}_j) \quad & \min \tilde{J}_j(u, t_f) \\ & \text{s.t. } \tilde{x}(s|u, t_f) \in W, \quad s \in [0, 1], \\ & (u, t_f) \in \mathcal{U} \times \mathcal{T}, \end{aligned}$$

where \tilde{J}_j is the j th component of \tilde{J} . The line joining these two points is called the *Utopia line*. Furthermore, we define the *Utopia point*, \tilde{J}^u , as:

$$\tilde{J}^u := (\tilde{J}_1(u^{1*}, t_f^{1*}), \tilde{J}_2(u^{2*}, t_f^{2*}))^\top.$$

Note that the Utopia point is generally unattainable. The NNC method generates points of the Pareto set in the normalized objective space. Thus, define

$$\begin{aligned} l_1 &:= \tilde{J}_1(u^{2*}, t_f^{2*}) - \tilde{J}_1(u^{1*}, t_f^{1*}), \\ l_2 &:= \tilde{J}_2(u^{1*}, t_f^{1*}) - \tilde{J}_2(u^{2*}, t_f^{2*}). \end{aligned}$$

Then, l_1 and l_2 give the distances between $\tilde{J}(u^{2*}, t_f^{2*})$ and \tilde{J}^u , and between $\tilde{J}(u^{1*}, t_f^{1*})$ and \tilde{J}^u , respectively. Using l_1 and l_2 , the normalized form of \tilde{J} is defined as

$$\bar{J}(u, t_f) := \left(\frac{\tilde{J}_1(u, t_f) - \tilde{J}_1(u^{1*}, t_f^{1*})}{l_1}, \frac{\tilde{J}_2(u, t_f) - \tilde{J}_2(u^{2*}, t_f^{2*})}{l_2} \right)^\top. \quad (10)$$

Let \bar{N} be the direction from $\bar{J}(u^{1*}, t_f^{1*})$ to $\bar{J}(u^{2*}, t_f^{2*})$ defined by

$$\bar{N} := \bar{J}(u^{2*}, t_f^{2*}) - \bar{J}(u^{1*}, t_f^{1*}).$$

In addition, compute a normalized increment, δ , along the direction \bar{N} for a prescribed number, m , of solutions as:

$$\delta := \frac{1}{m-1}.$$

Evaluate a set of evenly distributed points on the Utopia line as

$$\bar{X}_{pk} := \alpha_{1k} \bar{J}(u^{1*}, t_f^{1*}) + \alpha_{2k} \bar{J}(u^{2*}, t_f^{2*}),$$

where α_{1k} and α_{2k} are weights satisfying $0 \leq \alpha_{1k} \leq 1$, $0 \leq \alpha_{2k} \leq 1$, and $\alpha_{1k} + \alpha_{2k} = 1$. Note that α_{1k} , $k \in \{1, 2, \dots, m\}$, is incremented by δ between 0 and 1. Using the set of evenly distributed points on the Utopia line, we can generate a corresponding set of Pareto points by solving (EPU₂) sequentially with an additional constraint:

$$\bar{N}^\top \cdot (\bar{J} - \bar{X}_{pk}) \leq 0. \quad (11)$$

Specifically, for each generated point on the Utopia line, solve the following single-objective dynamic optimization problem for the k th point \bar{X}_{pk} :

$$\begin{aligned} (\text{EPU}_{2,k}) \quad & \min \bar{J}_2(u, t_f) \\ & \text{s.t. } \tilde{x}(s|u, t_f) \in W, \quad s \in [0, 1], \\ & \quad \bar{N}^\top \cdot (\bar{J} - \bar{X}_{pk}) \leq 0, \\ & \quad (u, t_f) \in \mathcal{U} \times \mathcal{T}. \end{aligned}$$

4.2 Single-objective solver

In this subsection, we will develop a gradient-based single-objective solver in conjunction with constraint transcription technique [21].

4.2.1 Constraint transcription

Constraint (8) is a continuous state inequality constraint. We will apply a constraint transcription technique to transcribe constraint (8) into a canonical constraint.

Let

$$\begin{aligned} h_\ell(\tilde{x}(s|u, t_f)) &:= x_\ell^* - \tilde{x}_\ell(s|u, t_f), \\ h_{5+\ell}(\tilde{x}(s|u, t_f)) &:= \tilde{x}_\ell(s|u, t_f) - x_{*\ell}, \quad \ell = 1, 2, \dots, 5. \end{aligned}$$

Then, constraint (8) is equivalent to the following equality constraint:

$$\tilde{G}(u, t_f) := \sum_{l=1}^{10} \int_0^1 \min\{0, h_l(\tilde{x}(s|u, t_f))\} ds = 0. \quad (12)$$

However, since $\min(\cdot, \cdot)$ is non-differentiable at the points where $h_l = 0$, standard optimization routines would have difficulties in dealing with such equality constraint. Thus, constraint (12) is approximated by

$$\tilde{G}^{\epsilon, \gamma}(u, t_f) := \gamma + \sum_{l=1}^{10} \int_0^1 \varphi^\epsilon(h_l(\tilde{x}(s|u, t_f))) ds \geq 0, \quad (13)$$

where $\epsilon > 0$, $\gamma > 0$ are two adjustable parameters; and

$$\varphi^\epsilon(\eta) = \begin{cases} \eta, & \text{if } \eta < -\epsilon, \\ -\frac{(\eta - \epsilon)^2}{4\epsilon}, & \text{if } -\epsilon \leq \eta \leq \epsilon, \\ 0, & \text{if } \eta > \epsilon. \end{cases}$$

Note that inequality constraint (13) is in the form of a canonical constraint. For each $\epsilon > 0$ and $\gamma > 0$, denote (EPU_j) and $(\text{EPU}_{2,k})$ replacing constraint (8) with inequality constraint (13) by $(\text{EPU}_j^{\epsilon, \gamma})$, $j = 1, 2$, and $(\text{EPU}_{2,k}^{\epsilon, \gamma})$, $k = 1, 2, \dots, m$, respectively. Then, we can solve a sequence of approximate problems $(\text{EPU}_j^{\epsilon, \gamma})$ and $(\text{EPU}_{2,k}^{\epsilon, \gamma})$ to obtain the solutions of (EPU_j) , $j = 1, 2$, and $(\text{EPU}_{2,k})$, $k = 1, 2, \dots, m$.

4.2.2 Gradient computation

In solving $(\text{EPU}_j^{\epsilon, \gamma})$, $j = 1, 2$, and $(\text{EPU}_{2,k}^{\epsilon, \gamma})$, $k = 1, 2, \dots, m$, the gradients of \tilde{J} , \bar{J} , and constraint functions (11) and (13) with respect to u and t_f are needed. Recall that \bar{J} is the normalized form of \tilde{J} and constraint function (11) is also a function of \tilde{J} . Therefore, we only need the gradients of \tilde{J} and constraint function (13) with respect to u and t_f . These gradient formulas are given in the following theorems. The proofs of these theorems are similar to those given for Theorems 1 and 2 in [12]. First, define

$$\tilde{f}(t_f, \tilde{x}(s), \tilde{x}(s - t_f^{-1}h)) := \begin{pmatrix} t_f \tilde{\mu}(\tilde{x}(s)) \tilde{x}_1(s - t_f^{-1}h) \\ -t_f \tilde{q}_2(\tilde{x}(s)) \tilde{x}_1(s - t_f^{-1}h) \\ t_f \tilde{q}_3(\tilde{x}(s)) \tilde{x}_1(s - t_f^{-1}h) \\ t_f \tilde{q}_4(\tilde{x}(s)) \tilde{x}_1(s - t_f^{-1}h) \\ t_f \tilde{q}_5(\tilde{x}(s)) \tilde{x}_1(s - t_f^{-1}h) \end{pmatrix},$$

$$\psi(s) := \begin{cases} \frac{\partial \phi(t_f s, \zeta)}{\partial s}, & s \leq 0, \\ \tilde{f}(t_f, \tilde{x}(s), \tilde{x}(s - t_f^{-1}h)), & s \in [0, 1], \end{cases}$$

and, for a given interval I ,

$$\chi_I(s) := \begin{cases} 1, & s \in I, \\ 0, & \text{otherwise.} \end{cases}$$

Then, the following theorem gives the gradient formulas of $\tilde{J}_1(u, t_f)$ with respect to u and t_f .

Theorem 1 Let $(u, t_f) \in \mathcal{U} \times \mathcal{T}$. Then

$$\begin{aligned} \frac{\partial \tilde{J}_1(u, t_f)}{\partial u} &= \lambda^\top(0) \frac{\partial \tilde{x}(0)}{\partial u} \\ &+ \int_{-t_f^{-1}h}^0 \lambda^\top(s + t_f^{-1}h) \frac{\partial \tilde{f}(t_f, \tilde{x}(s + t_f^{-1}h), \tilde{x}(s))}{\partial \tilde{x}(s)} \frac{\partial \phi(t_f s, \zeta)}{\partial u} ds, \end{aligned}$$

and

$$\begin{aligned} \frac{\partial \tilde{J}_1(u, t_f)}{\partial t_f} &= \tilde{x}_3(1) t_f^{-2} + \int_0^1 \lambda^\top(s) t_f^{-1} \tilde{f}(t_f, \tilde{x}(s), \tilde{x}(s - t_f^{-1}h)) ds \\ &+ \int_0^1 t_f^{-2} h \lambda^\top(s) \frac{\partial \tilde{f}(t_f, \tilde{x}(s), \tilde{x}(s - t_f^{-1}h))}{\partial \tilde{x}(s - t_f^{-1}h)} \psi(s - t_f^{-1}h) ds \\ &+ \int_{-t_f^{-1}h}^0 \lambda^\top(s + t_f^{-1}h) \frac{\partial \tilde{f}(t_f, \tilde{x}(s + t_f^{-1}h), \tilde{x}(s))}{\partial \tilde{x}(s)} \frac{\partial \phi(t_f s, \zeta)}{\partial t_f} ds, \end{aligned}$$

where $\lambda(\cdot)$ is the solution of the following costate system:

$$\begin{aligned} \dot{\lambda}(s) &= - \left(\frac{\partial \tilde{f}(t_f, \tilde{x}(s), \tilde{x}(s - t_f^{-1}h))}{\partial \tilde{x}(s)} \right)^\top \lambda(s) \\ &- \chi_{[0, 1 - t_f^{-1}h]}(s) \left(\frac{\partial \tilde{f}(t_f, \tilde{x}(s + t_f^{-1}h), \tilde{x}(s))}{\partial \tilde{x}(s)} \right)^\top \lambda(s + t_f^{-1}h), \quad s \in [0, 1], \end{aligned} \quad (14)$$

with the terminal conditions

$$\lambda(1) = (0, 0, -t_f^{-1}, 0, 0)^\top, \quad (15)$$

$$\lambda(s) = (0, 0, 0, 0, 0)^\top, \quad s > 1. \quad (16)$$

The next theorem gives the gradient formulas of $\tilde{J}_2(u, t_f)$ with respect to u and t_f .

Theorem 2 Let $(u, t_f) \in \mathcal{U} \times \mathcal{T}$. Then

$$\begin{aligned} \frac{\partial \tilde{J}_2(u, t_f)}{\partial u} &= (0, t_f^{-1})^\top + \bar{\lambda}^\top(0) \frac{\partial \tilde{x}(0)}{\partial u} \\ &+ \int_{-t_f^{-1}h}^0 \bar{\lambda}^\top(s + t_f^{-1}h) \frac{\partial \tilde{f}(t_f, \tilde{x}(s + t_f^{-1}h), \tilde{x}(s))}{\partial \tilde{x}(s)} \frac{\partial \phi(t_f s, \zeta)}{\partial u} ds, \end{aligned}$$

and

$$\begin{aligned} \frac{\partial \tilde{J}_2(u, t_f)}{\partial t_f} &= - \frac{u_2 - \tilde{x}_2(1)}{t_f^2} + \int_0^1 \bar{\lambda}^\top(s) t_f^{-1} \tilde{f}(t_f, \tilde{x}(s), \tilde{x}(s - t_f^{-1}h)) ds \\ &+ \int_0^1 t_f^{-2} h \bar{\lambda}^\top(s) \frac{\partial \tilde{f}(t_f, \tilde{x}(s), \tilde{x}(s - t_f^{-1}h))}{\partial \tilde{x}(s - t_f^{-1}h)} \psi(s - t_f^{-1}h) ds \\ &+ \int_{-t_f^{-1}h}^0 \bar{\lambda}^\top(s + t_f^{-1}h) \frac{\partial \tilde{f}(t_f, \tilde{x}(s + t_f^{-1}h), \tilde{x}(s))}{\partial \tilde{x}(s)} \frac{\partial \phi(t_f s, \zeta)}{\partial t_f} ds, \end{aligned}$$

where $\bar{\lambda}(\cdot)$ is the solution of the following costate system:

$$\begin{aligned} \dot{\bar{\lambda}}(s) = & - \left(\frac{\partial \tilde{f}(t_f, \tilde{x}(s), \tilde{x}(s - t_f^{-1}h))}{\partial \tilde{x}(s)} \right)^\top \bar{\lambda}(s) \\ & - \chi_{[0, 1-t_f^{-1}h]}(s) \left(\frac{\partial \tilde{f}(t_f, \tilde{x}(s + t_f^{-1}h), \tilde{x}(s))}{\partial \tilde{x}(s)} \right)^\top \bar{\lambda}(s + t_f^{-1}h), \quad s \in [0, 1], \end{aligned} \quad (17)$$

with the terminal conditions

$$\bar{\lambda}(1) = (0, -t_f^{-1}, 0, 0, 0)^\top, \quad (18)$$

$$\bar{\lambda}(s) = (0, 0, 0, 0, 0)^\top, \quad s > 1. \quad (19)$$

The last theorem gives the gradient formulas of $\tilde{G}^{\epsilon, \gamma}(u, t_f)$ with respect to u and t_f .

Theorem 3 Let $(u, t_f) \in \mathcal{U} \times \mathcal{T}$. Then for each $\epsilon > 0$ and $\gamma > 0$,

$$\begin{aligned} \frac{\partial \tilde{G}^{\epsilon, \gamma}(u, t_f)}{\partial u} = & \tilde{\lambda}^\top(0) \frac{\partial \tilde{x}(0)}{\partial u} \\ & + \int_{-t_f^{-1}h}^0 \tilde{\lambda}^\top(s + t_f^{-1}h) \frac{\partial \tilde{f}(t_f, \tilde{x}(s + t_f^{-1}h), \tilde{x}(s))}{\partial \tilde{x}(s)} \frac{\partial \phi(t_f s, \zeta)}{\partial u} ds, \end{aligned}$$

and

$$\begin{aligned} \frac{\partial \tilde{G}^{\epsilon, \gamma}(u, t_f)}{\partial t_f} = & \int_0^1 t_f^{-1} \tilde{\lambda}^\top(s) \tilde{f}(t_f, \tilde{x}(s), \tilde{x}(s - t_f^{-1}h)) ds \\ & + \int_0^1 t_f^{-2} h \tilde{\lambda}^\top(s) \frac{\partial \tilde{f}(t_f, \tilde{x}(s), \tilde{x}(s - t_f^{-1}h))}{\partial \tilde{x}(s - t_f^{-1}h)} \psi(s - t_f^{-1}h) ds \\ & + \int_{-t_f^{-1}h}^0 \tilde{\lambda}^\top(s + t_f^{-1}h) \frac{\partial \tilde{f}(t_f, \tilde{x}(s + t_f^{-1}h), \tilde{x}(s))}{\partial \tilde{x}(s)} \frac{\partial \phi(t_f s, \zeta)}{\partial t_f} ds, \end{aligned}$$

where $\tilde{\lambda}(\cdot)$ is the solution of the following costate system:

$$\begin{aligned} \dot{\tilde{\lambda}}(s) = & - \sum_{l=1}^{10} \frac{\partial \varphi^\epsilon(h_l(\tilde{x}(s)))}{\partial \tilde{x}(s)} - \left(\frac{\partial \tilde{f}(t_f, \tilde{x}(s), \tilde{x}(s - t_f^{-1}h))}{\partial \tilde{x}(s)} \right)^\top \tilde{\lambda}(s) \\ & - \chi_{[0, 1-t_f^{-1}h]}(s) \left(\frac{\partial \tilde{f}(t_f, \tilde{x}(s + t_f^{-1}h), \tilde{x}(s))}{\partial \tilde{x}(s)} \right)^\top \tilde{\lambda}(s + t_f^{-1}h), \quad s \in [0, 1], \end{aligned} \quad (20)$$

with the terminal condition

$$\tilde{\lambda}(s) = (0, 0, 0, 0, 0)^\top, \quad s \geq 1. \quad (21)$$

Theorems 1-3 provide the gradients of the objective vector $\tilde{J}(u, t_f)$ and the constraint function $\tilde{G}^{\epsilon, \gamma}(u, t_f)$ for each given $(u, t_f) \in \mathcal{U} \times \mathcal{T}$, $\epsilon > 0$, and $\gamma > 0$. Note that based on these gradients, the gradients of \tilde{J} and constraint function (11) with respect to u and t_f can be computed, accordingly. As a result, we present Algorithm 1 to solve (EPU $_j^{\epsilon, \gamma}$), $j = 1, 2$, and (EPU $_{2,k}^{\epsilon, \gamma}$), $k = 1, 2, \dots, m$.

Algorithm 1

Step 1: Provide an initial guess $(u, t_f) \in \mathcal{U} \times \mathcal{T}$, $\epsilon > 0$ and $\gamma > 0$, set parameters $0 < \beta_1 < 1$, $0 < \beta_2 < 1$, $\bar{\epsilon}$ and $\bar{\gamma}$.

Step 2: Compute $(u^{\epsilon, \gamma^*}, t_f^{\epsilon, \gamma^*})$.

Step 2.1: Solve system (6) from $s = 0$ to $s = 1$ to obtain the state solution denoted by $\tilde{x}(\cdot|u, t_f)$.

Step 2.2: Using $\tilde{x}(\cdot|u, t_f)$, compute $\tilde{J}(u, t_f)$ by (9) and $\tilde{G}^{\epsilon, \gamma}(u, t_f)$ by (13).

Step 2.3: Using $\tilde{x}(\cdot|u, t_f)$, solve costate systems (14)-(16), (17)-(19) and (20)-(21) from $s = 1$ to $s = 0$ to obtain costate solutions denoted by $\lambda(\cdot|u, t_f)$, $\bar{\lambda}(\cdot|u, t_f)$ and $\tilde{\lambda}(\cdot|u, t_f)$, respectively.

Step 2.4: Using $\tilde{x}(\cdot|u, t_f)$, $\lambda(\cdot|u, t_f)$, $\bar{\lambda}(\cdot|u, t_f)$ and $\tilde{\lambda}(\cdot|u, t_f)$, compute $\partial \tilde{J}(u, t_f)/\partial u$, $\partial \tilde{J}(u, t_f)/\partial t_f$, $\partial \tilde{G}^{\epsilon, \gamma}(u, t_f)/\partial u$ and $\partial \tilde{G}^{\epsilon, \gamma}(u, t_f)/\partial t_f$ by their respective formulas in Theorems 1-3. Furthermore, compute the gradients of \tilde{J} and constraint function (11) with respect to u and t_f .

Step 2.5: Solve $(\text{EPU}_j^{\epsilon, \gamma})$, $j \in \{1, 2\}$, or $(\text{EPU}_{2,k}^{\epsilon, \gamma})$, $k \in \{1, 2, \dots, m\}$, by gradient-based optimization (e.g., sequential quadratic programming (SQP) [17]) to obtain the corresponding $(u^{\epsilon, \gamma^*}, t_f^{\epsilon, \gamma^*})$.

Step 3: Check feasibility of $\tilde{G}(\tilde{x}(s|u^{\epsilon, \gamma^*}, t_f^{\epsilon, \gamma^*})) = 0$, $s \in [0, 1]$. If $(u^{\epsilon, \gamma^*}, t_f^{\epsilon, \gamma^*})$ is feasible, then go to Step 4. Otherwise, set $\gamma = \beta_1 \gamma$. If $\gamma \leq \bar{\gamma}$, we have an abnormal exit. Otherwise, go to Step 2.

Step 4: Set $\epsilon = \beta_2 \epsilon$ and $\gamma = \beta_2 \gamma$. If $\epsilon > \bar{\epsilon}$, then go to Step 2. Otherwise, output $(u^{\epsilon, \gamma^*}, t_f^{\epsilon, \gamma^*})$ and stop.

5 Numerical results

In the numerical computation, we write a Fortran subroutine, in which the gradient computation procedure presented in Algorithm 1 is combined with the optimization software NLPQLP—a Fortran implementation of SQP [18]—to solve the needed single-objective dynamic optimization problems. This subroutine used the 6th order Runge-Kutta method to solve the state and costate systems. Lagrange interpolation [19] is used whenever the Runge-Kutta method requires the value of the state or costate at an intermediate time between two adjacent knot points. In particular, the initial function, the bounds for the decision variables are $\phi(t, \zeta) = \zeta$, $a_1 = 5$, $b_1 = 20$, $c_1 = 0.02$, $c_2 = 400$, $d_1 = 0.202$, $d_2 = 500$, respectively. In Algorithm 1, we take $u = (0.102, 418.26)^\top$ and $t_f = 6.92$ as the initial values of decision variables, $\epsilon = 1.0 \times 10^{-2}$ and $\gamma = 2.5 \times 10^{-3}$ as the initial values for the approximating parameters, and parameters $\beta_1 = 0.5$, $\beta_2 = 0.1$, $\bar{\epsilon} = 1.0 \times 10^{-8}$ and $\bar{\gamma} = 1.0 \times 10^{-7}$.

To solve (BP) using the NNC method, we write a Fortran program combining the NNC method with the single-objective solution subroutine. By running this program with $m = 20$, evenly distributed points on the Pareto set is obtained as shown in Fig. 1. For comparison, we also write another Fortran program combining the CWS method, where the components of normalized objective vector (10) are weighted, with the single-objective solution subroutine to solve (BP). Running this program with $m = 20$ for evenly distributed weight, a non-uniform spread of points on the Pareto set is obtained as shown in Fig. 2. Note that a Pareto filter is designed to remove all non-globally Pareto solutions. This filter works by comparing a point in the presentation of the

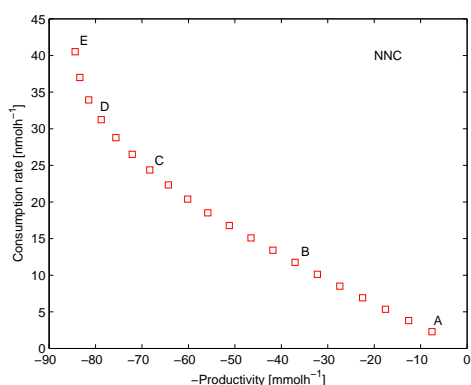


Fig. 1 Pareto set generated by NNC method.

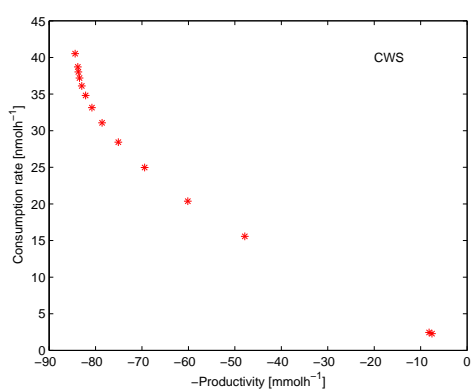


Fig. 2 Pareto set generated by CWS method.

Pareto set with every other generated point. If a point is dominated by other points in the representation of the Pareto set, then it is eliminated. Clearly, the Pareto set constructed from the points generated by the NNC method is much more accurate. Furthermore, it is observed that the computation time taken to generate the 20 points by the NNC method is 418.65 seconds. However, it takes 532.82 seconds by the CWS method. These simulations are carried out on a Thinkpad laptop computer with 2.8GHz CPU and 4GB memory. It can be seen that all representations of the Pareto set are generated within a few minutes. In addition to the fact that the NNC method generates a more accurate representation of the Pareto set than the CWS method, it also gives rise to faster computation time.

More specifically, five points A, B, C, D and E are taken from the representation of the Pareto set obtained by the NNC method (see Fig. 1). The corresponding weights, optimal solutions and objective values for points A, B, C, D and E are listed in Table 2. Obviously, these two objectives are conflict-

Table 2 The weights, optimal solutions and optimal objectives for points A, B, C, D and E.

	k	α_{1k}	α_{2k}	u^*	t_f^*	$J(u^*, t_f^*)$
A	1	0	1	$(0.0200, 499.8056)^\top$	5.0000	$(-7.5537, 2.2807)^\top$
B	7	0.3158	0.6842	$(0.1140, 418.2510)^\top$	5.0000	$(-37.0534, 11.7543)^\top$
C	14	0.6842	0.3158	$(0.2020, 418.2595)^\top$	6.0874	$(-68.3316, 24.3689)^\top$
D	17	0.8947	0.1053	$(0.2020, 437.2584)^\top$	7.8947	$(-81.4623, 33.9381)^\top$
E	20	1	0	$(0.2020, 475.0829)^\top$	9.5224	$(-84.3955, 40.5305)^\top$

ing. Emphasizing one of these two objectives can be achieved by selecting the weights α_{1k} and $1 - \alpha_{1k}$. In particular, maximizing productivity of 1,3-PD is focused when the value of α_{1k} is close to one. On the other hand, minimizing the consumption rate of glycerol is concentrated when the value of α_{1k} is close to zero. Moreover, the changes of the corresponding optimal concentrations of biomass, glycerol and 1,3-PD are plotted in Fig. 3. From Fig. 3, we see that the biomass for point D attains its upper bound (the horizontal red line) at the terminal time. In other words, the continuous state constraint is active at the terminal time. From Table 2 and Fig. 3, we can also confirm that the proposed solution method is effective in solving (BP).

6 Conclusions

In this paper, we investigated the bi-objective dynamic optimization of 1,3-PD production in a microbial batch process. We developed a numerical solution method to construct the Pareto set of the bi-objective dynamic optimization problem. This method was based on the NNC method in conjunction with the gradient-based single-objective solver. Numerical results verified the effectiveness of the proposed solution method.

Acknowledgements This work is supported by the Natural Science Foundation of China (No. 11201267), the Shandong Provincial Natural Science Foundation of China (Nos. ZR2015AL010, ZR2014FM029, ZR2015PG006, ZR2013AQ022) and the Australian Research Council (No. DP140100289).

References

1. Biebl, H., Menzel, K., Zeng, A.P., Deckwer, W.D.: Microbial production of 1,3-propanediol. *Appl. Microbiol. Biotechnol.* **52**, 289–297 (1999)
2. Chen, G., Wang, L., Loxton, R., Lin, Q.: Robust optimal control of a microbial batch culture process. *J. Optimiz. Theory App.* **167**, 342–362 (2015)
3. Cheng, K., Liu, H., Liu, D.: Multiple growth inhibition of *Klebsiella pneumoniae* in 1,3-propanediol fermentation. *Biotechnol. Lett.* **27**, 19–22 (2005)
4. Dai, C., Wang, Y., Ye, M.: A new multi-objective particle swarm optimization algorithm based on decomposition. *Inform. Sciences* **325**, 541–557 (2015)
5. Das, I., Dennis, J.E.: Normal-boundary intersection: A new method for generating the Pareto surface in nonlinear multicriteria optimization problems. *SIAM J. Optim.* **8**, 631–657 (1998)

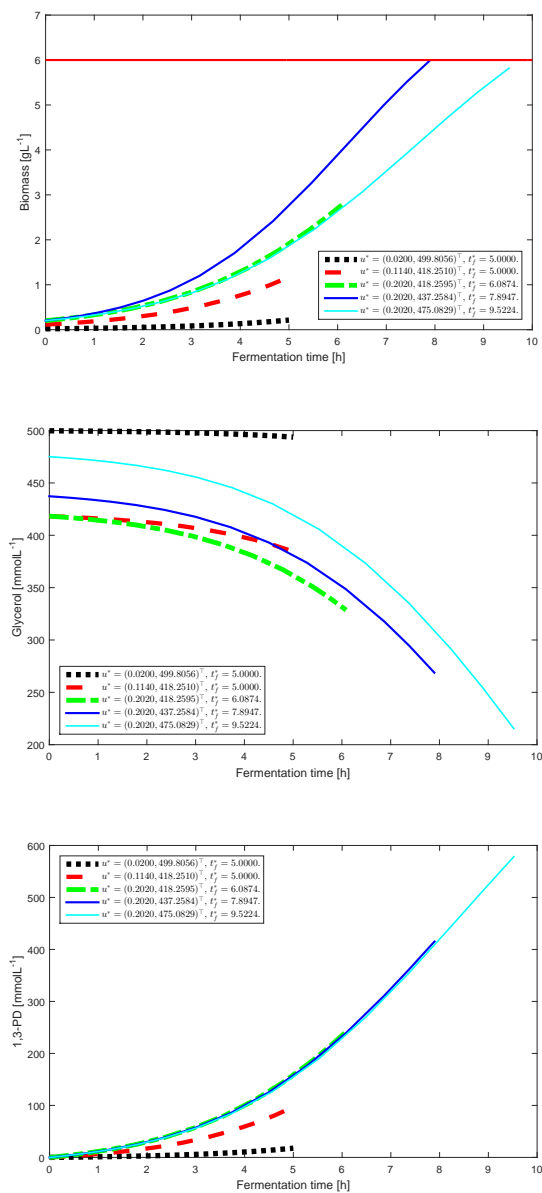


Fig. 3 The changes of concentrations of biomass, glycerol and 1,3-PD corresponding to the optimal solutions in Table 2.

6. Eichfelder, G.: An adaptive scalarization method in multiobjective optimization. *SIAM J. Optim.* **19**, 1694–1718 (2009)
7. Günzel, B.: Mikrobielle herstellung von 1,3-propandiol durch *Clostridium butyricum* und adsorptive Aufarbeitung von Diolen. Ph.D. Dissertation, TU Braunschweig, Germany, 1991
8. Hale, J.K., Verduyn Lune, S.M.: Introduction to Functional-Differential Equations, Springer-Verlag, Berlin, 1993
9. Hillermeier, C., Jahn, J.: Multiobjective optimization: Survey of methods and industrial applications. *Surv. Math. Ind.* **11**, 1–42 (2005)
10. Konak, A., Coit, D.W., Smith, A.E.: Multi-objective optimization using genetic algorithms: A tutorial. *Reliab. Eng. Syst. Safe.* **91**, 992–1007 (2006)
11. Liu, C.: Modelling and parameter identification for a nonlinear time-delay system in microbial batch fermentation. *Appl. Math. Model.* **37**, 6899–6908 (2013)
12. Liu, C., Loxton, R., Teo, K.L.: A computational method for solving time-delay optimal control problems with free terminal time. *Syst. Control Lett.* **72**, 53–60 (2014)
13. Logist, F., Houska, B., Diehl, M., Van Impe, J.: Fast Pareto set generation for nonlinear optimal control problems with multiple objectives. *Multidiscip. Optim.* **42**, 591–603 (2010)
14. Menzel, K., Zeng, A.P., Deckwer, W.D.: High concentration and productivity of 1,3-propanediol from continuous fermentation of glycerol by *Klebsiella Pneumoniae*. *Enzyme Microb. Technol.* **20**, 82–86 (1997)
15. Messac, A., Ismail-Yahaya, A., Mattson, C.A.: The normalized normal constraint method for generating the Pareto frontier. *Struct. Multidiscip. O.* **25**, 86–98 (2003)
16. Messac, A., Mattson, C.: Normal constraint method with guarantee of even representation of complete Pareto frontier. *AIAA J.* **42**, 2101–2111 (2004)
17. Nocedal, J., Wright, S.J.: Numerical Optimization, Springer-Verlag, New York, 1999
18. Schittkowski, K.: NLPQLP: A Fortran implementation of a sequential quadratic programming algorithm with distributed and non-monotone line search – user’s guide. University of Bayreuth, Bayreuth, Germany, 2007.
19. Stoer, J., Bulirsch, R.: Introduction to Numerical Analysis, Springer-Verlag, New York, 1980
20. Sun, Y.Q., Qi, W., Teng, H., Xiu, Z.L., Zeng, A.P.: Mathematical modeling of glycerol fermentation by *Klebsiella pneumoniae*: Concerning enzyme-catalytic reductive pathway and transport of glycerol and 1,3-propanediol across cell membrane. *Biochem. Eng. J.* **38**, 22–32 (2008)
21. Teo, K.L., Goh, C.J., Wong, K.H.: A Unified Computational Approach to Optimal Control Problems, Longman Scientific and Technical, Essex, 1991
22. Xiu, Z.L., Song, B.H., Sun, L.H., Zeng, A.P.: Theoretical analysis of effects of metabolic overflow and time delay on the performance and dynamic behavior of a two-stage fermentation process. *Biochem. Eng. J.* **11**, 101–109 (2002)
23. Xiu, Z.L., Zeng, A.P., An, L.J.: Mathematical modelling of kinetics and research on multiplicity of glycerol bioconversion to 1,3-propanediol. *J. Dalian Univ. Tech.* **40**, 428–433 (2000)
24. Yu, Y.S.: Optimal control of a nonlinear time-delay system in batch fermentation process. *Math. Probl. Eng.* **2014**, Article ID 478081, 7 pages (2014)
25. Yuan, J.L., Zhang, X., Zhu, X., Feng, E.M., Xiu, Z.L.: Pathway identification using parallel optimization for a nonlinear hybrid system in batch culture. *Nonlinear Anal. Hybrid Syst.* **15**, 112–131 (2015)
26. Zadeh, L.: Optimality and non-scalar-valued performance criteria. *IEEE Trans. Autom. Control* **8**, 59–60 (1963)
27. Zeng, A.P., Biebl, H.: Bulk chemicals from biotechnology: The case of 1,3-propanediol production and the new trends. *Adv. Biochem. Eng./Biotechnol.* **74**, 239–259 (2002)