Computers and Chemical Engineering 203 (2025) 109344

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/cace

Deep learning adaptive Model Predictive Control of Fed-Batch Cultivations

Niels Krausch [b], Martin Doff-Sotta [a], Mark Cannon [a,*], Peter Neubauer [b], Mariano Nicolas Cruz Bournazou [b]

[a] University of Oxford, Department of Engineering Science, Parks Road, Oxford, OX1 3PJ, UK
[b] Technische Universität Berlin, Institute of Biotechnology, Bioprocess engineering, Ackerstr. 76, Berlin, 13355, Germany

ARTICLE INFO

Keywords:
Nonlinear model predictive control
Robust adaptive control
Data-driven control
Convex optimization
Bioprocesses

ABSTRACT

Bioprocesses are often characterized by nonlinear and uncertain dynamics, posing particular challenges for model predictive control (MPC) algorithms due to their computational demands when applied to nonlinear systems. Recent advances in optimal control theory have demonstrated that concepts from convex optimization, tube MPC, and differences of convex functions (DC) enable efficient, robust online process control. Our approach is based on DC decompositions of nonlinear dynamics and successive linearizations around predicted trajectories. By convexity, the linearization errors have tight bounds and can be treated as bounded disturbances within a robust tube MPC framework. We describe a systematic, data-driven method for computing DC model representations using deep neural networks with a special convex structure, and explain how the resulting MPC optimization can be solved using convex programming. For the problem of maximizing product formation in a cultivation with uncertain model parameters, we design a controller that ensures robust constraint satisfaction and allows online estimation of unknown model parameters. Our results indicate that this method is a promising solution for computationally tractable, robust MPC of bioprocesses.

1. Introduction

The growing demand for cost-effective production of biologic drugs and sustainable production of goods has intensified the need for rapid development and control of bioprocesses (Rathore et al., 2021). This has motivated robotic parallel experimentation facilities capable of performing sophisticated cultivations (e.g. fed-batch, continuous) in high throughput. Some commercially available solutions are the robolector from m2p-labs, bioREACTOR 48 from 2mag, ambr250 from Sartorius, and bioXplorer from H.E.L.

Effective bioprocess development requires control algorithms that are robust to model uncertainty arising from incomplete knowledge of physical and biological mechanisms. Particularly in the early stages of a project, development is characterized by limited process information and large experimental design spaces, posing significant challenges for parallel bioreactor operation (Cruz Bournazou et al., 2017). Adaptive and robust Model Predictive Control (MPC) algorithms can potentially improve performance and experimental efficiency, even in cases of large uncertainty in model outputs (Krausch et al., 2022; Jabarivelisdeh et al., 2020; Tuveri et al., 2023). However, applications of MPC in this context have largely been restricted to nominal process models with no explicit accounting for parameter uncertainty and disturbances. For example, Kager et al. (2020) increased total product formation in a fungal process, but their approach is limited to the nominal case.

Bioprocess control using reinforcement learning often requires few prior modelling assumptions (e.g. Mowbray et al., 2022), but relies on heavy offline training and large data sets to handle process uncertainties, and is typically unable to guarantee robust constraint satisfaction. On the other hand, hybrid modelling approaches exploit available information on uncertainty by combining empirical models derived from data with first principles models (Mahanty, 2023; Shah et al., 2022; Bangi et al., 2022). Combined with robust MPC, these models can ensure robust constraint satisfaction when explicit uncertainty bounds are available. However MPC in this context involves the online solution of a nonlinear program (NLP) with high computational complexity. Moreover, the amount of computation required can in practice undermine guarantees of robust performance and may not be suitable for cultures of fast-growing microorganisms with short feeding control times.

The robust MPC strategy of this paper allows efficient optimization of controller actions and provides guarantees of robustness by design. We use a tube MPC algorithm to compute bounds on future system trajectories. This provides guarantees of robust stability and constraint

* Corresponding author.
E-mail addresses: niels.krausch@yokogawa.com (N. Krausch), martin.doff-sotta@eng.ox.ac.uk (M. Doff-Sotta), mark.cannon@eng.ox.ac.uk (M. Cannon), peter.neubauer@tu-berlin.de (P. Neubauer), mariano.n.cruzbournazou@tu-berlin.de (M.N. Cruz Bournazou).

https://doi.org/10.1016/j.compchemeng.2025.109344
Received 3 February 2025; Received in revised form 25 July 2025; Accepted 11 August 2025
Available online 20 August 2025

satisfaction in the presence of bounded modelling errors, disturbances and parametric uncertainty. Our approach uses differences of convex (DC) functions to represent the model dynamics on a compact set, allowing the NLP to be solved iteratively via a sequence of convex optimization problems (Lipp and Boyd, 2016). This ensures recursive feasibility and convergence of the iteration while requiring the solution of only one convex problem at each sampling instant.

The proposed methodology enables the construction of bioprocess models in DC form directly from data, or from a combination of data and physical models. Any process described by continuous dynamics can be approximated in DC form (Horst and Thoai, 1999). We describe a systematic method of learning DC representations by using pairs of input-convex neural networks (ICNNs) (Amos et al., 2017; Sankaranarayanan and Rengaswamy, 2022) to learn the model dynamics. The approach builds on Doff-Sotta and Cannon (2022), Buerger et al. (2024), and extends these by using more powerful ICNN models and by adopting a hybrid modelling approach that enables online parameter estimation. Combined with set-membership parameter estimation, this provides a robust adaptive MPC algorithm. The parameterization of trajectory bounds influences the computational complexity of the associated tube MPC optimization problem. Doff-Sotta and Cannon (2022) and Buerger et al. (2024), use elementwise bounds, but this requires $O(2^{n_x})$ inequality constraints in the MPC optimization (n_x denotes the model state dimension), causing significant computational burden for large systems. We show that simplex tubes, for which the dependence is $O(n_x)$, provide an effective alternative.

We describe simulations of a fed-batch bioreactor for the production of penicillin. This case study uses ICNNs to determine the model dynamics in DC form and exploits knowledge of parts of the model structure to enable estimation of parameters that are typically uncertain. We demonstrate the potential of the approach for optimizing product formation and identifying model parameters while robustly satisfying constraints despite the presence of unknown but bounded model disturbances.

2. Modelling and DC approximation using deep learning

We consider a perfectly mixed isothermal fed-batch bioreactor, a popular case study example from (Srinivasan et al., 2003). The model states are the cell concentration X [$\mathrm{g\,L^{-1}}$], product concentration P [$\mathrm{g\,L^{-1}}$], substrate concentration S [$\mathrm{g\,L^{-1}}$] and volume V [L]. The control input is the feed flow rate u [$\mathrm{L\,h^{-1}}$] of substrate at concentration S_i. The objective of the control strategy is to maximize the product concentration P after a period of T hours while satisfying state and control constraints.

The system model is given by

$$\dot{x} = \phi(x,u) + \xi \tag{1}$$

where $\dot{x} = \mathrm{d}x/\mathrm{d}t$, $x = (X,S,P,V)$ is the system state and ξ is an unknown bounded disturbance representing measurement noise and parameter errors not explicitly accounted for in the problem description. For $(x,u) \in \mathbb{X}\times\mathbb{U}$, where $\mathbb{X}$, $\mathbb{U}$ are compact convex state and control constraint sets, the function ϕ has the form

$$\dot{X} = \mu(S)X - \frac{u}{V_0+V}X \tag{2a}$$

$$\dot{S} = -\mu(S)\frac{X}{Y_{X/S}} - v\frac{X}{Y_{P/S}} + \frac{u}{V_0+V}(S_i - S) \tag{2b}$$

$$\dot{P} = vX - \frac{u}{V_0+V}P \tag{2c}$$

$$\dot{V} = u \tag{2d}$$

$$\mu(S) = \mu_{\max}\frac{S}{S+K_S+S^2/K_i}. \tag{2e}$$

Here $\mu_{\max}$ denotes the maximal growth rate ($0.02\,\mathrm{h^{-1}}$), K_S is the substrate affinity constant of the cells ($0.05\,\mathrm{g\,L^{-1}}$), K_i is an inhibition constant which inhibits growth at high substrate concentrations ($5\,\mathrm{g\,L^{-1}}$), v is the production rate ($0.004\,\mathrm{L\,h^{-1}}$), the inlet substrate concentration is $S_i = 200\,\mathrm{g\,L^{-1}}$, $Y_{X/S}$ is the yield coefficient of biomass per substrate ($0.4\ \mathrm{g}_X\ \mathrm{g}_S^{-1}$), and $Y_{P/X}$ is the yield coefficient of product per substrate ($1.2\ \mathrm{g}_P\ \mathrm{g}_S^{-1}$). The initial volume is $V_0 = 120\,\mathrm{L}$. We assume $Y_{X/S}$ and S_i are not exactly known but have known bounds as a result of variability in the microorganism phenotype and uncertainty in feed concentration due to imperfect measurements or mixing, respectively.

The initial conditions are $X(0) = 1\,\mathrm{g\,L^{-1}}$, $S(0) = 0.5\,\mathrm{g\,L^{-1}}$, $P(0) = 0\,\mathrm{g\,L^{-1}}$, and $V(0) = 0\,\mathrm{L}$, and the constraint sets are

$$\mathbb{X} = \{x : 0 \le x \le (3.7,\ 1,\ 3,\ 5)\}, \quad \mathbb{U} = \{u : 0 \le u \le 0.2\}. \tag{3}$$

A feedforward deep learning NN framework was used to approximate the dynamics ((2)a)–((2)e) as a difference of convex functions by subtracting the outputs of two feedforward ICNN subnetworks. An ICNN with L layers is characterized by a parameter set $p = \{M_{1:L}, Q_{0:L-1}, b_{0:L}\}$ and an input–output map $z_{L+1} = f(y;p)$, defined (with $(M_0, z_0) \equiv 0$) by

$$z_{l+1} = \sigma(M_l z_l + Q_l y + b_l), \quad l = 0,\dots,L-1, \tag{4}$$

and $z_{L+1} = M_L z_L + b_L$, where y is the input, z_l is the layer activation, M_l are non-negatively constrained weights ($\{M_l\}_{ij} \ge 0$, for all i,j,l), Q_l are input passthrough weights, b_l are bias terms and $\sigma(\cdot)$ is a componentwise convex, nondecreasing activation function. We use ReLU activation functions: $\sigma(\xi) = \max(\xi, 0)$. Each network layer is the composition of a linear function with a nondecreasing convex function; hence $z_{L+1} = f(y;p)$ is necessarily a convex map from y to z_{L+1}.

We define $y = (x,u)$, where $x = (X,S,P,V)$ and u are the state and control input of the system (1). The outputs of two ICNNs are subtracted and trained simultaneously to learn the dynamics of (1) as a difference of convex functions:

$$\dot{x} = f_1(x,u) - f_2(x,u) + \zeta, \tag{5}$$

where ζ is a bounded disturbance that accounts for the approximation error in $f_1 - f_2$ and the disturbance ξ in (1). Each of the two ICNNs consists of an input layer, L hidden layers with N_{node} activation functions ($z_l \in \mathbb{R}^{N_{\text{node}}}$), and an output layer. In this study $\phi(x,u)$ is approximated by $M_{L,1}z_{L,1} - M_{L,2}z_{L,2} + b_{L,1} - b_{L,2}$ where $z_{L,i}$, $i = 1$ or 2 is given by (4) with $L = 2$, $N_{\text{node}} = 32$.

The network was implemented using Keras (Chollet et al., 2015) and trained using the RMSProp optimizer with 10^5 samples of ((2)a)–((2)d) distributed across the compact sets $\mathbb{X}$ and $\mathbb{U}$, split into 80% training, 20% validation sets. To enable evaluation of each sampling point, the unknown parameters $Y_{X/S}$ and S_i are set to nominal values, denoted $\bar{Y}_{X/S}$, $\bar{S}_i$. Convexity of the models was evaluated by checking the Hessian matrices: $\nabla^2 f_i(x,u) \succeq 0$, $i = 1,2$, at points (x,u) in the validation set. Fig. 1 shows the DC decompositions representing $\dot{X}$ and $\dot{S}$ for given values of P, V, and u. As illustrated, the NN provides a good fit (MAE: 0.0012) for the model (2) (blue dots and blue surface), and the DC form of the decomposition is apparent in Fig. 1 (orange and green surfaces).

3. DC-TMPC framework with simplexes

Doff-Sotta and Cannon (2022) proposed a robust tube MPC algorithm based on successive linearization for DC systems. The so-called DC-TMPC algorithm capitalizes on the idea that optimization problems with inequality constraints involving DC functions can be approximated as convex problems by linearizing only the concave parts of the inequality constraints. Furthermore, bounds on the errors introduced by linearizing concave functions are trivial to compute and can be treated as disturbances in a robust MPC scheme. We apply this approach to the system (5) learned in DC form with additive disturbances, and we extend it by considering state tubes defined in terms of simplexes in order to reduce online computational load.

We can derive a discrete time model from (5) using, for example, forward Euler approximation:

$$x_{k+1} = \delta\big(f_1(x_k,u_k) - f_2(x_k,u_k)\big) + w_k, \tag{6}$$

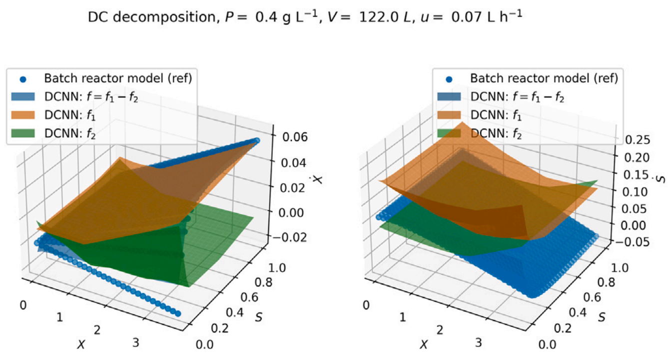

Fig. 1. DC decomposition: the actual model evaluated at 30 × 30 sample points (blue dots), the convex functions f_1 and f_2 (orange and green surfaces), and the DC decomposition $f = f_1 - f_2$ (blue surfaces) approximating $\dot{X}$ (left), and $\dot{S}$ (right), as functions of the concentrations X and S, with product concentration $P = 0.4\,\mathrm{g\,L^{-1}}$, volume $V = 122\,\mathrm{L}$, and feed rate $u = 0.07\,\mathrm{L\,h^{-1}}$.

where y_k denotes the value of variable y at discrete time index $k \in \{0, 1, 2, \dots\}$ and the sampling interval is $\delta = 1\,\text{hour}$. The disturbance w_k lies in a bounded set $\mathbb{W}$ for all k. The disturbance set $\mathbb{W}$ can be computed from bounds on the disturbance affecting the continuous time system (1) and the maximum pointwise error in the DC representation $f_1 - f_2$ of ϕ.

The tube MPC strategy is based on linearizing f_1 and f_2 in (6) around nominal predicted trajectories $\{x_k^\circ, u_k^\circ\}_{k=0:N}$ over a future horizon of N discrete time steps. For a given control sequence $\{u_k^\circ\}_{k=0:N}$, we compute $\{x_k^\circ\}_{k=0:N}$ by applying an ODE solver to the model (1) with nominal parameters (i.e. $\xi = 0$, $Y_{X/S} = \bar{Y}_{X/S}$, $S_i = \bar{S}_i$). We denote the state and input perturbations relative to x_k°, u_k° as $s_k = x_k - x_k^\circ$ and $v_k = u_k - u_k^\circ$ respectively. To reduce the effects of model uncertainty on predicted trajectories, v_k is defined by a two degree-of-freedom control law: $v_k = K_k s_k + c_k$, where $\{K_k\}_{k=0:N}$ are feedback gains and $\{c_k\}_{k=0:N}$ is a feedforward control sequence computed at each discrete time step by solving a sequence of convex optimization problems.

Bounds on the predicted trajectories of the model (6) are obtained by exploiting the componentwise convexity of f_i, $i = 1, 2$, which implies that each component of f_i is upper-bounded by the Jacobian linearization of f_i about any point in its domain. This provides the following bounds, which are by construction convex conditions on $\{s_k\}_{k=0:N}$ and $\{c_k\}_{k=0:N}$,

$$\begin{aligned} s_{k+1} \geq s_k &+ \delta\big(f_1(x_k^\circ + s_k, u_k^\circ + v_k) - f_1(x_k^\circ, u_k^\circ)\big) \\ &- F_{2,k} s_k - B_{2,k} c_k + \delta \max_{w \in \mathbb{W}} w_k \end{aligned} \tag{7a}$$

$$\begin{aligned} s_{k+1} \leq s_k &+ F_{1,k} s_k + B_{1,k} c_k \\ &- \delta\big(f_2(x_k^\circ + s_k, u_k^\circ + v_k) - f_2(x_k^\circ, u_k^\circ)\big) + \delta \min_{w \in \mathbb{W}} w_k \end{aligned} \tag{7b}$$

where $F_{i,k} = \delta(A_{i,k} + B_{i,k} K)$, $A_{i,k} = \frac{\partial f_i}{\partial x}(x_k^\circ, u_k^\circ)$, $B_{i,k} = \frac{\partial f_i}{\partial u}(x_k^\circ, u_k^\circ)$ for $i = 1, 2$.

To ensure robust constraint satisfaction and convergence of the successive linearization scheme, we define an uncertainty tube consisting of a sequence of sets $\{\mathbb{S}_k\}_{k=0:N}$ bounding the state perturbation trajectories under all realizations of model uncertainty. The tube $\{\mathbb{S}_k\}_{k=0:N}$ may be defined in terms of componentwise bounds (e.g. Doff-Sotta and Cannon, 2022), but the number of constraints needed then grows exponentially with the state dimension n_x. To reduce this to a linear dependence, we instead parameterize $\{\mathbb{S}_k\}_{k=0:N}$ in terms of simplexes:

$$\mathbb{S}_k = \left\{ s \in \mathbb{R}^{n_x} : \begin{bmatrix} -I \\ \underline{1}^\top \end{bmatrix} s \leq \begin{bmatrix} \alpha_k \\ \beta_k \end{bmatrix} \right\} \tag{8}$$

where $I \in \mathbb{R}^{n_x \times n_x}$ is the identity matrix, $\underline{1}$ is a vector of ones, and the vector α_k and scalar β_k are optimization variables. Conditions on $\{\alpha_k, \beta_k\}_{k=0:N}$ to ensure that the state perturbation s_k belongs to $\mathbb{S}_k$ for all $k \in \{0, \dots, N\}$ can be defined recursively as follows. Combining (7) and (8), we require, for all $s \in \mathbb{S}_k$,

$$\begin{aligned} \alpha_{k+1} \geq \max_{s \in \mathbb{S}_k} \Big(&-s + \delta f_2(x_k^\circ + s, u_k^\circ + K_k s + c_k) \\ &- \delta f_2(x_k^\circ, u_k^\circ) - F_{1,k} s - B_{1,k} c_k \Big) - \delta \min_{w \in \mathbb{W}} w \end{aligned} \tag{9a}$$

$$\begin{aligned} \beta_{k+1} \geq \max_{s \in \mathbb{S}_k} \underline{1}^\top \Big(&s + \delta f_1(x_k^\circ + s, u_k^\circ + K_k s + c_k) \\ &- \delta f_1(x_k^\circ, u_k^\circ) - F_{2,k} s - B_{2,k} c_k \Big) + \delta \max_{w \in \mathbb{W}} \underline{1}^\top w. \end{aligned} \tag{9b}$$

Conditions ((9)a)–((9)b) are equivalent to $n_x + 1$ convex inequalities in the vertices of $\mathbb{S}_k$, denoted $\{s_k^j\}_{j=0:n_x}$, given by

$$s_k^j = \begin{cases} -\alpha_k & j = 0 \\ -\alpha_k + (\beta_k + \underline{1}^\top \alpha_k)\, \underline{e}_j & j = 1, \dots, n_x \end{cases} \tag{10}$$

where $\underline{e}_j$ is the jth column of the identity matrix in $\mathbb{R}^{n_x \times n_x}$. Exploiting again the convexity of f_1 and f_2, conditions ((9)a)–((9)b) are therefore equivalent to the constraints, for $j = 0, 1, \dots, n_x$:

$$\begin{aligned} \alpha_{k+1} \geq &-s_k^j + \delta f_2(x_k^\circ + s^j, u_k^\circ + K_k s_k^j + c_k) - \delta f_2(x_k^\circ, u_k^\circ) \\ &- F_{1,k} s_k^j - B_{1,k} c_k - \delta w_{\min} \\ \beta_{k+1} \geq \underline{1}^\top \Big(&s_k^j + \delta f_1(x_k^\circ + s_k^j, u_k^\circ + K_k s_k^j + c_k) - \delta f_1(x_k^\circ, u_k^\circ) \\ &- F_{2,k} s_k^j - B_{2,k} c_k \Big) + \delta w_{\max} \end{aligned}$$

with $\underline{e}_i^\top w_{\min} = \min_{w \in \mathbb{W}} \underline{e}_i^\top w$ for all i and $w_{\max} = \max_{w \in \mathbb{W}} \underline{1}^\top w$.

4. MPC algorithm without parameter estimation

This section considers the case in which the parameters of (2) are known to the controller but the model (1) is affected by unknown bounded disturbances. We therefore assume that the discrete time model (6) has no parametric uncertainty (in particular $Y_{X/S}$ and S_i are equal to their nominal values $\bar{Y}_{X/S}$, $\bar{S}_i$), but is subject to disturbances with known bounds $w_k \in \mathbb{W}$ for all k. At each sampling instant $t = 0, \delta, 2\delta, \dots$, given the current state $x(t)$ of (2), we optimize the feedforward sequence $\{c_k\}_{k=0:N}$ and tube parameters $\{\alpha_k, \beta_k\}_{k=0:N}$ subject to ((9)a,b) and $x_k \in \mathbb{X}$, $u_k \in \mathbb{U}$ for all k by solving the following convex problem:

$$\underset{c_k, \alpha_k, \beta_k, k=0,\dots,N}{\text{maximize}} \sum_{k=0}^{N} (l_{x,k} - \gamma l_{u,k}) \tag{11a}$$

subject to, for all $k \in \{0, \dots, N-1\}$ and $j \in \{0, \dots, n_x\}$

$$x_k^\circ + s_k^j \in \mathbb{X}, \quad u_k^\circ + K s_k^j + c_k \in \mathbb{U} \tag{11b}$$

$$\begin{aligned} \alpha_{k+1} \geq &-s_k^j + \delta f_2(x_k^\circ + s_k^j, u_k^\circ + K_k s_k^j + c_k) - \delta f_2(x_k^\circ, u_k^\circ) \\ &- F_{1,k} s_k^j - B_{1,k} c_k - \delta w_{\min} \end{aligned} \tag{11c}$$

$$\begin{aligned} \beta_{k+1} \geq \underline{1}^\top \Big(&s_k^j + \delta f_1(x_k^\circ + s_k^j, u_k^\circ + K_k s_k^j + c_k) - \delta f_1(x_k^\circ, u_k^\circ) \\ &- F_{2,k} s_k^j - B_{2,k} c_k \Big) + \delta w_{\max} \end{aligned} \tag{11d}$$

$$l_{x,k} \leq \underline{e}_3^\top (x_k^\circ + s_k^j) \tag{11e}$$

$$l_{u,k+1} \geq (u_{k+1}^\circ + K s_{k+1}^j + c_{k+1} - u_k^\circ - K s_k^j - c_k)^2 \tag{11f}$$

and the initial constraints, for all $j \in \{0, \dots, n_x\}$

$$l_{u,0} \geq \left(u_0^\circ + K s_0^j + c_0 - u(t - \delta)\right)^2 \tag{11g}$$

$$\alpha_0 \geq x_0^\circ - x(t) \tag{11h}$$

$$\beta_0 \geq \underline{1}^\top \left(x(t) - x_0^\circ\right) \tag{11i}$$

and terminal constraints for all $k \in \{0, \dots, N_{\text{term}}\}$, $j \in \{0, \dots, n_x\}$

$$c_{N-k} = 0 \tag{11j}$$

$$x_{N-k}^\circ + s_{N-k}^j \leq x_{\text{term}}. \tag{11k}$$

A brief explanation of the problem formulation is as follows:

1. The objective function ((11)a) aims to maximize the product concentration via the term $l_{x,k}$, which is a lower bound on $P = \underline{e}_3^\top x$ due to ((11)e). The term $l_{u,k}$ smooths the solution by penalizing changes in the control input via ((11)f,g). The scalar constant γ balances these two competing objectives.
2. The state and control constraints in ((11)b) depend on the sets $\mathbb{X}$, $\mathbb{U}$ defined in (3). By definition all model states and control inputs must be non-negative, while the upper bounds $u \le 0.2$ and $x \le (3.7,\ 1,\ 3,\ 5)$ are context-specific.
3. The tube constraints ((11)c,d) are equivalent to ((9)a,b) and, with s_k^j defined in (10), ensure that $s_k \in \mathbb{S}_k$ for $1 \le k \le N$.
4. The initial constraints ((11)h,i) ensure $s_0 = x(t) - x_0^\circ \in \mathbb{S}_0$.
5. The terminal constraints ((11)j,k) allow for a guarantee that problem (11) remains feasible for all $t > 0$ if it is feasible at $t = 0$. Here N_{term} and $x_{\text{term}} \in \mathbb{X}$ are chosen so that the set $\{x_0 : x_k \le x_{\text{term}},\ k = 0, \dots, N_{\text{term}}\}$ is positively invariant.

If the substrate concentration S remains low for an extended period, the bioreactor effectively shuts down due to the depletion of the main carbon source. Using data from simulations of (1)–(2), we identify this condition as $S \le 0.1\,\mathrm{g\,L^{-1}}$ for a period of 5 h, and hence we define $x_{\text{term}} = (3.7,\ 0.1,\ 3,\ 5)$ and $N_{\text{term}} = 5$ in the terminal constraints ((11)k,l).

The solution, denoted $\{c_k^*\}_{k=0:N}$, of (11) is used to update the state and input trajectories $\{x_k^\circ, u_k^\circ\}_{k=0:N}$ to be employed at the next iteration by setting $s_0 \leftarrow x(t) - x_0^\circ$ and for $k = 0, 1, \dots, N$:

$$u_k^\circ \leftarrow u_k^\circ + K_k s_k + c_k^* \tag{12a}$$

$$s_{k+1} \leftarrow \Phi(x_k^\circ, u_k^\circ) - x_{k+1}^\circ \tag{12b}$$

$$x_{k+1}^\circ \leftarrow \Phi(x_k^\circ, u_k^\circ) \tag{12c}$$

where $\Phi(x_k, u_k)$ is the solution for x_{k+1} obtained from (1) under nominal conditions ($\xi = 0$, $Y_{X/S} = \bar{Y}_{X/S}$, $S_i = \bar{S}_i$) using an ODE solver (e.g. cvodes (Hindmarsh et al., 2005) implemented in CasADi (Andersson et al., 2019)).

The optimization (11) and update (12) are repeated until the control perturbation $\sum_k \|c_k^*\|$ and change in optimal cost fall below given thresholds, or until a maximum number of iterations is reached. The control input is then applied to the bioreactor as $u(\tau) = u_0^\circ$ for $\tau \in [t, t+\delta)$. At the next sampling instant, $t+\delta$, we set $x_0^\circ = x(t+\delta)$ and redefine $\{u_k^\circ, x_k^\circ\}_{k=0:N}$ using the optimal solution at the final iteration of time t by setting $s_0 \leftarrow 0$ and

$$u_k^\circ \leftarrow u_{k+1}^\circ + K_{k+1} s_k + c_{k+1}^*, \quad k = 0, 1, \dots, N-1 \tag{13a}$$

$$s_{k+1} \leftarrow \Phi(x_k^\circ, u_k^\circ) - x_{k+1}^\circ, \quad k = 0, 1, \dots, N \tag{13b}$$

$$x_{k+1}^\circ \leftarrow \Phi(x_k^\circ, u_k^\circ), \quad k = 0, 1, \dots, N \tag{13c}$$

with $u_{N+1}^\circ \leftarrow u^r$, where u^r is such that $x(t) \le x_{\text{term}}$ for all $t > 0$ if $x(0) \le x_{\text{term}}$ and $u(t) = u^r$. With $x_{\text{term}} = (3.7,\ 0.1,\ 3,\ 5)$ we find $u_r = 0.01\,\mathrm{L\,h^{-1}}$ meets this requirement for the system (1).

Since (12) and (13) use the nominal system model to update $\{u_k^\circ, x_k^\circ\}_{k=0:N}$, it cannot be guaranteed that the updated trajectory will result in a feasible updated problem (11). An alternative update strategy that provides a guarantee of feasibility is proposed in Lishkova and Cannon (2025). However, we use here a simpler approach, exploiting the observation that the nominal trajectory $\{u_k^\circ, x_k^\circ\}_{k=0:N}$ computed at a feasible previous iteration necessarily provides a set of linearization points such that problem (11) is feasible at the current iteration. This is a direct consequence of the tube $\{\mathbb{S}_k\}_{k=0:N}$ containing the state trajectories for all possible uncertainty realizations. Thus, if (11) is infeasible after the update (13), we instead define $\{x_k^\circ, u_k^\circ\}_{k=0:N}$ as a time-shifted version of the previous nominal trajectory:

$$u_k^\circ \leftarrow u_{k+1}^\circ, \quad k = 0, 1, \dots, N-1 \tag{14a}$$

$$x_k^\circ \leftarrow x_{k+1}^\circ, \quad k = 0, 1, \dots, N \tag{14b}$$

$$u_N^\circ \leftarrow u^r \tag{14c}$$

$$x_{N+1}^\circ \leftarrow \Phi(x_k^\circ, u_N^\circ) \tag{14d}$$

The update strategy in (12)–(14) ensures that (11) is recursively feasible, i.e. a feasible solution of (11) is obtained at each iteration and for all $t > 0$ if (11) is feasible at $t = 0$. The algorithm can be initialized at $t = 0$ by setting $x_0^\circ = x(0)$, $u_k^\circ = u^r$ and $c_k^* = 0$ for all k, and computing $\{x_k^\circ\}_{k=1:N}$ using (12).

By construction, the DC-TMPC strategy satisfies the constraints $x(n\delta) \in \mathbb{X}$, $u(n\delta) \in \mathbb{U}$ at all sampling instants $n \ge 0$ regardless of the number of iterations performed at each time step. Hence computation can be reduced to a single solution of the convex problem (11) at each time step. On the other hand, if there is no limit on the number of iterations, then for the limiting case of no model uncertainty ($\mathbb{W} \to \{0\}$), the DC-TMPC iteration converges to an optimal solution for the problem of maximizing product concentration for the system (1)–(2) subject to $(x, u) \in \mathbb{X} \times \mathbb{U}$ (see Lishkova and Cannon, 2025).

5. Adaptive MPC with online parameter estimation

This section discusses robust adaptive DC-TMPC subject to parameter uncertainty and unknown disturbances with known bounds. Since the uncertain parameters S_i and $Y_{X/S}^{-1}$ appear linearly in the bioreactor model (2), we can express (1) in a form that depends linearly on an unknown parameter vector θ,

$$\dot{x} = \phi(x, u) + \underline{e}_2 \psi(x, u)^\top \theta + \xi. \tag{15}$$

Here $\phi(x, u)$ is defined by ((2)a)–((2)e) with $Y_{X/S}, S_i$ replaced by their nominal values $\bar{Y}_{X/S}, \bar{S}_i$, and $\psi(x, u)$ and θ are the vectors

$$\psi(x, u) = \begin{bmatrix} -\mu(S)X \\ 100u/(V + V_0) \end{bmatrix}, \quad \theta = \begin{bmatrix} Y_{X/S}^{-1} - \bar{Y}_{X/S}^{-1} \\ (S_i - \bar{S}_i)/100 \end{bmatrix}.$$

Extending the approach of Sections 2–4 to account for uncertainty in θ, we propose a robust DC-TMPC strategy for controlling (15). This is combined with online estimation of θ to create a robust adaptive control algorithm.

Analogously to the DC model (6), we construct a DC representation $\psi(x, u) = g_1(x, u) - g_2(x, u)$, where $g_i(x, u)$ for $i = 1, 2$ is componentwise convex in (x, u), using the difference of two ICNNs. For this example we use feedforward NNs with the same structure as the ICNNs described in Section 2, but with 16 nodes in each hidden layer. The ICNNs are trained simultaneously on samples of the map $\psi(x, u)$ without knowledge of the true parameter values. Combining this model with (6), which was identified using the nominal parameters $\bar{Y}_{X/S}, \bar{S}_i$, we obtain

$$\begin{aligned} x_{k+1} &= \delta\big(f_1(x_k, u_k) - f_2(x_k, u_k)\big) \\ &\quad + \delta \underline{e}_2 \big(g_1(x_k, u_k) - g_2(x_k, u_k)\big)^\top \theta + w_k. \end{aligned} \tag{16}$$

A set $\mathbb{W}$ bounding the disturbance w_k can be constructed from bounds on the disturbance affecting the system (1) and the approximation errors in the DC representations of ϕ and ψ.

An initial bounding set Θ_0 containing θ is determined by prior parameter bounds. For example, with nominal parameter values $\bar{Y}_{X/S} = 0.45$, $\bar{S}_i = 195\,\mathrm{g\,L^{-1}}$, and bounds $Y_{X/S} \in [0.333, 0.5]$, $S_i \in [180, 220]$, we obtain

$$\theta \in \Theta_0 = \{\theta : (-0.222, -0.15) \le \theta \le (0.778, 0.25)\}, \tag{17}$$

and the actual parameter values $Y_{X/S} = 0.4$, $S_i = 200\,\mathrm{g\,L^{-1}}$ imply $\theta^* = (0.278, 0.05)$ is the true value of θ.

We use online set-membership estimation (Lorenzen et al., 2019; Lu et al., 2021) to update bounds on θ using online state measurements and prior disturbance bounds. At time $t = n\delta$, for $n = 0, 1, \dots$, the parameter set $\Theta_n = \{\theta : H\theta \le h_n\}$ is defined in terms of a vector h_n that is updated online and a constant matrix H that is chosen offline so that Θ_n has a fixed number of vertices. Thus if $H = [I\ \ -I]^\top$ so that h_n determines upper and lower bounds on the elements of θ, then Θ_n has 4 vertices. Let $D_n = D(x_n, u_n) = \delta \underline{e}_2 \big(g_1(x_n, u_n) - g_2(x_n, u_n)\big)^\top$ and $d_n = d(x_n, u_n) = \delta\big(f_1(x_n, u_n) - f_2(x_n, u_n)\big)$. Then (16) can be written as $x_{n+1} = D_n\theta + d_n + w_n$, and at time step $n+1$, θ must belong to an unfalsified parameter set given by $\{\theta : x_{n+1} - d_n - D_n\theta \in \mathbb{W}\}$. Hence, using observations of x_n over a window of N_{est} discrete time steps, we update the set Θ_n by solving a linear program,

$$h_{n,i} = \max_{\theta\in\Theta_{n-1}} H_i\,\theta$$

$$\text{subject to } x_{n-k+1} - d_{n-k} - D_{n-k}\theta \in \mathbb{W},\ \forall k \in \{1,\dots,N_{\text{est}}\}$$

for each element i of h_n, where $h_{n,i}$, H_i are the ith element and ith row of h_n, H. This update rule ensures that $\Theta_n \subseteq \cdots \subseteq \Theta_0$ and Θ_n converges asymptotically to the true parameter vector θ^* if the control input is persistently exciting (Lu et al., 2021).

We also compute a point estimate $\hat{\theta}_n$ online using Least Mean Squares (LMS) (Lorenzen et al., 2019). Given the nominal 1-step ahead prediction from (16), $\hat{x}_{1|n} = D_n\hat{\theta}_n + d_n$, and the actual system state x_{n+1}, we compute the update $\hat{\theta}_{n+1}$ at step $n+1$ as

$$\hat{\theta}_{n+1} = \Pi_{\Theta_{n+1}}\big[\hat{\theta}_n + \hat{\mu} D_n^\top (x_{n+1} - \hat{x}_{1|n})\big],$$

where $\Pi_\Theta[\cdot]$ denotes the (Euclidean) projection onto Θ, and $\hat{\mu}$ is an update gain satisfying $\frac{1}{\hat{\mu}} > \|D(x,u)\|^2$ for all $(x,u) \in \mathbb{X}\times\mathbb{U}$.

The parameter set estimate Θ_n is used to make the DC-TMPC optimization (11) robust to parameter uncertainty, while the point estimate $\hat{\theta}_n$ defines the nominal predicted trajectories $\{x_k^\circ, u_k^\circ\}_{k=0:N}$ at time $t = n\delta$. The resulting adaptive DC-TMPC algorithm has guaranteed robustness to parametric uncertainty and potentially better performance than the strategy of Section 4, albeit at the expense of increased computation.

In particular, when computing the nominal predicted trajectories $\{x_k^\circ, u_k^\circ\}_{k=0:N}$ the map $\Phi(x,u)$ in (12)–(14) is determined by solving (1) with $\xi = 0$ and the current parameter estimates

$$Y_{X/S} = (\hat{\theta}_{n,1} + \bar{Y}_{X/S}^{-1})^{-1}, \quad S_i = 100\,\hat{\theta}_{n,2} + \bar{S}_i,$$

where $\hat{\theta}_n = (\hat{\theta}_{n,1}, \hat{\theta}_{n,2})$. In addition, the tube $\{x_k^\circ + \mathbb{S}_k\}_{k=0:N}$ necessarily contains the system state under any possible model uncertainty realization when constraints ((11)c,d) are replaced by

$$\begin{aligned}
\alpha_{k+1} \geq\ & -s_k^j + \delta f_2(x_k^\circ + s_k^j, u_k^\circ + K_k s_k^j + c_k) - \delta f_2(x_k^\circ, u_k^\circ) \\
& - (F_{1,k} s_k^j + B_{1,k} c_k) - \delta w_{\min} \\
& + \underline{e}_2 \theta_+^{m\top}\big(\delta g_2(x_k^\circ + s_k^j, u_k^\circ + K_k s_k^j + c_k) - \delta g_2(x_k^\circ, u_k^\circ)\big) \\
& - \underline{e}_2 \theta_+^{m\top}(G_{1,k} s_k^j + B_{1,k}^g c_k) \\
& - \underline{e}_2 \theta_-^{m\top}\big(\delta g_1(x_k^\circ + s_k^j, u_k^\circ + K_k s_k^j + c_k) - \delta g_1(x_k^\circ, u_k^\circ)\big) \\
& + \underline{e}_2 \theta_-^{m\top}(G_{2,k} s_k^j + B_{2,k}^g c_k)
\end{aligned} \tag{18a}$$

$$\begin{aligned}
\beta_{k+1} \geq\ & \underline{1}^\top\big(s_k^j + \delta f_1(x_k^\circ + s_k^j, u_k^\circ + K_k s_k^j + c_k) - \delta f_1(x_k^\circ, u_k^\circ)\big) \\
& - \underline{1}^\top(F_{2,k} s_k^j + B_{2,k} c_k) + \delta w_{\max} \\
& + \theta_+^{m\top}\big(\delta g_1(x_k^\circ + s_k^j, u_k^\circ + K_k s_k^j + c_k) - \delta g_1(x_k^\circ, u_k^\circ)\big) \\
& - \theta_+^{m\top}(G_{2,k} s_k^j + B_{2,k}^g c_k) \\
& - \theta_-^{m\top}\big(\delta g_2(x_k^\circ + s_k^j, u_k^\circ + K_k s_k^j + c_k) - \delta g_2(x_k^\circ, u_k^\circ)\big) \\
& + \theta_-^{m\top}(G_{1,k} s_k^j + B_{1,k}^g c_k).
\end{aligned} \tag{18b}$$

Here $G_{i,k} = \delta(A_{i,k}^g + B_{i,k}^g K)$, $A_{i,k}^g = \frac{\partial g_i}{\partial x}(x_k^\circ, u_k^\circ)$, $B_{i,k}^g = \frac{\partial g_i}{\partial u}(x_k^\circ, u_k^\circ)$ for $i = 1,2$, and $\{\theta^m\}_{m=1:4}$ are the vertices of Θ_n, with

$$\theta_+^m = \max\{\theta^m, 0\}, \quad \theta_-^m = \min\{\theta^m, 0\}$$

denoting the non-negative and non-positive components of θ^m.

Problem (11), with ((18)a,b) replacing ((11)c,d) and with $\{x_k^\circ, u_k^\circ\}_{k=0:N}$ updated via (12)–(14), is necessarily recursively feasible since $\Theta_{n+1} \subseteq \Theta_n$ for all $n \geq 0$. In addition, the closed loop adaptive DC-TMPC strategy ensures satisfaction of the constraints $\big(x(t), u(t)\big) \in \mathbb{X}\times\mathbb{U}$ at all sampling instants $t = n\delta$, $n \geq 0$. These properties hold regardless of the number of iterations performed at each time step. Convergence of parameter estimates can be ensured by forcing the control inputs to be persistently exciting (e.g. Lu and Cannon, 2023).

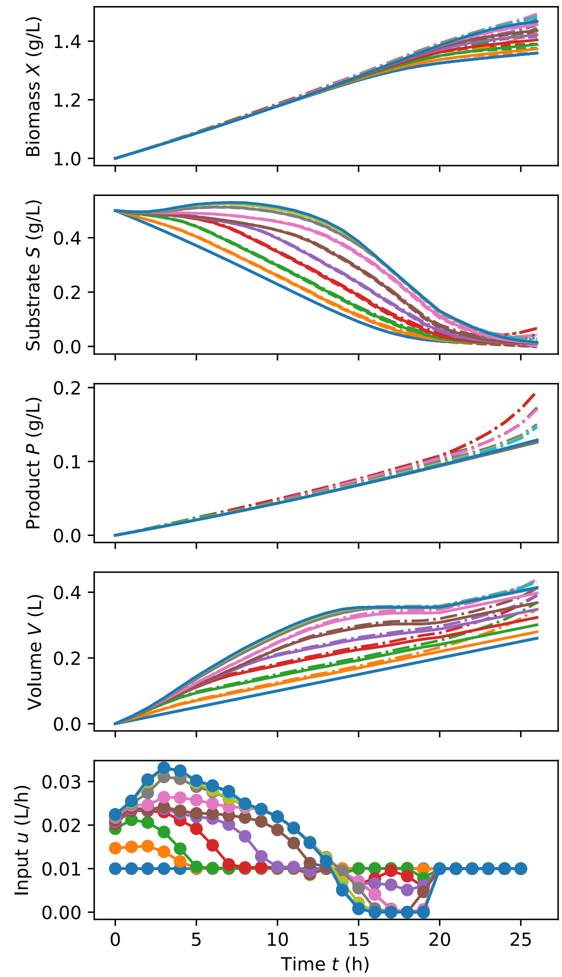

Fig. 2. Predicted state and control trajectories at initial time, $t = 0$. Solid lines: nominal predicted trajectories $\{x_k^\circ, u_k^\circ\}_{k=0:N}$, dashed/dash-dotted lines: lower/upper bounds on predicted states within $\{\mathbb{S}_k\}_{k=0:N}$.

6. Simulation results and discussion

The control algorithm was implemented in simulations of the fed-batch bioreactor described in Section 2. The control objective is to maximize product concentration over a period of $T = 100$ h subject to $x(t) \in \mathbb{X}$ and $u(t) \in \mathbb{U}$. Two pairs of ICNNs were trained as described in Sections 2 and 5 to provide DC model representations, and the DC-TMPC algorithm was formulated as described in Sections 4 and 5. The optimization (11), ((18)a,b) is a second order cone program since $\mathbb{X}$ and $\mathbb{U}$ are polytopic sets and the ICNN activation functions are piecewise linear ReLUs. We solve this problem using cvxpy (Diamond and Boyd, 2016) and mosek (MOSEK ApS, 2025).

The prediction horizon is $N = 25$ and the cost weight in ((11)a) is $\gamma = 0.1$. The disturbance bounds estimated from validation data distributed across the compact sets $\mathbb{X}$ and $\mathbb{U}$ are given by

$$\mathbb{W} = \big\{w : |w| \leq (10^{-2},\ 10^{-3},\ 10^{-3},\ 10^{-2})\big\}.$$

The gains $\{K_k\}_{k=0:N}$ were found by solving the linear quadratic regulator problem defined by the time-varying linearized model around nominal predicted trajectories with $Q = 10^{-5} I$, $R = 1$.

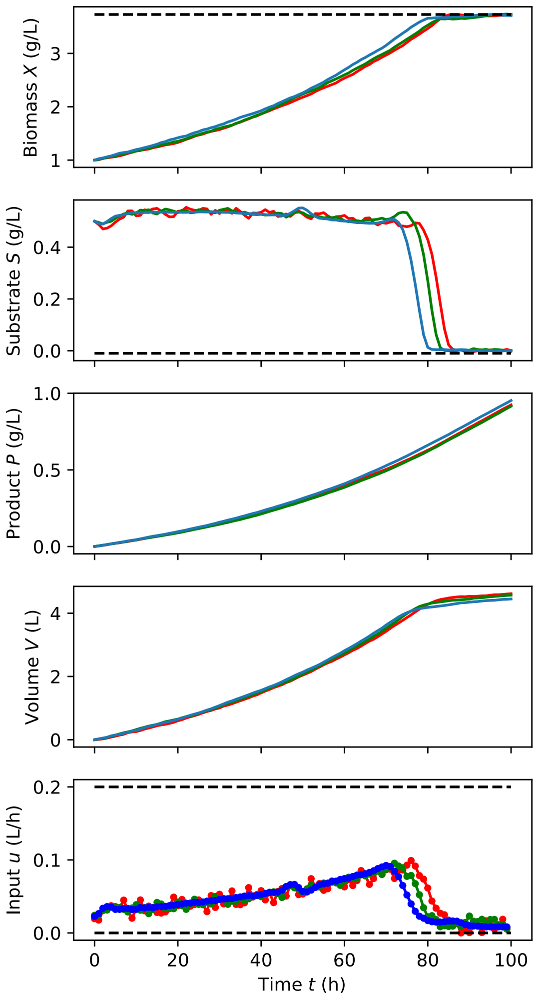

Fig. 3. Closed loop state and control trajectories. Robust adaptive MPC (blue lines), and responses when noise with bounds of ±0.005 (red) and ±0.0025 (green) is added to the control signal to speed up parameter set convergence.

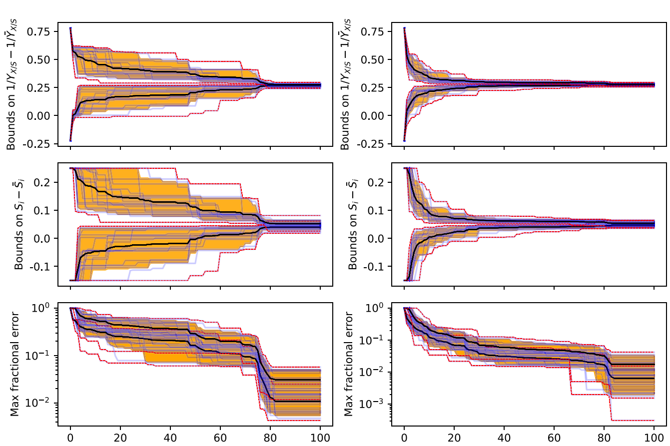

Fig. 4. Parameter set estimates for 20 closed loop simulations (left) and with noise bounded by ±0.005 (right) added to the control signal. Upper and lower parameter bounds: for individual simulations (blue lines), mean values (black lines), 90% confidence intervals (orange), and outer bounds (red dashed lines).

For the DC-TMPC algorithm of Section 4 without parameter uncertainty, a single iteration requires on average 0.6 s and at most 0.8 s (Apple M3 Pro, 36 GB memory). When parameter uncertainty is included as described in Section 5 the computation time increases to 4.5 s on average and maximum 7.5 s.

Predicted trajectories computed at the first time step ($t = 0$) are terminated when the control perturbation and changes in optimal cost fall below thresholds of 10^{-3} and 10^{-4}, which typically requires 10 iterations (Fig. 2). As expected, the nominal state trajectories at each iteration (solid lines) lie within the bounds on future states provided by the predicted tubes (dashed and dash-dotted lines). The effect of the terminal constraint, requiring $S_k \leq 0.01$ for $20 \leq k \leq 25$, can also be seen in Fig. 2.

Closed loop state and control trajectories are shown in Fig. 3. The evolution of parameter set estimates is shown in Fig. 4, which demonstrates faster convergence when random noise (independent, uniformly distributed within specified bounds) is added to the control input. The corresponding trajectories are more noisy (Fig. 3), and the improvement in estimated parameter accuracy is accompanied by small a reduction in final product concentration, which falls from $P(100) = 0.96\,\mathrm{g\,L^{-1}}$ to 0.92 (or 0.93) with noise bounds of ±0.005 (±0.0025 respectively). For comparison, if parameter estimates are not updated online and the parameter set remains equal to Θ_0 in (17) at all times, then the final product falls to $0.9\,\mathrm{g\,L^{-1}}$.

The multi-stage NMPC algorithm of Fiedler et al. (2023) applied to this case study with the scenario approach of Lucia and Engell (2013) for parameter uncertainty gives final product concentration $P(100) = 0.78\,\mathrm{g\,L^{-1}}$. The loss of performance in this case is due to a lack of explicit additive disturbance handling in this approach, which becomes infeasible for $t \geq 80\,\mathrm{h}$. In addition, it can only handle discrete combinations of parameter values due to its use of a scenario approach (whereas the approach of Section 5 provides robustness to all parameters in Θ_0), and it does not allow online parameter estimation or models that are defined directly by data, whereas both of these are features of the approach proposed in this paper.

7. Conclusions

This study demonstrates the potential of deep learning robust tube MPC for bioprocesses with uncertain model parameters and disturbances. The proposed approach employs DC representations of the system dynamics constructed using pairs of ICNNs. With this problem formulation, future performance is optimized by solving a sequence of convex problems obtained by linearizing concave model components. By convexity, robust constraint satisfaction and stability are ensured even if only one iteration is performed at each discrete time step. The paper's contributions are a DC model decomposition method using feed-forward neural networks, a robust adaptive tube MPC algorithm with online set-membership estimation of unknown model parameters, and the validation of these techniques on a benchmark fed-batch bioreactor case study.

Future work will incorporate tubes allowing more efficient scaling of computation with problem size, chance constraints handled using convex scenario programming, and testing on a parallel bioreactor system for bioprocess development of different host organisms and products.

CRediT authorship contribution statement

Niels Krausch: Software, Investigation. **Martin Doff-Sotta:** Software, Investigation. **Mark Cannon:** Supervision, Software, Funding acquisition, Conceptualization. **Peter Neubauer:** Supervision, Funding acquisition. **Mariano Nicolas Cruz Bournazou:** Supervision, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Mariano Nicolas Cruz Bournazou reports financial support was provided by German Federal Ministry of Education and Research (BMBF). Mark Cannon reports financial support was provided by Engineering and Physical Sciences Research Council. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We gratefully acknowledge the financial support of the German Federal Ministry of Education and Research (BMBF) (ref. 01DD20002A-KIWI Biolab) and the EPSRC (UKRI) Doctoral Prize scheme, UK (ref. EP/W524311/1).

Data availability

Data will be made available on request.

References

Amos, B., Xu, L., Kolter, J.Z., 2017. Input convex neural networks. In: Proceedings of the 34th International Conference on Machine Learning, vol. 70, PMLR, pp. 146–155.

Andersson, J.A.E., Gillis, J., Horn, G., Rawlings, J.B., Diehl, M., 2019. CasADi – A software framework for nonlinear optimization and optimal control. Math. Program. Comput. 11 (1), 1–36.

Bangi, M.S.F., Kao, K., Kwon, J.S.I., 2022. Physics-informed neural networks for hybrid modeling of lab-scale batch fermentation for β-carotene production using saccharomyces cerevisiae. Chem. Eng. Res. Des. 179, 415–423.

Buerger, J., Cannon, M., Doff-Sotta, M., 2024. Safe learning in nonlinear model predictive control. In: Proceedings of the 6th Annual Learning for Dynamics and Control (L4DC), vol. 242, PMLR, pp. 603–614.

Chollet, F., et al., 2015. Keras. https://keras.io.

Cruz Bournazou, M., Barz, T., Nickel, D., Lopez Cárdenas, D., Glauche, F., Knepper, A., Neubauer, P., 2017. Online optimal experimental re-design in robotic parallel fed-batch cultivation facilities. Biotechnol. Bioeng. 114 (3), 610–619.

Diamond, S., Boyd, S., 2016. CVXPY: A python-embedded modeling language for convex optimization. J. Mach. Learn. Res. 17 (83), 1–5.

Doff-Sotta, M., Cannon, M., 2022. Difference of convex functions in robust tube nonlinear MPC. In: 61st IEEE Conference on Decision and Control, Cancun. pp. 3044–3050.

Fiedler, F., Karg, B., Lüken, L., Brandner, D., Heinlein, M., Brabender, F., Lucia, S., 2023. do-mpc: Towards FAIR nonlinear and robust model predictive control. Control Eng. Pract. 140, 105676.

Hindmarsh, A., Brown, P., Grant, K., Lee, S., Serban, R., Shumaker, D., Woodward, C., 2005. SUNDIALS, suite of nonlinear and differential/algebraic equation solvers. ACM Trans. Math. Software 31 (3), 363–396.

Horst, R., Thoai, N.V., 1999. DC programming: overview. J. Optim. Theory Appl. 103, 1–43.

Jabarivelisdeh, B., Carius, L., Findeisen, R., Waldherr, S., 2020. Adaptive predictive control of bioprocesses with constraint-based modeling and estimation. Comput. Chem. Eng. 135, 106744.

Kager, J., Tuveri, A., Ulonska, S., Kroll, P., Herwig, C., 2020. Experimental verification and comparison of model predictive, PID and model inversion control in a penicillium chrysogenum fed-batch process. Process Biochem. 90, 1–11.

Krausch, N., Kim, J., Barz, T., Lucia, S., Groß, S., Huber, M., Schiller, S., Neubauer, P., Cruz Bournazou, M., 2022. High-throughput screening of optimal process conditions using model predictive control. Biotechnol. Bioeng. 119 (12), 3584–3595.

Lipp, T., Boyd, S., 2016. Variations and extension of the convex–concave procedure. Optim. Eng. 17, 263–287.

Lishkova, Y., Cannon, M., 2025. A successive convexification approach for robust receding horizon control. IEEE Trans. Autom. Control. (Early Access), URL https://doi.org/10.1109/TAC.2025.3558253.

Lorenzen, M., Cannon, M., Allgöwer, F., 2019. Robust MPC with recursive model update. Automatica 103, 467–471.

Lu, X., Cannon, M., 2023. Robust adaptive model predictive control with persistent excitation conditions. Automatica 152, 110959.

Lu, X., Cannon, M., Koksal-Rivet, D., 2021. Robust adaptive model predictive control: Performance and parameter estimation. Internat. J. Robust Nonlinear Control 31, 8703–8724.

Lucia, S., Engell, S., 2013. Robust nonlinear model predictive control of a batch bioreactor using multi-stage stochastic programming. In: European Control Conference. ECC, pp. 4124–4129.

Mahanty, B., 2023. Hybrid modelling in bioprocess dynamics: Structural variabilities,implementation strategies, and practical challenges. Biotechnol. Bioeng. 120, 2072–2091.

MOSEK ApS, 2025. The MOSEK optimizer API for python. v.11.0.. URL https://docs.mosek.com/latest/pythonfusion/index.html.

Mowbray, M., Petsagkourakis, P., Del Rio Chanona, E., Zhang, D., 2022. Safe chance constrained reinforcement learning for batch process control. Comput. Chem. Eng. 157, 107630.

Rathore, A.S., Mishra, S., Nikita, S., Priyanka, P., 2021. Bioprocess control: Current progress and future perspectives. Life (Basel) 11 (6), 557.

Sankaranarayanan, P., Rengaswamy, R., 2022. CDiNN—Convex difference neural networks. Neurocomputing 495, 153–168.

Shah, P., Sheriff, M.Z., Bangi, M.S.F., Kravaris, C., Kwon, J.S.I., Botre, C., Hirota, J., 2022. Deep neural network-based hybrid modeling and experimental validation for an industry-scale fermentation process: Identification of time-varying dependencies among parameters. Chem. Eng. J. 441, 135643.

Srinivasan, B., Bonvin, D., Visser, E., Palanki, S., 2003. Dynamic optimization of batch processes. Comput. Chem. Eng. 27 (1), 27–44.

Tuveri, A., Nakama, C.S., Matias, J., Holck, H.E., Jäschke, J., Imsland, L., Bar, N., 2023. A regularized moving horizon estimator for combined state and parameter estimation in a bioprocess experimental application. Comput. Chem. Eng. 172, 108183. http://dx.doi.org/10.1016/j.compchemeng.2023.108183.