Semi-Infinite Programming yields Optimal Disturbance Model for Offset-Free Nonlinear Model Predictive Control

Adrian Caspari[a], Hatim Djelassi[a], Adel Mhamdi[a], Lorenz T. Biegler[d], Alexander Mitsos*[,b,a,c]

[a]Process Systems Engineering (AVT.SVT), RWTH Aachen University, 52074 Aachen, Germany

[b]JARA-ENERGY, 52056 Aachen, Germany

[c]Energy Systems Engineering (IEK-10), Forschungszentrum Jülich, 52425 Jülich, Germany

[d]Carnegie Mellon University, Department of Chemical Engineering, Pittsburgh, PA 15213, USA

This is the authors' accepted manuscript of the following article:

A. Caspari, H. Djelassi, A. Mhamdi, L. T. Biegler, and A. Mitsos, "Semi-infinite programming yields optimal disturbance model for offset-free nonlinear model predictive control," *Journal of Process Control*, vol. 101, pp. 35–51, 2021. https://doi.org/10.1016/j.jprocont.2021.03.005. Manuscript received 4 July 2020; received in revised form 5 January 2021; accepted 26 March 2021; available online 7 April 2021.

Abstract:

Offset-free nonlinear model predictive control (NMPC) can eliminate the tracking offset associated with the presence of plant-model mismatch or other persistent disturbances by augmenting the plant model with disturbances and employing an observer to estimate both the states and disturbances. Despite their importance, a systematic approach for the generation of suitable disturbance models is not available.

We propose an optimization-based method to generate disturbance models based on sufficient observability conditions and generalize the theory of offset-free NMPC by allowing for (i) more measured variables than controlled variables and (ii) unmeasured controlled variables. Based on the sufficient conditions, we formulate a generalized semi-infinite program, which we reformulate and solve as a simpler semi-infinite program using a discretization algorithm. The solution furnishes the optimal disturbance model, which maximizes the set of those state, manipulated variable, and disturbance realizations, for which a sufficient observability condition is satisfied. The disturbance model is generated offline and can be used online for offset-free NMPC.

We apply the approach using three case studies ranging from small scale chemical reactor cases to a medium scale polymerization reactor case. The results demonstrate the validity and usefulness of the generalized theory and show that the model generation approach successfully finds suitable disturbance models for offset-free NMPC.

Keywords: offset-free nonlinear model predictive control, optimal disturbance modeling, unknown disturbances, plant model mismatch, semi-infinite programming, deterministic global optimization

*A. Mitsos, AVT Process Systems Engineering, RWTH Aachen University, 52074 Aachen, Germany, E-mail: amitsos@alum.mit.edu

1 Introduction

Model predictive control (MPC) is an established advanced process control concept, e.g., for industrial processes [1, 2]. One of the most popular MPC tasks is setpoint tracking. However, the inevitable discrepancy between the controller model and the real process, as well as persistent disturbances typically lead to an offset between the tracked process variables and their desired setpoints [3, 4]. In the following, we use for simplicity "disturbances" for both phenomena. In the case of economic MPC, this translates into suboptimal or even infeasible closed-loop process behavior [5]. For example, applying reduced dynamic controller models inherently introduces a disturbance, which may lead to undesired offset, e.g. [6, 7]. This motivates offset-free MPC [4, 8], i.e., exactly tracking given setpoints by adding constant differential states to the process model as a disturbance model (DM). The values of the disturbance states are determined using suitable state estimation. Offset-free closed-loop process behavior can be guaranteed under reasonable assumptions. We refer to the work of Pannocchia et al. [5] for an overview of offset-free MPC. While several works focused on offset-free linear MPC (LMPC), e.g., [3, 8, 9, 10, 11, 12], few articles are available on offset-free NMPC. Huang et al. [13] used an advanced-step NMPC and advanced-step moving horizon estimator for offset-free NMPC. Morari and Maeder [4] presented sufficient conditions on the DM, the controller, the observer, and the closed-loop behavior of the controlled system to guarantee offset-free tracking. Pannocchia [14] combined the ideas of offset-free NMPC for tracking and real-time optimization with modifier adaptation to achieve offset-free economic NMPC (eNMPC). Schulze et al. [7] used an output DM to eliminate an offset in the controlled and measured variables introduced by the application of a reduced dynamic controller model.

Although the choice of the DM is crucial and the appropriate assumptions for it exist, cf. [5], there are only a few works for the generation of DMs. In particular, the focus has been on linear MPC for constant setpoint tracking [15, 16]. In contrast, there is no general approach available for the generation of DMs in NMPC, as also mentioned in [5]. Furthermore, the existing works about offset-free NMPC [4, 5, 14] assumed the number of controlled variables (CVs) to equal the number of manipulated variables (MVs) and the CVs to be measured, which is not always given in industrial examples. For instance, in distillation columns, the temperature is typically used as a measurement whereas the unmeasured product purity is to be controlled [17, 18, 19, 6].

Herein, we present a general approach for the generation of DMs in offset-free NMPC. The DM results from the solution of an optimization problem, which embeds a sufficient observability condition for the DM to guarantee offset-free tracking behavior. More specifically, we propose a generalized semi-infinite program (GSIP), cf. [20], taking into account the structure of the augmented model. The GSIP is reformulated as a semi-infinite program (SIP), cf. [20]. Solving the SIP, we maximize the size of a set within the state-space of the augmented model, so that a sufficient observability condition for offset-free NMPC is satisfied within the set. That is, the augmented model is observable for all states that are elements of the set and the resulting disturbance model is optimal in the sense that it maximizes this set. The process dynamics are not included in the optimization problem explicitly but rather accounted for by the sufficient observability condition, which in turn does not require the integration of the augmented model [4]. Thus, the model generation approach does not require to treat a dynamic model and considers algebraic equations only. Furthermore, we relax the existing sufficient conditions on offset-free NMPC by allowing for more measured variables than CVs and unmeasured CVs while the existing literature about offset-free NMPC assumes the CVs to be measured [4]. Both more measured variables than CVs and unmeasured CVs are relevant in the application of NMPC to industrial processes. We illustrate the theoretical extensions and apply the new DM generation approach based on SIP to three numerical case studies ranging from small scale chemical reactor cases to a larger polymerization reactor case.

The remainder of the work is structured as follows. We present the theory of offset-free NMPC and extend the sufficient conditions from Morari and Maeder [4] in Section 2. Section 3 outlines our DM generation approach. We illustrate the results and DM generation approach in numerical case studies in Section 4 and draw conclusions in Section 5.

2 NMPC

We present the controlled system and the nominal NMPC first, before we motivate and introduce offset-free NMPC. For an overview of NMPC and offset-free NMPC, we refer to [2, 5].

Both nominal and offset-free NMPC aim at controlling the exact, but usually unknown real system

$$\begin{aligned} \boldsymbol{x}_\Phi(k+1) &= \boldsymbol{f}_\Phi(\boldsymbol{x}_\Phi(k), \boldsymbol{u}_\Phi(k), \boldsymbol{d}_\Phi(k)) \\ \boldsymbol{z}_\Phi(k) &= \boldsymbol{g}^z_\Phi(\boldsymbol{x}_\Phi(k), \boldsymbol{d}_\Phi(k)) \\ \boldsymbol{y}_\Phi(k) &= \boldsymbol{g}^y_\Phi(\boldsymbol{x}_\Phi(k), \boldsymbol{d}_\Phi(k)), \end{aligned} \tag{1}$$

with the differential states $\boldsymbol{x}_\Phi(k) \in \mathcal{X}_\Phi = \mathbb{R}^{n_{x_\Phi}}$, disturbances $\boldsymbol{d}_\Phi(k) \in \mathcal{D}_\Phi = \mathbb{R}^{n_{d_\Phi}}$, measurement $\boldsymbol{z}_\Phi(k) \in \mathcal{Z} = \mathbb{R}^{n_z}$, the CVs $\boldsymbol{y}_\Phi(k) \in \mathcal{Y} = \mathbb{R}^{n_y}$, the MVs $\boldsymbol{u}_\Phi(k) \in \mathcal{U} = \mathbb{R}^{n_u}$, and $\boldsymbol{f}_\Phi : \mathcal{X}_\Phi \times \mathcal{U} \times \mathcal{D}_\Phi \to \mathcal{X}_\Phi$, the measurement function $\boldsymbol{g}^z_\Phi : \mathcal{X}_\Phi \times \mathcal{D}_\Phi \to \mathcal{Z}$, and the output function $\boldsymbol{g}^y_\Phi : \mathcal{X}_\Phi \times \mathcal{D}_\Phi \to \mathcal{Y}$, and the discrete time point $k \in \mathbb{N}$.

The tracking error between the outputs $\boldsymbol{y}_\Phi$ and a reference signal $\boldsymbol{r}$ is defined by

$$\boldsymbol{e}(k) = \boldsymbol{y}_\Phi(k) - \boldsymbol{r}(k).$$

We assume that the disturbances and the reference signals are asymptotically constant, i.e.,

$$\begin{aligned} \boldsymbol{r}(k) &\to \boldsymbol{r}_\infty \\ \boldsymbol{d}_\Phi(k) &\to \boldsymbol{d}_{\Phi,\infty}. \end{aligned}$$

In offset-free NMPC, the aim is to control the system such that $\boldsymbol{y}_\Phi$ tracks the given reference signal $\boldsymbol{r}$, i.e., to achieve zero tracking error asymptotically:

$$\boldsymbol{e}(k) \to \mathbf{0}, k \to \infty.$$

2.1 Nominal NMPC

Nominal NMPC uses a nominal model of the plant given by

$$\begin{aligned} \boldsymbol{x}(k+1) &= \boldsymbol{f}(\boldsymbol{x}(k), \boldsymbol{u}(k)) \\ \boldsymbol{z}(k) &= \boldsymbol{g}^z(\boldsymbol{x}(k)) \\ \boldsymbol{y}(k) &= \boldsymbol{g}^y(\boldsymbol{x}(k)), \end{aligned} \tag{2}$$

with the differential states $\boldsymbol{x}(k) \in \mathcal{X} = \mathbb{R}^{n_x}$, measurements $\boldsymbol{z}(k) \in \mathcal{Z} = \mathbb{R}^{n_z}$, the CVs $\boldsymbol{y}(k) \in \mathcal{Y} = \mathbb{R}^{n_y}$, and $\boldsymbol{f} : \mathcal{X} \times \mathcal{U} \to \mathcal{X}$, the measurement function $\boldsymbol{g}^z : \mathcal{X} \to \mathcal{Z}$, and the output function $\boldsymbol{g}^y : \mathcal{X} \to \mathcal{Y}$.

Nominal NMPC uses the nominal plant model (2) as the controller model in state estimation and control. To estimate the states, nominal NMPC employs an observer of the form

$$\begin{aligned} \hat{\boldsymbol{x}}(k+1) &= \boldsymbol{f}(\hat{\boldsymbol{x}}(k), \boldsymbol{u}(k)) + \boldsymbol{l}_x(\boldsymbol{\epsilon}(k)) \\ \hat{\boldsymbol{z}}(k) &= \boldsymbol{g}^z(\hat{\boldsymbol{x}}(k)) \\ \boldsymbol{\epsilon}(k) &= \boldsymbol{z}_\Phi(k) - \hat{\boldsymbol{z}}(k), \end{aligned} \tag{3}$$

with the observer residual $\boldsymbol{\epsilon}$ and the observer gains $\boldsymbol{l}_x : \mathcal{Z} \to \mathcal{X}$, with $\boldsymbol{l}_x(\mathbf{0}) = \mathbf{0}$.

At each time k for a given reference $\boldsymbol{r}(k)$ and $\hat{\boldsymbol{x}}(k)$, the nominal NMPC solves the following optimization problem:

$$\begin{aligned} \min_{\substack{\bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}, \\ \boldsymbol{x}_1, \dots, \boldsymbol{x}_N \\ \boldsymbol{u}_0, \dots, \boldsymbol{u}_{N-1}}} \quad & F(\boldsymbol{x}_N - \bar{\boldsymbol{x}}) + \sum_{t=0}^{N-1} l(\boldsymbol{x}_t - \bar{\boldsymbol{x}}, \boldsymbol{u}_t - \bar{\boldsymbol{u}}) \\ s.t. \quad & \boldsymbol{x}_0 = \hat{\boldsymbol{x}}(k) \\ & \bar{\boldsymbol{x}} = \boldsymbol{f}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}) \\ & \boldsymbol{r}(k) = \boldsymbol{g}^y(\bar{\boldsymbol{x}}) \\ & \boldsymbol{x}_{t+1} = \boldsymbol{f}(\boldsymbol{x}_t, \boldsymbol{u}_t), \forall t \in \{0, ..., N-1\}, \end{aligned} \tag{4}$$

with $\boldsymbol{x}_t \in \mathbb{X}, \forall t \in \{1, ..., N\}$, $\boldsymbol{u}_t \in \mathbb{U}, \forall t \in \{0, ..., N-1\}$, $\bar{\boldsymbol{x}} \in \mathbb{X}, \bar{\boldsymbol{u}} \in \mathbb{U}$ are the states and inputs defining the target equilibrium, and $\mathbb{X} \subseteq \mathcal{X}$ the admissible state set and $\mathbb{U} \subseteq \mathcal{U}$ the admissible MV set.
We denote the solution of (4) by $\boldsymbol{U}^*(k) = \left(\bar{\boldsymbol{x}}^*(k), \bar{\boldsymbol{u}}^*(k), \boldsymbol{x}_1^*(k), ..., \boldsymbol{x}_N^*(k), \boldsymbol{u}_0^*(k), ..., \boldsymbol{u}_{N-1}^*(k)\right)$. $\boldsymbol{u}_0^*(k)$ is applied as (an implicit) nominal NMPC control law to the process

$$\boldsymbol{u}_\Phi(k) = \boldsymbol{c}(\hat{\boldsymbol{x}}(k), \boldsymbol{r}(k)) = \boldsymbol{u}_0^*(k). \tag{5}$$

Nominal NMPC leads to a tracking offset in the presence of disturbance, which we illustrate in the case studies in Section 4.

2.2 Offset-Free NMPC

We now present the theory and sufficient conditions for offset-free NMPC. Morari and Maeder [4] assumed that the measured variables are the CVs. We extend their work by proving that offset-free NMPC can be guaranteed in the case that the CVs are a subset of the measured variables or in the case that unmeasured CVs are predicted accurately at steady state. The theory is shown for ordinary differential equation systems (ODEs), although it can in principle be modified for differential-algebraic equation systems (DAEs) by applying the implicit function theorem.

To account for disturbances to eliminate the offset associated with nominal NMPC, offset-free NMPC uses a nominal model augmented with a DM, resulting in the following augmented model:

$$\begin{aligned} \boldsymbol{x}_{\text{aug}}(k+1) &= \boldsymbol{f}_{\text{aug}}(\boldsymbol{x}_{\text{aug}}(k), \boldsymbol{u}_{\text{aug}}(k), \boldsymbol{d}_{\text{aug}}(k)) \\ \boldsymbol{d}_{\text{aug}}(k+1) &= \boldsymbol{d}_{\text{aug}}(k) \\ \boldsymbol{z}_{\text{aug}}(k) &= \boldsymbol{g}^{z}_{\text{aug}}(\boldsymbol{x}_{\text{aug}}(k), \boldsymbol{d}_{\text{aug}}(k)) \\ \boldsymbol{y}_{\text{aug}}(k) &= \boldsymbol{g}^{y}_{\text{aug}}(\boldsymbol{x}_{\text{aug}}(k), \boldsymbol{d}_{\text{aug}}(k)), \end{aligned} \tag{6}$$

with the differential states $\boldsymbol{x}_{\text{aug}}(k) \in \mathcal{X} = \mathbb{R}^{n_x}$, the disturbances $\boldsymbol{d}_{\text{aug}}(k) \in \mathcal{D} = \mathbb{R}^{n_d}$, measurement $\boldsymbol{z}_{\text{aug}}(k) \in \mathcal{Z}$, the CVs $\boldsymbol{y}_{\text{aug}}(k) \in \mathcal{Y}$, the MVs $\boldsymbol{u}_{\text{aug}}(k) \in \mathcal{U}$, and $\boldsymbol{f}_{\text{aug}} : \mathcal{X} \times \mathcal{U} \times \mathcal{D} \to \mathcal{X}$, the measurement function $\boldsymbol{g}^{z}_{\text{aug}} : \mathcal{X} \times \mathcal{D} \to \mathcal{Z}$, and the output function $\boldsymbol{g}^{y}_{\text{aug}} : \mathcal{X} \times \mathcal{D} \to \mathcal{Y}$.

One possible choice would be a linear disturbance model added to the nominal model:

$$\boldsymbol{f}_{\text{aug}}(\boldsymbol{x}_{\text{aug}}(k), \boldsymbol{u}_{\text{aug}}(k), \boldsymbol{d}_{\text{aug}}(k)) = \boldsymbol{f}(\boldsymbol{x}_{\text{aug}}(k), \boldsymbol{u}_{\text{aug}}(k)) + \boldsymbol{B}_d \cdot \boldsymbol{d}_{\text{aug}}(k),$$

with $\boldsymbol{B}_d \in \mathbb{R}^{N_x \times N_d}$ that would have to be defined during DM generation. We denote a disturbance model as an exact disturbance model if the augmented system (6) and the real system (1) coincide. Later, we show that an exact disturbance model is not required to achieve offset-free tracking even in the case of unmeasured CVs.

To estimate the states and disturbances, offset-free NMPC employs an observer of the form

$$\begin{aligned} \hat{\boldsymbol{x}}_{\text{aug}}(k+1) &= \boldsymbol{f}_{\text{aug}}(\hat{\boldsymbol{x}}_{\text{aug}}(k), \boldsymbol{u}_{\text{aug}}(k), \hat{\boldsymbol{d}}_{\text{aug}}(k)) + \boldsymbol{l}_x\left(\boldsymbol{\epsilon}_{\text{aug}}(k)\right) \\ \hat{\boldsymbol{d}}_{\text{aug}}(k+1) &= \hat{\boldsymbol{d}}_{\text{aug}}(k) + \boldsymbol{l}_d\left(\boldsymbol{\epsilon}_{\text{aug}}(k)\right) \\ \hat{\boldsymbol{z}}_{\text{aug}}(k) &= \boldsymbol{g}^{z}_{\text{aug}}(\hat{\boldsymbol{x}}_{\text{aug}}(k), \hat{\boldsymbol{d}}_{\text{aug}}(k)) \\ \boldsymbol{\epsilon}_{\text{aug}}(k) &= \boldsymbol{z}_{\Phi}(k) - \hat{\boldsymbol{z}}_{\text{aug}}(k), \end{aligned} \tag{7}$$

with the observer residual $\boldsymbol{\epsilon}_{\text{aug}}$ and the observer gain $\boldsymbol{l}_d : \mathcal{Z} \to \mathcal{D}$, with $\boldsymbol{l}_d(\boldsymbol{0}) = \boldsymbol{0}$.

At each time k for a given reference $\boldsymbol{r}(k)$, $\hat{\boldsymbol{d}}_{\text{aug}}(k)$, $\hat{\boldsymbol{x}}_{\text{aug}}(k)$, the offset-free NMPC solves the following optimization problem

$$\min_{\substack{\bar{\boldsymbol{x}}_{\text{aug}},\bar{\boldsymbol{u}}_{\text{aug}},\\ \boldsymbol{x}_{\text{aug},1},\ldots,\boldsymbol{x}_{\text{aug},N}\\ \boldsymbol{u}_{\text{aug},0},\ldots,\boldsymbol{u}_{\text{aug},N-1}}} F(\boldsymbol{x}_{\text{aug},N}-\bar{\boldsymbol{x}}_{\text{aug}})+\sum_{t=0}^{N-1} l(\boldsymbol{x}_{\text{aug},t}-\bar{\boldsymbol{x}}_{\text{aug}},\boldsymbol{u}_{\text{aug},t}-\bar{\boldsymbol{u}}_{\text{aug}}) \tag{8a}$$

$$s.t.\quad \boldsymbol{x}_0=\hat{\boldsymbol{x}}_{\text{aug}}(k), \boldsymbol{d}_{\text{aug},0}=\hat{\boldsymbol{d}}_{\text{aug}}(k) \tag{8b}$$

$$\bar{\boldsymbol{x}}_{\text{aug}}=\boldsymbol{f}_{\text{aug}}(\bar{\boldsymbol{x}}_{\text{aug}},\boldsymbol{d}_{\text{aug},0},\bar{\boldsymbol{u}}_{\text{aug},N}) \tag{8c}$$

$$\boldsymbol{r}(k)=\boldsymbol{g}^{y}_{\text{aug}}(\bar{\boldsymbol{x}}_{\text{aug}},\boldsymbol{d}_{\text{aug},0}) \tag{8d}$$

$$\boldsymbol{x}_{\text{aug},t+1}=\boldsymbol{f}_{\text{aug}}(\boldsymbol{x}_{\text{aug},t},\boldsymbol{d}_{\text{aug},0},\boldsymbol{u}_{\text{aug},t}), \forall t\in\{0,\ldots,N-1\}, \tag{8e}$$

with $\boldsymbol{x}_{\text{aug},t}\in\mathbb{X}_{\text{aug}}, \forall t\in\{1,\ldots,N\}$, $\boldsymbol{u}_{\text{aug},t}\in\mathbb{U}_{\text{aug}}, \forall t\in\{0,\ldots,N-1\}$, $\bar{\boldsymbol{x}}_{\text{aug}}\in\mathbb{X}_{\text{aug}}, \bar{\boldsymbol{u}}_{\text{aug}}\in\mathbb{U}_{\text{aug}}$ are the states and inputs defining the target equilibrium, and $\mathbb{X}_{\text{aug}}\subseteq\mathcal{X}$ the admissible state set and $\mathbb{U}_{\text{aug}}\subseteq\mathcal{U}$ the admissible MV set.

We denote the solution of (8) by $\boldsymbol{U}^*{}_{\text{aug}}(k)=(\bar{\boldsymbol{x}}^*{}_{\text{aug}}(k),\bar{\boldsymbol{u}}_{\text{aug}}{}^*(k),\boldsymbol{x}^*{}_{\text{aug},1}(k),\ldots,\boldsymbol{x}^*{}_{\text{aug},N}(k),$ $\boldsymbol{u}^*{}_{\text{aug},0}(k),\ldots,\boldsymbol{u}^*{}_{\text{aug},N-1}(k))$. $\boldsymbol{u}_{\text{aug},0}{}^*(k)$ is applied as (an implicit) offset-free NMPC control law to the process

$$\boldsymbol{u}_{\Phi}(k)=\boldsymbol{c}_{\text{aug}}(\hat{\boldsymbol{x}}_{\text{aug}}(k),\hat{\boldsymbol{d}}(k),\boldsymbol{r}(k))=\boldsymbol{u}^*{}_{\text{aug},0}(k). \tag{9}$$

2.2.1 Assumptions and Sufficient Conditions for Offset-Free NMPC

We state several assumptions which are used later to prove offset-free tracking. The first two assumptions generalize the assumption of Morari and Maeder [4], where it is assumed that the measured variables are the CVs, implying $n_z=n_y$.

Assumption 1 (measurements) *There are n_z measurements and n_y CVs, with $n_z\geqslant n_y$. The measurements contain the CVs:*

$$z_{\text{aug},i}(k)=y_{\text{aug},i}(k), \forall i\in\{1,\ldots,n_y\}$$

$$z_{\Phi,i}(k)=y_{\Phi,i}(k), \forall i\in\{1,\ldots,n_y\}.$$

If Assumption 1 is not satisfied, we assume that the augmented system is exact for those CVs that are not measured:

Assumption 2 (exact steady state prediction) *The augmented system accurately predicts those CVs at the steady state which are not measured.*

$$y_{\text{aug},\infty,\text{i}}=y_{\Phi,\infty,\text{i}}, \forall i: y_{\infty,\text{i}}\neq z_{\infty,\text{i}}.$$

Assumption 2 is satisfied, e.g., when the disturbances are modeled exactly, so that the augmented system is exact. This may occur in practical applications when it is known that a persistent disturbance only affects single equations, inputs, or outputs. The case studies demonstrate that exact predictions can also be achieved by adding disturbances to those equations corresponding to the unmeasured CVs. Thus, the assumption is not as restrictive as it might sound and can easily be enforced, which we take into account in the DM generation procedure presented in Section 3. Below, we prove that offset-free tracking can be achieved when either Assumption 1 or 2 is satisfied.

Assumption 3 (observer) *The observer is designed to be nominally error-free at steady state, satisfying*

$$\boldsymbol{l}_d(\boldsymbol{\epsilon}) = \mathbf{0} \Rightarrow \boldsymbol{\epsilon} = \mathbf{0}$$

for all $\boldsymbol{\epsilon} \in \mathcal{Z}$.

Assumption 3 generally implies $n_d \geqslant n_z$ unless the measurements $\boldsymbol{z}$ are degenerate, e.g., in case they are correlated or trivially defined.

We use the following assumptions on the controller, the closed-loop system, and on the observability and controllability of the augmented model (6).

Assumption 4 (controller) *Let the control law be defined by* (9). *The controller is designed to be nominally error-free at steady state, i.e., for all* $\boldsymbol{d} \in \mathcal{D}$, $\boldsymbol{r} \in \mathcal{Y}$ *which yield strictly feasible targets,*

$$\boldsymbol{x} - \boldsymbol{f}_{\text{aug}}(\boldsymbol{x}, \boldsymbol{d}, \boldsymbol{c}_{\text{aug}}(\boldsymbol{x}, \boldsymbol{d}, \boldsymbol{r})) = \mathbf{0} \Rightarrow \boldsymbol{g}^y_{\text{aug}}(\boldsymbol{x}, \boldsymbol{d}) = \boldsymbol{r}$$

holds for all $\boldsymbol{x} \in \mathcal{X}$.

Assumption 4 implies in general $n_u \geqslant n_y$ barring degeneracy of the CVs $\boldsymbol{y}$. Thus, we exclude those cases where, e.g., the CVs are not independent of each other.

Assumption 5 (convergence of closed-loop system) *If the closed-loop system* (1), (7), (8), *and* (9) *is subject to an asymptotically constant reference and disturbance with* $\boldsymbol{r}(k) \to \boldsymbol{r}_\infty$, $\boldsymbol{d}_\Phi(k) \to \boldsymbol{d}_{\Phi,\infty}$ *as* $k \to \infty$, *then all states converge to a steady state* $\boldsymbol{y}_\Phi(k) \to \boldsymbol{y}_{\Phi,\infty}$, $\boldsymbol{u}_\Phi(k) \to \boldsymbol{u}_\infty$ *as* $k \to \infty$, *with* $\boldsymbol{y}_{\Phi,\infty} \in \mathcal{Y}$, $\boldsymbol{u}_\infty \in \mathcal{U}$, *and* $\boldsymbol{r}_\infty \in \mathcal{Y}$ *strictly in the interior of the feasible set.*

Assumption 6 (observability) *Consider the augmented model* (6). *For all* $\boldsymbol{z} \in \mathcal{Z}$, $\boldsymbol{u} \in \mathcal{U}$, *there exist* $\boldsymbol{x}^* \in \mathcal{X}$, $\boldsymbol{d}^* \in \mathcal{D}$ *such that*

$$\begin{aligned} \boldsymbol{x}^* &= \boldsymbol{f}_{\text{aug}}(\boldsymbol{x}^*, \boldsymbol{d}^*, \boldsymbol{u}) \\ \boldsymbol{z} &= \boldsymbol{g}^z_{\text{aug}}(\boldsymbol{x}^*, \boldsymbol{d}^*) \end{aligned} \tag{10}$$

holds. Furthermore, $(\boldsymbol{x}^*, \boldsymbol{d}^*)$ *is the unique solution to* (10) *for any given* $(\boldsymbol{u}, \boldsymbol{z})$.

As in [4], we assume non-degeneracy of the measurements $\boldsymbol{z}$. Then, (10) will have no solution unless $n_d \geqslant n_z$ and a unique solution requires $n_d = n_z$. Using Assumptions 3 and 6 justifies the focus on the case $n_d = n_z$ for the DM generation approach presented in Section 3. There, we present a DM generation approach based on an optimization problem that furnishes the optimal disturbance model. Although it would be possible to use n_d as a degree of freedom in the optimization problem, the theory requires $n_d = n_z$, as implied by Assumption 6. Therefore, we focus on $n_d = n_z$.

Assumption 7 (controllability) *Consider the augmented model (6). For all $\boldsymbol{r} \in \mathcal{Y}$, $\boldsymbol{d} \in \mathcal{D}$, there exist $\boldsymbol{x}^* \in \mathcal{X}$, $\boldsymbol{u}^* \in \mathcal{U}$ such that*

$$\begin{aligned} \boldsymbol{x}^* &= \boldsymbol{f}_{\text{aug}}(\boldsymbol{x}^*, \boldsymbol{d}, \boldsymbol{u}^*) \\ \boldsymbol{r} = \boldsymbol{y} &= \boldsymbol{g}^y_{\text{aug}}(\boldsymbol{x}^*, \boldsymbol{d}) \end{aligned} \tag{11}$$

holds. Furthermore, $(\boldsymbol{x}^*, \boldsymbol{u}^*)$ is the unique solution to (11) for any given $(\boldsymbol{d}, \boldsymbol{r})$.

With the assumptions and definitions, we can state the following theorem which is adapted from a theorem given in [4] by allowing more measurements than CVs and unmeasured CVs, i.e., the assumptions are relaxed compared to the assumptions in [4].

Theorem 1 (offset-free NMPC) *Consider the augmented model (6), the observer (7) and the controller (9). Let either Assumption 1 or 2 and Assumptions 3-7 be satisfied. Then $\boldsymbol{y}_\Phi(k) \to \boldsymbol{r}(k)$ as $k \to \infty$.*

The proof is in Appendix A.

We do not assume any specific form of the output or measurement function except non-degeneracy, i.e., both can be any function of states and disturbances.

We use the following assumption and sufficient condition from [4] for Assumption 6 (observability) to generate a DM in the next section.

Assumption 8 *The functions $\boldsymbol{f}_{\text{aug}}$ and $\boldsymbol{g}^z_{\text{aug}}$ are continuously differentiable, i.e., belong to C^1.*

We abbreviate the gradients of the augmented system by

$$\begin{aligned} \boldsymbol{G}_x(\boldsymbol{x}, \boldsymbol{d}) &:= \left.\frac{\partial \boldsymbol{g}^z_{\text{aug}}}{\partial \boldsymbol{x}_{\text{aug}}}\right|_{(\boldsymbol{x},\boldsymbol{d})} \\ \boldsymbol{G}_d(\boldsymbol{x}, \boldsymbol{d}) &:= \left.\frac{\partial \boldsymbol{g}^z_{\text{aug}}}{\partial \boldsymbol{d}_{\text{aug}}}\right|_{(\boldsymbol{x},\boldsymbol{d})} \\ \boldsymbol{F}_x(\boldsymbol{x}, \boldsymbol{d}, \boldsymbol{u}) &:= \left.\frac{\partial \boldsymbol{f}_{\text{aug}}}{\partial \boldsymbol{x}_{\text{aug}}}\right|_{(\boldsymbol{x},\boldsymbol{d},\boldsymbol{u})} \\ \boldsymbol{F}_d(\boldsymbol{x}, \boldsymbol{d}, \boldsymbol{u}) &:= \left.\frac{\partial \boldsymbol{f}_{\text{aug}}}{\partial \boldsymbol{d}_{\text{aug}}}\right|_{(\boldsymbol{x},\boldsymbol{d},\boldsymbol{u})} . \end{aligned}$$

Theorem 2 (sufficient condition for observability) *Consider the augmented model (6) with $n_d = n_z$, the fixed* *point* (10)*, and Assumption* 8*. Let*

$$\operatorname{rank} \underbrace{\begin{pmatrix} \boldsymbol{F}_x(\boldsymbol{x},\boldsymbol{d},\boldsymbol{u}) - \boldsymbol{I} & \boldsymbol{F}_d(\boldsymbol{x},\boldsymbol{d},\boldsymbol{u}) \\ \boldsymbol{G}_x(\boldsymbol{x},\boldsymbol{d}) & \boldsymbol{G}_d(\boldsymbol{x},\boldsymbol{d}) \end{pmatrix}}_{\boldsymbol{J}(\boldsymbol{x},\boldsymbol{d},\boldsymbol{u})} = n_x + n_d \tag{12}$$

hold. Then, there exists a neighborhood $\mathcal{U}_\infty \times \mathcal{Z}_\infty \subseteq \mathcal{U} \times \mathcal{Z}$ *of* $(\boldsymbol{u}, \boldsymbol{z})$ *where* (10) *has a unique solution, i.e., Assumption* 6 *is satisfied.*

The proof can be found in [4]. Note that the sufficient observability condition (12) depends on the states $\boldsymbol{x}$, disturbances $\boldsymbol{d}$, and MVs $\boldsymbol{u}$. To guarantee offset-free tracking, the condition has to be satisfied at the obtained steady state, cf. Theorem 1. Hence, the sufficient observability condition (12) can be used to generate suitable disturbance models satisfying (12) in the largest possible subset of the state space, thereby guaranteeing offset-free tracking within this subset. This is the basis for the DM generation approach presented in the next section. In the best case, offset-free tracking is guaranteed within the complete state space.

3 Disturbance Model Generation based on Semi-Infinite Programming

In this section, we describe the new DM generation approach. In the general case of a nonlinear augmented model (10), the sufficient observability condition (12) depends on $\boldsymbol{x}, \boldsymbol{u}$, and $\boldsymbol{d}$. We denote the set for which (12) is satisfied as observable set, similar to the feasible set in optimal experimental design in the context of guaranteed parameter estimation, e.g., [21, 22]. Hence, the observable set can be defined by $\mathbb{O} = \{\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d} : \left(\det(\boldsymbol{J}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d))\right)^2 \geqslant \varepsilon\}$. Fig. 1 illustrates an observable set for a constant value of $\boldsymbol{d}$. The augmented system can be used for offset-free NMPC in the entire observable set; if the controlled system achieves a steady state that lies within the projection of the observable set to the state space, offset-free tracking can be guaranteed. Whether the controlled system achieves a steady state lying in the observable set can be checked a-posteriori or enforced using state constraints. Thus, a large observable set is desirable. The augmented system (6) results from a given nominal model extended by a DM. Thus, the augmented system depends on the choice of the DM and so does the observable set. Different DMs yield different observable sets. This is the key idea behind the DM generation approach we propose. Here, we approximate the observable set by its inner orthotope, although other approximations could also be used, e.g., ellipsoids. We assume the DMs can be parameterized by parameters $\boldsymbol{p}^d \in \mathcal{P}^d$. These parameters may be real numbers as well as integers that define the structure and detailed form of the DMs. Here, we use integers, i.e., $\mathcal{P}^d$ is a discrete set. Although the DM generation approach is not restricted to discrete sets, we do not see an advantage in using continuous parameters to define the DM structure. The DM generation approach targets the DM resulting in the largest possible observable

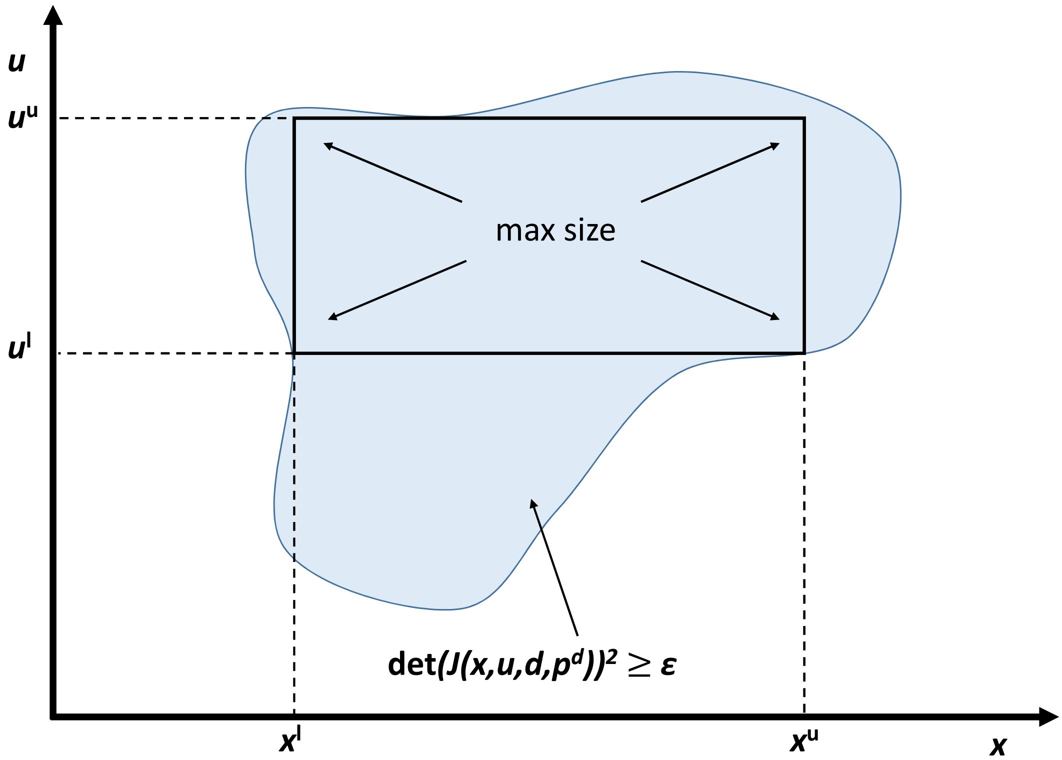


Fig. 1: Observable set and an inner orthotope. For a given parameterization of the DM by the parameters $\boldsymbol{p}^d$, the observable set is defined by $\mathbb{O} = \{\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d} : \left(\det(\boldsymbol{J}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d))\right)^2 \geqslant \varepsilon\}$. The inner orthotope we generate in the DM generation approach is the largest orthotope in $\mathbb{O}$, depending on the specific measure to characterize the size of the orthotope, e.g., volume, surface, or perimeter.

set. Therefore, we formulate a GSIP to maximize the inner orthotope of the observable set, i.e.,

$$\max_{\substack{\boldsymbol{p}^d \in \mathcal{P}^d \\ \boldsymbol{x}^u, \boldsymbol{x}^l \in \mathcal{X} \\ \boldsymbol{d}^u, \boldsymbol{d}^l \in \mathcal{D} \\ \boldsymbol{u}^u, \boldsymbol{u}^l \in \mathcal{U}}} \sum_{i=1}^{n_x} w_{x,i}(x_i^u - x_i^l) + \sum_{i=1}^{n_d} w_{d,i}(d_i^u - d_i^l) + \sum_{i=1}^{n_u} w_{u,i}(u_i^u - u_i^l) \tag{13a}$$

$$s.t. \quad \det\left(\boldsymbol{J}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d)\right)^2 \geqslant \varepsilon, \forall\, (\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}) \in [\boldsymbol{x}^l, \boldsymbol{x}^u] \times [\boldsymbol{u}^l, \boldsymbol{u}^u] \times [\boldsymbol{d}^l, \boldsymbol{d}^u] \tag{13b}$$

$$\boldsymbol{J}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) = \begin{pmatrix} \boldsymbol{F}_{\boldsymbol{x}}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) - \boldsymbol{I} & \boldsymbol{F}_{\boldsymbol{d}}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) \\ \boldsymbol{G}_{\boldsymbol{x}}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) & \boldsymbol{G}_{\boldsymbol{d}}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) \end{pmatrix} \tag{13c}$$

$$\boldsymbol{h}(\boldsymbol{x}^u, \boldsymbol{x}^l, \boldsymbol{u}^u, \boldsymbol{u}^l, \boldsymbol{d}^u, \boldsymbol{d}^l, \boldsymbol{p}^d) = \boldsymbol{0} \tag{13d}$$

$$\boldsymbol{g}(\boldsymbol{x}^u, \boldsymbol{x}^l, \boldsymbol{u}^u, \boldsymbol{u}^l, \boldsymbol{d}^u, \boldsymbol{d}^l, \boldsymbol{p}^d) \leqslant \boldsymbol{0}, \tag{13e}$$

where (13b) guarantees that (12) is satisfied. Note that $\boldsymbol{J}$ in (13b) now depends on $\boldsymbol{p}^d$ since the disturbance model is parameterized by $\boldsymbol{p}^d$ and that $n_z = n_d$, cf. Section 2.2.1. Since (13) has $n_p + 2n_x + 2n_z + 2n_u$ variables, the problem size scales linearly with the number of variables of the augmented model and the number of parameters to parameterize the DM. Note that we use $n_z = n_d$. Overall, the problem scales linearly with the number of parameters used to parameterize the DM, the number of differential states, the number of measurements, and the number of CVs. We can use the constraints (13d) and (13e) with the functions $\boldsymbol{h}$: $\mathcal{X} \times \mathcal{X} \times \mathcal{U} \times \mathcal{U} \times \mathcal{D} \times \mathcal{D} \times \mathcal{P}^d \to \mathbb{R}^{n_h}$ and $\boldsymbol{g} : \mathcal{X} \times \mathcal{X} \times \mathcal{U} \times \mathcal{U} \times \mathcal{D} \times \mathcal{D} \times \mathcal{P}^d \to \mathbb{R}^{n_g}$, e.g., if we aim at generating a model used to track one specific point, the constraint $\boldsymbol{g}^y_{\text{aug}}(\boldsymbol{x}, \boldsymbol{d}) = \boldsymbol{r}_\infty$ can directly be added to the problem. Furthermore, the constraints can be used to add disturbances to specific equations, e.g., to those equations which correspond to unmeasured CVs, thereby enabling exact steady state predictions for those variables as required by Assumption 2 for offset-free NMPC, cf. Theorem 1. On the other hand, without these constraints, a DM is generated by the solution of (13) which can be used for offset-free tracking within a set of the state space that is defined by the boundaries $\boldsymbol{x}^u, \boldsymbol{x}^l, \boldsymbol{d}^u, \boldsymbol{d}^l$, and $\boldsymbol{u}^u, \boldsymbol{u}^l$.

For linear offset-free MPC, i.e., linear models and linear disturbance models, cf. [8], the sufficient observability condition (12) does not depend on $\boldsymbol{x}, \boldsymbol{d}$, and $\boldsymbol{u}$. Given an observable nominal model, the sufficient observability condition (12) solely depends on the choice of disturbance model, and so does (13b). Hence, the GSIP (13) reduces to a finite optimization problem.

The determinant in (13b) may be difficult to evaluate for large matrices $\boldsymbol{J}$. If we assume that either $\boldsymbol{F_x}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) - \boldsymbol{I}$ or $\boldsymbol{G_d}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d)$ is invertible, we can instead make use of the Schur complement of $\boldsymbol{J}$ and equivalently replace (13b) by either $\det(\boldsymbol{S}_A(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d))^2 \geqslant \varepsilon / \det(\boldsymbol{F_x}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) - \boldsymbol{I})^2$ or $\det(\boldsymbol{S}_D(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d))^2 \geqslant \varepsilon / \det(\boldsymbol{G_d}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d))^2$, where $\boldsymbol{S}_A$ and $\boldsymbol{S}_D$ are the Schur complements of $\boldsymbol{J}$ defined by $\boldsymbol{S}_A = \boldsymbol{G_d}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) - \boldsymbol{G_x}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d)(\boldsymbol{F_x}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) - \boldsymbol{I})^{-1}\boldsymbol{F_d}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d)$ and
$\boldsymbol{S}_D = \boldsymbol{F_x}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) - \boldsymbol{I} - \boldsymbol{F_d}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d)(\boldsymbol{G_d}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d))^{-1}\boldsymbol{G_x}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d)$, cf. [23]. Note, that the determinants of the Schur complements and the determinants of $\boldsymbol{F_x}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) - \boldsymbol{I}$ and $\boldsymbol{G_d}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d)$ may be easier to compute, as the sizes of these matrices are smaller. (13b) could also be replaced using only the respective Schur complements with either $S_A(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) \geqslant \varepsilon$ or $S_D(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) \geqslant \varepsilon$, thereby only requiring evaluation of the determinant of these matrices. Although not yielding an equivalent GSIP, this would guarantee the satisfaction of the observability condition.

The constraint (13b) is problematic in those cases where $\boldsymbol{J}$ in (13) is structurally singular for given parameter values of $\boldsymbol{p}$. In this case, (13b) violates the linear independence constraint qualification (LICQ) and the Mangasarian-Fromovitz constraint qualification (MFCQ), cf. [24]. Therefore, we exclude a-priori those realizations of $\boldsymbol{p}^d$ leading to structural singularity of $\boldsymbol{J}$. This can be performed for instance using the approach that we provide in Appendix B. Herein, we solve optimization problems that become more and more restrictive. Thus, we first determine those realizations of $\boldsymbol{p}^d$ resulting in a $\boldsymbol{J}$ that is structurally singular and we exclude those parameter realizations before solving (13).

Alternatively, we could reformulate (13) using complementarity constraints, which can then be relaxed or reformulated to avoid violation of the constraint qualifications, cf. [24, 25, 26].

The GSIP (13) can be reformulated to account for other approximations of the observable set. In the specific case of an orthotope approximating the observable set, we can reformulate the GSIP (13) to a SIP using $\boldsymbol{x} = \boldsymbol{x}^l + \boldsymbol{\xi}^x \odot (\boldsymbol{x}^u - \boldsymbol{x}^l)$, $\boldsymbol{d} = \boldsymbol{d}^l + \boldsymbol{\xi}^d \odot (\boldsymbol{d}^u - \boldsymbol{d}^l)$, $\boldsymbol{u} = \boldsymbol{u}^l + \boldsymbol{\xi}^u \odot (\boldsymbol{u}^u - \boldsymbol{u}^l)$, adding $\boldsymbol{\xi}^x \in [0,1]^{n_x}, \boldsymbol{\xi}^u \in [0,1]^{n_u}, \boldsymbol{\xi}^d \in [0,1]^{n_d}$

as additional degrees of freedom, and substitute (13b) by

$$\det\Big(\boldsymbol{J}\big(\boldsymbol{x}^l + \boldsymbol{\xi}^x \odot (\boldsymbol{x}^u - \boldsymbol{x}^l), \boldsymbol{u}^l + \boldsymbol{\xi}^u \odot (\boldsymbol{u}^u - \boldsymbol{u}^l), \boldsymbol{d}^l + \boldsymbol{\xi}^d \odot (\boldsymbol{d}^u - \boldsymbol{d}^l), \boldsymbol{p}^d\big)\Big)^2 \geqslant \varepsilon,$$
$$\forall \left(\boldsymbol{\xi}^x, \boldsymbol{\xi}^d, \boldsymbol{\xi}^u\right) \in [0,1]^{n_x} \times [0,1]^{n_d} \times [0,1]^{n_u},$$

where $\odot$ is the component wise multiplication. The same reformulation was used by Lemonidis [27] to reformulate a GSIP resulting from the flexibility problem in kinetic model reduction. We expect the SIP to be easier to solve than the original GSIP with the algorithm that we use for the case studies. Thus, the following SIP arises:

$$\max_{\substack{\boldsymbol{p}^d \in \mathcal{P}^d \\ \boldsymbol{x}^u, \boldsymbol{x}^l \in \mathcal{X} \\ \boldsymbol{d}^u, \boldsymbol{d}^l \in \mathcal{D} \\ \boldsymbol{u}^u, \boldsymbol{u}^l \in \mathcal{U}}} \sum_{i=1}^{n_x} w_{x,i}(x_i^u - x_i^l) + \sum_{i=1}^{n_d} w_{d,i}(d_i^u - d_i^l) + \sum_{i=1}^{n_u} w_{u,i}(u_i^u - u_i^l) \tag{14a}$$

$$s.t. \quad \det\Big(\boldsymbol{J}\big(\boldsymbol{x}^l + \boldsymbol{\xi}^x \odot (\boldsymbol{x}^u - \boldsymbol{x}^l), \boldsymbol{u}^l + \boldsymbol{\xi}^u \odot (\boldsymbol{u}^u - \boldsymbol{u}^l), \boldsymbol{d}^l + \boldsymbol{\xi}^d \odot (\boldsymbol{d}^u - \boldsymbol{d}^l), \boldsymbol{p}^d\big)\Big)^2 \geqslant \varepsilon, \tag{14b}$$
$$\forall \left(\boldsymbol{\xi}^x, \boldsymbol{\xi}^d, \boldsymbol{\xi}^u\right) \in [0,1]^{n_x} \times [0,1]^{n_d} \times [0,1]^{n_u}$$

$$\boldsymbol{J}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) = \begin{pmatrix} \boldsymbol{F}_{\boldsymbol{x}}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) - \boldsymbol{I} & \boldsymbol{F}_{\boldsymbol{d}}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) \\ \boldsymbol{G}_{\boldsymbol{x}}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) & \boldsymbol{G}_{\boldsymbol{d}}(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}, \boldsymbol{p}^d) \end{pmatrix} \tag{14c}$$

$$\boldsymbol{h}(\boldsymbol{x}^u, \boldsymbol{x}^l, \boldsymbol{u}^u, \boldsymbol{u}^l, \boldsymbol{d}^u, \boldsymbol{d}^l, \boldsymbol{p}^d) = \boldsymbol{0} \tag{14d}$$

$$\boldsymbol{g}(\boldsymbol{x}^u, \boldsymbol{x}^l, \boldsymbol{u}^u, \boldsymbol{u}^l, \boldsymbol{d}^u, \boldsymbol{d}^l, \boldsymbol{p}^d) \leqslant \boldsymbol{0}. \tag{14e}$$

We embed (14) in the overall model generation approach which is depicted in Fig. 2. The approach starts with defining a parameterization of the DM and performing the preprocessing, i.e., excluding all those parameter realizations leading to a structural singularity of $\boldsymbol{J}$. In the case of unmeasured CVs, exact steady state predictions for these CVs are necessary to guarantee offset-free NMPC. This can be obtained by either considering an exact DM (if known) or by adding disturbances to those equations that correspond to the unmeasured CVs. After parameterizing the remaining part of the DM we decide whether specific constraints should be added to the SIP (14), e.g., if the resulting DM should be valid at a certain point (e.g. the setpoint). After parameterization and defining additional constraints, the SIP (14) is solved. A suitable DM results if sufficient optimal values are found for the DM parameterization of the bounds of the observable set, resulting from the solution of (14). These bounds are sufficient, for example, if the resulting orthotope includes the entire state space of the augmented model or a region that captures relevant parts of the state space. A new parameterization has to be defined if the SIP (14) is infeasible or the bounds of the observable set orthotope are not sufficient. If a different parameterization is not available, no suitable DM exists for the desired task. In the case studies presented in Section 4, we use linear disturbance models leading to a sufficiently large observable set, i.e., there is no need for refining the DM parameterization. Note, the proposed approach in Fig. 2 is applied offline. The approach finds the structure as well as the parameter values of the disturbance model. The resulting augmented model can then be used online for offset-free NMPC. The overall offset-free NMPC scheme including the model generation approach is illustrated in Fig. 3.

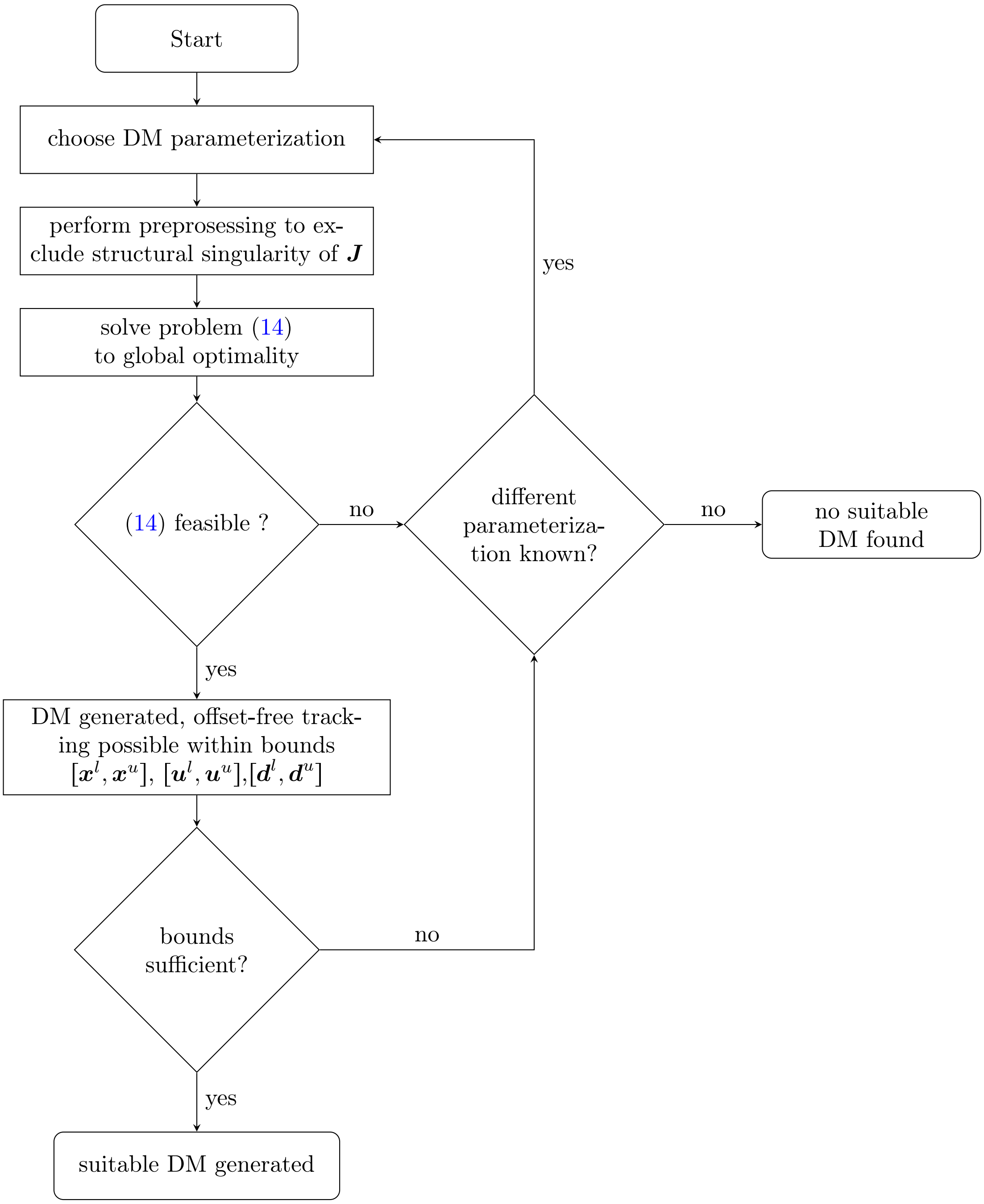


Fig. 2: DM generation approach.

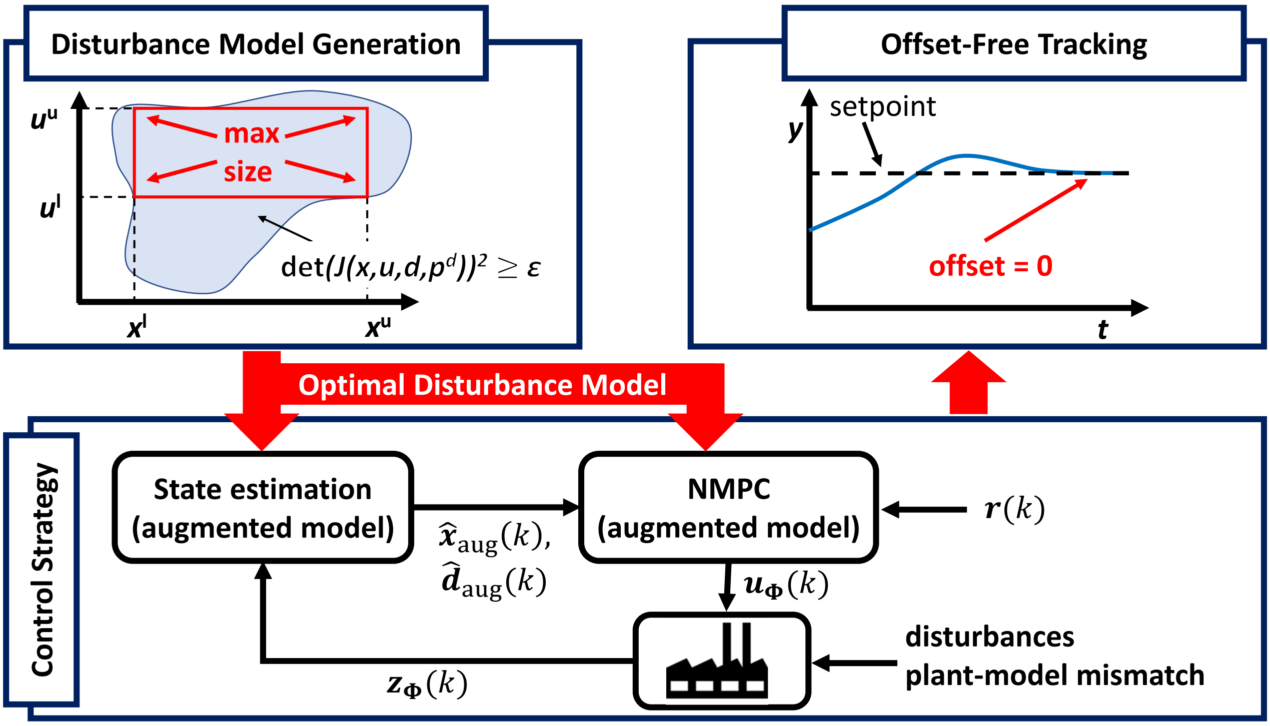


Fig. 3: Illustration of overall offset-free NMPC scheme including disturbance model generation. The optimal disturbancs models are generated offline and used online as controller models in state estimation and NMPC. Thus, offset-free tracking is guaranteed within the observable set of the augmented system (nominal model and optimal disturbance model).

3.1 Solving SIPs

The DM generation approach requires the solution of the SIPs (14), which have the following general form:

$$\begin{aligned} &\min_{\boldsymbol{\chi} \in \mathfrak{X}} f(\boldsymbol{\chi}) \\ s.t. \quad &\boldsymbol{g}(\boldsymbol{\chi}, \boldsymbol{\psi}) \leqslant \mathbf{0}, \forall \boldsymbol{\psi} \in \boldsymbol{\Psi}. \end{aligned} \tag{15}$$

We give a brief overview of the approach to solve (15) in this section. We use the algorithm of Djelassi and Mitsos [28]. While this algorithm is based on discretization, there are other approaches and we refer to [29, 30, 20] for thorough overviews of SIPs and solution techniques.

SIPs are mathematical programs of the form (15) with a finite number of variables and an infinite number of constraints, expressed by parameterized constraints $\boldsymbol{g}(\boldsymbol{\chi}, \boldsymbol{\psi}) \leqslant \mathbf{0}, \forall \boldsymbol{\psi} \in \boldsymbol{\Psi}$, which have to be satisfied for all possible realizations of their parameters $\boldsymbol{\psi} \in \boldsymbol{\Psi}$. The difficulty of SIPs arises from the infinite nature of the constraints $\boldsymbol{g}(\boldsymbol{\chi}, \boldsymbol{\psi}) \leqslant \mathbf{0}, \forall \boldsymbol{\psi} \in \boldsymbol{\Psi}$. A point $\bar{\boldsymbol{\chi}}$ is feasible, if $\bar{\boldsymbol{\chi}} \in \mathfrak{X}$ and if the supremum g_i^* of the lower-level program (LLP) is positive:

$$g_i^* = \sup_{\boldsymbol{\psi} \in \boldsymbol{\Psi}} g_i(\bar{\boldsymbol{\chi}}, \boldsymbol{\psi}) \leqslant 0, \;\; \forall i \in \{1, ..., N_g\}, \tag{16}$$

where N_g is the number of constraints. If the set $\boldsymbol{\Psi}$ is convex and the constraints $g_i(\boldsymbol{\chi}, \cdot)$ are concave on $\boldsymbol{\Psi}$, then the LLP (16) is a convex problem (given a constraint qualification) and can be replaced in the SIP (15) by the Karush-Kuhn-Tucker (KKT) conditions of the lower-level program (16). As a consequence, the SIP (15) reduces to a finite optimization problem. In the general case of non-concave constraints $\boldsymbol{g}(\boldsymbol{\chi}, \cdot)$, this approach is not applicable. In this case, discretization approaches, such as the algorithm proposed by Blankenship and Falk [31], can be used to solve the SIP (15), where the set $\boldsymbol{\Psi}$ is replaced by a finite discretization of $\boldsymbol{\Psi}$, leading to a finite optimization problem that approximates the SIP and can be solved using NLP solvers. The algorithm of Djelassi and Mitsos [28] is based on this paradigm. As an adaptive discretization algorithm in the vein of Blankenship and Falk [31], the hybrid discretization algorithm of Djelassi and Mitsos [28] relies on solving multiple finite NLP subproblems to global optimality. Indeed, the algorithm solves multiple problems that are derived from a given SIP by considering only a finite subset of the infinitely many constraints $\boldsymbol{g}(\boldsymbol{\chi}, \cdot) \leqslant \mathbf{0}, \forall \boldsymbol{\psi} \in \boldsymbol{\Psi}$, through replacement of the set $\boldsymbol{\Psi}$ by a finite discretization of $\boldsymbol{\Psi}$. Furthermore, to assess the feasibility of a given iterate, the maximum violation of the semi-infinite constraint is computed via the global solution of the lower-level program (16) of the SIP. Based on this, convergence and finite termination of the algorithm are guaranteed by refining the finite discretization of the set $\boldsymbol{\Psi}$ (thereby the set of constraints) if necessary and manipulating further algorithmic parameters appropriately [28].

4 Case Studies

We apply the proposed DM generation approach and use the generated models in closed-loop simulation case studies. We consider linear DM, where the disturbances are added to the differential equations, i.e., we use the following form of the augmented model (6):

$$\boldsymbol{f}_{\text{aug}}(\boldsymbol{x}(k), \boldsymbol{u}(k), \boldsymbol{d}(k)) = \boldsymbol{f}(\boldsymbol{x}(k), \boldsymbol{u}(k)) + \boldsymbol{B}_d(\boldsymbol{p}^d) \cdot \boldsymbol{d}(k), \tag{17}$$

where $\boldsymbol{p}^d \in \mathcal{P}^d = \{0,1\}^{n_z \times 1 + n_x - n_z}$ are the parameters of the DM gathered in the matrix

$$\boldsymbol{B}_d(\boldsymbol{p}^d) = \begin{pmatrix} p^d_{1,1} & 0 & 0 \\ \vdots & \ddots & 0 \\ p^d_{1,1+n_x-n_z} & \ddots & p^d_{n_z,1} \\ 0 & \ddots & \vdots \\ 0 & 0 & p^d_{n_z,1+n_x-n_z} \end{pmatrix}, \tag{18}$$

with which we can distribute n_z disturbances over n_x equations. Note that it is not relevant for the observability condition which specific disturbance is added to which equations. Hence, the disturbance model parameterization with the matrix (18) allows for all possible linear DM realizations. In the case of a linear DM, it is irrelevant which disturbance is added to an equation. There is no advantage of adding more than one disturbance to an equation or adding one disturbance to more than one equation. Adding one disturbance to multiple equations would only be useful if the exact DM, e.g., an input disturbance model, would be known. This could be considered in the DM generation approach by choosing an appropriate DM parameterization. Although this is a simple choice of a DM parameterization, we obtain augmented models that satisfy the sufficient condition for observability within the entire state space or within relevant regions of the state space. If this would not be the case, other parameterizations could be used. For example, universal approximators, such as polynomials or artificial neural networks [32] could be used allowing for any functional correlation to be used as a disturbance model. However, this could complicate the DM generation procedure, as it has to estimate the parameters of the functional correlation to determine the optimal DM. Thus, we use the simple linear DM form. Nevertheless, we demonstrate that we obtain suitable DMs that can be used for offset-free tracking. Furthermore, we target a suitable DM to track offset-free the given setpoints $\boldsymbol{r}_\infty$. Thus, we add $\boldsymbol{g}^y_{\text{aug}}(\boldsymbol{x}, \boldsymbol{d}) = \boldsymbol{r}_\infty$ to (14). This reduces the degrees of freedom in the DM generation problem. It would be possible to generate a DM that is suitable for offset-free tracking of setpoints within a range instead of discrete setpoints. Then, the constraints $\boldsymbol{g}^y_{\text{aug}}(\boldsymbol{x}, \boldsymbol{d}) = \boldsymbol{r}_\infty$ are not required for the respective setpoints that are to be tracked within a range. In addition, we want to add exactly n_z disturbances to the nominal model, as sufficient for offset-free tracking, and therefore use the integer constraint $\sum_{j=1}^{1+n_x-n_z} p^d_{i,j} = 1, \forall i \in \{1, ..., n_z\}$ and $\sum_{i=1}^{n_z} \sum_{j=1}^{1+n_x-n_z} p^d_{i,j} = n_z$ to (14).

We solve the resulting integer problems using enumeration by solving a SIP for every DM instance satisfying the integer constraints and finally select those DM for which the highest objective function results.

Using enumeration to solve the integer problems, we have to solve one SIP for every integer realization. We solve each individual SIP using the hybrid discretization algorithm proposed by Djelassi and Mitsos [28]. In particular, we employ a C++ implementation of the algorithm that makes use of the C++ library libALE [33]. To solve subproblems of the SIP algorithm to global optimality, the implementation employs the deterministic global optimization solver MAiNGO [34]. The parameters of the hybrid discretization algorithm are chosen to be 10^{-4} for the initial restriction parameter and 2 for the reduction parameter. The SIPs are solved to an absolute and relative optimality tolerance of 10^{-4} for the case studies I and II and 10^{-2} for case study III. All NLP subproblems are solved to optimality tolerances which are in line with the requirements outlined in [28] and all feasibility tolerances are set to 10^{-9}.

Since the underlying library MC++ [35] in MAiNGO does not provide the determinant as an intrinsic function, the only missing ingredient for the solution of (14) is an appropriate representation of the determinant in (14b). Accordingly, we employ a Laplace expansion [36] to represent the determinant symbolically. It is well known that Laplace expansion has unfavorable scaling of $\mathcal{O}(n!)$ [37, 36]. In contrast, other algorithms such as LU decomposition admit scaling of order $\mathcal{O}(n^3)$ [37, 36]. For the small-scale case studies discussed in the following, the Laplace expansion approach is feasible since the number of terms in the expansion remains manageable. For larger cases, the determinant could be calculated numerically. To still be able to calculate the relaxations, the selected numerical method must then satisfy the requirements for the generation of McCormick-based relaxations of algorithms as proposed in [38]. The most important requirement is that the algorithm must terminate in a known maximum number of iterations, which is given for the Laplace expansion as well as multiple alternative algorithms with a more favorable scaling behavior. The second crucial requirement is that the algorithms cannot have general if-then-else statements. In particular, such statements that control the progress of the algorithm, e.g., those found in pivoting in the solution of linear systems.

We consider the generated models in closed-loop NMPC case studies. We use multiple-shooting [39] (case study Chemical Reactor I) and single-shooting [40] (case studies Chemical Reactor II and Polymerization Reactor) implemented in DyOS [41] for the solution of the dynamic optimization problems of the NMPC. We apply multiple-shooting since it can be used for the dynamic optimization of unstable systems [39], such as the chemical reactor case study that we consider. The resulting nonlinear programming problems (NLPs) within the direct shooting approaches are solved to local convergence using SNOPT [42] with optimality tolerance of 10^{-4}. We apply the integrator NIXE [43] with integration tolerances of 10^{-8}. The process is emulated in-silico, using NIXE [43] with integration tolerances of 10^{-8}. We use an Extended Kalman Filter (EKF) [44] for state estimation. We assume that there is no measurement noise. However, measurement noises satisfying the assumption of asymptotically constant disturbances can be interpreted as disturbances and are, thus, covered by Theorem 1.

We present three case studies to illustrate the DM generation approach and the relevance and usefulness of the relaxed sufficient conditions in Theorem 1. The first case study uses $n_z = n_y$, whereas the second case study uses $n_z > n_y$. In the first and second case study, we first consider that all CVs are measured. Then we also deal with unmeasured CVs. The third case study assumes all CVs to be measured and demonstrates the application of the approach to a medium scale example. The first and second case studies use models with three and four differential states, respectively, and the model in the third case study comprises nine differential states.

4.1 Case Study I: Simple Chemical Reactor

We use the chemical reactor model from [8, 2, 5], which is provided in Appendix C. Fig. 4 shows a flowsheet of the chemical reactor. The feed stream to the reactor has the temperature T_0, flowrate F_0, and concentration c_0. A reaction takes place inside the reactor, which is surrounded by a cooling jacket, where cooling water is fed in with a temperature of T_c.

The CVs are $\boldsymbol{y}(t) = (c(t), h(t))$ and the MVs $\boldsymbol{u}(t) = (F(t), T_c(t))$, where F is the outlet flow rate and T_c is the cooling temperature, with the bounds $F(t) \in [0, 0.25]$ m^3/min and $T_c(t) \in [295, 305]$ K. We emulate a plant-model mismatch due to a disturbance of the feed flowrate F_0, i.e., the plant is simulated with $F_0(t) = 0.1$ m^3/min$, 0 \leqslant t < 5$ min and $F_0(t) = 0.11$ m^3/min$, t \geqslant 5$ min, whereas the NMPC controller model uses $F_0(t) = 0.1$ m^3/min$, \forall t$.

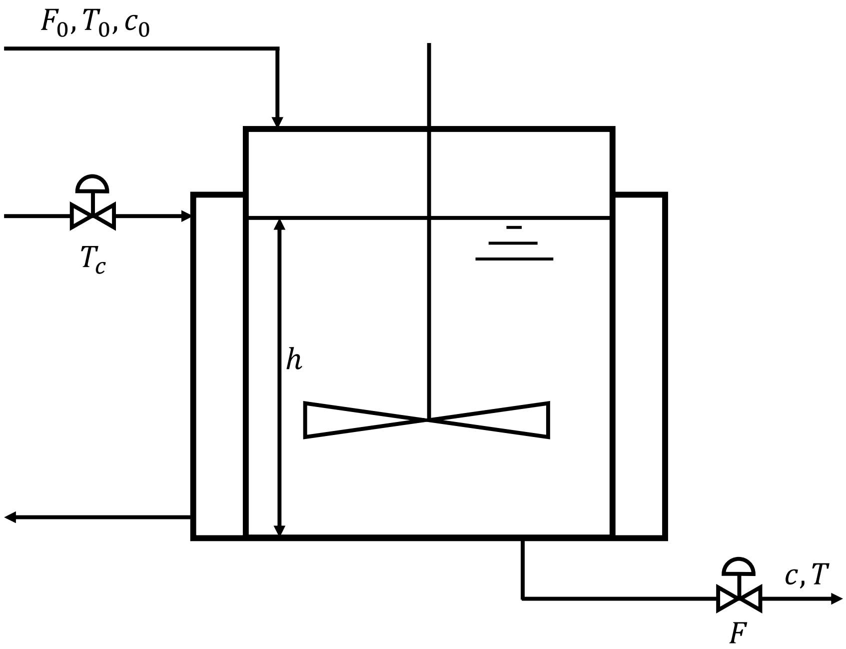


Fig. 4: Flowsheet of chemical reactor for case study I.

As in [8], the control task is to track a constant setpoint for c and h using the following objective in (8a)

$$l = (c_t - 0.5 \text{ kmol/m}^3)^2 + (h_t - 0.6599 \text{ m})^2.$$

The NMPC has a sampling time of 0.2 min and both control and prediction horizon set to 10 min. We discretize the continuous process model with the same sampling time of 0.2 min to obtain a discrete-time system to which we can apply the DM generation approach.

In the model generation approach, we use for (14) the constraints $T^l, T^u \in [250, 450]$ K, $T_c^l, T_c^u \in [295, 305]$ K, and $c = 0.5$ kmol/m^3, $h = 0.6599$ m, as we target offset-free tracking at these specific setpoint. We constrain the squared determinant in (14) to be larger than $\epsilon = 10^{-4}$.

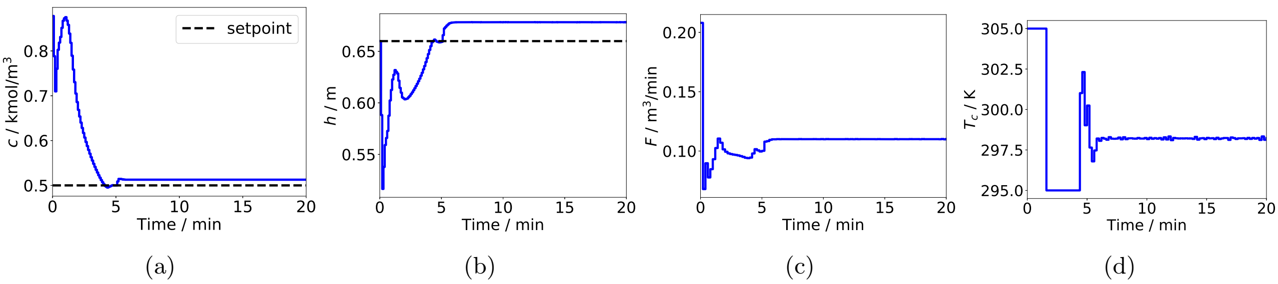


Fig. 5: Results of closed-loop simulation using NMPC without DM. Offset-free tracking is not achieved, see (a) and (b). (a) Reactor concentration (CV). (b) Reactor height (CV). (c) Feed flowrate (MV). (d) Cooling fluid temperature (MV).

Fig. 5 shows the closed-loop simulation results with nominal NMPC and full state-feedback, i.e., all states are measured. We see that the nominal NMPC leads to a steady state but with a clear offset. This demonstrates the need for offset-free NMPC.

4.1.1 All CVs measured

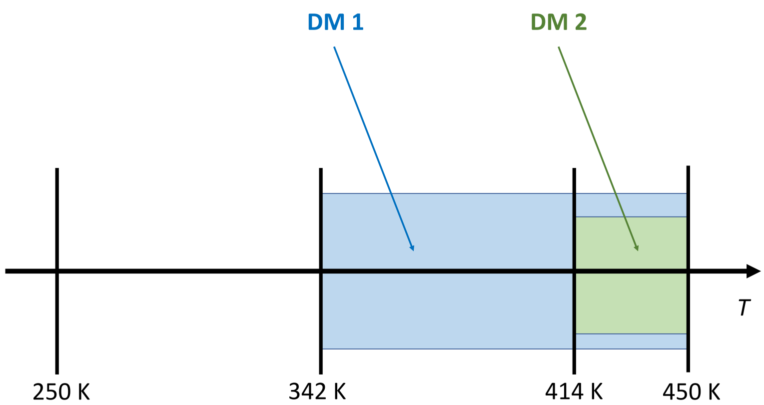


Fig. 6: Observable sets for reactor case study for augmented model with DM 1 and DM 2. Projection to T axis. DM 1 results in a larger observable set. The observable set has a size of zero for DM 3.

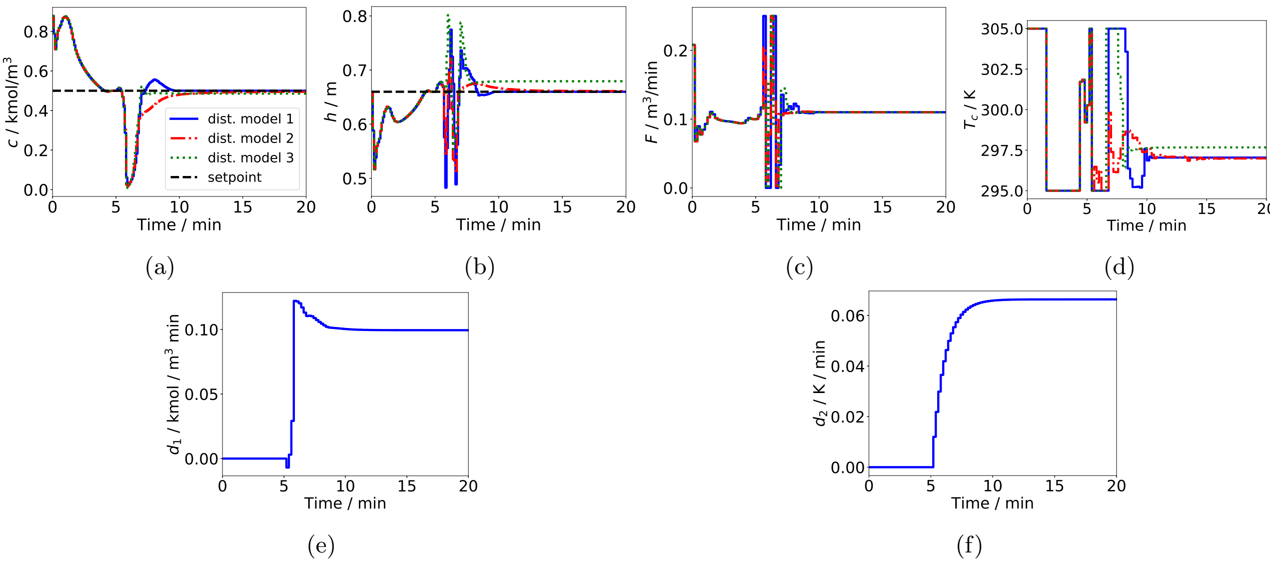


Fig. 7: Results of closed-loop simulation using NMPC with DM. DM 1 is the solution of the DM generation approach in Fig. 2. (a) Reactor concentration (CV). (b) Reactor height (CV). (c) Feed flowrate (MV). (d) Cooling fluid temperature (MV). (e) Disturbance 1 with DM 1. (f) Disturbance 2 with DM 1.

We assume the CVs to be measured, i.e., $\boldsymbol{z}(t) = (c(t), h(t))$ and use the weights $w_T = 1$ and $w_F = 500$ in (14). All other weights are zero as the determinants in (14) do not depend on the corresponding variables. The observable set does in turn include the whole state space for these variables.

We analyze the three possible DMs, where up to one disturbance is added to each differential equation. DM

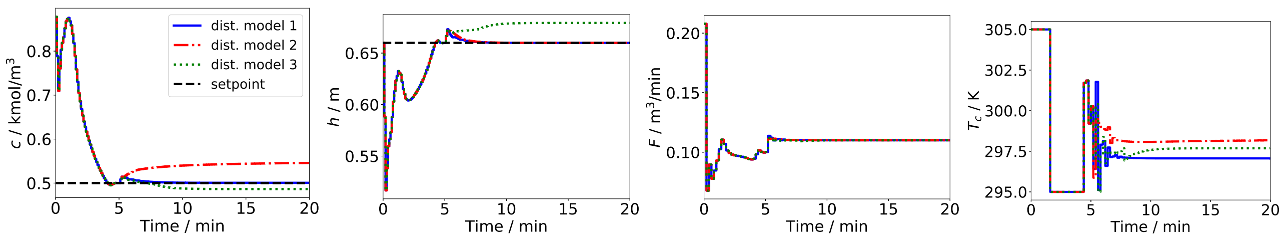
1, with $p^d_{1,1} = 1$, $p^d_{2,2} = 1$, DM 2, with $p^d_{1,2} = 1$, $p^d_{2,2} = 1$, and DM 3, with $p^d_{1,1} = 1$, $p^d_{2,1} = 1$.

For DM 2, we performed a polynomial regression to fit a polynomial of order 5 to the squared determinant. This is done since the original expression for the determinant contains a division by zero. However, the expression does not diverge due to this division by zero and a polynomial fit can be obtained. In the SIP we used the polynomial instead of the original squared determinant expression in (14b). We validate the optimization results a posteriori by calculating the actual value of the determinant.

For DM 1, the solution of (14) yields $T_l = 341.983$ K, and $T_u = 450$ K and the complete defined state space for the other variables. The SIP algorithm converges within 30 iterations and with a total of 55 subproblems solved. For DM 2, the solution of (14) yields $T_l = 413.878$ K, $T_u = 450$ K, and the complete defined state space as observable set for the other variables. The SIP algorithm converges within 11 iterations and with a total of 20 subproblems solved. For DM 3, the determinant in (14) is zero and therefore, problem (14) is infeasible. That is, DM 3 is not a suitable DM as the sufficient observability condition (12) is violated in the entire state space.

The projection of the observable sets to the T axis for DM 1 and DM 2 are illustrated in Fig. 6. We see that DM 1 results in a larger observable set. Consequently, DM 1 results from the proposed approach in Fig. 2. Fig. 7 shows that using DM 1, the setpoints can be tracked with zero offset. In contrast, there is an offset when the other DMs are used. The DM approach in Fig. 2 successfully finds a DM that can be used for offset-free NMPC. Figs. 7e and 7f shows the profiles of the disturbances with DM 1, which are estimated by the EKF. We see that they are at zero as long as no disturnbance exists and increase from 5 min on, when the disturbance exists, until they reach their steady state values.

4.1.2 Not all CVs measured



(a) (b) (c) (d)

Fig. 8: Results of closed-loop simulation using NMPC with DM. DM 1 is the solution. (a) Reactor concentration (unmeasured CV). (b) Reactor height (CV). (c) Feed flowrate (MV). (d) Cooling fluid temperature (MV)

Now, we consider the case that not all CVs are measured and assume the states T and h to be measured: $\boldsymbol{z}(t) = (T(t), h(t))$, i.e., c is an unmeasured CV, and use the weights $w_T = 1$ and $w_F = 500$ in (14). All other weights are zero as the determinants in (14) do not depend on the corresponding variables. The observable set does in turn include the whole state space for these variables.

We analyze the three possible DMs, i.e., DMs where up to one disturbance is added to each differential equation. DM 1, with $p^d_{1,1} = 1$, $p^d_{2,2} = 1$, DM 2, with $p^d_{1,2} = 1$, $p^d_{2,2} = 1$, and DM 3, with $p^d_{1,1} = 1$, $p^d_{2,1} = 1$.

For DM 1 the solution of the SIP (14) yields $T_l = 290.504$ K, $T_u = 450$ K, and the complete defined state space as observable set for the other variables. The SIP algorithm terminates within 10 iterations and with a total of 15 subproblems solved. For DM 2, the solution of (14) yields $T_l = 353.995$ K, $T_u = 450$ K, and the complete defined state space as observable set for the other variables. The SIP algorithm terminates within 15 iterations and with a total of 24 subproblems solved. For DM 3, (14) is known to be infeasible a priori since the sufficient observability condition (12) is not satisfied for this DM.

Theorem 1 assumes exact steady state predictions for the unmeasured CVs. As discussed previously, this can be achieved by adding disturbances to those equations corresponding to the unmeasured CVs. Note that with DM 2, no disturbance is added to the equation corresponding to the unmeasured CV c. To guarantee offset-free tracking, we add one disturbance to the first differential equation to achieve accurate steady state prediction for c, cf. Theorem 1. Therefore, we do not consider DM 2 in the model generation approach. This is realized by adding $p^d_1 = 1$ to (14). By this, we account for the unmeasured CV by adding one disturbance to the corresponding differential equation.

Consequently, DM 1 results from the DM generation approach in Fig. 2. Fig. 8 shows that using DM 1, the setpoints can be tracked with zero offset. In contrast, there is an offset when the other DMs are used.

The case study demonstrates that offset-free tracking can be achieved although not all CVs are measured. However, the unmeasured CVs are predicted exactly at the steady state, since DM 1 adds disturbances to the equation corresponding to the unmeasured CV. Although DM 3 adds a disturbance to this equation, too, the resulting augmented model does not satisfy the sufficient observability condition (12). In fact, it is not observable and the closed-loop results show an offset. DM 2 is observable due to (12), however, it does not add a disturbance to the equation corresponding to the unmeasured CV c and hence does not satisfy Assumption 2.

4.2 Case Study II: Chemical Reactor

We use the chemical reactor model from Santos et al. [45]. This case study is an illustrative example for $n_z > n_y$. The model is provided in Appendix D.

The CVs are the reactor height and temperature $\boldsymbol{y}(t) = (h(t), T_R(t))$ and the MVs are the reactor outlet flowrate and the cooling water flowrate $\boldsymbol{u}(t) = (F_3(t), F_J(t))$ with the bounds $F_3(t) \in [0, 12 \cdot 10^{-3}/60]$ mol/s and $F_J(t) \in [0, 76 \cdot 10^{-3}/60]$ mol/s.

As in [45], the control task is to track a constant setpoint for h and T_R using the following objective in (8a)

$$l = (h - 0.3 \text{ m})^2 + (10^{-2} \cdot (T_R - 37.5°\text{C}))^2$$

We use an NMPC sampling time of 30 s and both control and prediction horizon set to 10 min. We emulate a plant model mismatch by simulating the plant using the nominal model with $F_2(t) = 4 \cdot 10^{-3}$ mol/min, $0 \text{ h} \leqslant$

$t < 0.5$ h and $F_2(t) = 3.2 \cdot 10^{-3}$ mol/min, 0.5 h $\leqslant t$. In contrast, the NMPC controller model used $F_2(t) = 4 \cdot 10^{-3}$ mol/min, $\forall t$. We discretize the augmented model with the same sampling time of 30 s to obtain a discrete-time system to which we can apply the DM generation approach.

In the model generation approach, we constrain the squared determinant in (14) to be larger than $\epsilon = 10^{-4}$. The feasible state space is $\mathbb{R}$ for all bounds of the orthotope approximating the observable set.

We perform a closed-loop NMPC simulation with nominal NMPC and with full state-feedback. Fig. 9 shows the closed-loop results. We see that the plant model mismatch from 0.5 h on leads to an offset in both CVs (Figs. 9c and 9d). This demonstrates the need for offset-free NMPC.

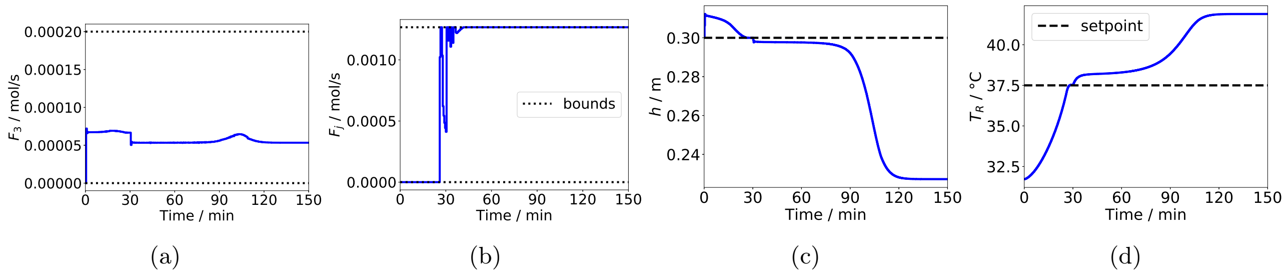


Fig. 9: Results of closed-loop simulation using nominal NMPC. Offset-free tracking is not achieved, see (c) and (d). (a) Product flowrate (MV). (b) Cooling water flowrate (MV). (c) Height of reactor holdup (CV). (d) Reactor temperature (CV).

4.2.1 All CVs measured

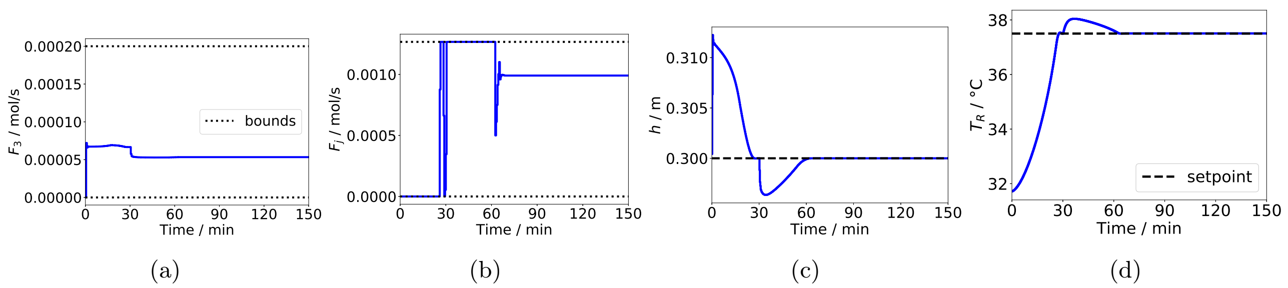


Fig. 10: Results of closed-loop simulation using offset-free NMPC with DM 3. DM 3 is the solution. (a) Product flowrate (MV). (b) Cooling water flowrate (MV). (c) Height of reactor holdup (CV). (d) Reactor temperature (CV).

The measurements are $\boldsymbol{z}(t) = (V(t), T_R(t), T_J(t))$ and we use the weights $w_{C_a} = 1$ and $w_{T_J} = 10^{-4}$ in (14). All other weights are zero as the determinants in (14) do not depend on the corresponding variables. The observable set does in turn include the whole state space for these variables. The variables V and T_R are defined by their setpoints as we target to generate a DM that can be used for offset-free NMPC of a specific setpoint.

We analyze the four possible DMs, given that we add three disturbances to four equations: DM 1, with $p^d_{1,1} = 1$, $p^d_{2,1} = 1$, $p^d_{3,1} = 1$, DM 2, with $p^d_{1,1} = 1$, $p^d_{2,1} = 1$, $p^d_{3,2} = 1$, DM 3, with $p^d_{1,1} = 1$, $p^d_{2,2} = 1$, $p^d_{3,2} = 1$, and and DM 4, with $p^d_{1,2} = 1$, $p^d_{2,2} = 1$, $p^d_{3,2} = 1$.

For this case study, the determinant in (14b) is independent of the state, input, and disturbance realization.

The determinant only depends on the choice of the disturbance model, i.e., on the realization of the disturbance model parameters $\boldsymbol{p}^d$. This is a special case where the SIP (14) would actually be finite optimization problem, as the left hand side of the constraint (14b) is independent of $\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}$ and only depends on $\boldsymbol{p}^d$. I.e., in this case, it would not be required to solve an SIP but the finite optimization problem that (14) reduces to. Nevertheless, the SIP (14) can be solved, which is rather simple in this special case. Furthermore, the determinants in (14b) are zero for DM 1, DM 2, and DM 4, and (14b) is, hence, not satisfied for these disturbance models (realizations of $\boldsymbol{p}^d$). As a consequence, solving the SIP (14) is rather simple and directly furnishes DM 3 and DM 3 is the result of the generation approach in Fig. 2. Appendix D provides a further analysis of the nominal model explaining why DM 3 is the only suitable DM. Thus, DM 3 can be used for offset-free tracking for all values of C_a, T_J and the observable set is the entire defined state space.

Fig. 10 shows the results of the closed-loop simulation. We see that the NMPC with DM 3 leads to offset-free closed-loop behavior.

4.2.2 Not all CVs measured

Now, we consider the case that CVs are unmeasured. The measurements are $\boldsymbol{z}(t) = (V(t), C_a(t), T_J(t))$, i.e., we have T_R as an unmeasured CV, and we use the weights $w_{C_a} = 1$ and $w_{T_J} = 1$ in (14). The variables V and T_R are defined by their setpoints as we target to generate a DM that can be used for offset-free NMPC of a specific setpoint.

We analyze the four possible DMs, given that we only add up to disturbances to the differential equations: DM 1, with $p^d_{1,1} = 1$, $p^d_{2,1} = 1$, $p^d_{3,1} = 1$, DM 2, with $p^d_{1,1} = 1$, $p^d_{2,1} = 1$, $p^d_{3,2} = 1$, DM 3, with $p^d_{1,1} = 1$, $p^d_{2,2} = 1$, $p^d_{3,2} = 1$, and DM 4, with $p^d_{1,2} = 1$, $p^d_{2,2} = 1$, $p^d_{3,2} = 1$.

According to Theorem 1, we can guarantee offset-free NMPC in the case of unmeasured CVs if the steady state predictions of the augmented model are exact for the unmeasured CVs. This can be achieved by adding disturbances to those equation that correspond to the unmeasured CVs. Here, we realize this by adding $p^d_3 = 1$ as a constraint to (14), thereby adding one disturbance to the equation correpsonding to the unmeasured CV T_R. This constraint excludes DM 2 as a solution of the SIP (14), i.e., DM 2 does not add a disturbance to the equation corresponding to the unmeasured CV T_R. Applying the model generation approach in Fig. 2, DM 1 results. Again, solving (14) was not required for this case study, as the determinant with DM 1 is constant and nonzero, whereas the determinants with the other DMs are constant and zero. Thus, DM 1 can be used for offset-free tracking for all values of C_a and T_J as the observable set is the complete state space.

Using DM 1 results in closed-loop profiles which are almost identical with those shown in Fig. 10. Offset-free tracking can be achieved although the CVs are not measured. Looking at the DM 1, the plant model, and the applied plant-model mismatch, we see that DM 1 is not the correct DM, i.e., it does not model the actual plant-model mismatch. However, using DM 1, the CVs can be predicted exactly in the steady state; the disturbances are added to those equations corresponding to the unmeasured CVs (in this case T_R). Thus, we can conclude that in the case of unmeasured CVs, a DM is suitable that satisfies (12) and adds disturbances to those equations that correspond to the unmeasured CVs. With these disturbances, the unmeasured CVs can be predicted exactly in the steady state, and Assumption 2 is satisfied.

Note that Assumption 2 could also be satisfied by using an exact DM. However, this is usually not available in real applications. Furthermore, we wanted to demonstrate that exact predictions can also be achieved without using an exact DM.

4.3 Case Study III: Polymerization Reactor

In the last case study, we use the polymerization reactor model presented in [46] and we refer to [46, 47] for more details of the model. The model is provided in Appendix E.

The CVs are $\boldsymbol{y}(t) = (T(t), T_{\mathrm{j}}(t))$ and the MVs are the reactor feed flowrate and the cooling water flowrate $\boldsymbol{u}(t) = (F(t), F_{\mathrm{j}}(t))$ with the bounds $F(t) \in [0, 10]$ L/s and $F_{\mathrm{j}}(t) \in [0, 10]$ L/s. The seven measurements are $\boldsymbol{z}(t) = (C_{\mathrm{i}}(t), C_{\mathrm{m}}(t), C_{\mathrm{b}}(t), C_{\mathrm{r}}(t), C_{\mathrm{br}}(t), T(t), T_{\mathrm{j}}(t))$. In turn, we have to add seven disturbances to the nominal process model, due to $n_d = n_z$. We emulate a plant model mismatch through modifying the heat transfer area of the plant by $UA_{\Phi} = 2UA = 2 \cdot 1560$ J/(s K) from $t = 1$ h on.

The NMPC minimizes the following objective function in (8a)

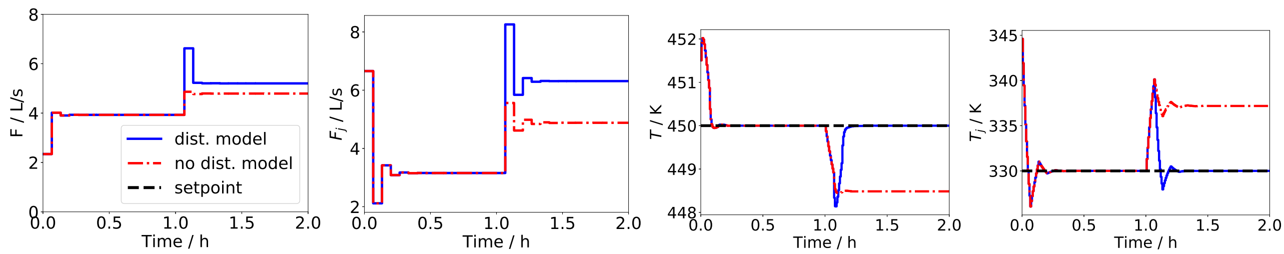
$$l = (T - 450 \text{ K})^2 + (T_j - 330 \text{ K})^2 \tag{19}$$

with a sampling time of 240 s and both control and prediction horizon set to 1 h. The process model is discretized using the same sampling time of 1 s to obtain a discrete time system to which we can apply the model generation approach.

Since the variables range over several orders of magnitudes, we scale the model so that the variables are in the range of $\mathcal{O}(1)$ and use the scaled model instead of the unscaled version.

We discretize the polymerization reactor model with the NMPC sampling time of 240 s. This leads to a discrete-time system to which we can apply the sufficient condition from Theorem 2, which in turn is used for the DM generation. In the model generation approach (Fig. 2), we use the bounds $C_{\mathrm{i}}^l, C_{\mathrm{i}}^u \in [0, 10] \cdot 10^{-4}$, $C_m^l, C_m^u \in [0.1, 0.2]$ mol/L, $C_b^l, C_b^u \in [0.4, 1.2]$ mol/L, $C_{\mathrm{br}}^l, C_{\mathrm{br}}^u \in [0, 2] \cdot 10^{-7}$ mol/L, $C_r^l, C_r^u \in [0, 2] \cdot 10^{-7}$ mol/L, $\mu_r^{0,l}, \mu_r^{0,u} \in [0, 2] \cdot 10^{-5}$ mol/L, $\mu_b^{0,l}, \mu_b^{0,u} \in [0, 2] \cdot 10^{-5}$ mol/L, $F^l, F^u \in [0, 10]$ L/s and $F_{\mathrm{j}}^l, F_{\mathrm{j}}^u \in [0, 10]$ L/s, and $T = 450$ K, $T_j = 330$ K, as these are the setpoints to be tracked. For (14), we use the weights $w_{C_m} = 0.1$, $w_{C_b} = 0.8$, $w_{C_{\mathrm{br}}} = 2 \cdot 10^{-7}$, $w_{C_r} = 2 \cdot 10^{-7}$, $w_{\mu_r^0} = 2 \cdot 10^{-5}$, and $w_{\mu_b^0} = 2 \cdot 10^{-5}$. All other weights are zero as the determinants in (14) do not depend on the corresponding variables. The observable set in turn includes the whole state space for these variables. The remaining states are defined by their setpoints as we target offset-free tracking at the specific setpoint. We constrain the squared determinant in (14) to be larger than $\epsilon = 10^{-4}$.

Fig. 11 shows the results of the closed-loop simulations with and without DM. We see that the NMPC



(a) (b) (c) (d)

Fig. 11: Results of closed-loop simulation using offset-free NMPC for polymerization reactor case study with and without DM. (a) Initiator flowrate (MV). (b) Reactor feed flowrate (MV). (c) Reactor temperature (CV). (d) Cooling water temperature (CV).

without DM leads to an offset in both tracked variables T and T_j due to the plant model mismatch exist. This demonstrates the need for offset-free NMPC.

We now focus on the generation of a suitable DM. Adding seven disturbances to the nine differential equations allows for 36 possible DMs ($\binom{9}{7} = 36$). Analyzing the 36 DMs, we see that 21 DMs result in a structural singularity for $\boldsymbol{J}$, which we exclude at the beginning of the model generation approach. Using enumeration, we solve (14) for the remaining DMs. While enumeration is feasible for medium-sized cases (as here), larger numbers of possible DMs would motivate the use of other techniques to treat the integers in (14), cf. [48].

Using the DM generation approach, we find five DMs that are suitable for offset-free NMPC in the entire defined state space, and three DMs which are suitable for offset-free NMPC only in a subset of the defined state space. Problem (14) is infeasible for six DMs, and these DMs are not suitable for offset-free NMPC. We do not find a solution of (14) for one DM, since the solver does not converge within 24 h. However, looking at the last iteration of these problems, we can guarantee that three of the four DMs admit an observable set which does not cover the entire defined state space. For the other DMs the solution of (14) for a specific disturbance model takes CPU times from few seconds up to few minutes. As we find five DMs, each of them can be chosen as a solution of (14). To demonstrate the suitability of the resulting DMs, from the five best suitable solutions we select the DM with $p^d_{i,1} = 1 \forall i \in \{1, ..., 5\}$, $p^d_{6,3} = 1$ and $p^d_{7,3} = 1$. All other parameter values of $\boldsymbol{p}^d$ are zero. I.e., one disturbance is added to every equation, except to the sixth and seventh equation. The corresponding observable set is the entire defined state space. The solution of (14) results directly after the first iteration within a few seconds.

Fig. 11 shows that NMPC with the DM achieves offset-free tracking and this case study, hence, demonstrates that our DM generation approach can be applied to medium scale case studies.

5 Conclusions

We extend the theory of offset-free NMPC and propose an approach for the generation of disturbance models for offset-free NMPC. Using relaxed assumptions, we show that offset-free tracking can be proven in the case of more measurements than CVs and the case of unmeasured but correctly predicted CVs at the steady state.

In the proposed model generation approach, the disturbance model results from the solution of a SIP. The disturbance model generation problem can be reformulated to a SIP, which is computationally easier to solve compared to a GSIP. The problem size of the SIP scales linearly with the number of variables of the augmented model. Solving the SIP requires to calculate a determinant depending on the augmented model for which we use symbolic expressions. However, calculating symbolic determinant expressions might be tedious for large-scale applications. Solving the SIPs using sequential deterministic global optimization allows us to embed numerical algorithms for determinant calculations in the optimization problems instead of just symbolic expressions. This favors the application of our approach to large-scale case studies.

We illustrate both the theoretical results and the disturbance model generation approach in numerical closed-loop case studies from small to medium scale. Thereby we demonstrate that by solving the disturbance model generation problem, a disturbance model is obtained which satisfies the sufficient observability criteria and can thus be used for offset-free tracking. The CPU times for the solution of the SIPs lies in the range of seconds to minutes, i.e., the CPU times can be regarded to be negligible, since the disturbance model generation problem is solved offline. This suggests that our approach can be applied to larger case studies. Further, the results illustrate the validity and practical relevance of the extension of the offset-free NMPC theory. Offset-free NMPC can be achieved if unmeasured CVs are predicted exactly at the steady state. This can be done by adding disturbances to those equations that correspond to the unmeasured CVs and adding further disturbances so that the observability criterion for the augmented model is satisfied. Hence, an exact disturbance model, i.e., a model that exactly models the disturbances, is not required for offset-free NMPC.

The application of the disturbance model generation approach to large scale case studies is left for future work. The size of the SIP of the disturbance model generation approach scales linearly with the number of variables of the augmented model and the number of parameters to parameterize the DM. It is difficult to state the limits of the disturbance model generation approach in terms of problem size of the SIPs in general. Although the tools used are generally not restricted to a specific problem size, solving SIPs is computationally expensive and might prohibit large-scale applications currently. Nevertheless, we present three case studies with up to nine differential equations and we could solve the disturbance model generation problems therein in just a few CPU minutes. Future works with larger case studies might exploit the specific structure of the optimization problems to facilitate the solution of the SIPs. Future work can apply the proposed approach to generate suitable disturbance models for reduced dynamic models, which introduce inherently a plant model mismatch, e.g. [6, 49]. On the theoretical side, the extension of offset-free NMPC to economic NMPC is an interesting future direction. Furthermore, future works can use this work to use NMPC models that can be applied for offset-free tracking in a certain range, e.g., to accelerate a flexible process operation with hierarchical control strategies [50]. Further, improving deterministic global optimization techniques is an ongoing path for future research for this problem class.

Acknowledgment

The authors gratefully acknowledge the financial support of the Kopernikus project SynErgie by the Federal Ministry of Education and Research (BMBF) and the project supervision by the project management organization Projektträger Jülich (PtJ). The authors thank Daniel Jungen and Jaromil Najman from AVT for their help with libALE and MAiNGO, respectively, and Jan Schulze for fruitful discussions.

Appendix

A Proof of Theorem 1

Proof By Assumption 5, a steady-state is reached. By Assumption 3 and Assumption 6, the observer is nominally error-free and satisfies at the steady-state

$$\hat{\boldsymbol{x}}_{\infty} = \boldsymbol{f}_{\text{aug}}(\hat{\boldsymbol{x}}_{\infty}, \hat{\boldsymbol{d}}_{\infty}, \boldsymbol{u}_{\infty})$$
$$\boldsymbol{z}_{\Phi,\infty} = \boldsymbol{g}^{z}_{\text{aug}}(\hat{\boldsymbol{x}}_{\infty}, \hat{\boldsymbol{d}}_{\infty})$$

where the inputs $\boldsymbol{u}_{\infty}$ result from the controller given by

$$\boldsymbol{u}_{\infty} = \boldsymbol{c}_{\text{aug}}(\hat{\boldsymbol{x}}_{\infty}, \hat{\boldsymbol{d}}_{\infty}, \boldsymbol{r}_{\infty}).$$

The control law exists by Assumption 7. The output variables at the steady-state are given by

$$\boldsymbol{y}_{\Phi,\infty} = \boldsymbol{g}^{y}_{\Phi}(\hat{\boldsymbol{x}}_{\Phi,\infty}).$$

By Assumption 4 we have

$$\boldsymbol{y}_{\text{aug},\infty} = \boldsymbol{g}^{y}_{\text{aug}}(\hat{\boldsymbol{x}}_{\infty}, \hat{\boldsymbol{d}}_{\infty}) = \boldsymbol{r}_{\infty}.$$

By Assumption 1 it holds that

$$y_{\Phi,\infty,i} = z_{\Phi,\infty,i} = z_{\text{aug},\infty,i} = y_{\text{aug},\infty,i} = r_{\infty,i}, \forall i \in \{1, ..., n_y\}.$$

By Assumption 2 it holds that

$$y_{\Phi,\infty,i} = z_{\Phi,\infty,i} = z_{\text{aug},\infty,i} = y_{\text{aug},\infty,i} = r_{\infty,i}, \forall i : y_{\infty,\text{i}} = z_{\infty,\text{i}}$$

and

$$y_{\Phi,\infty,i} = z_{\Phi,\infty,i} = z_{\text{aug},\infty,i} = y_{\text{aug},\infty,i} = r_{\infty,i}, \forall i : y_{\infty,\text{i}} \neq z_{\infty,\text{i}}$$

□

B Preprocessing to Exclude Disturbance Models leading to Structural Singularities

We propose a preprocessing strategy to excluding those realizations of the DM parameters $\boldsymbol{p}^d$ for which $\boldsymbol{J}$ in (13) is structurally singular. Recall that we assumed that $\boldsymbol{p}^d$ defines the structure of the DMs and hence $\boldsymbol{P}^d$ is a discrete set. While continuous sets would be possible, we do not see an advantage in defining the DM structure through continuous parameters. Furthermore, the algorithm provided here would need to be refined in case of a continuous set $\boldsymbol{P}^d$. The preprocessing uses the following optimization problem

$$\max_{\substack{\boldsymbol{p}^d \in \mathcal{P}^{d'} \\ \boldsymbol{x}\in\mathcal{X} \\ \boldsymbol{d}\in\mathcal{D} \\ \boldsymbol{u}\in\mathcal{U}}} \det\left(\boldsymbol{J}(\boldsymbol{x},\boldsymbol{u},\boldsymbol{d},\boldsymbol{p}^d)\right)^2 \tag{20a}$$

$$s.t. \qquad \boldsymbol{J}(\boldsymbol{x},\boldsymbol{u},\boldsymbol{d},\boldsymbol{p}^d) = \begin{pmatrix} \boldsymbol{F_x}(\boldsymbol{x},\boldsymbol{u},\boldsymbol{d},\boldsymbol{p}^d) - \boldsymbol{I} & \boldsymbol{F_d}(\boldsymbol{x},\boldsymbol{u},\boldsymbol{d},\boldsymbol{p}^d) \\ \boldsymbol{G_x}(\boldsymbol{x},\boldsymbol{u},\boldsymbol{d},\boldsymbol{p}^d) & \boldsymbol{G_d}(\boldsymbol{x},\boldsymbol{u},\boldsymbol{d},\boldsymbol{p}^d) \end{pmatrix} \tag{20b}$$

$$\boldsymbol{h}(\boldsymbol{x},\boldsymbol{u},\boldsymbol{d},\boldsymbol{p}^d) = \boldsymbol{0} \tag{20c}$$

$$\boldsymbol{g}(\boldsymbol{x},\boldsymbol{u},\boldsymbol{d},\boldsymbol{p}^d) \leqslant \boldsymbol{0}. \tag{20d}$$

The preprocessing strategy is presented in Fig. 12. The approach solves (20), where in each iteration an element is removed from the set $\mathcal{P}^{d'}$ until the set $\mathcal{P}^{d'}$ contains only those parameter realizations for which $\boldsymbol{J}$ in (13) is structurally singular. Meanwhile, the set $\mathcal{P}^{d''}$ accumulates all other parameter realizations, i.e., parameter values for which $\boldsymbol{J}$ in (13) is structurally non-singular. Finally, we update $\mathcal{P}^d$ with $\mathcal{P}^{d''}$, so that $\mathcal{P}^d$ contains only those parameter values for which $\boldsymbol{J}$ in (13) is structurally not singular. If the resulting $\mathcal{P}^{d''}$ is an empty set, then no suitable disturbance model exists with the given disturbance model parameterization. After the preprocessing, the resulting set $\mathcal{P}^d$ is used in (13), so that all parameter values for $\boldsymbol{p}^d$ are excluded for which $\boldsymbol{J}$ in (13) is structurally singular. If the resulting set $\mathcal{P}^d$ is empty, then there is no suitable disturbance model with the given parameterization, that satisfies the sufficient observability condition.

We can use the constraints (20c) and (20d) with the functions $\boldsymbol{h} : \mathcal{X} \times \mathcal{X} \times \mathcal{U} \times \mathcal{U} \times \mathcal{D} \times \mathcal{D} \times \mathcal{P}^d \to \mathbb{R}^{n_h}$ and $\boldsymbol{g} : \mathcal{X} \times \mathcal{X} \times \mathcal{U} \times \mathcal{U} \times \mathcal{D} \times \mathcal{D} \times \mathcal{P}^d \to \mathbb{R}^{n_g}$, e.g., if we aim at generating a model used to track one specific point, the constraint $\boldsymbol{g}^y_{\text{aug}}(\boldsymbol{x},\boldsymbol{d}) = \boldsymbol{r}_\infty$ can directly be added to the problem. In this case, we only exclude those parameter realizations leading to structural singularities under the given constraint, e.g., at one specific point in the state space.

Furthermore, the constraints can be used to add disturbances to specific equations, e.g., to those equations which correspond to unmeasured CVs, thereby enabling exact steady state predictions for those variables as required by Assumption 2 for offset-free NMPC, cf. Theorem 1.

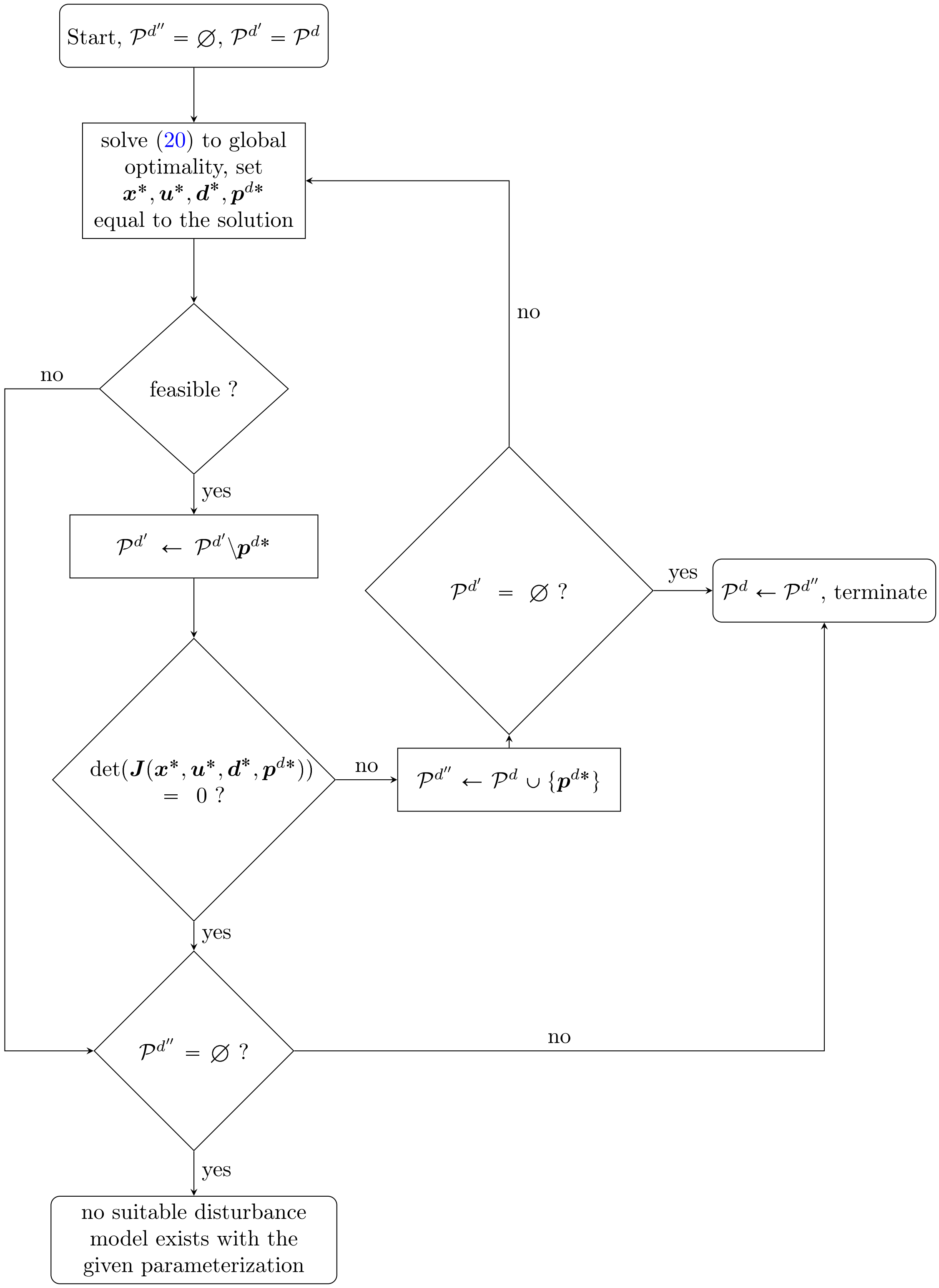


Fig. 12: Preprocessing for model generation approach to exclude disturbance model parameter values for which $\boldsymbol{J}$ in (13) is structurally singular.

C Process Model for Case Study I

We use the chemical reactor example from [8, 2, 5]. The process model is given by the following differential equations:

$$\left.\frac{\mathrm{d}c}{\mathrm{d}t}\right|_t = \frac{F_0(t)c_0 - F(t)c(t)}{\pi r^2 h(t)} - k_0 \exp\left(-\frac{E}{RT(t)}\right) c(t) \tag{21a}$$

$$\left.\frac{\mathrm{d}T}{\mathrm{d}t}\right|_t = \frac{F_0(t)T_0 - F(t)T(t)}{\pi r^2 h(t)} - \frac{\Delta H}{\rho C_p} k_0 \exp\left(-\frac{E}{RT(t)}\right) c(t) + \frac{2Uh(t)}{r\rho C_p}\left(T_c(t) - T(t)\right) \tag{21b}$$

$$\left.\frac{\mathrm{d}h}{\mathrm{d}t}\right|_t = \frac{F_0(t) - F(t)}{\pi r^2}. \tag{21c}$$

$c(t) \in \mathbb{R}$ is the concentration in the reactor, $T(t) \in \mathbb{R}$ the reactor temperature, $h(t) \in \mathbb{R}$ the reactor fluid height, $F_0(t) \in \mathbb{R}$ the reactor feed flow rate, $F(t) \in \mathbb{R}$ the reactor outlet flowrate, $c_0 \in \mathbb{R}$ the feed concentration, $T_0 \in \mathbb{R}$ the feed temperature, $T_c \in \mathbb{R}$ the cooling fluid temperature, $r \in \mathbb{R}$ the reactor radius, $\rho \in \mathbb{R}$ the reactor fluid density, $C_p \in \mathbb{R}$ the reactor fluid heat capacity, $\Delta H \in \mathbb{R}$ the reaction enthalpy, and $k_0 \in \mathbb{R}$ the reaction rate constant. Table 1 gives the values for the parameters in (21) from [8, 2].

Table 1: Parameter values for chemical reactor case study.

parameter	**value**
F_0	0.1 m^3/min
T_0	350 K
c_0	1 kmol/m^3
k_0	7.2 $\cdot 10^{10}$ min^{-1}
r	0.219 m
E/R	8750 K
U	54.94 kJ/min m^2 K
ρ	1000 kg/m^3
C_p	0.239 kJ/kg K
ΔH	-5 $\cdot 10^{-4}$ kJ/kmol

D Process Model for Case Study II

We use the chemical reactor model from Santos et al. [45]:

$$\left.\frac{\mathrm{d}V}{\mathrm{d}t}\right|_t = F_1 + F_2 - F_3(t) \tag{22a}$$

$$\left.\frac{\mathrm{d}C_a}{\mathrm{d}t}\right|_t = \frac{F_1}{V(t)}\big(C_{a,1} - C_a(t)\big) + \frac{F_2}{V(t)}\big(C_{a,2} - C_a(t)\big) - k_0 \exp\left(-\frac{E}{RT_R(t)}\right) \tag{22b}$$

$$\left.\frac{\mathrm{d}T_R}{\mathrm{d}t}\right|_t = \frac{1}{\beta_R(t)}\big(-Q_R(t) + Q_G(t)\big) \tag{22c}$$

$$\left.\frac{\mathrm{d}T_J}{\mathrm{d}t}\right|_t = \frac{1}{\beta_J(t)}\Big(\rho_J C_{p,J} F_J(t)\big(T_{J,0} - T_J(t)\big) + UA(t)\big(T_R(t) - T_J(t)\big)\Big) \tag{22d}$$

$$0 = \beta_R(t) - \rho C_p V(t) - \alpha_R \tag{22e}$$

$$0 = \beta_J(t) - \rho_J C_{p,J} V_{\mathrm{j}} - \alpha_J \tag{22f}$$

$$0 = Q_R(t) + \rho C_p F_1\big(T_1 - T_R(t)\big) + \rho C_p F_2\big(T_2 - T_R(t)\big) - UA(t)\big(T_R(t) - T_J(t)\big) \tag{22g}$$

$$0 = Q_G(t) + \Delta H_R V(t) k_0 \exp\left(-\frac{E}{RT_R(t)}\right) \tag{22h}$$

$$0 = A - \pi r\big(r + 2h(t)\big) \tag{22i}$$

$$0 = V(t) - V_0 - \pi r^2 h(t), \tag{22j}$$

where $V(t) \in \mathbb{R}$ is the reactor volume, $C_a(t) \in \mathbb{R}$ the molar fraction of the component a, $T_R(t) \in \mathbb{R}$ the reactor temperature, $T_J(t) \in \mathbb{R}$ the cooling jacket temperature, $F_1 \in \mathbb{R}$ and $F_2 \in \mathbb{R}$ the feed flowrates, $F_3(t) \in \mathbb{R}$ the leaving stream flowrate, $F_J(t) \in \mathbb{R}$ the cooling water flowrate, $C_{A,1} \in \mathbb{R}$ and $C_{A,2} \in \mathbb{R}$ the mole fractions of component a in the feed streams 1 and 2, respectively, $k_0 \in \mathbb{R}$ the reaction constant, $E \in \mathbb{R}$ the reaction activation energy, $R \in \mathbb{R}$ the ideal gas constant, $Q_R(t) \in \mathbb{R}$ the heat stream due to heat exchange with the cooling jacket and due to convection, $Q_G(t) \in \mathbb{R}$ the heat stream released from the reaction, $\Delta H_R \in \mathbb{R}$ the reaction enthalpy, $A \in \mathbb{R}$ the heat transfer surface, $h(t) \in \mathbb{R}$ the reactor height. The parameter values are given in Tab. 2.

As described in Section 4.2.2, only DM 3 satisfies the sufficient observability condition (14b). This can also be seen by analyzing the matrix $\boldsymbol{J}$ of resulting discrete time system of the model (22). For this, we substitute the algebraic equations (22e)-(22j) into the differential equations (22a)-(22d), derive the discrete time form of the resulting system and analyze the structure of the matrix $\boldsymbol{J}$ for this system. We see that equation (22b) is the only equation depending an C_a. Thus, no disturbance needs to be added to this equation to make the matrix $\boldsymbol{J}$ in (14c) non-singular. On the other hand, adding one disturbance to equation (22b) would make the matrix $\boldsymbol{J}$ in (14c) singular, as this disturbance could not be added to the other two equations (given the disturbance model structure in (17) and (18)), which would, however, be required to guarantee that $\boldsymbol{J}$ is non-singular. Thus, the only possible disturbance model left is DM 3.

A physical reasons behind the peculiarity that only DM 3 is a suitable disturbance model is that the disturbances are added to all MVs. Thus, the disturbances can propagate through the model so that the the unmeasured CV C_a can be tracked without offset. In other words, by augmenting the model for the MVs, we can obtain offset-free tracking of the unmeasured CV, i.e., we trace the plant-model mismatch back to the MVs. This restricts the selection of a possible disturbance model to just DM 3.

Table 2: Parameter values for chemical reactor case study II.

parameter	**value**
F_1	0 mol/s
$C_{A,1}$	$2.0 \cdot 10^3$ mol/m^3
$C_{A,2}$	$10 \cdot 10^3$ mol/m^3
C_p	4184 J/kg K
$C_{p,J}$	4184 J/kg K
E_a/R	10080 K
k_0	$6.20 \cdot 10^{14}$ mol/m^3 s
α_R	0 J/K
α_J	$7 \cdot 10^5$ J/K
ρ	1000 kg/m^3
ρ_j	1000 kg/m^3
T_1	294.15 K
T_2	294.15 K
U	900 W/m^2 K
r	0.232 m
V_0	$4.2 \cdot 10^{-3}$ m^3
ΔH_R	-33488 J/mol
V_j	0.014 m^3
$T_{J,0}$	299.15 K

E Process Model for Case Study III

The polymerization reactor model is presented in [46] and we refer to [46, 47] for more details of the model. The model equation are as follows:

$$\left.\frac{\mathrm{d}C_\mathrm{i}}{\mathrm{d}t}\right|_t = \frac{F_\mathrm{i}(t)C_\mathrm{i}^0 - (F + F_\mathrm{i}(t))C_\mathrm{i}(t)}{V} - k_d(t)C_\mathrm{i}(t) \tag{23a}$$

$$\left.\frac{\mathrm{d}C_\mathrm{m}}{\mathrm{d}t}\right|_t = \frac{FC_\mathrm{m}^0 - (F + F_i(t))C_\mathrm{m}(t)}{V} - k_p(t)C_\mathrm{m}(t)\big(\mu_\mathrm{r}^0(t) + \mu_\mathrm{b}^0(t)\big) \tag{23b}$$

$$\left.\frac{\mathrm{d}C_b}{\mathrm{d}t}\right|_t = \frac{FC_\mathrm{b}^0 - (F + F_\mathrm{i}(t))C_b(t)}{V} - C_b(t)\big(k_{i2}(t)Cr(t) + k_{fs}(t)\mu_\mathrm{r}^0(t) + k_{fb}(t)\mu_\mathrm{b}^0(t)\big) \tag{23c}$$

$$\left.\frac{\mathrm{d}C_\mathrm{r}}{\mathrm{d}t}\right|_t = 2f^* k_d(t)C_\mathrm{i}(t) - C_\mathrm{r}(t)\big(k_{i1}(t)C_\mathrm{m}(t) + k_{i2}(t)C_b(t)\big) \tag{23d}$$

$$\left.\frac{\mathrm{d}C_\mathrm{br}}{\mathrm{d}t}\right|_t = C_b(t)\Big(k_{i2}(t)C_\mathrm{r}(t) + k_{fb}(t)\big(\mu_\mathrm{r}^0(t) + \mu_\mathrm{b}^0(t)\big)\Big) - C_\mathrm{br}(t)\Big(k_{i3}(t)C_\mathrm{m}(t) + k_t(t)\big(\mu_\mathrm{r}^0(t) + \mu_\mathrm{b}^0(t) + C_\mathrm{br}(t)\big)\Big) \tag{23e}$$

$$\left.\frac{\mathrm{d}\mu_\mathrm{r}^0}{\mathrm{d}t}\right|_t = 2k_{i0}(t)C_\mathrm{m}(t)^3 + k_{i1}(t)C_\mathrm{r}(t)C_\mathrm{m}(t) + C_\mathrm{m}(t)k_{fs}(t)(\mu_\mathrm{r}^0(t) + \mu_\mathrm{b}^0(t)) - \big(k_p(t)C_\mathrm{m}(t) + k_t(t)(\mu_\mathrm{r}^0(t) + \mu_\mathrm{b}^0(t) + C_\mathrm{br}(t)) + k_{fs}(t)C_\mathrm{m}(t) + k_{fb}(t)C_b(t)\big)\mu_\mathrm{r}^0(t) + k_p(t)C_\mathrm{m}(t)\mu_\mathrm{r}^0(t) \tag{23f}$$

$$\left.\frac{\mathrm{d}\mu_\mathrm{b}^0}{\mathrm{d}t}\right|_t = k_{i3}(t)C_\mathrm{br}(t)C_\mathrm{m}(t) - \Big(k_p(t)C_\mathrm{m}(t) + k_t(t)\big(\mu_\mathrm{r}^0(t) + \mu_\mathrm{b}^0(t) + C_\mathrm{br}(t)\big) + k_{fs}(t)C_\mathrm{m}(t) + k_{fb}(t)C_b(t)\Big)\mu_\mathrm{b}^0(t) + k_p(t)C_\mathrm{m}(t)\mu_\mathrm{b}^0(t) \tag{23g}$$

$$\left.\frac{\mathrm{d}T}{\mathrm{d}t}\right|_t = \frac{(F + F_\mathrm{i}(t))(T^0 - T(t))}{V} - \frac{\Delta H k_p C_\mathrm{m}(t)(\mu_\mathrm{r}^0(t) + \mu_\mathrm{b}^0(t))}{\rho_\mathrm{s} C_{p,s}} - \frac{UA(T(t) - T_\mathrm{j}(t))}{\rho_\mathrm{s} C_{p,s} V} \tag{23h}$$

$$\left.\frac{\mathrm{d}T_\mathrm{j}}{\mathrm{d}t}\right|_t = \frac{F_\mathrm{j}(t)(T_\mathrm{j}^0 - T_\mathrm{j}(t))}{V_\mathrm{j}} + \frac{UA(T(t) - T_\mathrm{j})}{\rho_\mathrm{j} C_{p,j} V_\mathrm{j}}. \tag{23i}$$

$C_\mathrm{i}(t) \in \mathbb{R}$ is the initiator concentration, $C_\mathrm{m}(t) \in \mathbb{R}$ the monomer concentration, $C_\mathrm{b}(t) \in \mathbb{R}$ the butadiene concentration, $C_\mathrm{r}(t) \in \mathbb{R}$ the radical concentration, $C_\mathrm{br}(t) \in \mathbb{R}$ the branched radical concentration, $\mu_\mathrm{r}^0(t) \in \mathbb{R}$ the zeroth moment of dead radicals, $\mu_\mathrm{b}^0(t) \in \mathbb{R}$ the zeroth moment of dead butadiene, $T(t) \in \mathbb{R}$ the reactor temperature, $T_\mathrm{j}(t) \in \mathbb{R}$ the cooling water temperature, $F_\mathrm{i} \in \mathbb{R}$ the initiator feed flowrate, $F(t) \in \mathbb{R}$ the feed flowrate, $F_\mathrm{j}(t) \in \mathbb{R}$ the cooling water flowrate, $C_\mathrm{i}^0 \in \mathbb{R}$ the initiator feed concentration, $C_\mathrm{m}^0 \in \mathbb{R}$ the monomer feed concentration, $C_b^0 \in \mathbb{R}$ the butadiene feed concentration, $f^* \in \mathbb{R}$ the initiator efficiency, $\Delta H \in \mathbb{R}$ the reactor enthalpy, $V \in \mathbb{R}$ the reactor volume, $\rho_\mathrm{s} \in \mathbb{R}$ the reactor density, $C_{p,s} \in \mathbb{R}$ the reactor heat capacity, $U \in \mathbb{R}$ the heat transfer coefficient, $A \in \mathbb{R}$ the heat transfer area, $T^0 \in \mathbb{R}$ the feed temperature, $T_\mathrm{j}^0 \in \mathbb{R}$ the cooling water feed temperature, $V \in \mathbb{R}$ the cooling jacket volume, $\rho_j \in \mathbb{R}$ the cooling water density, $C_{p,j} \in \mathbb{R}$ the cooling water heat capacity. The reaction rates as presented in [47] are given by

$$c_1(t) = 2.57 - 0.00505T(t) \tag{23j}$$

$$c_2(t) = 9.56 - 0.0176T(t) \tag{23k}$$

$$c_3(t) = -3.03 + 0.00785T(t) \tag{23l}$$

$$k_d(t) = 9.1 \cdot 10^{13} \exp\big(-29508 \text{ (cal/mol)}/[R_g T(t)]\big) \tag{23m}$$

$$k_{i0}(t) = 1.1 \cdot 10^{5} \exp\big(-27340 \text{ (cal/mol)}/[R_g T(t)]\big) \tag{23n}$$

$$k_{i1}(t) = 1.0 \cdot 10^{7} \exp\big(-7067 \text{ (cal/mol)}/[R_g T(t)]\big) \tag{23o}$$

$$k_{i2}(t) = 2 \cdot 10^{6} \exp\big(-7067 \text{ (cal/mol)}/[R_g T(t)]\big) \tag{23p}$$

$$k_{i3}(t) = 1 \cdot 10^{7} \exp\big(-7067 \text{ (cal/mol)}/[R_g T(t)]\big) \tag{23q}$$

$$k_{p}(t) = 1 \cdot 10^{7} \exp\big(-7067 \text{ (cal/mol)}/[R_g T(t)]\big) \tag{23r}$$

$$k_{fs}(t) = 6.6 \cdot 10^{7} \exp\big(-14400 \text{ (cal/mol)}/[R_g T(t)]\big) \tag{23s}$$

$$k_{fb}(t) = 2.3 \cdot 10^{9} \exp\big(-18000 \text{ (cal/mol)}/[R_g T(t)]\big) \tag{23t}$$

$$k_t(t) = 1.7 \cdot 10^{9} \exp\Big(-843 \text{ (cal/mol)}/\big[T(t) - 2\big(c_1(t)x_s(t) + c_2(t)x_s(t)^2 + c_3(t)x_s(t)^3\big)\big]\Big) \tag{23u}$$

$$x_s(t) = (C_m^0 - C_{\mathrm{m}}(t))/C_m^0, \tag{23v}$$

where $R_g \in \mathbb{R}$ is the ideal gas constant and $x_s(t) \in \mathbb{R}$ the monomer conversion rate. Table 3 give the parameter values for (23).

Table 3: Parameter values for polymerization reactor case study.

parameter	**value**
F_{i}	1 L/s
C_{i}^0	0.981 mol/L
C_{m}^0	8.63 mol/L
$C_{p,j}$	1647.72 J/(kg K)
$C_{p,s}$	4054.7 J/(kg K)
f^*	0.57
ρ_j	1 kg/L
ρ_s	0.915 kg/L
UA	1560 J/(s K)
V	9450 L
V_{j}	2000 L
R_g	1.987 cal/(mol K)
ΔH	-69919.56 J/mol
C_{m}^0	1.05 mol/L
T_{j}^0	294 K
T^0	333 K

References

[1] S. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, vol. 11, no. 7, pp. 733–764, 2003.

[2] J. B. Rawlings, D. Q. Mayne, and M. M. Diehl, *Model Predictive Control: Theory, Computation, and Design, 2nd Edition*. Nob Hill Publishing, LLC, 2017.

[3] K. R. Muske and T. A. Badgwell, "Disturbance modeling for offset-free linear model predictive control," *Journal of Process Control*, vol. 12, no. 5, pp. 617–632, 2002.

[4] M. Morari and U. Maeder, "Nonlinear offset-free model predictive control," *Automatica*, vol. 48, no. 9, pp. 2059–2067, 2012.

[5] G. Pannocchia, M. Gabiccini, and A. Artoni, "Offset-free MPC explained: novelties, subtleties, and applications," *IFAC-PapersOnLine*, vol. 48, no. 23, pp. 342–351, 2015.

[6] A. Caspari, C. Offermanns, A.-M. Ecker, M. Pottmann, G. Zapp, A. Mhamdi, and A. Mitsos, "A wave propagation approach for reduced dynamic modeling of distillation columns: Optimization and control," *Journal of Process Control*, vol. 91, pp. 12 – 24, 2020.

[7] J. C. Schulze, A. Caspari, C. Offermanns, A. Mhamdi, and A. Mitsos, "Nonlinear model predictive control of ultra-high-purity air separation units using transient wave propagation model," *Computers & Chemical Engineering*, p. 107163, 2020.

[8] G. Pannocchia and J. B. Rawlings, "Disturbance models for offset-free model-predictive control," *AIChE Journal*, vol. 49, no. 2, pp. 426–437, 2003.

[9] U. Maeder and M. Morari, "Offset-free reference tracking for predictive controllers," in *2007 46th IEEE Conference on Decision and Control*, IEEE, 2007.

[10] U. Maeder, F. Borrelli, and M. Morari, "Linear offset-free model predictive control," *Automatica*, vol. 45, no. 10, pp. 2214–2222, 2009.

[11] U. Maeder and M. Morari, "Offset-free reference tracking with model predictive control," *Automatica*, vol. 46, no. 9, pp. 1469–1476, 2010.

[12] P. Tatjewski, "Disturbance modeling and state estimation for offset-free predictive control with state-space process models," *International Journal of Applied Mathematics and Computer Science*, vol. 24, no. 2, pp. 313–323, 2014.

[13] R. Huang, L. T. Biegler, and S. C. Patwardhan, "Fast offset-free nonlinear model predictive control based on moving horizon estimation," *Industrial & Engineering Chemistry Research*, vol. 49, no. 17, pp. 7882–7890, 2010.

[14] G. Pannocchia, "An economic MPC formulation with offset-free asymptotic performance," *IFAC-PapersOnLine*, vol. 51, no. 18, pp. 393–398, 2018.

[15] G. Pannocchia and A. Bemporad, "Combined design of disturbance model and observer for offset-free model predictive control," *IEEE Transactions on Automatic Control*, vol. 52, no. 6, pp. 1048–1053, 2007.

[16] G. Pannocchia, "Robust disturbance modeling for model predictive control with application to multivariable ill-conditioned processes," *Journal of Process Control*, vol. 13, no. 8, pp. 693–701, 2003.

[17] M. A. Al-Arfaj and W. L. Luyben, "Design and control of an olefin metathesis reactive distillation column," *Chemical Engineering Science*, vol. 57, no. 5, pp. 715–733, 2002.

[18] M. A. Al-Arfaj and W. L. Luyben, "Control of ethylene glycol reactive distillation column," *AIChE Journal*, vol. 48, no. 4, pp. 905–908, 2002.

[19] S. Skogestad, "The dos and don'ts of distillation column control," *Chemical Engineering Research and Design*, vol. 85, no. 1, pp. 13–23, 2007.

[20] M. López and G. Still, "Semi-infinite programming," *European Journal of Operational Research*, vol. 180, no. 2, pp. 491–518, 2007.

[21] A. R. G. Mukkula and R. Paulen, "Model-based design of optimal experiments for nonlinear systems in the context of guaranteed parameter estimation," *Computers & Chemical Engineering*, vol. 99, pp. 198–213, 2017.

[22] O. Walz, H. Djelassi, A. Caspari, and A. Mitsos, "Bounded-error optimal experimental design via global solution of constrained min–max program," *Computers & Chemical Engineering*, vol. 111, pp. 92–101, 2018.

[23] F. Zhang, ed., *The Schur Complement and Its Applications.* Springer-Verlag, 2005.

[24] L. T. Biegler, *Nonlinear Programming: Concepts, Algorithms, and Applications to Chemical Processes.* SIAM, 2010.

[25] A. U. Raghunathan and L. T. Biegler, "Mathematical programs with equilibrium constraints (MPECs) in process engineering," *Computers & Chemical Engineering*, vol. 27, no. 10, pp. 1381–1392, 2003.

[26] A. Caspari, L. Lüken, P. Schäfer, Y. Vaupel, A. Mhamdi, L. T. Biegler, and A. Mitsos, "Dynamic optimization with complementarity constraints: Smoothing for direct shooting," *Computers & Chemical Engineering*, vol. 139, p. 106891, 2020.

[27] P. Lemonidis, *Global Optimization Algorithms for Semi-Infinite and Generalized Semi-Infinite Programs.* PhD thesis, Department of Chemical Engineering, Massachusetts Institute of Technology, 2008.

[28] H. Djelassi and A. Mitsos, "A hybrid discretization algorithm with guaranteed feasibility for the global solution of semi-infinite programs," *Journal of Global Optimization*, vol. 68, no. 2, pp. 227–253, 2017.

[29] R. Hettich and K. O. Kortanek, "Semi-infinite programming: Theory, methods, and applications," *SIAM Review*, vol. 35, no. 3, pp. 380–429, 1993.

[30] R. Reemtsen and S. Görner, "Numerical methods for semi-infinite programming: A survey," in *Nonconvex Optimization and Its Applications*, pp. 195–275, Springer US, 1998.

[31] J. W. Blankenship and J. E. Falk, "Infinitely constrained optimization problems," *Journal of Optimization Theory and Applications*, vol. 19, no. 2, pp. 261–281, 1976.

[32] L. Ljung, "System identification," in *Signal Analysis and Prediction* (J. J. Benedetto, A. Procházka, J. Uhlíř, P. W. J. Rayner, and N. G. Kingsbury, eds.), Applied and Numerical Harmonic Analysis, pp. 163–173, Boston, MA: Birkhäuser Boston, 1998.

[33] H. Djelassi and A. Mitsos, "libALE – A library for algebraic-logical expression trees," *Process Systems Engineering (AVT.SVT), RWTH Aachen University, 52074 Aachen, Germany. https://git.rwth-aachen.de/avt.svt/public/libale.git*, 2019.

[34] D. Bongartz, J. Najman, S. Sass, and A. Mitsos, "MAiNGO – mccormick-based algorithm for mixed-integer nonlinear global optimization," tech. rep., Process Systems Engineering (AVT.SVT), RWTH Aachen University, 52074 Aachen, Germany, 2018.

[35] B. C. Chachuat, "MC++: Toolkit for construction, manipulation and bounding of factorable functions." https://omega-icl.github.io/mcpp/, 2006.

[36] D. Lay, *Linear algebra and its applications*. Boston: Pearson, 2016.

[37] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes*. Cambridge University Pr., 2007.

[38] A. Mitsos, B. Chachuat, and P. I. Barton, "Mccormick-based relaxations of algorithms," *SIAM Journal on Optimization*, vol. 20, no. 2, pp. 573–601, 2009.

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

[40] R. G. Brusch and R. H. Schapelle, "Solution of highly constrained optimal control problems using nonlinear programing," *AIAA Journal*, vol. 11, no. 2, pp. 135–136, 1973.

[41] A. Caspari, J. M. M. Faust, F. Jung, C. Kappatou, S. Sass, Y. Vaupel, R. Hannesmann-Tamás, A. Mhamdi, and A. Mitsos, "Dyos - a framework for optimization of large-scale differential algebraic equation systems," *Computer-Aided Chemical Engineering*, vol. 46, 2019.

[42] P. E. Gill, W. Murray, and M. A. Saunders, "SNOPT: An SQP algorithm for large-scale constrained optimization," *SIAM Rev.*, vol. 47, no. 1, pp. 99–131, 2005.

[43] R. Hannemann, W. Marquardt, U. Naumann, and B. Gendler, "Discrete first- and second-order adjoints and automatic differentiation for the sensitivity analysis of dynamic models," *Procedia Comput. Sci.*, vol. 1, no. 1, pp. 297–305, 2010.

[44] Analytic Sciences Corporation, *Applied Optimal Estimation.* MIT Press Ltd, 1974.

[45] L. O. Santos, P. A. Afonso, J. A. Castro, N. M. Oliveira, and L. T. Biegler, "On-line implementation of nonlinear MPC: an experimental case study," *Control Engineering Practice*, vol. 9, no. 8, pp. 847–857, 2001.

[46] A. Flores-Tlacuahuac, L. T. Biegler, and E. Saldívar-Guerra, "Dynamic optimization of HIPS open-loop unstable polymerization reactors," *Industrial & Engineering Chemistry Research*, vol. 44, no. 8, pp. 2659–2674, 2005.

[47] J. C. Verazaluce-García, A. Flores-Tlacuahuac, and E. Saldívar-Guerra, "Steady-state nonlinear bifurcation analysis of a high-impact polystyrene continuous stirred tank reactor," *Industrial & Engineering Chemistry Research*, vol. 39, no. 6, pp. 1972–1979, 2000.

[48] C. Floudas, *Nonlinear and mixed-integer optimization : fundamentals and applications.* New York: Oxford University Press, 1995.

[49] P. Schäfer, A. Caspari, A. M. Schweidtmann, Y. Vaupel, A. Mhamdi, and A. Mitsos, "The potential of hybrid mechanistic/data-driven approaches for reduced dynamic modeling: Application to distillation columns," *Chemie Ingenieur Technik*, 2020.

[50] A. Caspari, C. Tsay, A. Mhamdi, M. Baldea, and A. Mitsos, "The integration of scheduling and control: Top-down vs. bottom-up," *Journal of Process Control*, vol. 91, pp. 50 – 62, 2020.