Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 52-1 (2019) 667–672

Economic stochastic nonlinear model predictive control of a semi-batch polymerization reaction

Eric Bradford * **Marcus Reble** ** **Ala Bouaswaig** ** **Lars Imsland** *

* *Department of Engineering Cybernetics, Norwegian University of Science and Technology, Trondheim, Norway (e-mail: eric.bradford,lars.imsland@ntnu.no).*

** *BASF SE, 67056 Ludwigshafen, Germany (e-mail: marcus.reble,ala.bouaswaig@basf.com).*

Abstract: Batch processes are ubiquitous in the chemical industry and difficult to control, such that nonlinear model predictive control is one of the few promising control techniques. Many chemical process models however are affected by various uncertainties, which can lower the performance and lead to constraint violations. In this paper we propose a framework for output feedback stochastic nonlinear model predictive control (SNMPC) to consider the uncertainties explicitly, which are assumed to follow known probability distributions. Polynomial chaos expansions are employed both for the formulation of the SNMPC algorithm and a nonlinear filter for the estimation of the uncertain parameters online given noisy measurements. The effectiveness of the proposed SNMPC scheme was verified on an extensive case study involving the production of the polymer polypropylene glycol in a semi-batch reactor.

Keywords: Model-based control, Uncertain dynamic systems, Stochastic parameters, Nonlinear filters, Chemical process control, Output feedback, Stochastic programming, Polynomial chaos expansions

1. INTRODUCTION

Batch processes play a vital role for the production of high value products, which make-up an important portion of the revenue in the chemical industry. Batch processes are used due to their inherent flexibility to produce multiple products and deal with variations in feedstock, product specifications, and market demand (Nagy and Braatz, 2003). Most batch process control methods have focused on tracking recipes with empirical models as in iterative learning control (Lee and Lee, 2007). Due to the high competitiveness however, there is an increasing acceptance for advanced control methods.

Model predictive control (MPC) was developed in the late seventies and refers to a class of advanced control methods that make explicit use of a dynamic model. Most applications of MPC have been limited to linear MPC (LMPC), for which plants are assumed to be weakly nonlinear. Batch processes are often highly nonlinear and operated at unsteady state. In this case nonlinear MPC (NMPC) is the method of choice (Nagy et al., 2007). In addition, NMPC can be used directly to minimize economic costs, which is becoming increasingly of interest, known as economic MPC (EMPC) (Rawlings and Amrit, 2009). NMPC applications based on first principle models are becoming more popular due to advent of improved optimization algorithms (Biegler, 2010).

Most dynamic models involve significant uncertainties, which need to be taken into account explicitly to avoid infeasibilities and performance deterioration, including parametric uncertainties, external disturbances and state estimation errors. In EMPC the system is often driven close to the constraints, however relatively little attention has been paid to handling uncertainty in EMPC with few exceptions reported (Lucia et al., 2014).

Uncertainty in MPC can be either assumed to lie in a bounded set or to be stochastic with known probability density functions (pdf), which leads to robust or stochastic MPC formulations respectively. Robust NMPC methods include min-max NMPC and tube-based NMPC, which have been both extended to handle economic objectives in Bayer et al. (2016) and in Bayer et al. (2014) respectively. Alternatively, several algorithms for stochastic NMPC (SNMPC) have been proposed. A simple solution to SNMPC is given by the successive linearization of the nonlinear system and the subsequent application of stochastic LMPC approaches, such as stochastic tube-based MPC (Cannon et al., 2009). Unscented transformations have been used in Bradford and Imsland (2017b, 2018a) to propagate uncertainties in a SNMPC application. Both linearization and unscented transformation are computationally cheap, but are only applicable to moderately nonlinear systems. A sampling average approach was used in Bradford and Imsland (2017a) with variance reduction to reduce the required number of samples, while in Maciejowski et al. (2007) Markov Chain Monte Carlo is applied. Both procedures can approximate the SNMPC problem arbitrarily well with increasing sample size but the required number of samples quickly becomes intractable. If it is assumed that the stochastic uncertainties can only take a discrete set of realizations, then multi-stage NMPC formulations have been proposed (Lucia et al., 2013; Patrinos et al., 2014). In particular, this methodology has been extensively applied to EMPC problems (Lucia et al., 2014; Sopasakis et al., 2017). While nearly all SNMPC algorithms consider full state feedback, there are several exceptions. The Unscented transformation SNMPC used in Bradford and Imsland (2017b, 2018a) are based on feedback from the Unscented Kalman filter, in Homer and Mhaskar (2018) a Lyapunov based algorithm using FokkerPlanck equation for uncertainty propagation is combined with a probabilist high-gain observer for output feedback and lastly in Sehr and Bitmead (2017) the particle filter is used for both state estimation and uncertainty propagation.

10.1016/j.ifacol.2019.06.139

Most work in SNMPC has been concerned with the application of polynomial chaos expansions (PCE) first proposed in Fagiano and Khammash (2012). PCE have been shown to be significantly more efficient at propagating uncertainties than Monte Carlo methods for moderate numbers of uncertain parameters. The method has been applied by Mesbah et al. (2014) to an EMPC problem to obtain the required distribution of crystals of a batch process, which use Chebyshev's inequality to formulate the chance constraints. Alternatively, Streif et al. (2014) uses a sample based method instead on the PCE expansion to approximate the chance constraint, which is less conservative but also more expensive. PCE has further been applied in stochastic LMPC to great success for which the coefficients are found using the Galerkin method (Paulson et al., 2014; Lucia et al., 2015). A major disadvantage of PCE is the inherent exponential scaling with the number of uncertain parameters and the difficulty of dealing with time-varying uncertainties. Similarly to PCE it was suggested in Bradford and Imsland (2018b) to use Gaussian processes instead of orthogonal polynomials, which has the advantage that it considers the uncertainty from the approximation itself.

In this paper the PCE NMPC methodology is extended to handle the case of output feedback, i.e. where only noisy measurements of a measured output are available instead of the full state. To accomplish this, similarly as in the case of PCE LMPC (Mühlpfordt et al., 2016), we combine PCE NMPC with a recursive PCE filtering approach. The scheme is verified on a complex case study of a semi-batch polymerization reaction directly minimising the required batch time, while fulfilling several safety and product quality constraints. The paper consists of the following sections. In the next section the problem to be solved is defined and the main algorithm is introduced. In section 3 background on PCE is given, in section 4 a PCE filter is outlined and in section 5 a PCE SNMPC formulation is described. In section 6 a case study is introduced. Section 7 presents the results and discussion of this case study. Lastly, in section 8 conclusions are given.

2. PROBLEM SETUP

In this section we outline the problem to be solved, for which we propose a new framework. Consider a discrete-time nonlinear equation system with stochastic uncertainties:

$$\mathbf{x}(t+1) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t), \boldsymbol{\theta}), \quad \mathbf{x}(0) = \mathbf{x}_0(\boldsymbol{\theta}) \tag{1}$$

$$\mathbf{y}(t) = \mathbf{h}(\mathbf{x}(t), \boldsymbol{\theta}) + \boldsymbol{\nu} \tag{2}$$

where t is the discrete time, $\mathbf{x} \in \mathbb{R}^{n_\mathbf{x}}$ are the system states, $\mathbf{u} \in \mathbb{R}^{n_\mathbf{u}}$ denote the control inputs, $\boldsymbol{\theta} \in \mathbb{R}^{n_\boldsymbol{\theta}}$ are time-invariant uncertainties, $\mathbf{f} : \mathbb{R}^{n_\mathbf{x}} \times \mathbb{R}^{n_\mathbf{u}} \times \mathbb{R}^{n_\boldsymbol{\theta}} \to \mathbb{R}^{n_\mathbf{x}}$ represents the nonlinear dynamic system, $\mathbf{y} \in \mathbb{R}^{n_\mathbf{y}}$ denote the measurements, $\mathbf{h} : \mathbb{R}^{n_\mathbf{x}} \times \mathbb{R}^{n_\boldsymbol{\theta}} \to \mathbb{R}^{n_\mathbf{y}}$ are the output equations and $\boldsymbol{\nu} \in \mathbb{R}^{n_\mathbf{y}} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\nu}})$ is the measurement noise assumed to follow a zero mean multivariate normal distribution with known covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{\nu}}$. The initial condition $\mathbf{x}(0)$ is assumed to be a function of the same uncertain parameters expressed as $\mathbf{x}_0(\boldsymbol{\theta})$.

To express the probability distribution of $\boldsymbol{\theta}$ we use PCEs; for background information on PCEs refer to section 3. Let $\boldsymbol{\theta}_t(\boldsymbol{\xi})$ correspond to the PCE of $\boldsymbol{\theta}$ at time t. It is assumed that we are initially given a PCE of $\boldsymbol{\theta}$ denoted by $\boldsymbol{\theta}_0(\boldsymbol{\xi})$. In general this initial probability distribution will be broad with a large variance to represent the uncertainty in the value of $\boldsymbol{\theta}$. At each sampling time $t+1$ we measure a value of $\mathbf{y}(t+1)$ according to Eq.(2), which is then used to update $\boldsymbol{\theta}_t(\boldsymbol{\xi})$ to $\boldsymbol{\theta}_{t+1}(\boldsymbol{\xi})$ recursively using the PCE filter outlined in section 4 by updating the coefficients of the PCE.

It should be noted that the uncertainty of the current state estimate $\mathbf{x}(t)$ is a consequence of the uncertainty in $\boldsymbol{\theta}$ and can be expressed as a function of it, which we will denote as $\mathbf{x}(t) = \mathbf{x}_t(\boldsymbol{\theta})$. Often an explicit form of $\mathbf{x}_t(\boldsymbol{\theta})$ is not available and instead $\mathbf{x}_t(\boldsymbol{\theta})$ needs to be understood as the simulation forward from $\mathbf{x}(0) = \mathbf{x}_0(\boldsymbol{\theta})$ to $\mathbf{x}(t) = \mathbf{x}_t(\boldsymbol{\theta})$ using Eq.(1).

Given the PCE $\boldsymbol{\theta}_t(\boldsymbol{\xi})$ and the function $\mathbf{x}_t(\boldsymbol{\theta})$ at each discrete time t, we wish to control the dynamic system defined by Eq.(1) subject to chance constraints and a stochastic objective. To accomplish this we solve a probabilistic finite time-horizon optimal control problem repeatedly in MPC fashion at time t:

$$\begin{aligned}
&\underset{\mathbf{u}_N}{\text{minimize}} \quad \mathbb{E}(J(N, \mathbf{x}_t(\cdot), \mathbf{u}_N, \boldsymbol{\theta}_t(\boldsymbol{\xi}))) \\
&\text{subject to} \\
&\mathbf{x}(k+1) = \mathbf{f}(\mathbf{x}(k), \mathbf{u}(k), \boldsymbol{\theta}_t(\boldsymbol{\xi})) && \forall k \in \mathbb{N}_k \\
&\mathbb{P}(g_j(\mathbf{x}(k), \mathbf{u}(k)) \leq 0) \leq 1 - \epsilon && \forall (k,j) \in \mathbb{N}_{k+1} \times \mathbb{N}_g \\
&\mathbb{P}(g_j^N(\mathbf{x}(N), \mathbf{u}(N)) \leq 0) \leq 1 - \epsilon && \forall j \in \mathbb{N}_g^N \\
&\mathbf{u}(k) \in \mathbb{U}_k && \forall k \in \mathbb{N}_k \\
&\mathbf{x}(0) = \mathbf{x}_t(\boldsymbol{\theta}_t(\boldsymbol{\xi}))
\end{aligned} \tag{3}$$

where $\mathbb{N}_g = \{1, \ldots, n_g\}$, $\mathbb{N}_g^N = \{1, \ldots, n_g^N\}$, $\mathbb{N}_k = \{0, ..., N-1\}$, $\mathbb{N}_{k+1} = \{1, ..., N\}$, the expectation of $J(N, \mathbf{x}_t, \mathbf{u}_N, \boldsymbol{\theta})$ is the objective, N is the time horizon, the probability of the functions $g_j : \mathbb{R}^{n_\mathbf{x}} \times \mathbb{R}^{n_\mathbf{u}} \times \mathbb{R}^{n_\boldsymbol{\theta}} \to \mathbb{R}$ over all times and $g_j^N : \mathbb{R}^{n_\mathbf{x}} \times \mathbb{R}^{n_\mathbf{u}} \times \mathbb{R}^{n_\boldsymbol{\theta}} \to \mathbb{R}$ at the final time exceeding 0 should be less than ϵ, the constraints on the inputs are given by $\mathbb{U}_k \subset \mathbb{R}^{n_\mathbf{u}}$ and lastly $\mathbf{u}_N := \{\mathbf{u}(0), \ldots, \mathbf{u}(N-1)\}$ represents the control inputs.

The problem in Eq.(3) is intractable due to the requirement to propagate stochastic uncertainties through nonlinear transformations and in addition the multivariate integral definition of the chance constraints. Instead, we solve a simplified problem approximating Eq.(3) using PCEs outlined in section 5. Overall the algorithm we propose uses PCE to express the uncertainty $\boldsymbol{\theta}$ described in section 3, exploits this uncertainty description to control the dynamic system in Eq.(1) using PCE SNMPC introduced in section 5 and lastly uses the measurements from Eq.(2) to update the uncertainty description utilising a PCE filter outlined in section 4. The algorithm is summarised below.

Algorithm 1: Output feedback PCE SNMPC

Input : $\mathbf{f}(\mathbf{x}, \mathbf{u}, \boldsymbol{\theta})$, $\mathbf{h}(\mathbf{x}, \boldsymbol{\theta})$, $\boldsymbol{\Sigma}_{\boldsymbol{\nu}}$, $\boldsymbol{\theta}_0(\boldsymbol{\xi})$, $\mathbf{x}_0(\boldsymbol{\theta})$

for *each sampling time* $t = 0, 1, 2, \ldots$ **do**

(1) Solve PCE SNMPC problem with $\boldsymbol{\theta}_t(\boldsymbol{\xi})$ and $\mathbf{x}_t(\boldsymbol{\theta}_t(\boldsymbol{\xi}))$ and obtain optimal control actions

(2) Apply first part of the control actions to the plant

(3) Measure $\mathbf{y}(t+1)$

(4) Apply the PCE filter to update $\boldsymbol{\theta}_t(\boldsymbol{\xi})$ to $\boldsymbol{\theta}_{t+1}(\boldsymbol{\xi})$

(5) Determine $\mathbf{x}_{t+1}(\boldsymbol{\theta}_{t+1})$ using $\mathbf{f}(\cdot, \cdot, \boldsymbol{\theta}_{t+1})$ recursively from an updated initial condition $\mathbf{x}(0) = \mathbf{x}_0(\boldsymbol{\theta}_{t+1})$

end

3. BACKGROUND: PCE

The generalized polynomial chaos expansion (gPCE) scheme will be briefly outlined in this section, for more information refer to Mesbah et al. (2014); Xiu and Karniadakis (2003);

Eldred and Burkardt (2009). A second order process $\theta(\boldsymbol{\xi})$ can be expressed as the following convergent expansion:

$$\theta(\boldsymbol{\xi}) = \sum_{j=0}^{\infty} a_j \phi_{\boldsymbol{\alpha}_j}(\boldsymbol{\xi}) \tag{4}$$

where $\boldsymbol{\xi} \in \mathbb{R}^{n_\xi}$ is a n_ξ-dimensional random variable with a specified pdf, a_j denotes expansion coefficients and $\phi_{\boldsymbol{\alpha}_j} = \prod_{i=1}^{n_\xi} \phi_{\alpha_{j_i}}(\xi_i)$ denotes multivariate polynomials with $\phi_{\alpha_{j_i}}(\xi_i)$ being univariate polynomials of ξ_i of degree α_{j_i}.

The univariate polynomials are chosen according to the Askey scheme based on the probability distribution of the corresponding ξ_i to satisfy an orthogonality property, e.g. if ξ_i is a standard Gaussian random variable with zero-mean and unit variance, then $\phi_{\alpha_{j_i}}(\xi_i)$ are chosen as Hermite polynomials. Univariate Hermite polynomials He with degree j in terms of ξ_i are:

$$He_j(\xi_i) = (-1)^j \exp\left(\frac{1}{2}\xi_i^2\right) \frac{d^j}{d\xi_i^j} \exp\left(-\frac{1}{2}\xi_i^2\right) \tag{5}$$

For these orthogonal polynomials we have the useful property:

$$\langle \phi_i, \phi_j \rangle = \int \phi_i(\boldsymbol{\xi})\phi_j(\boldsymbol{\xi})p(\boldsymbol{\xi})d\boldsymbol{\xi} = \delta_{ij}\langle \phi_i^2 \rangle \tag{6}$$

where δ_{ij} is the Kronecker delta and $p(\boldsymbol{\xi})$ is the pdf of $\boldsymbol{\xi}$.

To approximate $\theta(\boldsymbol{\xi})$ for practical reasons the PCE in Eq.(4) needs to be truncated:

$$\theta(\boldsymbol{\xi}) = \sum_{0 \le |\boldsymbol{\alpha}| \le m} a_j \phi_{\boldsymbol{\alpha}_j}(\boldsymbol{\xi}) = \mathbf{a}^T \boldsymbol{\phi}(\boldsymbol{\xi}) \tag{7}$$

where $\boldsymbol{\phi}(\cdot) = [\phi_1(\cdot), \ldots, \phi_L(\cdot)]^T$ contains the multivariate polynomials of the expansion, m denotes the order of truncation and $|\boldsymbol{\alpha}| = \sum_{i=1}^{n_\xi} \alpha_i$. The truncated series consists of $L = \frac{(n_\xi + m)!}{n_\xi! m!}$ terms and $\mathbf{a} \in \mathbb{R}^L$ represents a vector of coefficients of these terms.

Next we need to evaluate the coefficients $\mathbf{a}$, which we accomplish by using the non-intrusive spectral projection approach based on the orthogonality property in Eq. (6):

$$a_j = \frac{\langle \theta(\boldsymbol{\xi}), \phi_j \rangle}{\langle \phi_j^2 \rangle} = \frac{1}{\langle \phi_j^2 \rangle} \int \theta(\boldsymbol{\xi})\phi_j(\boldsymbol{\xi})p(\boldsymbol{\xi})d\boldsymbol{\xi} \tag{8}$$

The evaluation of the integral in Eq.(8) can be approximated employing sample-based approaches. Quadrature methods are the most popular due to their significantly improved convergence rates compared to MC approaches for moderate dimensional problems. Quadrature methods take the following form:

$$\int \theta(\boldsymbol{\xi})\phi_j(\boldsymbol{\xi})p(\boldsymbol{\xi})d\boldsymbol{\xi} \approx \frac{1}{\langle \phi_j^2 \rangle} \sum_{q=1}^{N_q} w_q \theta(\boldsymbol{\xi}^{(q)})\phi_j(\boldsymbol{\xi}^{(q)}) \tag{9}$$

leading to the following sample estimate of the coefficients:

$$\hat{\mathbf{a}} = \mathbf{w}(\boldsymbol{\Theta})^T \boldsymbol{\Phi}(\boldsymbol{\Xi}) * \langle \boldsymbol{\phi}^2 \rangle^{-1} \tag{10}$$

where $*$ denotes element-wise multiplication, N_q is the total number of quadrature points, $\boldsymbol{\Xi} = [\boldsymbol{\xi}^{(1)}, \ldots, \boldsymbol{\xi}^{(N_q)}]^T \in \mathbb{R}^{N_q \times n_\xi}$ represents the quadrature sample design, $\mathbf{w}(\boldsymbol{\Xi}) = [w_1\theta(\boldsymbol{\xi}^{(1)}), \ldots, w_{N_q}\theta(\boldsymbol{\xi}^{(N_q)})]^T \in \mathbb{R}^{N_q}$, w_q the quadrature weights, $\langle \boldsymbol{\phi}^2 \rangle^{-1} = [\langle \phi_1^2 \rangle, \ldots, \langle \phi_L^2 \rangle] \in \mathbb{R}^L$, $\boldsymbol{\Phi}(\boldsymbol{\Xi}) = [\boldsymbol{\phi}(\boldsymbol{\xi}^{(1)}), \ldots, \boldsymbol{\phi}(\boldsymbol{\xi}^{(N_q)})]^T \in \mathbb{R}^{N \times L}$ and the response vector is given by $\boldsymbol{\Theta} = [\theta(\boldsymbol{\xi}^{(1)}), \ldots, \theta(\boldsymbol{\xi}^{(N_q)})] \in \mathbb{R}^{N_q}$.

The type of quadrature method is again chosen based on the pdf of $\boldsymbol{\xi}$. For standard Gaussian distributed ξ_i Gauss-Hermite quadrature is chosen. The number of points required depends on the order of accuracy required and the dimension of $\boldsymbol{\xi}$. To integrate polynomials correctly up to degree p, $N_q = (p+1)^{n_\xi}$ points are required. This quickly becomes prohibitive, so we instead use a sparse Gauss-Hermite (sGH) quadrature method in this work proposed in Jia et al. (2012).

Using the coefficient approximation from Eq. (10) we have a representation for the random variable $\theta(\boldsymbol{\xi})$ parametrized by $\boldsymbol{\xi}$. The polynomial chaos expansion may also be used to represent multivariate random variables. Let a multivariate stochastic variable be given by $\boldsymbol{\theta}(\boldsymbol{\xi}) = [\theta_1(\boldsymbol{\xi}), \ldots, \theta_{n_\theta}(\boldsymbol{\xi})]^T \in \mathbb{R}^{n_\theta = n_\xi}$ with coefficients collected in $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_{n_\theta}]$, which is parametrized in terms of standard normal variables $\boldsymbol{\xi}$ with the same dimension. The properties of $\boldsymbol{\theta}(\boldsymbol{\xi})$ are dependent on the coefficients $\mathbf{A}$ of the expansion.

Let each component of $\boldsymbol{\theta}(\boldsymbol{\xi})$ be given by a truncated PCE with the same order of truncation and the same number of terms L, then the moments of $\boldsymbol{\theta}(\boldsymbol{\xi})$ have a closed-form expression in terms of the PCE coefficients. Moments can be defined as:

$$M_{\mathbf{r}}(\mathbf{A}) = \int \prod_{i=1}^{n_\xi} \theta_i^{r_i}(\boldsymbol{\xi})p(\boldsymbol{\xi})d\boldsymbol{\xi} \tag{11}$$

where $\mathbf{r} \in \mathbb{R}^{n_\xi}$ is a vector defining the moments with $k = \sum_{i=1}^{n_\xi} r_i$ order.

The moments of the PCE expansion with the definition in Eq.(11) are (Dutta and Bhattacharya, 2010):

$$M_{\mathbf{r}}(\mathbf{A}) = \int \prod_{i=1}^{n_\xi} (\mathbf{a}_i^T \boldsymbol{\phi}(\boldsymbol{\xi}))^{r_i} p(\boldsymbol{\xi})d\boldsymbol{\xi} \tag{12}$$

4. PCE FILTER

The state estimation step concerns the update of $\boldsymbol{\theta}_{t-1}(\boldsymbol{\xi})$ to $\boldsymbol{\theta}_t(\boldsymbol{\xi})$ given the noisy measurements available, in which we assume that $\boldsymbol{\xi}$ follows a standard normal distribution. The following outline was taken from Madankan et al. (2013); Mühlpfordt et al. (2016). Let $D_t = \{\mathbf{y}(1) \ldots, \mathbf{y}(t)\}$ be the measurements collected up to time t and $\mathbf{y}(t)$ the most recent measurement. Bayes's rule can be employed to update $\boldsymbol{\theta}(\boldsymbol{\xi})$ recursively as:

$$p(\boldsymbol{\theta}(\boldsymbol{\xi})|D_t) = \frac{p(\boldsymbol{\theta}(\boldsymbol{\xi})|D_{t-1})p(\mathbf{y}(t)|\boldsymbol{\theta}(\boldsymbol{\xi}), D_{t-1})}{p(\mathbf{y}(t)|D_{t-1})} \tag{13}$$

where $p(\boldsymbol{\theta}(\boldsymbol{\xi})|D_{t-1})$ is the prior distribution of $\boldsymbol{\theta}(\boldsymbol{\xi})$ at time t given all observations up to time $t-1$, $p(\mathbf{y}(t)|\boldsymbol{\theta}(\boldsymbol{\xi}), D_{t-1})$ is the likelihood $\mathbf{y}(t)$ is observed given $\boldsymbol{\theta}(\boldsymbol{\xi})$ at time t, which does not depend on the observations D_{t-1} and D_{t-1} is therefore dropped. We define $p(\mathbf{y}(t)|\boldsymbol{\theta}(\boldsymbol{\xi})) = \mathcal{N}(\mathbf{y}(t)|\mathbf{h}(\mathbf{x}(t), \boldsymbol{\theta}(\boldsymbol{\xi})), \boldsymbol{\Sigma}_{\mathbf{v}})$ as multivariate normal likelihood with mean $\mathbf{h}(\mathbf{x}(t), \boldsymbol{\theta}(\boldsymbol{\xi}))$ and covariance $\boldsymbol{\Sigma}_{\mathbf{v}}$ evaluated at $\mathbf{y}(t)$. The pdf $p(\mathbf{y}(t)|D_{t-1})$ is the total probability of observation $\mathbf{y}(t)$ at time t given by:

$$p(\mathbf{y}(t)|D_{t-1}) = \int p(\mathbf{y}(t)|\boldsymbol{\theta}(\boldsymbol{\xi}))p(\boldsymbol{\theta}(\boldsymbol{\xi})|D_{t-1})d\boldsymbol{\theta} \tag{14}$$

Calculating Eq. (14) analytically is difficult and we therefore use sampling instead. We know the distribution $p(\boldsymbol{\theta}(\boldsymbol{\xi})|D_{t-1})$, since it is assumed that $\boldsymbol{\xi}$ follows a standard normal distribution. The functions $\boldsymbol{\theta}_{t-1}(\boldsymbol{\xi})$ and $\boldsymbol{\theta}_t(\boldsymbol{\xi})$ are the PCEs corresponding to the pdfs $p(\boldsymbol{\theta}(\boldsymbol{\xi})|D_{t-1})$ and $p(\boldsymbol{\theta}(\boldsymbol{\xi})|D_t)$ respectively. Latin hypercube sampling was applied together with the inverse normal cumulative transformation (Stein, 1987):

$$\alpha = \frac{1}{N_s} \sum_{s=1}^{N_s} p(\mathbf{y}(t)|\boldsymbol{\theta}_{t-1}(\boldsymbol{\xi}^{(s)})) \tag{15}$$

where α is the sample estimate of $p(\mathbf{y}(t)|D_{t-.1})$, N_s is the sample size and $\boldsymbol{\xi}^{(s)} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ are the sample points.

The prior distribution $p(\boldsymbol{\theta}(\boldsymbol{\xi})|D_{t-1})$ is given by the previous posterior distribution of $\boldsymbol{\theta}$ due to the assumed time-invariance. If we take both sides of Eq.(13) times $\prod_{j=1}^{n_\xi} \theta_j^{r_j}$ and integrate over both sides we obtain the following:

$$M_{\mathbf{r}}^{+} = \frac{\int \prod_{j=1}^{n_\xi} \theta_j^{r_j}(\boldsymbol{\xi}) p(\mathbf{y}(t)|\boldsymbol{\theta}(\boldsymbol{\xi})) p(\boldsymbol{\theta}(\boldsymbol{\xi})|D_{t-1}) d\boldsymbol{\theta}}{p(\mathbf{y}(t)|D_{t-1})} \quad (16)$$

where $M_{\mathbf{r}}^{+} = \int \prod_{j=1}^{n_\xi} \theta_j^{r_j}(\boldsymbol{\xi}) p(\boldsymbol{\theta}(\boldsymbol{\xi})|D_t) d\boldsymbol{\theta}$ and let $k = \sum_{j=1}^{n_\xi} r_j$. Now $M_{\mathbf{r}}^{+}$ refers to the various k-th order moments with respect to the updated distribution of $\boldsymbol{\theta}$, $p(\boldsymbol{\theta}(\boldsymbol{\xi})|D_t)$.

Now using the sample estimate in Eq.(15) and applying a further sample estimate to Eq.(16) we obtain:

$$M_{\mathbf{r}}^{(s)+} = \frac{1}{\alpha N_s} \sum_{s=1}^{N_s} \prod_{j=1}^{n_\xi} \theta_{t-1,j}^{r_j}(\boldsymbol{\xi}^{(s)}) p(\mathbf{y}(t)|\boldsymbol{\theta}_{t-1}(\boldsymbol{\xi}^{(s)})) \quad (17)$$

where $M_{\mathbf{r}}^{(s)+}$ is an approximation of the RHS of Eq.(16).

To update $\boldsymbol{\theta}_{t-1}(\boldsymbol{\xi})$ we match the moments found in Eq.(17) with those of the PCE $\boldsymbol{\theta}_t(\boldsymbol{\xi})$, which are a function of its coefficients as shown in Eq.(12). The PCE is then fitted by solving a nonlinear least-squares optimization problem:

$$\hat{\mathbf{A}}_t = \arg\min_{\mathbf{A}_t} \sum_{k \leq m} ||M_{\mathbf{r}}^{+}(\mathbf{A}_t) - M_{\mathbf{r}}^{(s)+}||_2^2 \quad (18)$$

where $k = \sum_{j=1}^{n_\xi} r_j$ was defined above as the order of the moments and hence m defines the total order of moments we want to match. $M_{\mathbf{r}}^{+}(\mathbf{A}_t)$ is parametrized by $\mathbf{A}_t$ as shown in Eq.(12). The estimated coefficients $\hat{\mathbf{A}}_t$ then define the updated PCE $\boldsymbol{\theta}_t(\boldsymbol{\xi})$ as required for Algorithm 1.

5. PCE SNMPC

In this section we formulate an approximate algorithm to solve the OCP in Eq.(3) using PCEs. We assume the time is t and we are given the PCE $\boldsymbol{\theta}_t(\boldsymbol{\xi})$ accounting for all the data available with the function $\mathbf{x}_t(\boldsymbol{\theta})$ describing the current state in terms of $\boldsymbol{\theta}$. The aim is to control the dynamic system in Eq.(1) given these uncertainty descriptions by reformulating Eq.(3). PCEs in this regard can be used to obtain accurate mean and variance predictions of nonlinear transformations, however estimating the chance constraints remains a difficult problem.

The mean and variance of a PCE expansion in terms of $\boldsymbol{\xi}$ of a 1-dimensional random variable γ with coefficients $\mathbf{a} \in \mathbb{R}^L$ using the definition in Eq.(12) can be expressed as follows:

$$\mathbb{E}(\gamma) \approx a_1 \quad (19)$$

$$\text{Var}(\gamma) \approx \sum_{i=2}^{L} a_i^2 \mathbb{E}(\Phi_i^2(\boldsymbol{\xi})) \quad (20)$$

We use Chebychev's inequality to robustly reformulate the chance constraints in terms of only the mean and variance of the constraint function. Let γ be a generic random variable with a finite variance, then (Mesbah et al., 2014):

$$\mathbb{P}(\gamma \leq 0) \geq 1 - \epsilon \Rightarrow \kappa_\epsilon \sigma_\gamma + \hat{\gamma} \leq 0, \ \kappa_\epsilon = \sqrt{(1-\epsilon)/\epsilon} \quad (21)$$

where $\epsilon \in (0,1) \subset \mathbb{R}$ is the probability that γ exceeds 0, $\hat{\gamma}$ and σ_γ^2 are the mean and variance of γ respectively.

Next we use results from section 3. In essence we evaluate the coefficients of a PCE expansion online using a quadrature rule as shown in Eq.(10). The quadrature sample design is given by $\boldsymbol{\Xi} = [\boldsymbol{\xi}^{(1)}, \ldots, \boldsymbol{\xi}^{(N_q)}]^T \in \mathbb{R}^{N_q \times n_\xi}$ with N_q being the number of sample points. Once these are defined the matrices $\boldsymbol{\Phi}(\boldsymbol{\Xi}) = [\boldsymbol{\phi}(\boldsymbol{\xi}^{(1)}), \ldots, \boldsymbol{\phi}(\boldsymbol{\xi}^{(N_q)})]^T$ and $\langle \boldsymbol{\phi}^2 \rangle^{-1}$ can be calculated offline. Each sample in $\boldsymbol{\Xi}$ represents a separate dynamic simulation according to Eq.(1), the data from which is then used according to Eq.(10) to determine the PCE coefficients online. These are then in turn used to estimate the mean and variance from Eqs.(20) and (19) to estimate the objective and chance constraints according to Chebychev's inequality in Eq.(3). It is important to note that the SNMPC algorithm can have a different order of PCE than $\boldsymbol{\theta}(\boldsymbol{\xi})$. The SNMPC algorithm to reformulate Eq. (3) can be stated as:

$$\begin{aligned}
&\underset{\mathbf{u}_N}{\text{minimize}} \quad \hat{a}_1^J \\
&\text{subject to} \\
&\mathbf{x}^{(i)}(k+1) = \mathbf{f}(\mathbf{x}^{(i)}(k), \mathbf{u}(k), \boldsymbol{\theta}_t(\boldsymbol{\xi}^{(i)})) \ \forall (k,i) \in \mathbb{N}_k \times \mathbb{N}_q \\
&\kappa_\epsilon \sum_{i=2}^{L} \left(\hat{a}_i^{g_{jk}}\right)^2 \mathbb{E}(\Phi_i^2(\boldsymbol{\xi})) + \hat{a}_1^{g_{jk}} \leq 0 \qquad \forall (k,j) \in \mathbb{N}_{k+1} \times \mathbb{N}_g \\
&\kappa_\epsilon \sum_{i=2}^{L} \left(\hat{a}_i^{g_j^N}\right)^2 \mathbb{E}(\Phi_i^2(\boldsymbol{\xi})) + \hat{a}_1^{g_j^N} \leq 0 \qquad \forall j \in \times \mathbb{N}_g^N \\
&\hat{\mathbf{a}}^J = \mathbf{w}(\boldsymbol{\Theta}^J)^T \boldsymbol{\Phi}(\boldsymbol{\Xi}) * \langle \boldsymbol{\phi}^2 \rangle^{-1} \\
&\hat{\mathbf{a}}^{g_{jk}} = \mathbf{w}(\boldsymbol{\Theta}^{g_{jk}})^T \boldsymbol{\Phi}(\boldsymbol{\Xi}) * \langle \boldsymbol{\phi}^2 \rangle^{-1} \qquad \forall (k,j) \in \mathbb{N}_{k+1} \times \mathbb{N}_g \\
&\hat{\mathbf{a}}^{g_j^N} = \mathbf{w}(\boldsymbol{\Theta}^{g_j^N})^T \boldsymbol{\Phi}(\boldsymbol{\Xi}) * \langle \boldsymbol{\phi}^2 \rangle^{-1} \qquad \forall j \in \mathbb{N}_g^N \\
&\mathbf{u}(k) \in \mathbb{U}_k \qquad \forall k \in \mathbb{N}_k \\
&\mathbf{x}^{(i)}(0) = \mathbf{x}_t(\boldsymbol{\theta}_t(\boldsymbol{\xi}^{(i)})) \qquad \forall i \in \mathbb{N}_q
\end{aligned} \quad (22)$$

where $\mathbf{x}^{(i)}$ denotes the state for each scenario i,
$\mathbf{w}(\boldsymbol{\Theta}) = [w_1 \Theta_1, \ldots, w_{N_q} \Theta_{N_q}]$ with w_i being the quadrature weights and the data matrices $\boldsymbol{\Theta}^J = [J(N, \mathbf{x}_t(\boldsymbol{\theta}_t(\boldsymbol{\xi}^{(1)}), \mathbf{u}_N, \boldsymbol{\theta}_t(\boldsymbol{\xi}^{(1)})), \ldots, J(N, \mathbf{x}_t(\boldsymbol{\theta}_t(\boldsymbol{\xi}^{(N_q)}), \mathbf{u}_N, \boldsymbol{\theta}_t(\boldsymbol{\xi}^{(N_q)}))]$,
$\boldsymbol{\Theta}^{g_{jk}} = [g_j(\mathbf{x}^{(1)}(k), \mathbf{u}(k)), \ldots, g_j(\mathbf{x}^{(N_q)}(k), \mathbf{u}(k))]$,
$\boldsymbol{\Theta}^{g_j^N} = [g_j^N(\mathbf{x}^{(1)}(N), \mathbf{u}(N)), \ldots, g_j^N(\mathbf{x}^{(N_q)}(N), \mathbf{u}(N))]$
Eq.(22) gives the required control inputs for Algorithm 1.

6. SEMI-BATCH REACTOR CASE STUDY

Algorithm 1 outlined in section 2 is applied to a semi-batch polymerization reactor for the production of polyol from propylene oxide (PO). An extensive model for this process has been presented in Nie et al. (2013a), which has been used in Jung et al. (2015) for NMPC and in Jang et al. (2016) for multi-stage NMPC. A schematic of the process is shown in Fig. 1.

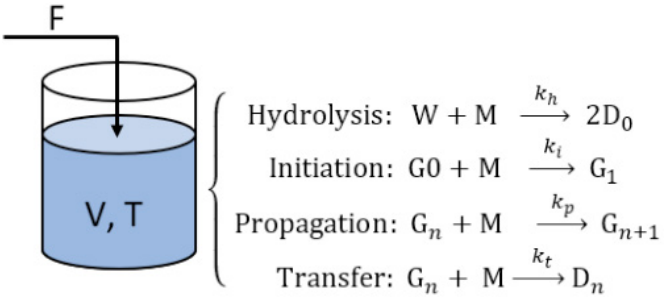

Fig. 1. F is the monomer feedrate, V and T are the volume and temperature of the liquid in the reactor respectively, W is water, M is the monomer, D_n and G_n are the dormant and active product chains with length n respectively.

To reduce the computational times we applied the method of moments (Rivero, 2005; Nie et al., 2013b) to derive differential equations for the average molecular weight. In addition, we disregard the balance equations for the unsaturated proportion of the polymer. Due the importance of temperature control a heat balance was added, in which perfect temperature control was previously assumed. This equation can nonetheless be found in Nie et al. (2013a), where it is used as a constraint. The objective was set to minimize batch time (t_f[s]) by varying the monomer feed rate F[mol/s] and the cooling water temperature T_C[K] to achieve a number average molecular weight ($NAMW$) of 450g/mol and ensure that the amount of the monomer (PO) contained in the reactor does not exceed 120ppm. During this operation the reactor temperature T[K] is constrained to remain below 420K. The chance of constraint violation was set to 0.1. We assume the amount of catalyst (n_C[mol]) and the pre-exponential coefficient of the propagation kinetic constant (A_p[m^3/mol/s]) to be uncertain and given by a PCE. Measurements during the reaction are the pressure (P[bar]) and temperature (T[K]) of the reactor. For discredization orthogonal collocation was employed. The optimization problems for the PCE SNMPC and PCE filter were solved using Casadi (Andersson et al., 2018) in conjunction with IPOPT (Wächter and Biegler, 2006). The control problem to be solved is summarised in Tab. 1. The missing parameter values and dynamic equation system can be found in Nie et al. (2013a).

Table 1. Specifications of control problem

States ($\mathbf{x}$)	m[g], PO[mol], W[mol], $T[K]$, X_0[mol], γ_0[mol], γ_1[mol]
Outputs ($\mathbf{y}$)	P[bar], T[K]
Output noise	$\mathbf{\Sigma_v} = \text{diag}(0.25, 0.01)$
Inputs (u)	F[mol], T_C[K]
Uncertainties	A_p[m^3/mol/s], n_C[mol]
Objective	minimize t_f[s]
Path constraints	$T[K] - 420 \leq 0$
End constraints	NAMW[g/mol] $- 450 \leq 0$, $120 -$ PO[ppm] ≤ 0
Probability	$\epsilon = 0.1$
Input constraints	$0 \leq F$[mol/s] ≤ 10, $298.15 \leq T_C$[K]
PCE SNMPC	PCE order = 3, sGH accuracy = 2, sGH manner = 1
PCE filter	Samples = 800, Moments considered = 5, PCE order = 3
Discredization	N = 12, Degree = 5
Initial PCE A_P	$9.05 + 0.25\xi_1 + 0.13(\xi_1 - 1)^2 + 0.04(\xi_1^3 - 3\xi_1)$
Initial PCE n_C	$6.91 + 0.25\xi_2 + 0.13(\xi_2 - 1)^2 + 0.04(\xi_2^3 - 3\xi_2)$
Reactor specs.	$V = 17$m^3, $UA = 1.5 \times 10^4 W/m^2/K$
Initial cond.	$m(0) = 1.6 \times 10^6 g$, $PO(0) = 10^4$mol, $W(0) = 10^3$mol, $X_0(0) = 0$mol, $T(0) = 378.15$K, $\gamma_0(0) = \gamma_0(1) = 10^4$mol

7. RESULTS AND DISCUSSION

Algorithm 1 outlined in section 2 was verified on the case study defined in the previous section firstly by running the NMPC on a specific realization of θ for plant model, in our case $[A_p, n_C] = [7200\text{m}^3/\text{mol/s}, 1700\text{mol}]$, significantly different from the nominal values $[8504\text{m}^3/\text{mol/s}, 1000\text{mol}]$. The results of this are shown in Fig. 2. Firstly, we can see from the first two row of graphs that the parameters are significantly better approximated at the final time than initially, which leads to a large reduction in uncertainty shown by sharper distribution in both cases. Nonetheless while little uncertainty remains of the value for A_p, n_C has still a high uncertainty with a clear bias towards a lower value. This is due to the influence of the prior, which assumed n_C to be around 1000mol. The next three rows of graphs show the control inputs and trajectories of constraints and objective. We can see that generally the batch time becomes less and less, which has two reasons. Firstly, the uncertainty is reduced at every sampling time making the algorithm less conservative and in addition the estimate of the amount catalyst is corrected upwards, which leads to higher NAMW in less time and higher consumption of monomer. First less monomer is fed, since the reactor starts with high concentrations of monomer to ensure the temperature constraint. Thereafter, the monomer is fed in at a maximum rate to reach the required NAMW in minimum time. Lastly, the monomer feedrate is reduced to 0, since at the final time the ppm needs to be less than 120. In this run the NAMW reaches a value of 451g/mol, while the monomer concentration becomes 36ppm. The relative conservativeness particularly with regards to the amount of monomer is due to the bias of the amount of catalyst to a lower value than the true value. The cooling water temperature is lowest at the beginning when the reaction rate is maximum, while at the end the cooling water temperature is relatively high since the reaction rate is close to zero due to the low monomer concentration.

Next the approach was applied to 100 MC samples of the plant according to the initial PCE representations of the uncertainties given in Tab. 1. We compare it to a NMPC approach, which uses the PCE filter to update the parameters, but ignores the distribution of the uncertain parameters and instead uses the mean value as the current best estimate. The first graph shows that the SNMPC variant is more conservative with on average longer batch times, which is expected since it accounts for the uncertainty. The next two graphs highlight the problem of ignoring the uncertainty on the parameters. For the SNMPC approach all of the scenarios obtain a NAMW larger than 450 and only 4% of the scenarios have a ppm larger than 120. This can be seen by the flat pdfs with nearly all the area of the curve in regions required by the chance constraints. For the NMPC scheme however the distributions are peaked around the required value with only 57% of scenarios reaching the required NAMW and 51% of scenarios exceeding 120ppm. Lastly, temperature control for both approaches is good showing that the uncertainties have little effect on the heat balance.

8. CONCLUSIONS

In conclusion, a novel algorithm for output feedback SNMPC has been proposed by employing PCE for both control and filtering. The SNMPC problem involved both objective and probability constraints based on general nonlinear functions. A challenging semi-batch reactor case study showed that the SNMPC framework is able to regulate the process with plant parameters vastly different from the nominal values. It managed to estimate more accurate parameter values, while still accounting for the remaining uncertainty adhering the constraints. In addition, it was shown that taking into account the uncertainty of the parameters is important even after the updates, since it otherwise leads to more than 50% of constraint violations of the end-point constraints.

ACKNOWLEDGEMENTS

Eric Bradford gratefully appreciates BASF for hosting his placement over the course of the Marie-Curie project and the helpful input and discussions for this work. This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 675215.

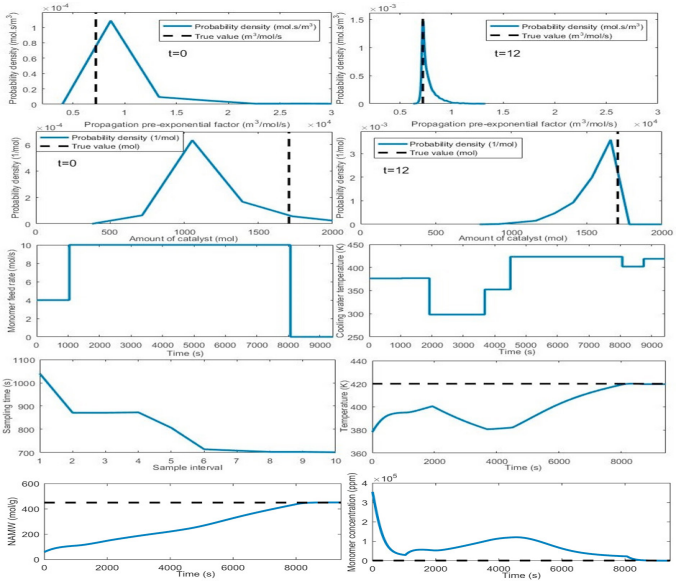

Fig. 2. Probability densities at initial and final time for both uncertainties and state trajectories for a plant model with $[A_p, n_C] = [7200\text{m}^3/\text{mol/s}, 1700\text{mol}]$

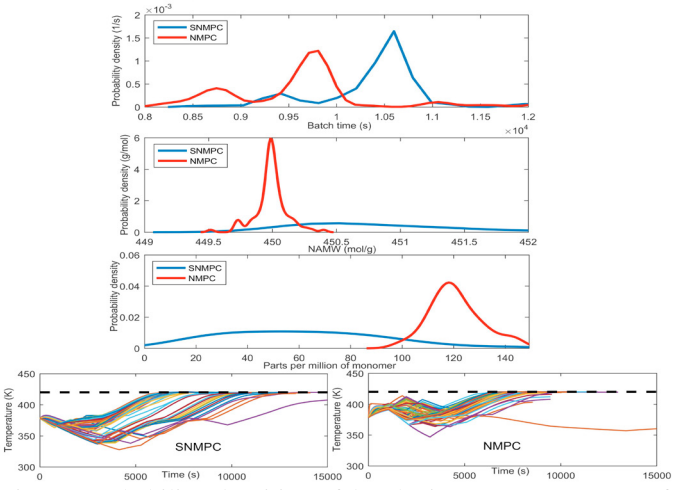

Fig. 3. Probability densities of batch time, NAMW, ppm of monomer at final time and temperature trajectories of NMPC and SNMPC based on 100 MC simulations

REFERENCES

Andersson, J.A.E., Gillis, J., Horn, G., Rawlings, J.B., and Diehl, M. (2018). CasADi: a software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, 1–36.

Bayer, F.A., Müller, M.A., and Allgöwer, F. (2014). Tube-based robust economic model predictive control. *Journal of Process Control*, 24(8), 1237–1246.

Bayer, F.A., Müller, M.A., and Allgöwer, F. (2016). Min-max economic model predictive control approaches with guaranteed performance. In *Decision and Control (CDC), 2016 IEEE 55th Conference on*, 3210–3215. IEEE.

Biegler, L.T. (2010). *Nonlinear programming: concepts, algorithms, and applications to chemical processes*, volume 10. Siam.

Bradford, E. and Imsland, L. (2017a). Expectation constrained stochastic nonlinear model predictive control of a batch bioreactor. *Computer Aided Chemical Engineering*, 40, 1621–1626.

Bradford, E. and Imsland, L. (2017b). Stochastic Nonlinear Model Predictive Control with State Estimation by Incorporation of the Unscented Kalman Filter. *arXiv preprint arXiv:1709.01201*.

Bradford, E. and Imsland, L. (2018a). Economic Stochastic Model Predictive Control Using the Unscented Kalman Filter. *IFAC-PapersOnLine*, 51(18), 417–422.

Bradford, E. and Imsland, L. (2018b). Stochastic Nonlinear Model Predictive Control Using Gaussian Processes. In *2018 European Control Conference (ECC)*, 1027–1034. IEEE.

Cannon, M., Ng, D., and Kouvaritakis, B. (2009). Successive linearization NMPC for a class of stochastic nonlinear systems. In *Nonlinear Model Predictive Control*, 249–262. Springer.

Dutta, P. and Bhattacharya, R. (2010). Nonlinear estimation with polynomial chaos and higher order moment updates. In *American Control Conference (ACC), 2010*, 3142–3147. IEEE.

Eldred, M. and Burkardt, J. (2009). Comparison of Non-Intrusive Polynomial Chaos and Stochastic Collocation Methods for Uncertainty Quantification. In *47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition*, 976. Aerospace Sciences Meetings.

Fagiano, L. and Khammash, M. (2012). Nonlinear stochastic model predictive control via regularized polynomial chaos expansions. In *51st IEEE Conference on Decision and Control (CDC)*, 142–147. IEEE.

Homer, T. and Mhaskar, P. (2018). Output-Feedback Lyapunov-Based Predictive Control of Stochastic Nonlinear Systems. *IEEE Transactions on Automatic Control*, 63(2), 571–577.

Jang, H., Lee, J.H., and Biegler, L.T. (2016). A robust NMPC scheme for semi-batch polymerization reactors. *IFAC-PapersOnLine*, 49(7), 37–42.

Jia, B., Xin, M., and Cheng, Y. (2012). Sparse-grid quadrature nonlinear filtering. *Automatica*, 48(2), 327–341.

Jung, T.Y., Nie, Y., Lee, J.H., and Biegler, L.T. (2015). Model-based on-line optimization framework for semi-batch polymerization reactors. *IFAC-PapersOnLine*, 48(8), 164–169.

Lee, J.H. and Lee, K.S. (2007). Iterative learning control applied to batch processes: An overview. *Control Engineering Practice*, 15(10), 1306–1318.

Lucia, S., Andersson, J.A.E., Brandt, H., Diehl, M., and Engell, S. (2014). Handling uncertainty in economic nonlinear model predictive control: A comparative case study. *Journal of Process Control*, 24(8), 1247–1259.

Lucia, S., Finkler, T., and Engell, S. (2013). Multi-stage nonlinear model predictive control applied to a semi-batch polymerization reactor under uncertainty. *Journal of Process Control*, 23(9), 1306–1319.

Lucia, S., Zometa, P., Kögel, M., and Findeisen, R. (2015). Efficient stochastic model predictive control based on polynomial chaos expansions for embedded applications. In *54th Annual Conference on Decision and Control (CDC)*, 3006–3012. IEEE.

Maciejowski, J.M., Visintini, A.L., and Lygeros, J. (2007). NMPC for complex stochastic systems using a Markov chain Monte Carlo approach. In *Assessment and Future Directions of Nonlinear Model Predictive Control*, 269–281. Springer.

Madankan, R., Singla, P., Singh, T., and Scott, P.D. (2013). Polynomial-chaos-based Bayesian approach for state and parameter estimations. *Journal of Guidance, Control, and Dynamics*, 36(4), 1058–1074.

Mesbah, A., Streif, S., Findeisen, R., and Braatz, R.D. (2014). Stochastic nonlinear model predictive control with probabilistic constraints. In *2014 American Control Conference*, 2413–2419. IEEE.

Mühlpfordt, T., Paulson, J.A., Braatz, R.D., and Findeisen, R. (2016). Output feedback model predictive control with probabilistic uncertainties for linear systems. In *American Control Conference (ACC), 2016*, 2035–2040. IEEE.

Nagy, Z.K. and Braatz, R.D. (2003). Robust nonlinear model predictive control of batch processes. *AIChE Journal*, 49(7), 1776–1786.

Nagy, Z.K., Mahn, B., Franke, R., and Allgöwer, F. (2007). Evaluation study of an efficient output feedback nonlinear model predictive control for temperature tracking in an industrial batch reactor. *Control Engineering Practice*, 15(7), 839–850.

Nie, Y., Biegler, L.T., Villa, C.M., and Wassick, J.M. (2013a). Reactor modeling and recipe optimization of polyether polyol processes: Polypropylene glycol. *AIChE Journal*, 59(7), 2515–2529.

Nie, Y., Biegler, L.T., Villa, C.M., and Wassick, J.M. (2013b). Reactor modeling and recipe optimization of ring-opening polymerization: Block copolymers. *Industrial & Engineering Chemistry Research*, 53(18), 7434–7446.

Patrinos, P., Sopasakis, P., Sarimveis, H., and Bemporad, A. (2014). Stochastic model predictive control for constrained discrete-time Markovian switching systems. *Automatica*, 50(10), 2504–2514.

Paulson, J.A., Mesbah, A., Streif, S., Findeisen, R., and Braatz, R.D. (2014). Fast stochastic model predictive control of high-dimensional systems. In *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*, 2802–2809. IEEE.

Rawlings, J.B. and Amrit, R. (2009). Optimizing process economic performance using model predictive control. In *Nonlinear model predictive control*, 119–138. Springer.

Rivero, P. (2005). Calculation method of molecular weight averages in polymerization with chain-length-dependent termination. *Journal of Polymer Research*, 11(4), 309–315.

Sehr, M.A. and Bitmead, R.R. (2017). Particle model predictive control: Tractable stochastic nonlinear output-feedback MPC. *IFAC-PapersOnLine*, 50(1), 15361–15366.

Sopasakis, P., Herceg, D., Patrinos, P., and Bemporad, A. (2017). Stochastic economic model predictive control for Markovian switching systems. *IFAC-PapersOnLine*, 50(1), 524–530.

Stein, M. (1987). Large sample properties of simulations using Latin hypercube sampling. *Technometrics*, 29(2), 143–151.

Streif, S., Karl, M., and Mesbah, A. (2014). Stochastic nonlinear model predictive control with efficient sample approximation of chance constraints. *arXiv preprint arXiv:1410.4535*.

Wächter, A. and Biegler, L.T. (2006). On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Mathematical programming*, 106(1), 25–57.

Xiu, D. and Karniadakis, G.E. (2003). Modeling uncertainty in flow simulations via generalized polynomial chaos. *Journal of computational physics*, 187(1), 137–167.