Data-driven Nonlinear MPC using Dynamic Response Surface Methodology

Federico Pelagagge [*] **Christos Georgakis** [**] **Gabriele Pannocchia** [*,***]

[*] *University of Pisa, Department of Civil and Industrial Engineering, Largo Lazzarino 2, 56126 Pisa (Italy)*
[**] *Department Chemical and Biological Engineering, Tufts University, Medford, Massachusetts*
[***] *Corresponding author: gabriele.pannocchia@unipi.it*

Abstract:
For many complex processes, it is desirable to use a nonlinear model in the MPC design, and the recently proposed Dynamic Response Surface Methodology (DRSM) is capable of accurately modeling nonlinear continuous processes over semi-infinite time horizons. We exploit the DRSM to identify nonlinear data-driven dynamic models that are used in an NMPC. We demonstrate the ability and effectiveness of the DRSM data-driven model to be used as the prediction model for a nonlinear MPC regulator. This DRSM model is efficiently used to solve a non-equally-spaced finite-horizon optimal control problem so that the number of decision variables is reduced. The proposed DRSM-based NMPC is tested on a representative nonlinear process, an isothermal CSTR in which a second-order irreversible reaction is taking place. It is shown that the obtained quadratic data-driven model accurately represents the open-loop process dynamics and that DRSM-based NMPC is an effective data-driven implementation of nonlinear MPC.

Keywords: Nonlinear MPC, Dynamic Response Surface Methodology, Data-driven MPC, Systems identification

1. INTRODUCTION

Model Predictive Control (MPC) is a powerful tool for process control, hence in recent decades, an extensive number of efforts among researchers have been carried out to improve the usability of MPC products (Darby and Nikolaou, 2012). Even though MPC is a good choice in handling complex interactions and multiple constraints, it still has some disadvantages. The implementation and maintenance costs for an MPC are, in many cases, higher than those of classical control structures. This is primarily due to the cost for the estimation of the related process models. The traditional MPC uses a linear model, but several processes are quite nonlinear necessitating the use of a Nonlinear MPC (NMPC). Most of the time, nonlinear process models are not easily available. Mostly because we do not know enough of the inner workings of a process to write down a knowledge-driven model. Even in cases, we achieve this, the derived model representation is too complex. For such reasons, a data-driven model may be preferred. It is derived by collecting process input and output data over a specific time window and using suitable algorithms to estimate the parameters of the postulated process model. Researchers have mainly focused on Nonlinear ARX (NARX) (Hong et al., 1996), Hammerstein-Wiener (H-W) models (van Wingerden and Verhaegen, 2009) and the Neural Network models (da Cruz Meleiro et al., 2009). These models are identified through Pseudo-Random Binary Signals (PRBS) or Generalized Binary Noise (GBN) (Tulleken, 1990) input experimentation near a steady state.

The Design of Dynamic Experiments (DoDE) (Georgakis, 2013), a generalization of the traditional Design of Experiments (DoE) (Montgomery, 2017), is a novel approach for data-driven process modeling. The available data for estimating the corresponding Dynamic Response Surface Methodology (DRSM) (Klebanov and Georgakis, 2016) model are collected at fixed time intervals during the experiments. Contrary to the static RSM, the model parameters in the DRSM are time-varying and require time-resolved measurements to be estimated. The DRSM model has already been used to estimate a dynamic model for use in an MPC and to calculate the optimal trajectory of a batch process (Wang and Georgakis, 2019). A transformation of the DRSM model into a Hammerstein-Wiener form was also used by an NMPC regulator.

The present paper aims to develop a methodology for the direct use of a DRSM model in an NMPC framework, and to this aim the DRSM model is suitably conceived to include the initial condition as one of the factors so that it can be directly used in a finite-horizon optimal control problem formulation. The DRSM model captures the nonlinear dynamics of a process quite accurately and can be used to develop a linear or a nonlinear recursive dynamic model. Firstly, a DRSM model is estimated representing the dynamics of the process over the design domain where the DoDE experiments are performed. Then, a DRSM-NMPC is designed to control a process. An isothermal CSTR is used here as a case study and the DRSM-NMPC regulator aims to track the desired product concentration. The control performances of DRSM-NMPC, based on a nonlinear dynamic model, are then tested and compared with the Linear DRSM-MPC and a nominal NMPC using the true process model. To streamline the material presentation, the DRSM model is presented for a single-input single-output (SISO) process, but then a brief discussion on how to treat the multi-input multi-output case is also presented.

The rest of the paper is organized as follows. The problem definition and the methodology for the DRSM identification are given in the following section 2. The proposed DRSM-NMPC is formulated in Section 3. The resulting DRSM-NMPC algorithm is tested on the simulated example in Section 4. Finally, conclusions are presented in Section 5.

2. DRSM MODEL

2.1 Plant dynamics

We consider a discrete-time nonlinear dynamic system:

$$\begin{aligned} \xi_p^+ &= f_p(\xi_p, u) \\ y_p &= h_p(\xi_p) + v_p \end{aligned} \tag{1}$$

where $\xi_p \in \mathbb{R}^{n_\xi}$, $u \in \mathbb{R}^{n_u}$ and $y_p \in \mathbb{R}^{n_y}$ are the current plant state, input and output, respectively, $\xi_p^+ \in \mathbb{R}^{n_\xi}$ are the successor states and v_p is the output noise. Although not explicitly mentioned in (1), the discretization time is denoted by h.

We assume that the plant dynamics (1) is not known, therefore in this section, a methodology is developed to identify a DRSM model from experimental data able to approximately capture the plant dynamics.

2.2 DRSM model

As anticipated, to simplify the discussion about the DRSM model, we consider a SISO plant, i.e. $n_u = n_y = 1$. We consider in this discussion a quadratic DRSM model given by:

$$y(\theta) = \beta_0(\theta) + \sum_{i=1}^{n} \beta_i(\theta) x_i + \sum_{i=1}^{n} \sum_{i<j}^{n} \beta_{ij}(\theta) x_i x_j + \sum_{i=1}^{n} \beta_{ii}(\theta) x_i^2 \tag{2}$$

where $y(\theta)$ is the model output value evaluated at the dimensionless time θ (later defined), and x_i are normalized model factors with $-1 \le x_i \le 1$, $i = 1, \cdots, n$, which are used to express the initial condition and applied input of each experiment, as later defined in (5). Other model classes, such as linear, cubic or higher order, can also be chosen. The dimensionless time θ is defined as follows:

$$\theta = 1 - \exp\left(-\frac{t}{t_c}\right) \tag{3}$$

where t_c characterizes the slowest dynamics of the plant. The time-varying parametric function $\beta_q(\theta)$ is defined with an orthogonal basis of $R+1$ Shifted Legendre Polynomials (SLP), $P_r(\cdot)$ with $r = 0, \ldots, R$, where r is the order of the SLP as:

$$\beta_q(\theta) = \gamma_{q,1} P_0(\theta) + \gamma_{q,2} P_1(\theta) + \cdots + \gamma_{q,R+1} P_R(\theta) \tag{4}$$

in which $q = 0, i, ij, ii$ for $i = 1, 2, \ldots, n$, $j > i$ and $\gamma_{q,r+1}$ are scalar coefficients.

Remark 1. The parameters of the DRSM model (2) are the SLP highest order R, the characteristic time t_c, and the SLP coefficients $\gamma_{q,r}$ for $r = 0, \ldots, R$ and $q = 0, i, ij, ii$.

We define a new vector variable $\eta = [y_0, u]$, where y_0 is the initial value of the output and u is the applied input, assumed constant in each experiment. For a SISO control plant the considered DRSM has two factors x_1 and x_2, i.e. $n = 2$, and η is related with them by the following normalization equation:

$$\eta_i = \eta_i^c + \Delta\eta_i x_i, \qquad i = 1, 2 \tag{5}$$

where η_i^c is a centering value, and $\Delta\eta_i$ determines the size of the design domain.

Let us note that an efficient data collection strategy is to implement a series of step changes in the process input u changing in the DRSM model the second factor x_2. The initial value y_0, corresponding to the first factor x_1, is collected by the experimental measurement at the end of the previous experimentation. This is based on the assumption that the experiments are executed one after the other and in a continuous operation plant. The procedure used to estimate the DRSM parameters is described in Wang and Georgakis (2017).

2.3 DRSM-based MPC nominal model

In order to design a DRSM-based MPC algorithm, we consider a nominal process model defined as in (2), and we rewrite it in the following compact form:

$$y(t) = \tilde{H}(t, x) \tag{6}$$

We note that, by definition (2), the function $\tilde{H}(t,x)$ is differentiable in its arguments. By inverting (5), we express x_i as a function of η_i:

$$x_i = \frac{\eta_i - \bar{\eta}_i}{\Delta\eta_i} \tag{7}$$

Then, we rewrite (6) as follows:

$$y(t) = H(t, y_0, u) \tag{8}$$

Furthermore, we can derive a recursive discrete-time model from (6) considering the discretization time h as generic time t, obtaining:

$$y(h) = H(h, y_0, u) \tag{9}$$

Finally, to simplify the notation and to further highlight the recursion, we indicate the initial condition y_0 as y, the output at the end of the discretization interval $y(h)$ as y^+, and rewrite (9) as follows:

$$y^+ = H(h, y, u) \tag{10}$$

Remark 2. While the model (10) is in discrete-time form, it explicitly contains the discretization interval h as an argument, so it can be used to predict future output for an arbitrary h. This feature is exploited in the MPC formulation described in the next section to reduce the number of optimization variables by solving the Finite Horizon Optimal Control Problem (FHOCP) with gradually increasing discretization intervals.

Remark 3. When the system has multiple inputs and multiple outputs, $n_u > 1$ and $n_y > 1$, a DRSM model in the same form of (2) is defined and identified for each output separately. In such case, for each output the DRSM model has (the same) $n = n_y + n_u$ factors, which are related to the vector variable $\eta = [y_0^T,\ u^T]^T \in \mathbb{R}^{n_y + n_u}$ by the same relation (7). By stacking the n_y DRSM models of each output, the recursive model (10) is obtained.

3. DRSM-BASED MPC

3.1 Augmented model

We remark that the nominal model dynamics (10) differs from the actual plant dynamics (1) due to modeling errors and the presence of unmodeled disturbances affecting the plant.

Hence, in order to cope with the plant model-mismatch a linearly augmented version of the model (10) for offset-free tracking DRSM-based MPC algorithms is implemented according to the guidelines described in (Pannocchia et al., 2015):

$$\begin{aligned} y^+ &= H(h, y, u) + d \\ d^+ &= d \end{aligned} \tag{11}$$

where $d \in \mathbb{R}^{n_y}$ are the so-called *disturbances*.

Assumption 4. The augmented system (11) is observable, according to (Pannocchia et al., 2015, Remark 8).

3.2 State and disturbance estimation

Let $\hat{y}^*_{k-1}$ and $\hat{d}^*_{k-1}$ be the estimates of y_{k-1} and d_{k-1} obtained using the output measurements at time $k-1$. Furthermore, $\hat{y}_k$ and $\hat{d}_k$, are the predicted values of y_k and d_k that can be obtained at time k using the augmented model (11), inputs and disturbances at time $k-1$, that is:

$$\begin{aligned} \hat{y}_k &= H(h, \hat{y}^*_{k-1}, u_{k-1}) + \hat{d}^*_{k-1} \\ \hat{d}_k &= \hat{d}^*_{k-1} \end{aligned} \tag{12}$$

Hence, we define the prediction errors at time k as:

$$\varepsilon_k = y_{p,k} - \hat{y}_k \tag{13}$$

therefore, we can write the filtering relations for the augmented states as:

$$\begin{aligned} \hat{y}^*_k &= \hat{y}_k + K_x \varepsilon_k \\ \hat{d}^*_k &= \hat{d}_k + K_d \varepsilon_k \end{aligned} \tag{14}$$

where the matrices $K_x \in \mathbb{R}^{n_y \times n_y}$ and $K_d \in \mathbb{R}^{n_d \times n_y}$ are chosen to form an asymptotically stable observer, which requires K_d to be invertible (Pannocchia et al., 2015).

3.3 Target calculation

A steady-state target calculation is required at time k to compute a feasible equilibrium $(\bar{u}_k, \bar{y}_k)$ to be tracked by the optimal control problem. Considering the current disturbance estimate $\hat{d}^*_k$ the following target optimization problem is solved each iterations.

$$(\bar{u}_k, \bar{y}_k) = \arg\min_{(u,y)} \ell_{ss}(y,u) \tag{15a}$$

subject to:

$$\ell_{ss}(y,u) = \frac{1}{2}\left(\Delta y^T Q_{ss} \Delta y\right) \tag{15b}$$

$$\Delta y = y_{sp} - y \tag{15c}$$

$$y = H(h, y, u) + \hat{d}^*_k \tag{15d}$$

$$u_{\min} \le u \le u_{\max} \tag{15e}$$

$$y_{\min} \le y \le y_{\max} \tag{15f}$$

in which $Q_{ss} \in \mathbb{R}^{n_y \times n_y}$ is symmetric positive definite and u_{sp} is the set point for the controlled variables. y_{sp} is the output set-point which the target optimization problem (15) aims to track.

3.4 Optimal control problem

Next, the input sequence is computed as the solution of a non-equally-spaced horizon optimal control problem. Let us denote $\boldsymbol{y} := \{\zeta_0, \zeta_1, \ldots, \zeta_N\}$ and $\boldsymbol{u} := \{\nu_0, \nu_1, \ldots, \nu_{N-1}\}$ some generic output and input sequences of optimization variables, which are not equally-spaced over the prediction horizon. As depicted in Fig. 1, the time interval between the indices i and $i+1$ has a total length of $J_i h$, that is it contains J_i sub-intervals of size h. We define the vector $\boldsymbol{J} := \{J_0, \ldots, J_{N-1}\}$ in which the value of $J_i > 0$ specifies the number of time steps between the optimization pairs (ζ_i, ζ_{i+1}) and (ν_i, ν_{i+1}). We can see as the input ν_i remains constant in that interval, while the output $\zeta_{i,j}$ is evaluated in each sub-interval, starting with $\zeta_{i,0} = \zeta_i$, thanks to the inherent multi-rate nature of the DRSM model (11), which allows us to predict the output at an arbitrary future time.

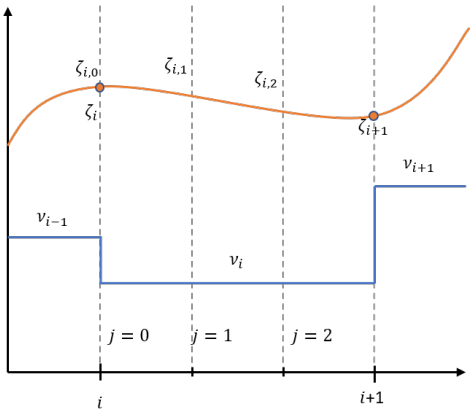

Fig. 1. Representation of subdivision of the prediction horizon between indices i and $i+1$ with the input (blue) and the output (orange). In this case $J_i = 3$.

The following finite-horizon optimal control problem (FHOCP) is solved at each decision time k:

$$(\boldsymbol{y}^\star_k, \boldsymbol{u}^\star_k) = \arg\min_{\boldsymbol{y},\boldsymbol{u}} \sum_{i=0}^{N-1} \ell(\zeta_i, \nu_i) \tag{16a}$$

subject to:

$$\ell(\zeta_i, \nu_i) = \sum_{j=0}^{J_i-1} \frac{1}{2}\left(\Delta\zeta_{i,j}^T Q \Delta\zeta_{i,j} + \Delta\nu_{i,j}^T S \Delta\nu_{i,j}\right) \tag{16b}$$

$$\zeta_0 = \hat{y}^*_k \tag{16c}$$

$$\zeta_{i+1} = H(J_i h, \nu_i, \zeta_i) + \hat{d}^*_k, \quad i = 0, \cdots, N-1 \tag{16d}$$

$$\zeta_{i,j} = H(jh, \nu_i, \zeta_i) + \hat{d}^*_k, \quad \forall i, \quad j = 0, \ldots, J_i - 1 \tag{16e}$$

$$\Delta\zeta_{i,j} = \zeta_{i,j} - \bar{y}_k \tag{16f}$$

$$\Delta\nu_{i,j} = \begin{cases} \nu_i - \nu_{i-1} & \text{if } j = 0 \\ 0 & \text{if } j \neq 0 \end{cases} \tag{16g}$$

$$u_{\min} \le \nu_i \le u_{\max} \tag{16h}$$

$$y_{\min} \le \zeta_i \le y_{\max} \tag{16i}$$

$$\zeta_N = \bar{y}_k \tag{16j}$$

where the penalty matrices $Q \in \mathbb{R}^{n_y \times n_y}$ and $S \in \mathbb{R}^{n_u \times n_u}$ are chosen symmetric positive definite. We note that the definition of $\Delta\nu_{i,j}$ implies that ν remains constant in the i-th interval. Moreover, the constraint $\zeta_N = \bar{y}_k$ is used for the feasibility of the FHOCP. Assuming that Problem (16) is feasible, the first input of the optimal sequence $\boldsymbol{u}^\star_k$ is implemented as usual in MPC.

$$u_k = \boldsymbol{u}^\star_k[0] \tag{17}$$

4. CASE STUDY: CSTR SISO SYSTEM

This section presents a case study to validate in simulation the proposed methodology, considering an experimental setup that can be realized in a continuously operating plant.

We consider an isothermal continuous-stirred tank reactor (CSTR) with the following second order reaction:

$$\mathrm{A} \xrightarrow{k_1} \mathrm{B} \quad r_1 = k_1 c_\mathrm{A}^2 \tag{18}$$

Species A is fed at the variable flowrate u with molar concentration $c_{\mathrm{A}0}$, while the product is the species B. The volume of the reactor V_r is constant. The dynamics of the system is defined as follows:

$$\frac{dy}{dt} = -\frac{u}{V_r}(c_{A0} - y) - k_1 y^2 \tag{19}$$

It is a SISO system in which the feed flow u is the input and the reagent concentration in the reactor y is the measurement output. We set the following constraints on y and u:

$$y_{\min} \leq y \leq y_{\max}$$
$$u_{\min} \leq u \leq u_{\max}$$

The constant parameters are reported in Table 1.

Table 1. Parameters and constraints of the isothermal CSTR system.

Quantity	Value	Unit
k_1	0.05	$L/(\text{mol min})$
V_r	2800	L
c_{A0}	1.5	mol/L
$u_{\max}$	150	L/min
$u_{\min}$	50	L/min
u^c	100	L/min
Δu	50	L/min
$y_{\max}$	1.5	mol/L
$y_{\min}$	0	mol/L
y^c	0.75	mol/L
Δy	0.75	mol/L

4.1 DRSM model: experiments and estimation

We define the two factors x_1 and x_2, that are used for the development of the DRSM model. The initial concentration y_0 and the input u are related with the factor with the following equations:

$$y_0 = y^c + x_1 \Delta y \tag{20a}$$
$$u = u^c + x_2 \Delta u \tag{20b}$$

We consider a set of $M = 16$ experiments in which the value of factor x_2 is defined as in Table 2, while the next value of factor x_1 is updated with the last output measurement of previous simulation. Moreover, we consider three repetitions of the experiments with the following noise:

$$\frac{v_p}{h_p(\xi_p)} \sim N(0,\sigma) \text{ with } \sigma = 0.02 \tag{21}$$

Note that $\sigma = 0.02$ is called $\pm 5\%$ because in a normal distribution 2.5σ cover almost all possible number with $N(0,\sigma)$.

For each of the values of the factors, we simulated the dynamic system (19), and we collected the data of the output $y(t)$ at the following time instants $t = h, 2h, 3h, \ldots, K_m h$. We set $h = 1$ min, and the total number of measurements in the m-th experiment $K_m = 100$. We remark that for estimating a DRSM model of the same accuracy, it is typically sufficient to have 12-15 data points for each experiment. This aspect can be relevant when the related measurements are expensive to be obtained, e.g. as in the case of product quality.

We estimated the DRSM model from the collected data.The model parameters t_c and R are estimated in the following intervals $t_c \in [10,\ 20]$ min, $R \in [1,\ 5]$. The DRSM identification software is developed in `Python 3.x` based on the methodology described in Section 2.The function `LassoLarsIC` of `scikit-learn` library is used for the Lasso regression. For the ordinary least squares the function `OLS` of the `statsmodels` library is also used.

The $t_c = 15.5$ min and the order of the SLP $R = 2$. The estimated model coefficients are reported in Table 3. The other parameters $\gamma_{0,3}$, $\gamma_{1,3}$, $\gamma_{2,3}$, $\gamma_{12,2}$, $\gamma_{11,1}$, $\gamma_{11,2}$, $\gamma_{11,3}$ are obtained equal to 0. It is interesting to observe how the use of a sparse regression allows obtaining a model with a small number of parameters. This makes the model simple and easy to use in an optimization framework. We represent in Figure 2 the plot of experiments for the identification of the DRSM. We note the presence of a quadratic model due to the nonlinear kinetic.

The DRSM requires performing a Lack-of-Fit (LoF) test to check the model validity as described in Wang and Georgakis (2017).We compare the minimized sum of square obtained in the final regression against the data against squares of the pure error estimated from the replicate runsusing an F-test. We consider 3 replicate runs of the experiments with $M_C = 3$. We obtained $F_0 = 0.41$ and $F_{\alpha,DoF_{LoF},DoF_{PE}} = 1.19$, so $F_0 < F_{\alpha,DoF_{LoF},DoF_{PE}}$; with a corresponding p-value of 0.999 and the LoF is not significant. This implies that the DRSM model has succeeded in representing all the non-random information in the data.

4.2 MPC closed-loop simulations

Three controllers are compared:

- MPC1: nominal NMPC, in which we assume perfect knowledge of the true process model (19)
- MPC2: linear DRSM-MPC with a linear DRSM model estimated
- MPC3: nonlinear DRSM-NMPC with the quadratic DRSM model estimated in this section.

In all controllers the following cost weights are used: $Q_{ss} = 1$, $Q = 1$, $S = 5 \cdot 10^{-4}$. The estimator is a deadbeat Kalman filter with $K_x = 0$, $K_d = 1$. The sampling time used for control is $h = 1$ min, and the prediction horizon 50 min is unevenly discretized using the following vector $\boldsymbol{J}$:

$$\boldsymbol{J} := \{1,1,1,1,1,1,1,1,1,1,4,4,4,4,4,5,5,10\}$$

A software implemented in Python with `CasADi` library and `IPOPT` optimizer has been used to solve this constrained nonlinear optimization problem. The simulation was performed on a PC with CPU Intel i5 7200u and the resulting computational cost is not an issue for this problem. The effectiveness of the DRSM-NPMC controller has been evaluated by introducing set-point changes according to the sequence reported in Table 4.

The performance of the designed controllers to a series of step changes on the set-point for output y are shown in Figure 3; the changes in u caused by the controllers are depicted in the upper figure, while the output is reported in the bottom figure. We note that in general, the set-point tracking performance is satisfactory. When the set-point is increased, at $t = 200$ min, the targeted concentration is reached in approximately 35 min. When the set-point is decreased, at $t = 400$ min, the dynamic response of the system is slower and the new set-point is reached in approximately 45 min. Table 5 reports the overall cost comparison of MPC1, MPC2 and MPC3. We observe the performance of MPC2 is slightly worse than that of controller MPC3, MPC1 is an idealistic controller, while MPC3 is better than MPC2. In Figure 4 MPC3 is used to control the system, where a uniform white noise, as in equation (21), is added to the output to emulate measurement noise. We note that the system is well controlled despite the noise.

Table 2. Experiments design

Expr	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
x_2	-1	-1	1	1	0	0	-1	1	0.75	0.75	-1	-1	-0.125	0	0	0
Start time	0	100	200	300	400	500	600	700	800	900	1000	1100	1200	1300	1400	1500

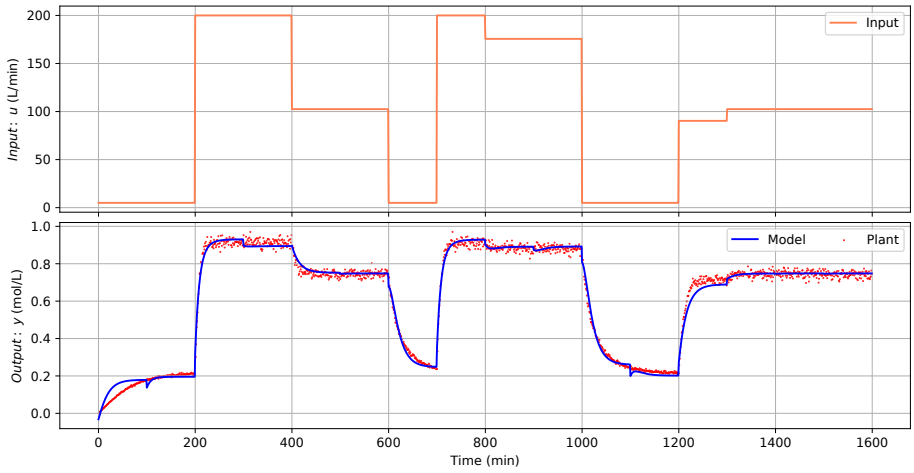

Fig. 2. Sixteen experiments (every 100 min) for the estimation of the DRSM of the CSTR. The red line represents noisy output data; the blue line represents the output of the estimated DRSM model; the orange line represents the input.

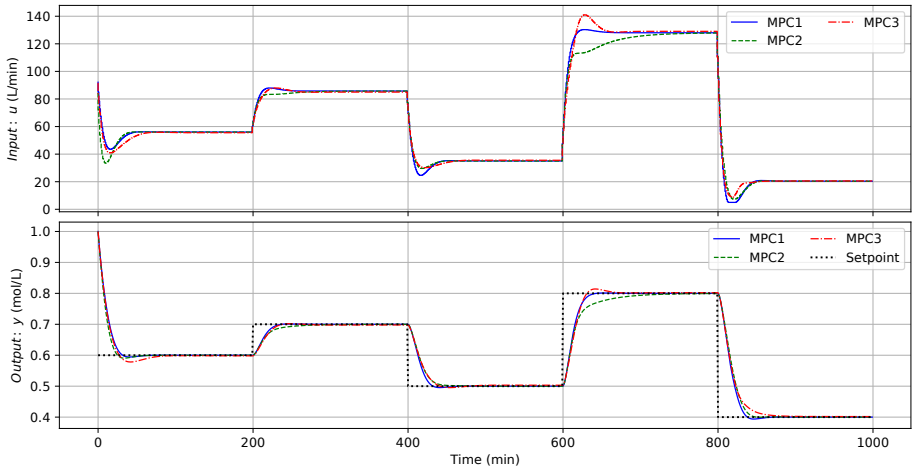

Fig. 3. Closed-loop results input (top) and output (bottom).

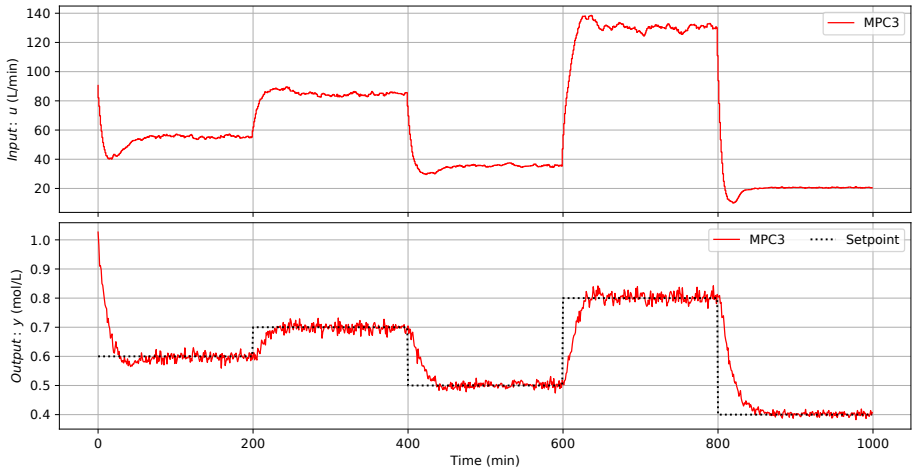

Fig. 4. Closed-loop results input (top) and output (bottom).

Table 3. γ parameters for the CSTR with confidence interval ($\alpha = 0.05$).

	Value
$\gamma_{0,1}$	0.6365 ± 0.0043
$\gamma_{0,2}$	0.1072 ± 0.0052
$\gamma_{1,1}$	0.2256 ± 0.0042
$\gamma_{1,2}$	-0.2160 ± 0.0051
$\gamma_{2,1}$	0.2015 ± 0.0034
$\gamma_{2,2}$	0.1439 ± 0.0040
$\gamma_{12,1}$	-0.0913 ± 0.0040
$\gamma_{12,3}$	0.0562 ± 0.0054
$\gamma_{22,1}$	-0.0555 ± 0.0057
$\gamma_{22,2}$	-0.0743 ± 0.0078
$\gamma_{22,3}$	-0.0438 ± 0.0052

Table 4. Output set-point

time (min)	y_{sp}
0-200	0.6
200-400	0.7
400-600	0.5
600-800	0.8
800-1000	0.4

Table 5. Overall cost comparison

Controller	Overall cost
MPC1	2.62
MPC2	2.70
MPC3	2.68

5. CONCLUSIONS

In this work, a data-driven dynamic model, the DRSM, has been used to model a continuous nonlinear process. The identified DRSM model is a sparse model that shows promising performance in modeling complex nonlinear dynamics, leaving a simple structure to be applied in the context of optimization and control. Then, the identified DRSM model is used for controlling a nonlinear CSTR process in an NMPC framework. From the DRSM model, a recursive nonlinear model is obtained directly choosing the discretization interval of the DRSM. This formulation proved to converge to plant set-point despite model uncertainty and white noise.

Further studies will focus on the design of the experiments, introducing experiments with different time lengths and optimal DOE design, especially for the MIMO case. Moreover, DRSM based eMPC will be analyzed; we will investigate and prove the convergence conditions and KKT matching upon convergence. Moreover, a future investigation will be focused on DRSM based economic MPC. Economic MPC (eMPC) is one of the most studied solution methods to overcome the hierarchical separation between economic optimization and control in the process industries (Rawlings et al., 2012). To deal with the plant-model mismatch Offset-Free eMPC (OF-eMPC) algorithms have been proposed in the literature (Vaccari and Pannocchia, 2016; Faulwasser and Pannocchia, 2019) with the integration of modifier adaptation (MA) (Marchetti et al., 2009).

REFERENCES

da Cruz Meleiro, L.A., Von Zuben, F.J., and Maciel Filho, R. (2009). Constructive learning neural network applied to identification and control of a fuel-ethanol fermentation process. *Engineering Applications of Artificial Intelligence*, 22(2), 201–215.

Darby, M.L. and Nikolaou, M. (2012). Mpc: Current practice and challenges. *Control Engineering Practice*, 20(4), 328–342.

Faulwasser, T. and Pannocchia, G. (2019). Towards a unifying framework blending RTO and economic MPC. *Industrial & Engineering Chemistry Research*, 58, 13583–13598.

Georgakis, C. (2013). Design of dynamic experiments: A data-driven methodology for the optimization of time-varying processes. *Industrial & Engineering Chemistry Research*, 52(35), 12369–12382.

Hong, T., Zhang, J., Morris, A., Martin, E., and Karim, M. (1996). Neural based predictive control of a multivariable microalgae fermentation. In *1996 IEEE International Conference on Systems, Man and Cybernetics. Information Intelligence and Systems (Cat. No. 96CH35929)*, volume 1, 345–350. IEEE.

Klebanov, N. and Georgakis, C. (2016). Dynamic response surface models: a data-driven approach for the analysis of time-varying process outputs. *Industrial & Engineering Chemistry Research*, 55(14), 4022–4034.

Marchetti, A.G., Chachuat, B., and Bonvin, D. (2009). Modifier-adaptation methodology for real-time optimization. *Industrial & Engineering Chemistry Research*, 48(13), 6022–6033.

Montgomery, D.C. (2017). *Design and analysis of experiments*. John wiley & sons.

Pannocchia, G., Gabiccini, M., and Artoni, A. (2015). Offset-free MPC explained: Novelties, subtleties, and applications. *IFAC-PapersOnLine*, 48(23), 342–351.

Rawlings, J.B., Angeli, D., and Bates, C.N. (2012). Fundamentals of economic model predictive control. In *51st IEEE Conference on Decision and Control*, 3851–3861.

Tulleken, H.J. (1990). Generalized binary noise test-signal concept for improved identification-experiment design. *Automatica*, 26(1), 37–49.

Vaccari, M. and Pannocchia, G. (2016). A modifier-adaptation strategy towards offset-free economic MPC. *Processes*, 5(1), 2.

van Wingerden, J.W. and Verhaegen, M. (2009). Closed-loop subspace identification of hammerstein-wiener models. In *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, 3637–3642. IEEE.

Wang, Z. and Georgakis, C. (2017). New dynamic response surface methodology for modeling nonlinear processes over semi-infinite time horizons. *Industrial & Engineering Chemistry Research*, 56(38), 10770–10782.

Wang, Z. and Georgakis, C. (2019). Identification of hammerstein-weiner models for nonlinear mpc from infrequent measurements in batch processes. *Journal of Process Control*, 82, 58–69.