Model-Plant Mismatch Detection for Cross-directional Processes

Abstract

This paper presents a two-component framework to detect model-plant mismatch (MPM) in cross-directional (CD) processes on paper machines under model-predictive control. First, routine operating data is used for system identification in closed loop; second, a one-class support vector machine (SVM) is trained to predict MPM. The iterative identification method alternates between identifying the finite impulse response coefficients of the spatial and temporal models. It converges, and the parameter estimates are asymptotically consistent. Coefficient estimates drawn from normal operation are used to train a one-class SVM, which then detects model-plant mismatch in subsequent routine operation. This approach applies to routine operating data without requiring external excitations. It can also distinguish mismatches in the process model from changes in the noise model. Examples of CD processes on paper machines are provided to verify the effectiveness of both components.

Keywords: Routine closed-loop identification, Support vector machine, High-order ARX, Cross-directional processes, Process monitoring.

1. Introduction

Model-predictive control (MPC) is used in a wide range of industrial processes, from refining petroleum and processing metal to making paper (Qin and Badgwell (2003)). MPC effectively handles complex multivariable processes. Its actuation commands optimize user-defined objectives while meeting physical constraints on inputs, outputs, and states. Underlying all these impressive features is a model of the process being controlled: an accurate model is essential. Indeed, most observed deteriorations in MPC performance can be traced to degradations in model quality (Botelho et al. (2016)). There is a clear need for an automated and highly-reliable scheme for monitoring the process model accuracy.

The discrepancy between a plant's true dynamics and the model used in MPC is known as model-plant mismatch (MPM). Incorrect process models can lead to suboptimal control actions or even closed-loop instability. Hence, detecting and eliminating MPM is priority. Despite recent progress (Wang et al. (2016); Julien et al. (2004)), two key issues remain: distinguishing between MPM and noise model changes (Sun et al. (2013); Botelho et al. (2016)), and the role of external excitations (Badwe et al. (2010, 2009)). First, changes in a process's noise model have some symptoms in common with MPM, such as inflating the variance of process variables. However, an accurate process model is more critical for the safe operation of the system. Thus an ideal MPM detection approach would be robust to changes in the noise model, and not mistake them for MPM (Botelho et al. (2016)). Separating these two effects is rather difficult. Approaches built on variance-based metrics (e.g., Harris (1989)), are at a disadvantage. Second, most current approaches to identify MPM directly rely on external excitations, such as dither signals or setpoint changes. Such interventions perturb the system and compromise its performance. It is clearly preferable to detect MPM during routine operation. In Lu et al. (2017b, 2020), we proposed a one-class SVM to detect errors in the plant model for low-dimensional systems. A sequence of measurements acquired when the system is operating with an accurate model is used to construct clusters of process and noise models. Then, the SVM compares process and noise models derived from subsequent routine operating data with their respective normal clusters. This approach effectively detects MPM and is robust to noise model changes. However, implementing it in spatially-distributed systems remains a challenging problem.

Our research is motivated by the control of paper machines. A typical paper machine transforms a slurry of water and wood fibres into a sheet of paper. An array of actuators at the headbox adjusts the properties of pulp across the sheet and an array of sensors at the far end of the machine measures the paper properties of interest. The control objective is to manipulate the actuators to achieve desirable product properties Morales and Heath (2011). The developing paper sheet moves in the machine direction (MD); the cross-direction (CD) is perpendicular to this. The CD process is large-scale and its process characteristics vary over time for a host of reasons, with user-commanded changes in the product properties being only the most obvious. Practitioners need to maintain high-quality CD models.

In this work, as in most industrial settings, we assume that the spatial and temporal responses are the same for all CD actuators, and also that these responses are separable. These conditions allow the CD process to be described by a high-dimensional Hammerstein model with a static (spatial) part concatenated with a dynamic part (Narendra and Gallman (1966)). Compared with Lu et al. (2017b, 2020), in this work, we propose a novel system identification method for efficiently identifying the complex CD process models. We also provide a theoretical guarantee on the performance of the proposed closed-loop identification, and this addresses the main challenge in developing the CD MPM detection algorithm.

This paper is organized as follows. A description of the closed-loop CD process is given in Section 2, together with an overview of our MPM detection scheme. Section 3 is devoted to the development of a routine CD closed-loop identification method. Section 4 details the application of a one-class SVM to MPM detection. Two illustrative simulations are provided in Section 5. Section 6 provides our conclusions.

Notation: Normal-weight symbols denote scalar-valued quantities. Bold font is reserved for vectors and matrices, as in $\mathbf{v}$ and $\mathbf{G}$. Signals like $s(t)$ have discrete inputs ($t = 0, 1, 2, \ldots$), and may be acted upon by the one-step shift operator q^{-1}; the corresponding long-term expectation is $\bar{\mathbb{E}}[s(t)] = \lim_{N\to\infty} \frac{1}{N}\sum_{t=1}^{N} \mathbb{E}[s(t)]$. We abbreviate "with probability one" as "w.p.1", and use $\|\cdot\|$ for the Euclidean norm for vectors and the Frobenius norm for matrices. For the direct sum of two sets, we write $\oplus$.

2. Preliminaries

2.1. CD process model

The following single-array CD process model is widely employed in paper machine control:

$$\mathbf{y}(t) = g(q^{-1})\mathbf{G}\mathbf{u}(t-d) + \mathbf{v}(t), \quad t \in \mathbb{Z}. \tag{1}$$

Here $\mathbf{y}(t) \in \mathbb{R}^m$ and $\mathbf{u}(t) \in \mathbb{R}^n$ represent the measured output signal (controlled variable, CV) and input signal (manipulated variable, MV), respectively. Note that the steady-state components for the MD process have been removed from the input and output in (1). Assume that $k \geq 1$ equally-spaced measurements are taken for each actuator, so that $m = kn$. In (1), $\mathbf{v}(t) \in \mathbb{R}^m$ is a colored noise vector, $d \in \mathbb{N}$ is a time delay in samples, and $g(q^{-1})$ is a first-order scalar filter modeling the process's temporal dynamics. Typically

$$g(q^{-1}) = \frac{h}{1 - fq^{-1}}, \tag{2}$$

where h is a constant gain and $f = \exp(-T_s/T_p)$, with time constant T_p and sampling interval T_s. Let $\boldsymbol{\theta}_T = [h\ f]^T \in \Omega_T$ with Ω_T denoting a compact set of feasible values. Denote $\boldsymbol{\theta}_T^\circ$ as the true temporal parameter. The constant matrix $\mathbf{G} \in \mathbb{R}^{m \times n}$ in (1) captures the steady-state spatial responses of the actuator array. Each column of $\mathbf{G}$ is a shifted version of the same symmetric spatial impulse response, in which the coordinate x determines

$$\begin{aligned} b(x) &= \frac{1}{2}\left[\varphi\left(\frac{x}{\xi} + \beta\right) + \varphi\left(\frac{x}{\xi} - \beta\right)\right], \text{ with} && (3)\\ \varphi(r) &= \gamma e^{-\alpha r^2}\cos(\pi r). && (4) \end{aligned}$$

The scalar parameters describing attenuation α, divergence β, gain γ, and width ξ are key elements of the system model. In principle, the elements of $\mathbf{G}$ should be $G_{ij} = b(i - \bar{c}_j)$, where $\bar{c}_j \in [1 : m]$ is the spatial response center of actuator j. (In the special case where $k = 1$, so $m = n$ and $\bar{c}_j = j$, matrix $\mathbf{G}$ is Toeplitz: $G_{ij} = b(i - j)$.) In practice, the exponential decay in (4) is so rapid that one can choose an integer order $p \geq 1$ and define $\mathbf{G}$ using the truncated function $b_p(x)$ that equals $b(x)$ whenever $|x| < p$, but has $b_p(x) = 0$ for $|x| \geq p$.

Now the matrix $\mathbf{G}$ has complicated nonlinear dependence on the scalar shape parameters $\alpha, \beta, \gamma, \xi$, but its dependence on the scalar values $b(0), b(1), \ldots, b(p-1)$ used to define its columns is linear. One can thus express $\mathbf{G} = \sum_{k=1}^{p} c_k \mathbf{E}_k$, $c_k = b(k-1)$, for suitable symmetric "basis matrices" $\mathbf{E}_k$ in which every entry is 0 or 1. One can use this decomposition to identify $\mathbf{G}$ by estimating the p parameters $c_1, \ldots, c_p$ instead of $\alpha, \beta, \gamma, \xi$. Note that the temporal gain h in (2) and the spatial gain γ in (4) enter the true model (1) only through their product $h\gamma$, so no generality is lost in setting $h = 1$ in practical problems.

2.2. CD noise model

A common approach to model CD noise is to choose a diagonal noise model while forcing the innovation sequence to have non-diagonal covariance matrix (Gorinevsky and Gheorghe (2003); Rigopoulos et al. (1997)), to capture the spatial correlation of colored noise $\mathbf{v}(t)$. The temporal correlation is modeled by a filter shared by all output channels[1]. In this manner,

$$\mathbf{v}(t) = H(q^{-1})\mathbf{e}(t), \tag{5}$$

where $H(q^{-1})$ is a scalar monic transfer function that is stable and inversely stable and $\mathbf{e}(t) \in \mathbb{R}^m$ is a zero-mean Gaussian white noise vector with covariance $E[\mathbf{e}(t)\mathbf{e}(t-s)^T] = \delta_{s,t}\boldsymbol{\Sigma} \in \mathbb{R}^{m \times m}$. Note that here $\boldsymbol{\Sigma}$ can be non-diagonal to represent the spatial correlations

[1] This assumption can be easily relaxed to allow for different noise models in each output channel and the identification method presented in this paper is still applicable with slight modifications.

of CD measurement noise. In general, it is difficult to acquire prior information about the true structure of $H(q^{-1})$ in (5). For closed-loop identification, especially the direct identification approach, incorrect specification of the noise model leads to bias in the process model estimate (Ljung (1999)). We propose a closed-loop identification method to address this issue in Section 3.

2.3. High-order ARX approximation of CD process model with FIR noise representation

A stable linear transfer function can be approximated arbitrarily well by a high-order FIR model (Ljung (1999); Zhu (2002)). Thus one can represent the CD process model (1)–(5) with a sufficiently high-order ARX structure, to avoid the bias issue in direct identification above. Specifically, we equivalently rewrite the CD model as follows (see Appendix A for a proof):

$$A(q^{-1}, \mathbf{a})\mathbf{y}(t) = B(q^{-1}, \mathbf{b}) \sum_{k=1}^{p} c_k \mathbf{E}_k \mathbf{u}(t-d) + \mathbf{e}(t), \tag{6}$$

where $A(q^{-1}, \mathbf{a}) = 1/H(q^{-1})$ is a monic scalar polynomial showing the FIR representation of the inverse of the noise model. In detail, $A(q^{-1}, \mathbf{a}) = 1 + \sum_{j=1}^{n_a} a_j q^{-j}, \mathbf{a} = [a_1 \ \dots \ a_{n_a}]^T$. Similarly, $B(q^{-1}, \mathbf{b}) = A(q^{-1}, \mathbf{a})g(q^{-1})$, $B(q^{-1}, \mathbf{b}) = \sum_{j=0}^{n_b} b_j q^{-j}, \mathbf{b} = [b_0 \ \cdots \ b_{n_b}]^T$. Define the parameter vector, $\boldsymbol{\theta}^T = [\mathbf{a}^T \ \mathbf{b}^T \ \mathbf{c}^T] \in \mathbb{R}^{n_a+n_b+1+p}$. Here, due to our stability assumptions on $A(q^{-1}, \mathbf{a})$ and $B(q^{-1}, \mathbf{b})$, finitely truncated expansions can approximate these polynomials. (See Ljung and Wahlberg (1992); Zhu and Hjalmarsson (2016).) The CD process model is transformed into an ARX-Hammerstein structure. We have the predictor form of (6) as

$$\widehat{\mathbf{y}}(t|t-1) = \begin{bmatrix} \boldsymbol{\psi}_y(t) & \boldsymbol{\psi}_{\bar{u}}(t-d) \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \mathbf{Cb} \end{bmatrix}, \tag{7}$$

where $\mathbf{C} = diag\{\mathbf{c}, \mathbf{c}, \dots, \mathbf{c}\}$, $\bar{\mathbf{u}}^k(t) = \mathbf{E}_k \mathbf{u}(t) \in \mathbb{R}^m$, $k = 1, \dots, p$, and

$$\boldsymbol{\psi}_y(t) = \begin{bmatrix} \boldsymbol{\psi}_{y_1}(t) \\ \vdots \\ \boldsymbol{\psi}_{y_m}(t) \end{bmatrix}, \quad \boldsymbol{\psi}_{\bar{u}}(t) = \begin{bmatrix} \boldsymbol{\psi}_{\bar{u}_1}(t) \\ \vdots \\ \boldsymbol{\psi}_{\bar{u}_m}(t) \end{bmatrix},$$

with

$$\begin{aligned} \boldsymbol{\psi}_{y_i}(t) &= [-y_i(t-1) \quad -y_i(t-2) \quad \cdots \quad -y_i(t-n_a)], \\ \boldsymbol{\psi}_{\bar{u}_i}(t) &= \left[\bar{u}_i^1(t) \ \dots \ \bar{u}_i^p(t) | \dots | \bar{u}_i^1(t-n_b) \ \dots \ \bar{u}_i^p(t-n_b)\right]. \end{aligned}$$

2.4. The presence of feedback

The version of CD MPC widely used in industry is based on quadratic programming (Fan (2003)). Typical constraints in the MPC algorithm include actuator limits, bounds on the changes between successive control actions, constraints on the averaged actuator profile in an array, and bending limits. According to Bemporad et al. (2002), when some of these constraints are active and varying, the MPC will display a piecewise linear or even nonlinear behavior. Hence, the manipulated variable at time t emerges from a feedback function $\mathbf{k}$ via

$$\mathbf{u}(t) = \mathbf{k}(\mathbf{u}^{t-1}, \mathbf{y}^t, t), \tag{8}$$

where $\mathbf{u}^{t-1} = \{\mathbf{u}(1), \ldots, \mathbf{u}(t-1)\}$ and $\mathbf{y}^t$ is defined in an analogous way. In the absence of excitations or setpoint changes, nonparametric identification methods often yield the controller inverse as a process model estimate (Söderström and Stoica (1988)). One remedy is to exploit prior knowledge (or a previous estimate) of the time-delay in the model structure in (6) when performing the high-order ARX identification. Note that our method essentially detects the discrepancy of the test data against training (or normal) data, rather than precisely estimating process models. Therefore, knowledge of the true time-delay is not necessary.

Another important concern is closed-loop identifiability. As shown in Söderström and Stoica (1988); Gevers et al. (2009); Shardt and Huang (2011), for linear feedback control, high-order regulators and larger time-delay in the process generally enhance the informativeness of closed-loop data. The specific relationships among these factors have been fully investigated in these references. However, as commented in Ljung (1999) (p. 432), time-varying or nonlinear regulators in (8) are usually enough to guarantee the informativeness of routine closed-loop data. The informativeness of closed-loop has been a bottleneck for closed-loop identification. For the CD process, the informativeness can be relatively easy to satisfy, given the complex constraints in CD MPC (any active constraint can render the MPC nonlinear and thus increase the informativeness). Moreover, complex operating conditions, such as disturbances and boundary effects, can enhance informativeness.

2.5. Model-plant Mismatch Detection

The lack of excitation in routine operating data typically produces parameter estimates with large variance. This can make it difficult to distinguish predictable inaccuracy from real MPM. One solution is to construct a boundary around the true model that captures the range of uncertainties attributable to model estimation (Lu et al. (2017b)). Any model that falls inside this boundary will be regarded as normal; models that fall outside are diagnosed with MPM. Fig. 1 illustrates the idea. Each point in the figure represents the FIR representation of a model. The models in circles are obtained from training data. The SVM they define determines the boundary of a benchmarking cluster. The FIR representations of models from the testing data are shown as crosses: those that lie inside the cluster are considered normal, and those that lie outside are considered mismatched.

For training, we start with an interval of routine operating data with satisfactory control performance. (Periods immediately following a closed-loop experiment are likely to involve accurate models, for example.) Consecutive closed-loop identifications are performed using a moving window, and the resulting collections of FIR coefficients provide the training data for a one-class SVM. We apply the same moving-window identification methods used in training to data collected during testing intervals, and then use the SVM to assess whether the resulting model estimates lie inside or outside the benchmarking cluster.

For the full sheet-making problem, we apply the techniques above separately to each of the temporal, spatial, and noise models. We prefer to use FIR structures to represent process and noise models to synthesize the effects of all parametric mismatches into a single overall metric. This becomes particularly important when the original model has high order, since it is possible that a single large parametric mismatch may not obviously impact the overall behavior of a model. In this sense, the FIR form is a more fundamental characterization of a given model.

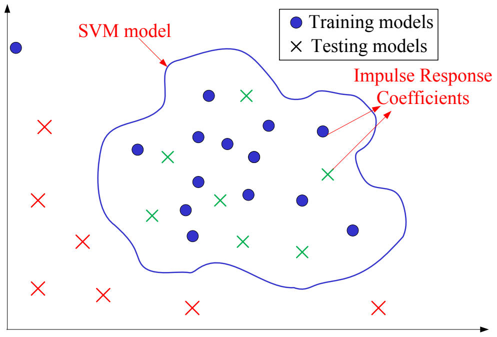

Figure 1: Illustration of the MPM detection idea. Here the training and testing models refer to the process model estimates from training and test data sets Lu et al. (2020).

3. Routine CD closed-loop Identification

This section presents a novel approach to CD identification that produces convergent and consistent estimates from routine closed-loop data. The basic techniques are similar to separable least-squares (Golub and Pereyra (2003)), alternately identifying the spatial model $\mathbf{G}_0$ and the temporal model $\{A_0(q^{-1}),\ B_0(q^{-1})\}$, until the parameters converge.

3.1. Parameter Estimation

Consider a set of input-output data generated according to (6) under the controller (8),

$$\mathbf{Z}^N = \{\mathbf{y}(1), \mathbf{u}(1), \ldots, \mathbf{y}(N), \mathbf{u}(N)\}. \tag{9}$$

Stack all the parameters to be estimated into $\boldsymbol{\theta}^T = [\mathbf{a}^T\ \mathbf{b}^T\ \mathbf{c}^T] \in \mathbb{R}^{n_a+n_b+1+q}$, and define the loss function in terms of the prediction error as follows:

$$V_N(\boldsymbol{\theta}) = \frac{1}{N}\sum_{t=1}^{N} \boldsymbol{\varepsilon}^T(t,\boldsymbol{\theta})\boldsymbol{\varepsilon}(t,\boldsymbol{\theta}), \quad \text{where} \quad \boldsymbol{\varepsilon}(t,\boldsymbol{\theta}) = \mathbf{y}(t) - \widehat{\mathbf{y}}(t|t-1) \in \mathbb{R}^m. \tag{10}$$

The optimal parameter estimate $\widehat{\boldsymbol{\theta}}_N$ is obtained by

$$\widehat{\boldsymbol{\theta}}_N = \arg\min_{\boldsymbol{\theta}\in\Omega} V_N(\boldsymbol{\theta}), \tag{11}$$

where $\Omega = \Omega_a \oplus \Omega_b \oplus \Omega_c$ is the parameter domain made up of compact convex sets Ω_a, Ω_b, and Ω_c, respectively containing $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$. The product term $\mathbf{Cb}$ in (7) makes the optimization problem (11) nonconvex. However, the simple product form of this problematic term suggests a separable least-squares approach. Fixing either $\mathbf{C}$ or $\mathbf{b}$ and optimizing over the other is convex. As we will show, alternating between these steps leads to a convergent iterative identification scheme. We must account for nonuniqueness in solving (11). Assuming $k \neq 0$, all the pairs $(\mathbf{b}/k, k\mathbf{c})$ describe the same model. Imposing a suitable normalization (Bai and Li (2004)) leads to the algorithm shown in Table 1. As we show in Theorem 1 below, this iterative scheme converges to stationary points.

Table 1: The implementation of routine CD closed-loop iterative identification

```
Algorithm for routine CD closed-loop identification
Input:  Set $\widehat{\mathbf{a}}^0 \leftarrow \mathbf{a}^i$, $\widehat{\mathbf{b}}^0 \leftarrow \mathbf{b}^i$ and $\widehat{\mathbf{c}}^0 \leftarrow \mathbf{c}^i$. $K \leftarrow$ maximum iteration number.
Loop:   for $k = 1, \ldots, K$, do
    1:  Fix the spatial parameter $\widehat{\mathbf{c}}^{k-1}$, and estimate the parameters of the high-order ARX part in (6) by solving this least-squares problem:
            $\{\widehat{\mathbf{a}}^k, \widehat{\mathbf{b}}^k\} = \arg \min_{\boldsymbol{a}\in\Omega_a, \boldsymbol{b}\in\Omega_b} V_N(\mathbf{a}, \mathbf{b}, \widehat{\mathbf{c}}^{k-1}).$        (12)
    2:  Normalize $\widehat{\mathbf{b}}^k$ as follows to address the non-identifiability
            $\rho_k = \text{sign}\left(\widehat{\mathbf{b}}^k(1)\right), \quad \widehat{\mathbf{b}}^k = \rho_k \frac{\widehat{\mathbf{b}}^k}{\|\widehat{\mathbf{b}}^k\|}.$        (13)
    3:  Fix the temporal parameter $\{\widehat{\mathbf{a}}^k, \widehat{\mathbf{b}}^k\}$, and estimate the spatial parameter in (6) by solving this nonlinear least-squares problem:
            $\widehat{\mathbf{c}}^k = \arg \min_{\mathbf{c}\in\Omega_c} V_N(\widehat{\mathbf{a}}^k, \widehat{\mathbf{b}}^k, \mathbf{c}).$        (14)
End for
    4:  Let $\widehat{\mathbf{a}} \leftarrow \widehat{\mathbf{a}}^K$, $\widehat{\mathbf{b}} \leftarrow \widehat{\mathbf{b}}^K$, $\widehat{\mathbf{c}} \leftarrow \widehat{\mathbf{c}}^K$ and denote $\widehat{\boldsymbol{\theta}}_N = [\widehat{\mathbf{a}}^T \ \widehat{\mathbf{b}}^T \ \widehat{\mathbf{c}}^T]^T$. Filter the input-output data
            $\widetilde{\mathbf{y}}(t) = A(q^{-1}, \widehat{\mathbf{a}})\mathbf{y}(t),$
            $\widetilde{\mathbf{u}}(t) = A(q^{-1}, \widehat{\mathbf{a}})\mathbf{G}(\widehat{\mathbf{c}})\mathbf{u}(t).$        (15)
    5:  Estimate temporal model $g(z^{-1}, \boldsymbol{\theta}_T)$ with $\widetilde{\mathbf{y}}(t)$, $\widetilde{\mathbf{u}}(t)$ by the following multi-experiment output-error identification:
            $\widetilde{\mathbf{y}}(t) = g(q^{-1}, \boldsymbol{\theta}_T)\widetilde{\mathbf{u}}(t-d) + \mathbf{e}(t).$        (16)
    6:  Denote $\widehat{\boldsymbol{\theta}}_T = [\widehat{h} \ \widehat{f}]^T$. Re-scale $\widehat{\mathbf{c}}$ as follows:
            $\widehat{\mathbf{c}} \leftarrow \widehat{\mathbf{c}}\widehat{f}/(1-\widehat{g}).$        (17)
    7:  Estimate the spatial parameter $\boldsymbol{\theta}_S$ in (4) via simple nonlinear least-squares. Call the result $\widehat{\boldsymbol{\theta}}_S$.
Output: Return the parameter estimates $\widehat{\boldsymbol{\theta}}_S$, $\widehat{\boldsymbol{\theta}}_T$, $\widehat{\mathbf{a}}$ and the noise covariance.
```

3.2. Convergence and consistency analysis

Assumption 1. The input-output data $\mathbf{Z}^N$ is bounded and generated according to the stable closed-loop system (6) with (8), where $N \gg n_a + n_b$ and $\mathbf{e}(t)$ is Gaussian white noise vector. In addition, (6) is uniformly stable for each $\boldsymbol{\theta}$ in Ω.

Assumption 2. The polynomials $A(q^{-1}, \mathbf{a})$ and $B(q^{-1}, \mathbf{b})$ are coprime for each $\boldsymbol{\theta}$ in the parameter set Ω. Also, some vector $\boldsymbol{\theta}^{\circ}$ in Ω corresponds to the true system. In particular, the orders n_a, n_b, and p in (6) are compatible with those of the true system.

Assumption 3. The closed-loop data $\mathbf{Z}^N$ are informative enough for the relevant closed-loop identification. In particular, we have both (a) and (b) below.

(a) The closed-loop input data $\mathbf{u}^N$ is strongly persistently exciting with orders at least n_b over the basis matrices $\mathbf{E}_k, k = 1, \ldots, p$. Symbolically,

$$\text{rank } \boldsymbol{\Phi}_u = p(1 + n_b), \quad \text{where} \quad \boldsymbol{\Phi}_u = \begin{bmatrix} \boldsymbol{\psi}_{\bar{u}}(1) \\ \vdots \\ \boldsymbol{\psi}_{\bar{u}}(N) \end{bmatrix}. \tag{18}$$

In words, $\boldsymbol{\Phi}_u$ has full column rank for any large N. This is similar to the persistent excitation requirement for input signals in open-loop identification.

(b) There does not exist a common linear time-invariant feedback relationship between inputs and outputs over all channels. Symbolically,

$$\overline{\mathbb{E}} \left\| R(q^{-1}) \mathbf{G}(\mathbf{c}) \mathbf{u}(t - d) + S(q^{-1}) \mathbf{y}(t) \right\|^2 > 0, \quad \forall \mathbf{c} \in \Omega_c, \tag{19}$$

where $R(q^{-1})$ and $S(q^{-1})$ are arbitrary scalar linear filters, and $\overline{\mathbb{E}}$ is the generalized expectation operator.

The above assumptions are fairly mild. In particular, it is easy to meet the persistent excitation requirement (18) in Assumption 3(a), since in closed-loop the input signal is filtered white noise that contains enough excitations to make $\boldsymbol{\Phi}_u$ full column rank, especially when N is large. For Assumption 3(b), one can find that all input-output channels have to share the same regulator in order to falsify (19). This assumption becomes more convincing given that most CD MPC has complex dynamics due to the complexity in the associated optimization and constraints (Zhu (2002)). In particular, the presence of actuator constraints in MPC and the resultant switching between active and inactive constraints due to disturbances provide closed-loop identifiability.

Theorem 1. *Consider the data $\mathbf{Z}^N$ in (9) under Assumptions (1)–(3). Apply Algorithm 1. If the parameter estimates $\widehat{\mathbf{a}}^k \neq 0$, $\widehat{\mathbf{b}}^k \neq 0$, $\widehat{\mathbf{c}}^k \neq 0$ for each iteration $k = 1, \ldots, K$, then we have the following.*

(i) If $N \gg n_a + n_b$ and the sequence of parameter estimates $\{\widehat{\boldsymbol{\theta}}_N^k\}$ converges, then its limit is a stationary point for $V_N(\boldsymbol{\theta})$. That is,

$$\widehat{\boldsymbol{\theta}}_N = \lim_{k \to \infty} \widehat{\boldsymbol{\theta}}_N^k \quad \Longrightarrow \quad \nabla V_N(\widehat{\boldsymbol{\theta}}_N) = \mathbf{0}. \tag{20}$$

(ii) The function $\overline{V}(\boldsymbol{\theta}) = \overline{\mathbb{E}}[\boldsymbol{\varepsilon}^T(t, \boldsymbol{\theta}) \boldsymbol{\varepsilon}(t, \boldsymbol{\theta})]$ is the uniform limit of the loss functions $V_N(\boldsymbol{\theta})$ as $N \to \infty$. That is,

$$\sup_{\boldsymbol{\theta} \in \Omega} |V_N(\boldsymbol{\theta}) - \overline{V}(\boldsymbol{\theta})| \to 0 \quad \textit{as } N \to \infty, \ \textit{w.p. } 1. \tag{21}$$

Consequently

$$\widehat{\boldsymbol{\theta}}_N \to \boldsymbol{\theta}^* \quad as\ N \to \infty,\ w.p.\ 1, \tag{22}$$

where $\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}\in\Omega}\ \overline{V}(\boldsymbol{\theta})$.

(iii) The parameter estimates $\widehat{\boldsymbol{\theta}}_N$ *are consistent, i.e., they capture the true value* $\boldsymbol{\theta}^\circ$ *in the limit:*

$$\boldsymbol{\theta}^\circ = \lim_{N\to\infty} \widehat{\boldsymbol{\theta}}_N \quad w.p.\ 1. \tag{23}$$

Proof. See Appendix B.

Remark 1. The convergence promised in Theorem 1(i) holds even when the assumptions on the informativeness of closed-loop data are not satisfied (Golub and Pereyra (1973)). Moreover, Bai and Li (2004) show that the accumulation points of $\{\widehat{\boldsymbol{\theta}}_N^k\}$ are stationary for $V_N(\widehat{\boldsymbol{\theta}})$ even when the sequence $\{\widehat{\boldsymbol{\theta}}_N^k\}$ cannot be shown to converge.

Remark 2. The consistency result in Theorem 1(iii) requires the knowledge of true time-delay. However, this requirement can be relaxed when implementing the MPM detection algorithm. As shown in Section 2.5, the essence is to detect the changes in the pattern of test data relative to the normal training data. For this purpose consistent parameter estimation may not be necessary and we could specify a (possibly inaccurate) time-delay value based on our prior knowledge. As a result, the identification method could produce biased parameter estimates. However, as long as the identified models from the test data are showing discrepancy with respect to the nominal models from the training data, the MPM detection algorithm will detect the difference and predict mismatch, regardless of the inconsistent parameter estimate due to incorrect time-delay.

4. CD MPM with One-class SVM

4.1. One-class Support Vector Machines

Detecting abnormality in various processes are essentially a novelty detection problem, since the number and variety of abnormal situations is typically too vast to imagine any single class expressing them all. The one-class SVM (Schölkopf et al. (2001)) is a well-established method for such cases. Here we briefly sketch the ideas to fix notation.

Consider a Euclidean space $\mathcal{X}$ of dimension r in which every possible data point can be expressed as a single vector $\mathbf{x}$. As detailed by Schölkopf et al. (2001), a cluster boundary in $\mathcal{X}$ can be constructed as the pre-image of a hyperplane in some high-dimensional "feature space" $\mathcal{F}$ under some mapping $\phi\colon \mathcal{X} \to \mathcal{F}$. Given ℓ data points $\mathbf{x}_1, \ldots, \mathbf{x}_\ell$, a natural quadratic programming problem in $\mathcal{F}$ determines the desired hyperplane by separating the origin from the cluster $\{\phi(\mathbf{x}_i)\}$. Convex duality allows this problem to be solved, and the results to be applied, entirely in the data space $\mathcal{X}$. A popular choice for ϕ in the development above corresponds to the Gaussian kernel

$$\kappa(\mathbf{x}_1, \mathbf{x}_2) = \exp\left(-\left\|\mathbf{x}_1 - \mathbf{x}_2\right\|^2/c\right), \qquad \mathbf{x}_1, \mathbf{x}_2 \in \mathcal{X}. \tag{24}$$

(Here $c > 0$ is a tuning parameter.) The dual optimization problem generated by the ℓ data points $\mathbf{x}_1, \ldots, \mathbf{x}_\ell$ is

$$\min_{\boldsymbol{\alpha} \in \mathbb{R}^\ell} \quad \frac{1}{2} \sum_{i=1}^{\ell} \sum_{j=1}^{\ell} \alpha_i \alpha_j \kappa(\mathbf{x}_i, \mathbf{x}_j) \tag{25}$$

$$\text{s.t.} \quad 0 \leq \alpha_i \leq \frac{1}{\nu \ell}, \quad i = 1, \ldots, \ell, \tag{26}$$

$$\sum_{i=1}^{\ell} \alpha_i = 1. \tag{27}$$

(Here $\nu \in (0, 1]$ is a tuning parameter that sets the balance between penalizing incorrect classifications and maximizing the separation margin in the feature space $\mathcal{F}$.) Given a solution $\widehat{\boldsymbol{\alpha}}$ for (25)–(27), the boundary of the cluster region corresponds to the zero-level set for the *prediction score*, p, defined by

$$p(\mathbf{x}) = \sum_{j=1}^{\ell} \widehat{\alpha}_j \left(\kappa(\mathbf{x}_j, \mathbf{x}) - \kappa(\mathbf{x}_j, \mathbf{x}_i) \right). \tag{28}$$

Given a test point $\mathbf{x}$ in the data space $\mathcal{X}$, we say that $\mathbf{x}$ lies inside the data-defined cluster if $p(\mathbf{x}) > 0$, and outside if $p(\mathbf{x}) < 0$. The index i in (28) can be any one for which the strict inequalities $0 < \widehat{\alpha}_i < 1/(\nu\ell)$ hold (compare (26)); all such i give the same definition.

4.2. Application of one-class SVM to CD MPM detection

As mentioned above, our approach to the full-sheet problem involves independent parallel application of SVM-based methods to three processes—spatial, temporal, and noise. The temporal case is representative, so we focus on it here.

4.2.1. SVM training

In the generic notation above, ℓ represents the number of moving windows in the training data set. We take for each training vector $\mathbf{x}_i$ the FIR coefficients of the temporal model estimated from observations in i-th moving window. In symbols,

$$\mathbf{x}_i = [\widehat{x}_i^1 \ \ldots \ \widehat{x}_i^{n_g}]^T, \qquad i = 1, 2, \ldots, \ell, \tag{29}$$

where n_g is the order of the identified FIR model. These ℓ vectors set up the dual problem (25)–(27), whose solution $\widehat{\boldsymbol{\alpha}}$ yields the desired prediction function as in (28).

4.2.2. Resampling

In practice, the amount of available training data may be in short supply due to frequent transitions in production tasks driven by customer demands. In such cases, we can synthesize additional vectors by re-sampling from the (estimated) probability distributions of the FIR coefficient vectors. Mahata and Söderström (2004) show that parameter estimates from the separable nonlinear least-squares method have Gaussian distributions (assuming the noise is Gaussian); their argument can be adapted to give the same conclusion for our iterative closed-loop identification algorithm. Thus we can use real operating data to construct rough estimators for the mean μ_k and variance σ_k of each FIR coefficient $\widehat{x}^k$, $k = 1, \dots, n_g$,

$$\widehat{\mu}_k = \mu(\widehat{x}_1^k, \dots, \widehat{x}_\ell^k), \quad \widehat{\sigma}_k = \sigma(\widehat{x}_1^k, \dots, \widehat{x}_\ell^k), \tag{30}$$

and synthesize compatible training models by drawing new FIR coefficients x_k from the corresponding Gaussian distributions. Typical choices for $\mu(\cdot)$ and $\sigma(\cdot)$ in (30) are the sample mean and sample variance.

Now the estimator in (30) is likely to underestimate the true variance when ℓ is small. To mitigate this, we introduce a scaling factor α_T for $\widehat{\sigma}_k$ (subscript T means temporal) and use Gaussian distributions of mean μ_k and variance $\alpha_T \sigma_k$ for resampling. In general, $\alpha_T \geq 1$: we use $\alpha_T \approx 1$ when ℓ is large enough to make mitigation unnecessary, and larger $\alpha_T > 1$ when ℓ is small. Intuitively, increasing α_T enlarges the variability considered "normal" and consequently reduces the sensitivity of the overall mismatch detection algorithm.

Of course, when plenty of historical data are available, re-sampling may not be needed.

4.2.3. SVM-based MPM detection

With the prediction score function $p = p_T(\mathbf{x})$ of (28) in hand, we generate new inputs in the same way as we gathered the original training data. That is, we take a temporal window of data with final time t, and apply closed-loop identification to calculate the corresponding vector $\mathbf{x} = \mathbf{x}(t)$ of FIR coefficients in $\mathbb{R}^{n_g}$. The sign of $p(\mathbf{x}(t))$ predicts whether this model is compatible with the cluster constructed using normal operation. A negative value predicts mismatch. For robustness, we use the *frequency of negative results* as an MPM indicator. Thus we introduce a number n_T to set the number of recent windows to consider, and define

$$s_T(t) = \frac{1}{n_T} \left| \{ t' : \; t - n_T + 1 \leq t' \leq t, \; p_T(\mathbf{x}(t')) < 0 \} \right| \tag{31}$$

(Here $|A|$ denotes the number of elements in a set A.) We have $s_T(t) = 0$ when all n_T samples up to time t are classified as normal and $s_T(t) = 1$ when all such samples are classified as abnormal. Intermediate situations give intermediate values; after choosing n_T, the user can specify a threshhold for s_T above which a temporal MPM alarm is raised.

Developments perfectly analogous to those just detailed for the temporal process lead to prediction functions $p_N(\mathbf{x})$ and $p_S(\mathbf{x})$ for the noise and spatial processes, with corresponding frequency measures $s_N(t)$ and $s_S(t)$. Fig. 5 illustrates the approach, providing plots of $p_N(t)$, $p_T(t)$, and $p_S(t)$ for a realistic scenario detailed below.

Table 2: Tuning parameters of the CD MPC

Tuning parameters	Values	Tuning parameters	Values
CV target weight	0.40	MV temporal movement weight	0.25
MV target weight	0.17	MV spatial picketing weight	0.14
CV target value	42	MV target value	0
Prediction horizon	25	Control horizon	4
Actuator bend limit	30	Actuator upper/lower average	±1
Actuator change rate limit	15	Actuator upper/lower limit	±20

5. Examples

This section verifies the proposed iterative algorithm for closed-loop identification and the effectiveness of our approach to mismatch detection using realistic simulations. The simulation platform used here was provided by Honeywell Process Solutions. It provides a high-fidelity surrogate for an actual paper machine. However, we stress that real paper machine operating data is critical to ensure the effectiveness of our algorithms, which will be part of the future work.

5.1. Example 1: Iterative CD identification in closed-loop routine operation

In a representative paper-making process, the MV is a vector of inputs for the $n = 74$ autoslice actuators across the headbox that release pulp slurry into the sheet-making process. The CV is a vector of $m = 222$ measurements of the dry weight of the paper product, taken at equally-spaced locations across the sheet as it emerges from the machine. (The symbols here correspond to the overview in subsection 2.1.) Samples are taken every $T_s = 12$ seconds, and the time-constant is $T_p = 126.5$ seconds. The time delay is $d = 2$ time steps. After discretization, the true CD process has these temporal and spatial parameters: $f^\circ = 0.9095, \quad \boldsymbol{\theta}_S^\circ = [0.38 \ \ 269\text{mm} \ \ 0.10 \ \ 1.5]^T, \quad d = 2$. The MPC controller detailed in Fan (2003) is installed. It enforces constraints of four types: bend limits for neighboring actuators, bound limits for the actuator average profile, upper and lower limits for the actuator profile, and limits on each actuator's rate of change. Key numerical values are given in Table 2. The true noise $\mathbf{v}^\circ$ is produced by passing Gaussian white noise $\mathbf{e}(t)$ with mean $\mathbf{0}$ through a high-pass filter H°: $\mathbf{v}^\circ(t) = H^\circ(q^{-1})\mathbf{e}(t), H^\circ(q^{-1}) = \frac{1-0.6q^{-1}+0.3q^{-2}-0.1q^{-3}}{1+0.4q^{-1}+0.1q^{-2}+0.05q^{-3}}$. The variance of each output channel is 0.01. The relevant simulation parameters are given in Table 3 (which also contains several parameters that will be used later). For this experiment, we assume an accurate plant model (no MPM). The setpoint is left unchanged during the entire simulation, i.e., the noise is the only external signal to the system. In addition, we assume that the true process delay is available and is incorporated into the identification algorithm to avoid estimating the inverse of controller as the process model. The simulation of the closed-loop system lasts for 120 minutes (600 samples of data), as shown in Figure 2. We collect the last 400 samples of data after initial transients for the closed-loop identification.

Table 3: Simulation and MPM detection parameters for Example 1 and Example 2

Parameters	Values	Parameters	Values
Actuator zone width	62.5194 mm	CD bin width	20.8398 mm
Sampling interval	12 seconds	Iteration number	5
Initial temporal $\boldsymbol{\theta}_T^i$	[0.82 0.18]	Initial spatial $\boldsymbol{\theta}_S^i$	[0.3 200 0.2 4.0]
Window size	80 min	Window step size	20 min
Training data size	800 min	Temporal α_T	3
Spatial α_S	1.5	Noise α_N	3
Time noise model changes	1200^{th} min	Time MPM occurs	1600^{th} min

The initial spatial and temporal parameters for the identification algorithm are given in Table 3. Like most Hammerstein model identifications, our algorithm converges rapidly, accurately approximating a local minimum in just a few iterations. The maximum iteration

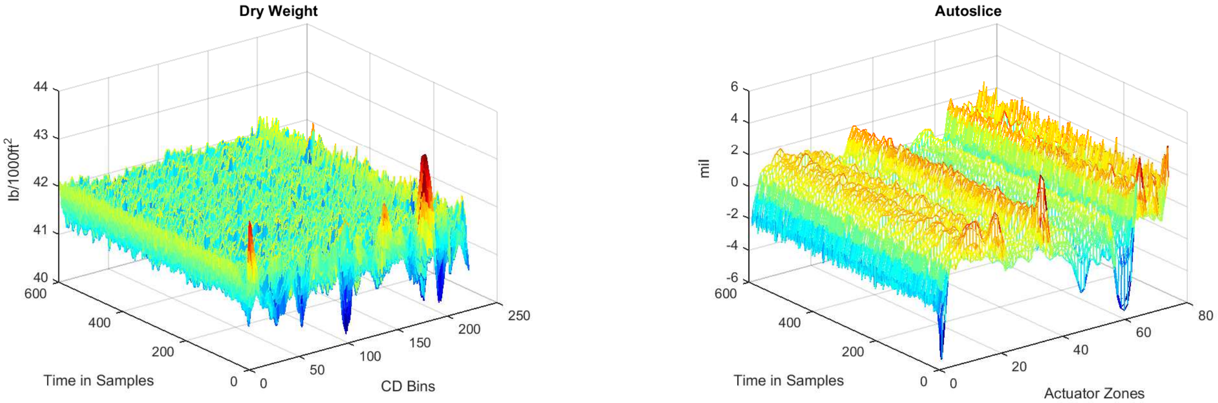

Figure 2: Simulated input-output data for the closed-loop CD process

number is set to $K = 5$. For the temporal model, we choose $n_a = 20$, $n_b = 50$; the spatial order is set to $p = 30$. Proper regularizations are necessary in executing this algorithm to smooth the estimated FIR coefficients. The regularization can also ensure the numerical stability incurred with the high-order least-squares problem (12) when the regressor matrix has a large condition number.

The left three plots in Fig. 3 compare the impulse responses of the true and estimated noise, temporal, and spatial models. The estimated models show high agreement with the true models, indicating the effectiveness of the proposed routine identification algorithm. More importantly, the noise and process models can be estimated independently, making it possible to distinguish between changes in the two models. After steps 4–6 in Algorithm 1, the estimated values of the temporal and spatial parameters are: $\widehat{f} = 0.9060$ $\widehat{\boldsymbol{\theta}}_S = [0.3442 \;\; 277.0566\,\mathrm{mm} \;\; 0.0694 \;\; 1.7889]^T$. These values agree well with true values. As an additional test, we repeat the entire closed-loop identification process above after swapping a low-pass filter H° into the noise model, producing $\mathbf{v}^\circ(t) = H^\circ(q^{-1})\mathbf{e}(t)$, $H^\circ(q^{-1}) = \frac{1+0.7q^{-1}+0.4q^{-2}}{1-0.5q^{-1}+0.1q^{-2}}$. The corresponding identification results are shown on the right side of Fig. 3. Again, the impulse responses for the estimated system and the true system are in excellent agreement. These results verify the effectiveness of our proposed CD closed-loop identification method.

5.2. Example 2: One-class SVM model-plant mismatch detection

Here we extend the simulation in Example 5.1 to focus on detecting model-plant mismatch. We specify a relatively small spatial scaling factor α_S to increase the sensitivity of our MPM detection algorithm to spatial mismatches. The main process characteristics given in Table 3 remain in force.

The simulation timeline for this example runs as follows. Initially there is neither MPM nor noise model change and the noise model $\mathbf{v}^\circ(t) = H^\circ(q^{-1})\mathbf{e}(t)$ involves $H^\circ(q^{-1}) = \frac{1-0.6q^{-1}}{1+0.4q^{-1}}$. After disposing of the initial transient part, we use the first 4000 samples (cf. Table 3) as the training data. Then we apply moving windows, each 400 samples wide, to the training data and identify spatial, temporal, and noise models in each window. A new window is initiated every 100 samples. Once this step is complete, we train one-class SVMs separately on these models and obtain the corresponding prediction functions. This procedure involves scaling the initial cluster and re-sampling. From then on, the SVMs start to predict mis-

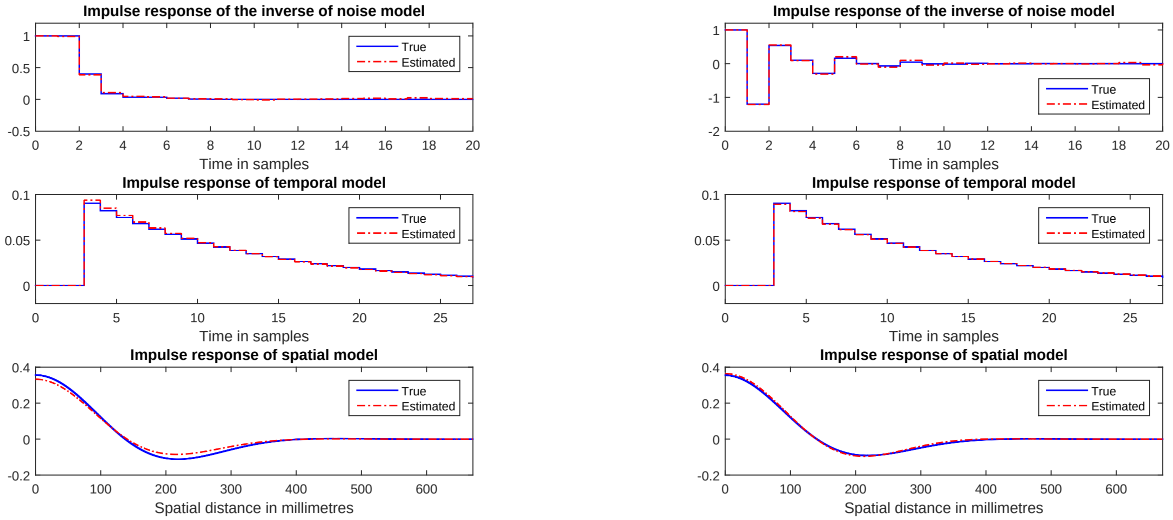

Figure 3: CD closed-loop iterative identification results: noise model is a high-pass filter (left); noise model is a low-pass filter (right);

matches. After 6000 samples, we *gradually* change the noise model to $\mathbf{v}^\circ(t) = H^\circ(q^{-1})\mathbf{e}(t)$ with $H^\circ(q^{-1}) = \frac{1+0.7q^{-1}}{1-0.5q^{-1}}$. Then, at index $t = 8000$, the width of true plant model begins to ramp up to 1.5 times its original value over a span of 300 samples. The true width parameter remains at its new value thereafter. Note that throughout this simulation there is no setpoint change or any other external excitation.

Colormaps showing the simulated input and output data are shown in Fig. 4. The plots highlight the times at which noise model change and spatial MPM are introduced. It is clear that the output variance increases after introducing the noise model change, and increases even more after adding the MPM. The left plots in Fig. 5 illustrate the spatial and temporal parameter estimates over all moving windows. The red dash-dotted line in each plot shows the true parameter value. The blue lines display the estimated parameter values in each moving window. Although these parameter estimates are rough due to low excitation levels in the routine closed-loop data, they still provide valuable indications on which parameter may have drifted. Moreover, this closed-loop identification algorithm can easily carry over to the data collected during a closed-loop identification experiment. This means that if a closed-loop identification is necessary, the user only needs to start injecting external excitations to the system, with the closed-loop identification algorithm operating continuously.

Detailed results on mismatch detection are shown on the right of Fig. 5. As a baseline for comparison, the online user-specified performance index in Lu et al. (2017a) is also used to monitor the output. This index is strongly affected by changes in output variance. It starts to decline immediately after the noise model changes, even before the MPM is added. This is the main drawback of variance-based performance indices: they convey no insight about the *cause* of the poor performance they detect. In contrast, the scores obtained from independent concurrent SVM predictions clearly indicate the reasons for poor performance. Witness the sharp and lasting drop in the SVM scores for the noise model shortly after $t = 6000$, while the temporal and spatial scores remain high, followed by a similar transition in the spatial

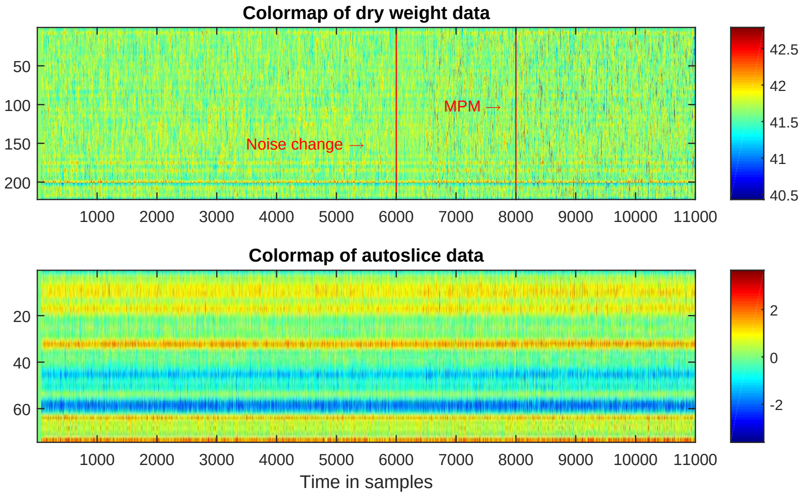

Figure 4: The colormap of simulated input-output data

SVM score soon after $t = 8000$ while the temporal score remains high. The accumulated score metric described in section 4 absorbs transient fluctuations in the scores, leading to a robust system for accurately detecting model-plant mismatch.

6. Summary

This work presents a novel MPM detection approach based on closed-loop identification method that provides consistent parameter estimates for the CD process. It applies to routine operating data without external excitation, provided that a few mild conditions on the informativeness of closed-loop data are satisfied.One-class SVM models are developed based on the clusters of model estimates from the training data, and used to predict classifications of models estimated from the test data. From the predictions, we are able to detect MPM. A key advantage of this approach is independent monitoring of changes in the process and noise models, leading to MPM alarms that are robust to noise model changes. Moreover, procedures for implementing this MPM detection framework are provided in this paper and additional techniques (e.g., resampling) are also introduced to address practical issues such as a shortage of training data. Two examples are presented to validate the effectiveness of the proposed methods. As part of the future work, we will test our results on real paper machine to validate the effectiveness of the proposed algorithm.

Appendix A. Proof of the asymptotic equivalence of (6) and (1)

Consider the generic scalar system $y(t) = G_s(q^{-1})u(t) + H_s(q^{-1})e(t)$, where $G_s(q^{-1})$ and $H_s(q^{-1})$ are stable and $H_s(q^{-1})$ is inversely stable. Dividing by H_s provides the equivalent representation $A_s(q^{-1})y(t) = B_s(q^{-1})u(t) + e(t)$ where $A_s(q^{-1}) = \frac{1}{H_s(q^{-1})} = 1 + \sum_{k=1}^{\infty} a_k^s q^{-k}$ and $B_s(q^{-1}) = \frac{G_s(q^{-1})}{H_s(q^{-1})} = \sum_{k=0}^{\infty} b_k^s q^{-k}$. Model the system as $A(q^{-1}, \eta^n) = B(q^{-1}, \eta^n)u_t + e_t$, where the vector η^n stacks all the coefficients in the polynomials $A(q^{-1}, \eta^n) = 1 + \sum_{k=1}^{n} a_k q^{-k}$, $B(q^{-1}, \eta^n) = \sum_{k=0}^{n} b_k q^{-k}$. Assume that $u(t)$ is sufficiently exciting. Define sample size as N and model order $n(N)$ as a function of N. Suppose that as $N \to \infty$, $n(N) \to \infty$ but

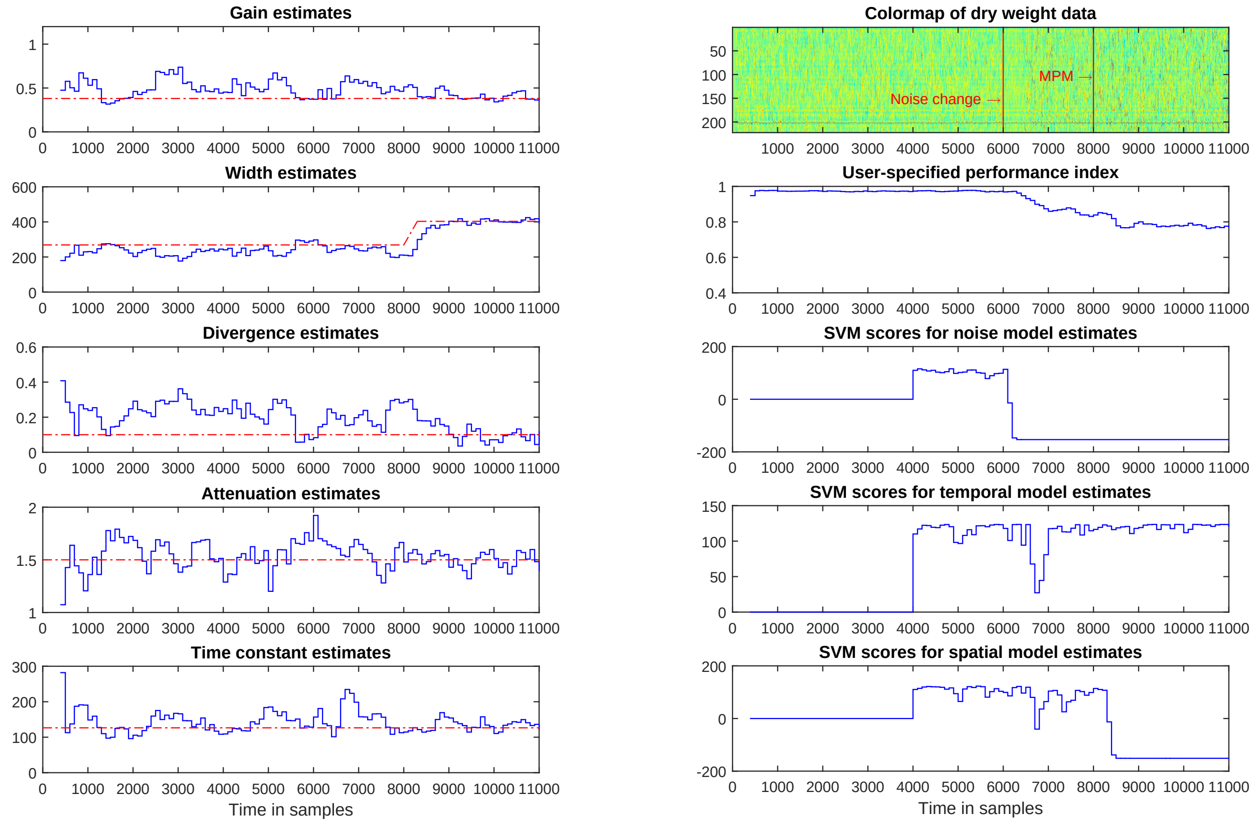

Figure 5: Simulated input-output data for the closed-loop CD process

$n(N)^{3+\delta}/N \to 0$, for some $\delta > 0$. That is, the model order goes to infinity as the sample size N increases, but at a rate quantifiably slower than linear in N. Under these conditions, Lemma 2.1 in Zhu and Hjalmarsson (2016) shows that as $N \to \infty$

$$\sup_{\omega} \left| A(e^{-j\omega}, \eta_N^{n(N)}) - A_s(e^{-j\omega}) \right| \to 0, \quad \sup_{\omega} \left| B(e^{-j\omega}, \eta_N^{n(N)}) - B_s(e^{-j\omega}) \right| \to 0. \tag{A.1}$$

Informally, identification of a generic scalar model can be achieved by identifying an ARX model of appropriate order.

Similarly, our multivariate CD model (1) can be written $A_m(q^{-1})y(t) = B_m(q^{-1})\mathbf{Gu}(t-d) + \mathbf{e}(t)$, where $A_m(q^{-1}) = 1/H(q^{-1})$, $B_m(q^{-1}) = g(q^{-1})/H(q^{-1})$. Given that both $A_m(q^{-1})$ and $B_m(q^{-1})$ are stable by assumption, we can approximate them by finite sums. Consider the finitely parameterized structure (6). Much as above, suppose that $\mathbf{Gu}(t-d)$ is sufficiently exciting, and that as $N \to \infty$, the degrees $n_a(N)$ and $n_b(N)$ increase to ∞ slowly enough that $n_a(N)^{3+\delta}/N \to 0$ and $n_b(N)^{3+\delta}/N \to 0$, for some $\delta > 0$. We find that, as $N \to \infty$,

$$\sup_{\omega} \left| A(e^{-j\omega}, \mathbf{a}) - A_m(e^{-j\omega}) \right| \to 0, \quad \sup_{\omega} \left| B(e^{-j\omega}, \mathbf{b}) - B_m(e^{-j\omega}) \right| \to 0. \tag{A.2}$$

The only difference is that the CD model is multivariate. Identifying the scalar filters $A(q^{-1}, \mathbf{a})$ and $B(q^{-1}, \mathbf{b})$ in (6) can be completed by considering it as a multiple-experiment identification. This confirms that the original CD multivariate model (1) can be approximated by a high-order ARX model as shown in (6).

Appendix B. Proof of Theorem 1

Defining

$$\mathbf{\Phi}_y = \begin{bmatrix} \boldsymbol{\psi}_{\bar{y}}(1) \\ \vdots \\ \boldsymbol{\psi}_{\bar{y}}(N) \end{bmatrix},$$

it then follows from (7) that

$$\widehat{\mathbf{Y}} = [\mathbf{\Phi}_y \ \mathbf{\Phi}_u] \begin{bmatrix} \mathbf{a} \\ \mathbf{Cb} \end{bmatrix} = \begin{bmatrix} \widetilde{\mathbf{\Phi}}_y \ \widetilde{\mathbf{\Phi}}_u \end{bmatrix} \begin{bmatrix} \mathbf{a} \\ \mathbf{Bc} \end{bmatrix}, \tag{B.1}$$

where $\mathbf{B} = \operatorname{diag}\{\mathbf{b}, \ldots, \mathbf{b}\}$. $\widetilde{\mathbf{\Phi}}_y$ and $\widetilde{\mathbf{\Phi}}_u$ are easily obtained by rearranging $\mathbf{\Phi}_y$ and $\mathbf{\Phi}_y$, respectively. The loss function (10) is expressed in a more compact form,

$$V_N(\boldsymbol{\theta}) = \|\mathbf{Y} - \widehat{\mathbf{Y}}\|_2^2 = \left\| \mathbf{Y} - [\mathbf{\Phi}_y \ \mathbf{\Phi}_u] \begin{bmatrix} \mathbf{a} \\ \mathbf{Cb} \end{bmatrix} \right\|_2^2. \tag{B.2}$$

(*i*) The proof extends the Theorem IV.1 in Bai and Li (2004). Note that to ease the notation, the following derivations will drop the hat in the parameter estimates from (12)–(14), but add a subscript "N" to stress the fact that these estimates are obtained from N samples of data. It is easy to see that through the iterative identification steps (12)-(14),

$$V_N(\boldsymbol{\theta}_N^k) = V_N(\mathbf{a}_N^k, \mathbf{b}_N^k, \mathbf{c}_N^k) \leq V_N(\mathbf{a}_N^k, \mathbf{b}_N^k, \mathbf{c}_N^{k-1}) \leq V_N(\mathbf{a}_N^{k-1}, \mathbf{b}_N^{k-1}, \mathbf{c}_N^{k-1}) = V_N(\boldsymbol{\theta}_N^{k-1}).$$

In other words, $V_N(\boldsymbol{\theta}_N^k)$ is nonincreasing in k. The normalization step in (13) ensures that the sequences $\{\mathbf{b}_N^k\}$ and $\{\mathbf{c}_N^k\}$ are bounded. (Otherwise, $V_N(\boldsymbol{\theta}_N^k)$ would be unbounded — contradicting the facts that $\mathbf{Z}^N$ is bounded, all model structures in Ω are stable, and that $V_N(\boldsymbol{\theta})$ is continuous in $\boldsymbol{\theta}$ and nonincreasing.)

Now concentrate on the convergence of the identification algorithms for large N. First let us establish a statement that when fixing the spatial parameter $\mathbf{c}$, $V_N(\boldsymbol{\theta})$ is convex in temporal parameters $\{\mathbf{a}, \mathbf{b}\}$ and vice versa. Based on (B.2), one has

$$V_N(\mathbf{a}, \mathbf{b}, \mathbf{c}) = \mathbf{Y}^T\mathbf{Y} - 2\mathbf{Y}^T\mathbf{\Phi} \begin{bmatrix} \mathbf{a} \\ \mathbf{Cb} \end{bmatrix} + \begin{bmatrix} \mathbf{a} \\ \mathbf{Cb} \end{bmatrix}^T \mathbf{\Phi}^T\mathbf{\Phi} \begin{bmatrix} \mathbf{a} \\ \mathbf{Cb} \end{bmatrix}, \quad \mathbf{\Phi} = [\mathbf{\Phi}_y \ \mathbf{\Phi}_u]. \tag{B.3}$$

When the parameter $\mathbf{c}$ is fixed, it is easy to derive that for $\lambda \in [0, 1]$,

$$\begin{aligned} V_N(\lambda\mathbf{a}_1 + (1-\lambda)\mathbf{a}_2, \lambda\mathbf{b}_1 + (1-\lambda)\mathbf{b}_2, \mathbf{c}) &= \lambda V_N(\mathbf{a}_1, \mathbf{b}_1, \mathbf{c}) + (1-\lambda)V_N(\mathbf{a}_2, \mathbf{b}_2, \mathbf{c}) \\ -\lambda(1-\lambda) \left(\begin{bmatrix} \mathbf{a}_1 \\ \mathbf{Cb}_1 \end{bmatrix} - \begin{bmatrix} \mathbf{a}_1 \\ \mathbf{Cb}_1 \end{bmatrix} \right)^T &\mathbf{\Phi}^T\mathbf{\Phi} \left(\begin{bmatrix} \mathbf{a}_2 \\ \mathbf{Cb}_2 \end{bmatrix} - \begin{bmatrix} \mathbf{a}_2 \\ \mathbf{Cb}_2 \end{bmatrix} \right). \end{aligned} \tag{B.4}$$

Due to Assumption 3, for any large N, $\mathbf{\Phi}$ has full column rank, which renders $\mathbf{\Phi}^T\mathbf{\Phi} > 0$. Therefore, with the fact that $\mathbf{a} \neq 0$, $\mathbf{b} \neq 0$, $\mathbf{c} \neq 0$, (B.4) implies that

$$V_N(\lambda\mathbf{a}_1 + (1-\lambda)\mathbf{a}_2, \lambda\mathbf{b}_1 + (1-\lambda)\mathbf{b}_2, \mathbf{c}) \geq \lambda V_N(\mathbf{a}_1, \mathbf{b}_1, \mathbf{c}) + (1-\lambda)V_N(\mathbf{a}_2, \mathbf{b}_2, \mathbf{c}).$$

This verifies the convexity of $V_N(\mathbf{a}, \mathbf{b}, \mathbf{c})$ with respect to $\mathbf{a}, \mathbf{b}$. A similar conclusion can be achieved for the convexity of $V_N(\mathbf{a}, \mathbf{b}, \mathbf{c})$ with respect to $\mathbf{c}$. An immediate consequence of these statements is that in each optimization of the iterative identification algorithm, there exists a unique and closed-form solution. It will be shown below that even though $V_N(\boldsymbol{\theta})$ may not be convex in $\boldsymbol{\theta}$, our algorithm can always converge to its local minimum.

Now define $\boldsymbol{\theta}^k$ to be the parameter estimate at k-th iteration. When the iterative identification algorithm (12)-(14) converges, i.e., $\boldsymbol{\theta}^k \to \boldsymbol{\theta}$ for some $\boldsymbol{\theta}$, it is necessary to show that $\nabla V_N(\boldsymbol{\theta}) = 0$. Suppose that $\nabla V_N(\boldsymbol{\theta}) \neq 0$, then there always exists a direction (e.g. negative gradient) along which $V_N(\boldsymbol{\theta})$ decreases. Because the solution to (12) and to (14) is unique, one can always minimize $V_N(\boldsymbol{\theta})$ sequentially to achieve another point $\boldsymbol{\theta}'$, such that $V_N(\boldsymbol{\theta}') \leq V_N(\boldsymbol{\theta})$. This contradicts the facts that $V_N(\boldsymbol{\theta})$ is nonincreasing and $\boldsymbol{\theta}^k$ converges to $\boldsymbol{\theta}$. Thus the statement in (i) holds.

(ii) Combining (6) and (8), one can see that the closed-loop system has the form

$$\mathbf{y}(t) = f_{\mathcal{S}}\left(t, \mathbf{y}^{t-1}, \mathbf{u}^{t-1}\right) + \mathbf{e}^{\circ}(t). \tag{B.5}$$

According to Assumption 1, the nonlinear closed-loop system (B.5) is exponentially stable, which satisfies S1-S3 in Ljung (1978). Moreover, with the parameterizations (6), the one-step-ahead predictor (7) of the model is differentiable with respect to parameter $\boldsymbol{\theta}$, which implies the condition M1 in Ljung (1978). Our selected quadratic criterion also meets the regularity condition C1 in that paper. As a result, Lemma 3.1 in Ljung (1978) applies to our scenario, which indicates the validity of (21). As the convergence in (21) is uniform in $\boldsymbol{\theta} \in \Omega$, (22) follows directly from (21).

(iii) From Assumptions 1 and 2, it follows that $\boldsymbol{\varepsilon}(t, \boldsymbol{\theta}^{\circ}) = \mathbf{e}^{\circ}(t)$. Therefore,

$$\begin{aligned} \overline{V}(\boldsymbol{\theta}) - \overline{V}(\boldsymbol{\theta}^{\circ}) \;\; = \;\; & \overline{\mathbb{E}}[\varepsilon(t, \boldsymbol{\theta}) - \varepsilon(t, \boldsymbol{\theta}^{\circ})]^T \varepsilon(t, \boldsymbol{\theta}^{\circ}) \\ & + \overline{\mathbb{E}}[\varepsilon(t, \boldsymbol{\theta}) - \varepsilon(t, \boldsymbol{\theta}^{\circ})]^T [\varepsilon(t, \boldsymbol{\theta}) - \varepsilon(t, \boldsymbol{\theta}^{\circ})]. \end{aligned} \tag{B.6}$$

Note that $\varepsilon(t, \boldsymbol{\theta}) - \varepsilon(t, \boldsymbol{\theta}^{\circ}) = \widehat{\mathbf{y}}(t|\boldsymbol{\theta}) - \widehat{\mathbf{y}}(t|\boldsymbol{\theta}^{\circ})$ which only depends on past input-output data and thus is not correlated with current noise $\varepsilon(t, \boldsymbol{\theta}^{\circ})$. Hence, the first term in (B.6) is zero. The second term is always nonnegative which means that $\overline{V}(\boldsymbol{\theta})$ is always greater than $\overline{V}(\boldsymbol{\theta}^{\circ})$ unless $\widehat{\mathbf{y}}(t|t-1, \boldsymbol{\theta}) = \widehat{\mathbf{y}}(t|t-1, \boldsymbol{\theta}^{\circ})$. Now let us show that under the informativeness condition of closed-loop data as in Assumption 3, $\widehat{\mathbf{y}}(t|\boldsymbol{\theta}^*) = \widehat{\mathbf{y}}(t|\boldsymbol{\theta}^{\circ})$ implies that $\boldsymbol{\theta}^* = \boldsymbol{\theta}^{\circ}$. From Assumption 2 and (7)

$$\widehat{\mathbf{y}}(t|\boldsymbol{\theta}^*) - \widehat{\mathbf{y}}(t|\boldsymbol{\theta}^{\circ}) = \boldsymbol{\psi}(t) \begin{bmatrix} \mathbf{a}^* - \mathbf{a}^{\circ} \\ \mathbf{Cb}^* - \mathbf{Cb}^{\circ} \end{bmatrix}, \quad \boldsymbol{\psi}(t) = \begin{bmatrix} \boldsymbol{\psi}_y(t) \; \boldsymbol{\psi}_{\bar{u}}(t-d) \end{bmatrix}. \tag{B.7}$$

Plugging this into the limit loss function (B.6) yields

$$\overline{\mathbb{E}}[\widehat{\mathbf{y}}(t|\boldsymbol{\theta}^*) - \widehat{\mathbf{y}}(t|\boldsymbol{\theta}^{\circ})] = \begin{bmatrix} \mathbf{a}^* - \mathbf{a}^{\circ} \\ \mathbf{Cb}^* - \mathbf{Cb}^{\circ} \end{bmatrix}^T \overline{\mathbb{E}}[\boldsymbol{\psi}^T(t)\boldsymbol{\psi}(t)] \begin{bmatrix} \mathbf{a}^* - \mathbf{a}^{\circ} \\ \mathbf{Cb}^* - \mathbf{Cb}^{\circ} \end{bmatrix}. \tag{B.8}$$

From Assumption 3a, (18) holds for any large N. Thus asymptotically, $\overline{E}[\boldsymbol{\psi}_{\bar{u}}(t)]$ has full column rank. Moreover, according to (19), columns of $\overline{E}[\boldsymbol{\psi}_{\bar{u}}(t-d)]$ are linearly independent of those in $\overline{E}[\boldsymbol{\psi}_{\bar{y}}(t)]$. This implies that $\overline{\mathbb{E}}[\boldsymbol{\psi}^T(t)\boldsymbol{\psi}(t)]$ has full column rank. Therefore, the only situation making $\overline{V}(\boldsymbol{\theta}^*) - \overline{V}(\boldsymbol{\theta}^{\circ}) = 0$ is $\mathbf{a}^* = \mathbf{a}^{\circ}$, $\mathbf{Cb}^* = \mathbf{Cb}^{\circ}$. Note that the rescaling (step 6) of the algorithm gives rise to $\mathbf{C}^* = \mathbf{C}^{\circ}$, $\mathbf{b}^* = \mathbf{b}^{\circ}$, and this ends the proof of (23).

Badwe, A.S., Gudi, R.D., Patwardhan, R.S., Shah, S.L., Patwardhan, S.C., 2009. Detection of model-plant mismatch in mpc applications. Journal of Process Control 19, 1305–1313.

Badwe, A.S., Patwardhan, R.S., Shah, S.L., Patwardhan, S.C., Gudi, R.D., 2010. Quantifying the impact of model-plant mismatch on controller performance. Journal of Process Control 20, 408–425.

Bai, E.W., Li, D., 2004. Convergence of the iterative Hammerstein system identification algorithm. IEEE Transactions on Automatic Control 49, 1929–1940.

Bemporad, A., Morari, M., Dua, V., Pistikopoulos, E.N., 2002. The explicit linear quadratic regulator for constrained systems. Automatica 38, 3–20.

Botelho, V., Trierweiler, J.O., Farenzena, M., Duraiski, R., 2016. Perspectives and challenges in performance assessment of model predictive control. The Canadian Journal of Chemical Engineering 94, 1225–1241.

Fan, J., 2003. Model predictive control for multiple cross-directional processes: Analysis, tuning, and implementation. Ph.D. thesis. University of British Columbia.

Gevers, M., Bazanella, A.S., Bombois, X., et al., 2009. Identification and the information matrix: How to get just sufficiently rich? IEEE Transactions on Automatic Control 54, 2828–2840.

Golub, G., Pereyra, V., 2003. Separable nonlinear least squares: the variable projection method and its applications. Inverse Problems 19, R1.

Golub, G.H., Pereyra, V., 1973. The differentiation of pseudo-inverses and nonlinear least squares problems whose variables separate. SIAM Journal on Numerical Analysis 10, 413–432.

Gorinevsky, D.M., Gheorghe, C., 2003. Identification tool for cross-directional processes. IEEE Transactions on Control Systems Technology 11, 629–640.

Harris, T.J., 1989. Assessment of control loop performance. The Canadian Journal of Chemical Engineering 67, 856–861.

Julien, R.H., Foley, M.W., Cluett, W.R., 2004. Performance assessment using a model predictive control benchmark. Journal of Process Control 14, 441–456.

Ljung, L., 1978. Convergence analysis of parametric identification methods. IEEE Transactions on Automatic Control 23, 770–783.

Ljung, L., 1999. System Identification: Theory for the User. Upper Saddle River: Prentice Hall.

Ljung, L., Wahlberg, B., 1992. Asymptotic properties of the least-squares method for estimating transfer functions and disturbance spectra. Advances in Applied Probability 24, 412–440.

Lu, Q., Forbes, M.G., Gopaluni, R.B., Loewen, P.D., Backström, J.U., Dumont, G.A., 2017a. Performance assessment of cross-directional control for paper machines. IEEE Transactions on Control Systems Technology 25, 208–221.

Lu, Q., Forbes, M.G., Loewen, P.D., Backström, J.U., Dumont, G.A., Gopaluni, R.B., 2020. Support vector machine approach for model-plant mismatch detection. Computers & Chemical Engineering 133, 106660.

Lu, Q., Gopaluni, R.B., Forbes, M.G., Loewen, P.D., Backström, J., Dumont, G.A., 2017b. Model-plant mismatch detection with support vector machines, in: IFAC 2017 World Congress, pp. 7993–7998.

Mahata, K., Söderström, T., 2004. Large sample properties of separable nonlinear least squares estimators. IEEE Ttransactions on Signal Processing 52, 1650–1658.

Morales, R.M., Heath, W.P., 2011. The robustness and design of constrained cross-directional control via integral quadratic constraints. IEEE Transactions on Control Systems Technology 19, 1421–1432.

Narendra, K., Gallman, P., 1966. An iterative method for the identification of nonlinear systems using a hammerstein model. IEEE Transactions on Automatic Control 11, 546–550.

Qin, S.J., Badgwell, T.A., 2003. A survey of industrial model predictive control technology. Control Engineering Practice 11, 733–764.

Rigopoulos, A., Arkun, Y., Kayihan, F., 1997. Identification of full profile disturbance models for sheet forming processes. AIChE Journal 43, 727–739.

Schölkopf, B., Platt, J.C., Shawe-Taylor, J., Smola, A.J., Williamson, R.C., 2001. Estimating the support of a high-dimensional distribution. Neural Computation 13, 1443–1471.

Shardt, Y.A., Huang, B., 2011. Closed-loop identification condition for armax models using routine operating data. Automatica 47, 1534–1537.

Söderström, T., Stoica, P., 1988. System Identification. Prentice-Hall, Inc.

Sun, Z., Qin, S.J., Singhal, A., Megan, L., 2013. Performance monitoring of model-predictive controllers via model residual assessment. Journal of Process Control 23, 473–482.

Wang, S., Simkoff, J.M., Baldea, M., Chiang, L.H., Castillo, I., Bindlish, R., Stanley, D.B., 2016. Data-driven plant-model mismatch quantification in input-constrained linear mpc. IFAC-PapersOnLine 49, 25–30.

Zhu, Y., 2002. Estimation of an N–L–N Hammerstein–Wiener model. Automatica 38, 1607–1614.

Zhu, Y., Hjalmarsson, H., 2016. The Box–Jenkins Steiglitz–McBride algorithm. Automatica 65, 170–182.