Preprints of the
9th International Symposium on Advanced Control of Chemical Processes
The International Federation of Automatic Control
June 7-10, 2015, Whistler, British Columbia, Canada

A Novel Algorithm for Model-Plant Mismatch Detection for Model Predictive Controllers

Y. Tsai * **R. B. Gopaluni** ** **D. Marshman** *** **T. Chmelyk** ****

* *University of British Columbia, Vancouver, V6T1Z3 BC Canada (e-mail: yttsai@chbe.ubc.ca).*
** *University of British Columbia, Vancouver, V6T1Z3 BC Canada (e-mail: gopaluni@chbe.ubc.ca).*
*** *Spartan Controls, Burnaby, V5A4X5 BC Canada (e-mail: marshman.devin@spartancontrols.com).*
**** *Spartan Controls, Vancouver, V5A4X5 BC Canada (e-mail: chmelyk.terry@spartancontrols.com).*

Abstract: For Model Predictive Controlled (MPC) applications, the quality of the plant model determines the quality of performance of the controller. Model Plant Mismatch (MPM), the discrepancies between the plant model and actual plant transfer matrix, can both improve or degrade performance, depending on the context in which performance is measured. In this paper, we do not use performance metrics or "yes-no"-type tests to merely diagnose the presence or absence of MPM in the plant matrix. Rather, we achieve the further goal of locating the exact MPM-affected elements within the plant matrix. Our proposed detection algorithm consists of two system identification experiments: the first experiment diagnoses the presence of MPM, and the second experiment pinpoints the exact MPM-affected elements. We then exercise the algorithm on artificial 3x3 and 5x5 plants suffering from sparse MPM, and demonstrate the algorithm's capability of correctly locating the MPM-affected entries.

Keywords: Modelling and Identification, Model-based Control, Process Applications

1. INTRODUCTION

Model-Predictive Control (MPC) is widely appreciated in the chemical process industries, and its superiorities over other types of controllers are as follows. First, MPC is capable of handling constraints on both the rates and magnitudes of the manipulated and output variables. Moreover, its predictive aspect allows for optimal control of deadtime-dominant processes. Despite these advantages, however, the performance of MPC depends critically on the quality of the model. Modelling errors can arise due to non-linear control behavior, time-varying plant parameters and dynamics, and drifting disturbance signals (Maciejowski, 2002). Since the real plant transfer matrix is never known exactly, a discrepancy always exists between the real plant and the plant model. This discrepancy is known as "Model-Plant Mismatch" (MPM). MPM may improve or degrade performance or control quality (Carlsson, 2010), depending on how the user defines these terms. Papers by Badwe et al. (2010) and Carlsson (2010) show that if control error magnitudes are used as a measure of "control quality," then MPM can possibly improve performance. Additionally, Jiang et al. (2012) has demonstrated that, even in the absence of MPM, MPC performance degradation can be caused by stringent MPC constraints, which prevents sufficient controller action for performance recovery. In light of these examples, performance degradation and MPM are not always correlated. Therefore, using the presence of performance degradation as an indication of MPM is unreliable. Nevertheless, in most cases, the symptom of performance degradation is realized long after significant MPM has occurred, which is why MPM detection is so important for many control engineers.

In this paper, we present a Closed-Loop algorithm with the following objectives: (1) Establish a quantitative metric to detect the existence of MPM at any time during plant operation, and (2) Identify the specific input-output model causing the MPM. We achieve Objective 1 by adapting the S_Δ method developed by Badwe et al. (2010). This test indicates the presence of MPM based on maximum singular values of the transfer matrix between the controller input and setpoint trajectories. The rows of the model matrix suffering from MPM are identified, but the specific MPM-affected elements remain unknown. If MPM is sparsely distributed along the rows of the plant transfer matrix, this experiment saves valuable time by quickly eliminating the MPM-free rows from further consideration. We accomplish Objective 2 by parameterizing an ARX model between the prediction error and controller input signals. Here, the rows marked by the S_Δ test are subjected to further experiments, in order to pinpoint the exact MPM-affected elements. Our presented approach provides significant computational savings, compared to calculating sample cross-correlations between prediction errors and controller inputs, as suggested by Webber and Gupta (2008). Although Carlsson (2010) and Badwe et al. (2009)

have suggested a partial-correlations technique which accomplishes Objectives 1 and 2, our work focuses on developing an algorithm that is more computationally friendly. Finally, we test our algorithm on artificially-constructed 3x3 and 5x5 plants to demonstrate its efficacy. Note that we only show the reliability of our algorithm under the important assumptions of persistent setpoint excitation, and a non-drifting disturbance sequence which resembles first-order-filtered white noise. One of our future goals is to develop a MPM detection algorithm that works reliably under circumstances in which these assumptions are relaxed.

2. MPM DETECTION ALGORITHM

2.1 Detecting the Existence of Row MPM: The S_Δ Test

Consider an Internal Model Control (IMC) structure with no measured disturbances at controller or plant inputs, which resembles the structure of most MPC controllers:

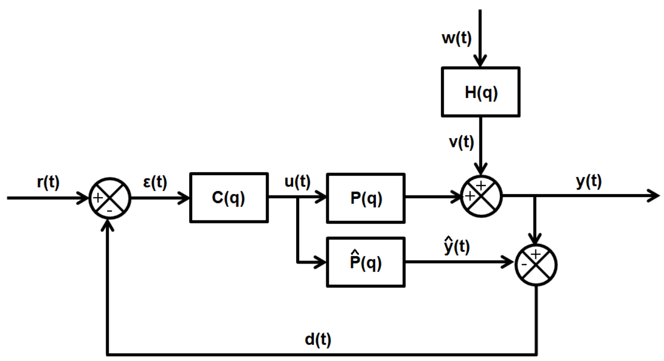

Fig. 1. Internal Model Controller (IMC) Structure

The various signals of interest include Setpoint $r(t)$, Controller Input $\epsilon(t)$, Plant Input $u(t)$, Real Output $y(t)$, Model Output $\hat{y(t)}$, Gaussian White Noise $w(t)$, First-Order-Filtered White Noise $v(t)$, and Prediction Error $d(t)$.

Moreover, the relevant transfer functions include Controller $C(q)$, Real Plant $P(q)$, Plant Model $\hat{P}(q)$, Model Plant Mismatch $\Delta(q) = P(q) - \hat{P}(q)$, and First-Order Noise Filter $H(q)$.

The noise sequence $v(t)$ can be of any form (white or coloured). To easily simulate the signals, we take $v(t)$ as white noise $w(t)$ (with zero mean and a specified covariance) passed through a first-order filter. However, for real industrial applications, any other noise form of $v(t)$ can be considered without any loss of generality, as long as it is non-drifting (Maciejowski, 2002). Badwe et al. (2010) has derived the relationships between the relevant signals for both the Single-Input, Single-Output (SISO) and Multi-Input, Multi-Output (MIMO) cases. The pertinent results of the lengthy derivations are summarized as follows:

$$\epsilon(t) = [I + \Delta(q) \cdot C(q)]^{-1} \cdot r(t) - [I + \Delta(q) \cdot C(q)]^{-1} \cdot v(t) \tag{1}$$

The term $S_\Delta(q) = [I + \Delta(q) \cdot C(q)]^{-1}$ is known as the "Relative Sensitivity". Consider the scenario of a MPM-affected plant, and define the actual (MPM-present) and design (MPM-absent) control errors $e_a(t)$ and $e_d(t)$, respectively, as:

$$e_a(t) = r(t) - y(t) = [I - \hat{P}(q) \cdot C(q)][I + \Delta(q) \cdot C(q)]^{-1} \tag{2}$$

$$\cdot [r(t) - v(t)] \tag{3}$$

$$e_d(t) = [I - \hat{P}(q) \cdot C(q)] \cdot [r(t) - v(t)] \tag{4}$$

For the sake of simplifying the manipulations, pre-filter the two control errors by $[I - \hat{P(q)} \cdot C(q)]^{-1}$:

$$e_{a_f}(t) = [I + \Delta(q) \cdot C(q)]^{-1} \cdot [r(t) - v(t)] \tag{5}$$

$$e_{d_f}(t) = r(t) - v(t) \tag{6}$$

We can mathematically express the ratio of the *relative magnitudes* of the actual error versus the designed error in the vector 2-norm, measured in the frequency domain *and within the control-relevant frequency range* Ω, as:

$$\max_{\omega \in \Omega, ||e_{d_f}(\omega)||_2 \neq 0} \frac{||e_{a_f}(\omega)||_2}{||e_{d_f}(\omega)||_2} = ||S_\triangle(\omega)||_\infty \tag{7}$$

In order to physically interpret this ratio, notice that the control error is a measure of deviation from perfect setpoint tracking. Therefore, it can be used as a *loose* measure of quality of control. A large control error indicates a large discrepancy between outputs and setpoints, which translates to an inferior quality of control, and vice versa. When no MPM exists, the actual and designed control errors will be equal; this is easily realized from Eq. 5 and Eq. 6, by substituting in $\Delta(q) = 0$. On the other hand, when MPM is present, then the actual and designed control error magnitudes will be different. The ratio of actual and designed control errors is akin to a sensitivity measure of variables compared against a best-case scenario, used frequently in other engineering contexts. For this reason, Badwe et al. (2010) terms this ratio, which is equivalent to the H_∞ norm of the $S_\Delta(q)$ transfer matrix, the "Relative Sensitivity Index" (RSI):

$$RSI \triangleq ||S_\Delta(\omega)||_\infty, \omega \in \Omega \tag{8}$$

Again, consider the case where no MPM is present (i.e. $\Delta(q) = 0$), and refer to Eq. 1. In this case, $S_\Delta(q) = I$, and therefore $||S_\Delta(\omega)||_\infty = ||I(\omega)||_\infty = 1$. On the other hand, if MPM is present, then $S_\Delta(q) \neq I$ and therefore $||S_\Delta(\omega)||_\infty \neq 1$. Therefore, a non-unity value of $||S_\Delta(\omega)||_\infty$ indicates the existence of MPM. In light of Eq. 5 and Eq. 6, if $||S_\Delta(\omega)||_\infty > 1$, then the actual control error is greater than the designed control error magnitude, indicating a degraded overall quality of control. The opposite is true for cases where $||S_\Delta(\omega)||_\infty < 1$, which implies that the actual control error magnitude is smaller than the designed control error magnitude, indicating an improved overall quality of control.

Eq. 1 suggests that we can obtain $S_\Delta(q)$ by performing Output-Error (OE) System Identification between the controller inputs $\epsilon(t)$ and setpoints $r(t)$, assuming that the setpoints $r(t)$ and disturbance sequence $v(t)$ are uncorrelated. On the other hand, if $r(t)$ and $v(t)$ are correlated, then we must use a high-order ARX model to obtain unbiased estimates of $S_\Delta(q)$ (Ljung, 1999). In terms of a frequency-domain interpretation of $S_\Delta(\omega), \omega \in \Omega$, the RSI represents the peak gain on a SISO Bode plot, or the maximum singular value on a MIMO singular value plot.

The individual rows of the $S_\Delta(q)$ matrix are the specific models describing the relationship between each controller input $\epsilon_i(t), i \in 1, 2, \cdots, n_y$ and setpoints $r(t)$. Therefore, we can compute the individual row $||S_\Delta(\omega)||_\infty$ values. Rows with non-unity $||S_\Delta(\omega)||_\infty$ values are the ones affected by MPM. Conversely, rows with $||S_\Delta(\omega)||_\infty$ values close to 1 indicate models affected by negligible MPM.

Note that the quality of the identified $S_\Delta(q)$ matrix is affected by: (1) whether the setpoint signal $r(t)$ is "sufficiently exciting," and (2) the nature of the disturbance sequence $v(t)$. According to Ljung (1999), "sufficient excitation" implies a zero-mean excitation signal $r(t)$ containing the full frequency band of 0 to f_N (where f_N is the Nyquist frequency). Obviously, this is unacceptable for most industrial plants, which instead rely on infrequent, mild setpoint changes for model identification. In these cases, the frequency bands of the excitation signals no longer contain the full range of 0 to f_N, and the estimation of S_Δ becomes imperfect, consequently diminishing the reliability of the MPM detection algorithm. Moreover, if the disturbance $v(t)$ cannot be assumed as zero-mean, Gaussian noise, the quality of the estimated S_Δ model deteriorates. We will address the technical and practical impacts of insufficient excitation and non-Gaussian disturbances in a subsequent paper.

Although S_Δ is a powerful yet simple metric which can detect the existence of MPM-affected rows in a MIMO plant matrix, it cannot exactly pinpoint the MPM-affected elements in each affected row. We accomplish this second objective using the method described in the following section.

2.2 Pinpointing the Specific Locations of MPM: Identifying the MPM Matrix Δ

After we detect the existence of MPM-affected rows in a MPC-controlled, MIMO plant transfer matrix, the next logical step is to correctly pinpoint the elements within the MPM-affected rows. The most natural approach is to identify the MPM matrix, $\Delta(q)$. Consider a MIMO plant which has n_u inputs and n_y outputs. The plant matrix is therefore n_y rows by n_u columns. Assume that the S_Δ test has determined the set of all MPM-affected rows, denoted by R. The task is to pinpoint the exact elements $\Delta_{i,j}(q), i \in R, j \in 1, \cdots, n_u$ which are non-zero. A possible method has been proposed by Webber and Gupta (2008), who detected MPM using the presence of significant cross-correlations between the prediction error $d(t)$ and an excitation signal $u_d(t)$ injected additively to the plant input $u(t)$. However, several pitfalls exist with the correlation method. First, it is computationally expensive. Correlations between each $d_i(t)$ and $u_j(t)$, $\forall i \in 1, 2, \cdots, n_y, \forall j \in 1, 2, \cdots, n_u$ require a total of $n_u \cdot n_y$ checks. This total number may be feasibly small for plants up to 5x5, but it would grow too quickly to be practical for plants of the order of 10x10 or more. Moreover, correlations are only perfect in systems operating within linear ranges (Ljung, 1999). Non-linear situations, such MPCs operating with active constraints, will cause the correlations to become imperfect. This may result in false positive or negative detections of MPM.

We present a different approach by considering the mathematical relationship between the prediction error $d(t)$, the plant input $u(t)$ and disturbance $v(t)$, which holds regardless of whether MPM is present:

$$d(t) = \Delta(q) \cdot u(t) + v(t) \tag{9}$$

To identify the transfer function $\Delta(q)$ between $d(t)$ and $u(t)$, Kano et al. (2010) explored the possibility of obtaining an unbiased FIR model of Δ using routine, closed-loop data. He concluded that this was impossible with no setpoint or input excitation, but did not address the case where setpoint or input excitation existed. If sufficient setpoint excitation is present, then we can use $d(t)$-$u(t)$ data to obtain unbiased estimates of the elements inside $\Delta(q)$. First, we single out the MPM-affected rows using the S_Δ row test described in the previous section. If MPM exists in the i_{th} control variable, then $i \in R$, and at least one $\Delta_{i,j}(q), j \in 1, 2, \cdots, n_u$ must be non-zero. In other words, $d_j(t)$ will have contribution from at least one $u_j(t), j \in 1, 2, \cdots, n_u$. Conversely, if no MPM exists in the i^{th} CV, then $i \notin R$, and all of $\Delta_{i,j}(q), j \in 1, 2, \cdots, n_u$ will be entirely zero. However, in the no-MPM case, $d(t)$ and $u(t)$ are not always uncorrelated. Carlsson (2010) provides several counter-examples that confound correlation analysis between $d(t)$ and $u(t)$. Despite of this possible pitfall, false positives are not a problem with this proposed method. If a transfer matrix row is MPM-free, then $||S_\Delta(\omega)||_\infty = 1$ for that row, thus eliminating the need for $\Delta(q)$ to be identified for said row.

Due to feedback, $u(t)$ is always correlated with $v(t)$. Therefore, we use a high-order, closed-loop ARX identification experiment (Ljung (1999)) to identify the rows of $\Delta(q)$ in which MPM exists. For each MPM-affected row, we use the signals $d_i(t), \forall i \in R$ and $u_1(t), u_2(t), \cdots, u_{n_u}(t)$ to identify the following ARX structure:

$$A_i(q) \cdot d_i(t) = B_{i,1}(q) \cdot u_1(t) + \cdots + B_{i,n_u}(q) \cdot u_{n_u}(t) + err_i(t), \tag{10}$$

$i \in R, t \in \mathbb{N}$. $err_i(t)$ is the noise term in the ARX model for the i^{th} prediction error (note that $err_i(t)$ is unrelated to the i^{th} disturbance, $v_i(t)$). Each $A(q)$ is an ARX polynomial of pre-specified order n_A:

$$A_i(q) = 1 + a_{i_1} \cdot q^{-1} + \cdots + a_{i_{n_A}} \cdot q^{-n_A} \tag{11}$$

and each $B_{i,j}(q), j \in 1, \cdots, n_u$ is an ARX polynomial of pre-specified order n_B:

$$B_{i,j}(q) = b_{i,j_1} \cdot q^{-1} + \cdots + b_{i,j_{n_B}} \cdot q^{-n_B} \tag{12}$$

Each $B_{i,j}(q)$ polynomial containing significantly non-zero coefficients indicate that $d_i(t)$ contains a contribution from $u_j(t)$, thus indicating that element $\Delta_{i,j}(q)$ contains MPM. A more mathematically rigorous definition of non-zero is required and currently under development. For now, in order to highlight the potential of this detection method, non-zero is loosely defined as a coefficient significantly greater in magnitude compared to all other coefficients (see Section 3 for examples).

The following visual example illustrates the use of a combination of S_Δ and Δ identification experiments to pinpoint MPM.

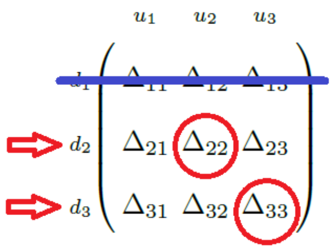

Fig. 2. Suppose that $||S_\Delta(\omega)||_\infty = 1$ for Row 1, and $||S_\Delta(\omega)||_\infty \neq 1$ for Rows 2 and 3. Then, Eq. 1 implies that Row 1 contains no MPM and can be eliminated, and Rows 2 and 3 contain MPM. Eq. 10 can be used to identify high-order ARX models for Rows 2 and 3 of $\Delta(q)$. If the polynomials $B_{2,2}(q)$ and $B_{3,3}(q)$ contain significantly non-zero coefficients, $\Delta_{2,2}(q)$ and $\Delta_{3,3}(q)$ are the MPM-affected entries.

In the worst-case scenario where MPM exists in every row, this method requires n_y experiments to identify $S_\Delta(q)$, and another n_y experiments to identify the ARX models between each $d_i(t), i \in 1, \cdots, n_y$ and $[u_1(t)\ u_2(t) \cdots u_{n_u}(t)]$, resulting in a total of $2 \cdot n_y$ experiments. This is a considerably smaller number compared to that required $(n_y \cdot n_u)$ for the correlation method suggested by Webber and Gupta (2008). In cases where MPM is sparse (i.e. not every row is MPM-affected), the proposed method results in even larger computational savings. The specific numbers of reduced computations are explored in more detail in the following sub-section.

2.3 Computational Savings

In our proposed, combined S_Δ and Δ identification experiments for MPM detection, the main advantage is the significantly reduced number of calculations required for large plants. Consider Fig. 1 for a plant with n_y setpoints, and n_u inputs.

In terms of the number of excitations required, we must excite n_u plant inputs to re-identify the plant model, if no MPM detection algorithm were to be used. On the other hand, if MPM detection were used, then we require n_y setpoint excitations to identify S_Δ, and another n_R excitations(n_R = total number of MPM-affected rows) to identify Δ. If $n_u > n_y + n_R$, then the proposed algorithm requires less excitation than plant re-identification, and vice-versa.

In terms of computational power required, however, we require $n_y \cdot n_u$ calculations to obtain all transfer matrix elements. If we use the MPM detection algorithm based on cross-correlations Webber and Gupta (2008), then we require the same number of calculations $(n_y \cdot n_u)$. On the other hand, the algorithm that we propose requires only n_y calculations to identify S_Δ, and an additional $n_R \cdot n_u$ calculations to identify Δ, for the MPM-affected rows (n_R = number of MPM-affected rows). This results in a total of $n_y + n_R \cdot n_u$ calculations for the proposed algorithm, compared to $n_y \cdot n_u$ calculations for the available alternatives. Obviously, if the plant is small, $n_y \cdot n_u \cong n_y + n_R \cdot n_u$. However, if the plant is large, then $n_y \cdot n_u >>> n_y + n_R \cdot n_u$, resulting in a dramatic reduction in required calculations. Nevertheless, if most elements in the plant matrix are believed to be affected by MPM, then complete plant re-identification should be considered.

3. SIMULATION EXAMPLES AND RESULTS

We demonstrate the feasibility and reliability of the previously proposed algorithm for MPM detection by applying it to simulation examples. First, we conduct the S_Δ test on a 3x3 plant containing no MPM, to demonstrate the absence of any false diagnosis of MPM. Then, we conduct the S_Δ test on a 3x3 plant containing sparse MPM entries, and identify Δ for the MPM-affected rows, to demonstrate the ability to correctly pinpoint all MPM-affected elements. For the final example, we exercise the algorithm on a 5x5 plant which also contains sparse MPM entries. Altogether, we show that the algorithm retains its reliability when tested on large systems, under the assumptions of sufficient setpoint excitation and non-drifting disturbance signals.

3.1 3x3 Plant: Case of No MPM

We first demonstrate the algorithm on a 3x3 ($n_u = n_y = 3$) plant artificially constructed in Matlab. The plant uses similar transfer functions as that of the famous Shell Benchmark problem, with reduced delays:

$$P(s) = \begin{pmatrix} \frac{4.05 \cdot e^{-6s}}{50s+1} & \frac{1.77 \cdot e^{-7s}}{60s+1} & \frac{5.88 \cdot e^{-6s}}{50s+1} \\ \frac{5.39 \cdot e^{-4s}}{50s+1} & \frac{5.72 \cdot e^{-3s}}{60s+1} & \frac{6.9 \cdot e^{-3s}}{40s+1} \\ \frac{4.38 \cdot e^{-5s}}{33s+1} & \frac{4.42 \cdot e^{-5s}}{44s+1} & \frac{7.2}{19s+1} \end{pmatrix} \tag{13}$$

In this case, no MPM is present, therefore $\hat{P}(s) = P(s)$. The goal is to observe whether the algorithm falsely identifies the presence of MPM, when none actually exists.

We simulate a completely unconstrained MPC controller with a sampling time of 1 sec, a prediction horizon of 100 sec, and a control horizon of 30 sec. The input weights are contained in a diagonal matrix $diag[1\ 1\ 1]$ (zero for all off-diagonal elements), the input weight rates $diag[0.1$

0.1 0.1], and the output weights $diag[1\ 1\ 1]$. We corrupt the plant output by a zero-mean measurement noise of standard deviation $5.0 \cdot 10^{-3}$, and a zero-mean unmeasured disturbance of standard deviation $7.5 \cdot 10^{-3}$ passed through a first-order filter of $diag[\frac{z}{z-0.95}\ \frac{z}{z-0.95}\ \frac{z}{z-0.95}]$. Finally, we excite all setpoint channels using Pseudo-Random Binary Sequence (PRBS) of frequency band 0 to f_N, where f_N is the Nyquist frequency.

We identify the $S_\Delta(q)$ matrix using Matlab's *arx* function, choosing 4th-order polynomials (i.e. $n_A = n_B = 4$). The B(q) polynomials contain delays of $[n_{k,B_1}\ n_{k,B_2}\ n_{k,B_3}]$, estimated by Matlab's *delayest* function. Then, the singular values of $S_\Delta(q)$ are plotted in the frequency domain for each row. Observe that since no MPM is present, both the overall and row $||S_\Delta(\omega)||_\infty$ values are essentially one (zero on a log plot).

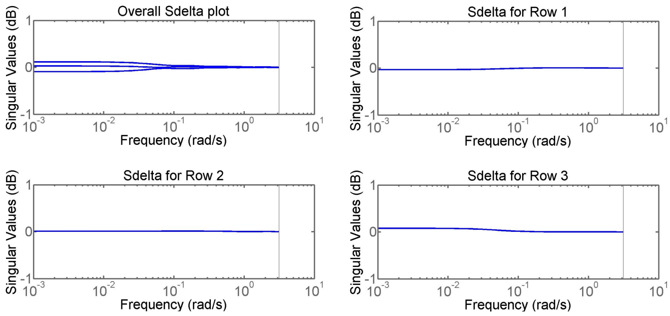

Fig. 3. Overall and row $S_\Delta(\omega)$ plots for the no-MPM case. Since $||S_\Delta(\omega)||_\infty = 1$ for every row, the S_Δ test has correctly diagnosed that every element in the model matrix is MPM-free.

Since the S_Δ test shows that no MPM is present anywhere within the plant, we need not identify Δ.

3.2 3x3 Plant: Case of Sparse MPM

Here we simulate a 3x3 plant suffering from sparse MPM, using the same MPC controller and tuning parameters as for the no-MPM case. To create the sparse MPM-affected elements, we introduce specific modelling errors on transfer matrix elements $\hat{P}_{12}(s), \hat{P}_{13}(s)$, and $\hat{P}_{21}(s)$. The plant transfer matrix remains the same as in the previous no-MPM case, while the plant model matrix now suffers from MPM (affected elements indicated in **bold**):

$$\hat{P}(s) = \begin{pmatrix} \frac{4.05 \cdot e^{-6s}}{50s+1} & \boldsymbol{\frac{1.77 \cdot e^{-4s}}{45s+1}} & \boldsymbol{\frac{3.5 \cdot e^{-6s}}{50s+1}} \\ \boldsymbol{\frac{5.39 \cdot e^{-4s}}{70s+1}} & \frac{5.72 \cdot e^{-3s}}{60s+1} & \frac{6.9 \cdot e^{-3s}}{40s+1} \\ \frac{4.38 \cdot e^{-5s}}{33s+1} & \frac{4.42 \cdot e^{-5s}}{44s+1} & \frac{7.2}{19s+1} \end{pmatrix} \quad (14)$$

In the MPM ($\Delta(q)$) matrix, $\Delta_{12}(q)$ is a combination of time constant and delay mismatch, $\Delta_{13}(q)$ a severe gain mismatch, and $\Delta_{21}(q)$ a time constant mismatch. After applying the S_Δ test, we observe non-unity values of $||S_\Delta(\omega)||_\infty$ in Rows 1 and 2, indicating the presence of MPM for these rows. For Row 3, $||S_\Delta(\omega)||_\infty = 1$, meaning that this row is MPM-free.

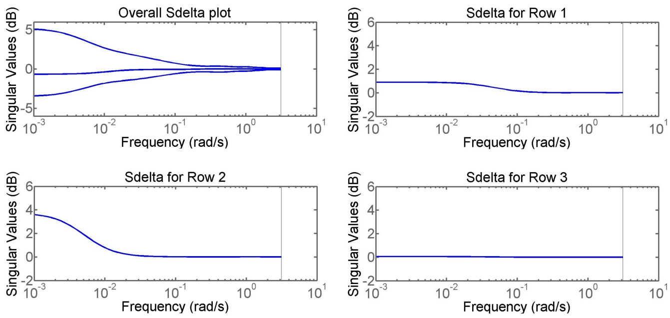

Fig. 4. Overall and row $S_\Delta(\omega)$ plots for the sparse MPM case, with modelling errors in $\hat{P}_{12}(s), \hat{P}_{13}(s)$, and $\hat{P}_{21}(s)$. $||S_\Delta(\omega)||_\infty \neq 1$ for Rows 1 and 2, and $||S_\Delta(\omega)||_\infty = 1$ for Row 3. Therefore, the S_Δ test has successfully diagnosed the presence of MPM in Rows 1 and 2, and the absence of MPM in Row 3, in the model matrix.

The question now is whether the non-zero elements of the MPM $Delta(q)$ matrix, namely $\Delta_{12}(q), \Delta_{13}(q)$, and $\Delta_{21}(q)$, can be correctly located. To confirm this, we identify the rows of the Δ matrix using signals d_1, $[u_1\ u_2\ u_3]$, and d_2, $[u_1\ u_2\ u_3]$, again choosing 4^{th} order ARX polynomials ($n_A = n_B = 4$) for the models. A delay of k units simply indicates a multiplication of each ARX coefficient with q^{-k}:

Table 1. ARX model between d_1 and u

Polynomial	q^{-1}	q^{-2}	q^{-3}	q^{-4}	Delay	Fit(%)
$A(q)$	$-9.534 \cdot 10^{-1}$	$-1.877 \cdot 10^{-2}$	$-4.068 \cdot 10^{-3}$	$-2.605 \cdot 10^{-3}$	0	91.06
$B_{u_1}(q)$	$-2.084 \cdot 10^{-4}$	$6.264 \cdot 10^{-4}$	$1.047 \cdot 10^{-4}$	$4.149 \cdot 10^{-4}$	7	
$B_{u_2}(q)$	$\boldsymbol{-3.923 \cdot 10^{-2}}$	$-1.595 \cdot 10^{-3}$	$-8.174 \cdot 10^{-4}$	$\boldsymbol{2.856 \cdot 10^{-2}}$	5	
$B_{u_3}(q)$	$-6.525 \cdot 10^{-4}$	$-3.240 \cdot 10^{-4}$	$\boldsymbol{4.667 \cdot 10^{-2}}$	$9.816 \cdot 10^{-4}$	5	

Table 2. ARX model between d_2 and u

Polynomial	q^{-1}	q^{-2}	q^{-3}	q^{-4}	Delay	Fit(%)
$A(q)$	$-9.969 \cdot 10^{-1}$	$-3.565 \cdot 10^{-2}$	$-4.611 \cdot 10^{-2}$	$-9.971 \cdot 10^{-3}$	0	93.07
$B_{u_1}(q)$	$\boldsymbol{-6.943 \cdot 10^{-3}}$	$-1.945 \cdot 10^{-4}$	$1.300 \cdot 10^{-4}$	$-5.290 \cdot 10^{-4}$	5	
$B_{u_2}(q)$	$6.211 \cdot 10^{-4}$	$2.838 \cdot 10^{-4}$	$2.922 \cdot 10^{-4}$	$7.952 \cdot 10^{-4}$	5	
$B_{u_3}(q)$	$6.971 \cdot 10^{-4}$	$1.867 \cdot 10^{-4}$	$-1.914 \cdot 10^{-4}$	$2.671 \cdot 10^{-4}$	2	

In each table, we highlight significantly non-zero coefficients in **bold**. Notice that the quality of the ARX model fits are extremely high. In the ARX model between $d_1(t)$ and $u(t)$ (Table 1), the polynomials $B_{u_2}(q)$ and $B_{u_3}(q)$ contain significantly non-zero coefficients, compared to all other coefficients. Similarly, in the model between $d_2(t)$ and $u(t)$ (Table 2), the polynomial $B_{u_1}(q)$ contains a significantly non-zero coefficient compared to all others. These results indicate that $\Delta_{12}(q)$, $\Delta_{13}(q)$, and $\Delta_{21}(q)$ are the non-zero elements in the $\Delta(q)$ matrix, which means that $\hat{P}_{12}(s), \hat{P}_{13}(s)$, and $\hat{P}_{21}(s)$ have been correctly identified as the MPM-affected elements.

3.3 5x5 Plant: Case of Sparse MPM

For the final simulation example, we exercise the algorithm on a 5x5 plant suffering from sparse MPM. Specifically, we introduce modelling errors in elements $\hat{P}_{11}(s), \hat{P}_{21}(s), \hat{P}_{23}(s), \hat{P}_{42}(s)$, and $\hat{P}_{45}(s)$. Suppose all 25 elements of the true plant transfer matrix are $\frac{5 \cdot e^{-5s}}{50s+1}$, without loss of generality. The model matrix contains the following elements (with the MPM-affected elements highlighted in **bold**):

$$\hat{P}(s) = \begin{pmatrix} \dfrac{\mathbf{5 \cdot e^{-8s}}}{\mathbf{50s+1}} & \dfrac{5 \cdot e^{-5s}}{50s+1} & \dfrac{5 \cdot e^{-5s}}{50s+1} & \dfrac{5 \cdot e^{-5s}}{50s+1} & \dfrac{5 \cdot e^{-5s}}{50s+1} \\ \dfrac{\mathbf{4 \cdot e^{-5s}}}{\mathbf{60s+1}} & \dfrac{5 \cdot e^{-5s}}{50s+1} & \dfrac{\mathbf{5 \cdot e^{-5s}}}{\mathbf{25s+1}} & \dfrac{5 \cdot e^{-5s}}{50s+1} & \dfrac{5 \cdot e^{-5s}}{50s+1} \\ \dfrac{5 \cdot e^{-5s}}{50s+1} & \dfrac{5 \cdot e^{-5s}}{50s+1} & \dfrac{5 \cdot e^{-5s}}{50s+1} & \dfrac{5 \cdot e^{-5s}}{50s+1} & \dfrac{5 \cdot e^{-5s}}{50s+1} \\ \dfrac{5 \cdot e^{-5s}}{50s+1} & \dfrac{\mathbf{3.5 \cdot e^{-5s}}}{\mathbf{50s+1}} & \dfrac{5 \cdot e^{-5s}}{50s+1} & \dfrac{5 \cdot e^{-5s}}{50s+1} & \dfrac{\mathbf{5 \cdot e^{-3s}}}{\mathbf{75s+1}} \\ \dfrac{5 \cdot e^{-5s}}{50s+1} & \dfrac{5 \cdot e^{-5s}}{50s+1} & \dfrac{5 \cdot e^{-5s}}{50s+1} & \dfrac{5 \cdot e^{-5s}}{50s+1} & \dfrac{5 \cdot e^{-5s}}{50s+1} \end{pmatrix} \quad (15)$$

We use similar MPC controllers and reference signals as in the two previous 3x3 plant examples. To detect MPM, we perform the S_Δ test on the plant. Rows 1, 2, and 4 yield non-unity values of $||S_\Delta(\omega)||_\infty$, showing the correct diagnosis of MPM within these rows. In Rows 1 and 2, the MPMs consist of severe time constant and delay mismatches, while the gain mismatches are mild. Therefore the peaks of $S_\Delta(\omega)$ are smaller in these rows, compared to Row 4 where a severe gain mismatch exists in element Δ_{42}. Rows 3 and 5 yield $||S_\Delta(\omega)||_\infty$ values of 1, showing the correct diagnosis of absence of MPM in these rows.

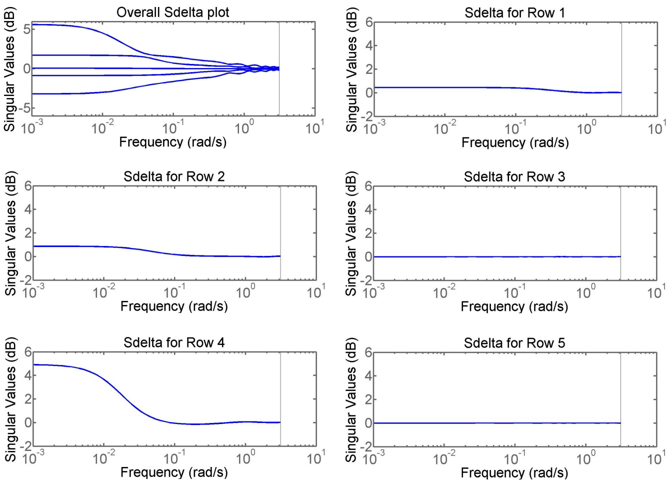

Fig. 5. Overall and row $S_\Delta(\omega)$ plots for a 5x5 plant suffering from sparse MPM. $||S_\Delta(\omega)||_\infty \neq 1$ for Rows 1, 2, and 4, and $||S_\Delta(\omega)||_\infty = 1$ for Rows 3 and 5. Therefore, we correctly detect the presence of MPM in Rows 1, 2, and 4, and the absence of MPM in Rows 3 and 5.

Next, we identify the rows of $\Delta(q)$ between each MPM-affected prediction error $d_1(t)$, $d_2(t)$, and $d_4(t)$, and inputs $u_1(t) \cdots u_5(t)$. The 4th-order ARX polynomial coefficients are tabulated below, again with the significantly non-zero elements highlighted in **bold**:

Table 3. 5x5 plant: ARX model – d_1, u

Polynomial	q^{-1}	q^{-2}	q^{-3}	q^{-4}	Delay	Fit(%)
$A(q)$	$-9.802 \cdot 10^{-1}$	$-8.626 \cdot 10^{-7}$	$2.902 \cdot 10^{-7}$	$2.810 \cdot 10^{-6}$	0	99.98
$B_{u_1}(q)$	$\mathbf{9.901 \cdot 10^{-2}}$	$1.501 \cdot 10^{-7}$	$1.452 \cdot 10^{-6}$	$\mathbf{-9.901 \cdot 10^{-2}}$	6	
$B_{u_2}(q)$	$4.441 \cdot 10^{-6}$	$1.210 \cdot 10^{-6}$	$-6.071 \cdot 10^{-7}$	$-9.904 \cdot 10^{-6}$	0	
$B_{u_3}(q)$	$-2.929 \cdot 10^{-7}$	$1.675 \cdot 10^{-7}$	$-7.289 \cdot 10^{-9}$	$2.825 \cdot 10^{-7}$	0	
$B_{u_4}(q)$	$4.946 \cdot 10^{-6}$	$1.550 \cdot 10^{-6}$	$-5.482 \cdot 10^{-7}$	$-1.046 \cdot 10^{-5}$	0	
$B_{u_5}(q)$	$-8.064 \cdot 10^{-6}$	$-2.030 \cdot 10^{-6}$	$1.444 \cdot 10^{-6}$	$1.870 \cdot 10^{-5}$	0	

Table 4. 5x5 plant: ARX model – d_2, u

Polynomial	q^{-1}	q^{-2}	q^{-3}	q^{-4}	Delay	Fit(%)
$A(q)$	-2.924	2.851	$-9.262 \cdot 10^{-1}$	$-1.426 \cdot 10^{-6}$	0	93.32
$B_{u_1}(q)$	$6.365 \cdot 10^{-9}$	$\mathbf{3.289 \cdot 10^{-2}}$	$\mathbf{6.417 \cdot 10^{-2}}$	$\mathbf{3.129 \cdot 10^{-2}}$	5	
$B_{u_2}(q)$	$1.774 \cdot 10^{-6}$	$-2.009 \cdot 10^{-6}$	$-1.048 \cdot 10^{-6}$	$-1.458 \cdot 10^{-6}$	0	
$B_{u_3}(q)$	$-7.731 \cdot 10^{-8}$	$\mathbf{-9.705 \cdot 10^{-2}}$	$\mathbf{1.925 \cdot 10^{-1}}$	$\mathbf{-9.544 \cdot 10^{-2}}$	5	
$B_{u_4}(q)$	$2.023 \cdot 10^{-6}$	$-2.440 \cdot 10^{-6}$	$-1.154 \cdot 10^{-6}$	$-1.517 \cdot 10^{-6}$	0	
$B_{u_5}(q)$	$-3.422 \cdot 10^{-6}$	$3.997 \cdot 10^{-6}$	$1.994 \cdot 10^{-6}$	$2.690 \cdot 10^{-6}$	0	

Table 5. 5x5 plant: ARX model – d_4, u

Polynomial	q^{-1}	q^{-2}	q^{-3}	q^{-4}	Delay	Fit(%)
$A(q)$	-1.967	$9.672 \cdot 10^{-1}$	$6.931 \cdot 10^{-7}$	$-2.757 \cdot 10^{-7}$	0	95.78
$B_{u_1}(q)$	$-4.219 \cdot 10^{-7}$	$1.023 \cdot 10^{-5}$	$-1.302 \cdot 10^{-5}$	$2.554 \cdot 10^{-6}$	0	
$B_{u_2}(q)$	$1.272 \cdot 10^{-7}$	$-4.88 \cdot 10^{-9}$	$\mathbf{2.970 \cdot 10^{-2}}$	$\mathbf{-2.931 \cdot 10^{-2}}$	4	
$B_{u_3}(q)$	$-3.961 \cdot 10^{-7}$	$9.853 \cdot 10^{-6}$	$-1.225 \cdot 10^{-5}$	$2.561 \cdot 10^{-6}$	0	
$B_{u_4}(q)$	$7.261 \cdot 10^{-7}$	$-1.877 \cdot 10^{-5}$	$2.380 \cdot 10^{-5}$	$-5.191 \cdot 10^{-6}$	0	
$B_{u_5}(q)$	$\mathbf{-6.622 \cdot 10^{-2}}$	$\mathbf{6.491 \cdot 10^{-2}}$	$\mathbf{9.901 \cdot 10^{-2}}$	$\mathbf{9.770 \cdot 10^{-2}}$	4	

In the ARX model between $d_1(t)$ and $u(t)$ (Table 3), the polynomials $B_{u_1}(q)$ contain significant non-zero coefficients, compared to all others. This is also the case in the model between $d_2(t)$ and $u(t)$ (Table 4) for the polynomials $B_{u_1}(q)$ and $B_{u_3}(q)$, and in the model between $d_4(t)$ and $u(t)$ (Table 5) for the polynomials $B_{u_2}(q)$ and $B_{u_5}(q)$. These results indicate that by identifying $\Delta(q)$, we have correctly located the MPM (Δ) matrix entries that are non-zero, namely $\Delta_{11}(q)$, $\Delta_{21}(q)$, $\Delta_{23}(q)$, $\Delta_{42}(q)$, and $\Delta_{45}(q)$.

4. LIMITATIONS

In this final section, we outline the main theoretical and practical limitations that this MPM detection algorithm possesses, with respect to the control engineers who will be the likely users of this algorithm.

The first two important assumptions are sufficient set-point excitation and zero-mean, first-order-filtered noise disturbance. If any of these assumptions are relaxed, the proposed algorithm may fail to detect MPM reliably. False positives and missed negatives may be observed, since the $S_\Delta(q)$ and $\Delta(q)$ matrices are no longer fully identifiable. Iqbal et al. (2014) and Badwe et al. (2009) have both suggested MPM detection approaches using partial correlations between prediction errors d and inputs u. For these methods, the amount of excitation required does not exceed that provided by typical bump tests. Therefore, we will consider adapting these methods in future work concerning the relaxation of previously stated assumptions. Moreover, we use the presence of non-zero ARX polynomial coefficients in the $\Delta(q)$ matrix to detect whether the prediction errors $d(t)$ contains a contribution from an input $u_j(t), j \in 1, \cdots, n_u$. Currently, we define non-zero by a magnitude that is greater than all other coefficients. However, when the plant becomes corrupted by non-white or non first-order-filtered noise, the qualities of the ARX models will deteriorate, and a more reliable measure of non-zero is required. One preliminary idea is to use the mean and covariances of each estimated ARX coefficient to generate a confidence interval, which will help determine whether said coefficient is non-zero. Finally, in order to obtain high-quality fits for the S_Δ and Δ models, we require accurate estimates of delays (between ϵ and r, and between d and u, respectively). This justifies the need for a computationally inexpensive delay estimation technique for plants larger than 5x5. We are focusing present efforts on using constrained 1-norm (L1) minimization procedures, which determine signal input delays using FIR coefficients.

From a practical perspective, our proposed method only diagnoses the exact locations of MPM-affected elements. However, if the type (gain, time constant, time delay) corresponding to each MPM, as well as the relative severity of each type were also determined, then the control engineer could make educated decisions on the effort involved in controller re-tuning. For instance, delay and time constant mismatches can be more malignant compared to gain mismatches. The reason behind this is that MPM contributions to performance degradation are rectified by changing model gains incrementally, in order to recover performance. However, the same cannot be done easily with time constants and delays, making these types of MPM more undesirable than gain MPMs. Additionally, the type and severity indices can also provide physical insight to the causes behind the observed MPM. For example, if only a gain MPM exists, the MPM can be safely associated with process parameter changes over time, while the process remains in a linear range of operation (Olivier and Craig, 2013). The controller can be easily re-tuned by incrementally re-adjusting the gain without having to re-identify the entire plant model. On the other hand, time constant and delay MPMs indicate process dynamic changes over time, which are significantly more difficult to rectify. A severity index comparison would allow an educated decision on whether the re-tuning of few transfer elements is feasible, as opposed to re-identifying the entire plant model.

5. CONCLUSION

In this paper, we have presented the mathematical details behind the proposed 2-step algorithm of MPM detection, which pinpoints the exact elements in a plant transfer matrix that are affected by MPM. In the first step, we diagnose the presence (or lack) of MPM in each plant matrix row using the S_Δ test, a metric which measures the H_∞ norm of the ratio between the actual and designed control errors. The $S_\Delta(q)$ matrix is identified between the plant's controller inputs $\epsilon(t)$ and setpoints $r(t)$. In the second step, the MPM-affected rows of the plant matrix are singled out, and the $\Delta(q)$ identification experiment is performed. Specifically, we conduct ARX experiments between the MPM-affected prediction errors $d(t)$ and all inputs $u(t)$. By doing so, we identify the non-zero entries in the MPM matrix $\Delta(q)$, and hence the specific inputs causing MPM. The most important underlying assumptions of the proposed algorithm are that the setpoint be sufficiently exciting (either white noise or PRBS with frequency range 0-f_N, Nyquist frequency), and that the disturbance sequence be first-order filtered, zero-mean, non-drifting white noise. If these assumptions no longer apply, then the reliability of the proposed algorithm becomes questionable. We will explore the consequences of these scenarios in a future paper, as well as develop new, reliable identification techniques under these circumstances. Finally, we have demonstrated efficacy of this algorithm on artificial 3x3 and 5x5 plants suffering from sparse MPM. In every case, the S_Δ test has correctly identified the MPM-affected CVs, and the Δ identification experiment has correctly pinpointed the exact MPM-affected elements in every MPM-affected row of the plant matrix.

ACKNOWLEDGEMENTS

The writers would like to extend profound gratitude towards Spartan Controls, for providing practical insight and expertise with cutting-edge process control software and methods used in industry, as well as NSERC for its generous financial support for this project.

REFERENCES

Badwe, A., Gudi, R., and Patwardhan, R. (2009). Detection of model-plant mismatch in mpc applications. Journal of Process Control, 19, 31305–1313.

Badwe, A., Patwardhan, R., and Shah, S. (2010). Quantifying the impact of model-plant mismatch on controller performance. Journal of Process Control, 20, 408–425.

Carlsson, R. (2010). A practical approach to detection of plant model mismatch for mpc.

Iqbal, N., Yusoff, N., and Tufa, L. (2014). Comparison between arx and fir decorrelation methods in detecting model-plant mismatch. Journal of Applied Sciences 14, 15, 1711–1719.

Jiang, H., Shah, S., and Huang, B. (2012). Model analysis and performance analysis of two industrial mpcs. ScienceDirect: Control Engineering Practice, 20, 219–235.

Kano, M., Shigi, Y., and Hasabe, S. (2010). Detection of significant model-plant mismatch from routine operation data of model predictive control system. International Symposium on Dynamics and Control of Process Systems.

Ljung, L. (1999). System Identification: Theory for the User. Prentice Hall, New Jersey.

Maciejowski, J. (2002). Predictive Control with Constraints. Prentice Hall, London.

Olivier, L. and Craig, I. (2013). Model-plant mismatch detection and model update for a run-of-mine ore milling circuit under model predictive control. Journal of Process Control, 23, 100–107.

Webber, J. and Gupta, Y. (2008). A closed-loop cross-correlation method for detecting model mismatch in mimo model-based controllers. ISA Transactions, 47, 395–400.