User Friendly Robust MPC Tuning of Uncertain Paper-Making Processes *

Ning He^{*} Dawei Shi^{*} Jiadong Wang^{*} Michael Forbes^{**} Johan Backström^{**} Tongwen Chen^{*}

 * Department of Electrical and Computer Engineering, University of Alberta, Edmonton, Alberta, Canada T6G 2V4 (e-mail: nhe, dshi, jiadong, tchen@ualberta.ca)
 ** Honeywell Process Solutions, 500 Brooksbank Avenue, North Vancouver, BC, Canada, V7J 3S4(e-mail: michael.forbes,

johan.backstrom@honeywell.com)

Abstract: A robust tuning problem for a two-degree-of-freedom model predictive controller is explored for single-input, single-output uncertain paper-making processes. The objective is to achieve satisfactory closed-loop responses, as measured by overshoots, settling times and output oscillations with user-specified parametric uncertainties. As the output variation cannot be easily specified by the end users, two methods are proposed to connect a total variation specification to user-friendly indices, based on which two algorithms are designed to solve the tuning problem. An application to a process extracted from the pulp and paper industry is employed to verify the effectiveness of the proposed algorithms.

Keywords: Controller tuning, Model predictive control, Robust performance, Pulp and paper, Total variation

1. INTRODUCTION

In the pulp and paper industry, two types of control problems are involved in the papermaking process: machine directional (MD) control and cross directional (CD) control; the objectives of which are to ensure that the paper products satisfy certain quality requirements (Chu, Forbes, Backström, Gheorghe, and Chu, 2011). The controller tuning of the MD Model Predictive Control (MPC) then becomes an important factor in successful paper production. However, the existing tuning approaches normally focus on objectives provided in a norm space (Qin and Badgwell, 2003), which is not intuitive for the end users to understand and specify compared with the time domain performance requirements, e.g., overshoots and settling times. Additionally, model uncertainty, which is unavoidable in process operation and modeling, is normally considered as unstructured uncertainty, which is also not familiar to the end users compared with parametric uncertainty.

Considering these facts, a robust MPC tuning approach that guarantees the required time domain performance with parametric uncertainty is developed in this work. A large number of approaches for the MPC tuning problem have been reported in the literature. The first type of the tuning approaches achieved the desired closed-loop performance by matching the closed-loop controller or performance with a pre-assigned controller or performance, see e.g., Di Cairano and Bemporad (2010) and Shah and Engell (2010). Another type of approaches investigated the relationship between the system outputs and the effect of MPC tuning parameters by approximation, see e.g., Al-Ghazzawi, Ali, Nouh, and Zafiriou (2001) and Garriga and Soroush (2008). Besides, some analytical approaches were proposed in Wojsznis, Gudaz, Blevins, and Mehta (2003) and Bagheri and Sedigh (2013). In the above MPC tuning approaches, only a small number of them take model uncertainty into account. In Han, Zhao, and Qian (2006), the authors employed a min-max strategy to handle model uncertainty explicitly, which could achieve strong robustness and small overshoots. Júnior, Martins, and Kalid (2014) proposed an optimal tuning approach based on particle swarm optimization, in which the Morari resiliency index and the condition number were applied as the performance measure. In Tran, Ozkan, and Backx (2012), the tuning parameters were computed by searching for an optimal bandwidth that gave a trade-off between robustness and nominal performance. In Garriga and Soroush (2010), an in-depth review of the results on MPC tuning was provided. Despite of the progress made in the area, an easyto-use robust MPC tuning approach is still missing and is desired in the pulp and paper industry.

The starting point of this work is the MPC tuning structure in Chu, Forbes, and Backström (2013); the tuning problem is formulated as a two-degree-of-freedom (2-DOF) optimization problem. This framework is also employed in Shi, Wang, Forbes, Backström, and Chen (2014), the purpose of which amounts to automatic computation of the tuning parameters of the MD-MPC to achieve performance requirements on worst-case overshoots and settling times. Although the almost-optimal tuning results can be obtained by the algorithm in Shi et al. (2014), the responses can be oscillatory as aggressive control signals are needed to achieve the smallest setting time without limiting the variations in the process response. These oscillations lead to additional wear and tear of the control valves and make the system more sensitive to actuator saturation (Shi, Wang, and Ma, 2011), causing performance downgrade as well as the increase of the maintenance cost. To overcome this difficulty, the requirement on total variation is taken into account in this work. Compared with classical performance indexes like overshoots and

^{*} This work was supported by an NSERC CRD project with Honeywell Vancouver as the industrial partner.

settling times, the total variation is normally not familiar to the end users, and thus it is neither suitable nor userfriendly to invite the end users to manually specify the requirement on total variation. Therefore, the specification on total variation is implicitly made according to the requirement on either overshoots or other familiar time domain performance measures in this work.

The main contributions are summarized as follows:

- Two methods to specify the output oscillation via overshoots and decay ratios are designed. The method based on overshoots leads to a simple and fast tuning algorithm while the method via decay ratios can significantly reduce the conservativeness of tuning.
- Two efficient contour-line based parameter autotuning algorithms under user-specified parametric uncertainties are proposed by using the unimodality and monotonicity properties of the conflicting time response measures. The efficiency of the tuning algorithms is verified on process models used for MD-MPC of a paper machine at an industrial site.

2. PRELIMINARIES AND PROBLEM FORMULATION

In the MD-MPC control problem, the 2-DOF MPC control structure and its components proposed in Chu et al. (2013) are first introduced (see Fig. 1), and then the tuning problem is formulated.



Fig. 1. The 2-DOF MPC control system.

2.1 MD Model and Model Uncertainty

Normally, it is the preferred industrial practice to model the SISO MD process (G_p in Fig. 1) using a FOPDT model structure:

$$G_p(s) = \frac{g}{T_p s + 1} e^{-t_d s},\tag{1}$$

where g, T_p and t_d denote the MD process gain, time constant and time delay, respectively. The discrete model is $G_p(z) = g \frac{bz^{-1}}{1-az^{-1}} z^{-T_d}$ ($a = e^{-\Delta T/T_p}$, b = 1-a, T_d is the discretized version of t_d , and ΔT is the sampling period). Since $G_p(s)$ cannot be exactly known, a MD model $G_0(s)$ is identified to approximate $G_p(s)$:

$$G_0(s) = \frac{g_0}{T_{p0}s + 1}e^{-t_{d0}s}.$$
(2)

The MD model parameters g_0 , T_{p0} , and t_{d0} are identified via the input/output data of the real process, and is used to predict the state for the MD-MPC; the discrete model of $G_0(s)$ is obtained in the same way as that of $G_p(s)$. However, it is inevitable that the identified model $G_0(s)$ is different from $G_p(s)$. To take into account the model mismatch, the parametric uncertainty is used, which refers to the difference in the model parameters, namely,

$$g \in [\underline{g}, \overline{g}], \ T_p \in [\underline{T}_p, \overline{T}_p], t_d \in [\underline{t}_d, \overline{t}_d],$$
 (3)

Copyright © 2015 IFAC

based on which a set of possible perturbed plant models can be denoted as

 $\Pi := \left\{ G_p(s) : g \in [\underline{g}, \overline{g}], T_p \in [\underline{T}_p, \overline{T}_p], t_d \in [\underline{t}_d, \overline{t}_d] \right\}.$ (4) The parametric uncertainty is employed in this work, because it is easier for the end users to understand and specify compared with the unstructured uncertainty, which requires additional knowledge of robust control theory.

2.2 MPC formulation

In the 2-DOF MPC structure, the MD-MPC controller basically accounts to the following quadratic programming problem

$$\min_{\Delta \mathbf{U}} \quad \mathbf{J} = (\hat{\mathbf{Y}} - \mathbf{Y}_{ref})^T Q_1 (\hat{\mathbf{Y}} - \mathbf{Y}_{ref}) + \Delta \mathbf{U}^T Q_2 \Delta \mathbf{U}$$

$$+ (\mathbf{U} - \mathbf{U}_{ref})^T Q_3 (\mathbf{U} - \mathbf{U}_{ref})$$
s.t. $\hat{x}(k+i) = A^i \hat{x}(k) + \sum_{j=1}^{\min\{\mathrm{H}_u, i\}} A^{i-j} B \Delta u(k+j-1),$

$$\hat{y}(k+i) = C\hat{x}(k+i), \text{ for } i = 1, 2, \dots, \mathbf{H}_{p}$$

where

$$\hat{\mathbf{Y}} = \begin{bmatrix} \hat{y}(k+1)\\ \hat{y}(k+2)\\ \vdots\\ \hat{y}(k+\mathbf{H}_{p}) \end{bmatrix}, \Delta \mathbf{U} = \begin{bmatrix} \Delta u(k)\\ \Delta u(k+1)\\ \vdots\\ \Delta u(k+\mathbf{H}_{u}-1) \end{bmatrix}, \quad (5)$$
$$\mathbf{U} = \begin{bmatrix} 1\\ 1\\ \vdots\\ 1 \end{bmatrix} u(k-1) + \begin{bmatrix} 1 & 0 & \cdots & 0\\ 1 & 1 & \ddots & \vdots\\ \vdots & \ddots & \ddots & 0\\ 1 & \cdots & 1 & 1 \end{bmatrix} \Delta \mathbf{U}, \quad (6)$$

where H_p and H_u are prediction and control horizon, U_{ref} and \mathbf{Y}_{ref} are the reference signal vector of \mathbf{U} and $\hat{\mathbf{Y}}$, respectively, and Q_1 , Q_2 and Q_3 are weighting matrices. In the paper-making MPC tuning problem, the constraints are normally first ignored and will be re-introduced and checked after obtaining of the tuning parameters (Chu et al., 2011).

2.3 2-DOF Tuning Structure

In the 2-DOF MPC control system, the filters F_r and F_d also form an essential part. These filters are employed for filtering the output target, $y_{tgt}(k)$, and the estimated disturbance, $\hat{d}(k) := y(k) - \hat{y}(k)$, respectively. With the filtered signals, the reference trajectory is obtained as below:

$$\mathbf{Y}_{\mathrm{ref}}(k) = \begin{bmatrix} y_{\mathrm{ref}}(k+1) \\ \vdots \\ y_{\mathrm{ref}}(k+\mathrm{H}_{\mathrm{p}}) \end{bmatrix} = F_r y_{tgt}(k) - F_d \hat{d}(k).$$

 F_r and F_d are projection filters generated with $f_r(z)$ and $f_d(z)$, based on $y_{ref}(z) = f_r(z)y_{tgt}(z) - f_d(z)\hat{d}(z)$ (Chu et al., 2013); $f_r(z)$ and $f_d(z)$ are the so-called reference tracking filter and disturbance rejection filter:

$$f_r(z) = \frac{b_r z^{-1}}{1 - a_r z^{-1}} z^{-T_{d0}}, \ f_d(z) = \frac{b_d z^{-1}}{1 - a_d z^{-1}} z^{-T_{d0}}, \ (7)$$

where $a_r = e^{-\frac{\Delta T}{\lambda T_{p0}}}$, $b_r = 1 - a_r$, $a_d = e^{-\frac{\Delta T}{\lambda_d T_{p0}}}$, $b_d = 1 - a_d$, ΔT is the sampling period and T_{d0} is the discretized version of t_{d0} . Thus, the MPC performance can be adjusted by tuning λ and λ_d (which we also refer to as λ -parameters hereafter) and setting $Q_1 = I$, $Q_2 = Q_3 = 0$, which simplifies the tuning problem.

2.4 Performance Measures and Tuning Problem

Overshoots and settling times are used as the main performance measures for MPC tuning in this work, as they are straightforward and well-suited for practitioners to evaluate control performance. Since the performance concerned here is for a set of perturbed systems in Π , the worstcase performance has to be considered. The worst-case overshoot and settling time are defined as follows.

Definition 1. (Worst-case overshoot). The worst-case overshoot OS of a set of step responses with the same final value is the maximum value in all responses minus the final value divided by the final value.

Definition 2. (Worst-case settling time). The worst-case settling time T_s of a set of step responses with the same final value is the time required for all the responses to reach and stay within a range of a pre-specified percentage of the final value.

Since the traditional approach is not applicable to the characterization of the OS and T_s for a set of systems, a heuristic approach introduced in Shi et al. (2014) is employed, which obtains the OS and T_s via the following eight extreme-case systems:

$$\Pi_E := \left\{ G_p(s) : g \in \{\underline{g}, \overline{g}\}, T_p \in \{\underline{T}_p, \overline{T}_p\}, t_d \in \{\underline{t}_d, \overline{t}_d\} \right\}.$$
(8)

Since OS and T_s depend on the values of λ and λ_d in the tuning problems, the notations $OS(\lambda, \lambda_d)$ and $T_s(\lambda, \lambda_d)$ can be used to express these relationships.

Another key time-domain performance index is the total variation (Skogestad and Postlethwaite, 1996), which measures the output oscillation for general systems. Mathematically, it is defined as

$$tv := \sup_{\Delta T} \sum_{k=0}^{\infty} |y(k+1) - y(k)|.$$
 (9)

For ease of implementation, we may ignore the effect of the sampling time and replace ∞ by n, assuming that the system converges to the target value within n steps. Based on these simplifications, the worst-case total variation can be defined as below:

Definition 3. (Worst-case total variation).

$$TV := \max_{G_p \in \Pi} \sum_{k=1}^{n} |y(k) - y(k-1)|.$$
 (10)

As the extreme behavior of step responses mostly happens at the extreme process parameters, the OS, T_s as well as TV can be respectively approximated using the worst overshoot, the worst settling time and the worst total variation of the extreme-case systems.

In this work, the tuning objective is to determine λ and λ_d so that the closed-loop system in Fig. 1 is robustly stable and the output tracks its target with a fast response, a small overshoot and small output oscillation. However, there exist multiple conflicts in achieving the targets. For example, a small overshoot often results in a large settling time; while a small settling time can be associated with a large overshoot and a large total variation. To make a tradeoff in the tuning process, we tune λ and λ_d by minimizing the settling time with the resultant OS and TV lie in certain tolerable regions. Mathematically, it can be formulated into the following optimization problem:

$$\min_{\lambda,\lambda_d} T_s(\lambda,\lambda_d)
s.t. OS(\lambda,\lambda_d) \le OS^*, TV(\lambda,\lambda_d) \le TV^*,$$
(11)

Copyright © 2015 IFAC

where OS^* and TV^* refer to the specifications on $OS(\lambda, \lambda_d)$ and $TV(\lambda, \lambda_d)$, respectively.

Apart from the difficulties in solving this problem, the choice of TV^{*} is a nontrivial issue. Unlike the specification of the worst-case overshoot OS^{*} (which can be intuitively chosen, for example, 10% by the end users according to their requirements), the appropriate value of TV^* is not easy to be determined by the end users. Thus, TV^* should be determined either automatically or based on the specification of decay ratios (another well-understood performance measure in quality control in the pulp and paper industry), to maintain the user friendliness of the proposed tuning algorithms.

3. EFFICIENT TUNING WITH TIME-DOMAIN PERFORMANCE

The first approach to the optimization problem in (11)is presented in this section, and the TV^* is determined automatically in this approach. As the constraint on total variation is introduced to limit the potential oscillations in the responses, it is possible to design the value of TV^{*} based on OS^* , for which we have the following relationship. Due to space limitations, the proof is not shown here.

Proposition 4. If $TV(\lambda, \lambda_d) \leq TV^*$ and $OS(\lambda, \lambda_d) > 0$, then С

$$DS(\lambda, \lambda_d) \le (TV^* - 1)/2.$$
(12)

Proposition 4 indicates that the specification on TV^* can be chosen as $1 + 2OS^*$ to guarantee a smooth response according to the specification on OS^{*}. Note that by choosing $TV^* = 1 + 2OS^*$, the requirement on overshoots can be fulfilled, which simplifies the problem in (11). This further reduces the requirement of users' knowledge on the process, as only the requirement on overshoot OS^* is needed, which is normally familiar to the end-users of a commercial quality control software.

Thus, the problem in (11) now reduces to

$$\min_{\substack{\lambda,\lambda_d}} T_s(\lambda,\lambda_d)
s.t. TV(\lambda,\lambda_d) \le TV^*,$$
(13)

where $TV^* = 1 + 2OS^*$.

3.1 Empirical Monotonicity Properties of TV with Respect to λ and λ_d

The empirical unimodality and monotonicity properties of TV with respect to λ and λ_d are investigated and utilized to solve the tuning problem in (13), because analytical expressions of TV, even for standard second-order systems, seem not to exist (Shi et al., 2014). As λ controls the speed of the response, a larger value of λ leads to a smoother response and thus a smaller total variation. In this regard, $TV(\lambda, \lambda_d)$ can be empirically treated as a monotonically decreasing function of λ . This property is illustrated through numerical simulations, and Fig. 2 shows the typical monotonicity relationship.

3.2 The Contour-line Optimal Tuning Algorithm

The contour-line based tuning algorithm is proposed based on the monotonicity and unimodality properties. Here we assume the amount of total variation allowed (or equivalently, the overshoot specification according to Proposition 4) is relative small such that the constraint in problem



Fig. 2. Empirical monotonicity properties of the worst-case total variation with respect to λ and λ_d .

(13) is active for the optimal solution; this is intuitive in that for small total variation specifications, the optimal λ -parameters will behave aggressively to speed up the response, to achieve a small settling time. This assumption is mainly made to reduce the computational burden of solving the tuning problem, as the line-search procedure to find the value of λ that corresponds to the smallest T_s given a fixed value of λ_d and $\text{TV}(\lambda, \lambda_d) \leq \text{TV}^*$ can be avoided. Based on the above analysis, the optimization problem can be converted to

$$\min_{\lambda,\lambda_d} T_s(\lambda,\lambda_d)
s.t. TV(\lambda,\lambda_d) = TV^*.$$
(14)

In this way, it suffices to minimize the settling time $T_s(\lambda, \lambda_d)$ along the contour line $TV(\lambda, \lambda_d) = TV^*$. To solve this problem, two algorithms are proposed: the first algorithm (Algorithm 1) finds the corresponding λ on the contour for a fixed value of λ_d (denoted as λ^c), while the second algorithm (Algorithm 2) is utilized to find the optimal λ_d that leads to a smallest value of settling time, based on Algorithm 1. The algorithms are further interpreted in details in the following.

Algorithm 1 Find $\lambda^c(\mathrm{TV}^*, \lambda_d)$ and $T_s^c(\mathrm{TV}^*, \lambda_d)$.

1:	Input TV [*] , λ_d and the uncertainty intervals $[g, \overline{g}]$,
	$[\underline{T}_p, \overline{T}_p]$ and $[\underline{t}_d, \overline{t}_d];$
2:	Input ϵ ; $\triangleright \epsilon = 0.1$ by default
3:	$\underline{\lambda} \leftarrow 0.02; \ \overline{\lambda} \leftarrow 100;$
4:	while $\overline{\lambda} - \underline{\lambda} > \epsilon \mathbf{do}$
5:	$\lambda \leftarrow (\bar{\lambda} + \underline{\lambda}) \times 0.5;$
6:	Evaluate the total variation $TV(\lambda, \lambda_d)$ via (10);
7:	if $TV(\lambda, \lambda_d) - TV^* > 0$ then
8:	$\underline{\lambda} \leftarrow \lambda;$
9:	else
10:	$\lambda \leftarrow \lambda;$
11:	end if
12:	end while
13:	$\lambda^{c}(\mathrm{TV}^{*},\lambda_{d}) \leftarrow (\lambda+\underline{\lambda})/2, T_{s}^{c}(\mathrm{TV}^{*},\lambda_{d}) \leftarrow T_{s}(\lambda^{c},\lambda_{d});$
14:	end

Interpretations for Algorithm 1: Since $\text{TV}(\lambda, \lambda_d)$ is a monotonic function of λ for a fixed λ_d , the search of λ^c can be achieved by a bisection search. Here $T_s^c(\text{TV}^*, \lambda_d)$ represents the dependence of the worst-case settling time on TV^{*} and λ_d .

Copyright © 2015 IFAC

Algorithm 2 Tuning of λ and λ_d

- 1: Input the uncertainty intervals $[g, \overline{g}], [\underline{T}_p, \overline{T}_p]$ and $[\underline{t}_d, t_d];$
- 2: Input the overshoot specification OS*;
- 3: Calculate the total variation specification according to $TV^* = 1 + 2OS^*;$ $\triangleright \epsilon = 0.1$ by default
- 4: Input ϵ ;
- 5: $\underline{\lambda}_d \leftarrow \underline{\lambda}_d^*; \ \overline{\lambda}_d \leftarrow 100;$
- 6: while $\bar{\lambda}_d \underline{\lambda}_d > \epsilon$ do
- 7:
- 8:
- $\begin{array}{l} \lambda_{d1} \leftarrow \underline{\lambda}_{d} + (\bar{\lambda}_{d} \underline{\lambda}_{d}) \times 0.382; \\ \lambda_{d2} \leftarrow \underline{\lambda}_{d} + (\bar{\lambda}_{d} \underline{\lambda}_{d}) \times 0.618; \\ \text{Numerically evaluate the} \end{array}$ the settling 9: times $T_s^c(\mathrm{TV}^*, \lambda_{d1}) \text{ and } T_s^c(\mathrm{TV}^*, \lambda_{d2}) \text{ based on Algorithm 1;}$ if $T_s^c(\mathrm{TV}^*, \lambda_{d1}) > T_s^c(\mathrm{TV}^*, \lambda_{d2})$ then $\lambda_{dd} \leftarrow \lambda_{d1};$
- 10:
- 11:
- else12:
- $\bar{\lambda}_d \leftarrow \lambda_{d2};$ end if 13:
- 14:

15: end while 16: $\lambda_d \leftarrow (\bar{\lambda}_d + \underline{\lambda}_d)/2, \ \lambda \leftarrow \lambda^c(\mathrm{TV}^*, \lambda_d);$

17: **end**

Interpretations for Algorithm 2: The assumption utilized here is that $T_s^c(\mathrm{TV}^*, \lambda_d)$ is a unimodal function of λ_d ; the underlying cause is that λ_d controls the stability of the system. In this way, the algorithm uses golden search to find the optimal λ_d that achieves the smallest worst-case settling time.

4. DETERMINE TV* FROM THE DECAY RATIO

In Section 3, an efficient auto-tuning approach is designed to determine TV^{*} with OS^{*}. It gives a fast and simple tuning procedure while OS \leq OS^{*} is guaranteed; but it may introduce conservativeness in the tuned OS due to overlooking oscillation in the output response. In this section, we aim at providing an alternative heuristic userfriendly approach for specifying TV* through decay ratios to reduce the conservativeness in the tuning results.

4.1 Determine TV^{*} Using Decay Ratios

The decay ratio (denoted as DR) is defined as the ratio between two consecutive maxima of the step output, which is often used to measure the output oscillation for secondorder linear systems. From the engineering perspective, the response of the system in Fig. 1 can be approximated by that of a second-order system; and therefore it is reasonable to set TV^* equal to the total variation of a second-order linear system with a maximum allowed decay ratio DR^{*}, which is specified by users with process knowledge (e.g., 1/4).

In this approach, we assume that the DR^* is chosen by users. When users have limited knowledge of the process or have no specific requirement on the output oscillation, DR^* will be set to 1/4 by default, as "one quarter decay" ratio" is normally used as the design criterion for controller tuning (Levine, 2010). By empirically assuming that each lower peak amplitude is a half of the previous upper peak amplitude in the output response, which is based on "one quarter decay ratio" criterion, we arrive at the following formula to approximate TV^* from DR^* :

$$TV^* = 1 + \frac{3OS^*}{1 - DR^*},$$
 (15)

based on which Algorithm 3 is designed as below.

The above formula (15) can be obtained based on the definitions of the worst-case total variation and decay ratio with DR=DR^{*}; the key factor utilized here is that the amplitude of the first upper peak of the step output of a system is also equal to the overshoot.

Algorithm 3 Find λ^c and T_s^c with TV^{*} specified by (15)

1: Input TV^{*}, λ_d and $[\underline{g}, \overline{g}], [\underline{T}_p, \overline{T}_p]$ and $[\underline{t}_d, \overline{t}_d];$ $\triangleright \epsilon = 0.1$ by default 2: Input ϵ ; 3: $\lambda \leftarrow 0.02; \bar{\lambda} \leftarrow 100;$ 4: whichInUse ='TV': 5: while $\bar{\lambda} - \lambda > \epsilon$ do $\lambda \leftarrow (\bar{\lambda} + \lambda) \times 0.5;$ 6: Numerically evaluate $TV(\lambda, \lambda_d)$ and $OS(\lambda, \lambda_d)$; 7: 8: switch whichInUse case 'TV' 9: if $TV(\lambda, \lambda_d) - TV^* > 0$ then 10: 11: $\underline{\lambda} \leftarrow \lambda;$ else 12:if $\mathrm{OS} < \mathrm{OS}^*$ then 13: $\bar{\lambda} \leftarrow \lambda;$ 14: else 15:whichInUse ='OS';16:17: $\underline{\lambda} \leftarrow \lambda;$ end if 18:19: end if 20: case 'OS'21: if $OS(\lambda, \lambda_d) - OS^* > 0$ then 22: $\underline{\lambda} \leftarrow \lambda;$ 23:else $\lambda \leftarrow \lambda;$ 24:end if 25:26:end switch 27: end while 28: $\lambda^{c}(\mathrm{TV}^{*},\lambda_{d}) \leftarrow (\bar{\lambda}+\underline{\lambda})/2, T^{c}_{s}(\mathrm{TV}^{*},\lambda_{d}) \leftarrow T_{s}(\lambda^{c},\lambda_{d});$ 29: **end**

The above approach of specifying TV^* reduces the conservativeness in controlling output oscillation; but it also leads the possibility that the tuned OS is greater than OS^{*}. To avoid this issue, we introduce a switching tuning mechanism in the tuning of λ^c , which guarantees OS \leq OS^{*}. The detailed tuning procedure is given in the Algorithm 3.

4.2 Comparison of Algorithms 2 and 3

To solve the tuning problem in (11), Algorithms 2 and 3 are proposed in this work. Algorithm 2 determines the specification on the worst-case total variation automatically by exploiting the relationship between the overshoot and total variation of a step response. For the same specification on worst-case overshoots, Algorithm 3 potentially leads to a smaller settling time, as the corresponding specification on the worst-case total variation made according to (15) is greater than that of Algorithm 2; however, the sacrifice is that end users are required to specify the preference on decay ratios and the relationship between the overshoot specification and the total variation specification is heuristic rather than rigorously proven. For the same specification on the worst-case total variation, the tuning results of Algorithm 2 inherit a larger chance of optimality as the worst-case settling time is directly minimized, while the switching mechanism in Algorithm 3 has a potential effect of deviating the tuning results from the optimal values.

Table 1. Comparison of different tuning algorithms ($OS^* = 20\%$)

			(==,,,,		
	TV^*	TV	OS	TS	λ	λ_d
A2	1.4	1.39	0.1%	2760s	10.1277	3.9868
A3	1.8	1.7920	5.4%	2385s	8.1844	3.4943
OSAl		2.52	16.4%	2055s	6.7059	2.8709

5. INDUSTRIAL EXAMPLES

In this section, we apply the proposed results to an example extracted from real applications of machine directional paper machine control to illustrate the efficiency of the tuning algorithms. The following nominal system is considered:

$$G_0(s) = \frac{0.0135}{60s+1} e^{-90s}.$$
 (16)

This models a papermaking process from stock to conditioned weight. This model was identified using an advanced industrial control software package and used by an MPC controller for a real paper machine. The prediction and control horizons are set to $H_p = 42$ and $H_u = 20$, respectively; Q_1 , Q_2 and Q_3 are the same as shown in Section 2.3. The uncertainty level is defined as $[-\underline{r}\%, \overline{r}\%]$, which means that the real model parameters are within the following ranges:

$$T_{p} \in [(1 - \underline{r}\%)T_{p0}, (1 + \bar{r}\%)T_{p0}], t_{d} \in [(1 - \underline{r}\%)t_{d0}, (1 + \bar{r}\%)t_{d0}], g \in [(1 - \underline{r}\%)g_{0}, (1 + \bar{r}\%)g_{0}].$$
(17)

A large uncertainty level [-50%, 90%] is used for this model. First, we apply the proposed tuning algorithms, i.e., Algorithms 2 and 3, on this model, and then compare the obtained tuning results with the result obtained by the algorithm proposed in Shi et al. (2014), which we call "OS Algorithm" hereafter $^{1}.$ The specification of the worst-case overshoot is set to be $OS^* = 20\%$, and the tuning results for all algorithms are shown in Fig. 3 and Table 1, where we use OSAl, A2, A3 to represent the OS Algorithm, Algorithm 2, and Algorithm 3, respectively. From these results, the effect of considering total variation in the MPC tuning is apparent: 1) the obtained envelope responses by Algorithms 2 and 3 are much smoother than that of the OS Algorithm, although the responses obtained by Algorithm 3 is faster than that of Algorithm 2; 2) the resultant settling times of responses via Algorithms 2 and 3 are increased compared with that by the OS Algorithm, which is a natural tradeoff that has to be paid for smoother responses.

Now, we apply the tuning results to the Honeywell real time MPC + Simulator environment. To account for the model mismatch, the real time process is taken as

$$G_0(s) = \frac{0.0246}{109.2s+1} e^{-163.8s},$$
(18)

which lies within the uncertainty level [-50%, 90%] of the nominal process in (16). The initial operating conditions, y(0) = 432, u(0) = 3790, are obtained from the actual operating conditions. The optimization parameters of the MPC (H_p , H_u , Q_1 , Q_2 and Q_3) are chosen to be the same as in the above tuning procedure, and the constraints on the control signals are designed as follows:

3411 gpm
$$\leq U \leq$$
 4169 gpm,
-379 gpm $\leq \Delta U \leq$ 379 gpm. (19)

To consider possible changes of the operating conditions, a set-point change of 2 lbs/1000 ft² is made at t = 300s;

 $^{^1\,}$ Note that the OS Algorithm finds the MPC tuning parameters that minimize the worst-case settling time while considering the upper bound OS* on the worst-case overshoot.



Fig. 3. Step response envelopes obtained by the OS Algorithm, Algorithm 2 and Algorithm 3



Fig. 4. Real time MPC + simulator results

an output disturbance of 2 lbs/1000 ft² is introduced at t = 2250s; an input disturbance of 80 gpm is in effect at t = 4500s; the measurement noise is zero-mean Gaussian with standard deviation 0.1 lbs/1000 ft². The tuning results in Table 1 are applied to the process simulator and Honeywell real-time MPC for performance comparison. The tuning results are shown in Fig. 4. Despite the huge model-plant mismatch and the measurement noise, the responses of the system corresponding to all the three pairs of λ -values always robustly track the set-point for all the changes of the operating conditions, which indicates the effectiveness of the proposed algorithms. In addition, Fig. 4 also shows that the proposed algorithms lead to much smoother responses compared with that obtained without taking total variation into account, namely, the OS Algorithm, which further illustrates the effectiveness of considering total variation in design.

6. CONCLUSION

In this work, the proposed technique provides a solution to the challenge of finding MD-MPC tuning parameters to meet intuitive, time-domain, robust performance specifications for a process that can be described by a first-order plus deadtime model with easy to understand parametric uncertainties. The overall concept behind this approach may also be relevant to other applications where robust tuning is desired, but practitioner capabilities dictate that performance and uncertainty specifications should take a simple and intuitive form.

Copyright © 2015 IFAC

REFERENCES

- Al-Ghazzawi, A., Ali, E., Nouh, A., and Zafiriou, E. (2001). On-line tuning strategy for model predictive controllers. *Journal of Process Control*, 11(3), 265–284.
- Bagheri, P. and Sedigh, A.K. (2013). Analytical approach to tuning of model predictive control for first-order plus dead time models. *IET Control Theory and Applications*, 7(14), 1806–1817.
- Chu, D., Forbes, M., Backström, J., Gheorghe, C., and Chu, S. (2011). Model predictive control and optimization for papermaking processes. In T. Zheng (ed.), Advanced Model Predictive Control, 309–341. InTech. Chu, D., Forbes, M., and Backström, J. (2013). Technique
- Chu, D., Forbes, M., and Backström, J. (2013). Technique for converting a model predictive control (MPC) system into an explicit two degrees of freedom (2DOF) control system. U.S. Patent, Application No. 13/907495.
- Di Cairano, S. and Bemporad, A. (2010). Model predictive control tuning by controller matching. *IEEE Transac*tions on Automatic Control, 55(1), 185–190.
- Garriga, J.L. and Soroush, M. (2010). Model predictive control tuning methods: A review. *Industrial & Engineering Chemistry Research*, 49(8), 3505–3515.
 Garriga, J. and Soroush, M. (2008). Model predictive
- Garriga, J. and Soroush, M. (2008). Model predictive controller tuning via eigenvalue placement. In American Control Conference, 429–434.
- Han, K., Zhao, J., and Qian, J. (2006). A novel robust tuning strategy for model predictive control. In 6th World Congress on Intelligent Control and Automation (WCICA), 6406–6410. Dalian, P.R. China.
- (WCICA), 6406–6410. Dalian, P.R. China. Júnior, G.A.N., Martins, M.A.F., and Kalid, R. (2014). A PSO-based optimal tuning strategy for constrained multivariable predictive controllers with model uncertainty. ISA Transactions, 53(2), 560–567.
- Levine, W. (2010). The Control Handbook. Electrical Engineering Handbook. Taylor & Francis. Qin, S.J. and Badgwell, T.A. (2003). Reduced order
- Qin, S.J. and Badgwell, T.A. (2003). Reduced order process modelling in self-tuning control. Control Engineering Practice, 11(7), 733 – 764.
- Shah, G. and Engell, S. (2010). Tuning MPC for desired closed-loop performance for SISO systems. In 18th Mediterranean Conference on Control Automation (MED), 628–633. Marrakech, Morocco.
- Shi, D., Wang, J., Forbes, M., Backström, J., and Chen, T. (2014). Robust tuning of machine directional predictive control of paper machines. *Industrial & Engineering Chemistry Research*, under review.
- Shi, D., Wang, J., and Ma, L. (2011). Design of balanced proportional-integral-derivative controllers based on bilevel optimisation. *IET Control Theory & Applications*, 5(1), 84–92.
- Skogestad, S. and Postlethwaite, I. (1996). Multivariable Feedback Control: Analysis and Design. John Wiley & Sons, New York.
- Tran, Q.N., Özkan, L., and Backx, A.C.P.M. (2012). MPC tuning based on impact of modelling uncertainty on closed-loop performance. In AIChE Annual Meeting, 186e. Pittsburgh, PA.
- Wojsznis, W., Gudaz, J., Blevins, T., and Mehta, A. (2003). Practical approach to tuning MPC. ISA Transactions, 42(1), 149–162.