Tuning Guidelines for Model-Predictive Control

Mohammed Alhajeri and Masoud Soroush*
Department of Chemical and Biological Engineering
Drexel University
Philadelphia, PA 19104

October 24, 2021

FOURTH REVISED VERSION

Submitted for Publication in I&EC Research

Keywords: Model-Predictive Control, Tuning Guidelines, Dynamic Matrix Control, Generalized Predictive Control, Kalman Filter

*Corresponding Author; (215) 895-1710 (voice); (215) 895-5837 (fax); ms1@drexel.edu

Abstract

This paper reviews available tuning guidelines for model predictive control (MPC), from theoretical and practical perspectives. Its primary focus is on the guidelines introduced since the publication of our previous review of MPC tuning guidelines in this same journal in 2010. Since then, new guidelines based on approaches such as pole placement and multi-objective optimization have been proposed, and more auto-tuning methods have been introduced. This review covers different implementations of MPC such as dynamic matrix control, generalized predictive control, and state-space-model predictive control that requires Kalman filter tuning. The closed-loop performances of a distillation column and the Shell fractionator under model-predictive controllers tuned using four different tuning guidelines are compared through numerical simulations.

1. Introduction

Model predictive control (MPC) is now a mature control technology [1]. Since the pioneering work of Zadeh *et al.* [2], Propoi [3], and Rafal and Stevens [4], it has evolved significantly and has been implemented in many industries [5-12]. MPC is particularly suitable for complex multivariable systems with constraints [1]. Currently-active research areas in MPC include economic MPC [13-15] and stochastic MPC [16-18].

The formulation and implementation of MPC have evolved considerably since its introduction. They include linear quadratic Gaussian (LQG) [19], dynamic matrix control (DMC) [20], model algorithmic control (MAC) [21], state-space MPC, and nonlinear MPC. As MPC evolved, more types of constraints were added to the formulation, more complex performance indices were considered, and set-point tracking was replaced with reference trajectory tracking. The latter has allowed for adjusting the speed and shape of the closed-loop output response. With the addition of these appealing features, MPC became more flexible, but its complexity and number of tunable parameters increased. The tunable parameters now include prediction horizons, control horizons, weights on the magnitudes of the controlled variables, weights on the rates of change of manipulated variables, weights on the magnitudes of manipulated variables, reference trajectory parameters, soft constraint weights, and the model horizon (when the controller is based on a finite impulse- or step-response model).

The high number of tunable parameters has motivated many studies on MPC tuning. Existing MPC tuning methods have different forms. Some are in the form of equations describing the tuning parameters in terms of process dynamics parameters such as dead times, time constants, and the sampling period. For example, Shridhar and Cooper's guideline [22] is based on the assumption that process dynamics can be represented by a first-order plus dead time model. There are some in the form of optimization problems whose solutions are MPC tuning parameter values. There are also heuristic tuning methods. An MPC implementation first requires the development or identification of a suitable and accurate process model. The MPC formulation is intuitive and easy to understand. Some control practitioners [23] believe that MPC based on an accurate process model is easy to tune; an accurate model simplifies the task of finding a balanced trade-off between controller robustness and performance.

There have been several review papers on MPC tuning. Yamuna Rani and Unbehauen [24] surveyed tuning methods that had been proposed for DMC and generalized predictive controller (GPC) during 1984–1995. Garriga and Soroush [19] provided a comprehensive review of theoretical and heuristic tuning strategies for different MPC implementations such as DMC, GPC, state space, and max-plus-linear. Their review covered publications until 2009. This paper reviews MPC tuning guidelines that have been introduced since then. These tuning guidelines are for MPC implementations such as DMC, GPC, and state space. Two case studies, one on a distillation column and the other on the Shell fractionator, are presented to compare the closed-loop performances obtained by using four of the tuning guidelines reviewed in this article.

2. Mathematical Preliminaries

To define the tunable parameters precisely and for completeness, a mathematical formulation of MPC is first needed. Here, for the ease of instruction, we consider a linear, discrete-time, time-invariant state-space process model in the form:

$$\left.\begin{aligned} \boldsymbol{x}(k+1) &= \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\boldsymbol{u}(k) + \boldsymbol{G}\boldsymbol{w}(k) \\ \boldsymbol{y}(k) &= \boldsymbol{C}x(k) + \boldsymbol{v}(k) \end{aligned}\right\} \tag{1}$$

where $\boldsymbol{x} = \left[x_1, \cdots, x_{n_x}\right]^T$ is the vector of state variables, $\boldsymbol{y} = \left[y_1, \cdots, y_{n_y}\right]^T$ is the vector of controlled variables, and $\boldsymbol{u} = \left[u_1, \cdots, u_{n_u}\right]^T$ is the vector of manipulated variables. Here, k represents the sampling time instant. $\boldsymbol{A} \in R^{n_x \times n_x}$, $\boldsymbol{B} \in R^{n_x \times n_u}$, $\boldsymbol{C} \in R^{n_y \times n_x}$, and $\boldsymbol{G} \in R^{n_x \times n_w}$ are all constant matrices. $\boldsymbol{w} \in R^{n_w \times 1}$ is the vector of state disturbance variables, and $\boldsymbol{v} \in R^{n_y \times 1}$ is the vector of output disturbance variables. Each component of $\boldsymbol{w}$ and $\boldsymbol{v}$ is assumed to be a white noise sequence (Gaussian sequence with zero mean). The covariance matrices of $\boldsymbol{w}$ and $\boldsymbol{v}$ are represented by $\boldsymbol{Q_w}$ and $\boldsymbol{Q_v}$, respectively. Estimates of the state variables are obtained using a Kalman filter:

$$\hat{\boldsymbol{x}}(k+1|k) = \boldsymbol{A}\hat{\boldsymbol{x}}(k|k-1) + \boldsymbol{B}\boldsymbol{u}(k) + \boldsymbol{A}\boldsymbol{L}(k)\left(\boldsymbol{y}(k) - \boldsymbol{C}\hat{\boldsymbol{x}}(k|k-1)\right) \quad (2)$$

where the Kalman gain is given by:

$$\boldsymbol{L}(k) = \boldsymbol{F}(k|k-1)\boldsymbol{C^T}[\boldsymbol{C}\boldsymbol{F}(k|k-1)\boldsymbol{C^T} + \boldsymbol{Q_v}]^{-1} \quad (3)$$

Here, $\boldsymbol{F}(k|k-1)$ denotes the covariance matrix of the estimation error, $\boldsymbol{e}(k) = \boldsymbol{x}(k) - \hat{\boldsymbol{x}}(k|k-1)$. $\boldsymbol{F}(k|k-1)$ is obtained by solving the Ricatti equation:

$$\boldsymbol{F}(k+1|k) = \boldsymbol{A}\boldsymbol{F}(k|k-1)\boldsymbol{A^T} + \boldsymbol{G}\boldsymbol{Q_w}\boldsymbol{G^T} - \boldsymbol{A}\boldsymbol{L}(\boldsymbol{k})\boldsymbol{C}\boldsymbol{F}(k|k-1)\boldsymbol{A^T} \quad (4)$$

Consider a general moving-horizon optimization problem that minimizes the following cost function:

$$\min_{u_1(k),\cdots,u_1(k+M_1-1),\cdots,u_{n_u}(k),\cdots,u_{n_u}(k+M_{n_u}-1)} \left\{\sum_{j=1}^{n_y} \sum_{l=1}^{P_j-P_{o_j}+1} q_{jl} \left[y_{r_j}\left(k+l+P_{o_j}-1\right) - \hat{y}_j\left(k+l+P_{o_j}-1\right)\right]^2 + \sum_{i=1}^{n_u} \sum_{l=1}^{M_i} \delta_{jl}[u_i(k+l-1)]^2 + \sum_{i=1}^{n_u} \sum_{l=1}^{M_i} r_{il}[u_i(k+l) - u_i(k+l-1)]^2 \right\} \quad (5)$$

subject to:

$$\boldsymbol{x}(k+1) = \boldsymbol{Ax}(k) + \boldsymbol{Bu}(k) + \boldsymbol{Gw}(k) \tag{6}$$

$$\boldsymbol{y}(k) = \boldsymbol{C}x(k) + \boldsymbol{v}(k) \tag{7}$$

$$u_i(k+l) = u_i(k+l-1), \quad i = 1, \dots, n_u, \; l = M_i, \cdots \tag{8}$$

$$\boldsymbol{Du}(k) \leq \boldsymbol{d}, \quad k = 0,1, \cdots \tag{9}$$

$$\boldsymbol{Hx}(k) \leq \boldsymbol{h}, \quad k = 0,1, \cdots \tag{10}$$

y_{r_j} represents the reference trajectory of the controlled variable y_j, defined by:

$$y_{r_j}(k) = y_j(k) \tag{11}$$

$$y_{r_j}(k+1) = \hat{y}_j(k+1) \tag{12}$$

$$\vdots$$

$$y_{r_j}(k+\gamma_j-1) = \hat{y}_j(k+\gamma_j-1) \tag{13}$$

$$y_{r_j}(k+\gamma_j) = (1-\beta_j)y_{sp_j}(k) + \beta_j\hat{y}_j(k+\gamma_j-1) \tag{14}$$

$$y_{r_j}(k+\gamma_j+1) = (1-\beta_j)y_{sp_j}(k) + \beta_j\hat{y}_j(k+\gamma_j) \tag{15}$$

$$\vdots$$

$$y_{r_j}(k+P_j) = (1-\beta_j)y_{sp_j}(k) + \beta_j\hat{y}_j(k+P_j-1) \tag{16}$$

where β_j is the tuning parameter of the reference trajectory y_{r_j} $(0 \leq \beta_j < 1)$ and sets the speed of the y_j response, and γ_j is the relative degree (order) of the controlled variable y_j with respect to **u**.

The performance index of (5) in terms of:

$$Y(k) = \begin{bmatrix} Y_1(k) \\ \vdots \\ Y_{n_y}(k) \end{bmatrix}, \quad Y_r(k) = \begin{bmatrix} Y_{r_1}(k) \\ \vdots \\ Y_{r_{n_y}}(k) \end{bmatrix}, \quad U(k) = \begin{bmatrix} U_1(k) \\ \vdots \\ U_{n_u}(k) \end{bmatrix}$$

where

$$Y_j(k) = \begin{bmatrix} y_{r_j}\left(k + P_{o_j}\right) \\ \vdots \\ y_{r_j}\left(k + P_j\right) \end{bmatrix}, \ j = 1, \cdots, n_y; \qquad Y_{r_j}(k) = \begin{bmatrix} y_{r_j}\left(k + P_{o_j}\right) \\ \vdots \\ y_{r_j}\left(k + P_j\right) \end{bmatrix}, \ \ j = 1, \cdots, n_y;$$

$$U_j(k) = \begin{bmatrix} u_j(k) \\ \vdots \\ u_j\left(k + M_j - 1\right) \end{bmatrix}, \ j = 1, \cdots, n_u$$

takes the form:

$$\min_{U(k)} \{[Y_r(k) - Y(k)]^T \mathbf{Q} [Y_r(k) - Y(k)] + U(k)^T \Delta\, U(k)$$
$$+ [U(k) - U(k-1)]^T \boldsymbol{R} [U(k) - U(k-1)]\}$$

where

$$\boldsymbol{Q} = \text{diag}\left\{q_{11}, \cdots, q_{1(P_1 - P_{o_1} + 1)}, \ \cdots \ , q_{n_y 1}, \cdots, q_{n_y\left(P_{n_y} - P_{o_{n_y}} + 1\right)}\right\},$$

$$\boldsymbol{R} = \text{diag}\left\{r_{11}, \cdots, r_{1(M_1 - 1)}, \ \cdots \ , r_{n_u 1}, \cdots, r_{n_u(M_{n_u} - 1)}\right\},$$

$$\Delta = \text{diag}\left\{\delta_{11}, \cdots, \delta_{1(M_1 - 1)}, \ \cdots \ , \delta_{n_u 1}, \cdots, \delta_{n_u(M_{n_u} - 1)}\right\}.$$

The following notations will be used in the next sections: $\boldsymbol{P} = [P_1 \ \cdots \ P_{n_y}]^T$, $\boldsymbol{P}_o = [P_{o_1} \cdots P_{o_{n_y}}]^T$, and $\boldsymbol{M} = [M_1 \cdots M_{n_u}]^T$.

3. Tuning Guidelines

3.1. Horizons

The prediction horizons, control horizons, and model horizon (in the case that the model is a finite step- or impulse-response model) are important tuning parameters in MPC. This section reviews tuning guidelines for these horizons.

3.1.1. Prediction Horizons

A pair of lower and upper prediction-horizon limits, P_{o_i} and P_i, specify the range of the time window into the future (in terms of number of sampling periods) over which the response of a controlled variable y_i is predicted by a plant model and is optimized. $[k + P_{o_i}, k + P_i]$ is the time window, where k represents the present time instant. In general, the prediction horizons should be adequately large so that controlled output predictions can represent a significant portion of the dynamics of the process under consideration; adequately large values should be selected for P_1, ⋯, P_{n_y} to ensure closed-loop stability and robustness. However, as P_1, ⋯, P_{n_y} increase, the computational cost of solving the MPC optimization problem increases.

If the plant under consideration is delay-free, typically $P_{o_1} = 1, \cdots, P_{o_{n_y}} = 1$. For plants with time delays, the following guideline is proposed:

$$P_{o_1} = 1 + \frac{\min_j\left(\sum_{j=1}^{nu} \theta_{1j}\right)}{t_s}, \cdots, P_{o_{n_y}} = 1 + \frac{\min_j\left(\sum_{j=1}^{nu} \theta_{n_y j}\right)}{t_s}.$$

The upper limits of the prediction horizons, $P_1, \cdots, P_{n_y}$, should be large enough. In general, the robustness and stability of MPC increase with increased P_1, ⋯, P_{n_y}. Beyond certain values, further increases in P_1, ⋯, P_{n_y} do not alter the control performance significantly, although they intensify the computational burden of the controller and may deteriorate the robustness of the controller[25]. Many tuning guidelines have been proposed for setting the upper limits of the prediction horizons, P_1, ⋯, P_{n_y}. The paragraphs in the rest of this section review guidelines proposed for tuning P_1, ⋯, P_{n_y} in MPC implementations such as GPC, DMC, and the state-space. Bagheri *et al.* [26] assumed a first-order plus dead time (FOPDT) model for unconstrained single-input single-output (SISO) systems. Using a pole-placement approach, they proposed setting the prediction horizon upper limit (prediction horizon) according to $P_1 \geq \frac{\theta}{t_s} + 1$, where θ is the SISO plant dead time and t_s is the sampling period. In other words, the prediction horizon upper limit (prediction horizon) should be greater than or equal to the total time delay of the discrete-time process model, where the total time delay is the number of sampling periods of the process deadtime plus one. The one sampling period time delay is generated by the time discretization of a dynamical model. Note that when $P_1 < \frac{\theta}{t_s} + 1$, the present value of the manipulated variable does not affect the controlled variable model-predictions over the prediction window. The same authors [27] extended their previous study to unconstrained multi-input multi-output (MIMO) systems by using the same pole placement approach. They assumed that the system is square, and all elements of its transfer function matrix are FOPDT transfer functions. This implies setting:

$$P_1 \geq 1 + \frac{\max_j\left(\sum_{j=1}^{n_u} \theta_{1j}\right)}{t_s}, \cdots, P_{n_y} \geq 1 + \frac{\max_j\left(\sum_{j=1}^{n_u} \theta_{n_y j}\right)}{t_s}.$$

They set all control horizons to one and used the same SISO guideline for the prediction horizons to develop guidelines for tuning the matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$. Gholaminejad *et al.* [28] developed an approach for adaptive MPC tuning for time-variant plants. The approach involves online identification of a FOPDT model and then calculation of the prediction horizons using the identified model parameter values and an existing tuning guideline that requires the parameter values.

For example, the SISO guide of Maurath *et al.* [29] can be extended to MIMO plants; choose the prediction horizon of a controlled variable such that the horizon is greater than or equal to the rise time for 80 percent of the steady-state of the controlled variable and is less than or equal to the rise time for 90 percent of the steady-state of the controlled variable. Ebrahimi *et al.* [30]

recommended using a slightly larger lower limit for the prediction horizon in SISO plants; that is, $T_s \leq P_1$. For SISO plants, Neshasteriz *et al.* [31] suggested setting the prediction horizon, P_1, according to $\frac{\theta}{t_s} + 1 \leq P_1 \leq \frac{\theta}{t_s} + M_1$; the lower bound is exactly that of Bagheri *et al.* [26]. Their guideline states that the upper limit of the prediction window should be less than or equal to the number of sampling periods of the plant deadtime plus the control horizon. In this case, the number of sampling periods over which the plant output response in projected into the future is less than or equal to the control horizon. Here, this tuning guide like is extended to MIMO systems:

$$\frac{\max_j\left(\sum_{j=1}^{n_u} \theta_{1j}\right)}{t_s} + 1 \leq P_1 \leq \frac{\max_j\left(\sum_{j=1}^{n_u} \theta_{1j}\right)}{t_s} + \max_j\left(\sum_{j=1}^{n_u} \mathrm{M}_j\right),\ \cdots,$$

$$\frac{\max_j\left(\sum_{j=1}^{n_u} \theta_{n_y j}\right)}{t_s} + 1 \leq P_{n_y} \leq \frac{\max_j\left(\sum_{j=1}^{n_u} \theta_{n_y j}\right)}{t_s} + \max_j\left(\sum_{j=1}^{n_u} \mathrm{M}_j\right).$$

Tran *et al.* [32] studied GPC tuning for MIMO systems and suggested setting $P_j \geq 3(n+1)$, $j = 1, \cdots, n_y$, where n is the degree of the characteristic polynomial (CP) of the plant. These prediction horizons are sufficiently large to account for process dynamics. Yamashita *et al.* [33] set $P_1 = \cdots = P_{n_y} = 0.8T_s/t_s$, where T_s is the largest settling time of the controlled variables in open loop. Shehu *et al.* [34] and Shah *et al.* [35-36] used a different method and suggested setting the prediction horizon $P_1 = N_2 - N_1 + 1$, where $N_1 = \theta/t_s$, and N_2 is the upper prediction bound that a user sets according to the application. Sha'aban *et al.* [37] proposed setting the prediction horizon $P_1 = (\theta + 5\tau_c)/t_s$, where τ_c is the process time constant, while Neshasteriz *et al.* [38] suggested setting $P_1 = (\theta/t_s + M_1)$ and $P_{o1} = (\theta/t_s + 1)$. In the articles [39-42], the tuning guidelines proposed by Shridhar and Cooper [43] was used to set the prediction horizon. However, in Ref.[44] the tuning guideline proposed by Clarke and Mohtadi [45] was used.

Júnior *et al.* [46] and Taeib *et al.* [47] found optimal values of $P_1, \cdots, P_{n_y}$ and other tuning parameters by minimizing a time-weighted multi-objective performance index subject to the plant model with its worst-case parameter values. In this case, at each time instant, MPC should solve two optimization problems: an internal problem, the solution of which is MPC tunable parameter values; and an external problem, the solution of which is the current values of the manipulated variables. Turki *et al.* [48] also suggested an optimization approach for tuning predication horizons in MPC of linear controllable MIMO plants with active constrains. Their optimization problem has three performance indices (a stability degree index, an error index, and a rapidity index). The prediction-horizon tuning guidelines that were reviewed in this section are summarized in Tables 1 and 2, and those reviewed in Ref. [19] are presented in Tables S1 and S2 of Supplementary Information (SI).

Table 1. Tuning guidelines for the lower limit of the prediction horizon.

Ref.	P_{o_i}	Method
[31]	$\frac{\min_j \theta_{ij}}{t_S} + 1$	GPC
[38]	$\frac{\min_j \theta_{ij}}{t_S} + 1$	GPC
[46]	via optimization	State-space
[39-40,39,40]	[43]	DMC
[44]	Same as in Ref.[49-50]	GPC
[48, 51 52]	0	State-space
[41]	0	DMC

Table 2. Tuning guidelines for the upper prediction horizons.

Ref.	P_i	Method
[26-27, 53]	$\frac{\max_j \theta_{ij}}{t_S} + 1$	DMC [26], GPC [27], State-space [53]
[31]	$\frac{\max_j \theta_{ij}}{t_S} + M_i$	GPC

32	$3(n_a+1)$	GPC
33	$0.8T_s$	State-space
54	$0.8T_s \leq P_i \leq 0.9T_s$	State-space
30	$P_i \geq T_s$	GPC
35-36	$\Lambda_2 - \Lambda_1 + 1$	GPC
38	$\max_j \theta_{ij} + M_i$	GPC
46, 48	via optimization	State-space
39-42	$\max_j \left[\frac{\theta_{ij} + 5\tau_{c_{ij}}}{t_s} + 1\right]$	DMC
55-56	large value	GPC
44	$2n_a - 1 < P_i < \frac{t_{95}}{t_s}$	GPC
51-52, 57	Very large value	State-space
58	M_i	State-space
47	via optimization	
37	$\max_j \left[\frac{\theta_{ij} + 5\tau_{c_{ij}}}{t_s}\right]$	DMC
59	$\frac{\theta_{11} + \tau_{c_{11}}}{t_s}$	SISO DMC

3.1.2. Control Horizons

The control horizon M_{i} is the number of values of the manipulated variable u_{i} that a model predictive controller calculates at each time instant; at each time instant a model predictive controller calculates $u_{\mathrm{i}}(k), \cdots, u_{\mathrm{i}}(k+M_{\mathrm{i}}-1)$, as it sets $u_i(k+j) = u_i(k+j-1)$, $j = M_{\mathrm{i}}, M_{\mathrm{i}}+1, \cdots$. As the control horizons increase, the aggressiveness, the ability to stabilize unstable plants, and computational cost (dimension of the optimization problem that has to be solved each time instant) of the controller increase, but the robustness of the control system decrease [60]. This subsection reviews the tuning guidelines proposed for control horizons.

Bagheri *et al.* [26, 53] recommended setting $M_i \leq 2, i = 1, \cdots, n_u$, as they did not observe any appreciable improvement in closed-loop performance beyond these values, and small $M_1, \cdots, M_{n_u}$

values decreases the dimension of the optimization problem that MPC should solve each time instant. Using the same justifications, studies in Ref.[27, 34-36] suggested setting $M_i \leq 2,\ i = 1, \cdots,$ n_u, but those in Ref. [32] recommended setting $M_i < 3,\ i = 1, \cdots, n_u$. For a SISO plant, Sha'ban *et al.* [37] achieved satisfactory performance and robustness by setting $M = 3$. Salem *et al.* [61] set $M = 3$ and achieved a desired closed-loop output response (with a short T_s, a short rise time (T_r), a small tracking error, and a low overshoot). Yamashita *et al.* [33, 54] recommended $3 \leq M_i \leq 5$, , $i = 1, \cdots, n_u$, while for a SISO plant Ebrahimi *et al.* [30] suggested setting $M = P - d$, where d is the the dead time in a FOPDT process model of the plant. For SISO plants, Ref. [48, 51, 55, 62] suggested setting $M = P$, while Ref. [39-42] recommended using the guideline of Shridhar *et al.* [43], and Ref.[44] used the guideline of Clarke and Mohtadi [45]. The control-horizon tuning guidelines that were reviewed in this section are summarized in Table 3, and those reviewed in Ref. [19] are presented in Table S3 of the SI.

Table 3. Tuning guidelines for the control horizon.

Ref.	M_i	Method
26-27, 53	≤ 2	DMC [26], GPC [27], State-space [53]
32	< 3	GPC
33, 54	$3 \leq M_i \leq 7$	State-space
30	$P_i - \max_j d_{ij}$	GPC
35-36	1	GPC
46, 58	via optimization	State-space
37	3	DMC
39-42	Same as in Ref.[43]	DMC
55-56	P_i	GPC
44	49-50	GPC
48, 51-52	P_i	State-space
57	large value	State-space
28	2	State-space
47	via optimization	
59	2	SISO DMC

3.1.3. Model Horizon

A finite impulse- or step-response model is typically used in DMC. These models have a horizon, called the model horizon, that should be set. The model horizon affects the condition number of the matrix **A**; as the model horizon increases, the condition number of the matrix **A** increases [63].

According to Seborg *et al.* [60], the model horizon, N, should be typically in the range of $30 - 120$. A general rule is to set $N = T_s/t_s$, which ensures that the model can describe the plant output response entirely from the time a plant input change is made to the time when the output response reaches steady-state conditions[60]. Bagheri *et al.* [39] suggested setting prediction and control horizons according to the guidelines of Shridhar and Cooper [22], but proposed setting $N = 2\left(\frac{5\tau_c}{t_s}\right) + \frac{\theta}{t_s} + 1$, which yields a model horizon greater than their suggested prediction horizon. Klopot et al.[89] suggested setting $N = \frac{3\tau_c + \theta}{t_s}$. The model-horizon tuning guidelines reviewed in Ref. [19] are listed in Table S4 of the SI.

3.2. Weight Matrices (Q, R, and Δ)

The weight matrix $\boldsymbol{Q}$ includes penalties on the output errors, while $\boldsymbol{R}$ and Δ include penalties on the rates of input changes and the input magnitudes, respectively, as defined in Section 2.2. The elements of the weighting matrix $\boldsymbol{Q}$ penalize the deviation of the controlled outputs from their reference trajectories. Their relative values represent the relative importance of the controlled outputs. They are reflective of the relative costs of the controlled variables deviating from their reference trajectories and therefore their set points. The elements of $\boldsymbol{R}$ and Δ strongly affect the magnitudes and the rates of change of the controller outputs (manipulated variables). Their relative values of the elements of R are indicative of the relative costs of the manipulated inputs.

There are numerous different guidelines for setting the penalty matrices $\boldsymbol{R}$, $\boldsymbol{Q}$, and Δ. Here, analytical and optimization-based tuning guidelines for $\boldsymbol{R}$ and $\boldsymbol{Q}$ are discussed. It is assumed that $\Delta = 0$, unless a guideline is mentioned. Assuming FOPDT plant models and using the pole placement concept, Bagheri *et al.* [26] and Bagheri and Khaki-Sedigh[53] developed closed-form equations for calculating the weight matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ for SISO DMC. They tested their tuning guidelines on plants with FOPDT models and found that the guidelines provide fast tracking and good performance (low overshoot and short settling time). Bagheri and Khaki-Sedigh[53] tested their proposed tuning guidelines on an unstable FOPDT plant, whose output showed no overshoot and a short settling time, and on a pH neutralization process, whose output showed a small overshoot and a short settling time. However, the guidelines did not provide satisfactory closed-loop performance when used for time-variant plants and for plants with very poor models. Gholaminejad *et al.* [28] continued the work of Bagheri *et al.* [26] and developed new closed-form equations for setting $\boldsymbol{Q}$ and $\boldsymbol{R}$. When applied to the same pH neutralization process, the tuning guidelines proposed in Ref.[28] provided a better tracking performance than those presented in Ref.[26]. Bagheri *et al.* [27] extended the work presented in Ref.[26] to MIMO systems represented by state-space models, and applied the resulting guidelines to the same pH neutralization process and the Wood and Berry process [64]. Using a controller-matching approach for a GPC and a linear time-invariant controller, Tran *et al.* [32] developed closed-form equations for $\boldsymbol{R}$ and $\boldsymbol{Q}$. They showed the effectiveness of their tuning guidelines by applying the guidelines to a binary distillation column. Belda *et al.* [65] develop relationships between $\boldsymbol{Q}$ and its covariance and between $\boldsymbol{R}$ and its covariance. The application of the proposed approach to a multidimensional robotic system led to a satisfactory closed-loop response.

Many tuning guidelines suggest setting $\boldsymbol{Q} = I$ and $\boldsymbol{R} = \rho I$, where ρ is a scalar also known as a move suppression factor. While these settings simplify the task of MPC tuning, they take away some flexibility of MPC and limit the degree of control quality that MPC can provide. This approach, finding a suitable value of ρ instead of finding weight matrices, simplifies the tuning task. With these settings, Bagheri *et al.* [66] applied DMC to SISO systems with an FOPDT model and analytically found a range for ρ that provides robustness to model uncertainties. They applied their tuning guidelines to FOPDT-MPC of a laboratory-scale level process and to a pressure control system, and observed desired tracking performances in both cases. Performing sensitivity analyses, Bagheri *et al.* [39] developed the following closed-form equation for ρ in terms of the process dynamic parameters (i.e., time constant and dead time) for SISO DMC based on a FOPDT model:

$$\rho = a\left(\frac{\theta}{\tau} + 0.94\right)^{0.15} \sigma^{0.9} K^2 \tag{17}$$

where K is the steady state gain, and a and σ are set according to: (i) (a, σ) = (0.11, 0.1) when the output error is more important; (ii) (a, σ) = (6.67,10) when the control effort is more important; and (iii) (a, σ) = (0.84, 1) when both control effort and output error are important. Bagheri and Khaki-Sedigh[40] recommended setting (a, σ) = (0.84,1), which provided a smoother response and better setpoint tracking than the guidelines of Shridhar and Cooper [43] when applied to the pH neutralization process. Neshasteriz *et al.* [38] applied the same concept to GPC MPC based on a SOPDT model. They conducted optimization and simulations, leading to the following equations for setting ρ:

$$\rho = \frac{(p_1\, p_2)^{b_1}}{(p_1 + p_2)^{b_2}} + b_3 (p_1 p_2)^{b_4} \sigma - \sigma^{b_5}$$

$b_1 = -0.064, \quad b_2 = 0.8, \qquad b_3 = 2.0374, \qquad b_4 = 2.0510, \qquad b_5 = 0.0746$

$$\rho = (\omega)^{b_1}(\omega + \varepsilon)^{b_2} + b_3(\omega)^{b_4}\sigma - \sigma^{b_5} \tag{18}$$

$b_1 = -0.0342,\ b_2 = -0.0460,\ b_3 = 1.1365,\ b_4 = 0.0242, \qquad b_5 = 0.0021$

where $\sigma = 1$ in this study, and the parameters p_1, p_2, ω, and ε represent the two poles, natural frequency, and damping ratio of a second-order-plus-deadtime (SOPDT) plant. This tuning guideline outperformed the guidelines of Clark and Mohtadi [49] in terms of the integral of squared error (ISE) and overshoot. This approach can be used in online tuning. Using the same approach, Ebrahimi *et al.* [30] developed the following closed form equation for the scalar ρ:

$$\rho = 0.317\,\Gamma^{1.26}\beta^{0.422}\exp\left(-0.113\frac{d}{\tau}\right) + 0.852\,\Gamma^{0.92}\exp(-208.064\beta^{5.071}) + 0.029\,\Gamma^{1.551}\left(\frac{d}{\tau}\right)^{0.202} \tag{19}$$

where Γ is a user-set parameter that reflects the importance of the control effort, d is the plant dead time in terms of number of sampling periods, and β $(0 \leq \beta \leq 1)$ is the reference trajectory parameter. The effectiveness of this approach was examined for both SISO and MIMO systems with FOPDT models; it yielded a better performance (with less output ISE, no overshoot, and faster response) compared to the guidelines of Sridhar and Cooper [22].

Huusom *et al.* [55] proposed an offset-free state-space SISO MPC based on a autoregressive with exogenous terms (ARX) model. They suggested setting weight matrices $\boldsymbol{Q} = 0.5\boldsymbol{I}$, $\boldsymbol{R} = \rho\boldsymbol{I}$, and $\Delta = 0$, where ρ was obtained based on input and output variances of the closed-loop system. Their approach involves calculating the variances at different values of ρ, from minimum variance control to no control, and choosing the value of ρ corresponding to the inflection point of a log-

log plot of the variances. This ρ value was found to give a good balance between input and output variance.

For state-space MPC, Yamashita *et al.* [33] recommended setting $\boldsymbol{Q} = \Delta = I$ and $\boldsymbol{R} = \rho I$, and using a very small value for ρ. They set $\rho = 10^{-2}$ in three different examples (a SISO system with SOPDT model, where the tuning approach resulted in a shorter settling time; a Shell heavy oil fractionator, where the controller achieved the desired setpoint tracking; and a crude distillation unit, in which the tuning approach yielded a minimum control effort). Kokate *et al.* [41] suggested setting ρ to a small value such as 0.1 in DMC, which outperformed a well-tuned proportional–integral–derivative (PID) controller in terms of overshoot, response smoothness, and robustness. For a state-space implementation of MPC, Burgos *et al.* [67] suggested setting $Q = C^T C$ and $R = \rho I$. Since their feedback loop was disconnected with $\rho = 0$, they suggested using a small value for ρ. This approach was applied to a quadrotor reference tracking problem and yielded satisfactory setpoint tracking. Conversely, Tsoeu *et al.* [57] recommended setting ρ to a value close to 1.

Optimization-based tuning methods have been developed for different control methods including MPC. Based on optimization, Turki *et al.* [48] recommended setting $\boldsymbol{Q} = C^T C$ and $\boldsymbol{R} = \rho I$. When applied to SISO and MIMO systems with FOPDT models, this tuning strategy was found to yield better performance in terms of response smoothness and speed, compared to those of Ebrahimi and Bagheri [66] and Iglesias *et al.*[68] Vallerio *et al.* [69] employed a multi-objective optimization approach to tune $\boldsymbol{Q}$ and $\boldsymbol{R}$ by minimizing the deviation of the measured input and output from their references. This approach can also be used to tune β and Δ. Yamashita *et al.* [70] also used a multi-objective optimization approach to obtain the $\boldsymbol{Q}$, $\boldsymbol{R}$, and Δ values that minimize the L_2-norm of the errors (i.e., the setpoint error and the reference error for input) subject to the upper and lower bounds on each tuning parameter (q_{ij}, r_{ij}, and δ_{ij} elements). This approach was applied to MPC of a crude distillation unit, leading to a lower controller computational cost and less controller effort. Shah *et al.* [35] studied MPC tuning for SISO systems, via closed-loop pole placement when constraints were not active. The relationships between the tuning parameters and the desired closed-loop poles and zeros were derived and then an optimization problem that minimized the difference between the desired closed-loop CP and process closed-loop CP was solved to obtain optimal values of $\boldsymbol{Q}$ and $\boldsymbol{R}$. This approach was applied to GPC of a reactor, leading to a desired performance in terms of a rise time and overshoot. The same authors [36] then presented a systematic approach for determining MIMO GPC tuning parameters. This approach is based on the description of a desired behavior of the closed-loop system. The robustness of the closed loop system to model mismatch was studied using robust linear control theory. After developing desired closed-loop transfer functions, the weight matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ were obtained in two steps. First, the gain of the desired transfer function was obtained by minimizing the error between the true transfer function and the desired transfer function. Second, the $\boldsymbol{Q}$ and $\boldsymbol{R}$ were obtained by solving an optimization problem that also minimized the error between the true transfer function and the desired transfer function. This approach was tested on a binary distillation column, and the resulting control quality were satisfactory (less overshoot and a shorter rise time). Di Cairano *et al.* [71] developed two methods for selecting the MPC weight matrices. These methods are applicable to MIMO plants and are based on matching an unconstrained linear model-predictive controller to a desired unconstrained linear time-invariant (LTI) state feedback controller. The matching problems were formulated as constrained optimization problems, the solutions of which are the optimal values of the weight matrices, $\boldsymbol{Q}$ and $\boldsymbol{R}$. Júnior *et al.* [46] introduced an optimization-based method to tune the parameters $\boldsymbol{Q}$, $\boldsymbol{R}$, $\boldsymbol{P}$, and $\boldsymbol{M}$ of a constrained MPC with model uncertainty. Their method considered a worst-case control scenario in terms of the resiliency index[72] and the condition number in the model uncertainty description. The user can choose desired performance functions. The resulting mixed-integer constrained nonlinear multi-objective optimization problem was solved using a particle swarm optimization technique. The ability of this tuning method to handle plant-model mismatch was evaluated by testing the method on the Shell heavy oil fractionator benchmark problem [73]; the controller provided a closed-loop performance close to the nominal one even in the presence of model uncertainty. Yamashita *et al.* [54] proposed two multi-objective optimization-based tuning methods for the weight matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ in the state-space formulation of MPC. One of the methods involve input-output pairing, normalizing manipulated inputs and controlled outputs, and solving an optimization problem that minimizes the sum of the squared errors between the output closed-loop response and a reference trajectory. The other method considers the same performance index but finds feasible $\boldsymbol{Q}$ and $\boldsymbol{R}$ values that provide the response closest to a desired feasible response. When applied to the Shell heavy oil fractionator benchmark problem [73], the second method provided a closed-loop response similar to the one reported in Ref. [69]. Abrashov *et al.* [74] sought a robust multivariable generalized-predictive controller, where a robust controller was defined as one having the same nominal performance for a wide range of plant-model parameter values. Their optimization approach calculates the values of the weight matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ that minimize the variance of a gain factor subject to constraints on the deviation of the closed-loop plant-output response from a desired (reference) one. This approach was applied to a steering-wheel position control problem, and a satisfactory performance (maximum overshoot ratio of 15%) was achieved. Santos *et al.* [75] presented a method to tune MPC for non-square systems. Their controller formulation included soft constraints on the controlled variables to maintain the variables within their desired ranges. Their controller tuning approach involved three steps: (i) the selection of a desired closed-loop response; (ii) the calculation of optimal scales for the manipulated variables; and (iii) the calculation of the weight matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ via solving an optimization problem that minimizes both the deviation of the simulated closed-loop plant output response from a desired reference response and soft constraint slack variables. This approach can handle disturbances and model-plant mismatch. It provided satisfactory control performance when it was tested on the Shell benchmark problem. Romero *et al.* [44] developed an MPC approach for tuning $\boldsymbol{Q}$ and $\boldsymbol{R}$ for the GPC formulation; they calculated optimal values of $\boldsymbol{Q}$ and $\boldsymbol{R}$ by solving an optimization problem that minimized the gain margin subject to high-frequency noise rejection, satisfactory output-disturbance rejection, robustness to variations in the gain of the plant, and a limit on the phase margin. Olesen *et al.* [51] [52] proposed an optimization-based tuning approach for offset-free MPC based on a MIMO ARX model. By modifying the stochastic part of the ARX model, an additional tuning parameter, α_j, was introduced for each controlled output. Olesen *et al.* [51] [52] calculated the optimal values of the parameters $q_{ij}\left(i = 1, \cdots, n_y;\ j = 1, \cdots, P_i - P_{o_i} + 1\right)$, $r_{ij}\ \ (i = 1,\ \ \cdots,\ \ n_u;\ j = 1,\ \ \cdots,\ \ M_i)$, and $\alpha_j\left(j = 1, \cdots, n_y\right)$ by solving an optimization problem that minimized the integral of the absolute error (IAE) in the presence of the setpoint and disturbance changes, subject to $\alpha_j \in [0, 1]\left(j = 1, \cdots, n_y\right)$, and $q_{ij}\left(i = 1, \cdots, n_y;\ j = 1, \cdots, P_i - P_{o_i} + 1\right)$ and $r_{ij}\ \ (i = 1,\ \ \cdots,\ \ n_u;\ j = 1,\ \ \cdots,\ \ M_i)$ being within their the lower and upper bounds. This approach was applied to the Wood–Berry distillation column [64] and a cement mill process, and was shown to achieve better setpoint tracking and disturbance rejection compared with nominal model-predictive controllers. Suzuki *et al.* [76] studied tuning of the weight matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ for SISO and MIMO FOPDT model-based MPC. They obtained optimal values of the weight matrices by solving an optimization problem that minimized the overshoot, steady-state error, settling time, and rise time subject to upper and lower bounds on the elements of the penalty matrices. Francisco *et al.* [58] proposed a method to tune MPC by setting $\boldsymbol{Q} = I$, while $\boldsymbol{R}$ and M were optimally obtained by solving an optimization problem that minimizes the control effort and the disturbance effect subject to $f < 1$, where f was a function of the ∞-norm of user-set weights and sensitivity functions between the load disturbance and the output. This tuning approach was tested on the Benchmark Simulation Model no. 1 (BSM1) [58]; the results were satisfactory in terms of control effort and disturbance rejection. The weight-matrix tuning guidelines that were reviewed in this section are summarized in Tables 4, 5 and 6, and those reviewed in Ref. [19] are presented in Tables S5 and S6 of the SI.

Table 4. Tuning guidelines for the weights on the controlled variable errors (PP = pole placement; SA: sensitivity analysis).

Ref.	$\boldsymbol{Q}$	Method
26-27, 53	via PP	DMC [26], GPC [27], State-space [53]
30-31, 38, 55-56	$\boldsymbol{I}$	GPC
32, 35-36, 44, 74	via optimization	GPC
33, 57-58	$\boldsymbol{I}$	State-space
46, 51-52, 54, 70, 77-78	via optimization	State-space
30, 79	$\boldsymbol{I}$	
48, 67	$\boldsymbol{C}^T\boldsymbol{C}$	State-space
39-42, 66	$\boldsymbol{I}$	DMC
47, 69, 71, 75-76, 80	via optimization	
81	via SA	
82	$\boldsymbol{I}\rho_q,\ 0 < \rho_q \leq 1$	State-space
28	via PP	State-space
65	$F\left(C_y^{-1}\right)$ C_y: output covariance	
59	$\boldsymbol{I}$	SISO DMC

Table 5. Tuning guidelines for the weights on the rates of change of inputs (PP = pole placement; SA: sensitivity analysis).

Ref.	$\boldsymbol{R}$	Method
26-27, 53	via PP	DMC [26], GPC [27], State-space [53]
31	via PP	GPC

33, 67	$\boldsymbol{I}\rho,\ \rho$ is very small	State-space
46, 51-52, 54, 58, 70, 77-78	via optimization	State-space
30	$\boldsymbol{I}\rho$ $\rho = 0.317\ \Gamma^{1.26}\beta^{0.422} \exp\left(-0.113\frac{d}{\tau_c}\right) + 0.852\ \Gamma^{0.92} \exp(-208.064\beta^{5.071}) + 0.029\ \Gamma^{1.551}\left(\frac{d}{\tau_c}\right)^{0.202}$	GPC
32, 35-36, 44, 74	via optimization	GPC
38	$\boldsymbol{I}\rho$ $\rho = \frac{(p_1\ p_2)^{b_1}}{(p_1 +\ p_2)^{b_2}} + b_3 \times (p_1 p_2)^{b_4} \times \sigma - \sigma^{b_5}$ $b_1 = -0.064,\ b_2 = 0.8,\ \ b_3 = 2.0374,\ b_4 = 2.0510,\ \ b_5 = 0.0746$ $\rho = (\omega)^{b_1}(\omega + \varepsilon)^{b_2} + b_3 \times (\omega)^{b_4} \times \sigma - \sigma^{b_5}$ $b_1 = -0.0342, b_2 = -0.0460, b_3 = 1.1365,\ b_4 = 0.0242,\ b_5 = 0.0021$	GPC
30, 79	$\boldsymbol{0}$	
39-40	$\boldsymbol{I}\rho$ $\rho = a\ (\frac{\theta}{\tau_c} + 0.94)^{0.15} * \sigma^{0.9} * K^2$	DMC
69, 71, 75-76 47, 80	via optimization	
55	$\boldsymbol{I}\rho$ ρ achieves minimum input-output variance	GPC
56	$\boldsymbol{I}\rho$ ρ =0.01 for unconstrained MPC, ρ =0.1 for constrained MPC	GPC
81	via SA	
66	$\boldsymbol{I}\rho,\ \Phi_L < \rho < \Phi_H$	DMC
82	$\boldsymbol{I}\rho_r,\quad 0 < \rho_r \le 1$	State-space
57	$\boldsymbol{I}\rho,\quad \rho\ close\ to\ 1$	State-space
41	$\boldsymbol{I}\rho,\quad \rho = 0.1$	DMC
48	$\boldsymbol{I}\rho,\ \rho$ VO	State-space
28	via PP	State-space
65	$\boldsymbol{R} = F(C_u^{-1}),\quad C_u$ = input covariance	
42	via optimization	DMC
59	$\boldsymbol{\lambda I}$ $\lambda = \Gamma P k^2,\ \ \frac{0.0146\tau_{c_{11}}}{\theta_{11} + \tau_{c_{11}}} \le k$	SISO DMC

Table 6. Tuning guidelines for the weights on the magnitudes of inputs.

Ref.	Δ	Method
33	$\boldsymbol{I}$	State-space
30, 79	$\boldsymbol{0}$	
80	via optimization	

[59]	**0**	Stable Systems with a FOPDT Model

3.3 Reference Trajectory

The value of a reference trajectory parameter β_j directly affects the speed of the response of the controlled output y_j. As defined by Eqs.(11)-(16), the values of $\beta_1, \cdots, \beta_{n_y}$ should be within [0, 1]. As $\beta_j \rightarrow 1$, the reference trajectory of y_j and thus the closed-loop response of y_j becomes more sluggish [60].

Yamashita *et al.* [33] extended an existing method that considers a reference trajectory for output tracking, to the case of zone control and input target. In the output zone control strategy, instead of forcing a controlled output to be at its setpoint value, MPC forces the controlled output to be within an output zone. Yamashita *et al.* [33] assumed that a real-time optimizer calculates and supplies feasible setpoint targets to MPC. With this assumption, there will not be any conflicting setpoint targets or output zone violations. Yamashita *et al.* [33] suggested setting each β_j such that $\beta_j = \exp\left(-\frac{t_s}{\tau_{c_j}\Omega_j}\right)$, where τ_{c_j} is the apparent time constant of y_j and Ω_j is a user-set performance factor. As Ω_j increases, the speed of the response of y_j decreases. The reference-trajectory tuning guidelines that were reviewed in this section are summarized in Table 7.

Table 7. Tuning guidelines for the reference trajectory parameter.

Ref.	β_j
[33]	$\exp\left(-\frac{t_s}{\tau_{c_j}\Omega_j}\right)$
[30, 79]	$F\ (\Lambda_{RT}\ , \Lambda_{DR})$

4. Kalman Filter Gain and Covariance Matrices

The Kalman filter (KF) have been used widely for state estimation in the presence of disturbances and model uncertainty [83]. This section reviews methods that have been developed for setting Kalman filter covariance matrices $\boldsymbol{Q_w}$ and $\boldsymbol{R_v}$ since the publication of the review article by Garriga and Soroush [19] in 2009.

In the past decade, within the framework of MPC several approaches have been introduced to estimate the covariance matrices $\boldsymbol{Q_w}$ and $\boldsymbol{R_v}$. These include covariance matching [84] and maximum likelihood [85-86]. Hredzak *et al.* [87] implemented state-space MPC on a hybrid battery-ultracapacitor power source in real time. They proposed initially setting $\boldsymbol{Q_w}$ = I and $\boldsymbol{R_v}$ = I. The measurement covariance $\boldsymbol{R_v}$ was then set according to φI, where φ is a scalar. A small value was recommended for φ, since the larger is the φ value, the weaker is the measurement-based updating state estimation. The value of φ was found through empirical tuning. As for $\boldsymbol{Q_w}$, which is a diagonal matrix with $q_{w_{ii}}$ elements, a good value for each element was found by observing the performance of the state estimator (KF) both in simulations and in real time. A large value for $q_{w_{ii}}$ was recommended, since the larger $q_{w_{ii}}$ is, the stronger is the measurement-based updating state estimation. Once $\boldsymbol{R_v}$ and $\boldsymbol{Q_w}$ were found, the KF gain matrix was calculated as explained in Section 2. The controller was able to maintain the battery current and state of charge, and the ultracapacitor current and voltage within their limits[87].

5. Self- (Auto-) Tuning Methods

Self-tuning has an apparent benefit; that is, a control engineer is no longer required to be highly knowledgeable about the process system to tune a controller. Self-tuning methods update tuning parameters using an optimization method. Thus, controllers are optimally tuned. However, this requires solving an additional complex optimization problem online at each time instant.

Taeib *et al.* [47] calculated the tuning parameters (***P*, *M*, $\boldsymbol{R}$**, and $\boldsymbol{Q}$) via the minimization of a weighted sum of the squared output error and the squared control effort, subject to both $\boldsymbol{Q}$ and $\boldsymbol{R}$ being positive and the horizons being positive integers. This approach was tested on a nonlinear chemical reactor. This study showed that the tuning method provides a faster and less oscillatory closed-loop controlled output response. He *et al.* [30, 79] proposed a two degrees of freedom automated tuning approach for MPC of SISO systems. They added two filters to the reference trajectory − a reference tracking filter with a tuning parameter, λ_{RT}, and a disturbance rejection filter with a tuning parameter, λ_{DR}. The weight matrices were then set according to: $\boldsymbol{Q} = I$ and $\boldsymbol{R} = \Delta = 0$. An optimization method was applied to minimize the settling time to obtain the λ_{RT} and λ_{DR} values, subject to constraints on the overshoot and the rates of change of the controlled variables. This approach was applied to an industrial paper-making process and yielded a very smooth response. Waschl *et al.* [77] also proposed a self-tuning approach for tuning $\boldsymbol{Q}$ and $\boldsymbol{R}$ for state-space MPC. In their approach, $\boldsymbol{Q}$ and $\boldsymbol{R}$ are optimally obtained by solving an optimization problem that minimized the tracking error and the applied actuator energy is subject to $\boldsymbol{Q} > 0$ and $\boldsymbol{R} > 0$. This approach requires a predefined tracking performance and control action scenarios. It was tested on an integral gas engine, where the compressor was subjected to a load change and a setpoint change. The performance of model-predictive controller tuned using this approach was then compared to that of a model-predictive controller tuned offline by an expert control engineer. MPC tuned using this approach shows optimal performance in MIMO systems. Waschl *et al.* [78] extended this work to obtain offset-free state-space MPC. Their MPC formulation included a well-tuned proportional–integral (PI) controller as a reference. This approach yielded an impressive increase in the efficiency and performance of MPC when applied to the Wood and Berry distillation column example [64].

Tran et al. [88] used the settings $\boldsymbol{Q} = I$ and $\boldsymbol{R} = \rho\, I$, where ρ was obtained by solving an optimization problem that minimized the variance of the output error; this yielded excellent tracking behavior when applied to a binary distillation column. Jeronymo *et al.* [42] obtained an optimal ρ value for a SISO plant with a FOPDT model-based DMC by solving two optimization problems. The first problem minimized the imaginary part of the zeros to null and minimized the absolute value of the poles to achieve the shortest T_s. The optimal ρ value resulting from the first problem was applied as an initial solution to the second problem, which minimized a weighted sum of the setpoint tracking error and control effort to obtain an optimal value of ρ. When applied to a nonlinear control valve, this approach showed a better setpoint tracking performance than the tuning guidelines of Shridhar and Cooper [22].

Moumouh et al.[89] proposed an optimization-based approach to tune parameters of MPC for constrained second-order SISO processes. Their approach uses particle-swarm optimization (PSO) and machine learning to tune the control and prediction horizons and the weights on the rate of change of the manipulated input while assuming $\boldsymbol{Q} = I$. They tested their approach on a FOPDT process, which showed a satisfactory MPC performance in terms of the level of overshoot, the integral of squared error, the response time, and the rise time. Turki et al. [90] proposed an MPC tuning approach for SISO linear time-variant systems. Their approach guarantees closed-loop stability and is computationally efficient. It determines the control and prediction horizons and the weights on the rate of change of the manipulated input while assuming $\boldsymbol{Q} = I$. It determines the control horizon by solving a linear matrix inequality problem. The approach was tested on an example with a second-order transfer function.

Santos et al.[91] extended their previous work [75] to propose an optimization-based tuning approach for robust MPC of constrained square systems. They showed the performance of MPC tuned using this approach by implementing MPC on a spherical tank system and on a continuous stirred tank reactor with a separation column and a recycle. Ira et al.[92] studied auto-tuning of MIMO MPC using a neural network model that described the dependence of the closed-loop MPC performance in terms of control quality measures such overshoot and settling time, on the MPC weights. They demonstrated the control quality obtained using this tuning approach through simulating MPC of air path in a diesel engine.

6. Constrained MPC Tuning Guidelines

A major advantage of MPC is that its output (control action) is optimal in the presence of constraints; the action is the solution to a desired constrained optimization problem that is solved at each time instant. MPC tuning guidelines that consider the impacts of constraints on the closed-loop performance have also been developed. Huusom *et al.*[55] studied tuning of SISO MPC in the presence of magnitude and rate-of-change input constraints. They set $\boldsymbol{Q} = 0.5\boldsymbol{I}$, $\boldsymbol{R} = \rho\boldsymbol{I}$, $\Delta = 0, \beta_1 = 0, P_{o_1} = 1,$ and $M_1 = P_1$. They suggested using a sufficiently large value for the prediction horizon P_1 by inspecting the open-loop response of the plant. They proposed using input and output variances of the closed-loop system to set ρ automatically. Their approach involves calculating the variances at different values of ρ, from minimum variance control to no control, and choosing the value of ρ corresponding to the inflection point of a log-log plot of the variances. This ρ value was found to give a good balance between input and output variance.

Turki *et al.* [48] also suggested an optimization approach for tuning predication horizons in MPC of linear controllable MIMO plants with active constrains. Their optimization problem has three performance indices (a stability degree index, an error index, and a rapidity index). Based on the optimization, Turki *et al.* [48] recommended setting $\boldsymbol{Q} = C^T C$ and $\boldsymbol{R} = \rho \boldsymbol{I}$.

Júnior *et al.* [46] introduced a method to optimally tune constrained MPC with model uncertainty. Their method requires solving a multi-objective optimization problem that provides robust tuning. They solved the optimization problem using a PSO technique and tested the tuning guidelines on the Shell heavy oil fractionator benchmark example [73] with a large process-model mismatch. They reported that the controller was able to achieve a closed-loop response close to the nominal behavior even in the presence of model uncertainty. Waschl *et al.* [78] extended the work to obtain offset-free state-space MPC, where the MPC used a well-tuned proportional–integral (PI) controller as a reference in the closed loop to ensure an offset-free MPC. This approach yielded an impressive increase in the efficiency and performance of MPC by reducing computation when applied to the Wood–Berry distillation column problem [64].

6. Case Study I: Wood and Berry Distillation Column

We also consider the Wood and Berry binary (methanol-water) pilot-scale distillation column example [64]. The column has eight trays, a total condenser, and a basket-type reboiler. The overhead and bottom methanol compositions (x_D and x_B, wt. %) are controlled by adjusting the reflux flowrate and the bottom steam flowrate (R and S, lb/min). This example has been used in many benchmark control studies. The manipulated variables are assumed to have the following ranges: $1.15 \leq R \leq 1.35$ lb/min and $1.15 \leq S \leq 1.35$ lb/min. Within MATLAB, Simulink and the MPC toolbox were used to simulate the process under a model-predictive controller. For this example, the MPC tuning guidelines proposed in Ref.[22, 51, 67, 78] yielded the controllable tunable parameter values given in Table 8.

Table 8. Controller tuning parameter values for the distillation column.

Parameter	Ref.[22]	Ref.[51]	Ref.[67]	Ref.[78]
P_{o_1}, P_{o_2}	1	1	1	1
P_1, P_2	108	400	300	300
M_1, M_2	2	300	5	30
$\boldsymbol{Q}$	I	$87\begin{bmatrix}1 & 0\\ 0 & \frac{5.8}{8.7}\end{bmatrix}$	I	$140{,}000\begin{bmatrix}1 & 0\\ 0 & \frac{2.0}{1.4}\end{bmatrix}$
$\boldsymbol{R}$	$71\begin{bmatrix}1 & 0\\ 0 & \frac{23.5}{7.1}\end{bmatrix}$	$49{,}000\begin{bmatrix}1 & 0\\ 0 & \frac{6.9}{4.9}\end{bmatrix}$	$0.1I$	$39{,}000\begin{bmatrix}1 & 0\\ 0 & \frac{1.0}{39.0}\end{bmatrix}$
Δ	0	0	0	0
β_1, β_2	0	0	0	0

Figures 1 and 2 compare the distillation column input and output closed-loop responses obtained under MPC with the four sets of tunable parameter values (Table 8). In the case of a step change (from 70 to 90 at $t = 0$) in the x_D setpoint (Figure 1), the best responses of both controlled variables were achieved using the tuning guidelines proposed in Ref. [67, 78], which penalized the rates of change of the manipulated inputs least (which have the highest controlled-variable weight to input-rate-of-change weight ratios [10, 10; 140/39, 200, respectively]). However, the worst controlled variable responses were obtained with the guidelines suggested in Ref. [54], which has the lowest controlled-variable weight to input-rate-of-change weight ratios of 87/49,000 and 58/69,000. As expected from the controlled variable responses, the controller actions were initially most aggressive when the tuning guidelines proposed in Ref. [67, 78] were used, and were most sluggish when the guidelines in Ref. [54] were employed.

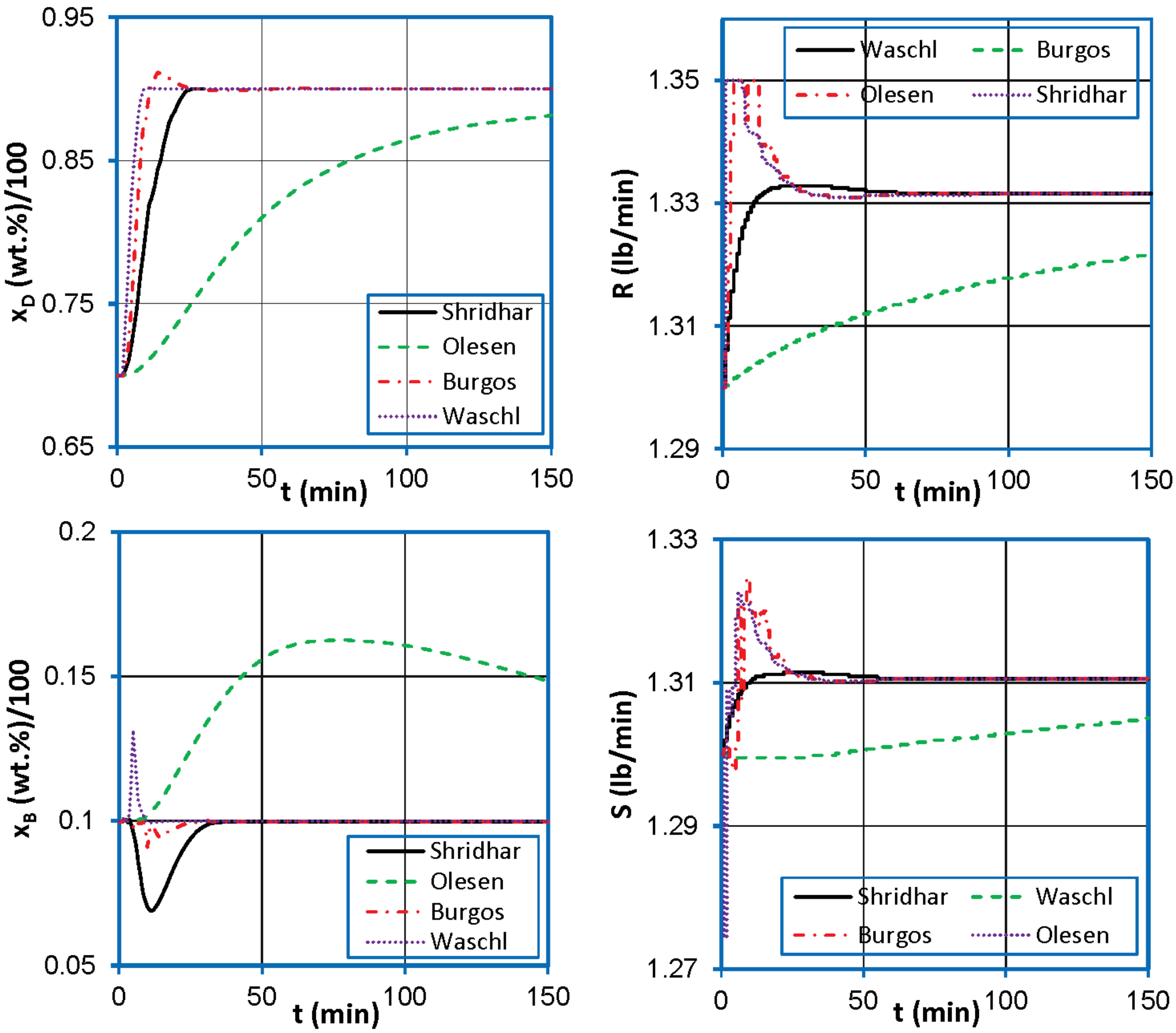

Figure 1. Input and output responses of the distillation column under MPC tuned with the four tuning approaches (Shridhar and Cooper[22], Olesen et al.[51], Burgos et al.[67], and Waschl et al.[78]) when a step change was made in the x_D set point.

In the case of a step change (from 10 to 60 at $t = 0$) in the x_B setpoint (Figure 2), best x_B responses of both controlled variables were achieved using the tuning guidelines proposed in Ref. [67, 78], and the best x_D response of was achieved using the tuning guidelines proposed in Ref. [81]

However, the worst x_B response was obtained with the guidelines suggested in Ref. [54], which is the same as what was observed in the case of a step change in the x_D setpoint. The controller actions were initially most aggressive when the tuning guidelines proposed in Ref. [67, 78] were used, and were most sluggish when the guidelines in Ref. [54] were employed.

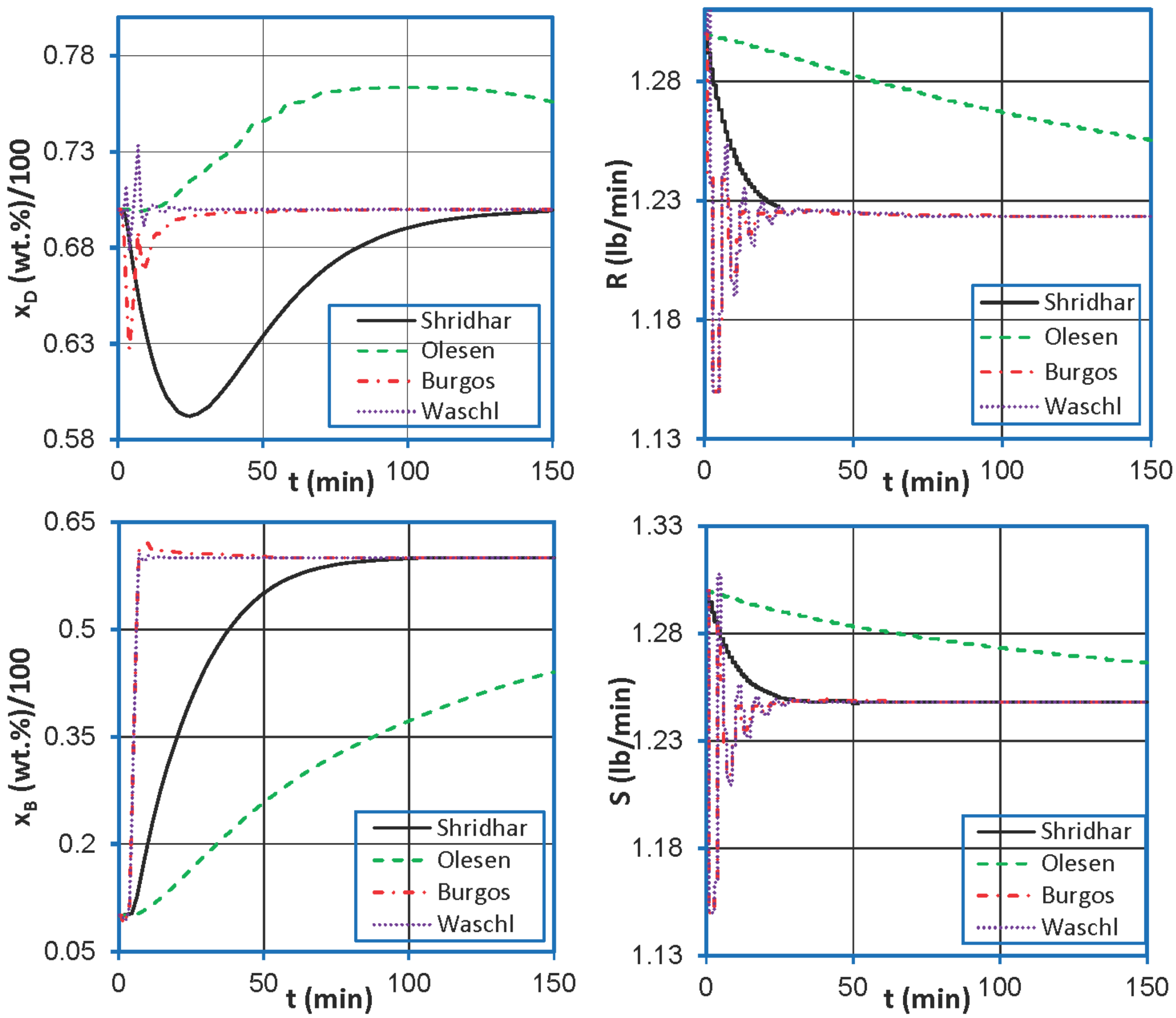

Figure 2. Input and output responses of the distillation column under MPC tuned with the four tuning approaches (Shridhar and Cooper[22], Olesen et al.[51], Burgos et al.[67], and Waschl et al.[78]) when a step change was made in the x_B set point.

7. Case Study II: Shell Heavy Oil Fractionator

We considered the Shell heavy oil fractionator described in Ref.[93]. This control problem has been used widely in the process control literature to evaluate a variety of control methods. The fractionator is described by the following 3×3 transfer function matrix:

$$G(s) = \begin{bmatrix} \frac{4.05e^{-27s}}{50s+1} & \frac{1.77e^{-28s}}{60s+1} & \frac{5.88e^{-27s}}{50s+1} \\ \frac{5.39e^{-18s}}{50s+1} & \frac{5.72e^{-14s}}{60s+1} & \frac{6.90e^{-15s}}{40s+1} \\ \frac{4.38e^{-20s}}{33s+1} & \frac{4.42e^{-22s}}{44s+1} & \frac{7.2}{19s+1} \end{bmatrix}$$

where the manipulated variables (u_1,u_2, and u_3) are the top drawn flow rate, the side drawn flow rate, and the bottoms reflux heat duty, respectively. Two controlled variables (y_1 and y_2) are the top and side end point compositions, respectively, and the third controlled variable (y_3) is the bottoms reflux temperature. The variables are all in the deviation form, and thus: $y_1(t = 0) = y_2(t = 0) = y_3(t = 0) = 0$. The manipulated and controlled variable constraints are: $-0.4 \leq u_i \leq 0.4$, i = 1, 2, 3. Simulation studies were carried out to study the performance of MPC tuned using the guidelines given in Ref. [30, 54, 57, 67] in the presence of a step change in each setpoint at t = 1 min, while the other two setpoints remained unchanged. Two cases were considered: nominal case (no process-model mismatch) and uncertainty case (when all nine steady-state gains of the model used in the controller are higher by 10%). MATLAB, Simulink and the MPC toolbox were used to simulate the process under model-predictive control. For this example, the MPC tuning guidelines given in Ref.[30, 54, 57, 67] yielded the controllable tunable parameter values given in Table 9.

T

able 9. Controller tuning parameter values for the Shell Fractionator.

Parameter	Ref.[30]		Ref.[54]	Ref. [67]	Ref.[57]
$P_{o_1}, P_{o_2}, P_{o_3}$	1		1	1	1
P_1, P_2, P_3	250		100	300	400
M_1, M_2, M_3	222		5	3	200
$\boldsymbol{Q}$	$\boldsymbol{I}$		$\begin{bmatrix} 5 & 0 & 0 \\ 0 & 4.96 & 0 \\ 0 & 0 & 2.91 \end{bmatrix}$	$\boldsymbol{I}$	$\boldsymbol{I}$
$\boldsymbol{R}$	$\begin{bmatrix} 1.705 & 0 & 0 \\ 0 & 1.702 & 0 \\ 0 & 0 & 1.6805 \end{bmatrix}$		$\begin{bmatrix} 0.001 & 0 & 0 \\ 0 & 0.0239 & 0 \\ 0 & 0 & 0.98 \end{bmatrix}$	$0.001\boldsymbol{I}$	$0.997\boldsymbol{I}$
Δ	0		0	0	0
$\beta_1, \beta_2, \beta_3$	0		0	0	0

Figures 3-5 compare the fractionator input and output closed-loop responses obtained under MPC with the four sets of tunable parameter values given in Table 9 and in the absence of any process-model mismatch. Figure 3 shows the fractionator input and output responses to a +0.7 step change in the y_1 setpoint; the four tuning approaches provide similarly good (in terms of the integral of squared error) y_1 responses. However, they yielded different y_2 and y_3 responses; the approach of Ref. [67] provided the best y_2 response, while the approaches of Ref.[30, 57] yielded equally superior y_3 responses. Figure 4 shows the fractionator input and output responses to a +0.6 step change in the y_2 setpoint. It depicts that the tuning approaches of Ref.[30, 57] provided similarly better y_1, y_2 and y_3 responses. Figure 5 shows the fractionator input and output responses to a +0.35 step change in the y_3 setpoint. It illustrates that the tuning approaches of Ref.[30, 57] provided similarly superior y_1 and y_2 responses, and that of Ref.[54] provided the best y_3 response.

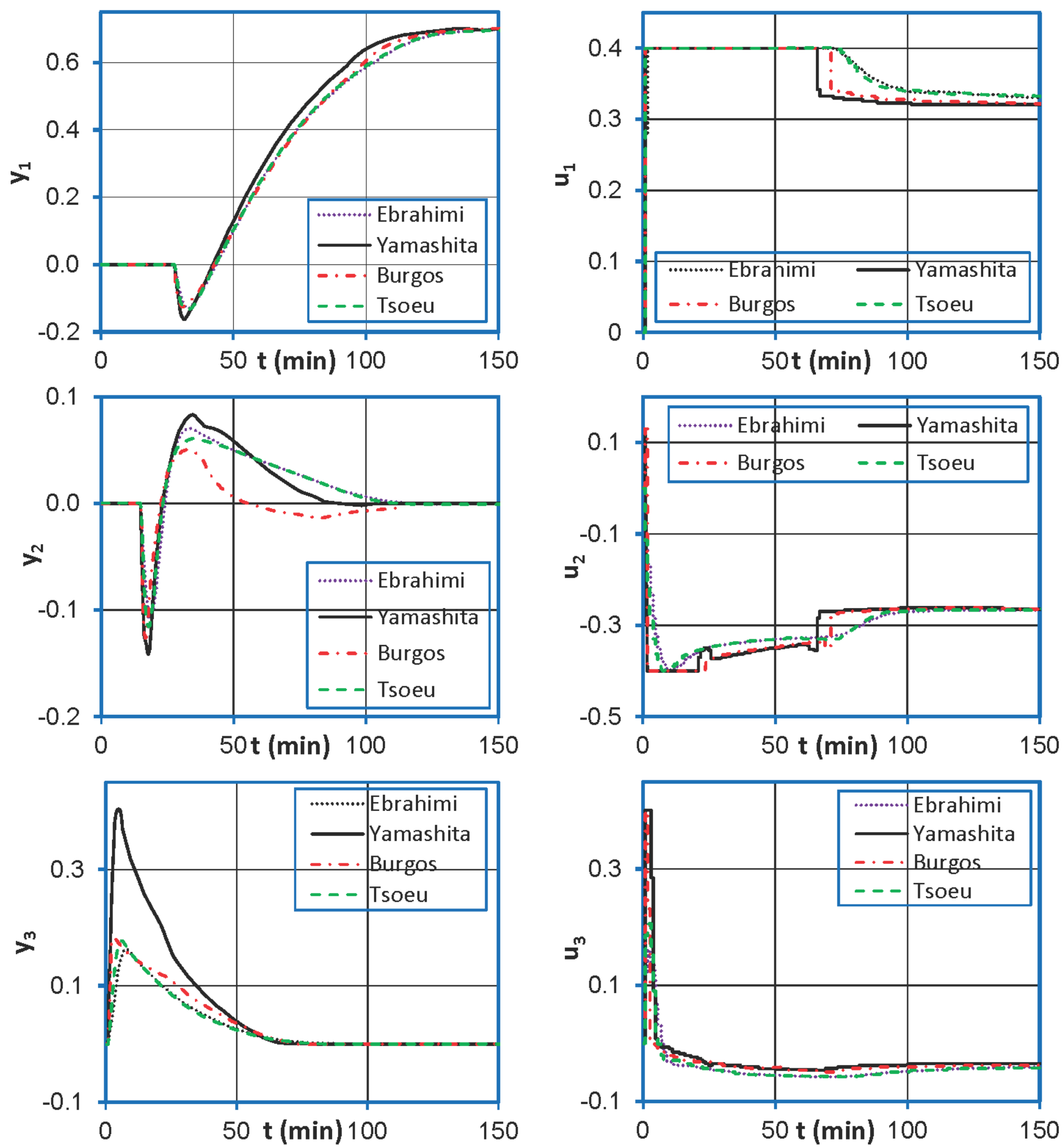

Figure 3. Input and output responses of the Shell fractionator under MPC tuned with the four tuning approaches (Ebrahimi et al. [30], Yamashita et al. [54], Burgos et al.[67], and Tsoeu and Koetje [57]) when the step change was made in the y_1 set point (no model-process mismatch).

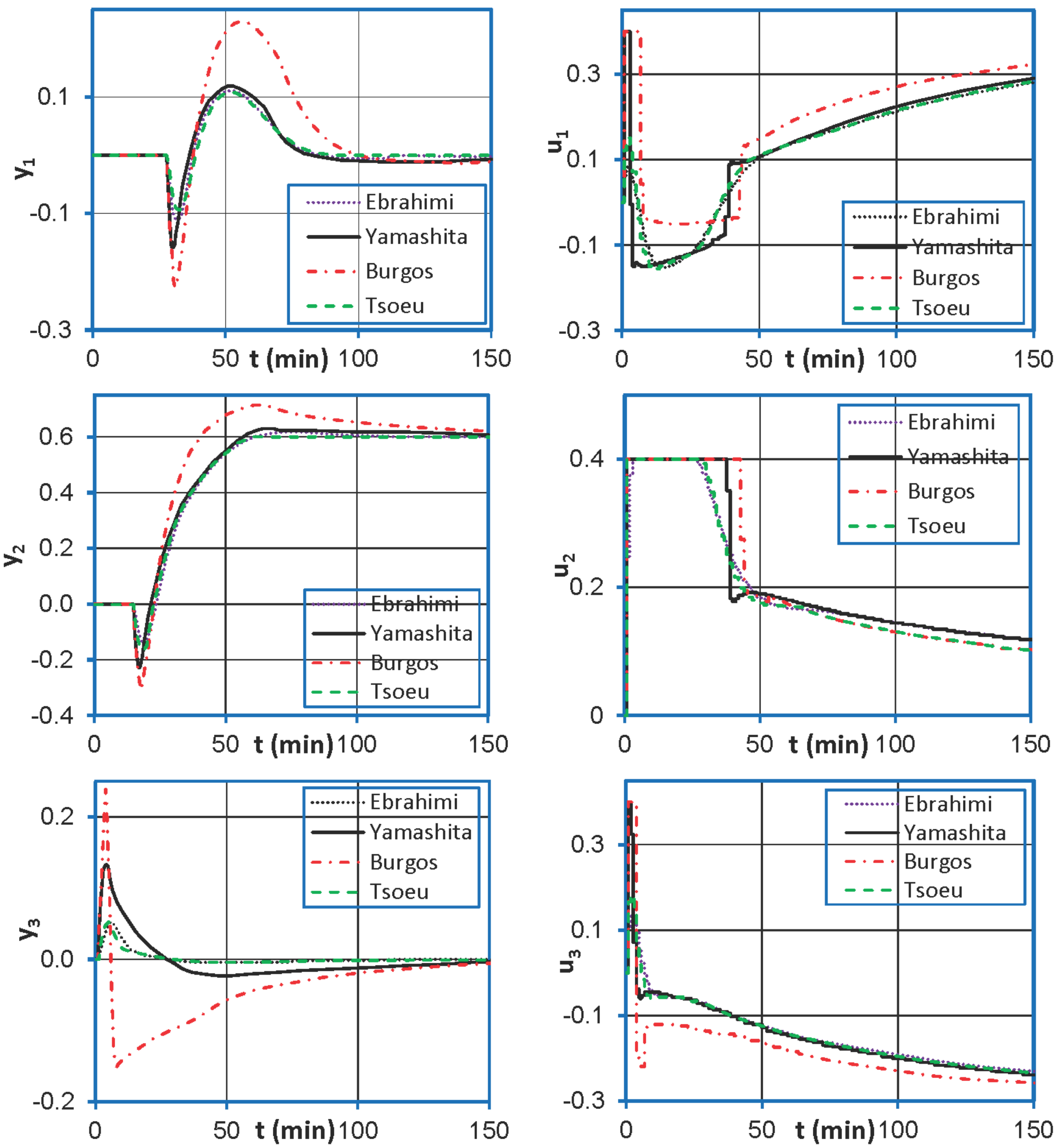

Figure 4. Input and output responses of the Shell fractionator under MPC tuned with the four tuning approaches (Ebrahimi et al. [30], Yamashita et al. [54], Burgos et al.[67], and Tsoeu and Koetje [57]) when the step change was made in the y_2 set point (no model-process mismatch).

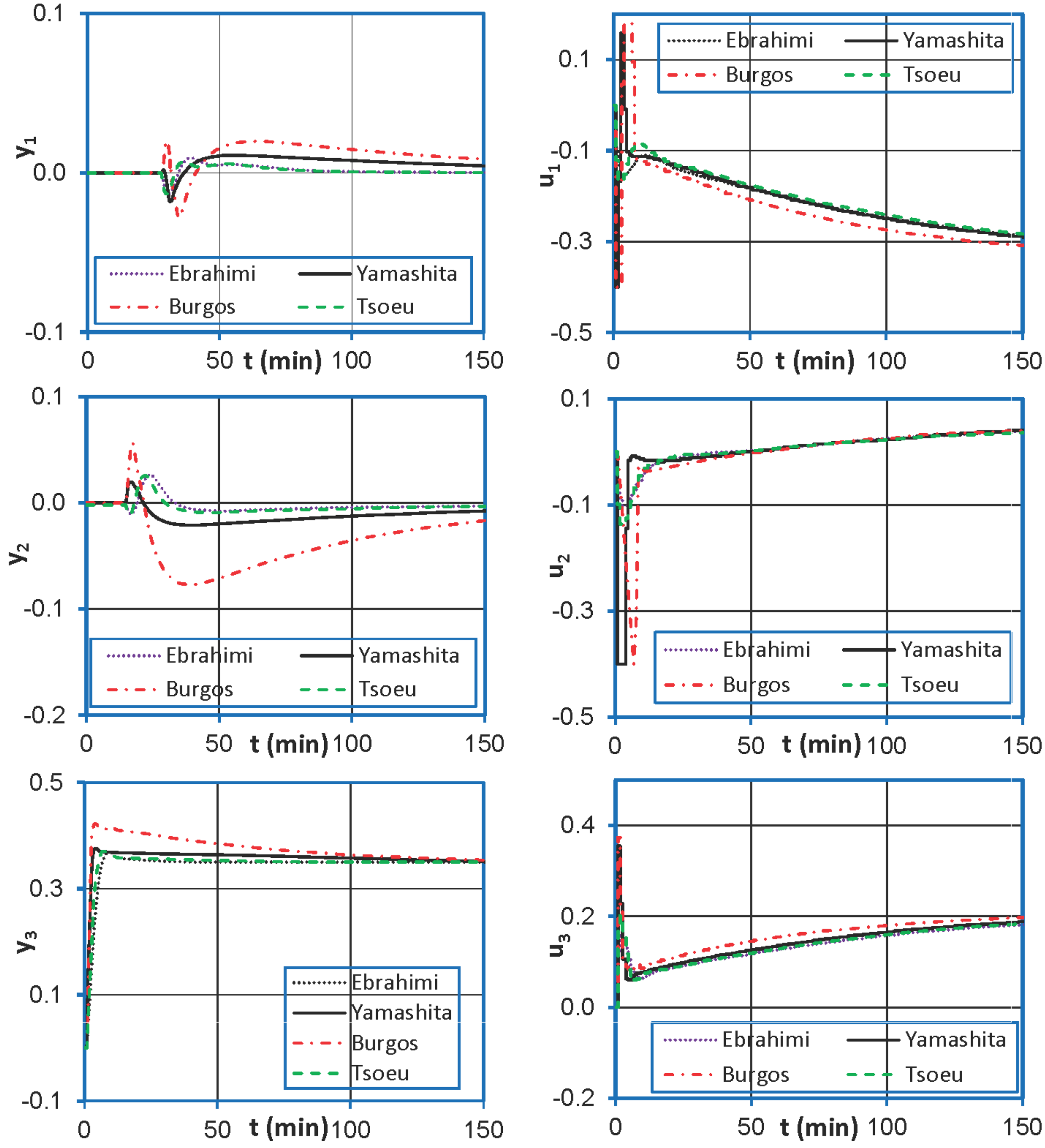

Figure 5. Input and output responses of the Shell fractionator under MPC tuned with the four tuning approaches (Ebrahimi et al. [30], Yamashita et al. [54], Burgos et al.[67], and Tsoeu and Koetje [57]) when the step change was made in the y_3 set point (no model-process mismatch).

Figures 6-8 compare the fractionator input and output closed-loop responses obtained under MPC with the four sets of tunable parameter values given in Table 9 and in the presence of the process-model mismatch. They show that MPC tuned using the four approaches provided the same relative control performances as in the case of no model-process mismatch. Furthermore, they point to the similar robustness of MPC tuned using the four approaches.

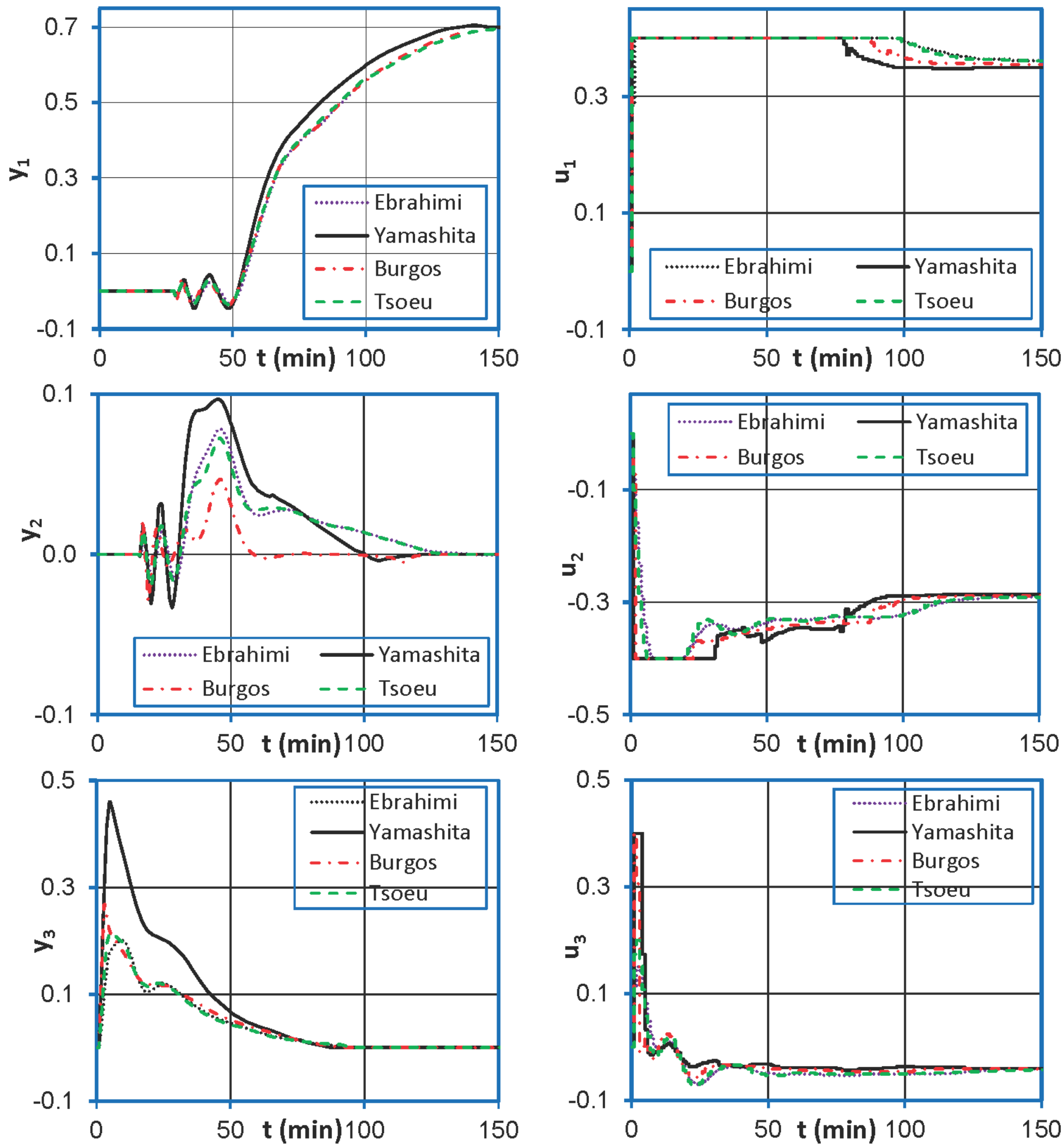

Figure 6. Input and output responses of the Shell fractionator under MPC tuned with the four tuning approaches (Ebrahimi et al. [30], Yamashita et al. [54], Burgos et al.[67], and Tsoeu and Koetje [57]) when the step change was made in the y_1 set point (in the presence of model-process mismatch).

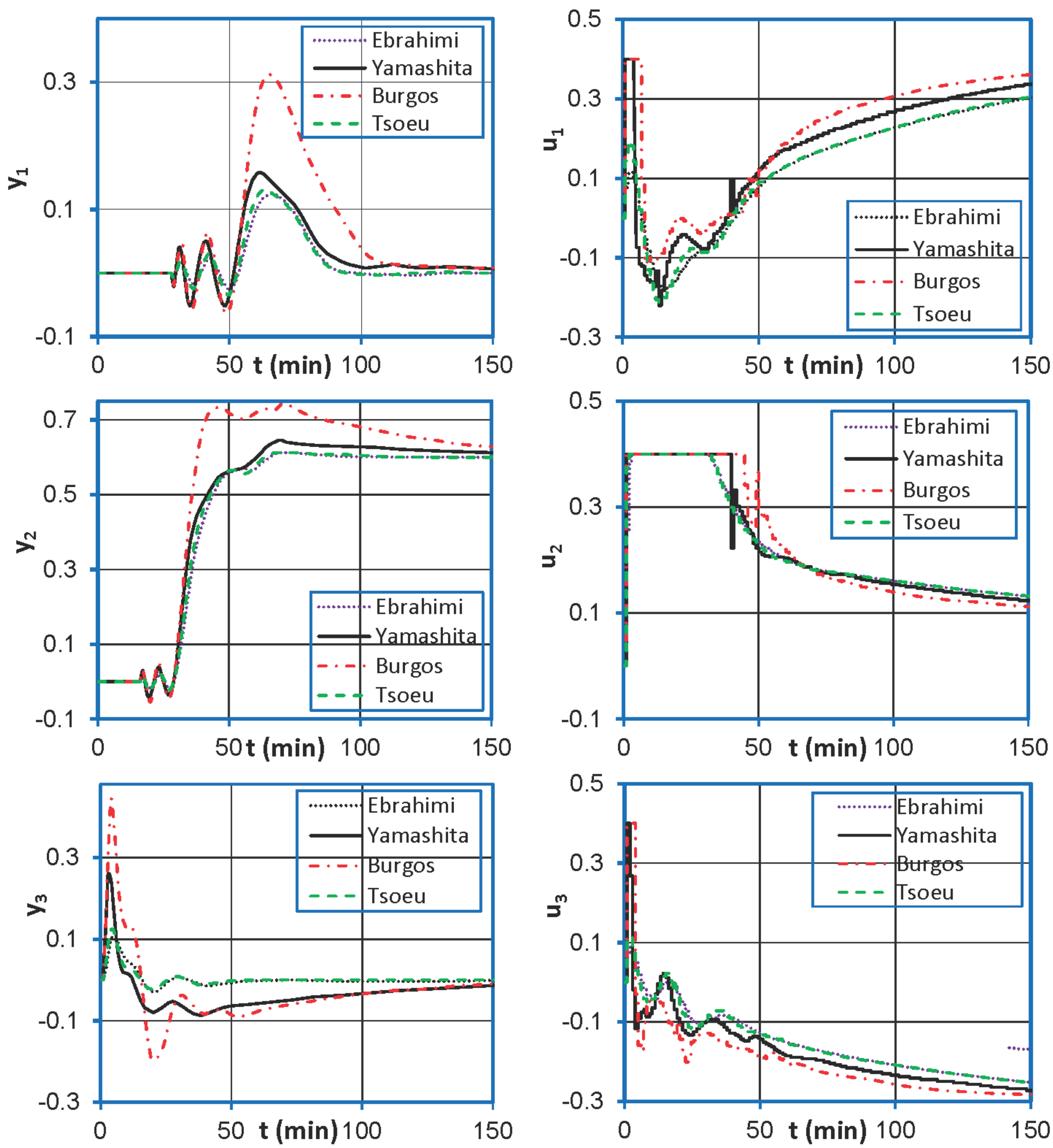

Figure 7. Input and output responses of the Shell fractionator under MPC tuned with the four tuning approaches (Ebrahimi et al. [30], Yamashita et al. [54], Burgos et al.[67], and Tsoeu and Koetje [57]) when the step change was made in the y_3 set point (in the presence of model-process mismatch).

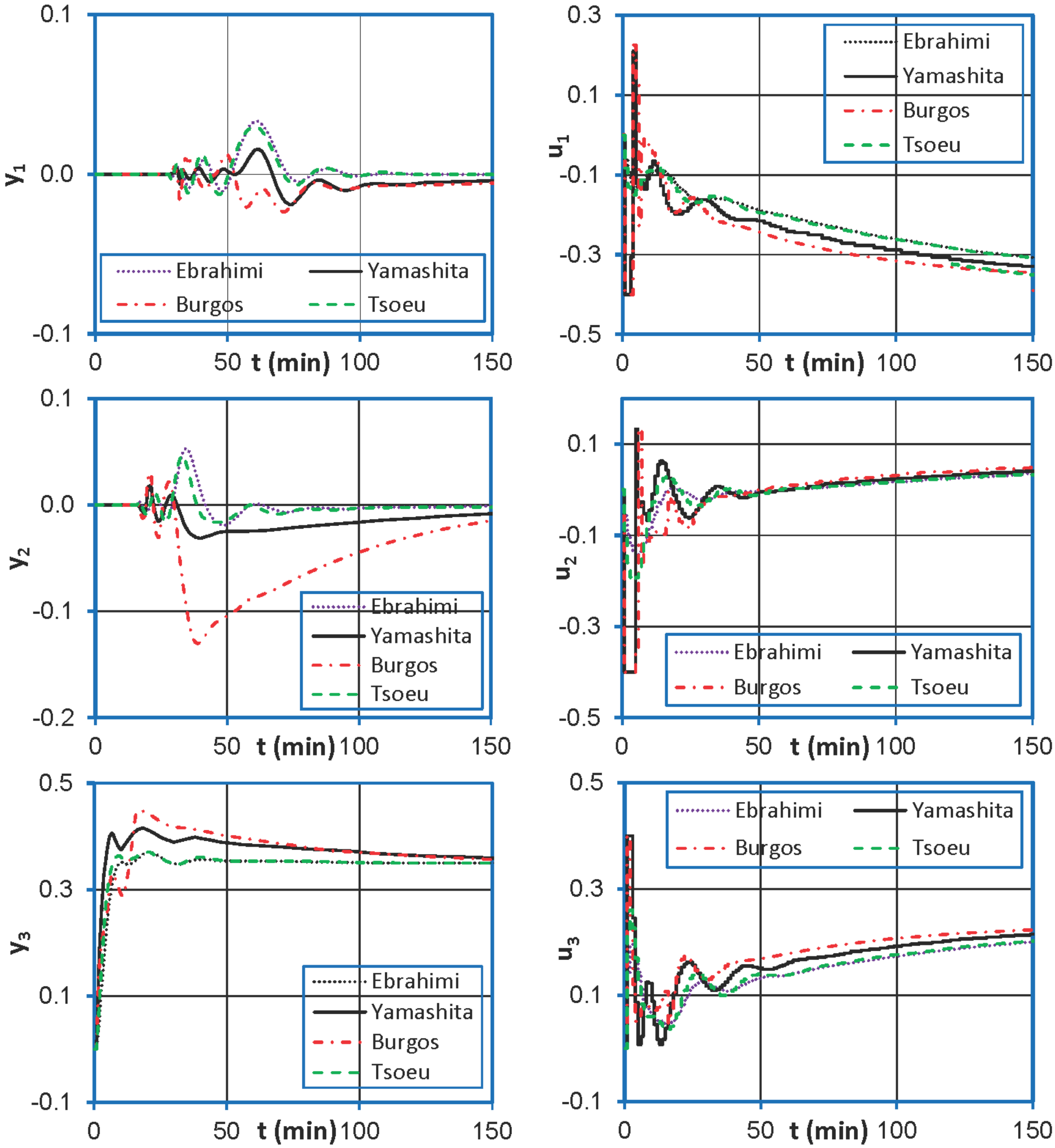

Figure 8. Input and output responses of the Shell fractionator under MPC tuned with the four tuning approaches (Ebrahimi et al. [30], Yamashita et al. [54], Burgos et al.[67], and Tsoeu and Koetje [57]) when the step change was made in the y_3 set point (in the presence of model-process mismatch).

8. Conclusion

A review of various MPC tuning strategies introduced since 2009 was presented. The review covers different MPC implementations including DMC, GPC and the state space. Optimization-based MPC tuning methods offer flexibility, as they allow for customized accounting for desired closed-loop properties and important process constraints. With the availability of increasingly faster computers and more powerful numerical methods, auto tuning MPC methods are becoming attractive. A tuning guideline that is implemented should suit the MPC formulation. The use of an accurate dynamic process model eases MPC tuning. When a model-predictive controller shows poor performance, the process model used in the MPC design should first be inspected to ensure its adequate accuracy in predicting the controlled variables.

8. Acknowledgment

M.A. would like to express his sincere gratitude to Kuwait University for its support through the KU-scholarship program. M.S. would like to acknowledge financial support from the U.S. National Science Foundation under Grant CBET-1704915. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

9. Supporting Information

MPC tuning guidelines introduced before 2010 (reviewed in Ref. [19]).

Notation

a User-set parameter [39]

$\boldsymbol{A}$ Plant state matrix

$\boldsymbol{B}$ Plant input matrix

$\boldsymbol{C}$	Plant output matrix
d	Number of sampling periods of dead time
$\boldsymbol{D}$	Input constraint matrix
$\boldsymbol{F}$	Covariance matrix of the estimation error
f	Function of the ∞-norm of user-set weights and the sensitivity function between a load disturbance and an output in Ref.[58]
$\boldsymbol{G}$	Lower triangular matrix of estimated model coefficients
g_i	Coefficient of a step response model
$\boldsymbol{H}$	Outputs constraint matrix
h_{min}	Smallest output constraint bound
H_∞	Linear time-invariant controller
k	Sampling instance
K	Steady-state gain
$\boldsymbol{M}$	Vector of control horizons
n_u	Number of manipulated variables
n_x	Number of state variables
n_y	Number of controlled variables
N	Degree of the characteristic polynomial
$\boldsymbol{N}$	Vector of model horizons
$\boldsymbol{P}$	Vector of the upper limits of the prediction horizons
$\boldsymbol{P_0}$	Vector of the lower limits of the prediction horizons
p_i	ith pole
$\mathbf{Q}$	Matrix of weights on controlled variables
$\boldsymbol{Q_w}$	Covariance matrix of w
$\boldsymbol{Q_v}$	Covariance matrix of v
$\boldsymbol{R}$	Weights on the rates of change of manipulated inputs
T_r	Process rise time, s
T_s	Largest controlled variable settling time in open loop, s
t_s	Sampling period, s
$\boldsymbol{u}$	Vector of manipulated variables

$\boldsymbol{x}$ Vector of state variables

$\hat{\boldsymbol{x}}$ Vector of state variables estimates

$\mathbf{Y}$ Vector of model outputs/controlled variables

$\boldsymbol{y_m}$ Vector of measurements of the controlled variables

y_{sp_j} Setpoint for the controlled variable y_j

y_{r_j} Reference trajectory for the controlled variable y_j

$\hat{y}_j$ Predicted value of the controlled variable y_j

<u>Greek</u>

α_j Parameter of a filtered white noise process in the stochastic part of the ARX model in Ref.[51-52]

β_j Reference trajectory parameter

σ User-set parameter in Ref.[39]

Δ Matrix of weights on manipulated variables magnitudes

Γ User-set tuning parameter representing control importance in Ref.[30]

ε Damping ratio

φ Scalar quantity

λ_{DR} Disturbance rejection filter parameter in Ref.[30, 79]

λ_{RT} Reference tracking filter parameter in Ref. [30, 79]

Ω_i User-set performance factor in Ref.[33]

ω Natural frequency of oscillation

τ_c Process time constant

ρ Scalar quantity

ρ_1 Value of ρ obtained from the first optimization in Ref.[42]

$\rho_{optimal}$ Optimal value of ρ found in Ref.[42]

θ Dead time of a SISO plant in terms of time, s

θ_{ij} *ij*th dead time of a MIMO plant in terms of time (θ_{ij} is the deadtime between the *i*th controlled variable and the *j*th manipulated variable), s

References

1. Darby, M. L.; Nikolaou, M., MPC: Current practice and challenges. *Control Engineering Practice* **2012,** *20* (4), 328-342.
2. Zadeh, L.; Whalen, B., On optimal control and linear programming. *IRE Transactions on Automatic Control* **1962,** *7* (4), 45-46.
3. Propoi, A. I., Use of LP Methods for Synthesizing Sampled-Data Automatic Systems. *Automation & Remote Control* **1963,** *24*, 837-.
4. Rafal, M. D.; Stevens, W. F., Discrete dynamic optimization applied to on-line optimal control. *AIChE journal* **1968,** *14* (1), 85-91.
5. Morari, M.; Lee, J. H., Model predictive control: past, present and future. *Computers & Chemical Engineering* **1999,** *23* (4-5), 667-682.
6. Qin, S. J.; Badgwell, T. A., A survey of industrial model predictive control technology. *Control engineering practice* **2003,** *11* (7), 733-764.
7. Bequette, B. W., Non-Linear Model Predictive Control: A Personal Retrospective. *The canadian journal of chemical engineering* **2007,** *85* (4), 408-415.
8. Garcia, C. E.; Morari, M., Internal model control. A unifying review and some new results. *Industrial & Engineering Chemistry Process Design and Development* **1982,** *21* (2), 308-323.
9. Allgöwer, F.; Zheng, A., *Nonlinear model predictive control*. Birkhäuser: 2012; Vol. 26.
10. Findeisen, R.; Allgöwer, F.; Biegler, L. T., *Assessment and future directions of nonlinear model predictive control*. Springer: 2007; Vol. 358.
11. Kouvaritakis, B.; Cannon, M., *Non-linear Predictive Control: theory and practice*. Iet: 2001.
12. Garcia, C. E.; Prett, D. M.; Morari, M., Model predictive control: theory and practice—a survey. *Automatica* **1989,** *25* (3), 335-348.
13. Ferramosca, A.; Rawlings, J. B.; Limón, D.; Camacho, E. F. In *Economic MPC for a changing economic criterion*, Decision and Control (CDC), 2010 49th IEEE Conference on, IEEE: 2010; pp 6131-6136.
14. Touretzky, C. R.; Baldea, M., Integrating scheduling and control for economic MPC of buildings with energy storage. *Journal of Process Control* **2014,** *24* (8), 1292-1300.
15. Heidarinejad, M.; Liu, J.; Christofides, P. D., State-estimation-based economic model predictive control of nonlinear systems. *Systems & Control Letters* **2012,** *61* (9), 926-935.
16. Mesbah, A., Stochastic model predictive control: An overview and perspectives for future research. *IEEE Control Systems* **2016,** *36* (6), 30-44.
17. Di Cairano, S.; Bernardini, D.; Bemporad, A.; Kolmanovsky, I. V., Stochastic MPC with learning for driver-predictive vehicle control and its application to HEV energy management. *IEEE Trans. Contr. Sys. Techn.* **2014,** *22* (3), 1018-1031.
18. Mesbah, A.; Streif, S.; Findeisen, R.; Braatz, R. D. In *Stochastic nonlinear model predictive control with probabilistic constraints*, American Control Conference (ACC), 2014, IEEE: 2014; pp 2413-2419.
19. Garriga, J. L.; Soroush, M., Model predictive control tuning methods: A review. *Industrial & Engineering Chemistry Research* **2010,** *49* (8), 3505-3515.
20. Cutler, C. R., and B. L. Ramaker In *Dynamic Matrix Control - A Computer Control Algorithm*, Proceedings of the Joint Automatic Control Conference, San Francisco, CA, AIChE: San Francisco, CA, 1980.

21. Richalet, J., and A. Rault, J. L. Testud, and J. Papon, Model Predictive Heuristic Control: Applications to Industrial Processes. *Automatica* **1978,** *14*, 413-428.
22. Shridhar, R., and D. J. Cooper, A Tuning Strategy for Unconstrained Multivariable Model Predictive Control. *Industrial & Engineering Chemistry Research* **1998,** *37*, 4003-4016.
23. Badgwell, T. A., Review Paper on MPC Tuning. Soroush, M., Ed. Philadelphia, PA, 2008.
24. Yamuna Rani, K., and H. Unbehauen, Study of Predictive Controller Tuning Methods. *Automatica* **1997,** *33* (12), 2243-2248.
25. Qi, Z. K.; Fisher, D. G. In *Robust stability of model predictive control*, Proceedings of 1994 American Control Conference-ACC'94, IEEE: 1994; pp 3258-3262.
26. Bagheri, P.; Ali, P. B.; Sedigh, A. K.; Sedigh, K., Analytical approach to tuning of model predictive control for first-order plus dead time models. *IET Control Theory & Applications* **2013,** *7* (14), 1806-1817.
27. Bagheri, P.; Khaki-Sedigh, A., An analytical tuning approach to multivariable model predictive controllers. *Journal of Process Control* **2014,** *24* (12), 41-54.
28. Gholaminejad, T.; Khaki-Sedigh, A.; Bagheri, P. In *Adaptive tuning of model predictive control based on analytical results*, Control, Instrumentation, and Automation (ICCIA), 2016 4th International Conference on, IEEE: 2016; pp 226-232.
29. Maurath, P. R., and A. J. Laub, D. E. Seborg, and D. A. Mellichamp, Predictive Controller Design by Principal Components Analysis. *Industrial & Engineering Chemistry Research* **1988,** *27*, 1204-1212.
30. Ebrahimi, Z.; Bagheri, P.; Khaki-Sedigh, A. In *Tuning of generalized predictive controllers for first order plus dead time models based on ANOVA*, 2015 23rd Iranian Conference on Electrical Engineering, 10-14 May 2015; 2015; pp 928-933.
31. Neshasteriz, A.; Sedigh, A. K.; Sadjadian, H., Generalized predictive control and tuning of industrial processes with second order plus dead time models. *Journal of Process Control* **2010,** *20* (1), 63-72.
32. Tran, Q. N.; Özkan, L.; Backx, A., Generalized predictive control tuning by controller matching. *Journal of Process Control* **2015,** *25*, 1-18.
33. Yamashita, A. S.; Alexandre, P. M.; Zanin, A. C.; Odloak, D., Reference trajectory tuning of model predictive control. *Control Engineering Practice* **2016,** *50*, 1-11.
34. Shehu, I. A.; Wahab, N. A. In *Applications of MPC and PI controls for liquid level control in coupled-tank systems*, Automatic Control and Intelligent Systems (I2CACIS), IEEE International Conference on, IEEE: 2016; pp 119-124.
35. Shah, G.; Engell, S. In *Tuning MPC for desired closed-loop performance for SISO systems*, Control & Automation (MED), 2010 18th Mediterranean Conference on, IEEE: 2010; pp 628-633.
36. Shah, G.; Engell, S. In *Tuning MPC for desired closed-loop performance for MIMO systems*, Proceedings of the 2011 American Control Conference, June 29 2011-July 1 2011; 2011; pp 4404-4409.
37. Sha'aban, Y. A.; Lennox, B.; Laurí, D., PID versus MPC performance for siso dead-time dominant processes. *IFAC Proceedings Volumes* **2013,** *46* (32), 241-246.
38. Neshasteriz, A. R.; Khaki-Sedigh, A.; Sadjadian, H. In *An analysis of variance approach to tuning of generalized predictive controllers for second order plus dead time models*, Control and Automation (ICCA), 2010 8th IEEE International Conference on, IEEE: 2010; pp 1059-1064.

39. Bagheri, P.; Khaki-Sedigh, A., An ANOVA based analytical dynamic matrix controller tuning procedure for FOPDT models. *AUT Journal of Modeling and Simulation* **2010,** *42* (2), 55-64.
40. Bagheri, P.; Khaki-Sedigh, A., Tuning of dynamic matrix controller for FOPDT models using analysis of variance. *IFAC Proceedings Volumes* **2011,** *44* (1), 12319-12324.
41. Kokate, R.; Waghmare, L.; Deshmukh, S. In *Review of tuning methods of DMC and performance evaluation with PID algorithms on a FOPDT model*, Advances in Recent Technologies in Communication and Computing (ARTCom), 2010 International Conference on, IEEE: 2010; pp 71-75.
42. Jeronymo, D. C.; Coelho, A. A., Minimum Realization Tuning Strategy for Dynamic Matrix Control. *IFAC Proceedings Volumes* **2014,** *47* (3), 1314-1319.
43. Shridhar, R.; Cooper, D. J., A tuning strategy for unconstrained SISO model predictive control. *Industrial & Engineering Chemistry Research* **1997,** *36* (3), 729-746.
44. Romero, M.; Tejado, I.; De Madrid, A.; Vinagre, B., Tuning predictive controllers with optimization: Application to GPC and FGPC. *IFAC Proceedings Volumes* **2011,** *44* (1), 10824-10829.
45. Clarke, D. W.; Mohtadi, C.; Tuffs, P., Generalized predictive control—Part I. The basic algorithm. *Automatica* **1987,** *23* (2), 137-148.
46. Júnior, G. A. N.; Martins, M. A.; Kalid, R., A PSO-based optimal tuning strategy for constrained multivariable predictive controllers with model uncertainty. *ISA transactions* **2014,** *53* (2), 560-567.
47. Taeib, A.; Chaari, A., Optimal Tuning Strategy for MIMO Fuzzy Predictive Controllers. *International Journal of Fuzzy System Applications (IJFSA)* **2015,** *4* (4), 87-99.
48. Turki, M.; Langlois, N.; Yassine, A. In *A practical tuning approach for multivariable model predictive control*, Control & Automation (ICCA), 2017 13th IEEE International Conference on, IEEE: 2017; pp 1107-1112.
49. Clarke, D. W., and C. Mohtadi and P. S. Tuffs, Generalized Predictive Control-Part I. The Basic Algorithm. *Automatica* **1987,** *23* (2), 137-148.
50. Clarke, D. W., and C. Mohtadi and P. S. Tuffs, Generalized Predictive Control-Part II. Extensions and Interpretations. *Automatica* **1987,** *23* (2), 149-160.
51. Olesen, D. H.; Huusom, J. K.; Jørgensen, J. B. In *A tuning procedure for ARX-based MPC of multivariate processes*, American Control Conference (ACC), 2013, IEEE: 2013; pp 1721-1726.
52. Olesen, D. H.; Huusom, J. K.; Jørgensen, J. B. In *Optimization based tuning approach for offset free MPC*, 10th European Workshop on Advanced Control and Diagnosis, Technical University of Denmark (DTU): 2012.
53. Bagheri, P.; Khaki-Sedigh, A., Closed form tuning equations for model predictive control of first-order plus fractional dead time models. *International Journal of Control, Automation and Systems* **2015,** *13* (1), 73-80.
54. Yamashita, A.; Zanin, A.; Odloak, D., Tuning of model predictive control with multi-objective optimization. *Brazilian Journal of Chemical Engineering* **2016,** *33* (2), 333-346.
55. Huusom, J. K.; Poulsen, N. K.; Jørgensen, S. B.; Jørgensen, J. B., Tuning SISO offset-free model predictive control based on ARX models. *Journal of Process Control* **2012,** *22* (10), 1997-2007.

56. Huusom, J. K.; Poulsen, N. K.; Jørgensen, S. B.; Jørgensen, J. B. In *Tuning of methods for offset free MPC based on ARX model representations*, American Control Conference (ACC), 2010, IEEE: 2010; pp 2355-2360.
57. Tsoeu, M. S.; Koetje, T. In *Unconstrained MPC tuning for prediction accuracy in Networked Control Systems*, Networking, Sensing and Control (ICNSC), 2011 IEEE International Conference on, IEEE: 2011; pp 474-479.
58. Francisco, M.; Vega, P.; Revollar, S. In *Model predictive control of BSM1 benchmark of wastewater treatment process: a tuning procedure*, Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on, IEEE: 2011; pp 7057-7062.
59. Klopot, T.; Skupin, P.; Metzger, M.; Grelewicz, P., Tuning strategy for dynamic matrix control with reduced horizons. *ISA transactions* **2018,** *76*, 145-154.
60. Seborg, D. E.; Mellichamp, D. A.; Edgar, T. F.; Doyle III, F. J., *Process dynamics and control*. John Wiley & Sons: 2010.
61. Salem, F.; Mosaad, M. I., A comparison between MPC and optimal PID controllers: Case studies. **2015**.
62. Huusom, J. K.; Poulsen, N. K.; Jørgensen, S. B.; Jørgensen, J. B., Noise modelling and MPC tuning for systems with infrequent step disturbances. *IFAC Proceedings Volumes* **2011,** *44* (1), 11226-11232.
63. Georgiou, A., and C. Georgakis and W. L. Luyben, Nonlinear Dynamic Matrix Control for High-Purity Distillation Columns. *AIChE Journal* **1988,** *34* (8), 1287-1298.
64. Wood, R.; Berry, M., Terminal composition control of a binary distillation column. *Chemical Engineering Science* **1973,** *28* (9), 1707-1717.
65. Belda, K., On-line parameter tuning of model predictive control. *IFAC Proceedings Volumes* **2011,** *44* (1), 5489-5494.
66. Bagheri, P.; Sedigh, A. K., Robust tuning of dynamic matrix controllers for first order plus dead time models. *Applied Mathematical Modelling* **2015,** *39* (22), 7017-7031.
67. Burgos, J. G.; Martínez, C. A. L.; van de Molengraft, R.; Steinbuch, M., Frequency Domain Tuning Method for Unconstrained Linear Output Feedback Model Predictive Control. *IFAC Proceedings Volumes* **2014,** *47* (3), 7455-7460.
68. Iglesias, E. J.; Sanjuán, M. E.; Smith, C. A., Tuning equation ford dynamic matrix control in siso loops. *Ingeniería y desarrollo* **2006,** (19).
69. Vallerio, M.; Van Impe, J.; Logist, F., Tuning of NMPC controllers via multi-objective optimisation. *Computers & Chemical Engineering* **2014,** *61*, 38-50.
70. Yamashita, A. S.; Zanin, A. C.; Odloak, D., Tuning the model predictive control of a crude distillation unit. *ISA transactions* **2016,** *60*, 178-190.
71. Di Cairano, S.; Bemporad, A., Model predictive control tuning by controller matching. *IEEE Transactions on Automatic Control* **2010,** *55* (1), 185-190.
72. Morari, M., Design of resilient processing plants—III: A general framework for the assessment of dynamic resilience. *Chemical Engineering Science* **1983,** *38* (11), 1881-1891.
73. Prett, D. M.; García, C. E.; Ramaker, B. L., *The second shell process control workshop: solutions to the shell standard control problem*. Elsevier: 2017.
74. Abrashov, S.; Airimitoaie, T. B.; Lanusse, P.; Aioun, F.; Malti, R.; Moreau, X.; Guillemard, F., Model Predictive Control Tuning: Methods and Issues. Application to steering wheel position control. *IFAC-PapersOnLine* **2017,** *50* (1), 11331-11336.

75. Santos, J. E. W.; Trierweiler, J. O.; Farenzena, M., Model Predictive Control Tuning Strategy for Non-Square Systems and Range Controlled Variables Based on Multi-Scenarios Approach. *Industrial & Engineering Chemistry Research* **2017,** *56* (40), 11496-11506.
76. Suzuki, R.; Kawai, F.; Nakazawa, C.; Matsui, T.; Aiyoshi, E., Parameter optimization of model predictive control by PSO. *Electrical Engineering in Japan* **2012,** *178* (1), 40-49.
77. Waschl, H.; Alberer, D.; del Re, L., Numerically efficient self tuning strategies for MPC of integral gas engines. *IFAC Proceedings Volumes* **2011,** *44* (1), 2482-2487.
78. Waschl, H.; Jørgensen, J. B.; Huusom, J. K.; Del Re, L., A tuning approach for offset-free MPC with conditional reference adaptation. *IFAC Proceedings Volumes* **2014,** *47* (3), 3062-3067.
79. He, N.; Shi, D.; Wang, J.; Forbes, M.; Backström, J.; Chen, T., Automated two-degree-of-freedom model predictive control tuning. *Industrial & Engineering Chemistry Research* **2015,** *54* (43), 10811-10824.
80. Ammar, M. E. In *Tuning model predictive controllers for cross-direction processes*, Control, Decision and Information Technologies (CoDIT), 2016 International Conference on, IEEE: 2016; pp 746-751.
81. Jain, A.; Schildbach, G.; Fagiano, L.; Morari, M., On the design and tuning of linear model predictive control for wind turbines. *Renewable Energy* **2015,** *80*, 664-673.
82. Oliveira, D.; Rodrigues, E. M.; Godina, R.; Mendes, T. D.; Catalão, J. P.; Pouresmaeil, E. In *MPC weights tunning role on the energy optimization in residential appliances*, Power Engineering Conference (AUPEC), 2015 Australasian Universities, IEEE: 2015; pp 1-6.
83. Anderson, B., JB Moore Optimal filtering Englewood cli s. New Jersey: Prentice-Hall: 1979.
84. Myers, K., and B. Tapley, Adaptive Sequential Estimation with Unknown Noise Statistics. *IEEE Transactions on Automatic Control* **1976,** *AC-21* (4), 520-523.
85. Bohlin, T., Four Cases of Identification of Changing Systems. In *System Identification: Advances and Case Studies*, 1st ed.; Mehra, R., and D. Lainiotis, Ed. Academic: New York, 1976.
86. Kashyap, R., Maximum Likelihood Identification of Stochastic Linear Systems. *IEEE Transactions on Automatic Control* **1970,** *AC-15* (1), 25-34.
87. Hredzak, B.; Agelidis, V. G.; Jang, M., A model predictive control system for a hybrid battery-ultracapacitor power source. *IEEE Transactions on Power Electronics* **2014,** *29* (3), 1469-1479.
88. Tran, Q. N.; Scholten, J.; Ozkan, L.; Backx, A., A model-free approach for auto-tuning of model predictive control. *IFAC Proceedings Volumes* **2014,** *47* (3), 2189-2194.
89. Moumouh, H.; Langlois, N.; Haddad, M. In *A Novel Tuning approach for MPC parameters based on Artificial Neural Network*, 2019 IEEE 15th International Conference on Control and Automation (ICCA), IEEE: 2019; pp 1638-1643.
90. Turki, M.; Langlois, N.; Yassine, A. In *An analytical tuning approach for adaptive mpc parameters applied to ltv siso systems*, 2018 Annual American Control Conference (ACC), IEEE: 2018; pp 1534-1539.
91. Santos, J. E. W.; Trierweiler, J. O.; Farenzena, M., Robust tuning for classical MPC through the Multi-Scenarios Approach. *Industrial & Engineering Chemistry Research* **2019,** *58* (8), 3146-3158.

92. Ira, A. S.; Shames, I.; Manzie, C.; Chin, R.; Nešić, D.; Nakada, H.; Sano, T. In *A machine learning approach for tuning model predictive controllers*, 2018 15th International Conference on Control, Automation, Robotics and Vision (ICARCV), IEEE: 2018; pp 2003-2008.
93. Prett, D.; Morari, M. In *Shell control problem*, Shell Process Control Workshop, Butterworth Publishers, Stoneham, MA, 1987; pp 355-360.