Preprints of the 19th World Congress
The International Federation of Automatic Control
Cape Town, South Africa. August 24-29, 2014

A model-free approach for auto-tuning of model predictive control

Quang N Tran * **Joni Scholten** * **Leyla Özkan** *
A.C.P.M. Backx *

* *Electrical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands (e-mail: n.q.tran@tue.nl; j.scholten.1@student.tue.nl; l.ozkan@tue.nl; a.c.p.m.backx@tue.nl).*

Abstract: A two-layer approach for the auto-tuning of model predictive control (MPC) is proposed. The bottom layer computes the weighting matrices of the cost function from a desired closed-loop bandwidth while the top layer aims at finding the optimal bandwidth. This optimum corresponds to the optimal balance between the robustness and nominal performance of the closed-loop system. To find the optimal bandwidth, the extremum seeking (ES) algorithm, a form of non-model-based adaptive optimisation, is proposed. The auto-tuning approach is tested on a binary distillation column model. It is shown that the auto-tuning approach enables the MPC system to track its optimal closed-loop bandwidth and therefore obtain the minimum output variance.

Keywords: Model predictive control, closed-loop bandwidth, auto-tuning, extremum seeking.

1. INTRODUCTION

Model Predictive Control (MPC) has been widely accepted as the control strategy for multiple input/multiple output (MIMO) systems and for processes with constraints. MPC usually solves an optimization problem with a quadratic cost function online. Like any other model-based operation support systems, the performance of MPC depends on the accuracy and appropriate maintenance of the model. Model-plant mismatch will degrade the performance of MPC if proper supervision is not conducted. Therefore, the influence of modelling uncertainty on MPC is of significant importance.

Apart from modelling and identification advances, the tuning of MPC can also be considered as a method to handle the robustness of the system. In literature, there have been several studies which dealt with modelling uncertainty in the context of controller tuning. Lee and Yu (1994) proposed a tuning method consisting of two steps: The first step is to tune the controller for nominal stability; the second step is to de-tune the controller by adjusting the Kalman filter gain and the covariance of the disturbance. The de-tuning method is based on the analysis of the sensitivity and complementary sensitivity functions in the frequency domain. A number of works (Rowe and Maciejowski (2000b), Rowe and Maciejowski (2000a), Shah and Engell (2011) and Tran et al. (2014)) investigated the matching of MPC to an H_∞ controller, so that the MPC can inherit the robustness of the H_∞ controller when constraints are inactive. On the other hand, Han et al. (2006) used min-max algorithms when the parametric model uncertainty is considered. Despite these research efforts in tuning of MPC in literature, the tuning strategies in practice that considered robustness often result in a conservative performance, which is too far from the optimal balance between robustness and nominal performance. Therefore, this research focuses on an auto-tuning method which finds the optimal tuning to achieve this balance.

Auto-tuning methods studied for MPC usually hinge upon minimising a selected performance index (Garriga and Soroush (2010)). In Al-Ghazzawi et al. (2001), every time the predicted closed-loop response is expected to exceed the specification bounds (e.g. 5% of the set-point value), the auto-tuning algorithm is turned on to adjust tuning parameters and steer the response into the limits. The predicted violation of the bounds is then considered as the performance index of the auto-tuning method. In Lee et al. (2008), the genetic algorithm (GA) and fuzzy decision making are used to satisfy different performance preferences such as the level of overshoot or the settling time of controlled outputs. Suzuki et al. (2007) employs the particle swarm optimisation (PSO) to minimise the performance index, which covers the steady-state error, settling time, rising time and the maximum value of the outputs at the end time.

In process industry, the main purpose of MPC is to reduce the variance of the output and therefore to operate the system closely to its constraint. Hence, in this work, the key performance indicator of the auto-tuning method is the output variance. The output variance is also considered in the tuning procedure of Huusom et al. (2012) and Olesen et al. (2013). In Tran et al. (2013), Tran et al. (2012) and Özkan et al. (2012), it is shown that in the presence of model-plant mismatch and measurement noise, there ex-

⋆ The research leading to these results has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° 257059, the 'Autoprofit' project (www.fp7-autoprofit.eu).

ists an optimal closed-loop bandwidth which corresponds to the minimum output variance. Hence, the aim of the auto-tuning method is to find this optimal closed-loop bandwidth which is determined by the weighting factors in the cost function. To this end, the extremum seeking (ES) method which can find the minimum of an unknown function is investigated.

ES is a form of non-model-based adaptive optimisation. It belongs to the class of gradient-based optimisation techniques and deals with systems that have an unknown input/output relation, but are known to have an extremum. The basic idea is to perturb the input of the system with a so-called dither signal and multiply the system output with the same dither signal. The output of this multiplication is an estimate of the system's gradient and this gradient estimate is used to move the input parameter to its extremum, which minimises (or maximises) the output of the system. The method has existed since the 1920s, was extensively investigated in the 1950s and 1960s and has gained renewed interest since the beginning of the new millennium (Stanković and Stipanović (2010) and Tan et al. (2010)). It has been used successfully for PID tuning (Killingsworth and Krstić (2006)), maximum power point tracking of a photo-voltaic system (Levya et al. (2011)), tuning the parameters of mobile sensors (Stanković and Stipanović (2010)), and many other applications (Tan et al. (2010)).

The remainder of this paper is organised as follows. Section 2 provides background information on MPC and introduces the two-layer auto-tuning method. Section 3 introduces and analyses the ES algorithm proposed. Section 4 provides the method to select the tuning parameters based on the link between the closed-loop bandwidth and the penalty weights. Section 5 illustrates the method with a binary distillation column. Conclusion and recommendations are given in Section 6.

2. BACKGROUND INFORMATION

A model describing the behaviour of the plant is at the heart of all MPC systems. It is assumed that the system is represented by a discrete-time model of the form

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k)$$
$$\mathbf{y}_p(k) = \mathbf{C}\mathbf{x}(k) \tag{1}$$

where $\mathbf{x}(k) \in \mathbb{R}^{n_x}$, $\mathbf{u}(k) \in \mathbb{R}^{n_u}$ and $\mathbf{y}_p(k) \in \mathbb{R}^{n_y}$ represent the states, the controlled inputs and the plant outputs of the system. This model is used to compute system predictions over a finite prediction horizon of N samples. The MPC uses these predictions to find the input sequence that minimises the cost function given by:

$$V(k) = \sum_{i=1}^{N} \hat{\boldsymbol{\epsilon}}(k+i|k)\mathbf{Q}\hat{\boldsymbol{\epsilon}}(k+i|k) + \sum_{i=1}^{N_c} \boldsymbol{\Delta}\hat{\mathbf{u}}(k+i|k)\mathbf{R}\boldsymbol{\Delta}\hat{\mathbf{u}}(k+i|k) \tag{2}$$

where $\hat{\boldsymbol{\epsilon}}(k+i|k) = \hat{\mathbf{y}}_p(k+i|k) - \mathbf{r}(k+i|k) \in \mathbb{R}^{n_y}$ is the output error at sample $k+i$; $\boldsymbol{\Delta}\hat{\mathbf{u}}(k+i|k) \in \mathbb{R}^{n_u}$ is the input increment at sample $k+i$ and N_c is the control horizon. $\mathbf{Q} = \mathbf{Q}^\top > 0$ and $\mathbf{R} = \mathbf{R}^\top > 0$ are weighting matrices on the output error and input increments. MPC solves the finite-time optimal control problem online at each time instant. Only the first element of the solution is implemented and this procedure is repeated in the next time instant. The strategy enables MPC to deal explicitly with MIMO plants and system constraints.

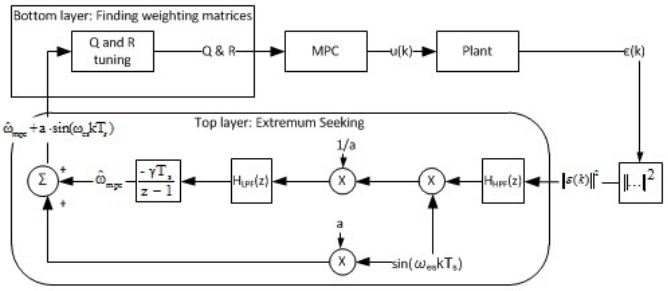

Fig. 1. Overview of the 2-layer auto-tuning method.

As stated in Tran et al. (2012), the variance of the output error β is a good indicator for the performance of the closed-loop system. Under the assumption that $\mu := E[\boldsymbol{\epsilon}(k)] = 0$, the output variance is represented by:

$$\beta = E[||\boldsymbol{\epsilon}(k) - \mu||^2] = E[||\boldsymbol{\epsilon}(k)||^2]. \tag{3}$$

Tran et al. (2012) shows that an increase in the closed-loop bandwidth ω_{mpc}, corresponding to reducing the penalty on the input energy, leads to a better rejection of (low-frequency) disturbances. This in turn results in a reduction in β. On the other hand, this also renders the controller more sensitive to modelling errors and measurement noise, which results in an increase of β. This means that there exists an optimal bandwidth ω^*_{mpc} which is a trade-off between performance and robustness and therefore results in a minimisation of β. In other words, there exists a function $\beta = f(\omega_{mpc})$ and this function is known to have an extremum, as shown in Figure 2.

Based on these observations, the proposed auto-tuning method consists of two layers:

- Top layer: Finding the optimal closed-loop bandwidth ω_{mpc} using extremum seeking.
- Bottom layer: Translating the closed-loop bandwidth from the top layer into the weighting matrices $\mathbf{Q}$ and $\mathbf{R}$.

A schematic overview of the auto-tuning method is given in Figure 1.

3. TOP LAYER: OPTIMAL CLOSED-LOOP BANDWIDTH

The top layer in the auto-tuning approach aims at finding the optimal ω_{mpc} by minimising the output variance β in (3). Although the relationship between β and ω_{mpc} is non-linear and a-priori unknown, it is assumed that an extremum $\beta^* = f(\omega^*_{mpc})$ exists.

3.1 Gradient Estimation By Input Perturbation

ES tries to estimate the gradient of the nonlinear mapping f by perturbing its input with a periodic dither signal with radial frequency ω_{es} and processing its output β. This subsection gives an analysis of the ES system of the top layer in Figure 1, based on the analysis found in Levya et al. (2011). A more in-depth analysis of the ES algorithm can be found in Krstić and Wang (2000) and Tan et al.

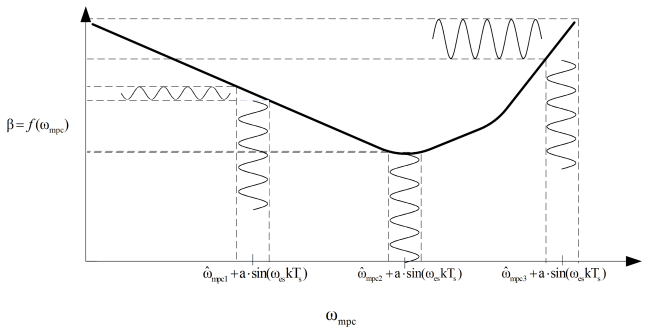

Fig. 2. Response of β to sinusoidal perturbation around various $\hat{\omega}_{mpc}$.

(2010). In this study, the dither signal is assumed to be a sine wave, although other types of dither can be used as well (Tan et al. (2008)). Figure 2 shows the response of a nonlinear function with a global minimum to such a sinusoidal perturbation signal around three different $\hat{\omega}_{mpc}$. At $\hat{\omega}_{mpc1}$, $\frac{\partial f(\hat{\omega}_{mpc1})}{\partial \omega_{mpc}} < 0$. As a result, the phase of the dither signal in the output of the nonlinear function is inverted. However, at $\hat{\omega}_{mpc3}$, $\frac{\partial f(\hat{\omega}_{mpc3})}{\partial \omega_{mpc}} > 0$, the phase of the dither signal component remains unchanged.

This figure also shows that perturbation around $\hat{\omega}_{mpc3}$ results in a larger amplification of the dither signal,compared to perturbation around $\hat{\omega}_{mpc1}$. Around $\hat{\omega}_{mpc2}$, which is close to the extremum ω^*_{mpc}, the dither signal is hardly visible in the output, because $|\frac{\partial f(\hat{\omega}_{mpc2})}{\partial \omega_{mpc}}| \approx 0$. Furthermore, in this figure, $|\frac{\partial f(\hat{\omega}_{mpc2})}{\partial \omega_{mpc}}| < |\frac{\partial f(\hat{\omega}_{mpc1})}{\partial \omega_{mpc}}| < |\frac{\partial f(\hat{\omega}_{mpc3})}{\partial \omega_{mpc}}|$. Intuitively, the dither signal component in the output can be regarded as an estimate of the local gradient around a certain closed-loop bandwidth $\hat{\omega}_{mpc}$.

The mathematical analysis of the ES algorithm is carried out by representing β in its first order Taylor expansion:

$$\beta = f(\hat{\omega}_{mpc}) + \frac{\partial f(\hat{\omega}_{mpc})}{\partial \omega_{mpc}} a \sin(\omega_{es} k) + O(a^2 \sin^2(\omega_{es} k)). \quad (4)$$

The first step in the algorithm is processing β by a discrete high-pass filter $H_{HPF}(z)$. The main function of this filter is to discard any DC component $\overline{\beta}$ that might be present in β, so that the tuning of $\hat{\omega}_{mpc}$ is based solely on variations in β and not on any DC offset. This filter is designed to only induce a negligible phase shift at the frequency $\hat{\omega}_{mpc}$ and its implementation will be treated later in Section 3.2. The filtered output p can be represented by

$$p(k) \simeq f(\hat{\omega}_{mpc}) - \overline{\beta} + a \frac{\partial f(\hat{\omega}_{mpc})}{\partial \omega_{mpc}} \sin(\omega_{es} k). \quad (5)$$

Next, the signal p is multiplied by the dither signal shape and its output is scaled by a factor $\frac{1}{a}$, which results in:

$$q(k) \simeq \frac{1}{a} \left(f(\hat{\omega}_{mpc}) - \overline{\beta} \right) \sin(\omega_{es} k) + \frac{\partial f(\hat{\omega}_{mpc})}{\partial \omega_{mpc}} \sin^2(\omega_{es} k). \quad (6)$$

The expression $\sin^2(\omega_{es} k)$ can be rewritten, using

$$\sin^2(\omega_{es} k) = \frac{1}{2} - \frac{1}{2} \cos(2\omega_{es} k). \quad (7)$$

Applying (7) to (6) results in

$$q(k) \simeq \frac{1}{2} \frac{\partial f(\hat{\omega}_{mpc})}{\partial \omega_{mpc}} + \frac{1}{a} \left(f(\hat{\omega}_{mpc}) - \overline{\beta} \right) \sin(\omega_{es} k) - \frac{1}{2} \frac{\partial f(\hat{\omega}_{mpc})}{\partial \omega_{mpc}} \cos(2\omega_{es} k). \quad (8)$$

Only the first term in Equation (8) is of interest. The low-pass filter $H_{LPF}(z)$ is used to discard all the harmonics of the dither signal frequency in q, resulting in the filtered signal

$$s(k) \simeq \frac{1}{2} \frac{\partial f(\hat{\omega}_{mpc})}{\partial \omega_{mpc}} \quad (9)$$

which is an approximation of the gradient of $f(\omega_{mpc})$. The last step is performed by a discrete-time integrator block that calculates a new $\hat{\omega}_{mpc}$ according to:

$$\hat{\omega}_{mpc}(k) = \hat{\omega}_{mpc}(k-1) - \gamma T_s s(k-1) \quad (10)$$

where γ represents the integrator gain and $T_s = \frac{1}{f_s}$ denotes the sampling time of the system. A negative value for s will cause the algorithm to increase $\hat{\omega}_{mpc}$, while a positive s will result in a decrease of $\hat{\omega}_{mpc}$. If s is close to zero, $\hat{\omega}_{mpc}$ is stable and in close vicinity of the extremum ω^*_{mpc}.

3.2 Filter Design

Low-Pass Filter $H_{LPF}(z)$ As mentioned earlier, the key function of the low-pass filter is to obtain the DC component of (8). The fact that the integral of a sinusoid over any integer multiple of its period is equal to zero can be used to design this low-pass filter:

$$\int_0^{T_{es}} \sin(m \cdot \omega_{es} t) dt = 0 \text{ for } m \in \mathbb{N}^+. \quad (11)$$

A method to obtain the low-pass effect is to calculate the average of q over a full period of the dither signal, $T_{es} = \frac{2\pi}{\omega_{es}}$. The discrete-time representation of this filter is

$$s(k) = \sum_{n=k-M+1}^{k} q(n) \approx \frac{1}{2} \frac{\partial f(\hat{\omega}_{mpc})}{\partial \omega_{mpc}} \quad (12)$$

$$M = \left\lfloor \frac{T_{es}}{T_s} + 0.5 \right\rfloor \quad (13)$$

In practice $\frac{T_{es}}{T_s}$ might be a rational number, in that case it needs to be rounded to the nearest integer to ensure that $M \in \mathbb{N}$, which is shown in (13). Furthermore, it is assumed that $T_{es} >> T_s$, which is a valid assumption since ω_{es} must be slower than the plant dynamics. This filter has the following Z transform:

$$H_{LPF}(z) = \sum_{n=0}^{M-1} \frac{1}{M} z^{-n} \quad (14)$$

The structure of (14) is that of a low-pass FIR filter. The FIR filter calculates a sum over M samples at every sampling instant. The structure of (14) can be rewritten into the equivalent IIR structure, which is much more efficient for large M and represented by:

$$H_{LPF}(z) = \frac{1}{M} \frac{1 - z^{-M}}{1 - z^{-1}} \quad (15)$$

and thus

$$s(k) = s(k-1) + \frac{1}{M}q(k) - \frac{1}{M}q(k-M). \quad (16)$$

Besides removing harmonics of the dither frequency, the low-pass filter has an extra function. It performs the averaging needed to obtain a good estimate of β based on samples of $||\boldsymbol{\epsilon}(k)||^2$.

High Pass Filter $H_{HPF}(z)$ The objective of the high-pass filter is to remove any DC value present in (4), while introducing little phase distortion at the frequency of interest ω_{es}. A solution that meets both requirements is to use the low-pass filter structure of (15) to build a new filter as follows:

$$\begin{aligned} H_{HPF}(z) &= 1 - H_{LPF}(z) = 1 - \frac{1}{M}\frac{1-z^M}{1-z^{-1}} \\ &= \frac{\frac{M-1}{M} - z^{-1} + \frac{1}{M}z^{-M}}{1-z^{-1}} \end{aligned} \quad (17)$$

and thus

$$p(k) = \quad p(k-1) + \frac{M-1}{M}\beta(k) - \beta(k-1) + \frac{1}{M}\beta(k-M). \quad (18)$$

Since the the low-pass filter averages the input over T_{es}, its output is mainly DC. Subtracting this DC value will introduce no phase distortion at the dither frequency component if $\frac{T_{es}}{T_s} = M$ since in that case, the low-pass filter averages exactly a full period of the dither signal. If $\frac{T_{es}}{T_s} > M$, the averaging misses at most half a sample; otherwise, if $\frac{T_{es}}{T_s} < M$, the averaging redundantly covers at most half a sample:

$$\left|\frac{T_{es}}{T_s} - M\right| = \left|\frac{T_{es}}{T_s} - \left\lfloor\frac{T_{es}}{T_s} + 0.5\right\rfloor\right| \leq 0.5 \quad (19)$$

This introduces a phase distortion at the dither frequency component of p. The significance of this error can be reflected in the following value:

$$\epsilon_m = \frac{\left|\frac{T_{es}}{T_s} - M\right|}{\frac{T_{es}}{T_s}} \leq \frac{T_s}{2T_{es}} \quad (20)$$

Equation (20) shows that when T_{es} goes to infinity, the error in the averaging, denoted by ϵ_m, approaches zero. If T_{es} goes to infinity, so does M, according to (13). Despite not being shown here, it is verified that a bigger M also results in a smaller worst-case phase distortion.
Equations (8) and (9) suggest that the high-pass filter is the least important component in the ES scheme of Figure 1, since the combination of multiplication with the dither signal, followed by the low-pass filter, eliminates the remaining DC term $\left(f(\hat{\omega}_{mpc}) - \overline{\beta}\right)$ in (5).

Filter Transients Both the high-pass and the low-pass filters in the ES scheme of Figure 1 introduce transients. During these transients the gradient estimate s given in (9) is incorrect. Therefore, the integrator should only be enabled after these transients have died out. The duration of the transient introduced by each filter equals one period of the dither signal T_{es}, according to (14), (15) and (17). Enabling both filters requires $2T_{es}$ for the transient effects to disappear. When only the low-pass filter is active, this time is halved to T_{es}.

3.3 Parameter Selection

This subsection discusses the influence of the different parameters of the ES scheme in Figure 1. A thorough stability analysis is found in Krstić and Wang (2000) and Tan et al. (2006).

Dither Frequency ω_{es} Selecting ω_{es} is a trade-off between speed of convergence and precision. The dither signal should vary slowly enough for the plant to settle and thus preventing the plant dynamics from interfering with the peak seeking scheme (Krstić and Wang (2000)). On the other hand, increasing the dither frequency allows the integrator gain to be increased proportionally, while retaining the same domain of attraction (Tan et al. (2006)). As a rule of thumb, the dither frequency should be slower than the open-loop dynamics of the plant to obtain a useful signal to noise ratio (SNR) at the input of the ES scheme. It is worth mentioning that although the exact bandwidth of the plant is unknown, the plant model inside the MPC system can be used to determine a suitable value for ω_{es}.

Dither Signal Gain a The amplitude of the dither signal is a trade-off between accuracy and precision. The influence on accuracy can be deduced from (4), since it shows that by increasing a, the approximation error $O(a^2\sin^2(\omega_{es}k))$ also rises. A larger a will thus result in a larger offset of $\hat{\omega}_{mpc}$ with respect to ω^*_{mpc} (Tan et al. (2010)). On the other hand, a smaller a leads to a reduction in precision, as the decrease in the amplitude of the modulated gradient in (4) brings about a deterioration of the SNR at the input of the ES scheme.

Integrator Gain γ The value of the integrator gain is a trade-off between speed of convergence, precision and stability. A higher gain results in faster convergence, but the influence of any noise present in the output of the low-pass filter becomes more dominant. Furthermore, there is an upper limit on the integrator gain with respect to the stability of the adaptation loop. Increasing the integrator beyond this value will render the adaptation unstable. The exact value of this critical value depends on the second derivative, $\frac{\partial^2 f(\hat{\omega}_{mpc})}{\partial \omega^2_{mpc}}$, which is unknown (Moase et al. (2010)).

ES Dither Signal Shape In Tan et al. (2008), it is stated that when using the same ES parameters, a square wave dither signal results in faster convergence, compared to a sine wave. This is mainly because in that case, the factor $\frac{1}{2}$ in (9) vanishes. Furthermore, the paper states that the domain of attraction and accuracy do not change if both the integrator gain and dither amplitude approach zero. In addition, since a small value of γ slows down the convergence of the adaptation loop, which is not desired, the effect of using square wave dither when γ is close to its critical value has to be investigated.

4. BOTTOM LAYER: FROM CLOSED-LOOP BANDWIDTH TO WEIGHTING MATRICES

Based on the engineering rules of choosing the horizons (Garriga and Soroush (2010)), in this work, the prediction horizon is fixed so as to cover the main dynamics of the system, and the control horizon is chosen according to the computational capacity of the system. From a desired closed-loop bandwidth ω_{mpc}, several methods can be used to find the tuning parameters $\mathbf{Q}$ and $\mathbf{R}$ of MPC. Rowe and Maciejowski (2000b), Shah and Engell (2011) and Tran et al. (2014) aim at matching the MPC to a desired H_∞

controller while Lee and Yu (1994) tunes the Kalman filter and disturbance model to obtain the desired bandwidth.

Tran et al. (2012) and Özkan et al. (2012) show that the weighting factors of the input energy and output energy are correlated. Fixing the input weights and increasing the output weights, or fixing the output weights and reducing the input weights, both raise the closed-loop bandwidth. Hence, in this work, the input weight is used to manipulate the closed-loop bandwidth: Weighting matrix $\mathbf{Q} = \mathbf{I}$ and $\mathbf{R} = \rho\mathbf{I}$, where $\mathbf{I}$ is the identity matrix and ρ is a scalar. In that case, the closed-loop bandwidth is solely determined by the selection of the scalar ρ.

When the constraints are active, MPC can be considered to be a linear time-invariant controller. Therefore, it is possible to derive the sensitivity and complementary sensitivity functions of an MPC as in a linear controller. In an $n \times n$ system, the sensitivity function has n cut-off frequencies corresponding to n directions of the system. To find the ρ that corresponds to the desired closed-loop bandwidth ω_{mpc}, the following optimisation problem is solved:

$$\min_{\rho} ||\omega_{mpc} - \omega_{cs}(\sigma_1(\rho))||^2 \tag{21}$$

where $\omega_{cs}(\sigma_1(\rho))$ is the crossover frequency of the minimum singular value of the sensitivity function. Tran et al. (2012) shows that this problem is convex.

5. EXAMPLE: DISTILLATION COLUMN MODEL

A model of a binary distillation column is used to analyse the behaviour of ES in combination with frequency-domain tuning of an MPC. The characteristics of this model are:

- The plant output $\mathbf{y}_p(k)$ consists of two variables, namely the top and bottom compositions of the distillation column.
- The output reference $\mathbf{r}(k)$ is set to obtain a top composition of 0.95 [mole fraction] and a bottom composition of 0.05 [mole fraction].
- The measurement noise $\mathbf{n}(k)$ is modelled by white noise with power $\sigma_n = 0.01$.
- The manipulated variables are liquid flow and vapour flow.
- The feed rate and feed composition are unmeasured disturbances and modelled by white noise filtered by a second-order low-pass filter with cutoff frequency of 0.01 π [rad/min]. The feed-rate variance is set to 64 [Kmol/min] and the feed composition variance is set to 0.0025 [mole fraction]
- An estimate of the open-loop bandwidth of the model used in the MPC is 0.001 [rad/min].
- The closed-loop bandwidth lies in the interval $[0; 0.1]$ [rad/min], based on the analysis of the sensitivity functions, so the ES algorithm remains within these boundaries.
- The sampling time T_s is 5 minutes.

Specific parameters of the distillation column are given in Table 1.

The auto-tuning method is applied to the column with two different starting closed-loop bandwidths: 0.02 rad/min and 0.08 rad/min. The ES runs in the top layer of the auto-tuning to find the optimal closed-loop bandwidth and optimisation problem (21) is solved to find the weighting factors in the bottom layer while the column is in closed loop.

Table 1. Details of distillation column model

Description	Value	Units
Number of trays	110	
Feed location	39	
Relative volatility	1.35	
Feed rate	215	[Kmol/min]
Feed liquid fraction	1	
Feed composition	0.65	[mole fraction]
Molar holdup	30	[Kmol]

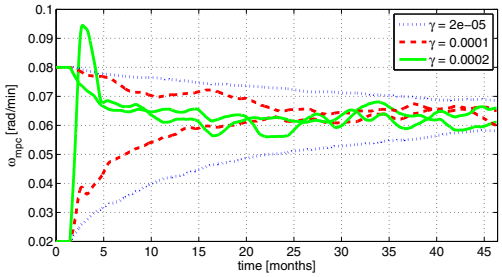

Fig. 3. Simulation results of MPC tuning. $\omega_{es} = 0.0001$, starting points: $\hat{\omega}_{mpc} = 0.02$ and $\hat{\omega}_{mpc} = 0.08$.

In these simulations, the dither signal is a sine wave of amplitude $a = 0.005$, the low-pass filter is enabled whereas the high-pass filter is disabled, since this results in halving the transient duration as discussed in Section 3. Figure 3 shows the convergence of the closed-loop bandwidth for $\omega_{es} = 0.0001$ [rad/min], while Figure 4 shows the results for $\omega_{es} = 0.0002$ [rad/min] with different values of the integrator gain γ. The x-axis is the time of the simulation and the y-axis is the closed-loop bandwidth.
Both Figures 3 and 4 show that increasing γ results in faster convergence, but the estimate of the extremum ω^*_{mpc} becomes more noisy. The solid line in Figure 3 shows that starting from $\hat{\omega}_{mpc} = 0.02$ [rad/min] with $\gamma = 0.0002$, there appears a significant overshoot in the adaptation loop. However, starting at $\hat{\omega}_{mpc} = 0.08$ [rad/min], under the same conditions, does not produce that overshoot. This stems from the fact that the second derivative is different for two different starting points. This behaviour is a known weakness of the classic ES algorithm. In Moase et al. (2010), the ES scheme is complemented with a compensator that adapts γ and a, based on local estimates of the second derivative. This modification might increase both the speed of convergence and stability of the ES scheme used in this paper, but this is still under investigation.
Figures 3 and 4 show that by increasing ω_{es}, the SNR increases, but the adaption loop is stable for higher integrator gains, which results in faster convergence. These results agree with the theory discussed in Section 3. The increase in SNR is a result of two factors, namely the low-pass character of the plant, which results in more damping of the dither signal at higher frequencies, and the length of the low-pass filter, which becomes smaller for higher dither frequencies.

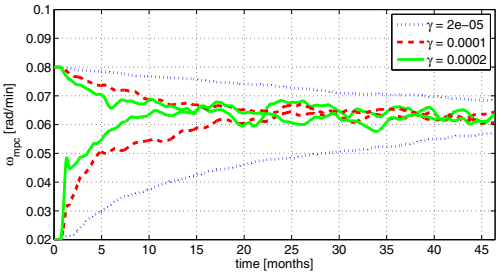

Fig. 4. Simulation results of MPC tuning. $\omega_{es} = 0.0002$, starting points: $\hat{\omega}_{mpc} = 0.02$ and $\hat{\omega}_{mpc} = 0.08$.

6. CONCLUSION AND RECOMMENDATIONS

This paper presented a way to automatically tune the closed-loop bandwidth of an MPC system by using ES. The ES algorithm has been implemented and tested on a model of a binary distillation column. ES indeed enables the MPC system to track its optimal closed-loop bandwidth.

However, the main issues of ES are speed of convergence and stability. The dynamics of the distillation column limit the speed at which the ES algorithm converges. The selection of ES parameters also affects the convergence of the algorithm. In the future, the use of a Newton-like adaptation law in the the classic ES scheme based on Moase et al. (2010) can be considered in choosing the parameters of ES.

Other filter designs might result in better performance. Further research could look into the benefits of using a high-pass filter although the analysis of Section 3 suggests that the high-pass filter is the least critical component. In addition, reducing the duration of the filter transients could improve the speed of convergence. The use of a square-wave dither signal will also be investigated.

REFERENCES

Al-Ghazzawi, A., E., E.A., Nouh, A., and Zafiriou, E. (2001). On-line tuning strategy for model pre-dictive controllers. *Journal of Process Control*, 11(3), 265–284.

Garriga, J.L. and Soroush, M. (2010). Model predictive control tuning methods: A review. *Ind. Eng. Chem. Res.*, 49(8), 35053515.

Han, K., Zhao, J., and Qian, J. (2006). A novel robust tuning strategy for model predictive control. In *The Sixth World Congress on Intelligent Control and Automation*. Dalian, China.

Huusom, J.K., Poulsen, N.K., Jørgensen, S.B., and Jørgensen, J.B. (2012). Tuning siso offset-free model predictive control based on arx models. *Journal of Process Control*, 22(10), 1997–2007.

Killingsworth, N.J. and Krstić, M. (2006). Pid tuning using extremum seeking: online, model-free performance optimization. *IEEE Control Systems Magazine*, 26(1), 70–79.

Krstić, M. and Wang, H. (2000). Stability of extremum seeking feedback for general nonlinear dynamic systems. *Automatica*, 36(1), 595–601.

Lee, J.H. and Yu, Z.H. (1994). Tuning of model predictive controllers for robust performance. *Computers and Chemical Engineering*, 18(1), 15–37.

Lee, J.V.D., Svrcek, W., and Young, B. (2008). A tuning algorithm for model predictive controllers based on genetic algorithms and fuzzy decision making. *ISA Transactions*, 47(1), 53–59.

Levya, R., Artillan, P., Cabal, C., Estibals, B., and Alonso, C. (2011). Dynamic performance of maximum power point tracking circuits using sinusoidal extremum seeking control for photovoltaic generation. *International Journal of Electronics*, 98(4), 529–542.

Moase, W.H., Manzie, C., and Brear, M.J. (2010). Newton-like extremum-seeking for the control of thermoacoustic instability. *IEEE Transactions on Automatic Control*, 55(9), 2094–2105.

Olesen, D.H., Huusom, J.K., and Jørgensen, J.B. (2013). A tuning procedure for arx-based mpc of multivariate processes. In *Proceedings of the 2013 American Control Conference*. Washington, DC, USA.

Özkan, L., Meijs, J., and Backx, A. (2012). A frequency domain approach for mpc tuning. In *Proceedings of the 11^{th} International Symposium on Process Systems Engineering*. Singapore.

Rowe, C. and Maciejowski, J. (2000a). Robust finite horizon mpc without terminal constraints. In *Proceedings of the 39^{th} IEEE Conference on Decision and Control*. Sydney, Australia.

Rowe, C. and Maciejowski, J. (2000b). Tuning mpc using h_∞ loop shaping. In *Proceedings of the American Control Conference*. Chicago, IL, USA.

Shah, G. and Engell, S. (2011). Tuning mpc for desired closed-loop performance for mimo systems. In *Proceedings of the 2011 American Control Conference*. San Francisco, CA, USA.

Stanković, M.S. and Stipanović, D.M. (2010). Extremum seeking under stochastic noise and applications to mobile sensors. *Automatica*, 46(8), 1242–1251.

Suzuki, R., Kawai, F., Ito, H., Nakazawa, C., Fukuyama, Y., and Aiyoshi, E. (2007). Automatic tuning of model predictive control using particle swarm optimization. In *Proceedings of the 2007 IEEE Swarm Intelligence Symposium*. Honolulu, HI, USA.

Tan, Y., Moase, W.H., Manzie, C., Nešić, D., and Mareels, I. (2010). Extremum seeking from 1922 to 2010. In *Proceedings of the 29th Chinese Control Conference*. Beijing, China.

Tan, Y., Nešić, D., and Mareels, I. (2006). On non-local stability properties of extremum seeking control. *Automatica*, 44(5), 889–903.

Tan, Y., Nešić, D., and Mareels, I. (2008). On the choice of dither in extremum seeking systems: A case study. *Automatica*, 44(5), 1446–1450.

Tran, Q.N., Özkan, L., and Backx, A. (2012). Mpc tuning based on impact of modeling uncertainty on closed-loop performance. In *Proceedings of the 2012 AICHE annual meeting*. Pittsburg, PA, USA.

Tran, Q.N., Özkan, L., and Backx, A. (2013). Auto-tuning of mpc based on frequency domain specifications. In *Proceedings of the 32nd Benelux Meeting on Systems and Control*. Houffalize, Belgium.

Tran, Q.N., Octaviano, R., Özkan, L., and Backx, A. (2014). Generalized predictive control tuning by controller matching. In *Proceedings of the 2014 American Control Conference*. Portland, OR, USA.