Graphical Abstract

Self-tunable approximated explicit MPC: Heat exchanger implementation

Lenka Galčíková, Juraj Oravec, Petronela Belková

Highlights

Self-tunable approximated explicit MPC: Heat exchanger implementation

Lenka Galčíková, Juraj Oravec, Petronela Belková

- Research highlight 1
- Research highlight 2

Self-tunable approximated explicit MPC: Heat exchanger implementation

Lenka Galčíková, Juraj Oravec, Petronela Belková

[a] *Slovak University of Technology in Bratislava, Faculty of Chemical and Food Technology, Institute of Information Engineering, Automation, and Mathematics,, Radlinského 9, Bratislava, 81237, Slovakia*

Abstract

max 250 words!

Keywords:

1. Introduction

(skor intro?)The benefit in form of lower computational complexity in the control phase comes hand in hand with a drawback. The size of the parametric solution may be so large, that it becomes impractical to utilize for two reasons: (i) memory footprint is higher than the available memory size of the control unit, (ii) the computational time associated with finding the optimal control action is higher than the available time period for control action implementation. Although this control strategy has its challenges, it is still very beneficial for practical usage for its benefits.

2. Preliminaries

In this section, the theoretical background necessary for this work is summarized. First, the explicit model predictive control is recalled. Next, the tunable technique of the approximated explicit model predictive control is introduced. Last but not least, the ideas of self-tunable technique of the approximated explicit MPC are presented.

2.1. Explicit model predictive control

Explicit model predictive control [1] represents a parametric solution of the model predictive control which makes it suitable for online implementation. As the explicit solution is available, it is not necessary to solve the optimization problem in every control step. As this work deals with practical implementation, let us consider the optimization problem in the following form:

$$\min_{u_0,u_1,\ldots,u_{N-1}} \quad \sum_{k=0}^{N-1} \left((y_k - y_{\mathrm{ref}})^{\intercal} Q_{\mathrm{y}} (y_k - y_{\mathrm{ref}}) + u_k^{\intercal} R u_k + x_{\mathrm{I},k}^{\intercal} Q_{\mathrm{x}} x_{\mathrm{I},k} \right) \tag{1a}$$

$$\text{s.t.:} \quad \widetilde{x}_{k+1} = \widetilde{A}\,\widetilde{x}_k + \widetilde{B}\,u_k, \tag{1b}$$

$$y_k = \widetilde{C}\,\widetilde{x}_k, \tag{1c}$$

$$u_k \in \mathcal{U}, \tag{1d}$$

$$y_k \in \mathcal{Y}, \tag{1e}$$

$$\widetilde{x}_0 = \theta, \tag{1f}$$

$$k = 0, 1, \ldots, N-1, \tag{1g}$$

where k denotes the step of the prediction horizon N. The parameter $\theta \in \Theta$ in Eq. (1f) represents the initial condition of the optimization problem for which it is parametrically precomputed. Variables $\widetilde{x} \in \mathbb{R}^{n_{\widetilde{\mathrm{x}}}}$, $u \in \mathbb{R}^{n_{\mathrm{u}}}$, $y \in \mathbb{R}^{n_{\mathrm{y}}}$ are vectors of corresponding extended system states, control inputs and system outputs, respectively. The prediction model in (1b)–(1c) has the form of extended linear time-invariant (LTI) system for given extended state matrix $\widetilde{A} \in \mathbb{R}^{n_{\widetilde{\mathrm{x}}} \times n_{\widetilde{\mathrm{x}}}}$, extended input matrix $\widetilde{B} \in \mathbb{R}^{n_{\widetilde{\mathrm{x}}} \times n_{\mathrm{u}}}$ and extended output matrix $\widetilde{C} \in \mathbb{R}^{n_{\mathrm{y}} \times n_{\widetilde{\mathrm{x}}}}$. The sets $\mathcal{U} \subseteq \mathbb{R}^{n_{\mathrm{u}}}$, $\mathcal{Y} \subseteq \mathbb{R}^{n_{\mathrm{y}}}$ are sets of physical constraints on inputs and outputs, respectively. The penalty matrix $Q_{\mathrm{y}} \in \mathbb{R}^{n_{\mathrm{y}} \times n_{\mathrm{y}}}$ penalizes the squared control error, i.e., the deviation between the output and output reference value y_{ref}. The matrix $R \in \mathbb{R}^{n_{\mathrm{u}} \times n_{\mathrm{u}}}$ penalizes the squarred value of control inputs. To obtain the offset-free control results, the built-in integrator was included in the state-space model, see e.g. [2]. The value of integrator is also penalized in the cost function with the penalty matrix $Q_{\mathrm{x}} \in \mathbb{R}^{n_{\mathrm{y}} \times n_{\mathrm{y}}}$.

The model extended with the built-in integrator in (1b)–(1c) can be rewritten as:

$$\widetilde{x}_{k+1} = \begin{bmatrix} x_{k+1} \\ x_{\mathrm{I},k+1} \end{bmatrix} = \begin{bmatrix} A & 0 \\ -T_{\mathrm{s}} C & I \end{bmatrix} \begin{bmatrix} x_k \\ x_{\mathrm{I},k} \end{bmatrix} + \begin{bmatrix} B \\ I \end{bmatrix} u_k, \tag{2a}$$

$$y_k = \begin{bmatrix} C & 0 \end{bmatrix} \begin{bmatrix} x_k \\ x_{\mathrm{I},k} \end{bmatrix}, \tag{2b}$$

where $x_{\mathrm{I}} \in \mathbb{R}^{n_{\mathrm{y}}}$ is the integral of the control error, T_{s} denotes the sampling time and matrices A, B, C are well-known linear time-invariant state-space matrices that form the extended model. Thanks to this extension and penalization in the cost function in Eq. (1a), not only the control error are penalized, but also its integral, which leads to similar results as incorporating integral part in PID controller.

Solving the MPC quadratic optimization problem in Eq. (1) parametrically leads to explicit solution in the form of piece-wise affine control law consisting of r critical regions:

$$u(\theta) = \begin{cases} F_1\,\theta + g_1 & \text{if} \quad \theta \in \mathcal{R}_1, \\ F_2\,\theta + g_2 & \text{else if} \quad \theta \in \mathcal{R}_2, \\ \quad\vdots & \\ F_r\,\theta + g_r & \text{else if} \quad \theta \in \mathcal{R}_r, \end{cases} \tag{3}$$

where F_i and g_i respectively are the slope and affine section of the corresponding control law. The piece-wise affine function from Eq. (3) is stored and used in the online, i.e., control phase. Based on identifying the critical region where the parameter θ belongs, the optimal control input is calculated based on the associated control law.

Note, many other formulations of the optimization problems are elaborated, mainly in the terms of cost functions. Also, the incremental (velocity) formulation of the state-space model is common, but leads to further extendsion of the vector of parameters θ and therefore the explicit MPC complexity increases. Another option for offset-free tracking is also disturbance observer. For such an overview see e.g. ...TODO. Klauco reference governors? minimovka zdroje + dizertacka str.33

2.2. Tunable explicit model predictive control

The aggressivity of the controller and whole nature of the control can be influenced by tuning the penalty matrices in the optimization problem (1). When the optimization problem is precomputed offline to obtain the parametric solution, it is not possible to tune the controller afterwards. As the operation conditions and demands on controller setup may differ throughout control, the ability to adjust the controller aggressivity can be very beneficial.

The idea of approximated tunable explicit MPC comes from the work [3], where the control action is calculated based on linear interpolation between two boundary control actions. These control actions result from evaluating two boundary explicit MPCs. The boundary explicit controllers have the same structure and setup, except one of the penalty matrices – the tuned one.

The boundary matrices are diagonal square matrices such that $\lambda_{i,\mathrm{L}} \leq \lambda_{i,\mathrm{U}}$, $\forall i = 1, \ldots, s$, where λ denotes the vector of penalty matrix eigenvalues, L and U denote the lower and upper boundary setup respectively, and s is the rank of the tuned penalty matrix.

When considering the penalty matrices from cost function (1a), the penalty matrices are scaled in the following way:

$$R = (1-\rho)R_{\mathrm{L}} + \rho R_{\mathrm{U}}, \tag{4a}$$

$$Q_{\mathrm{x}} = (1-\rho)Q_{\mathrm{x,L}} + \rho Q_{\mathrm{x,U}}, \tag{4b}$$

$$Q_{\mathrm{y}} = (1-\rho)Q_{\mathrm{y,L}} + \rho Q_{\mathrm{y,U}}, \tag{4c}$$

where ρ represents the scaling parameter and $0 \leq \rho \leq 1$ holds. Based on (4), it is possible to choose online any controller setup from the lower to the upper boundary of the tuned matrix. Note, it is practical to tune only one matrix, i.e., to store only two controllers corresponding to the boundary matrices. To determine which penalty matrix should be tuned, it is suggested to examine the control results subject to the specific system and to try systematic tuning of all the penalty matrices.

When the scaling parameter ρ is determined based on the current control conditions, the optimal control action is evaluated using the two optimal controllers. Based on the boundary control actions, the interpolated, i.e., tuned control action is calculated:

$$u = (1-\rho)u_{\mathrm{L}} + \rho u_{\mathrm{U}}, \tag{5a}$$

where u_{L} and u_{U} denote the optimal control actions from the lower and upper boundary controller respectively. The tuning of the controller online comes with a cost of storing and evaluating two explicit controllers. Nevertheless, the ability to tune the controller may be more important in many practical applications.

Note, the concept of explicit MPC tuning can be applied also to other MPC formulations, based on the current specific needs. In this paper, the penalty matrices of cost function (1a) are considered, as it is necessary to satisfy offset-free reference tracking.

Remark 2.1. *If stability guarantee and recursive feasibility is required, it is suggested to follow the instructions from [4]. In order to satisfy the requirements, the study introduces a procedure of choosing the terminal penalty and terminal set in the two boundary controllers.*

2.3. Self-tunable explicit model predictive control

The advantage of tunable controller brings a question of how to design the logic of tuning the scaling parameter ρ. In this section, the idea of online self-tuning is summarized [5]. The concept of self-tuning provides the possibility to adjust the aggressiveness of the controller without the necessity to intervene and tune the penalty matrices during control.

The need for real-time controller tuning often arises from tracking a time-varying reference. The work [5] focuses on adjusting the penalty matrix when the reference value is changed. The further the reference value is from the steady state, the more aggressive controller is tuned. The idea behind the suggested scaling liest in compensation of the nonlinear behavior of the system.

The procedure of tuning the controller is based on evaluating the difference between the reference and the steady state, and using this deviation to scale the value of control action. First, the maximal absolute value of reference which can be set during control is defined. This value can be determined based on the general knowledge of future reference values. Another suggestion is to set the maximal deviation $d_{\max}$ simply based on the constraints on system outputs:

$$d_{\max} = \max(|y_{\min}|, y_{\max}), \tag{6}$$

where $y_{\min}$ and $y_{\max}$ are respectively lower and upper bound on the output variable. Using the information about the maximal possible deviation $d_{\max}$, the scaling parameter ρ can be calculated as the ration between the current reference value and the maximal deviation:

$$\rho = \frac{|y_{\text{ref}}|}{d_{\max}}. \tag{7}$$

Based on Eq. (7), the property $0 \leq \rho \leq 1$ holds, as $|y_{\text{ref}}| \leq d_{\max}$. As a result, the parameter ρ represents a way how to normalize the deviation from steady state and is exploited to scale the penalty matrix or at the end of the day, the control action.

When considering tuning the control action based on Eq. (5a), a higher value of scaling parameter ρ leads to approaching the upper boundary controller and vice versa. When tuning e.g. the matrix Q_{y} penalizing the control error, a higher ratio ρ would lead to more aggressive control actions. When controlling to the reference values closer to the steady state, the parameter *rho* decreases and the control becomes sluggish.

Remark 2.2. *The parameter* $d_{\max}$ *is vector in general, as it depends on the size of system outputs. If* $d_{\max}$ *is scalar, the paramter* ρ *is scalar as well and can be directly utilized to scale the control action. If multiple outputs are controlled, it is suggested to calculate the scaling parameter based on the maximal ratio:*

$$\rho = \max\left(\frac{|y_{\text{ref}}|}{d_{\max}}\right). \tag{8}$$

Note, the relations (7) and (8) operate with the absolute value of the reference. It is not taken into account, whether the reference change was positive or negative, i.e., whether it changed upwards or downwards. As many plants have non-symmetric behavior, the positivity or negativity of the reference change could be considered in the controller tuning to increase the control performance.

3. Methodology

3.1. Self-tunable technique for systems with asymmetric behavior

TODO: rozdelit dva spominane pristpy? dat dva navody?

TODO: spomenut, ze to je vhodne celkovo aj pre klasicke online MPC

4. Results and discussion

In this section, the results of the proposed tuning method are presented. The plant on which the control was implemented is a laboratory liquid-to-liquid plate heat exchanger Armfield Process Plant Trainer PCT23 [6], see Fig. 1. The cold feed as well as heating medium are transported to the heat exchanger by two peristaltic pumps. The flow rate of the feed is constant, while the aim is to track the reference value of its temperature. Therefore, the controlled variable is the feed temperature T at the outlet of the heat exchanger and the manipulated variable is the voltage U corresponding to the power of the heating medium pump. The voltage is set in percentage, while it is constrained from 20% to 100%. As the heat exchange is a nonlinear and non-symmetric process [7], this heat exchanger represents a suitable candidate for the presented controller tuning strategy.

The matrices of the linear state-space model of the plant, discretized with sampling time $T_\mathrm{s} = 1\,\mathrm{s}$, are

$$A = \begin{bmatrix}0.839\end{bmatrix}, \tag{9a}$$
$$B = \begin{bmatrix}0.039\end{bmatrix}, \tag{9b}$$
$$C = \begin{bmatrix}1\end{bmatrix}, \tag{9c}$$
$$D = \begin{bmatrix}0\end{bmatrix}. \tag{9d}$$

The constraints are considered in the terms of manipulated variable, i.e.

$$-15\% \leq u \leq 65\%. \tag{10}$$

Note, the variable u represents the manipulated variable in the deviation form. The values of feed temperature and voltage of the heating medium pump corresponding to zero steady state are respectively $T^\mathrm{s} = 35$ °C and $U^\mathrm{s} = 35\%$.

The penalty matrices of the problem Eq. (1) were systematically tuned and the corresponding control was implemented on the laboratory heat exchanger for every MPC setup. The aim was to determine, which penalty matrix will be the tuned one. Based on observations, the most significant effect on the control trajectories had tuning the penalty matrix Q_y, while still preserving a satisfactory control performance, i.e., without steady-state control error and significant oscillations around the setpoint. The boundary values of matrix Q_y were tuned as $Q_\mathrm{y,L} = 100$ and $Q_\mathrm{y,U} = 1000$. The built-in integrator was penalized with $Q_\mathrm{x2} = 1$ and the manipulated variable with $R = 10$. The prediction horizon N was 20 steps long. The explicit model predictive controllers were constructed in MATLAB R2020b using the Multi-Parametric Toolbox 3 [8]. The controllers were implemented to track a time-varying reference. For the first 200 seconds, the system was in the steady state. After that the reference changed its value twice upwards and twice downwards. The reference changes also acquired different sizes in order to examine the proposed tuning method as it is dependent on the size of the reference step change. The control results for both control setups are compared in Fig. 2 for the controlled variable, and in Fig. 3 for the manipulated variable.

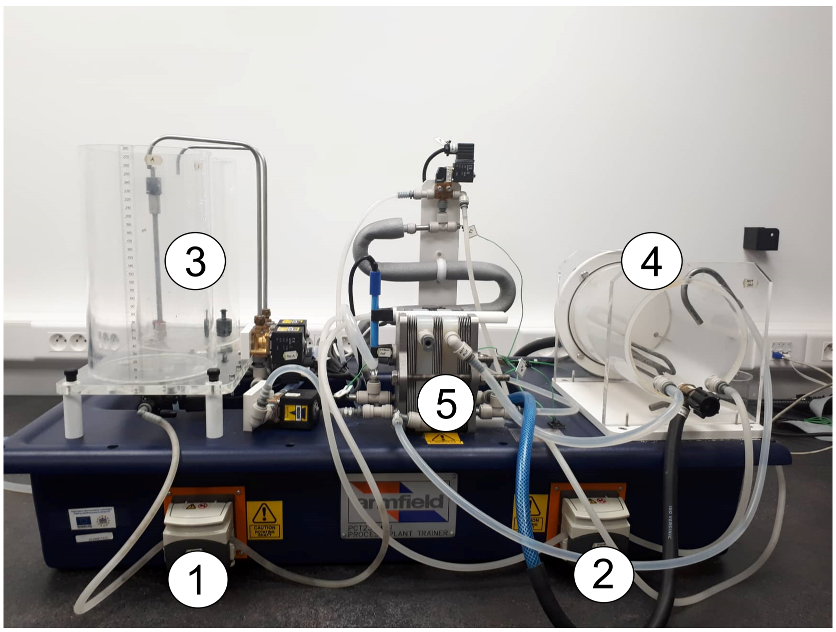

Figure 1: Laboratory heat exchanger Armfield Process Plant Trainer PCT23: feed pump (1), heating medium pump (2), feed tanks (3), heater for heating medium (4), heat exchanger (5).

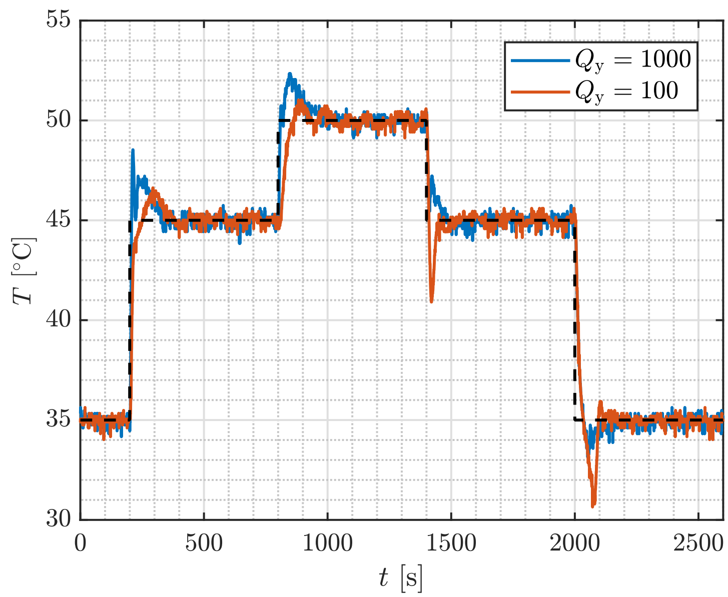

Figure 2: Controlled variable trajectory for two boundary controllers. The solid lines represent the controlled temperature and the dashed line represents the reference value.

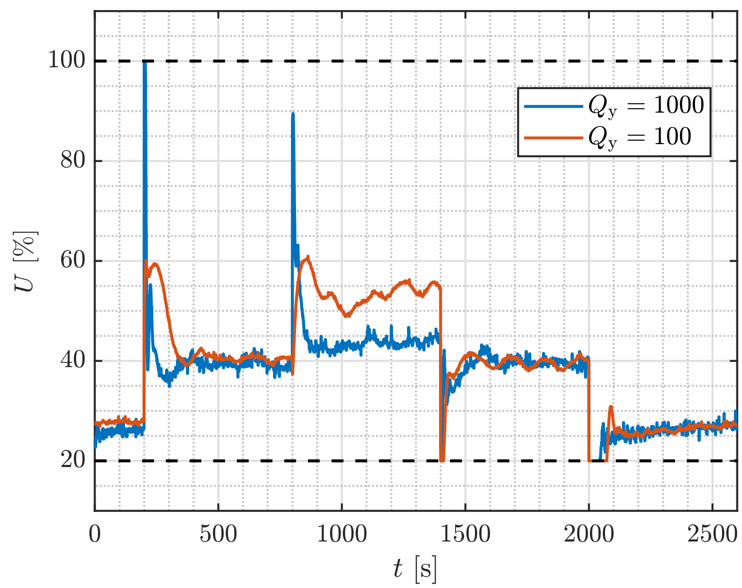

Figure 3: Manipulated variable trajectory for two boundary controllers. The solid lines represent the voltage and the dashed lines represent the constraints.

The trajectories in Fig. 2 show the non-symmetric nature of controlling the process of heat exchange mainly when observing the overshoots and undershoots. When applying the control associated with the lower bound $Q_{\mathrm{y,L}}$, significant undershoots are present when tracking the reference downwards, i.e., when the reference change is negative. On the contrary, when implementing the controller associated with $Q_{\mathrm{y,U}}$, the undershoots are negligible, but significant overshoots can be seen when tracking the reference upwards.

These observations form the base for the strategy of self-tuning the penalty matrix Q_{y}. The strategy follows the ideas summarized in Section 3.1.

TODO: dopisat strucne princip ladenia podla toho, ako bude spisana teoria v 3.1. odkazat sa na rovnice atd...

The control results of the selt-tunable technique compared to the boundary controllers can be seen in Fig. 4 for the controlled variable, and in Fig. 5 for the manipulated variable. It can be seen that the tuned controller combined the benefits of the two boundary controllers. The overshoots and undershoots were reduced, as in the first half of control the penalty matrix Q_{y} acquired value from the first half of the penalty interval. When tracking the reference with negative change, the penalty matrix acquired the values from the second half of the interval, i.e., closer to the upper bound $Q_{\mathrm{y,U}}$. The similarity with the boundary controllers can be seen also on the manipulated variable trajectory. Note, the constraints on the input variable were satisfied as they were scaled based on the boundary controllers which are constructed considering the contraints.

TODO: what about state constraints?

The control performance was also investigated quantitatively. The Table 1 summarizes the evaluated control performance criteria for the tuned and two boundary controllers. The control performance is evaluated for every reference step change respectively. The criteria are integral square error ISE, maximal overshoot/undershoot $\sigma_{\max}$, settling time t_{ϵ} for 5%-neighbourhood of the reference value, and the volume of the heating medium utilized for the corresponding control. To provide better readability of the results, the best (i.e. minimal) values are written in bold.

It can be seen in Table 1, that real-time tuning of the controller helped to improve two to three criteria when tracking each reference value. The cost for this is the energy associated with control. Although the integral square error, maximal overshoot/undershoot and settling time decreased, the volume of consumed heating medium did not. Nevertheless, the average deterioration in the terms of consumed heating medium is approximately

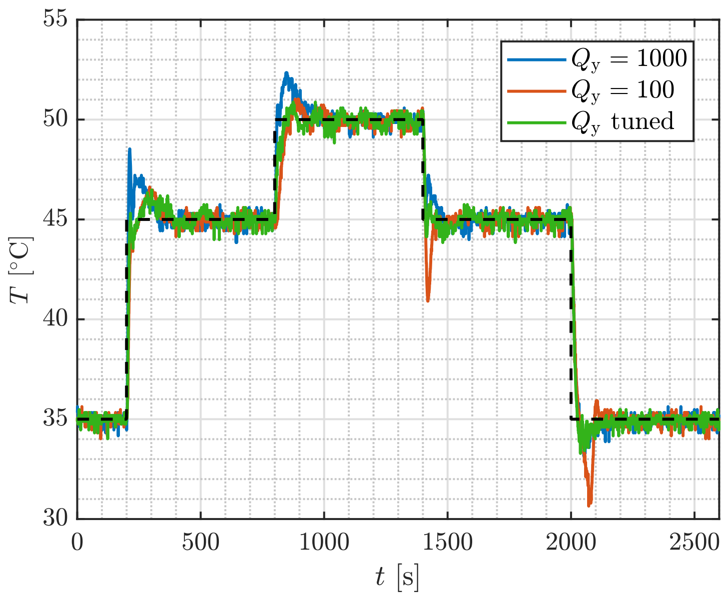

Figure 4: Controlled variable trajectory for two boundary controllers and the tuned one. The solid lines represent the controlled temperature and the dashed line represents the reference value.

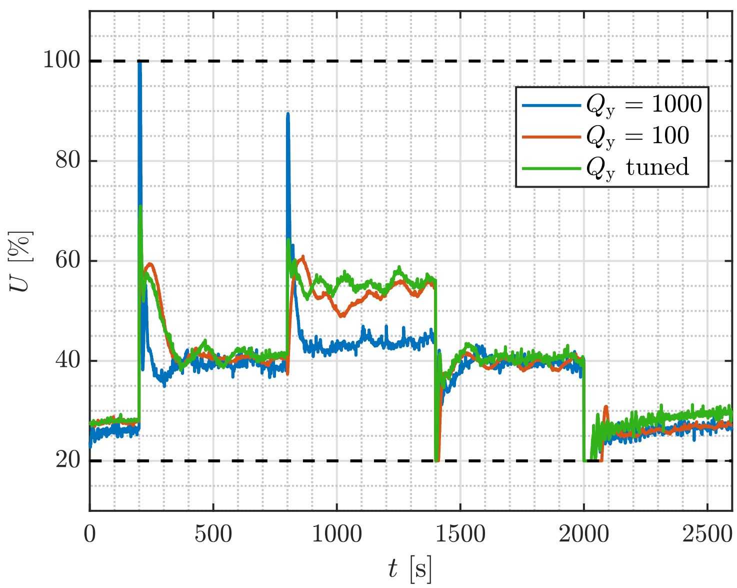

Figure 5: Manipulated variable trajectory for two boundary controllers and the tuned one. The solid lines represent the voltage and the dashed lines represent the constraints.

Table 1: Control performance criteria.

Reference step change	Q_y	ISE [°C^2 s]	σ_max [%]	t_ϵ [s]	V [l]
1	1000	714	33.50	16.5	**2.12**
	100	867	16.65	12.5	2.36
	tuned	**678**	**15.19**	**9.5**	2.38
2	1000	365	47.20	**5**	**2.49**
	100	606	23.25	26.5	3.19
	tuned	**248**	**19.13**	9.5	3.35
3	1000	245	**18.92**	**6.5**	**2.00**
	100	398	79.64	31	**2.00**
	tuned	**186**	24.59	**6.5**	2.10
4	1000	1024	18.43	22.5	0.94
	100	1402	41.87	90	**0.93**
	tuned	**967**	**16.49**	**18.5**	1.10

17%.

The improvement or deterioration in percentage of using the tunable controller relative to the second best setup is summarized in Table 2 for every reference step change separately. The negative numbers represent deterioration compared to the best controller setup in the corresponding reference tracking.

Table 2: Improvement or deterioration of the control performance in percentage.

Reference step change	ISE	σ_max	t_ϵ	V
1	5.04	8.77	24.00	-12.26
2	32.05	17.72	-90.00	-34.54
3	24.08	-29.97	0	-4.48
4	5.57	10.53	17.78	-18.28

It can be seen that implementing a tunable controller leads to improved control performance in many criteria. Utilizing a controller with a scalable aggressivity according to the operating conditions leads in general to higher accuracy, lower overshoot and faster achieving the reference value. Although the volume of the heating medium is not saved in this control scenario, the improved control performance may lead to other benefits. TODO: krajsie skomentovat vyhody zlepsenia kvality riadenia

5. Conclusion

References

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

[2] D. D. Ruscio, Model predictive control with integral action: A simple mpc algorithm, Modeling, Identification and Control 34 (3) (2013) 119–129.

[3] M. Klaučo, M. Kvasnica, Towards on-line tunable explicit mpc using interpolation, in: Preprints of the 6th IFAC Conference on Nonlinear Model Predictive Controle, IFAC, Madison, Wisconsin, USA, 2018.

[4] J. Oravec, M. Klaučo, Real-time tunable approximated explicit mpc, Automatica 142 (2022) 110315.

[5] L. Galčíková, M. Horváthová, J. Oravec, M. Bakošová, Self-tunable approximated explicit model predictive control of a heat exchanger, Chemical Engineering Transactions, 2022, Vol. 94 (94) (2022) 1015–1020.

[6] Introduction Manual, PCT23-MkII Process Plant Trainer, Armfield, 2007.

[7] B. G. Liptak, Instrument Engineers' Handbook, 4th Edition, Vol. 2: Process Control and Optimization, CRC Press, London, 2005.

[8] M. Herceg, M. Kvasnica, C. Jones, M. Morari, Multi-parametric toolbox 3.0, in: European Control Conference, 2013, pp. 502–510.