sciendo

Agile manoeuvres using model predictive control

Kristína Fedorová, Peter Bakaráč, Michal Kvasnica

Faculty of Chemical and Food Technology, Slovak University of Technology,
Radlinského 9, 812 37 Bratislava

Abstract: This paper shows how model predictive control (MPC) can be used to perform agile manoeuvres in a pendulum-on-a-cart system, which is an abstraction of many mechanical systems commonly used in the industry, such as cranes. Specifically, the problem of moving a cart on which a pendulum is mounted using a free joint is rapidly moved from one position to another one while mitigating the swings of the pendulum. To achieve this goal, an optimization-based MPC strategy is employed that selects the control moves while minimizing the chosen cost function and, simultaneously, enforcing constraint satisfaction. As the controlled system is nonlinear, two options are considered. The first one solves the nonlinear MPC problem in an approximate fashion using the so-called random shooting approach. The second method is based on the first one approximating the nonlinear system by a linear one, followed by applying convex MPC techniques. The performance of both strategies was compared by means of real-time experiments.

Keywords: nonlinear model predictive control, convex model predictive control, random shooting, linearization

Introduction

Chemical reactors, heat exchangers and many other processes in the chemical industry perform with nonlinear dynamics which makes their control a challenging task. Especially when the operation of such processes accentuate safety. Control strategy offering the most benefits is model predictive control (MPC) (Maciejowski, 2002) due to its ability to perform optimized control actions while state and input constraints are enforced. Therefore, MPC has become the most preferred and popular control approach in the process industry, as proven by the survey of Qin and Badgwell (2003). Many successful applications of the MPC strategy control chemical reactors (Oravec and Bakošová, 2012; Bakošová and Oravec, 2014; Oravec and Bakošová, 2015; Bakošová et al., 2013), heat exchangers (Oravec et al., 2016a, 2018), and distillation columns (Martin et al., 2013). Alternatively, MPC can also be used as a supervisory control layer that can be coupled with conventional PID controllers (Klaučo et al., 2017).

The principle of MPC is to solve an optimization problem based on a mathematical model of the controlled system. Additionally, future behavior of the system is predicted and control inputs are optimized over the whole prediction horizon. In case of a linear model of the system, the evaluation of control input can be fast and relatively easy. Limitations of the linear model stem in a narrow range of its validity and often insufficient description of all the distinctions of the real process. To tackle these limitations, a nonlinear model of the process can be used. Performing nonlinear MPC can lead to non-convex optimization problems that require increased computational power. In fact, obtaining such a solution is unattainable within one sampling period if the process is characterized by fast dynamics.

In recent years, reducing of computational complexity affiliated with non-convex optimization problems within nonlinear MPC (NMPC) has become a serious research field (Allgöwer and Zheng, 2012). Various methods intended for the solution of such optimization problems have been described (Wright and Nocedal, 1999; Čižniar et al., 2009; Kirkpatrick et al., 1983; Poli et al., 2007). Some of the methods are focused on searching for the minimum in a certain local area. Therefore, there is always a possibility of solution suboptimality. Searching for the global minimum can be performed in a stochastic or deterministic way. Simulated annealing is a stochastic method based on inspecting an enormous number of solutions and subsequent convergence to the lowest value of the cost function of the NMPC. On the other hand, the drawback of the deterministic methods lies in their applicability in small size problems.

One of the systems with fast and nonlinear dynamics is also a pendulum-on-a-cart. The problem of unwanted oscillations at the output occurs during the system control. The pendulum system represents a crane moving cargo. If this crane is in the vicinity of other objects, its oscillation is very undesirable as it can cause considerable damage. At the same time, the speed at which the crane transmits objects plays an important role. The faster the device runs, the more it can transfer and the more efficient it is. Higher speeds, however, cause larger oscillations to be mitigated. The compromise between the speed of operation and the oscillation amplitude is agile maneuvers which operate with the device the fastest way possible to avoid violation of the constraints given for the oscillation amplitude.

This paper focuses on the solution of this NMPC problem employing the random shooting approach (Dyer et al., 2014). Application of this method is spread in machine learning (Sahoo et al., 2018) or robotics (Piovesan and Tanner, 2009) fields. Elegance and simplicity of the method is based on the fast randomly generated control input sequences pursued by investigation of the constraints enforcing and evaluating the cost function value. The sequence with the lowest value of the performance index is considered as the solution. Due to the random selection, solution sub-optimality can occur, while increased number of random sequences reduces the extent of such sub-optimality. However, the solution is feasible and it guarantees safe operation of the manipulated system.

Control performance of NMPC and MPC based on the linear model were compared in this paper. Scenarios of the agile maneuvers control in the pendulum-on-a-cart system were the same for both control strategies.

Theoretical

We considered a control of a system in the discrete-time domain represented by the state-update equation

$$x(t+\Delta) = f(x(t),\ u(t)), \tag{1}$$

and the output equation

$$y(t) = g(x(t),\ u(t)), \tag{2}$$

where $x(t) \in \mathbb{R}^n$ represents the state vector at time t, $u(t) \in \mathbb{R}^m$ are the control commands, and $y(t) \in \mathbb{R}^p$ are the controlled outputs. Additionally, $x(t+\Delta)$ represents the successor state. Finally, $f{:}\mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^n$ and $g{:}\mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}^p$ are, in general, nonlinear functions.

The objective of model predictive control is to use system equations (1) and (2) to predict future evolution of the system and to optimize the control inputs to achieve optimal behavior while meeting all system constraints. Mathematically, the MPC problem can be stated as

$$J^*(x_0) = \min_{u_0,\dots,u_{N-1}} \sum_{k=0}^{N-1} \left(\|x_k - x_{\text{ref}}\| + \|u_k\| \right), \tag{3a}$$

$$\text{s.t. } x_{k+1} = f(x_k,\ u_k),\ k = 0,\ \dots,\ N-1, \tag{3b}$$

$$y_k = g(x_k,\ u_k),\ k = 0,\ \dots,\ N-1, \tag{3c}$$

$$x_k \in \mathcal{X},\ k = 0,\ \dots,\ N-1, \tag{3d}$$

$$u_k \in \mathcal{U},\ k = 0,\ \dots,\ N-1, \tag{3e}$$

$$y_k \in \mathcal{Y},\ k = 0,\ \dots,\ N-1, \tag{3f}$$

$$x_0 = x(t), \tag{3g}$$

where x_k, u_k, y_k are the state, input, and output predictions at the k-th step of the prediction horizon, respectively, $||\cdot||$ is any vector norm, and $\mathcal{X}, \mathcal{U}, \mathcal{Y}$ are state, input, and output constraint sets, respectively. Optimization is performed over a finite prediction horizon N and yields the sequence $\{u^*_0, \dots, u^*_{N-1}\}$ of control moves that are optimal for the given initial condition, $x(t)$ per (3g) and the given reference value, x_{ref}.

The trouble with optimal control inputs from (3) is that if functions $f(\cdot)$ and $g(\cdot)$ in (3b)–(3c) are nonlinear, or if the constraints sets are non-convex, (3) becomes a non-convex optimization problem that is difficult to solve to global optimality. In the subsequent sections, two principal ways allowing (3) to be solved quickly are reviewed.

The first method to solve a nonlinear MPC problem of (3) quickly is the so-called random shooting method, introduced in more details in Bakaráč and Kvasnica (2018). In principle, the method investigates a (possibly large) number of randomly generated control sequences $\{u_0, \dots, u_{N-1}\}$. For each random sample, the response of the nonlinear system, i.e., the sequences $\{x_1, \dots, x_N\}$ and $\{y_0, \dots, y_{N-1}\}$ are calculated by (1) and (2), respectively, as a forward simulation of the nonlinear system starting from $x_0 = x(t)$. Subsequently, the validity of constraints in (3d)–(3f) is checked and any infeasible trajectories are discarded. Among the feasible trajectories, the random shooting method selects the sequence that yields the lowest value of the cost function (3a).

The advantage of random shooting is that it can cope with arbitrary nonlinear dynamics in (1) and (2) since only forward simulations are required. Moreover, arbitrary non-convex constraint sets can be employed. Therefore, the procedure can be used to investigate a very large number of randomly generated input trajectories in a short time. However, the obvious disadvantage is that the resulting control actions are merely sub-optimal. Moreover, due to their random generation, they can exhibit significant chattering and oscillations, which might not be desirable from a practical point of view. However, the sub-optimality can be mitigated by resorting to a very large number of random sequences (e.g. one million).

Another approach is to linearize the nonlinear dynamics in (1)–(2) around some chosen operating point, e.g. x_S and u_S. Applying the truncated Taylor expansion yields the linear approximation:

$$x(t+\Delta) \approx \frac{\partial f(x,u)}{\partial x}|_{x_s,u_s}(x-x_s)+ \\ +\frac{\partial f(x,u)}{\partial u}|_{x_s,u_s}(u-u_s), \tag{4}$$

and

$$y(t) \approx \frac{\partial g(x,u)}{\partial x}|_{x_s,u_s}(x-x_s)+ \\ +\frac{\partial g(x,u)}{\partial u}|_{x_s,u_s}(u-u_s), \tag{5}$$

which can be written as

$$x(t+\Delta) = Ax(t) + Bu(t), \tag{6a}$$

$$y(t) = Cx(t) + Du(t), \tag{6b}$$

where matrices A, B, C, D result from the partial derivatives of $f(\cdot)$ and $g(\cdot)$.

The advantage of the linear prediction in (6) is that when it is used in (3b)–(3c), the MPC problem becomes a convex optimization problem, which can be easily solved using conventional optimization techniques, provided the constraint sets $\mathcal{X}$, $\mathcal{U}$, $\mathcal{Y}$ are convex sets. If they are not, their respective inner convex approximations can be derived e.g. by maximization of the inscribed ellipsoid volume, or by searching for the largest inscribed hyper-box. The latter approach is preferred as the constraints in (3d)–(3f) become linear and are easier to tackle compared to quadratic constraints arising from an ellipsoid inner approximation. If, moreover, the norms in (3a) are squared 2-norms, the MPC problem (3) becomes a convex quadratic program that can be solved e.g. by active-set or interior-point methods.

Experimental

In this work, model predictive control was applied to control a laboratory mechanical system consisting of a cart on which a pendulum is mounted using a free joint, shown in Fig. 1. Such a system represents a crane moving a cargo from one position to another one as quickly as possible while mitigating the swings of the cargo to prevent collision with obstacles. Dynamics of the pendulum is represented by four nonlinear state equations in the continuous-time domain:

$$\dot{x}_1 = x_2, \tag{7a}$$

$$\dot{x}_2 = \frac{3}{4L}u\cos(x_1) + \frac{3g}{4L}\sin(x_1) - bx_2, \tag{7b}$$

$$\dot{x}_3 = x_4, \tag{7c}$$

$$\dot{x}_4 = u, \tag{7d}$$

along with the output equation

$$y = L\sin(x_1) + x_3. \tag{8}$$

Here, x_1 represents the pendulum's angle, x_2 is the angular velocity of the pendulum, x_3 is the position of the cart, and x_4 is its velocity. The input of the system is u, the acceleration of the cart. The output from the system, y is the x-position of the pendulum's endpoint. Model parameters in (7) are L = 21 cm, g = 9,81 m s^{-2}, and b = 1. The control objective is to change the position of the end point of the pendulum in the shortest possible time. The trajectory of the movement has to be such that all states of the system enforce the state and input constraints:

$$-\infty < x_1 < \infty, \tag{9a}$$

$$-\infty < x_2 < \infty, \tag{9b}$$

$$-0.25 \le x_3 \le 0.25, \tag{9c}$$

$$-2.0 \le x_4 \le 2.0, \tag{9d}$$

$$-\infty < y \le 0.45, \tag{9e}$$

$$-4 \le u \le 4. \tag{9f}$$

Naturally, as the cart moves it induces oscillations of the pendulum, which are not desired as they may lead to the pendulum's endpoint colliding with obstacles. Therefore, the $y \le 0.45$ constraint must be met at all times to avoid such collisions.

To achieve a collision-free transition from one position to an another, model predictive control was applied. Specifically, the continuous-time system in (7) was first discretized using the sampling time Δ = 0.02 s and the forward Euler discretization. Specifically, given a continuous-time state equation $\dot{x} = f(x, u)$, its discrete-time version was achieved:

$$x(t+\Delta) = x(t) + \Delta f(x, y), \tag{10}$$

Alternatively, if the forward Euler discretization is imprecise, explicit Runge-Kutta discretization can also be applied. Specifically, the second-order explicit Runge-Kutta method yields the discrete-time system:

$$x(t+\Delta) = x(t) + \\ +\Delta f(x(t) + 0.5\Delta f(x(t), u(t)), u(t) + 0.5\Delta). \tag{11}$$

With the discrete-time version of the dynamics in (7) at hand, the MPC problem in (3) was formulated with a prediction horizon of N = 20 steps. The cost

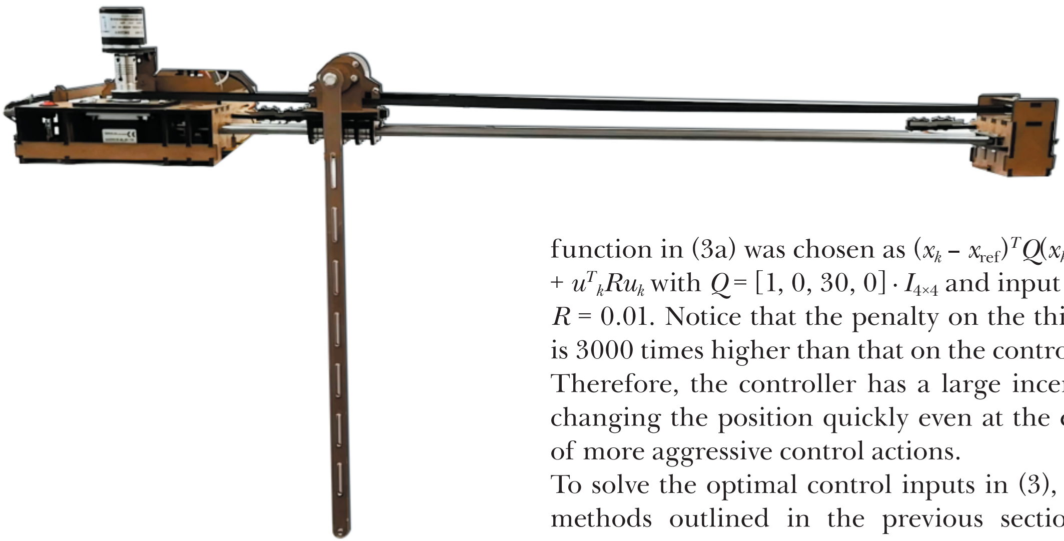

Fig. 1. Laboratory Pendulum.

function in (3a) was chosen as $(x_k - x_{\mathrm{ref}})^T Q (x_k - x_{\mathrm{ref}}) + u^T{}_k R u_k$ with $Q = [1, 0, 30, 0] \cdot I_{4\times4}$ and input penalty $R = 0.01$. Notice that the penalty on the third state is 3000 times higher than that on the control input. Therefore, the controller has a large incentive of changing the position quickly even at the expense of more aggressive control actions.

To solve the optimal control inputs in (3), the two methods outlined in the previous section were applied. First, the random shooting approach was applied in conjunction with the discretized

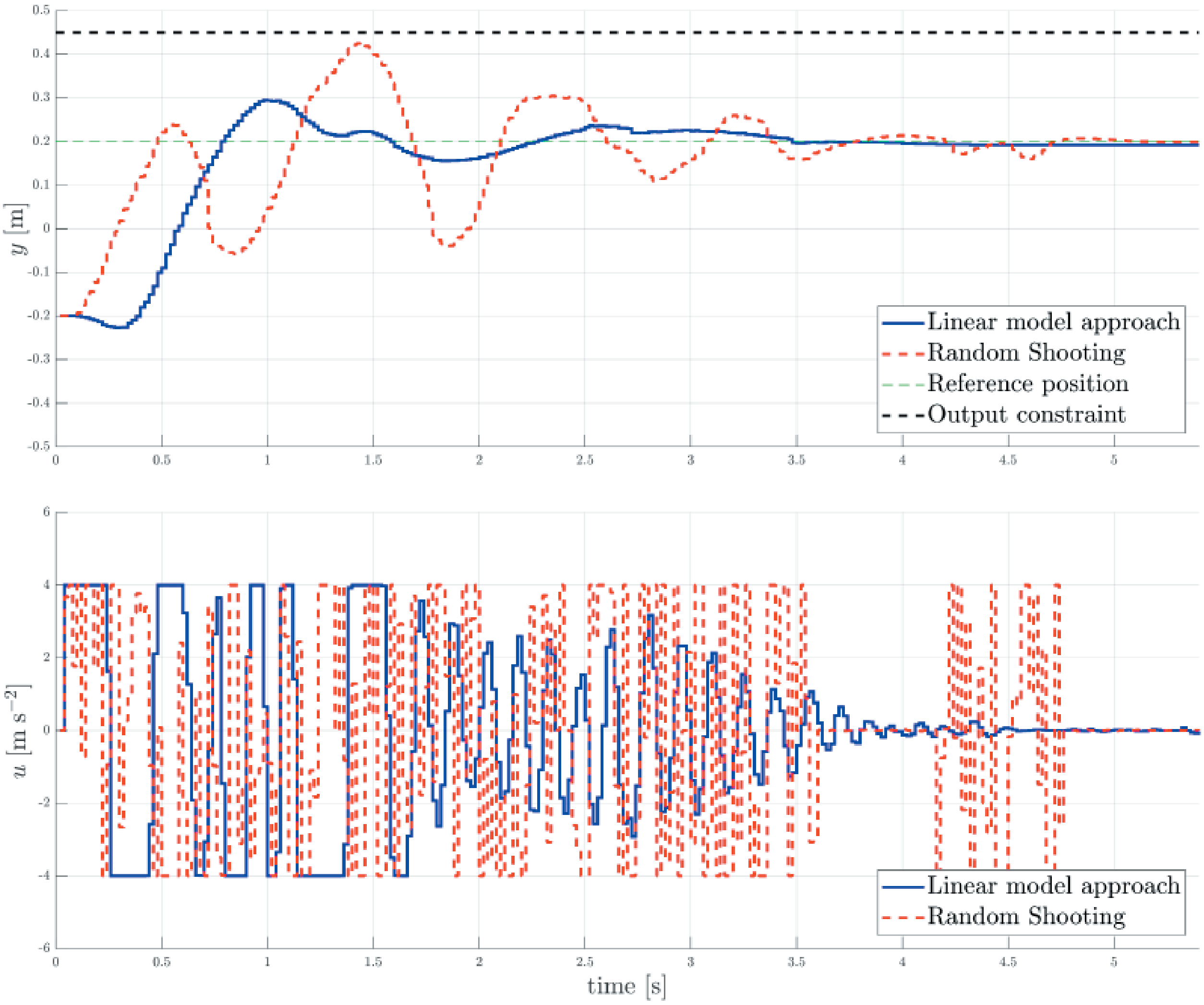

Fig. 2. First graph shows the position of the pendulum's endpoint over time. The blue solid line represents performance under linearization-based MPC, while the profiles for random shooting are shown using red dashed lines. The reference position of the pendulum's endpoint is represented by green dashed line and the constraint for this output by the black dashed line. Secong graph shows the respective control actions over time using the same color scheme.

nonlinear dynamics from (7). The random shooting controller was constructed using the RandShootMPCController method of the MPT toolbox, which also allows compiling the controller into a mex function for faster execution. To apply convex MPC, the dynamics was first linearized around a zero position and speed, providing (after discretization) the linear prediction model (6). Note that all constraints in (9) are already linear (thus the sets $\mathcal{X}$, $\mathcal{U}$, $\mathcal{Y}$ in (3d)–(3f) are convex), therefore no further processing was necessary. The resulting quadratic program was then solved using the Gurobi solver.

Both approaches were tested in experiments using a laboratory pendulum lab process, described in more details in Oravec et al. (2016b). However, not all state variables could be directly measured. Specifically, only x_1 (pendulum's angle) and x_3 (cart's position) can be directly accessed. As MPC is a state-feedback control policy, the remaining states have to be estimated by devising a suitable state observer, such as the Luenberger observer. The state estimation as well as computation of control actions from (3) and all data processing were implemented in Matlab on a PC connected to the pendulum via a USB cable. The MPC controller ran on a sampling time of 20 ms using the fixed-rate periodic timer in Matlab.

Results and discussion

To compare the control performance of each strategy, the following scenario was considered. A cart carrying a pendulum is in the initial position $x_3(0) = -0.2$ m in relation to the center of the range of its possible movements. At the beginning, the pendulum is in a steady downright position with $x_1(0) = x_2(0) = 0$. The objective of the controller is to move the cart to the desired final position $x_3(T) = 0.2$ m while satisfying all constraints presented in (9). Specifically, the pendulum's endpoint position must not exceed 0.45 m while the cart changes its position from –0.2 m to 0.2 m.

Experimental results are shown in Fig. 2. The first graph shows the measured position of the pendulum's endpoint over time. As it can be seen, both approaches are able to move the cart to the desired position within four seconds while respecting the constraints on the pendulum's endpoint position. However, the linearization-based method performs significantly better, exhibiting a smaller overshoot. The random shooting method, on the other hand, shows more oscillations, which is due to the fact that it generates control actions at a random basis. This can be observed from the bottom graph in Fig. 2. However, both controllers meet the specified control goal with a similar settling time.

It should be noted that the oscillatory behavior of the control actions is partly due to the physical construction of the device which only provides a finite resolution of measurements. Therefore, only sensory data are sampled, which leads to deteriorated performance.

Conclusions

This paper shown how MPC can be applied to perform agile manoeuvres using the cart-and-pole laboratory setup. Two versions of the MPC problem were considered. The first one employed a full nonlinear description of the processes but provided only a sub-optimal solution to the optimization problem. The second version was based on linearization of the nonlinear dynamics, followed by formulation and solution of the MPC problem as a convex quadratic program. Both versions of the controller were implemented in real time with satisfactory results using a sampling time of 20 ms.

Acknowledgements

The authors gratefully acknowledge the contribution of the Slovak Research and Development Agency under project APVV 15-0007 and of the Scientific Grant Agency of the Slovak Republic under grant 1/0585/19, as well as the Research & Development Operational Programme for project University Scientific Park STU in Bratislava, ITMS 26240220084, supported by Research 7 Development Operational Programme funded by the ERDF.

References

Allgöwer F, Zheng A (2012) Nonlinear model predictive control, volume 26. Birkhäuser.

Bakaráč P, Kvasnica M (2018) Fast nonlinear model predictive control of a chemical reactor: a random shooting approach. Acta Chimica Slovaca, 11(2): 175–181.

Bakošová M, Oravec J (2014) Robust mpc of an unstable chemical reactor using the nominal system optimization. Acta Chimica Slovaca, 7(2): 87–93.

Bakošová M, Oravec J, Matejičková K (2013) Model predictive control-based robust stabilization of a chemical reactor. Chemical Papers, 67(9): 1146–1156.

Čižniar M, Podmajerský M, Hirmajer T, Fikar M, Latifi MA (2009) Global optimization for parameter estimation of differential-algebraic systems. Chemical Papers, 63(3): 274–283.

Dyer M, Kannan R, Stougie L (2014) A simple randomised algorithm for convex optimisation. Mathematical Programming, 147(1–2): 207–229.

Kirkpatrick S, Gelatt C, Vecchi M (1983) Optimization by simulated annealing. Science, 220(4598): 671–680.

Klaučo M, Kalúz M, Kvasnica M (2017) Real-time implementation of an explicit mpc-based reference governor for control of a magnetic levitation system. Control Engineering Practice, (60): 99–105.

Maciejowski J (2002) Predictive control: with constraints. Pearson education.

Martin P, Odloak D, Kassab F (2013) Robust model predictive control of a pilot plant distillation column. Control Engineering Practice, 21(3): 231–241.

Oravec J, Bakošová M (2012) Robust constrained mpc stabilization of a cstr. Acta Chimica Slovaca, 5(2): 153–158.

Oravec J, Bakošová M (2015) Robust model-based predictive control of exothermic chemical reactor. Chemical Papers, 69(7).

Oravec J, Bakošová M, Mészáros A, Míková N (2016a) Experimental investigation of alternative robust model predictive control of a heat exchanger. Applied Thermal Engineering, 105: 774–782.

Oravec J, Bakošová M, Trafczynski M, Vasičkaninová A, Mészáros A, Markowski M (2018) Robust model predictive control and pid control of shell-andtube heat exchangers. Energy, 159: 1–10.

Oravec J, Kalúz M, Bakaráč P, Bakošová M (2016b) Improvements of educational process of automation and optimization using 2d plotter. In Preprints of the 11th IFAC Symposium on Advances in Control Education, volume 11, pp. 16–21.

Piovesan J, Tanner H (2009) Randomized model predictive control for robot navigation. In IEEE International Conference on Robotics and Automation, pp. 94–99.

Poli R, Kennedy J, Blackwell T (2007) Particle swarm optimization. Swarm intelligence, 1(1): 33–57.

Qin S, Badgwell T (2003) A survey of industrial model predictive control technology. Control engineering practice, 11(7): 733–764.

Sahoo S, Lampert C, Martius G (2018) Learning equations for extrapolation and control. arXiv preprint arXiv:1806.07259.

Wright S, Nocedal J (1999) Numerical optimization. Springer Science, 35(67–68): 7.