Slovak University of Technology in Bratislava
Institute of Information Engineering, Automation, and Mathematics

PROCEEDINGS

17$^{\mathrm{th}}$ International Conference on Process Control 2009

Hotel Baník, Štrbské Pleso, Slovakia, June 9 – 12, 2009

ISBN 978-80-227-3081-5

`http://www.kirp.chtf.stuba.sk/pc09`

Editors: M. Fikar and M. Kvasnica

Herceg, M., Kvasnica, M., Fikar, M., Čirka, Ľ.: Real-Time Control of a Thermo-Optical Device Using Polynomial Approximation of MPC Scheme, Editors: Fikar, M., Kvasnica, M., In *Proceedings of the 17th International Conference on Process Control '09*, Štrbské Pleso, Slovakia, 332–340, 2009.

Full paper online: `http://www.kirp.chtf.stuba.sk/pc09/data/abstracts/082.html`

REAL-TIME CONTROL OF A THERMO-OPTICAL DEVICE USING POLYNOMIAL APPROXIMATION OF MPC SCHEME

M. Herceg, M. Kvasnica, M. Fikar, and L'. Čirka

Institute of Information Eng., Automation, and Mathematics,
Slovak University of Technology,
81237 Bratislava, Slovakia
martin.herceg, michal.kvasnica, miroslav.fikar,
lubos.cirka@stuba.sk

Abstract: This paper deals with real-time control of a thermo-optical device. A polynomial approximation of the optimal Model Predictive Control (MPC) feedback law is employed as a controller. Such an approximate controller enjoys the key benefits of MPC schemes, namely it provides all-time constraint satisfaction and closed-loop stability guarantees. The main advantage of the proposed approximation scheme is that it can be implemented in real time using very limited computational resources.

Keywords: model predictive control, multiparametric programming, polynomial approximation

1. INTRODUCTION

MPC is a leading strategy in the control industry which offers optimal management of processes and, equally important, meets satisfaction of plant constraints (Maciejowski, 2002; Camacho and Bordons, 1999). Based on the process model, MPC approach foresees the future behavior of the process and searches for the best possible control inputs. This process is repeated every time as new process measurements arrive. As the search for best inputs is achieved by solving an optimization problem, process constraints can be easily handled which makes MPC superior to traditional proportional-integral-derivative (PID) controllers.

Since the general introduction of predictive control by Clarke et al. (1987), numerous MPC techniques have been established and most of them ended as commercially available products (Qin and Badgewell, 2003). This variety of MPC schemes can be separated in two groups, depending on how the particular optimization problem is solved. In the first group the optimization problem is solved on-line, that is, as the plant is under operation. This case applies to the majority of the processes in chemical industry with slow dynamics where there is enough time and computational resources for the optimization to terminate in time.

In the second group the optimization problem is solved *off-line*, that is, before plant's start-up. This approach is appealing especially for simple plants with fast dynamics, e.g. from electrotechnical industry (Geyer et al., 2008; Mariethoz et al., 2008a). This approach is often referred to as *explicit* MPC and for a recent survey see Alessio and Bemporad (2008). In the explicit MPC approach, most of the computational burden arising from optimization is shifted before the implementation phase, and the resulting controller is precomputed for all admissible operating conditions. The controller takes of a form of a piecewise affine (PWA) function mapping the initial conditions to the optimal sequence of control inputs. The implementation of such controllers then consists of a mere evaluation of such a function for the currently measured value of plant states. However, the complexity of the solution (and hence the complexity of the implementation phase) grows, in the worst case, exponentially withe the problem size (Zeilinger et al., 2008).

Up to date the best implementation scheme for evaluation of a PWA functions is the translation to a *binary search tree*, where the implementation complexity is logarithmic in the number of regions (Tøndel et al., 2003). However, when considering application where the sampling frequency is very high, even the binary search tree algorithm can be of prohibitive computational complexity. An alternative approach based on polynomial approximation of the explicit MPC control law has been developed recently Kvasnica et al. (2008). This method offers a suboptimal replacement of PWA function by a polynomial control law which significantly reduces requirements for storage and on-line evaluation. So far this method has been tested on a model of a DC-DC buck converter (Mariethoz et al., 2008b). This paper presents a benchmark experiment for this method where the controlled plant is represented by a thermo-optical device with fast dynamics and rapid sampling.

2. DEVICE DESCRIPTION

The uDAQ28/LT thermal-optical system is an experimental device aimed primarily for education purposes Huba et al. (2006). The device allows for real time measurement and control of temperature and light intensity. It can be connected to a personal computer via an universal serial bus (USB) without requiring an input-output card (Fig. 1). Data acquisition and real-time control of the uDAQ28/LT device is carried out in the Matlab/Simulink environment which allows very easy manipulation with the device.

The plant represents a dynamical system which combines slow and fast dynamics. The slow process is characterized by a heat transfer and the fast process corresponds to light emission. Both processes are caused by an embedded light bulb which is controlled by an input voltage signal. In general, the plant is characterized by five inputs and eight outputs whereas only three controlled inputs and three measured outputs are of interest. A precise description of these signals is given in Tab. 1.

The construction of the device suggests offers two main control loops. The primal loop regulates the light bulb intensity by manipulating the input voltage (or input voltage to LED [1] diode). The second loop maintains the inner temperature in safety limits by manipulating the revolutions of a cooling fan. Presence of physical constraints on manipulated and controlled variables makes the control task challenging and the device has often been used for benchmark of constrained PID control approaches (Huba and Vrančič, 2007).

Fig. 1. Front view on a thermo-optical device uDAQ28/LT.

Table 1. Description of measured and controlled signals.

Signal Name	Range
Input voltage to light bulb	0-5 V
Input voltage to cooling fan	0-5 V
Input voltage to LED	0-5 V
Inner temperature	0-100 deg C
Light intensity	not given
Revolutions of the cooling fan	0-6000 rpm

3. IDENTIFICATION AND PWA MODEL

In the sequel, only the optical channel of the light-bulb is considered. This decision is motivated by the fact that this channel is represented by a fast dynamics, which makes real-time implementation of a control system a challenging task. Due to very fast responses of the light channel, the sampling rate was selected the lowest admissible by Windows, i.e. $T_s = 0.05\,\mathrm{s}$. As the optical channel is sampled, it immediately suggests identification of input-output relations in discrete time.

Input-output relations of the optical channel have been identified with the help of IDTOOL Toolbox (Čirka et al., 2006) as a second order discrete transfer function

$$G(z^{-1}) = \frac{bz^{-2}}{1 + a_1 z^{-1} + a_2 z^{-2}} \tag{1}$$

where b, a_1, a_2 are constant parameters and z^{-1} is a discrete time delay operator (Mikleš and Fikar, 2007). IDTOOL toolbox contains the recursive least squares method of Kulhavý and Kárný (1984) which provides very good estimates of the unknown parameters. However, as transfer function is valid only locally, the identification was performed over four operating points and

[1] Light Emitting Diode

Table 2. Identification data over four operation points.

	input	output	b	a_1	a_2
(1)	1.3	6.84	2.03	-1.07	0.46
(2)	2.5	19.46	3.56	-0.97	0.43
(3)	3.5	32.09	4.51	-0.91	0.41
(4)	4.5	45.86	5.39	-0.87	0.40

the results are summarized in Tab. 2. For the use in explicit MPC scheme, the input-output representation (1) is transformed to a discrete state-space model. It is achieved by introducing state variables with discrete time instant k, i.e. $v_1(k) = y(k-1)$, $v_2(k) = y(k-2)$ and the state space model reads

$$v_1(k+1) = -a_1 v_1(k) - a_2 v_2(k) + bw(k) \quad (2a)$$
$$v_2(k+1) = v_1(k) \quad (2b)$$
$$y(k) = v_2(k). \quad (2c)$$

In (2) $w(k)$ represents the input voltage applied directly to the plant and $y(k)$ is the measured output. Voltage input is constrained

$$w(k) \in [0, 5]\,\mathrm{V} \quad (3)$$

and the measured output lies inside the interval

$$y(k) \in [0, 55] \quad (4)$$

of light intensity units (are not given in the reference manual). The overall input-output behavior of the optical channel can be recovered by aggregation of the local linear models (2) which forms piecewise linear or PWA model. Here, the operating area is first split into regions and local linear models are assigned to each such region. The overall behavior of PWA model is then driven by switching between the locally valid models using logical IF-THEN rules. To perform partitioning of the operating area according to linearization points in Tab. 2, a Voronoi diagram (Aurenhammer, 1991) is constructed, which directly returns partitions of the state space as a sequence of convex polytopes. This operation was executed using one of the routines included in MPT toolbox (Kvasnica et al., 2004) and it returned following regions:

$$\mathcal{R}_1 = \{v(k) \mid 0 \leq v_2(k) \leq 13.15\} \quad (5a)$$
$$\mathcal{R}_2 = \{v(k) \mid 13.15 \leq v_2(k) \leq 25.77\} \quad (5b)$$
$$\mathcal{R}_3 = \{v(k) \mid 25.77 \leq v_2(k) \leq 38.97\} \quad (5c)$$
$$\mathcal{R}_4 = \{v(k) \mid 38.97 \leq v_2(k) \leq 55\} \quad (5d)$$

To each of the regions (5), a corresponding local linear dynamics (2) is assigned, and it forms overall PWA model. Although PWA models are, in general, still non-linear, the underlying piecewise linearity allows for somewhat simpler controller design compared to full non-linear setups. Specifically, MPC problems based on PWA models can be solved explicitly, where the solution is obtained as a look-up table, easily implementable in real time.

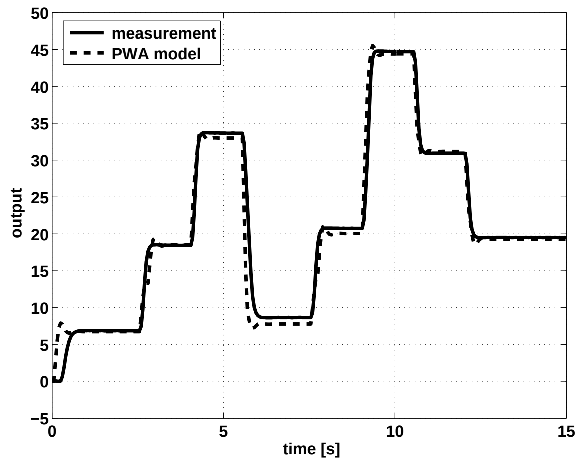


Fig. 2. Verification of the PWA model.

The output from PWA model has been compared to the real measured output from the plant and the result is depicted in Fig. 2. For the given scenario PWA model follows correctly the plant's output, thus the accuracy of the model is verified. It can be noticed that at the beginning there is larger mismatch between the plant and the model. It is caused by physical properties of a filament in bulb which requires certain time to incandesce from a cold startup. As this phase is over, the PWA model correctly captures the optical channel of the plant and it can be employed for MPC design.

4. CONSTRAINED PREDICTIVE CONTROL

MPC is an optimization-based approach which requires solving an optimization problem every time as new measurements are known. Information about the controlled plant are included into optimization problem by a process model upon which the future behavior is estimated. Control input is then picked as the one from all possible future realizations which fulfills specified criteria, e.g. constraint satisfaction, and it is applied to the plant. By this way an optimal operation of the plant can be attained. Solving of optimization problem at each sampling instant is, however, limited by a computational resources available for particular implementation scheme. Especially, if the process has a very fast dynamics, which is the case here, it might be not possible to solve MPC within one sampling instant, i.e. 0.05 s. Therefore the optimization problem needs to be pre-solved for all possible feedback information and this the core of multiparametric solutions to MPC or explicit MPC. This section applies the explicit MPC approach based on PWA models. For more details, the reader is referred to Borrelli (2003) and Baotić et al. (2006).

Table 3. Matrices of the normalized model (8).

A_1 B_1 f_1	1.072	-0.464	0.277	-1.492
	1	0	0	0
A_2 B_2 f_2	0.969	-0.431	0.485	-0.642
	1	0	0	0
A_3 B_3 f_3	0.913	-0.410	0.616	0
	1	0	0	0
A_4 B_4 f_4	0.868	-0.402	0.735	0.471
	1	0	0	0

4.1 *Prediction Model*

In order to prevent numerical issues when employing the PWA model for MPC synthesis, it is advised to perform coordinate transformation and normalization. This can be achieved by introducing normalized variables x_1, x_2 and u as follows:

$$x_1(k) = \frac{v_1(k) - v_{1,\text{ref}}}{\bar{v}_1}, \tag{6a}$$

$$x_2(k) = \frac{v_2(k) - v_{2,\text{ref}}}{\bar{v}_2}, \tag{6b}$$

$$u(k) = \frac{w(k) - w_{\text{ref}}}{\bar{w}}. \tag{6c}$$

The suffix "ref" represent the desired steady state value, i.e.

$$v_{1,\text{ref}} = 32.09, \quad v_{2,\text{ref}} = 32.09, \quad w_{\text{ref}} = 3.5 \tag{7}$$

which is basically the linearization point of the third dynamics (see Tab. 2) and $\bar{v}_1 = 3.67$, $\bar{v}_2 = 3.67$, $\bar{v} = 0.5$ are constants. Applying the normalization, the transformed PWA model yields

$$f_{\text{PWA}}(x(k), u(k)) = A_i x(k) + B_i u(k) + f_i \tag{8}$$

where $i = 1, 2, 3, 4$ and state update matrices are given in Tab. 3. The state space model (8) is associated with the following regions

$$\mathcal{D}_1 = \{x(k) \mid -8.75 \le x_2(k) \le -5.16\} \tag{9a}$$
$$\mathcal{D}_2 = \{x(k) \mid -5.16 \le x_2(k) \le -1.72\} \tag{9b}$$
$$\mathcal{D}_3 = \{x(k) \mid -1.72 \le x_2(k) \le 1.88\} \tag{9c}$$
$$\mathcal{D}_4 = \{x(k) \mid 1.88 \le x_2(k) \le 6.25\} \tag{9d}$$

Besides the dynamics as in (8), the following constraints are assumed to be imposed on the behavior of the prediction model:

$$\mathcal{X} = \{x(k) \mid -8.75 \le x_1(k) \le 6.25, \; -8.75 \le x_2(k) \le 6.25\} \tag{10a}$$
$$\mathcal{U} = \{u(k) \mid -7 \le u(k) \le 3\}. \tag{10b}$$

State constraints $\mathcal{X}$ are derived from the operating range of light intensity (4) and input constraints $\mathcal{U}$ represent the saturation limits (3).

4.2 *Control Problem*

The aim of the control strategy is to find an optimal sequence of control inputs such that all system states are driven to a desired equilibrium. The equilibrium is given by the linearization point for the third PWA dynamics (8) and in the transformed coordinates (6) it is exactly the origin, i.e. $x_1(k) = 0$, $x_2(k) = 0$, $u(k) = 0$. Mathematically, the problem can be formulated as to find a sequence of future control moves $U = [u(k),\, u(k+1),\, \dots,\, u(k+N-1)]$ up to horizon $N \in \mathbb{N}^+$ which steer the system states/input to the origin while satisfying constraints (10). More precisely,

$$\min_U \sum_{j=0}^{N-1} |Qx(k+j)|_1 + |Ru(k+j)|_1 \tag{11a}$$
$$\text{s.t.} \quad x(k+1) = f_{\text{PWA}}(x(k), u(k)) \tag{11b}$$
$$x(k+N) \in \mathcal{X}_f \tag{11c}$$
$$x(k+j) \in \mathcal{X} \tag{11d}$$
$$u(k+j) \in \mathcal{U} \tag{11e}$$

where $x(k) = [x_1(k),\, x_2(k)]^T$ represents the state vector, the function $f_{\text{PWA}}(\cdot)$ describes the PWA model defined in (8) and the sets $\mathcal{X}$, $\mathcal{U}$ are the constraints on input and state variables given by (10). The set $\mathcal{X}_f$ is introduced to obtain closed-loop stability guarantees (Mayne et al., 2000). The index 1 in the cost function (11a) denotes the 1-norm of given expression (sum of absolute values of vector components), matrices Q and R represent weighing factors.

Due to the presence of switching rules in the PWA model (8), the overall optimization problem (11) can be cast, using additional binary variables, as a mixed-integer linear program (MILP) (Bemporad and Morari, 1999). To solve the MILP problem (11) for all admissible initial conditions, the problem is solved using *multiparametric* programming Borrelli (2003), implemented in freely available tools (Kvasnica et al., 2004).

4.3 *Explicit MPC Synthesis*

Solving problem (11) in a multiparametric fashion a closed form solution $u(k)$ as PWA function which maps $x(k)$ onto $\mathcal{U}$. In particular, as was shown by Borrelli (2003), we have $u(k) = F_i x(k) + G_i$ if $x(k) \in \mathcal{P}_i$ for $i = 1, \dots, n_{\text{reg}}$. Here, $P_i = \{x(k) \mid H_i x(k) \le K_i\}$ are polyhedral sets (regions) of the state-space. Similarly, a closed-form expression for the optimal cost function (11a) is again a PWA function of the state, i.e. $V(k) = M_i x(k) + L_i$ if $x(k) \in \mathcal{P}_i$.

The problem (11) has been solved with parameters $N = \infty$, $Q = I$, $R = 0.5$ with the help of the MPT toolbox (Kvasnica et al., 2004). The choice of $N = \infty$ guarantees that the obtained MPC feedback law will provide closed-loop stability (Baotić et al., 2006).The resulting PWA control law builds a look-up table divided into 118 regions, defined in variables $x_1(k)$, $x_2(k)$, and these regions are plotted in Fig. 3(a). Over each one of these regions a local feedback law is defined as illustrates Fig. 3(b). Similarly, the cost function is shown in Fig. 3(c). Note that in the case of multiparametric MILP solutions, the resulting PWA control law can be discontinuous (Fig. 3(b)) and defined over a nonconvex set. This is a consequence of using binary variables to encode the IF-THEN rules which describe behavior of the PWA prediction model.

To implement the resulting look-up table in the on-line experiment, one has to store and evaluate the data. While storing part is limited by the available memory, the evaluation task is limited by the sampling time. The complexity of both tasks depend on the number of regions n_{reg}. Assuming that we have enough memory to store the look-up table, one have to still evaluate the PWA law. In fact, this task comprises of two steps

(1) region identification
(2) evaluation of PWA law

from which the first part consumes the most time. Even with the use of binary search tree algorithm, where the evaluation time is logarithmic in n_{reg} (Tøndel et al., 2003), the scheme can still be prohibitive for implementation. Motivated by this fact, the goal is to apply the approximation scheme of Kvasnica et al. (2008) where the whole look-up table is replaced by one polynomial, which is very cheap to implement. To do so, we have to find the set of all perturbations of the control law under which the closed loop renders stability. This will be explained in the next section.

4.4 *Stability Tubes*

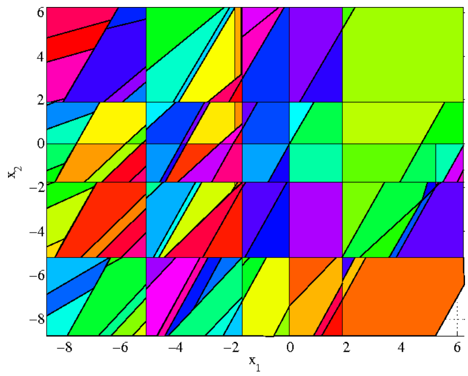
As was shown by Christophersen (2007), the explicit feedback law described in the previous section is just one of many stabilizing feedbacks. Specifically, based on the explicit solution to (11), one can compute the family of controllers which all stabilize the control model (8). This family is characterized by sets in the state and input space and is called *stability tubes.*

Definition 1. (Christophersen (2007)). Let $V(x)$ be a Lyapunov function for the system (8) with $x \in \mathcal{X}$ under a stabilizing controller $u(x) \in \mathcal{U}$. Then the set

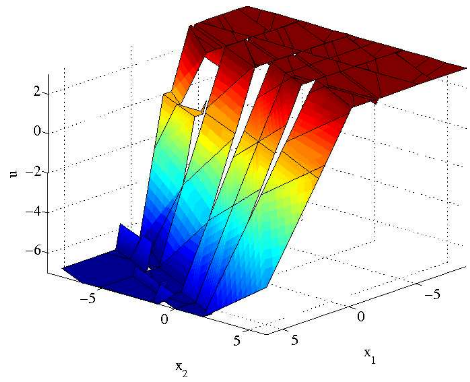
$$\mathcal{S}(V,\beta) := \{x(k) \in \mathcal{X},\ u(k) \in \mathcal{U}, \qquad (12)$$
$$V(x(k+1)) - V(x(k)) \leq -\beta(\|x(k)\|)\}\,.$$

is called a stability tube.

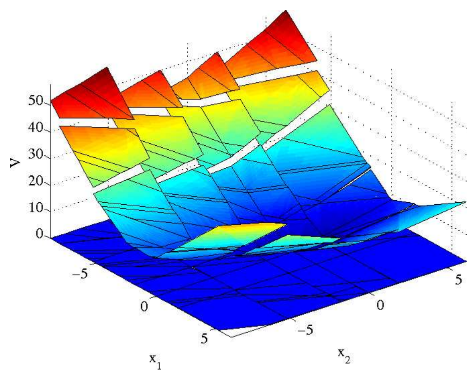
In other words, stability tubes are sets where the given Lyapunov value function $V(x)$ for system (8) decreases with a factor $\beta(\|x(k)\|) =$



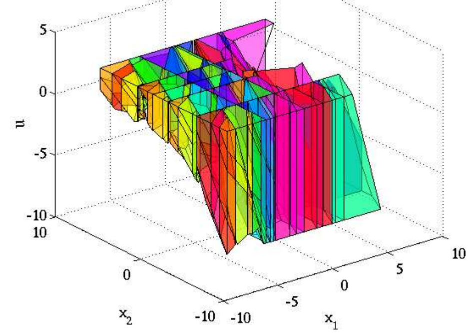
(a) Regions of the look-up table.



(b) Local control laws over each region.



(c) Value function.



(d) Stability tubes.

Fig. 3. Explicit solution to Problem (11) consists of PWA map defined over 118 regions.

$\beta\|x(k)\|_p \in \mathbb{R}$, $\beta > 0$, $p = \{1, \infty\}$. Any control input from within the stability tube will ensure that the closed loop will be stable and constraints on variables won't be violated. Precisely,

Theorem 2. (Christophersen (2007)). Let the assumptions of Definition 1 be fulfilled. Then every control law $u(x(k))$, $x(k) \in \mathcal{X}$, (also *any* sequence of control samples $u(k)$) fulfilling

$$(x(k), u(k)) \in \mathcal{S}(V, \beta) \tag{13}$$

asymptotically stabilizes the system (8) to the origin, $\forall x(k) \in \bigcup_i \mathcal{P}_i$.

Algorithm for computing the stability tubes and all the relating routines are included in the MPT Toolbox. Firstly, one has to find a piecewise affine Lyapunov function for the closed loop system. As the optimal solution is computed with the infinite horizon, the value function in Fig. 3(c) is a Lyapunov function. Secondly, one can apply routines for computing the stability tubes and the result is a collection of polyhedrals in the joint x-u space and it is shown in Fig. 3(d).

4.5 *Polynomial Approximation*

Using the approximation scheme of Kvasnica et al. (2008), the goal is to find a polynomial control law of the form

$$\mu(x) = [a_{11},\, a_{12}] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + [a_{21},\, a_{22}] \begin{bmatrix} x_1^2 \\ x_2^2 \end{bmatrix} + [a_{31},\, a_{32}] \begin{bmatrix} x_1^3 \\ x_2^3 \end{bmatrix} \tag{14}$$

which, when applied as a state feedback, guarantees closed-loop stability and constraint satisfaction. Theorem 2 provides a sufficient condition for existence of such a polynomial feedback law in the sense that if $(x, \mu(x)) \in \mathcal{S}(V, \beta)$, $\forall x \in \bigcup_i \mathcal{P}_i$, then $\mu(x)$ will provide closed-loop stability and constraint satisfaction. Therefore the search for suitable polynomial coefficients a_{ij} of (14) can be cast as the following optimization problem:

$$\min_{a_{11},\ldots,a_{32}} \sum_j \|(u(x) - \mu(x))\| \tag{15a}$$

$$\text{s.t.} \quad (x, \mu(x)) \in \mathcal{S}(V, \beta). \tag{15b}$$

From all possible choices of $\mu(x)$ which satisfy (15b), cost function (15a) is used to select the coefficients which provide best approximation of the optimal feedback law $u(x)$. As was shown in Kvasnica et al. (2008), optimization problem (15) can be formulated as a semidefinite programming problem, which can be solved using off-the-shelf tools.

The main advantage of the polynomial feedback law (14), compared to the MPC controller based on evaluating PWA feedback law, is reduction of the total implementation and storage cost. On the storage side, only the coefficients a_{ij} need to be recorded in the memory, compared to storing the regions $\mathcal{P}_i$ and the feedback laws F_i and G_i for the PWA feedback law. The on-line implementation cost is also greatly reduced, as only polynomial evaluation for a given x need to be performed to obtain a stabilizing control action.

Table 4. Coefficients of the approximated polynomial (14).

a_{11}, a_{12}	-0.8718, -0.0007
a_{21}, a_{22}	-0.0519, 0.0004
a_{31}, a_{32}	0.0019, 0.0001

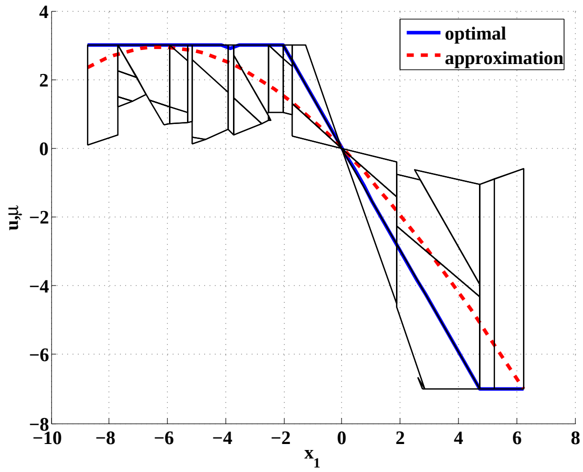


Fig. 4. Cross-section of the control laws through $x_2 = 0$.

The approximation scheme has been applied to obtain polynomial control law of type (14) with help of YALMIP (Löfberg, 2004). Computed coefficients are given in Tab. 4. Illustration of the approximation scheme is shown in Fig. 4 which represents a cross-section in stability tubes along the coordinate $x_2 = 0$. The polyhedral sets in Fig. 4 demonstrate the space of the stability tubes where there exist a stabilizing control law according to Theorem 2. Inside this space the approximated polynomial (14) has been fitted and it is shown in Fig. 4 with a dashed line while the optimal control law is depicted with solid line.

5. REAL-TIME IMPLEMENTATION

In this section computational requirements are evaluated for the optimal and approximated controller. Both controllers are applied in the real-time experiment and measured performance is discussed.

5.1 *Computational Demands*

Implementation of the optimal controller in the on-line experiment is limited by the sampling time $T_s = 0.05\,\mathrm{s}$. If the look-up table, obtained previously and consisting of 118 regions, is stored and evaluated using the binary search tree algorithm (Tøndel et al., 2003), the number of floating point operations per second (FLOPS) which are required to evaluate such a controller for a given initial condition is at most 41. The memory requirements are 2832 bytes for the control law and 1536 bytes for the search tree which gives a total of 4368 bytes.

In the polynomial approximation scheme, the number of FLOPS depend on the degree of approximated polynomial and on the polynomial degree. By considering the polynomial (14) with degree of three, the upper bound for evaluation FLOPS is 14, less than a half of the runtime for the binary search tree. More prominent, however, is the drop in memory consumption. As state above, the explicit MPC solution with 118 regions requires 4368 bytes of memory storage, while to store the polynomial feedback law (14), mere 24 bytes of memory are required (6 polynomial coefficients, each of them consuming 4 bytes when represented as floating point numbers).

5.2 *Experimental Data*

The optimal explicit MPC controller as well as the polynomial feedback strategy have been implemented in real time and obtained results are shown in Figs. 5(a), 5(b) and 5(c). The plots represent the transition from the initial condition $x_0 = [-8.7, -8.7]^T$ to the origin. Input signal generated by the optimal controller immediately jumps to the upper limit and then gently approaches the origin. In the polynomial controller this effect is different, the controller is slightly slower, but the same stabilizing effect is achieved. State and input profiles converge to desired steady state, hence the control objective was met with both approaches. It is interesting to note that a polynomial controller acts better (in the sense of the selected performance criterion (11a)) than the optimal one. In particular, (11a) evaluates to 146.34 when the optimal MPC controller is used as a feedback, compared to value of (11a) amounting to 142.96 for the case where the polynomial controller was used. This small difference can be attributed to the fact that the optimal controller is more sensitive to changes of the states. Nevertheless, the difference is small enough to say that both controllers share roughly the same performance while the approximated controller is significantly cheaper than the optimal one.

Performance of both controllers has not been tested on disturbance attenuation because this effect cannot be fully compensated by any of the used controllers since they do not contain an integration part. Moreover, these effects are too small to satisfactory evaluate the performance of both controllers while showing their advantages (e.g. constraint satisfaction).

5.3 *Conclusion*

Main motivation of this paper was to demonstrate a cheap alternative to explicit MPC scheme based on polynomial approximation of the optimal feedback law. Control of the optical channel of uDAQ28/LT device is considered as a benchmark example to polynomial approximation scheme of Kvasnica et al. (2008). The process is identified as a second order linear discrete time system around four operating points. Based on the identification data, PWA model is constructed and deployed for MPC design. The MPC problem is solved in the multiparametric fashion, i.e. precomputing the controller for whole possible operating conditions, and the result is stored as a look-up table. The properties of the explicit solution are further exploited and a family of all stabilizing controllers is constructed. From this family, one controller of a special polynomial structure has been selected, which implementation cost is the cheapest. The polynomial controller has been experimentally tested in the closed loop, and has shown good results.

Acknowledgment

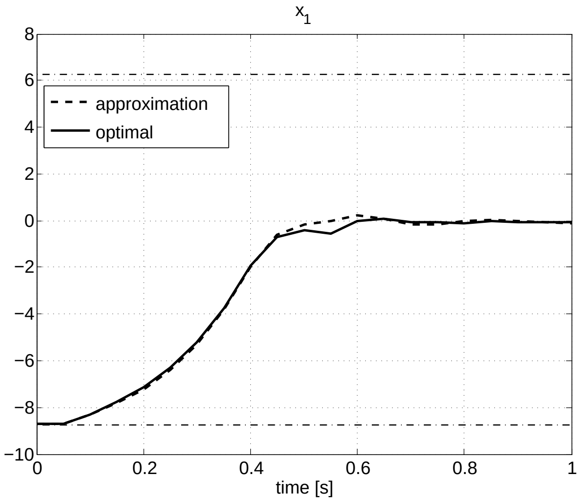
The authors are pleased to acknowledge the financial support of the Scientific Grant Agency of the Slovak Republic under the grants 1/0071/09 and 1/4055/07. This work was supported by the Slovak Research and Development Agency under the contracts No. VV-0029- 07 and No. LPP-0092-07.

References

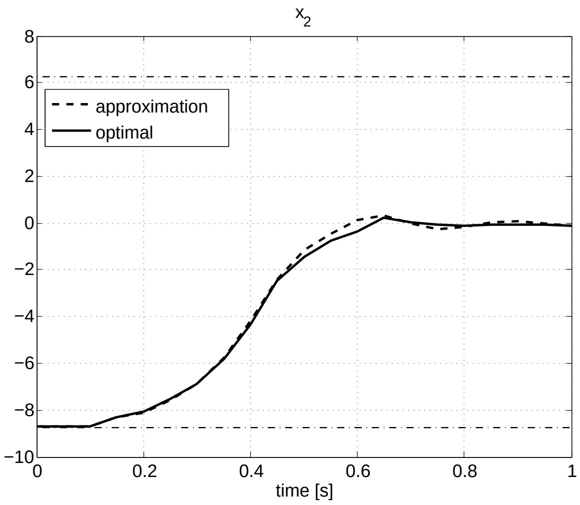
A. Alessio and A. Bemporad. A survey on explicit model predictive control. In L. Magni, D. Raimondo, and F. Allgöwer, editors, *International Workshop on Assessment and Future Directions of Nonlinear Model Predictive Control*, Pavia, Italy, 2008.

F. Aurenhammer. Voronoi diagramns – survey of a fundamental geometric data structure. *ACM Computing Surverys*, 3(23):345–405, 1991.

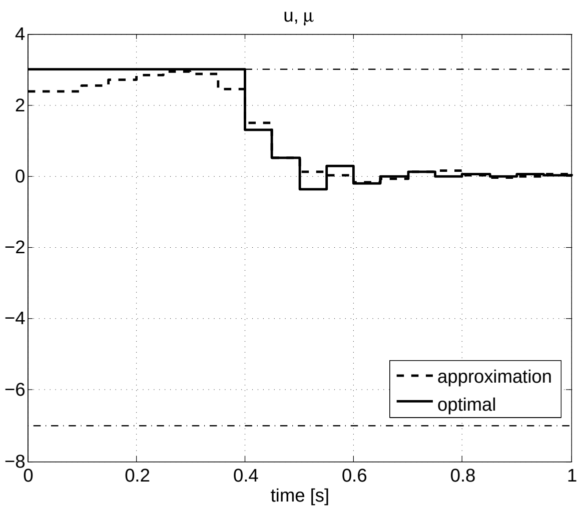
M. Baotić, F. J. Christophersen, and M. Morari. Constrained Optimal Control of Hybrid Systems With a Linear Performance Index. *IEEE*



(a) Profiles of the state variable x_1.



(b) Profiles of the state variable x_2.



(c) Input profiles.

Fig. 5. Experimental data for optimal and polynomial controller.

Trans. on Automatic Control, 51(12):1903–1919, December 2006.

A. Bemporad and M. Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–427, March 1999.

F. Borrelli. Constrained Optimal Control of Linear and Hybrid Systems. In *Lecture Notes in Control and Information Sciences*, volume 290. Springer, 2003.

E. F. Camacho and C. Bordons. *Model Predictive Control*. Springer, 1999.

F. J. Christophersen. Optimal Control of Constrained Piecewise Affine Systems. In *Lecture Notes in Control and Information Sciences*, volume 359. Springer-Verlag, 2007.

D. W. Clarke, C. Mohtadi, and P. S. Tuffs. Generalized predictive control – Parts I-II. *Automatica*, 23(2), 1987.

T. Geyer, G. Papafotiou, and M. Morari. Hybrid Model Predictive Control of the Step-Down DC-DC Converter. *IEEE Transactions on Control Systems Technology*, 16(6):1112–1124, November 2008.

M. Huba and D. Vrančič. Constrained control of the plant with two different modes. In J. Mikleš, M. Fikar, and M. Kvasnica, editors, *16th Int. Conf. Process Control*, pages paper Le–Tu–5, 205p.pdf, 2007.

M. Huba, P. Kurčík, and M. Kamenský. *Thermo-optical device uDAQ28/LT*. STU Bratislava, Illkovičova 3, Bratislava, 2006. In Slovak.

R. Kulhavý and M. Kárný. Tracking of slowly varying parameters by directional forgetting. In *Proceedings of the 9th IFAC World Congres*, Budapest, Hungary, 1984.

M. Kvasnica, P. Grieder, M. Baotic, and M. Morari. Multi-Parametric Toolbox (MPT). In *Hybrid Systems: Computation and Control*, pages 448–462, March 2004. URL `http://control.ee.ethz.ch/~ MPT`.

M. Kvasnica, F. J. Christophersen, M. Herceg, and M. Fikar. Polynomial approximation of closed-form MPC for piecewise affine systems. In *Proceedings of the 17th IFAC World Congres,*, pages 3877–3882, Seoul, South Korea, 2008.

J. Löfberg. Yalmip : A toolbox for modeling and optimization in MATLAB. In *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004. URL `http://control.ee.ethz.ch/~joloef/yalmip.php`.

J. M. Maciejowski. *Predictive Control with Constraints*. Prentice Hall, 2002.

S. Mariethoz, A. G. Beccuti, G. Papafotiou, and M. Morari. Sensorless explicit model predictive control of the DC-DC buck converter with inductor current limitation. In *IEEE Applied Power Electronics Conference*, February 2008a.

S. Mariethoz, M. Herceg, and M. Kvasnica. Model Predictive Control of buck DC-DC converter with nonlinear inductor. In *IEEE COMPEL Workshop on Control and Modeling for Power Electronics*, Zurich, Switzerland, August 2008b.

D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36:789–814, 2000.

J. Mikleš and M. Fikar. *Process Modelling, Identification, and Control*. Springer Verlag, Berlin Heidelberg, 2007.

S. J. Qin and T. A. Badgewell. A survey of industrial model predictive control technology. *Control engineering Practice*, 11:733–764, 2003.

P. Tøndel, T. A. Johansen, and A. Bemporad. Evaluation of Piecewise Affine Control via Binary Search Tree. *Automatica*, 39(5):945–950, May 2003.

Ľ. Čirka, M. Fikar, and P. Petruš. IDTOOL 4.0 - A Dynamical System Identification Toolbox for MATLAB/Simulink. In *14th Annual Conference Proceedings: Technical Computing Prague 2006*, pages 29–29. The MathWorks, Inc. & HUMUSOFT s.r.o. & Institute of Chemical Technology in Prague, October 2006. URL `http://www.kirp.chtf.stuba.sk/~cirka/idtool/?IDTOOL`.

M. Zeilinger, C. N. Jones, and M. Morari. Realtime suboptimal Model Predictive Control using a combination of Explicit MPC and Online Computation. In *Conference on Decision and Control, CDC*, Cancun, Mexico, December 2008.