Model Predictive Control for Offset-Free Reference Tracking

Květoslav Belda
Department of Adaptive Systems
The Institute of Information Theory and Automation of the Czech Academy of Sciences
Pod Vodárenskou věží 4, 182 08 Prague 8, Czech Republic
e-mail: *belda@utia.cas.cz*

Abstract — **The paper deals with the offset-free reference tracking problem of the Model Predictive Control (MPC). That problem is considered for a class of the constant or occasionally changed constant reference signals. Proposed solution arises from a simple subtraction of the ARX model of two consecutive time steps. The solution is adapted to a state-space form and it corresponds to usual predictive control design without increase of the design complexity. The construction of the prediction equations and predictive controller structure is explained in the paper.**

Keywords — offset-free reference tracking, predictive control, ARX model, state-space model, multi-input multi-output system, robotic system, mechatronic system

I. Introduction

Offset-free reference tracking is a frequent requirement in many industrial control applications. The problem solution is especially important for mechatronic systems like robots and manipulators at manipulation and positioning operations. In these applications, reaching accurate positions and dwell in them without any offsets or oscillation is required.

Model Predictive Control (MPC) is an efficient control method for many industrial applications due to its flexibility and comprehensibility. However, an undesirable offset of the system output from the set point (reference signal) may occur in any basic MPC implementation owing to its positional, proportional character. Offsets appear if unmeasured steadily constant disturbance enters the system or also if a mismatch between the real system and the model used for predictions in the MPC exists.

Several approaches to avoid the undesirable offset behaviour are in the literature. They are based either on external integrators or on specific incremental modifications. A conventional approach includes an integrator in the control loop as in the case of the PI control [1]. The integrator in the loop, but outside of the MPC controller, produces the I-control channel. However, this integrator is not under any optimization.

The approach base on the external integrator solves simply the offset problem, but it may cause unsafe control actions for systems with high dynamics and over-actuated systems [6]. In Fig. 1, there are possible cases for one selected control channel applied to an over-actuated robotic system with the offset-free reference tracking problem caused by a constant disturbance. It shows both uncompensated and compensated real cases (hardware in the loop) and ideal case (simulation experiments).

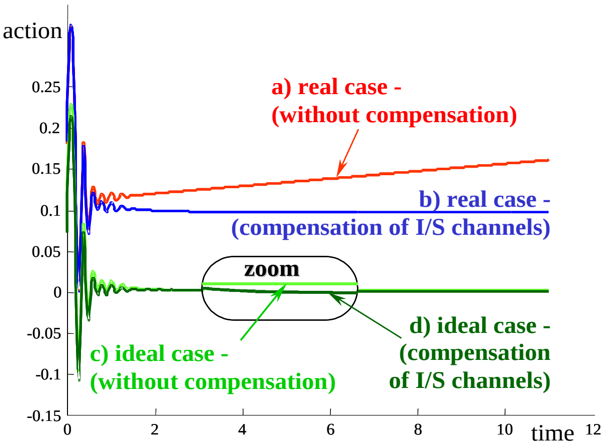


Fig. 1. Comparison of possible control actions:
a), b) real cases caused by model mismatch and constant disturbance;
c), d) ideal cases without disturbance but altered model.

Real offset-free approaches consider constant disturbances directly in their design, i.e. ‘offset-free property’ is directly involved in the resultant action. These approaches are based on the determination of steady-state reference signals using an internal disturbance model and simultaneous searching for the stabilizing controller [1], [2], [4], [5]. This arrangement means to use a number of additional variables corresponding to number of controlled variables. It may lead to a significant increase in the dimension of the control design [16]. Furthermore, for the state-space realization, the approaches are usually combined with some system state-space estimation in the case of unmeasured or immeasurable state-space variables [3].

This paper deals with the solution of the offset-free reference tracking problem by a simple differential rearrangement of the models describing a controlled system. A proposed approach leads to a specific application of the measured system output in a MPC design and it keeps dimensional range as usual MPC approaches [9]. It can be applied for both ARX models (i.e. Input/Output models or data driven models) and state-space models [16], [18].

The paper is organized as follows. Section II defines the used model. Section III deals with the model rearrangement with a composition of specific equations of predictions. Section IV discusses a suitable mathematical realization of the predictive controller, i.e. the optimization of a MPC cost function. Section V demonstrates the theoretical results achieved in Sections III and IV on several comparative examples.

TELEN2016001
DOI 10.14311/TEE.2016.1.008

II. Model Definition

A model, used for description of a controlled system, influences significantly the design of control actions [10]. The best results of the control process are achieved, when the model is composed via thoroughgoing mathematical and physical analysis. It is often difficult. In such cases, the system is modelled approximately or identified experimentally from the available data. However, it can lead to some mismatch between the obtained model and real controlled system.

A selection of the appropriate model is determined by a MPC algorithm, in which the model is applied. Due to dominant digital realization of automating devices, the discrete control procedures are preferred. Thus, models for the control design are composed as discrete in spite of the fact that majority of the controlled systems are continuous. Furthermore, the discrete realization naturally determines a finite time range for the proper computation of the control.

The design of the MPC algorithms usually uses the state-space model formulation:

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{A}\,\mathbf{x}_k + \mathbf{B}\,\mathbf{u}_k + \mathbf{v}_k \\ \mathbf{y}_k &= \mathbf{C}\,\mathbf{x}_k + \mathbf{e}_k \end{aligned} \tag{1}$$

where $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ are state, input and output matrices respectively; $\mathbf{u}_k$ and $\mathbf{y}_k$ are system inputs and outputs; $\mathbf{x}_k$ is a state of the system, $\mathbf{v}_k$ is a process disturbance and $\mathbf{e}_k$ is a measurement disturbance.

In this manner, as a clue, the state-space formulation is used for the composition of the equations of predictions. However, the intended ARX model (AutoRegressive model with eXternal input) is considered as an initial model. The ARX models belong to Input/Output or data-driven models and work only with external system signals. They correspond closely to the mathematical and physical analysis. Note, that the direct use of the state-space models leads to similar consequences.

Let us consider an usual ARX model for the Single-Input Single-Output (SISO) system, which describes relations just among system inputs and outputs:

$$y_k = \sum_{i=1}^{n} b_i u_{k-i} - \sum_{i=1}^{n} a_i y_{k-i} + e_k \tag{2}$$

where n is order of a controlled system; $y_{()}$ and $u_{()}$ are values of the system output and input; and e_k is error or some noise of measurement of the system output y_k. To compose the equations of predictions, the ARX model can be written in the following vector form:

$$\begin{aligned} y_k &= [-a_n \cdots -a_2 \; -a_1\,][\,y_{k-n} \cdots y_{k-2} \; y_{k-1}]^T \\ &+ [b_n \cdots b_2 \; b_1\,][\,u_{k-n} \cdots u_{k-2} \; u_{k-1}]^T + e_k \end{aligned} \tag{3}$$

A similar form is also convenient for the Multi-Input Multi-Output (MIMO) systems:

$$\mathbf{y}_k = \sum_{i=1}^{n} \mathbf{B}_i \mathbf{u}_{k-i} - \sum_{i=1}^{n} \mathbf{A}_i \mathbf{y}_{k-i} + \mathbf{e}_k \tag{4}$$

where n is still order of the controlled system; $\mathbf{y}_{()}$ and $\mathbf{u}_{()}$ are vectors of the output and input values as follows

$$\mathbf{y}_{k-i} = \begin{bmatrix} y_{(1)\,k-i} \\ \vdots \\ y_{(ny)\,k-i} \end{bmatrix}, \quad \mathbf{u}_{k-i} = \begin{bmatrix} u_{(1)\,k-i} \\ \vdots \\ u_{(nu)\,k-i} \end{bmatrix} \tag{5}$$

and $\mathbf{e}_k$ is a ny dimensional error vector corresponding to the measurement noise of the system outputs $\mathbf{y}_k$. The parameters are included in matrices $\mathbf{B}_i$ and $\mathbf{A}_i$

$$\mathbf{B}_i = \begin{bmatrix} b_i^{11} & \cdots & b_i^{1\,nu} \\ \vdots & \ddots & \vdots \\ b_i^{ny\,1} & \cdots & b_i^{ny\,nu} \end{bmatrix}, \quad \mathbf{A}_i = \begin{bmatrix} a_i^{11} & \cdots & a_i^{1\,ny} \\ \vdots & \ddots & \vdots \\ a_i^{ny\,1} & \cdots & a_i^{ny\,ny} \end{bmatrix} \tag{6}$$

Then, a suitable form like form (3) is:

$$\begin{aligned} \mathbf{y}_k &= [-\mathbf{A}_n \cdots -\mathbf{A}_2 \; -\mathbf{A}_1\,][\mathbf{y}_{k-n}^T \cdots \mathbf{y}_{k-2}^T \; \mathbf{y}_{k-1}^T]^T \\ &+ [\mathbf{B}_n \cdots \mathbf{B}_2 \; \mathbf{B}_1\,][\mathbf{u}_{k-n}^T \cdots \mathbf{u}_{k-2}^T \; \mathbf{u}_{k-1}^T]^T + \mathbf{e}_k \end{aligned} \tag{7}$$

The value of the system output $\mathbf{y}_k$ is modelled by its deterministic part $\hat{\mathbf{y}}_k$ and stochastic unpredicted part $\mathbf{e}_k$:

$$\mathbf{y}_k = \hat{\mathbf{y}}_k + \mathbf{e}_k \tag{8}$$

or similarly for the state-space systems:

$$\begin{aligned} \mathbf{x}_{k+1} &= \hat{\mathbf{x}}_{k+1} + \mathbf{v}_k \\ \mathbf{y}_k &= \hat{\mathbf{y}}_k + \mathbf{e}_k \end{aligned} \tag{9}$$

The defined model forms are initial forms for the construction of equations of predictions in the following section.

III. Equations of Predictions

Let us consider, for the sake of generality, the ARX model (7) describing the MIMO system and topical measurement of the system output $\mathbf{y}_k$ and known past system outputs $\mathbf{y}_{()}$ and inputs $\mathbf{u}_{()}$ in time steps $k-1, \cdots k-n$. The aim is to prepare specific equations of predictions, which lead to incremental algorithm of the MPC.

Let us start from a simple subtraction of the ARX model in the predictor form in two consecutive time steps $k+1$ and k:

$$\begin{aligned} \mathbf{y}_{k+1} - \mathbf{y}_k = \; & [-\mathbf{A}_n \; \cdots \; -\mathbf{A}_1\,] \times \\ & [(\mathbf{y}_{k-n+1} - \mathbf{y}_{k-n})^T, \; \cdots, \; (\mathbf{y}_k - \mathbf{y}_{k-1})^T\,]^T \\ & + [\mathbf{B}_n \; \cdots \; \mathbf{B}_1\,] \times \\ & [(\mathbf{u}_{k-n+1} - \mathbf{u}_{k-n})^T, \; \cdots, \; (\mathbf{u}_k - \mathbf{u}_{k-1})^T\,]^T \\ & + \mathbf{e}_{k+1} - \mathbf{e}_k \end{aligned} \tag{10}$$

DOI 10.14311/TEE.2016.1.008

Furthermore, let us define a predictable deterministic estimate $\Delta\hat{\mathbf{y}}_{k+1}$ of the increment $\Delta\mathbf{y}_{k+1}$ between two consecutive time steps:

$$\begin{aligned}\Delta\hat{\mathbf{y}}_{k+1} &= E(\mathbf{y}_{k+1}-\mathbf{y}_k)\\ &=[-\mathbf{A}_n \cdots -\mathbf{A}_1][\Delta\mathbf{y}^T_{k-n+1}\cdots\Delta\mathbf{y}^T_k]^T \\ &+\ [\mathbf{B}_n\cdots\mathbf{B}_1][\Delta\mathbf{u}^T_{k-n+1}\cdots\Delta\mathbf{u}^T_k]^T\end{aligned} \tag{11}$$

where E is the mean operator (expectation).

The expression (11) of the increment $\Delta\mathbf{y}_{k+1}$ can be transformed to the following state-space like form:

$$\begin{bmatrix}\Delta\mathbf{y}_{k-n+2}\\ \Delta\mathbf{y}_{k-n+3}\\ \vdots\\ \Delta\mathbf{y}_k\\ \Delta\hat{\mathbf{y}}_{k+1}\end{bmatrix} = \begin{bmatrix}\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\\ \vdots\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{1} & \mathbf{0} & \cdots & \mathbf{0}\\ \mathbf{0} & \ddots & \ddots & \vdots\\ \vdots & \ddots & \ddots & \mathbf{0}\\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{1}\end{bmatrix}\\ -\mathbf{A}_n & -\mathbf{A}_{n-1}\cdots-\mathbf{A}_2\ -\mathbf{A}_1\end{bmatrix}\begin{bmatrix}\Delta\mathbf{y}_{k-n+1}\\ \Delta\mathbf{y}_{k-n+2}\\ \vdots\\ \Delta\mathbf{y}_{k-1}\\ \Delta\mathbf{y}_k\end{bmatrix}$$

$$+\begin{bmatrix}\begin{bmatrix}\mathbf{0}\\ \mathbf{0}\\ \vdots\\ \mathbf{0}\end{bmatrix} & \begin{bmatrix}\mathbf{0} & \mathbf{0} & \cdots & \mathbf{0}\\ \mathbf{0} & \ddots & \ddots & \vdots\\ \vdots & \ddots & \ddots & \mathbf{0}\\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{0}\end{bmatrix}\\ \mathbf{B}_n & \mathbf{B}_{n-1}\cdots\ \mathbf{B}_2\ \ \mathbf{B}_1\end{bmatrix}\begin{bmatrix}\Delta\mathbf{u}_{k-n+1}\\ \Delta\mathbf{u}_{k-n+2}\\ \vdots\\ \Delta\mathbf{u}_{k-1}\\ \Delta\mathbf{u}_k\end{bmatrix} \tag{12}$$

The state-space form (12) can be written in the matrix notation:

$$\Delta\hat{\mathbf{x}}_{k+1} = \mathbf{A}\,\Delta\mathbf{x}_k + \mathbf{B}\,\Delta\tilde{\mathbf{u}}_k \tag{13}$$

Then, the predicted output increment $\Delta\hat{\mathbf{y}}_{k+1}$ is expressed by the output increment equation:

$$\begin{aligned}\Delta\hat{\mathbf{y}}_{k+1} &= \mathbf{C}\,\Delta\hat{\mathbf{x}}_{k+1}\\ \mathbf{C} &= [\mathbf{0}\quad \mathbf{0}\quad \cdots\quad \mathbf{0}\quad \mathbf{1}]\end{aligned} \tag{14}$$

Finally, the equation for the resulting future system output based on the topical measurement output $\mathbf{y}_k$ is

$$\hat{\mathbf{y}}_{k+1} = \mathbf{y}_k + \Delta\hat{\mathbf{y}}_{k+1} \tag{15}$$

Equations (12) – (15) give a suitable prediction form, which is formally similar to the usual standard state-space notations [7].

Using the recursive procedure, the evolution of predictions for the state-like output increments is the following:

$$\begin{aligned}\Delta\hat{\mathbf{x}}_{k+1} &= \mathbf{A}\,\Delta\mathbf{x}_k + \mathbf{B}\,\Delta\tilde{\mathbf{u}}_k\\ \Delta\hat{\mathbf{x}}_{k+2} &= \mathbf{A}^2\,\Delta\mathbf{x}_k + \mathbf{A}\,\mathbf{B}\,\Delta\tilde{\mathbf{u}}_k + \mathbf{B}\,\Delta\tilde{\mathbf{u}}_{k+1}\\ &\ \vdots\\ \Delta\hat{\mathbf{x}}_{k+N} &= \mathbf{A}^N\,\Delta\mathbf{x}_k + \mathbf{A}^{N-1}\,\mathbf{B}\,\Delta\tilde{\mathbf{u}}_k + \cdots + \mathbf{B}\,\Delta\tilde{\mathbf{u}}_{k+N-1}\end{aligned} \tag{16}$$

A corresponding evolution for the outputs is expressed as:

$$\begin{aligned}\hat{\mathbf{y}}_{k+1} &= \mathbf{y}_k + \mathbf{C}\,\Delta\hat{\mathbf{x}}_{k+1}\\ \hat{\mathbf{y}}_{k+2} &= \mathbf{y}_k + \mathbf{C}\,\Delta\hat{\mathbf{x}}_{k+1} + \mathbf{C}\,\Delta\hat{\mathbf{x}}_{k+2}\\ &\ \vdots\\ \hat{\mathbf{y}}_{k+N} &= \mathbf{y}_k + \mathbf{C}\,\Delta\hat{\mathbf{x}}_{k+1} + \cdots + \mathbf{C}\,\Delta\hat{\mathbf{x}}_{k+N}\end{aligned} \tag{17}$$

After replacement of the predicted increments in (17) by the expressions from (16), the basis of equations of predictions is the following:

$$\begin{aligned}\hat{\mathbf{y}}_{k+1} &= \mathbf{y}_k + \mathbf{C}\mathbf{A}\Delta\mathbf{x}_k\ \ + \mathbf{C}\mathbf{B}\Delta\tilde{\mathbf{u}}_k\\ \hat{\mathbf{y}}_{k+2} &= \mathbf{y}_k + \mathbf{C}\mathbf{A}\Delta\mathbf{x}_k\ \ + \mathbf{C}\mathbf{B}\Delta\tilde{\mathbf{u}}_k\\ &\quad + \mathbf{C}\mathbf{A}^2\Delta\mathbf{x}_k + \mathbf{C}\mathbf{A}\mathbf{B}\Delta\tilde{\mathbf{u}}_k + \mathbf{C}\mathbf{B}\Delta\tilde{\mathbf{u}}_{k+1}\\ &\ \vdots\\ \hat{\mathbf{y}}_{k+N} &= \mathbf{y}_k + \mathbf{C}\mathbf{A}\Delta\mathbf{x}_k\ \ + \mathbf{C}\mathbf{B}\Delta\tilde{\mathbf{u}}_k + \cdots\\ &\cdots\ + \mathbf{C}\mathbf{A}^N\Delta\mathbf{x}_k + \mathbf{C}\mathbf{A}^{N-1}\mathbf{B}\Delta\tilde{\mathbf{u}}_k + \cdots + \mathbf{C}\mathbf{B}\Delta\tilde{\mathbf{u}}_{k+N-1}\end{aligned} \tag{18}$$

This basis can be written in the adapted notation:

$$\begin{bmatrix}\hat{\mathbf{y}}_{k+1}\\ \hat{\mathbf{y}}_{k+2}\\ \vdots\\ \hat{\mathbf{y}}_{k+N}\end{bmatrix} = \begin{bmatrix}\mathbf{1}\\ \mathbf{1}\\ \vdots\\ \mathbf{1}\end{bmatrix}\mathbf{y}_k + \begin{bmatrix}\mathbf{C}\mathbf{A} & \\ \mathbf{C}\mathbf{A}+\mathbf{C}\mathbf{A}^2 & \\ \vdots & \ddots\\ \mathbf{C}\mathbf{A}+\cdots+\mathbf{C}\mathbf{A}^N & \end{bmatrix}\Delta\mathbf{x}_k$$

$$+\begin{bmatrix}\mathbf{C}\mathbf{B} & \mathbf{0} & \cdots & \mathbf{0}\\ \mathbf{C}\mathbf{A}\mathbf{B}+\mathbf{C}\mathbf{B} & \mathbf{C}\mathbf{B} & \ddots & \vdots\\ \vdots & & \ddots & \mathbf{0}\\ \mathbf{C}\mathbf{A}^{N-1}\mathbf{B}+\cdots+\mathbf{C}\mathbf{B} & & \cdots & \mathbf{C}\mathbf{B}\end{bmatrix}\Delta\tilde{\mathbf{u}}_{k+N-1} \tag{19}$$

The increments of the state-like vector and control vectors in (19) are defined in this way:

$$\begin{aligned}\Delta\mathbf{x}_k &= [\underbrace{(\mathbf{y}_{k-n+1}-\mathbf{y}_{k-n})^T,\ \cdots,\ (\mathbf{y}_k-\mathbf{y}_{k-1})^T}_{\text{known past}}]^T\\ \Delta\tilde{\mathbf{u}}_{k+N-1} &= [\ \underbrace{\Delta\mathbf{u}^T_{k-n+1}\ \cdots\ \Delta\mathbf{u}^T_{k-1}}_{\text{known past } = \Delta\tilde{\mathbf{u}}_{k-1}}\ \ \underbrace{\Delta\mathbf{u}^T_k\ \cdots\ \Delta\mathbf{u}^T_{k+N-1}}_{\text{unknown future } = \Delta\mathbf{u}}\ ]^T\end{aligned} \tag{20}$$

The set of the equations (19) considering the notation from (20) can be written as the usual generalized prediction form as follows:

$$\hat{\mathbf{y}}\ =\ \mathbf{f} + \mathbf{G}\Delta\mathbf{u} \tag{21}$$

where $\mathbf{f}$ represents the system response without changes of control actions, i.e. $\Delta\mathbf{u} = \mathbf{0}$:

$$\mathbf{f}\ =\ \mathbf{f}_{\mathbf{y}_k} + \mathbf{f}_{\Delta\mathbf{x}_k} + \mathbf{f}_{\Delta\tilde{\mathbf{u}}_{k-1}} \tag{22}$$

IV. Computation of MPC Actions

Model Predictive Control refers to a class of computer control algorithms that utilize an explicit process model to predict the future response of a controlled system. At each control interval, i.e. prediction horizon, a MPC algorithm attempts to optimize future system behaviour by computing a sequence of future control adjustments. The first input of the optimal control sequence is then sent to the system, and the entire calculation is repeated at subsequent control intervals [5]. It means that the prediction horizon of the optimization problem is shifted forward along the time axis and the procedure is repeated again [7], [8].

Computation of the MPC actions consists in the searching for optimal control actions, which minimize a quadratic cost function (23). In it, the future possible system outputs are substituted by their predictions (21) using the system model (4) or (9) [9], [15]:

$$J_k = \sum_{j=No+1}^{N} \left\| (\hat{\mathbf{y}}_{k+j} - \mathbf{w}_{k+j})\, \mathbf{Q}_\mathbf{y} \right\|^2 + \sum_{j=1}^{Nu} \left\| \mathbf{u}_{k+j-1}\, \mathbf{Q}_\mathbf{u} \right\|^2 \qquad (23)$$

or with increments of control actions:

$$J_k = \sum_{j=No+1}^{N} \left\| (\hat{\mathbf{y}}_{k+j} - \mathbf{w}_{k+j})\, \mathbf{Q}_\mathbf{y} \right\|^2 + \sum_{j=1}^{Nu} \left\| \Delta\mathbf{u}_{k+j-1}\, \mathbf{Q}_\mathbf{u} \right\|^2 \qquad (24)$$

where N, No and Nu are horizons of prediction, initial insensitivity and control respectively; $\mathbf{Q}_\mathbf{y}$ and $\mathbf{Q}_\mathbf{u}$ are weighting control parameters: output and input penalizations; and $\mathbf{w}_{k+j}$ are values of the reference signal [8]. The cost functions (23) or (24) can be written in a condensed notation

$$J_k = \mathbf{J}^T\mathbf{J} = (\hat{\mathbf{y}} - \mathbf{w})^T \mathbf{Q_y}^T \mathbf{Q_y} (\hat{\mathbf{y}} - \mathbf{w}) + \mathbf{u}^T \mathbf{Q_u}^T \mathbf{Q_u}\, \mathbf{u} \qquad (25)$$

It can be expressed in a matrix form

$$J_k = [(\hat{\mathbf{y}} - \mathbf{w})^T \;\; \mathbf{u}^T] \begin{bmatrix} \mathbf{Qy} & \mathbf{0} \\ \mathbf{0} & \mathbf{Qu} \end{bmatrix}^T \begin{bmatrix} \mathbf{Qy} & \mathbf{0} \\ \mathbf{0} & \mathbf{Qu} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{y}} - \mathbf{w} \\ \mathbf{u} \end{bmatrix} \qquad (26)$$

with respect to the predictions

$$\hat{\mathbf{y}} = \mathbf{f} + \mathbf{G}\mathbf{u} \qquad (27)$$

where $\hat{\mathbf{y}}$, $\mathbf{w}$ and $\mathbf{u}$ ($= \mathbf{u}$ or $\Delta\mathbf{u}$) are the vectors for the given prediction horizon N:

$$\begin{aligned} \hat{\mathbf{y}} &= [\hat{\mathbf{y}}_{k+1}^T, \cdots, \hat{\mathbf{y}}_{k+N}^T]^T \\ \mathbf{w} &= [\mathbf{w}_{k+1}^T, \cdots, \mathbf{w}_{k+N}^T]^T \\ \mathbf{u} &= [\mathbf{u}_{k}^T, \cdots, \mathbf{u}_{k+N-1}^T]^T \\ \Delta\mathbf{u} &= [\Delta\mathbf{u}_{k}^T, \cdots, \Delta\mathbf{u}_{k+N-1}^T]^T \end{aligned} \qquad (28)$$

The minimization of the cost of the function (25) can be provided in one shot as a solution of the least squares (LS) problem [12].

The LS solution is applied to the system of algebraic equations, which represents square root part of (26):

$$\min_{\mathbf{u}} \mathbf{J} = \begin{bmatrix} \mathbf{Qy} & \mathbf{0} \\ \mathbf{0} & \mathbf{Qu} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{y}} - \mathbf{w} \\ \mathbf{u} \end{bmatrix} = \mathbf{0}$$

$$\text{with respect to } \; \hat{\mathbf{y}} = \mathbf{f} + \mathbf{G}\mathbf{u}$$

$$\begin{bmatrix} \mathbf{Qy} & \mathbf{0} \\ \mathbf{0} & \mathbf{Qu} \end{bmatrix} \begin{bmatrix} \mathbf{G} & \mathbf{w} - \mathbf{f} \\ \mathbf{I} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ -\mathbf{I} \end{bmatrix} = \mathbf{0} \qquad (29)$$

Thus, after several modifications, the appropriate system of algebraic equations for the LS solution is:

$$\begin{bmatrix} \mathbf{Qy\,G} \\ \mathbf{Qu} \end{bmatrix} \mathbf{u} - \begin{bmatrix} \mathbf{Qy}\,(\mathbf{w} - \mathbf{f}) \\ \mathbf{0} \end{bmatrix} = \mathbf{0}$$

$$\mathbf{A} \quad \mathbf{u} - \quad \mathbf{b} \quad = \mathbf{0} \qquad (30)$$

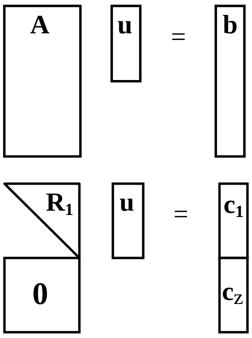
The equation system (30) can be efficiently evaluated by an orthogonal-triangular decomposition. Its appropriate algorithm is described in [13].

$$\mathbf{Q}^T \mathbf{A}\,\mathbf{u} = \mathbf{Q}^T \mathbf{b}$$

$$\text{with respect to } \; \mathbf{A} = \mathbf{Q}\,\mathbf{R}$$

$$\mathbf{R}_1 \;\; \mathbf{u} = \;\; \mathbf{c}_1 \qquad (31)$$

The orthogonal matrix $\mathbf{Q}^T$ transforms the system matrix $\mathbf{A}$ to the upper triangle $\mathbf{R}_1$ as indicated:



The vector $\mathbf{c}_z$ is a loss vector. Its Euclidean norm $\|\mathbf{c}_z\|$ equals to the square-root of the optimal cost function minimum, i.e. scalar $\sqrt{J}$, where $J = \mathbf{c}_z^T\,\mathbf{c}_z$.

Unknown control actions from the system (31) can be determined by a backward substitution. From the obtained vector $\mathbf{u}$, which represents the designed control actions or increments for the whole horizon N, only first appropriate action is really applied to the controlled system.

Considering the incremental algorithm, the real whole action is constructed as a sum of increments with the previous action vector as follows:

$$\mathbf{u}_k = \mathbf{u}_{k-1} + \Delta\mathbf{u}_k \qquad (32)$$

DOI 10.14311/TEE.2016.1.008

V. Comparative Examples

In this section, three comparative examples are discussed. The first one (Fig. 2) illustrates the standard MPC control without the offset-free property, i.e. the steady-stay offset (error) is obvious in the figure part $w(t)-y(t)$. The second and third examples (Fig. 3 – 5) show the application of the MPC with the offset-free reference tracking explained in sections III and IV. For experiments, the model of the second order system is considered in the continuous domain as the real system representative:

$$G_s = \frac{1}{s^2+2s+1} \tag{33}$$

and in the discrete domain for the MPC design:

$$G_{z^{-1}} = \left.\frac{0.0047+0.0044z^{-1}}{1-1.81z^{-1}+0.82z^{-2}}\right|_{Ts\,=\,0.1s} \tag{34}$$

including parameter variation to simulate the model mismatch. The MPC controller is set as follows:

$$N=10,\ \ N_u=10,\ \ No=0,$$

$$\mathbf{Q}_y=\mathbf{I},\ \ \mathbf{Q}_u=diag(\cdots 0.1\cdots)|_{stiff}$$

$$\text{or }\ \mathbf{Q}_u=diag(\cdots 4\cdots)|_{soft}$$

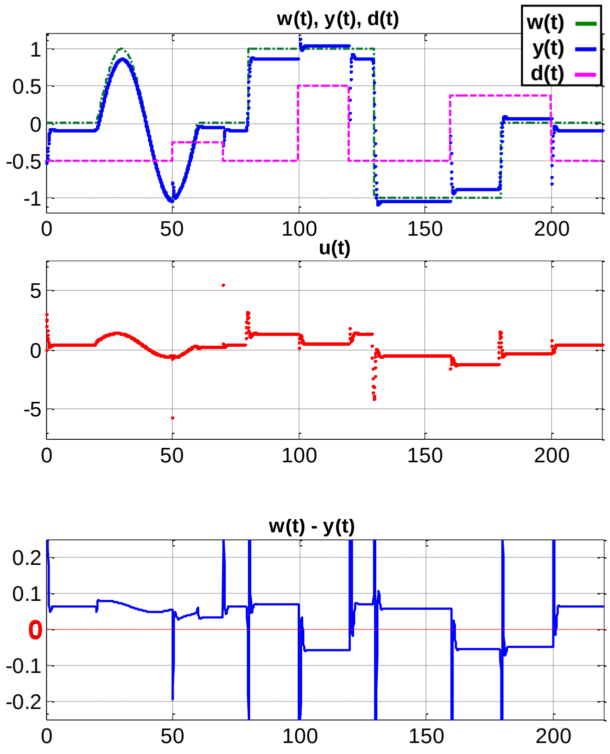


Fig. 2. Reference tracking with offset: MPC with stiff setting.

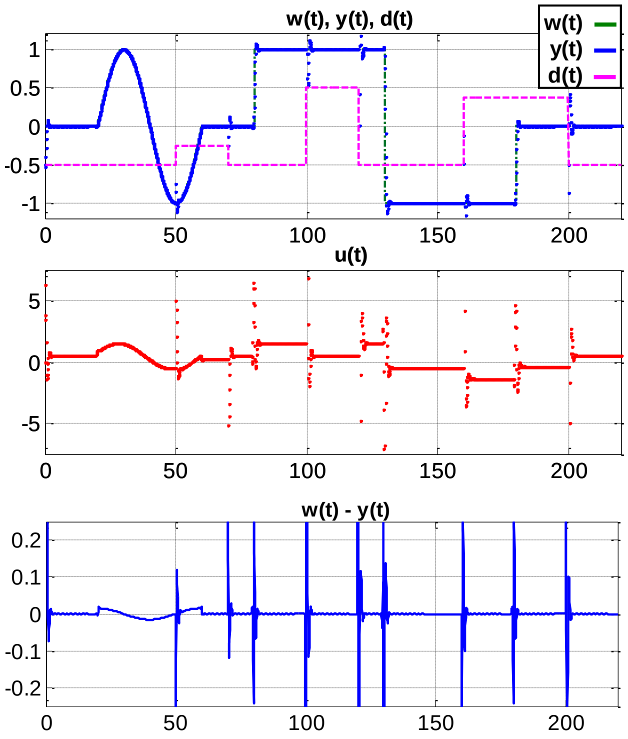


Fig. 3. Offset-free reference tracking: MPC with stiff setting.

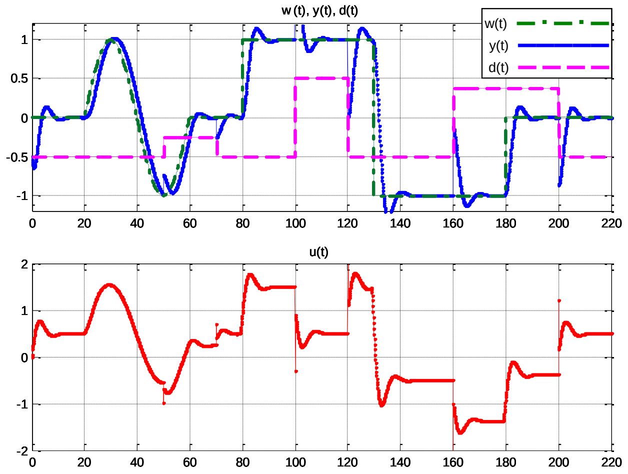


Fig. 4. Offset-free reference tracking with model mismatch and step disturbance $d(t)$: MPC with soft setting, detailed view of $w(t)$, $y(t)$, $u(t)$.

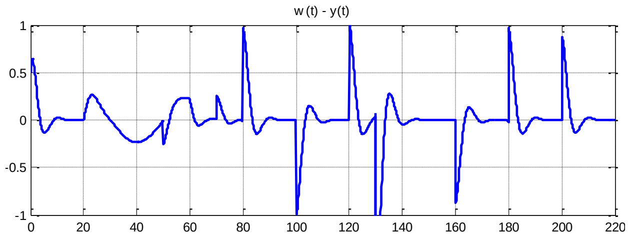


Fig. 5. Offset-free reference tracking with model mismatch and step disturbance $d(t)$: MPC with soft setting, detail of $e(t)=w(t)-y(t)$.

[DOI 10.14311/TEE.2016.1.008](https://doi.org/10.14311/TEE.2016.1.008)

In Fig. 4 and 5, the soft setting $\mathbf{Q}_u = diag(\cdots 4 \cdots)$ is used to show integrative behaviour of the MPC controller. The controller, as the other MPC or LQ controllers with some incremental algorithm, compensates naturally effects of penalizations in the cost function as in the artificial assumption with zero penalization of the system input and no disturbance. Several comparable examples for usual state-space models are shown, e.g., in [15], [16] and [17].

When the level of the constant disturbance $d(t)$ is changed (time instants 50 s, 70 s, 100 s, 120 s, 160 s and 200 s), the MPC controller reintegrates these changes and stabilizes itself in some new level of control, if however the reference signal does not change continuously as e.g. the sinusoidal shape at the beginning of the experiment: 20 s – 60 s. Speed of the controller response corresponds to the ratio of the $\mathbf{Q}_y$ and $\mathbf{Q}_u$, which determines the convergence of the error $e(t) = w(t) - y(t)$ to the zero level. Note, in Fig. 4, the dot and dash line is the reference signal; dashed line represents levels of the constant disturbance $d(t)$.

Due to step changes of the disturbance signal $d(t)$, the MPC controller generates isolated actions, which subside in several few time steps. These actions are shown as the isolated points in time records $u(t)$ in all appropriate figures. In transition responses, the different effect of the input penalization $\mathbf{Q}_u$ appears.

In the case of the offset MPC, the whole magnitude of $u(t)$ is penalized. On the other hand, in the case of the offset-free reference tracking MPC, only the increments $\Delta u(t)$ are penalized. In the latter case, the resultant magnitude of the control actions $u(t)$ is not limited, but only its increment or decrement respectively. Thus, the response to huge change may cause a huge action, but its damping may be slow and vibrating due to presence of an integrative property.

VI. Conclusion

In this paper, the offset-free problem of the Model Predictive Control was discussed for the case of the asymptotic tracking of the reference. The proposed solution arises from natural principle of the state-space formulation of the model predictive control, but the ARX form is used as initial model.

Except for using of the model obtained from a mathematical-physical analysis, the explained procedure can also be operated as an adaptive realization with some on-line identification. It does not need any additional variables or any disturbance models [3], [4]. The dimensionality of the design is the same as in the case of the conventional MPC realization without the offset-free property [6] in comparison with a conversion of the ARX model to some state-space form without any guarantee of the physical meaning of the state or with augmented states [16].

The solution is suitable just for mechatronic systems like robots and manipulators, where a deterministic behaviour is dominant. It uses information from the typical measurement of the system output and the record of the previous systems inputs and outputs, a number of which follows form order of the controlled system.

The proposed approach is suitable for systems with an adequate signal-to-noise ratio of the measured system output, as it is usual in mechatronic systems. A temporary noise increase can be solved via the on-line control parameter tuning [14].

The proposed solution was illustrated by comparative examples of the conventional (positional) MPC controller and MPC controller (incremental) with offset-free property.

References

[1] J.K. Huusom, N.K. Poulsen, S.B. Jørgensen, and J.B.Jørgensenb, "ARX-Model based model predictive control with offset-free tracking", in *Proc. 20th European Symposium on Computer Aided Process Engineering*, 2010, pp. 1–6.

[2] U. Maeder and M. Morari, "Offset-free reference tracking for predictive controllers", in *Proc. 46th IEEE Conference on Decision and Control*, New Orleans, LA, USA, 2007, pp. 5252–5257.

[3] U. Maeder, F., Borrelli and M. Morari, "Linear offset-free model predictive control", in *Automatica* 45, 2009, pp. 2214–2222.

[4] G. Pannocchia and J.B. Rawlings, "Disturbance models for offset-free model predictive control", in *AIChE Journal*, 2003, 49, (2), pp. 426–437. http://dx.doi.org/10.1002/aic.690490213

[5] S.J. Qin and T.A. Badgwell, "A survey of industrial model predictive control technology", in *Control Engineering Practice*, 2003, 11, pp. 733–764. http://dx.doi.org/10.1016/S0967-0661(02)00186-7

[6] M. Valášek, K. Belda and M. Florian, "Control and calibration of redundantly actuated parallel robots", in *Proc. 2nd Int. Conf. Parallel Kinematic Machines*, Germany, 2002, pp. 411–427.

[7] J.A. Rossiter, Model-based predictive control a practical approach, CRC Press, 2003.

[8] J.M. Maciejowski, *Predictive control with constrains*, Prentice Hall, 2002.

[9] A. Ordys and D. Clarke, "A state - space description for GPC controllers", in *Int. J. Systems SCI.*, 1993, 24, (9), pp. 1727–1744. http://dx.doi.org/10.1080/00207729308949590

[10] K. Belda and J. Böhm, "Adaptive Generalized Predictive Control for Mechatronic Systems", *WSEAS Transactions on Systems*, 5(8), Greece 2006, pp. 1830-1837.

[11] E. Mosca, Optimal, predictive and adaptive control, Prentice Hall, 1995.

[12] Ch.L. Lawson and R.J. Hanson, *Solving least squares problems*, Siam, Prentice-Hall, 1995. http://dx.doi.org/10.1137/1.9781611971217

[13] H.G. Golub and Ch.F.L. Van, *Matrix computations*, The Johns Hopkins Univ. Press, 1989.

[14] K. Belda, "On-line Parameter Tuning of Model Predictive Control", in *Preprints of the 18th IFAC World Congress*, Milano, Italy 2011, pp. 5489–5494.

[15] K. Belda and D. Vošmik, "Speed Control of PMSM Drives by Generalized Predictive Algorithms", Proc. of *IECON 2012*, Canada, pp. 2002-2007.

[16] L. Wang, Model Predictive Control System Design and Implementation Using MATLAB®, Springer, 2009.

[17] K. Belda, and D. Vošmik, "Explicit Generalized Predictive Algorithms for Speed Control of PMSM Drives". *Proc. of IECON 2013*, Austria, pp. 2830-2835. http://dx.doi.org/10.1109/iecon.2013.6699580

[18] K. Belda, "On Offset Free Generalized Predictive Control", *Proc. of Conf. on Process Control 2013*. Slovakia, pp. 395-400. http://dx.doi.org/10.1109/pc.2013.6581443

TELEN2016001
DOI 10.14311/TEE.2016.1.008