00040

The Open University of Sri Lanka
Department of Electrical and Computer Engineering
ECX6242 – Modern Control Systems
Final Examination – 2014/2015

Date: 2015-09-06 Time:0930-1230

Answer **five** questions by selecting **at least two questions** from each of the **sections A** and **B**.

Section A

Select at least **two** questions from this section.

Q1.

(a) Describe controllability and observability. [6]

(b) Consider the system represented by the equation

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ -10 & -7 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t)$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} x(t)$$

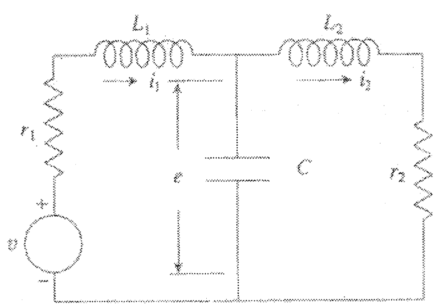
Find the transfer function of the system. [8]

(c) Check for the controllability of the system. [6]

Q2.

(a) Outline the Laplace transform method of determining the state transition matrix that is required in the solution of the state equation. [6]

(b) Develop the state equation and the output equation of the circuit shown in following figure. [14]



Q3.

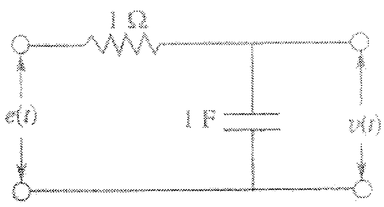
(a) Briefly describe Lyapunov's method for the determination of the stability of non-linear systems. [6]

(b) If $\frac{dx_1}{dt} = ax_1 + bx_2$and $\frac{dx_2}{dt} = cx_1 + dx_2$, Determine the sufficient conditions on a, b, c and d so that the asymptotically stable condition can be achieved. Choose $W = x_1^2 + x_2^2$ to apply the Lyapunov's theory. [14]

Q4.

(a) Define the stability of discrete control systems and explain the Jury's test of stability. [6]

(b) Find the output voltage in discrete form of the RC circuit in the following figure when the input voltage is applied as follows.



$e(t) = e(nT)$ where $nT \leq t \leq (n+1)T$ and $T = 1\ s$ [14]

Section B

Select at least **two** questions from this section. [20 Marks for each]

The questions in this section are based on the research paper reproduced at the end of this question paper. Devote at least half an hour to reading through the paper. Use your own words in your answers so as to demonstrate that you have understood the concepts described in the paper, do not copy extracts from the paper itself.

Q5. Explain what is a predictive control system is.

Q6. Discuss two major properties of non-linear control systems.

Q7. Compare unconstrained and constrained minimization process.

Q8. Explain the proposed methodology in the article briefly in your words.

Note:

Laplace transform	Corresponding z-transform
$\frac{1}{s}$	$\frac{z}{z-1}$
$\frac{1}{s^2}$	$\frac{Tz}{(z-1)^2}$
$\frac{1}{s^3}$	$\frac{T^2 z(z+1)}{2(z-1)^3}$
$\frac{1}{s+a}$	$\frac{z}{z-e^{-aT}}$
$\frac{1}{(s+a)^2}$	$\frac{Tze^{-aT}}{(z-e^{-aT})^2}$
$\frac{a}{s(s+a)}$	$\frac{z(1-e^{-aT})}{(z-1)(z-e^{-aT})}$
$\frac{b-a}{(s+a)(s+b)}$	$\frac{z(e^{-aT}-e^{-bT})}{(z-e^{-aT})(z-e^{-bT})}$
$\frac{(b-a)s}{(s+a)(s+b)}$	$\frac{(b-a)z^2-(be^{-aT}-ae^{-bT})z}{(z-e^{-aT})(z-e^{-bT})}$
$\frac{a}{s^2+a^2}$	$\frac{z \sin aT}{z^2-2z\cos aT+1}$
$\frac{s}{s^2+a^2}$	$\frac{z^2-z\cos aT}{z^2-2z\cos aT+1}$
$\frac{s}{(s+a)^2}$	$\frac{z[z-e^{-aT}(1+aT)]}{(z-e^{-aT})^2}$

Model Predictive Control with Nonlinear State Space Models

Martin Rau and Dierk Schröder
Technical University Munich
Institute for Electrical Drive Systems
Arcisstraße 21, D-80333 München, Germany.
Martin.Rau@ei.tum.de

Abstract: This paper presents a model predictive control scheme based on nonlinear state space models. The considered class of systems is supposed to be separable into a linear part and a nonlinear feedback path. Therefore, the overall discrete–time dynamic system is nonlinear. Most of the existing model predictive control algorithms for nonlinear systems require the solution of a non–convex nonlinear optimization problem within the interval of one sample time step. This seems to be practical impossible in systems with fast sample rates as they occur in electrical drive systems.

In order to facilitate the predictive control algorithm for real–time applications, the nonlinear feedback path is linearized along a reference trajectory within the prediction horizon. This results in a linear time–variant model, where the nonlinearity is mapped to the time variance of the model. The trajectory for linearization can either be the reference trajectory in the prediction horizon or must be generated based on other available information of the system. The prediction j steps ahead and the control law in analogy to generalized predictive control can be calculated analytically in absence of constraints. However, the system's nonlinearity is taken into account by the linearization along a trajectory at every integration and prediction step. The inclusion of constraints in the optimization problem results in a quadratic program for which efficient solution methods exist. This leads to a computationally more practical predictive control concept for nonlinear systems applicable to fast processes even in presence of constraints.

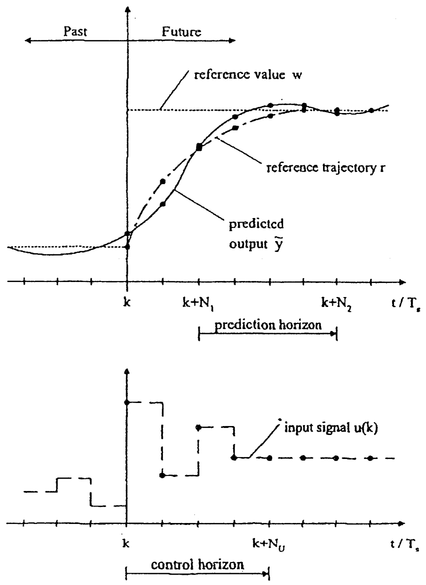


Fig.1: Principle of model based predictive control.

1 Introduction

Model predictive control is a widely used control concept for over 15 years especially in the process industry [1, 2, 3, 4]. Applications in the field of electrical drives are quite rare [6]. The core of a model predictive controller is a process model. Any process model, capable of predicting future output signals based on future input signals and initial values, can be used. With this process model, the future dynamic behavior of the real plant is predicted within a prediction horizon N_2. These predicted output signals are used to minimize an open loop performance criterion (e.g. the sum of squared control errors within the prediction horizon) and to calculate the input signals $u(k)$ within a control horizon N_u. Outside the control horizon, the input signal $u(k)$ remains constant. The calculated input signals are fed into the plant until a new measurement gets available. This procedure is repeated with a new prediction and control horizon and is called receeding horizon control. The receeding horizon strategy makes a closed loop control law from the originally open loop minimization. The minimization step can easily include constraints, such that input, output or state constraints can be taken into account already in the controller design. The principle of model based predictive control (MPC) is depicted in fig. 1. The most famous representative MPC is the *generalized predictive controller* (GPC) of [1, 2], which is based on a linear, discrete-time transfer function model. A state-space representation of GPC can be found in [8]. Both can be used for linear, time-invariant processes only. One main advantage is the analytical solution of the minimization problem in absence of constraints. In case of constraints, GPC results in a quadratic program, for wich very efficient and fast numerical algorithms exist.

For high precision applications, nonlinear process models have to be used for prediction [5]. Generally, this results in a non-convex nonlinear program, which is difficult to solve due to the following reasons: The computational expense for a nonlinear program is much higher than that of a quadratic program. This restricts the application of nonlinear process models to relatively slow processes. A second drawback is the fact, that non-convex nonlinear programs have several local minima. Therefore, methods for obtaining the global minimum have to be applied (which increases the computational expense even more).

This paper introduces a model predictive control concept, which combines the advantages of both, linear an nonlinear process models for prediction. The process model is assumed to be separable into a linear dynamic part with a nonlinear feedback path. This type of process model is called system with isolated nonlinearity. As in all predictive control concepts, the reference trajectory is known within the prediction horizon. The nonlinear feedback path is linearized along this trajectory. The resulting system is linear, but time-variant and is used as a prediction model. Its accuracy is better than that of a linearized model around one single point of operation, because the nonlinearity is taken into account along the whole trajectory. The computational expense of the linear time-variant model is almost the same as for a linear time-invariant model. Without considering constraints, the minimization problem can be solved analytically and no numerical optimization algorithms need to be applied. When including constraints, the resulting optimization problem is a quadratic program and can be solved with the same efficient algorithms as GPC. In this paper, the higher accuracy of a nonlinear process model is combined with the ability to apply quadratic programming techniques for online optimization.

2 Process Model

Throughout this paper, we will consider single-input signal-output (SISO) systems. An extension to systems with multiple in- or outputs is possible without an increase of complexity. The SISO system under consideration is described by a nonlinear discrete time state space model of degree N with an isolated nonlinearity $\mathcal{NL}$.

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{A}\cdot\mathbf{x}(k)+\mathbf{b}\cdot\Delta u(k)+\mathbf{k}\cdot\mathcal{NL}(y(k)) \quad (1)\\ y(k) &= \mathbf{c}^T\cdot\mathbf{x}(k)+d\cdot\Delta u(k) \end{aligned}$$

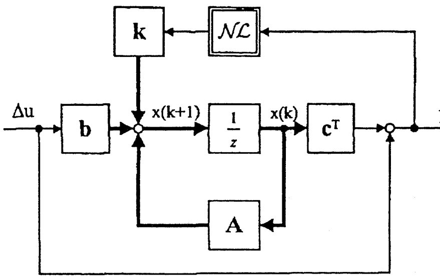


Fig.2: Signal flow chart of the considered system.

A signal flow chart representation of the model is shown in fig. 2. The system matrices $\mathbf{A}$, $\mathbf{b}$, $\mathbf{c}$, d and $\mathbf{k}$ are constant and of appropriate dimensions. The vector $\mathbf{k}$ describes the coupling of the nonlinearity into the system. Δu denotes the increment of the input signal between two sampling instants. Any state-space system with an input signal u (instead of Δu) can be transformed to equation (1) by adding the additional state variable $u(k-1)$. The difference operator Δ is defined as $\Delta = 1 - z^{-1}$ (z^{-1} being the one-step backward shift operator). The system matrices have to be known, the state variables are assumed to be measurable and the nonlinearity $\mathcal{NL}$ may be unknown. If the nonlinearity and the state variables are not available, a combined observer and identificator for this class of system can be applied [10]. For all further calculations, we assume that the nonlinearity is known and the observed and real states are identical. In order to take advantage of a simplified calculation of the control law, the system model in equation (1) is linearized along the known reference trajectory $r(k)$ of the output signal $y(k)$. The reference trajectory is the desired output signal of the system; it has to be known within the prediction horizon from time step k to $k+N_2$, where N_2 is the upper prediction horizon. The linearization of the isolated nonlinearity along the reference trajectory $r(k)$ gives

$$\begin{aligned} \mathcal{NL}(y(k)) &\approx \mathcal{NL}(r(k)) + \left.\frac{d\mathcal{NL}}{dy}\right|_{y=r(k)} \cdot (y(k)-r(k)) \\ &= \mathcal{NL}(r(k)) + \left.\frac{d\mathcal{NL}}{dy}\right|_{y=r(k)} \cdot (\mathbf{c}^T\cdot\mathbf{x}(k) \\ &\quad +d\cdot\Delta u(k) - r(k)) \qquad (2) \end{aligned}$$

This approximation results in a simplified linear, but time variant model of the system in equation (1). This method differs from an ordinary linearization around a fixed operation point by the fact, that the reference trajectory (which must be known in any predictive controller) is the basis for linearization, and therefore the approximation takes into account the nonlinearity within the prediction horizon. The resulting linear time variant model is

$$\mathbf{x}(k+1) = \mathbf{A}(k)\cdot\mathbf{x}(k)+\mathbf{b}(k)\cdot\Delta u(k)+\mathbf{k}\cdot v(k) \quad (3)$$

with

$$\mathbf{A}(k) = \mathbf{A} + \mathbf{k}\,\mathbf{c}^T \cdot \left.\frac{d\mathcal{NL}}{dy}\right|_{y=r(k)} \quad (4)$$

$$\mathbf{b}(k) = \mathbf{b} + \mathbf{k} \cdot d \cdot \left.\frac{d\mathcal{NL}}{dy}\right|_{y=r(k)} \quad (5)$$

$$v(k) = \mathcal{NL}(r(k)) - r(k) \cdot \left.\frac{d\mathcal{NL}}{dy}\right|_{y=r(k)} \quad (6)$$

The output equation of (1) is not changed by the linearization. When driving the system along the reference trajectory, the nonlinearity has the same effect as a time variant disturbance signal $v(k)$. This implies, that the controller to develop is able to drive the system near the reference trajectory and does not allow large deviations. The prediction of future system states and output signals is performed with the model in equation (3), where the values of $\mathbf{A}(k)$, $\mathbf{b}(k)$ and $v(k)$ are all known within the prediction horizon; they depend only on the reference trajectory $r(k)$.

3 Predictive Control Law

In this section, we derive the predictive control laws for the unconstrained and constrained case. The performance index of the predictive controller is a weighted sum of squared control errors and control moves [1]. It is chosen to

$$J = \sum_{j=N_1}^{N_2} (r(k+j) - \tilde{y}(k+j))^2 + \lambda \cdot \sum_{j=0}^{N_u} (\Delta u(k+j))^2 \quad (7)$$

Variables marked by $\tilde{}$ are predicted values. In equation (7), k is the current time step. The upper prediction horizon N_2 should be chosen such, that the dominating time responses lie within this horizon. With the lower horizon N_1, it is possible to allow control errors at the beginning of the horizon and to penalize them between N_1 and N_2 only. The control horizon N_u indicates the number of allowed control moves within the horizon. After N_u control moves, the system input Δu is zero (u is constant). This is a common measure to reduce the computational expense, although only a suboptimal solution is found [1, 8]. The weighting factor λ adjusts the relation between the weighting of the control errors and the control moves. The higher λ is chosen, the slower will be the resulting controller.

In order to minimize the cost function (7), future system outputs are required. They are not available but can be predicted based on the system model in equation (3). A j-step ahead predictor is derived by continuing equation (3):

$$\begin{aligned}
\tilde{y}(k+1) &= \mathbf{c}^T \cdot (\mathbf{A}(k) \cdot \mathbf{x}(k) + \mathbf{b}(k) \cdot \Delta u(k) && (8)\\
&+ \mathbf{k} \cdot v(k)) + d\Delta u(k+1) \\
\tilde{y}(k+2) &= \mathbf{c}^T (\mathbf{A}(k+1)\mathbf{A}(k)\mathbf{x}(k) && (9)\\
&+ \mathbf{A}(k+1)\mathbf{b}(k)\Delta u(k) + \mathbf{b}(k+1)\Delta u(k+1) \\
&+ \mathbf{A}(k+1)\mathbf{k}v(k) + \mathbf{k}v(k+1)) + d\Delta u(k+2) \\
\tilde{y}(k+3) &= \mathbf{c}^T (\mathbf{A}(k+2)\mathbf{A}(k+1)\mathbf{A}(k)\mathbf{x}(k) && (10)\\
&+ \mathbf{A}(k+2)\mathbf{A}(k+1)\mathbf{b}(k)\Delta u(k) \\
&+ \mathbf{A}(k+2)\mathbf{b}(k+1)\Delta u(k+1) \\
&+ \mathbf{b}(k+2)\Delta u(k+2) \\
&+ \mathbf{A}(k+2)\mathbf{A}(k+1)\mathbf{k}v(k) \\
&+ \mathbf{A}(k+2)\mathbf{k}v(k+1) + \mathbf{k}v(k+2)) \\
&+ d\Delta u(k+3)
\end{aligned}$$

This scheme can be continued to the upper prediction horizon N_2, i.e. $y(k+N_2)$. In order to facilitate notation, it is convenient to define the following vectors containing future signals.

$$\begin{aligned}
\tilde{\mathbf{y}} &= [\, y(k+N_1) \ \ldots \ y(k+N_2) \,] && (11)\\
\Delta\mathbf{u} &= [\, \Delta u(k) \ \ldots \ \Delta u(k+N_u) \,] && (12)\\
\mathbf{v} &= [\, v(k) \ \ldots \ v(k+N_2-1) \,] && (13)\\
\mathbf{r} &= [\, r(k+N_1) \ \ldots \ r(k+N_2) \,] && (14)
\end{aligned}$$

where $\mathbf{v}$ and $\mathbf{r}$ are known in advance, since they only depend on the reference trajectory. The predicted output signals $\tilde{\mathbf{y}}$ are now expressed in matrix-vector form:

$$\mathbf{y} = \mathbf{F} \cdot \mathbf{x}(k) + \mathbf{H} \cdot \Delta\mathbf{u} + \mathbf{G} \cdot \mathbf{v} \quad (15)$$

The matrices $\mathbf{F}$, $\mathbf{H}$ and $\mathbf{G}$ are derived from equations (8) to (10). They are formed according to the following rules:

$$\mathbf{F} = \begin{bmatrix} \mathbf{c}^T \prod_{n=N_1-1}^{0} \mathbf{A}(k+n) \\ \mathbf{c}^T \prod_{n=N_1}^{0} \mathbf{A}(k+n) \\ \vdots \\ \mathbf{c}^T \prod_{n=N_2-1}^{0} \mathbf{A}(k+n) \end{bmatrix} \quad (16)$$

$$\mathbf{H} = \begin{bmatrix} h_{N_1-1,N_1-1} & \cdots & h_{N_1-1,N_1-N_u} \\ \vdots & \ddots & \vdots \\ h_{N_2-1,N_2-1} & \cdots & h_{N_2-1,N_2-N_u} \end{bmatrix} \quad (17)$$

The elements of $\mathbf{H}$ are:

$$h_{i,j} = \begin{cases} \mathbf{c}^T \left[\prod_{n=i}^{i-j+1} \mathbf{A}(k+n)\right] \mathbf{b}(k+i-j) & : j > 0 \\ \mathbf{c}^T \mathbf{b}(k+i-j) & : j = 0 \\ d & : j = -1 \\ 0 & : j < -1 \end{cases} \quad (18)$$

The matrix $\mathbf{G}$ contains the effects of the nonlinearity and is defined by:

$$\mathbf{G} = \begin{bmatrix} g_{N_1-1,N_1-1} & \cdots & g_{N_1-1,N_1-N_2} \\ \vdots & \ddots & \vdots \\ g_{N_2-1,N_2-1} & \cdots & g_{N_2-1,N_2-N_2} \end{bmatrix} \quad (19)$$

Note, that the dimensions of **H** and **G** are different, since the control horizon N_u has only effect on **H**. The elements of **G** are:

$$g_{i,j} = \begin{cases} \mathbf{c}^T \left[\prod_{n=i}^{i-j+1} \mathbf{A}(k+n) \right] \mathbf{k} & : j > 0 \\ \mathbf{c}^T \mathbf{k} & : j = 0 \\ 0 & : j < 0 \end{cases} \quad (20)$$

With the order N of the system in equation (3), the matrices have the following dimensions:

$$\mathbf{F} \in \mathbb{R}^{N_2-N_1+1\times N} \quad \mathbf{H} \in \mathbb{R}^{N_2-N_1+1\times N_u} \quad (21)$$
$$\mathbf{G} \in \mathbb{R}^{N_2-N_1+1\times N_2} \quad (22)$$

The cost function of equation (7) is now rewritten in matrix-vector notation and the minimization problem is stated. Using the notation of equations (15) to (19), the cost function is

$$\begin{aligned} J &= (\mathbf{r} - \mathbf{F}\mathbf{x}(k) - \mathbf{H}\Delta\mathbf{u} - \mathbf{G}\mathbf{v})^T (\mathbf{r} - \mathbf{F}\mathbf{x}(k) \\ &- \mathbf{H}\Delta\mathbf{u} - \mathbf{G}\mathbf{v}) + \lambda \Delta\mathbf{u}^T \Delta\mathbf{u} \end{aligned} \quad (23)$$

The solution of minimizing equation (23) gives the vector of control actions $\Delta\mathbf{u}$. The first element of $\Delta\mathbf{u}$ is used as input signal for the real process, all other elements are not used for control, but can serve as initial values for the next optimization run. The minimization of (23) and the calculation of the necessary matrices is repeated at every integration step. The minimization procedure itself depends on whether constraints are considered or not.

3.1 Unconstrained Minimization

In absence of constraints, the minimum of J can be calculated analytically. By setting the gradient of J to zero

$$\frac{\partial J(\Delta\mathbf{u})}{\partial \Delta\mathbf{u}} = 0 \quad (24)$$

and solving the resulting linear equation, the optimal solution for $\Delta\mathbf{u}$ is

$$\Delta\mathbf{u} = \left(\mathbf{H}^T\mathbf{H} + \lambda\mathbf{I}\right)^{-1} \mathbf{H}^T (\mathbf{r} - \mathbf{F}\mathbf{x}(k) - \mathbf{G}\mathbf{v}) \quad (25)$$

It can easily be shown, that the matrix $\mathbf{H}^T\mathbf{H} + \lambda\mathbf{I}$ (which is the quadratic term of (23)) is symmetric and positive definite for any positive λ, which implies, that the inverse always exists and that the optimum is a unique minimum. The first element of $\Delta\mathbf{u}$ is used as the input signal for the process. The following steps have to be repeated at every integration step: Calculation of the matrices **F**, **H** and **G**, minimization by evaluating equation (25) and extracting the first element of $\Delta\mathbf{u}$. The computational burden compared to linear time-invariant systems is only increased by the recalculation of the matrices **F** to **G** due to the time-variance of the prediction model.

The controller parameters N_1, N_2, N_u and λ have to be adjusted according to the dominant time constants of the process and the desired speed of the closed loop dynamics. Stability is not guaranteed for every choice of these parameters [11]; for stability for any choice of the controller parameters, the infinite horizon predictive control concept [12] may be adopted.

3.2 Constrained Minimization

When considering constraints on the control signals $\Delta u(k)$, $u(k)$ and the states $\mathbf{x}(k)$, the cost function (23) remains the same. This paper only deals with constraints of the input signal u and Δu, but state constraints can be taken into account in a similar way. Input constraints are divided in two types of inequalities: one for constraints on control increments Δu and one for the resulting input signal u. The control increments may not exceed a certain minimal and maximal value as defined by equation (26).

$$\Delta u_{min} \le \Delta u(k+j) \le \Delta u_{max} \quad \forall\ j = 0 \ldots N_u \quad (26)$$

Equation (26) has to be valid for all time steps inside the control horizon. The bounds on Δu can be combined in the following linear inequalities with $\mathbf{N}_{\Delta u} = \mathbf{I}$:

$$\mathbf{N}_{\Delta u} \cdot \Delta\mathbf{u} \le \mathbf{g}^u_{\Delta u} \quad (27)$$
$$-\mathbf{N}_{\Delta u} \cdot \Delta\mathbf{u} \le \mathbf{g}^l_{\Delta u} \quad (28)$$

The vectors $\mathbf{g}^u_{\Delta u}$ and $\mathbf{g}^l_{\Delta u}$ are defined as

$$\mathbf{g}^u_{\Delta u} = [\Delta u_{max} \ldots \Delta u_{max}]^T \quad (29)$$
$$\mathbf{g}^i_{\Delta u} = [-\Delta u_{min} \ldots - \Delta u_{min}]^T \quad (30)$$

In every practical application, the input signal u is also limited due to actuator saturation. This type of constraint is given by

$$u_{min} \le u(k+j) \le u_{max} \quad \forall\ j = 0 \ldots N_u \quad (31)$$

Since the free variable of the cost function (23) is Δu, equation (31) has to be transformed into a linear inequality in Δu. The input signal $u(k+j)$ can be expressed as

$$u(k+j) = u(k-1) + \sum_{i=0}^{j} \Delta u(k+i) \quad (32)$$

where the value $u(k-1)$ is known at time step k. The inequalities of equation (31) can be rewritten in the optimization variable Δu.

$$\sum_{i=0}^{j} \Delta u(k+i) \le u_{max} - u(k-1) \quad (33)$$

$$-\sum_{i=0}^{j} \Delta u(k+i) \le -u_{min} - u(k-1) \quad (34)$$

Equations (33) and (34) have to be fulfilled in the control horizon for $j = 0 \ldots N_u$ and are again transformed into a linear system of inequalities.

$$\mathbf{N}_u \cdot \Delta\mathbf{u} \le \mathbf{g}^u_u \quad (35)$$
$$-\mathbf{N}_u \cdot \Delta\mathbf{u} \le \mathbf{g}^l_u \quad (36)$$

with the following definitions:

$$\mathbf{N}_u = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 1 & 1 & & \vdots \\ \vdots & \vdots & \ddots & 0 \\ 1 & 1 & \cdots & 1 \end{bmatrix} \quad (37)$$

$$\mathbf{g}_u^u = [u_{max}-u(k-1)\ldots u_{max}-u(k-1)]^T \quad (38)$$

$$\mathbf{g}_u^l = [-u_{min}+u(k-1)\ldots -u_{min}+u(k-1)]^T$$

Inequalities (27), (28), (35) and (36) can be combined into one inequality constraint, such that the following quadratic program has to be solved:

$$\min_{\Delta u} J \quad \text{with} \quad \mathbf{N}\cdot\Delta\mathbf{u} < \mathbf{g} \quad (39)$$

N and g contain all $\mathbf{N}_i$-matrices, respectively all $\mathbf{g}_i^j$-vectors. The quadratic program in equation (39) has to be solved a every integration step with efficient numerical algorithms [13, 14].

By linearizing the isolated nonlinearity along the reference trajectory, it was possible to use a accurate, linear time-variant prediction model and to reduce the resulting optimization problem in presence of constraints to a quadratic program. This fact makes the proposed predictive control method a useful tool for real-time control of fast processes, where nonlinear programming techniques are not possible.

4 Example: Two-Mass System

The following short example, shows some simulation results for a typical mechatronic drive system. A rotating two mass system with nonlinear friction characteristic is investigated. Note, that the emphasis is not on a sophisticated friction model, but on the predictive control concept itself. Results for a standard PI-controller and the proposed predictive control concept without constraints are presented to show the improved performance. The system is described by the following continuous time state space equation:

$$\dot{\mathbf{x}} = \underbrace{\begin{bmatrix} -\frac{d}{J_1} & -\frac{c}{J_1} & \frac{d}{J_1} \\ 1 & 0 & -1 \\ \frac{d}{J_2} & \frac{c}{J_2} & -\frac{d}{J_2} \end{bmatrix}}_{\mathbf{A}} \cdot\mathbf{x} + \underbrace{\begin{bmatrix} \frac{1}{J_1} \\ 0 \\ 0 \end{bmatrix}}_{\mathbf{b}} \cdot u \quad (40)$$

$$+ \underbrace{\begin{bmatrix} 0 \\ 0 \\ -\frac{1}{J_2} \end{bmatrix}}_{\mathbf{k}} \cdot \underbrace{6.4\cdot\arctan(10\cdot x_3)}_{\mathcal{NL}(x_3)}$$

$$y = \underbrace{\begin{bmatrix} 0 & 0 & 1 \end{bmatrix}}_{\mathbf{c}^T}\cdot\mathbf{x} \quad (41)$$

The system parameters are: $J_1 = 0.166$ [kgm^2], $J_2 = 0.33$ [kgm^2], $c = 400$ [Nm/rad] and $d = 0.011$ [Nms/rad]. The corresponding discrete time model in the form of equation (1) is achieved by discretization with the zero-order-hold method. The nonlinearity is a simple model for a stick–slip friction characteristic. Its dimension is such, that the effect on the output signal y is significant. The reference trajectory is a sine wave; this may be a periodic positioning procedure. Fig. 3 shows the reference signal and the output signal, when the system is controlled by a standard PI–controller. The negative effect of the friction torque is especially visible when the output crosses zero (stick–slip).

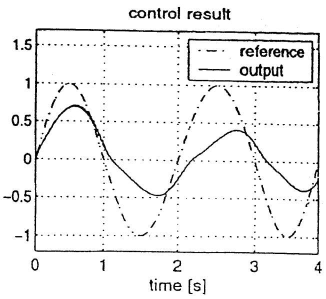


Fig.3: Control result with a standard PI–controller

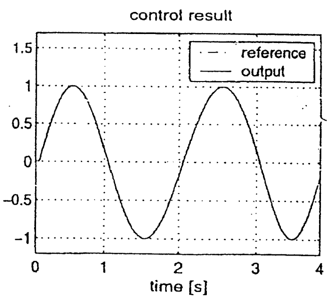


Fig.4: Control result with the model predictive controller.

When the same system is controlled by the proposed model predictive control scheme without respecting constraints, the nonlinearity is taken into account in the controller and therefore, the effect of the nonlinearity is reduced significantly. Fig. 4 shows the reference trajectory and the output signal, fig. 5 shows the corresponding unconstrained input torque. The output signal is very close to the reference trajectory and the accuracy is improved by magnitudes.

Now we suppose, that the input torque is limited to ±10 [Nm]. This constraint is taken into account at each integration step by solving a quadratic program. Fig. 6 shows the reference trajectory and the output signal, fig. 7 shows the corresponding constrained input torque. The output signal is still very close to the reference trajectory and the control signal fulfills the desired constraint. Other constraints for state variables and/or

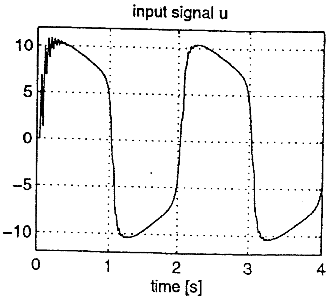


Fig.5: Input signal generated by the model predictive controller.

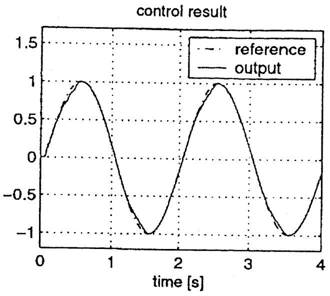


Fig.6: Control result with the constrained model predictive controller.

the output signal, e.g. overshoot constraints, could be introduced to improve the dynamic behavior.

5 Conclusion

The proposed nonlinear model predictive control concept is able to take into account an isolated nonlinearity by linearizing along the reference trajectory in the prediction horizon. This is more accurate than a linearization only at each integration step. The main property of linear model predictive control, the quadratic optimization problem, is preserved. This is a great computational advantage when including constraints in the optimization procedure. The motivation for this control concept is the fact, that the isolated nonlinearity is taken into account in the controller and the optimization problem remains easy to solve (analytically or by a quadratic program). The proposed method is practically relevant due to the reduced computation compared to nonlinear model predictive controllers.

References

[1] Clarke, D. W., Mohtadi, C., Tuffs, P. S.: *Generalized Predictive Control - Part I. The Basic Algorithm.* Automatica, Vol 23, No. 2, pp. 137–148, 1987.

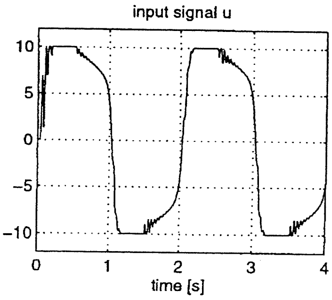


Fig.7: Input signal generated by the constrained model predictive controller.

[2] Clarke, D. W., Mohtadi, C., Tuffs, P. S.: *Generalized Predictive Control – Part II. Extensions and Interpretations.* Automatica, Vol 23, No. 2, pp. 149–160, 1987.

[3] Garcia, C. E., Prett, D. M., Morari, M.: *Model Predictive Control: Theory and Practice – a Survey.* Automatica, Vol 25, No. 3, pp. 335–348, 1989.

[4] Camacho, E. F., Bordons, C.: *Model Predictive Control.* Springer Verlag, London Berlin New York, 1999.

[5] Mayne, D. Q., Michalska, H.: *Receding Horizon Control of Nonlinear Systems.* IEEE Trans. on Automatic Control, Vol. 35, No. 7, pp. 814-824, 1990.

[6] Kuntze, H., Jacubasch, A., Richalet, J.: *On the Application of a New Method for Fast and Robust Position Control of Industrial Robots.* IEEE Int. Conf. on Robotics and Automation, Philadelphia, 1988.

[7] Chen, H., Allgöwer, F.: *A Quasi-Infinite Horizon Nonlinear Model Predictive Control Scheme with Guaranteed Stability.* Automatica, Vol 34, No. 10, pp. 1205–1217, 1998.

[8] Kraus, P., Daß, K., Rake, H.: *Model Based Predictive Controller with Kalman filtering for State Estimation.* In *Advances in Model Based Predictive Control.* Oxford University Press, pp. 69–83, 1994.

[9] Gracia, C. E.: *Quadratic Dynamic Matrix Control of Nonlinear Processes. An Application to a Batch Reaction Process.* AIChE, Annual Meeting, San Francisco, 1984.

[10] Schröder, D.: *Intelligent Observer and Control Design for Nonlinear Systems.* Springer Verlag, Berlin Heidelberg New York, 2000.

[11] Bitmead, R. R., Gevers, M., Wertz, V.: *Adaptive Optimal Control – The Thinking Man's GPC.* Prentice Hall, New York, 1990.

[12] Rawlings, J. B., Muske, K. R.: *The Stability of constrained receding horizon control.* IEEE Trans. on Automatic Control, AC-38 (10), pp. 1512–1516, 1993.

[13] Coleman, T. F., Li, Y.: *A Reflective Newton Method for Minimizing a Quadratic Function Subject to Bounds on some of the Variables.* SIAM Journal on Optimization, Vol. 6, Number 4, pp. 1040-1058, 1996.

[14] The Mathworks: *Optimization Toolbox – User's Guide.* The Mathworks, 2000.