Ročník 2011 electroscope Číslo II

MODEL PREDICTIVE CURRENT CONTROLLER OF INDUCTION MACHINE

D. Zámečník, I. Veselý
Department of Control and Instrumentation, BUT in Brno,
Kolejní 4, Brno
E-mail : xzamec11@stud.feec.vutbr.cz, xvesel43@stud.feec.vutbr.cz

Abstract:
Model Predictive Control (MPC) is a modern approach belonging to the group of "long range predictive control" which enables control system taking into account in system manner constraints given by physical limitations. The state of art in most drive applications in industry is the Field Oriented Control (FOC). This paper deals with MPC controller replacing PI controllers in the current loop of the FOC cascade structure. Matlab/Simulink environment was used for carrying out simulations on the model of induction machine.

INTRODUCTION

Model Predictive Control (MPC) is an efficient methodology to solve complex multivariable system with constraints. The linear MPC has developed since the 70s of the past century. There are three main parts of MPC: cost function (usually quadratic), prediction discrete-time model of system (it must be as accurate as possible), and numerical optimization.
An optimal sequence of input actions over the whole prediction horizon is precalculated with respect to optimal future behavior of system.

Receding or moving horizon control (RHC) is a basic feature of MPC ensuring a feedback control: only the first computed optimal input action from the sequence of action inputs is transmitted into system, the remaining optimal input actions are discarded and at next sample time, which is based on new measurements, is the calculation repeated.

Field oriented control (FOC) is a cascade structure (with two loops) which enables to control an AC motor like a DC motor i.e. by controlling flux and torque separately. The inner loop and outer loop controls currents and flux/speed, respectively. The inner loop must be as fast as possible to ensure acceptable dynamic behavior of the whole system.

FIELD ORIENTED CONTROL

FOC is a long-term standard approach in the field of AC motors control. FOC is consisted of both Clarke and Park transformations.
Basic scheme of FOC is shown at the Fig. 1. In a red rectangle are current PI controllers which will be replaced by MPC controller later in this article.

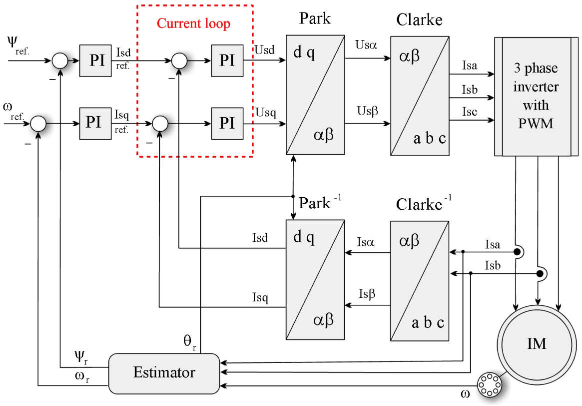

Fig. 1: The FOC scheme

Clarke transformation

The stator current (usually three phases) can be expressed as complex vector. All components of stator current (i_a, i_b, i_c) are created by one space stator current vector $\mathbf{i_s}$. This vector can express in complex plain by only two coordinates (axis i_α, i_β). This projection from three components (i_a, i_b, i_c) into 2D complex plain (i_α, i_β) is referred as 3/2 or Clarke transformation, equations are below.

$$i_{s,\alpha} = i_a$$
$$i_{s,\beta} = \frac{1}{\sqrt{3}} i_a + \frac{2}{\sqrt{3}} i_b \qquad (1)$$

Park transformation

The Park transformation is also referred as $(\alpha,\beta) \rightarrow (d,q)$ transformation it shown in Fig. 2 and it is possible to imagine it as a projection that transforms 2D orthogonal system (α,β) in the rotating reference frame aligned with the rotor flux (d,q):

- **d**-component is aligned with the rotor flux position θ, it implicates reactive power.
- **q**-component is aligned with the torque, it implicates active power.

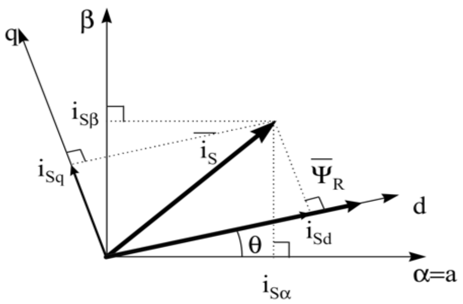

Fig. 2: Park transformation

The relation between above-mentioned reference frames is a simple rotation (2).

$$\begin{aligned} i_{sd} &= i_{s\alpha}\cos(\theta) + i_{s\beta}\sin(\theta) \\ i_{sq} &= -i_{s\alpha}\sin(\theta) + i_{s\beta}\cos(\theta) \end{aligned} \quad (2)$$

MODEL PREDICTIVE CONTROL

Model predictive control is consisted of prediction model, cost function (mostly quadratic), and numerical solution. The controller uses the prediction of states and outputs to determine appropriate actions.

Prediction model

Linear MPC use a linear and discrete-time model to predict the future behavior of system. The most general model is described by state-space model (3).

$$\begin{aligned} \mathbf{x}(t+1) &= \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) &= \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \end{aligned} \quad (3)$$

by subject the box constraints:

$$y_{\min} \le y \le y_{\max},\ u_{\min} \le u \le u_{\max} \quad (4)$$

The system future response is determined by input actions, model parameters and actual system state. Normal (hard) constraints cannot be exceeded, because the crash of numerical optimization algorithm (controller being in infeasibility region) occurs in the case of violation this constraints.
The induction machine equations in a rotating reference frame with velocity ω_k have a form (5).

$$\begin{aligned} \mathbf{u}_s &= R_s\mathbf{i}_s + \frac{\partial \mathbf{\Psi}_s}{\partial t} + j\omega_k\mathbf{\Psi}_s \\ 0 &= R_r\mathbf{i}_r + \frac{\partial \mathbf{\Psi}_r}{\partial t} + j(\omega_k - \omega)\mathbf{\Psi}_r \\ \mathbf{\Psi}_s &= L_s\mathbf{i}_s + L_m\mathbf{i}_r \\ \mathbf{\Psi}_r &= L_r\mathbf{i}_r + L_m\mathbf{i}_s \end{aligned} \quad (5)$$

For field-oriented control aligned with rotor flux the components are: $\mathbf{\Psi}_r = \psi_{rd}$, $\psi_{rq} = 0$ (and its derivative equals zero).
We can obtain first-order linear systems with two states (two inputs and outputs, too) after same transformations and by neglecting crosscoupling effect according to [1].

$$\mathbf{x} = \begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix},\quad \mathbf{u} = \begin{bmatrix} u_{sd} \\ u_{sq} \end{bmatrix},\quad \mathbf{y} = \begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix} \quad (6)$$

Resulting MIMO continuous state-space model is represented by (x):

$$\begin{aligned} \frac{\partial \mathbf{x}(t)}{\partial t} &= \begin{bmatrix} a & 0 \\ 0 & a \end{bmatrix}\mathbf{x}(t) + \begin{bmatrix} b & 0 \\ 0 & b \end{bmatrix}\mathbf{u}(t) \\ \mathbf{y}(t) &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\mathbf{x}(t) \end{aligned} \quad (7)$$

, where the coefficients *a*,*b* are defined by induction machine parameters according to following relations:

$$b = \frac{L_r}{L_s L_r - L_m^2},\ a = \frac{L_r^2 R_s + L_h^2 R_r}{L_r L_m^2 - L_s L_r^2}. \quad (8)$$

Motor parameters

R_s	Resistance of stator	0.894Ω
R_r	Resistance of rotor	0.850 Ω
L_s	Stator inductance	0.119 H
L_r	Rotor inductance	0.118 H
L_m	Mutual inductance	0.112 H

Cost function and optimal solution

A total system response is defined by sum of both forced and free response depends only on the past input actions and initial state, respectively.

$$\mathbf{u}(t,N) = \left[\mathbf{u}(t)^T, \mathbf{u}^T(t+1), \ldots, \mathbf{u}^T(t+N-1)\right]^T \quad (9)$$

The state response has a normal and a matrix form:

$$\begin{aligned} \mathbf{x}(t+k) &= \mathbf{A}^k\mathbf{x}(t) + \sum_{i=t}^{t+k-1}\mathbf{A}^i\mathbf{B}\mathbf{u}(k-1-i) \\ \mathbf{x}(t+k) &= \mathbf{P}\mathbf{x}(t) + \mathbf{H}\mathbf{u}(t) \end{aligned} \quad (10)$$

, where *N* is a finite prediction horizon. A quadratic cost function that covers this prediction horizon is defined by (11) and in matrix form by (12).

$$J(\mathbf{x}(t),\mathbf{u}(t,N)) = \frac{1}{2}\sum_{k=t}^{t+N-1}[\mathbf{x}^T(k)\mathbf{Q}\mathbf{x}(k) + \mathbf{u}^T(k)]$$
$$+\frac{1}{2}\mathbf{x}^T(k+N)\mathbf{Q}\mathbf{x}(k+N) \quad (11)$$

$$J(\mathbf{x}(t),\mathbf{u}(t,N)) = \frac{1}{2}\mathbf{u}^T(t,N)\mathbf{T}\mathbf{u}(t,N) + \mathbf{x}(t)^T$$
$$+\frac{1}{2}\mathbf{x}(t)^T\mathbf{Y}\mathbf{x}(t) \quad (12)$$

Where $\mathbf{T}$=diag($\mathbf{R}$,…,$\mathbf{R}$)+$\mathbf{H}^T\mathbf{O}\mathbf{T}$, $\mathbf{F}=\mathbf{P}^T\mathbf{O}\mathbf{H}$, $\mathbf{Y}=\mathbf{P}^T\mathbf{O}\mathbf{P}$, $\mathbf{O}$=diag($\mathbf{Q}$,…$\mathbf{Q}$,$\mathbf{Q}_N$). Matrix Q penalizes output state over whole prediction horizon except last step. The terminal state is penalized by $\mathbf{Q}_N$, where $\mathbf{Q}_N>\mathbf{Q}$, to achieve better stability performance. Matrix **R** penalizes input action signal **u**.
Optimal input action for constrained system with quadratic cost function is defined by equation (13).

$$\mathbf{u}^*_{t,N}(\mathbf{x}(t)) = \underset{\mathbf{u}_{t,N}}{\operatorname{argmin}}\left[\frac{1}{2}\mathbf{u}^T_{t,N}\mathbf{T}\mathbf{u}_{t,N} + \right.$$
$$\left. + \mathbf{x}^T\mathbf{F}\mathbf{u}_{t,N} \middle| \mathbf{G}\mathbf{u}_{t,N} \le \mathbf{w} + \mathbf{E}\mathbf{x}(t)\right] \quad (13)$$

Numerical optimization

QP is mathematical optimization for quadratic objective function and linear constraints. Ways how to solve a numerical optimization of QP are twofold.

- Implicit controller solves quadratic programming task online, i.e. in each single sampling interval solves quadratic problem. This way demands a huge computational effort during evaluation.
- Explicit controller calculates for all combinations of states all control laws (during design phase of controller). It solves not QP, but multi-parametric QP (mp-QP), where initial state vector is a parameter. Main idea is to remove a huge computational effort into offline part. In online part, the explicit controller only searches out state space and choose an appropriate solution which enables faster solution in evaluation. The state space is divided into linear separated regions (given by length of prediction horizon and number of state variables and constraints).

The solution of optimization (explicit controller) is the resulting control law. The input action is a piecewise affine function of actual state vector and control law i.e. $\mathbf{u}^*=\mathbf{K}_i\mathbf{x}+q_i$ for each *i-th* region.

In this article was Multi-Parametric Toolbox for Matlab [5], which enables design, analysis and deployment with wide range of systems.

Extension for tracking problem

In many practical applications we need the free-offset tracking control (reference tracking). One option how to include it into MPC control abilities is following.
A new augmented state vector is introduced into model and consequently new state space equations have been derived:

$$\tilde{\mathbf{x}}(k+1) = \begin{bmatrix} \mathbf{A} & 0 & \mathbf{B} \\ 0 & \mathbf{I} & 0 \\ 0 & 0 & \mathbf{I} \end{bmatrix}\tilde{\mathbf{x}}(k) + \begin{bmatrix} \mathbf{B} \\ 0 \\ \mathbf{I} \end{bmatrix}\Delta\mathbf{u}, \quad (14)$$

where $\tilde{\mathbf{x}}(k) = [\mathbf{x}, \mathbf{x}_{ref}, \mathbf{u}(k-1)]^T$ and input action is $\mathbf{u}(k) = \mathbf{u}(k-1) + \Delta\mathbf{u}$.

SIMULATIONS

An explicit linear MPC controller, which was used in the simulations, was designed as follows: the controller can handle tracking reference value (not only control to origin), length of prediction horizon *N*=7, a sampling period T=125μs (very short). The constraints were on input -200 ≤ u ≤ 200V, and output -21 ≤ y ≤ 21A. Cost matrices were set as follows: $\mathbf{Q}=8000\mathbf{I}$, $\mathbf{Q}_N=15000\mathbf{I}$, and $\mathbf{R}=3\mathbf{I}$, where **I** is identity matrix (2x2).
We have the very fast dynamic system (short sample period) consequently the explicit controller has been chosen.

Process of simulation

t(s)	*0*	*0.25*	*1.3*	*1.8*	*2*	*2.5*	*3*
ω (s^{-1})	0	40	40	40	80	80	-80
T(Nm)	0	0	60	0	0	-50	-50

We can see patterns of the transformed stator currents i_d,i_q MPC in a comparison with well-tuned PI controllers depicted at the Figures 3 and 4, respectively.

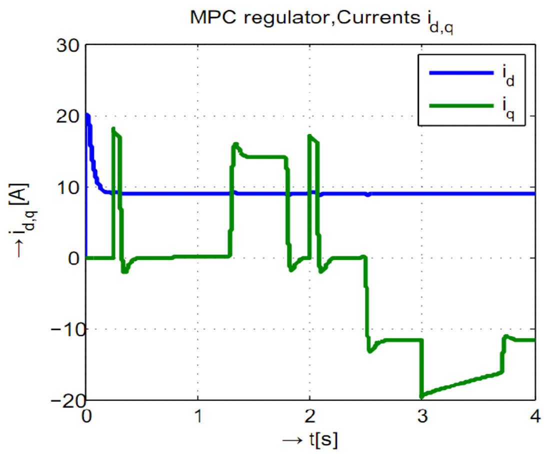

Fig. 3: MPC controller *d,q* currents

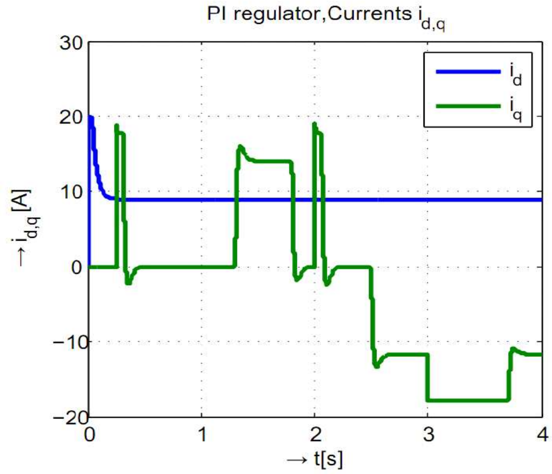

Fig. 4: PI controller *d,q* currents

At the Fig. 5 is shown Simulink model. At the figures 6 and 7 are depicted patterns of velocity and three stator voltages (u_a, u_b,u_c), respectively.

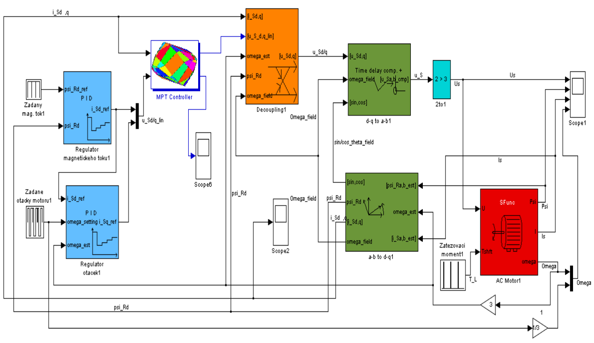

Fig. 5: Simulink model with MPC controller

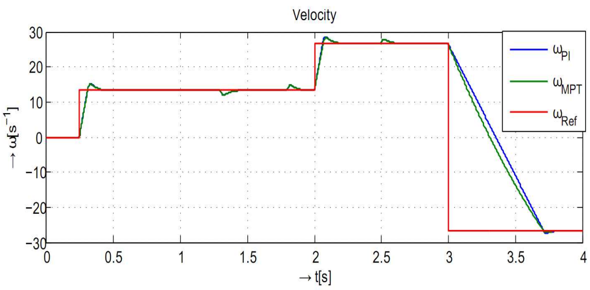

Fig. 6: Pattern of velocity

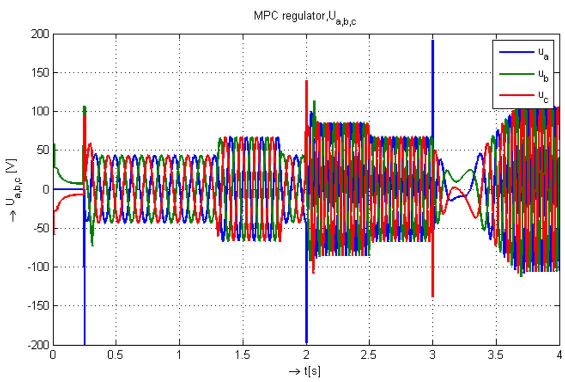

Fig. 7: Pattern of three stator voltages

CONCLUSION

The Linear MPC controller was used in the current loop of the field oriented control cascade structure.
The explicit solution of optimization (mp-QP) was used for a very fast system (induction machine), which had an short sampling period T_s=125μs.
The MPC controller was compared with the PI controller.
Simulation showed that the more complicated MPC controller did not surpassed simple and well-tuned PI controller, but the certain different in the torque component is showed at the Fig. 6 and Fig.3,4 during the reversion of velocity (from time 3s). MPC controller is only little bit better than PI, moreover it has the ability to satisfy the constraints on output.

However, since linear MPC with a very simplified model and only in the current loop was used, so it can be assumed that using more accurate model of induction machine and by introducing more complex controller can achieve the more interesting results, especially in the field of ensuring constraints on outputs and states.

ACKNOWLEDGEMENT

The research has been supported by Czech Science Foundation under the project 102/09/H081 SYNERGY - Mobile Sensoric Systems and Network.

This work was supported by grant "Research of Modern Methods and Approaches in Automation" from the Internal Grant Agency of Brno University of Technology (grant No. FEKT-S-11-6).

REFERENCES

[1] Qin, S.J.; Badgwell, T.A.: An overview of industrial model predictive control applications, In J.C.Kantor, C.E.Garcia, and B.Carnahan editors, *Fifth International Conference on Chemical Process Control – CPC*, 1996, Vol.5, pp. 232-256.

[2] Kennel, R., Linder, A., Linke, M.: Generalized Predictive Control (GPC) - Ready for use in drive applications?, *PESC Record - IEEE Annual Power Electronics Specialists Conference*, 2001, Vol.4, pp. 1839-1844.

[3] Bemporad, A.; Morari, M.; Dua, V.; Efstratois, N.; Pistikopoulos, N.: The explicit linear quadratic regulator for constrained systems, *Automatica*, 2001, Vol. 38, pp. 3-20.

[4] Linder, A.; Kennel, R.: Model Predictive Control for Electrical Drives, *Power Electronics*

Specialists Conference, 2005. PESC '05. IEEE 36th, pp.1793-1799.

[5] Kvasnica, M.; Grieder, P.; Baoti, M.: Multi-Parametric Toolbox (MPT) [online]. Place: Automatic Control Laboratory, ETH Zurich, Switzerland, 2004 [cit. 2011-05-13]. Available from WWW: < http://control.ee.ethz.ch/~mpt/>

[6] Bolognani, S.; Peretti, L.; Zigliotto, M.: Design and Implementation of Model Predictive Control for Electrical Motor Drives, *IEEE Transactions on industrial electronics*, 2009, Vol. 56, n.6, pp. 1925-1936.

[7] Miranda, H.; Cortes, P.; Yuz, J.I.; Rodriguez, J.: Predictive Torque Control of Induction Machines Based on State-Space Models, *Industrial Electronics, IEEE Transactions on*, June 2009, Vol.56, no.6 pp. 1916-1924.

[8] Geyer, T.; Papafotiou, G.; Morari, M.: Model Predictive Direct Torque Control Part I: Concept, Algorithm, and Analysis, *IEEE Transactions on industrial electronics*, 2009, vol.56, n. 6.

[9] Papafotiou, G.; Kley, J.; Papadopoulos, K.; Bohren, P.; Morari, M.: Model Predictive Direct Torque Control Part II: Implementation and Experimental Evaluation, *IEEE Transactions on industrial electronics*, 2009, Vol. 56, n. 6.