MODEL PREDICTIVE CURRENT CONTROLLER IN THE FIELD ORIENTED CONTROL OF INDUCTION MACHINE

Dušan Zámečník, Ivo Veselý
Doctoral Degree Programme (2), FEEC BUT
E-mail: xzamec11@stud.feec.vutbr.cz, xvesel43@stud.feec.vutbr.cz

Supervised by: Pavel Václavek
E-mail: vaclavek@feec.vutbr.cz

Abstract: Two approaches are used in the most controllers for linear system with constraints: anti-windup and model predictive control. The Model predictive control can satisfy physical limitations or constraints in system manner because the constraints are involved in a optimization. The Field oriented control is state of art in most drive applications. This paper deals with the model predictive controller used instead PI controller in the current loop of the field oriented control. Resulting simulations carried out on induction machine model in Matlab/Simulink environment.

Keywords: model predictive control, induction machine, constraints

1 INTRODUCTION

Model predictive control (MPC), also referred to as receding horizon control, has become very popular methodology, especially for linear constrained systems. Linear MPC is popular since the 70s of the past century. First articles, that forms basics of MPC (moving horizon, linear programming), was emerged in 1963 and 1978 by Propoi and Richalet et al. , respectively. The state of art is MPC a standard advanced control technology for the process industry it provides a systematic approach[1] to control the complex multivariable dynamical system with constraints.

Receding or moving horizon is an essential feature of MPC. At each sampling interval, starting at the current state, an open-loop optimal control sequence is solved over a finite horizon. Only first optimal signal is send to the system. At the next time step (after a new measurement of the states) a new optimal sequence is calculated over a shifted horizon.

2 FIELD ORIENTED CONTROL

Field oriented control (FOC) is long-term standard approach in the field of elektric drive control. FOC is consisted of both Clarke and Park transformation. The entire scheme of FOC is pointed to at figure 1 which containts both Park and Clarke transformation in both forward and inverse direction.

2.1 CLARKE TRANSFORMATION

Three phase stator current can be expressed as complex space vector. All stator currents (i_a, i_b, i_c) creates only one space stator current vector $\mathbf{i_s}$ which can express in complex plain by only two ortoghonal axis called i_α, i_β. Projection stator currents into the 2D plane, also reffered as $(a,b,c) \rightarrow$

[1] Non systematic or ad hoc approach was used by practitioners before MPC without solid theoretical background but with the same performance.

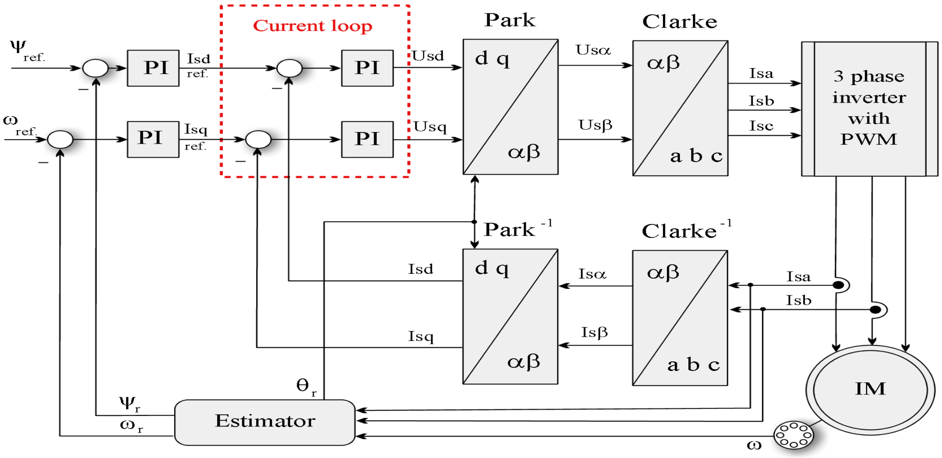

Figure 1: The entire scheme of FOC

(α, β) or transformation 3/2 has equations below:

$$\begin{aligned} i_{s,\alpha} &= i_a \\ i_{s,\beta} &= \frac{1}{\sqrt{3}} i_a + \frac{2}{\sqrt{3}} i_b \end{aligned} \quad . \tag{1}$$

2.2 PARK TRANSFORMATION

Also reffered as $(\alpha, \beta) \to (d, q)$ is a projection that transform 2D orthogonal system in the rotating reference frame alligned with the rotor flux $\vec{\Psi}_r$. The meaning of (d, q) is: **d**-component is aligned with the rotor flux, and it implicates reactive power, **q**-component is aligned with the torque, and it implicates active power. The relation between above-mentoined reference frame is:

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

where θ is the rotor flux position.

3 MODEL PREDICTIVE CONTROL

Model predictive controller uses the prediction of states and outputs to determine appropriate action and simultaneosly take into account the constraints.

3.1 PREDICTION MODEL

Linear MPC use a linear model to predict the system behaviour. A distrete-time state-space model (3) is used to describe future response of system:

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

by subject the constraints:

$$y_{min} \le y \le y_{max}, \quad u_{min} \le u \le u_{max}. \tag{4}$$

The system future response is based on future control action, model parameters and the actual system state[1]. Normal (hard) constraints (4) can not be exceeded, because in the case of violation this the crash of numerical optimisation algorithm occurs (controller being in infeasibility region). This fact causes the introducing a soft constraint. Soft and hard constraints are used on the ouput and on the input of system, respectively. The equations of induction machine in a reference frame rotating with the speed ω_k have a form:

$$\begin{aligned} \mathbf{u_s} &= R_s\mathbf{i}_s + \frac{\partial \Psi_s}{\partial t} + j\omega_k\Psi_s, && (5)\\ 0 &= R_r\mathbf{i}_r + \frac{\partial \Psi_r}{\partial t} + j(\omega_k - \omega)\Psi_r, && (6)\\ \Psi_s &= L_s\mathbf{i}_s + L_m\mathbf{i}_r, && (7)\\ \Psi_r &= L_r\mathbf{i}_r + L_m\mathbf{i}_s, && (8) \end{aligned}$$

After same transformations, simplifications and neglecting decoupling according to [4] we can obtain first-order linear system with two states, inputs and outputs:

$$\mathbf{x} = \begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} u_{sd} \\ u_{sq} \end{bmatrix}, \quad \mathbf{y} = \begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix}. \tag{9}$$

The resulting state-space continuous-time model is represented below:

$$\begin{aligned} \frac{\partial \mathbf{x}(t)}{\partial t} &= \begin{bmatrix} a & 0 \\ 0 & a \end{bmatrix} \mathbf{x}(t) + \begin{bmatrix} b & 0 \\ 0 & b \end{bmatrix} \mathbf{u}(t), \\ \mathbf{y}(t) &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{x}(t), \end{aligned} \tag{10}$$

where the coefficients a, b are determined from induction machine parameter by equations:

$$b = \frac{L_r}{L_sL_r - L_m^2}, \quad a = \frac{L_r^2R_s + L_h^2R_r}{L_rL_m^2 - L_sL_r^2}. \tag{11}$$

3.2 COST FUNCTION AND OPTIMAL SOLUTION

Firstly, a total system state response is created by sum of forced and free response depending only on action signal and on initial states, respectively.

Now we need to define the input vector $\mathbf{u}(t,N)$ with formulation $\mathbf{u}(t+k|t) = \mathbf{u}_{t+k}$, and the system state response in matrix form:

$$\begin{aligned} \mathbf{u}(t,N) &= \left[\mathbf{u}_t^T, \mathbf{u}_{t+1}^T, \ldots, \mathbf{u}_{t+N-1}^T\right], && (12)\\ \mathbf{x}(t+k) &= \mathbf{A}^k\mathbf{x}(t) + \sum_{i=t}^{t+k-1} \mathbf{A}^i\mathbf{B}\mathbf{u}(k-1-i), && (13)\\ \left[\mathbf{x}_t^T, \mathbf{x}_{t+1}^T, \mathbf{x}_{t+2}^T, \ldots \mathbf{x}_{t+N}^T\right] &= \mathbf{P}\mathbf{x}(t) + \mathbf{H}\mathbf{u}(t), && (14) \end{aligned}$$

where N is a prediction horizon. Secondly, a quadratic cost function that covers the finite prediction horizon N is defined similiary:

$$\begin{aligned} J\left(\mathbf{x}(t), \mathbf{u}_t, \ldots, \mathbf{u}_{t+N-1}\right) &= \frac{1}{2}\sum_{k=t}^{t+N-1} \left[\mathbf{x}_k^T\mathbf{Q}\mathbf{x}_k + \mathbf{u}_k^T\mathbf{R}\mathbf{u}_k\right] + \frac{1}{2}\mathbf{x}_{t+N}^T\mathbf{Q}_N\mathbf{x}_{t+N}, && (15)\\ J\left(\mathbf{x}(t), \mathbf{u}(t,N)\right) &= \frac{1}{2}\mathbf{u}_{t,N}^T\mathbf{T}\mathbf{u}_{t,N} + \mathbf{x}^T\mathbf{F}\mathbf{u}_{t,N} + \frac{1}{2}\mathbf{x}^T(t)\mathbf{Y}\mathbf{x}(t), && (16) \end{aligned}$$

where $\mathbf{T}$ =diag$(\mathbf{R}, \ldots, \mathbf{R}) + \mathbf{H}^T\mathbf{O}\mathbf{T}$, $\mathbf{F} = \mathbf{P}^T\mathbf{O}\mathbf{H}$, $\mathbf{Y} = \mathbf{P}^T\mathbf{O}\mathbf{P}$, $\mathbf{O}$ =diag$(\mathbf{Q}, \ldots, \mathbf{Q}, \mathbf{Q}_N)$. Matrix $\mathbf{Q}$ penalizes output state over entire prediction horizon except last step. The final state step of prediction horizon is penalized by $\mathbf{Q_N} > \mathbf{Q_N}$ so that we have achieve better stability controller. Matrix $\mathbf{R}$ penalizes input signal $\mathbf{u}$.

Optimal control signal for constrained system with quadratic cost funcion is determined:

$$\mathbf{u}^*_{t,N}(\mathbf{x}(t)) = \arg\underbrace{\min}_{\mathbf{u}_{t,N}} \left[\frac{1}{2}\mathbf{u}^T_{(t,N)}\mathbf{T}\mathbf{u}_{t,N} + \mathbf{x}^T\mathbf{F}\mathbf{u}_{t,N} | \mathbf{G}\mathbf{u}_{t,N} \leq \mathbf{w} + \mathbf{E}\mathbf{x}(t)\right]. \tag{17}$$

3.3 NUMERICAL OPTIMISATION

There are two ways to solve a numerical optimisation of QP.

Online controller solves quadratic programming task. This demands huge computational effort (powerful CPU), also it is primary approach for calculating of control sequence.

Offline controller solves not QP but multi-parametric QP (mpQP). Main idea of this approach is divide computation into offline and online part. It can be imagine, that all solution for all parameters (initial state **x** is now a parameter) has calculated in offline part, and the control algorithm only search solutions according to initial state and reference. It follows less computational effort but more memmory requirements for this controller. The resulting control law is piecewise affine function i.e. for each *i-th* region exist linear control law in form: $\mathbf{u}^* = -\mathbf{K}_i\mathbf{x} + q_i$.

In this article was used Multi-Parametric Toolbox for Matlab [3], that enable design, analysis and much more with wide scale of systems. The requirement of constrained manipulated value is satisfied by the numerical solver YALMIP which is distributed with this toolbox.

4 SIMULATIONS

Explicit linear MPC controller was designed as follows: controller can handle tracking setpoint, prediction horizon $N = 7$, a sampling period $T = 125\mu s$, constraints were: on the input $-200 \leq u \leq 200V$ and output $-21 \leq i \leq 21A$. Cost matrices were set as follows: $\mathbf{Q} = 8000\mathbf{I}, \mathbf{Q_n} = 15000\mathbf{I}$, and $\mathbf{R} = 3\mathbf{I}$, where $\mathbf{I}$ is identity matrix.

Process simulation parameters is in Tab. 1. A comparison the patterns of the transformed currents i_d, i_q between the MPC controller and the well-tuned PI controller with saturation is presented in Fig. 2 and a velocity is depicted in Fig. 3.

t[s]	0	0.25	1.3	1.8	2	2.5	3
$\omega[s^{-1}]$	0	40	40	40	80	80	-80
T[Nm]	0	0	60	0	0	-50	-50

Table 1: Process simulation

5 CONCLUSION

The Linear MPC controller was used in the current loop of the field oriented control cascade structure. The explicit solution of optimisation (mp-QP) was used for a very fast system (induction machine), which had an extremely short sampling period $T_s = 125\mu s$. The MPC controller was compared with the PI controller, even though the proper is general to compare with PID. The state of the field oriented control in the current loop is only PI controller, no PID. Moreover the first-order model is used and that implies that the D-component is not required to control. Simulation showed that the more complicated MPC controller did not surpassed simple and well-tuned PI controller with saturation on output, but the certain different in the torque component is showed at the Fig. 3 and Fig. 2 during the reversing of velocity (from time 3s). 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

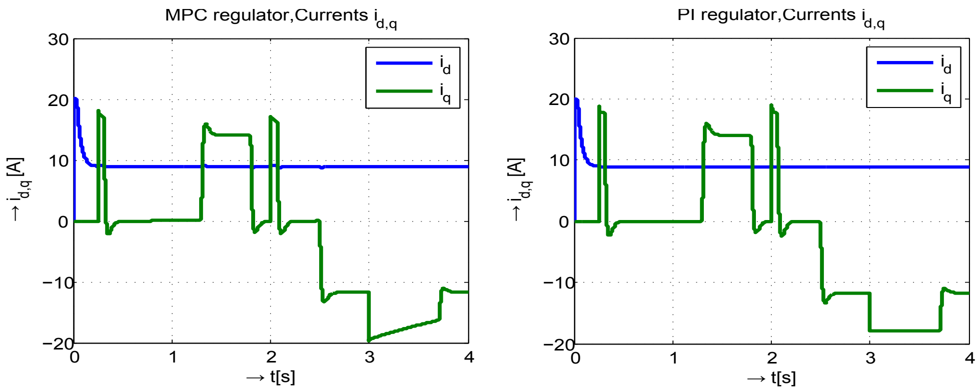

Figure 2: Comparison MPC and PI: $I_{dq}(A)$ currents

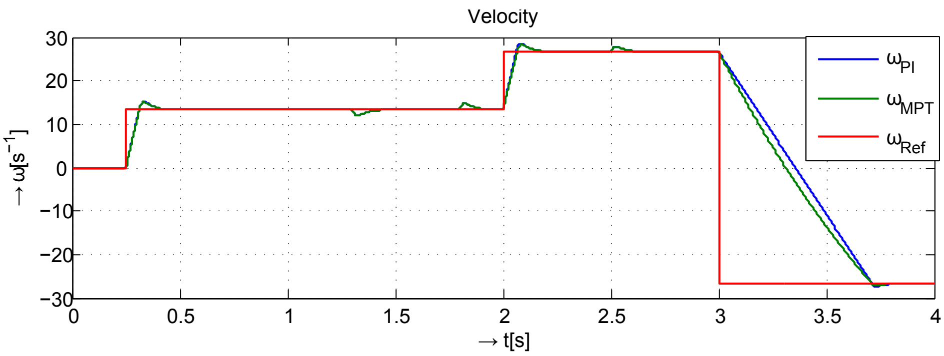

Figure 3: Comparison PI and MPC: angular velocity $\omega(s^{-1})$

by the exemption from the concept of FOC can achive the more interresting results, especially in the field of ensuring constraints.

ACKNOWLEDGEMENT

This work was supported by grant "Modern Methods and Approaches in Automation" from the Internal Grant Agency of Brno University of Technology (grant No. FEKT-S-10-12).

REFERENCES

[1] 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.

[2] Qin, S.J.; Badgwell, T.A.: A survey of industrial model predictive control technology,*Control Engineering Practice,* 2002, Vol. 11, pp. 733-764.

[3] Kvasnica, M.; Grieder, P.; Baoti, M.: Multi-Parametric Toolbox (MPT), 2004, [Available online] URL: <http://control.ee.ethz.ch/∼mpt/>.

[4] Linder, A.; Kennel, R.: Model Predictive Control for Electrical Drives, *Power Electronics Specialists Conference, 2005. PESC '05. IEEE 36th,* pp.1793-1799.