Sensorless Determination of Induction Motor Drive Speed Using MRAS Method

Pavel Karlovský, Radek Linhart, Jiří Lettl
Department of Electric Drives and Traction
Faculty of Electrical Engineering
Czech Technical University in Prague
Prague, Czech Republic
karlopav@fel.cvut.cz, linharal@fel.cvut.cz, lettl@fel.cvut.cz

Abstract – In most cases, using of an optical sensor represents the most common solution to obtain the induction motor actual speed. As in some cases the use of the speed sensor carries number of problems, an effort to find a solution that does not require a speed sensor is obvious in recent time. The paper presents some results of employing MRAS (Model Reference Adaptive System) method for sensorless induction motor shaft speed determination. The working principles and the possible algorithm mathematical derivation are described. Results of the drive performance simulation in MatLab/Simulink environment are presented. Finally, the algorithm is verified on the real induction motor drive utilizing dSPACE platform. Obtained results demonstrate good function and performance of the system.

Keywords - Induction motor drive; sensorless speed determination, MRAS method.

I. INTRODUCTION

In recent time, in the field of electric drives, the most widely used one is a drive with an induction motor. Mainly it is caused by its simplicity, robustness, and low price. On the other hand, such induction motor drive needs sophisticated control algorithms if speed or torque control is required. Most of the algorithms require the information about shaft rotational speed in order to work properly. The common way of the speed information obtaining is use of a speed sensor use. However, the use of such a sensor is accompanied with a lot of undesired effects [1]. The most important disadvantage is the complexity of the sensor which decreases the robustness of the whole drive. Other disadvantage is the increased price of the drive. Because of these reasons, it is desired to eliminate the sensor and obtain the speed information in some other way. This requirement can be fulfilled by some special methods. The inputs for these methods can be only measurable variables, such as supply voltage and stator currents. The most common ones are open-loop calculation, MRAS, observers or Kalman filters [2] – [3]. Each of them can be implemented in numerous ways and each of those methods has its advantages and disadvantages. This paper utilizes one possible implementation of the MRAS method [4] - [9].

II. MRAS METHOD

A. Working principle

The MRAS methods are based on the principle of two models calculating the same variable and control mechanism. The first model is called reference model. Its purpose is to calculate value of variable x from the measured quantities. This value of x is regarded to be correct and therefore the reference. The second model is called adaptive and it estimates the same variable x from some measured variables and from estimated variable. In the text, the calculated x that comes out from the adaptive model is marked with a circumflex as $\hat{x}$. The estimated variable is unknown and is treated as a parameter $\hat{p}$. The outputs of these two models are subtracted one from the other and the error between them is then an input for the control mechanism which adjusts the estimated parameter. The purpose of the control mechanism is to drive the error towards zero value. When the error is near zero, the both models have the same output. As we regard the reference model to be correct and the models have the same output, we can say that also the adaptive model is correct and therefore the estimated parameter is set correctly. In figure 1, a scheme of working principle of this method is depicted. The inputs of the reference model are U1 and U2, and variable $x$ is the reference model output. The inputs of the adaptive model are U2 and a parameter $\hat{p}$. The calculated variable $\hat{x}$ is the adaptive model output. The difference $\epsilon$ between $x$ and $\hat{x}$ is the input of the controller block that adjusts the parameter $\hat{p}$.

![Figure 1: MRAS working principle](image-url)
B. Rotor Flux Based MRAS

The MRAS methods can generally calculate almost any parameter of the motor. The method implemented in the article obtains the angular speed of the motor. The calculated variable in both models is the rotor flux vector and the unknown parameter is the angular speed. The r model inputs are stator voltage and stator current vectors. Therefore, we can write

\[ U_1 = V_s \]
\[ U_2 = I_s \]
\[ X = \Psi_r \]
\[ \dot{X} = \dot{\Psi}_r \]
\[ \dot{P} = \dot{\omega}_r \]

where \( V_s \) is the stator voltage vector, \( I_s \) is the stator current vector, \( \Psi_r \) is the rotor magnetic flux vector, \( \omega_r \) is the angular shaft speed.

The mathematical derivation of the method is based on the equations for induction motor (1) – (4). All the equations are calculated in a stator reference frame.

\[ v_s = R_s \cdot i_s + \frac{d\Psi_s}{dt} \]  
(1)

\[ 0 = R_s \cdot i_s + \omega_s \cdot \Psi_r \]  
(2)

\[ \Psi_s = L_s \cdot i_s + L_M \cdot i_r \]  
(3)

\[ \Psi_r = L_r \cdot i_r + M \cdot i_s \]  
(4)

where \( R_s \) is the stator resistance, \( \Psi_s \) is the stator magnetic flux vector, \( R_s \) is the rotor resistance, \( L_s \) is the stator inductance, \( L_m \) is the magnetizing inductance, \( L_r \) is the rotor inductance, \( i_s \) is the rotor current vector.

The reference model contains two measured variables. They are the reconstructed voltage vector and measured stator current vector. From them, the rotor flux vector can be calculated (5).

\[ \frac{d\Psi_r}{dt} = L_r \left( u_r - R_s \cdot i_s - L_s \cdot \sigma \cdot \frac{di_s}{dt} \right) \]  
(5)

Where \( \sigma \) is leakage coefficient calculated as \( \sigma = \frac{i_m}{i_s} \).

The adaptive model contains only measured current vector and shaft angular speed treated as parameter. It calculates the rotor flux vector as well according to (6).

\[ \frac{d\dot{\Psi}_r}{dt} = \left( \dot{\omega}_r - \frac{1}{\tau_r} \right) \cdot \dot{\Psi}_r + \frac{L_m}{\tau_r} \cdot i_s \]  
(6)

where \( \tau_r \) is rotor time constant calculated as \( \tau_r = \frac{R_r}{L_s} \).

The control mechanism adjusts the parameter. At first, the error between outputs of models is calculated in equation (7).

\[ e(t) = \dot{\Psi}_{ra} \cdot \Psi_{rb} - \Psi_{rb} \cdot \Psi_{ra} \]  
(7)

Then from the error the angular speed is set in controller. In this case, the PI controller is used and calculations are performed according to equation (8).

\[ \ddot{\omega}_r(t) = K_p \cdot e(t) + K_i \int_0^t e(t) dt \]  
(8)

If the MRAS method shall work properly the parameters of the induction machine equivalent circuit must be known. From equations for the reference and adaptive models it is evident that parameters \( R_s, L_r, L_m, R_r, \) and \( L_s \) are required.

III. IMPLEMENTATION

To verify the proposed algorithm it was examined by simulation in MatLab/Simulink environment, first. Then the algorithm functionality was tested on the real induction motor drive.

Nominal values of the four-pole induction motor of 5.5 kW used at real drive testing and at MatLab/Simulink simulation are shown in table 1.

<table>
<thead>
<tr>
<th>Table 1: Nominal values of IM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter</td>
</tr>
<tr>
<td>----------</td>
</tr>
<tr>
<td>( U_n )</td>
</tr>
<tr>
<td>( I_0 )</td>
</tr>
<tr>
<td>( P_n )</td>
</tr>
<tr>
<td>( \omega_n )</td>
</tr>
<tr>
<td>( Y )</td>
</tr>
</tbody>
</table>

The necessary parameters of induction motor equivalent circuit (\( R_s, R_r, L_m, L_{mn}, L_{mn0} \)) were measured under specific conditions that have been preserved during the whole measurement to prevent changing the parameter values. The induction motor equivalent circuit parameters are recapitulated in table 3.

<table>
<thead>
<tr>
<th>Table 2: Parameters of IM equivalent circuit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter</td>
</tr>
<tr>
<td>----------</td>
</tr>
<tr>
<td>( R_s )</td>
</tr>
<tr>
<td>( L_{mn} )</td>
</tr>
<tr>
<td>( R_r )</td>
</tr>
<tr>
<td>( L_{mn0} )</td>
</tr>
<tr>
<td>( L_m )</td>
</tr>
</tbody>
</table>

The practical implementation was carried out by means of the dSPACE DS1103 platform controlled through the connected computer. To the computer control the ControlDesk program was employed. It is able to visualize online the program variables and record them. The processor time loop of the control algorithm was set to 50 \( \mu \text{s} \). The shaft speed and generated torque were adjustable in order to achieve different working points. The induction motor was controlled by scalar control with PWM signal at transistor switching frequency of 2 kHz as the power inverter. The load was made up of a separately excited DC motor with its terminals connected to the resistance and its shaft connected to the IM shaft. The measurement of the angular speed of the shaft was performed by the tachogenerator and the signal was used for verifying the precision of the designed and
implemented algorithm. The scheme in figure 2 shows the hardware composition.

**Figure 2: The workplace block diagram**

The implementation of individual models and the whole MRAS technique are depicted in figures 3, 4, and 5. As both of the models use integrations, a problem with DC offset can occur. For that reason, a simple filter eliminating DC component was implemented into the integrators.

**Figure 3: Reference model**

**Figure 4: Adaptive model**

**Figure 5: Whole MRAS model**

The constant of the PI controller were tuned to give the best response without overshoot. The values were set according to table 3.

**Table 3: PI controller values**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_p$</td>
<td>28</td>
</tr>
<tr>
<td>$K_i$</td>
<td>10000</td>
</tr>
</tbody>
</table>

IV. RESULTS

Running at different angular speeds of the induction motor shaft was performed in order to verify the designed technique. Both simulation and experimental testing were performed.

A. Simulation

The simulation runs for various different speed references from 300 rpm to 1500 rpm were performed. Obtained results are presented in figure 6 where speed reference, motor speed, and calculated speed are displayed.

**Figure 6: Simulation of speed in different working points**

A detail of transient during speed reference change from 1200 rpm to 1500 rpm is displayed in figure 7. Here the reference, calculated and real speeds are shown. For this transient, a rotor flux of reference and adaptive models are also shown as well as the error between them. It is evident, that the calculated angular speed matches the real value of the motor model very well and the error is very small. The error between calculated fluxes represents the PI controller input and it is quickly pushed towards zero by the controller. When the transient goes away, the error returns to zero.

**Figure 7: Detail of the transition**

B. Experiment

To further verify the algorithm, the experimental measurement was performed as well. The same run sequence of the reference angular shaft speeds as in simulation was chosen. The measurement results are shown in figure 8. Speed measured by tachogenerator,
reference speed, and speed calculated by the MRAS algorithm are plotted. Similarly as in simulations, the model’s speed follows the real motor’s speed quickly and precisely.

![Figure 8: Measurement on real drive](image)

A detail of speed measurement during the speed reference change from 1200 rpm to 1500 rpm is presented in figure 9.

![Figure 9: Measurement on real drive in detail](image)

Ultimately, examination of the method’s accuracy in numerous working points from -1500 rpm to 1500 rpm was performed. The error was calculated as a ratio between the real and calculated speed value. This is presented in figure 10.

![Figure 10: Error between the real and calculated angular speed](image)

From the figure it is evident, that the method works with very small error in the area of higher speeds, but the method is less accurate and the error gets greater in the area of low speeds. The exact measured values are summarized in table 4.

<table>
<thead>
<tr>
<th>Speed [rpm]</th>
<th>Error [%]</th>
<th>Speed [rpm]</th>
<th>Error [%]</th>
</tr>
</thead>
<tbody>
<tr>
<td>80</td>
<td>-35</td>
<td>600</td>
<td>-0.502</td>
</tr>
<tr>
<td>100</td>
<td>-27</td>
<td>700</td>
<td>-0.143</td>
</tr>
<tr>
<td>120</td>
<td>-17</td>
<td>800</td>
<td>-0.125</td>
</tr>
<tr>
<td>130</td>
<td>-15</td>
<td>900</td>
<td>0</td>
</tr>
<tr>
<td>150</td>
<td>-11</td>
<td>1000</td>
<td>0.201</td>
</tr>
<tr>
<td>180</td>
<td>-9</td>
<td>1100</td>
<td>0.182</td>
</tr>
<tr>
<td>200</td>
<td>-8.5</td>
<td>1200</td>
<td>0.251</td>
</tr>
<tr>
<td>300</td>
<td>-4.04</td>
<td>1300</td>
<td>0.309</td>
</tr>
<tr>
<td>400</td>
<td>-2.02</td>
<td>1400</td>
<td>0.358</td>
</tr>
<tr>
<td>500</td>
<td>-1.21</td>
<td>1500</td>
<td>0.401</td>
</tr>
</tbody>
</table>

V. Conclusion

The paper presents one possible option of MRAS techniques of determining the speed of the induction motor drive. In the paper the mathematical analysis of the method is provided. The algorithm simulation in the MatLab/Simulink environment and implementation on the real induction motor drive using dSPACE platform are performed. The obtained results of the simulation and experimental measurement are shown and a comparison in multiple working points is presented.

The advantage of the method is its relatively easy implementation. The other advantage is its good accuracy in normal speed operation. From the results it is clear that the method works reliably in steady states and transients in area of normal speeds with the error less than 0.5%.

The disadvantage is the greater error in the area of lower speeds, where the error increases quickly. Other disadvantage is the necessity of knowledge of all parameters of induction motor equivalent circuit.

Acknowledgment

This material is based on the work supported by the Technology Agency of the Czech Republic under the grant for Competence Centres programme project No. TE02000103 and on the work supported by the Student Grant Agency of Czech Technical University in Prague under grant No. SGS16/152/0H/3/2T/13.

References