

A Comparison of Stator Flux Linkage Estimators for a Direct Torque Controlled PMSM Drive

T.J. Vyncke, R.K. Boel and J.A.A. Melkebeek

Department of Electrical Energy, Systems and Automation (EESA)

Ghent University (UGent), Sint-Pietersnieuwstraat 41, B-9000 Gent, Belgium

phone: +32 (0)9 264 3442, fax: +32 (0)9 264 3582, e-mail: Thomas.Vyncke@UGent.be

Abstract—In an increasing number of applications highly dynamic electrical drives, characterized by high quality torque control, are demanded. Direct torque control (DTC) for AC machines, permanent magnet synchronous motors (PMSM) or induction machines, can provide this accurate and fast torque control.

When applying DTC the change of the stator flux linkage vector is controlled, based on torque and flux errors. As such the estimation of the stator flux linkage is essential for a DTC drive. Furthermore the quality of the estimation directly determines the capability of the drive.

In the literature several possible solutions for the estimation of the stator flux linkage are proposed. However, a comprehensive comparison between these solutions is not present. This paper gives an overview of several techniques for the estimation of the stator flux linkage for DTC in PMSMs. The theoretical advantages and disadvantages of the methods are outlined. After a short discussion on the effects of erroneous estimations the results from simulations for the different methods are reviewed. It is shown that, despite their simplicity stabilized voltage model methods can offer good performance. Still they can not reach the performance of an extended Kalman filter implementation of a current model. Aspects of the practical implementation on FPGA are discussed.

I. INTRODUCTION

The use of highly dynamic electrical drives in a wide variety of applications has increased steadily in recent years. Within this market AC machines, and recently especially permanent magnet synchronous machines (PMSM's), have obtained dominance due to their characteristics of high efficiency, high power density and reliability. These highly dynamic electrical drives have to provide accurate and fast torque control together with the highest possible efficiency.

To this end rotor flux field oriented control has become the de facto industry standard to control the torque and flux levels of AC machines. For induction motors (IM's) however an alternative control scheme, direct torque control (DTC), was proposed in [1] and became very popular in the past two decades [2]. The basic principle of DTC is to directly select stator voltage vectors according to the differences between actual and reference value for torque and stator flux linkage in a stationary reference frame. As such DTC for induction machines is inherently motion-state sensorless. In the past decade several authors [3]–[6] have proposed ways to adapt DTC to work with PMSM's.

The basic principles of DTC, i.e. controlling the change of the amplitude and the position of the stator flux linkage vector, are given in section II. As such the stator flux linkage vector has to be estimated with a high accuracy. Several techniques to achieve this estimation have been reported in literature [7]–[13], however a comprehensive comparison of these methods is lacking. In this paper such an overview is given, therefore several methods are discussed in section III. In order to compare the obtained results in simulations (section V) a short discussion on the effects of erroneous estimation is given in section IV. Section VI gives details about the implementation on FPGA.

II. PRINCIPLES OF DTC PMSM

The electromagnetic torque T of an PMSM, is given by

$$T = \frac{3N_p}{4L_d L_q} (2|\underline{\Psi}_f| L_q \sin \delta - |\underline{\Psi}_s| (L_q - L_d) \sin 2\delta) \quad (1)$$

where δ denotes the load angle, the angle between the stator flux linkage $\underline{\Psi}_s$ and permanent magnet flux linkage $\underline{\Psi}_f$ vectors in the stationary α/β reference frame. The number of pole pairs is denoted by N_p , L_d and L_q are the direct and quadrature stator inductances respectively. The equation is valid for PMSM with pole-saliency, where $L_d < L_q$, and for PMSM with a uniform air gap, where due to $L_d = L_q = L_s$ the expression for torque is simplified. From (1) can be seen that for constant stator flux linkage, the torque is changed by changing the load angle δ . The stator flux vector can be changed by applying a voltage vector from the inverter:

$$\underline{\Psi}_s = \int_0^t (\underline{V}_s - R_s \underline{I}_s) dt + \underline{\Psi}_s|_{t=0} \quad (2)$$

Any of the six active voltage vectors has a radial and a tangential component with respect to the flux vector. The first changes the amplitude of the stator flux linkage. The second changes the rotation speed of the stator flux vector and thus the load angle. In basic DTC, the differences between actual and reference value for torque and stator flux linkage are supplied to hysteresis controllers. With these error signals and the sector in which the vector is situated, the most appropriate voltage vector follows from a switching table. Other implementations of DTC calculate a most appropriate voltage vector in a corrective or predictive fashion and realize it by means of space vector modulation. Either way the flux level is controlled by controlling the stator flux linkage amplitude and the torque is controlled by the angular change of the stator flux linkage vector. As such the knowledge of the stator flux linkage vector is crucial for a correct operation of the drive.

III. STATOR FLUX LINKAGE ESTIMATION METHODS

As it is impossible to measure the stator flux linkage, it has to be estimated. In theory equation (2) can be used for this estimation when stator voltages and currents are measured. The use of a pure open-loop integration however has its disadvantages, as discussed further on.

In the literature a number of different methods are proposed, a short overview is given in [8] and [7]. Most can be divided either in the class of voltage model based methods or the current model based methods. The first are based on the voltage model of the machine, given by equation (2) and offer as distinct advantages the independence on the rotor position and the fact that only one parameter (the stator resistance R_s) needs to be known. However the value of R_s has to be adapted online or the detrimental effects of the incorrect value of R_s need to be eliminated.

Open-loop integrators are not stable and thus the integrator has to be stabilized by an external adaptation (making the method

closed-loop). At this point it is also useful to note that, unlike in IM's, the initial value of the stator flux vector $\underline{\Psi}_s|_{t=0}$ is not zero in PMSM's and is determined by the rotor position. As a result it has to be measured or estimated for a correct starting of the drive, preferably in a sensorless fashion.

The current model based methods are, for a surface permanent magnet synchronous machine (SPMSM) in the stationary $\alpha\beta$ reference frame, defined by:

$$\Psi_\alpha = LI_\alpha + \Psi_f \cos(\theta) \quad (3)$$

$$\Psi_\beta = LI_\beta + \Psi_f \sin(\theta). \quad (4)$$

As is clear from equations (3)-(4), these methods are dependent on the rotor position and the stator inductance. The resulting need for a position sensor is, especially in DTC which is an inherently position sensorless method, considered as a major disadvantage. Also the increased parameter dependence on the inductances is, considering the saturation, a disadvantage.

A. Voltage model based methods

Several estimation techniques have been reported, here we firstly discuss two open-loop methods (integrator and low-pass filter) and subsequently two closed-loop versions. The main idea for the closed-loop versions is to use the integrator, but to add a stabilizing feedback.

1) *Open-loop integrator*: The calculation of the stator flux linkage vector is done by integrating the EMF as shown in equation (2). Three main problems occur:

- any offset in the measurements of voltages or currents leads to large drifts in the estimated stator flux linkage
- if the integrator is initialized with the wrong value (e.g. the initial rotor position), a DC offset is present at the output of the integrator
- a wrong value of the stator resistance R_s will cause significant errors, especially at low speeds and high loads.

2) *Open-loop LPF*: In order to overcome the problems of the pure integrator, often an open-loop low-pass filter is used, for which the transfer function is given by:

$$\frac{Y}{X} = \frac{1}{1 + j\tau\omega} \quad (5)$$

where τ denotes the time constant and ω denotes the frequency. As an LPF is equivalent with a high-pass filter followed by an integrator, the DC offsets are removed and drift is avoided. When using an LPF great care should be given to the selection of the cut-off frequency. A low cut-off frequency means a high time constant so the flux estimation is too slow, when however a high cut-off frequency is selected the lower limit of the speed operating range of the drive is increased. The phase shift and gain of the filter at frequency ω

$$\phi = \text{atan}(\tau\omega) \quad (6)$$

$$K = \frac{1}{\sqrt{1 + (\tau\omega)^2}} \quad (7)$$

determine the performance of this estimation method as the resulting attenuation and phase lag of the estimated flux vector are the main disadvantages. Methods [14] exist to partially compensate these, however most are based on steady-state considerations and do not guarantee good transient behavior.

3) *Closed-loop integrator with PCLPF*: In [8] a programmable cascade of low-pass filters (PCLPF) is proposed to reset a pure integrator and thus remove the offset errors, however no details are given on the implementation or obtained results. As detailed in [15], a PCLPF is obtained when n low-pass filters with software-settable gain and time constant are put in series. The total phase shift and gain for n identical

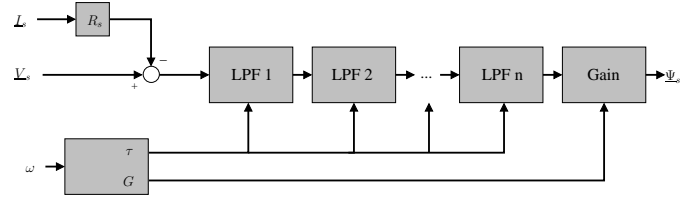


Figure 1: Programmable cascade of low-pass filters

LPF's are given by

$$\phi_T = n\phi = n \text{atan}(\tau\omega) \quad (8)$$

$$K_T = nK = \frac{1}{\sqrt{(1 + (\tau_1\omega)^2)^n}} \quad (9)$$

For a pure integrator it is well known that the phase lag is $\pi/2$ and that the gain is $1/\omega$. For the PCLPF to behave as an integrator at the fundamental frequency, the following conditions for the time constant and compensation gain G have to be programmed:

$$\tau = \frac{\tan(\frac{\pi/2}{n})}{\omega} \quad (10)$$

$$G = \frac{\sqrt{(1 + (\tau_1\omega)^2)^n}}{\omega} \quad (11)$$

In order to use the PCLPF we have to obtain the fundamental frequency of the voltage signal, which is possible by calculating the stator flux vector speed or the rotor speed, but special attention needs to be given to transient situations. As is clear in equations (10)-(11), the time constant and gain approach infinity as ω approaches zero. As such the operation of the PCLPF becomes troublesome at low frequencies.

The PCLPF behaves as an integrator at the fundamental frequency of the voltage (solving the problems of phase lag and attenuation), while still eliminating DC components that cause drift. The price to pay for this performance is a higher computational load and the loss of the higher frequency components in the stator flux linkage. The PCLPF calculates the stator flux linkage at the fundamental frequency, but not the instantaneous flux. While this can be useful in certain algorithms, it does not allow to keep the stator flux magnitude within the hysteresis band of a switching-table DTC.

The result of the PCLPF will however not drift, so it gives a reference for the pure integrator. The PCLPF can then be used to re-initialize the pure integrator, either at fixed time-intervals or whenever a certain threshold is crossed. In the simulations (section V) the algorithm resets the integrator to the value of the PCLPF every time the euclidian distance between the tips of the stator flux linkage vectors according to the integrator and the PCLPF is higher than 2 times the hysteresis threshold¹. At very low speeds the reset action is suppressed. Here again the algorithm is sensitive to errors in the stator resistance.

4) *Closed-loop integrator with PI*: As is shown in the previous item, pure integrators can be used if they are stabilized by external feedback. In [12] the input for the integrator $\underline{V}_s - R_s \underline{I}_s$ is augmented with a corrective input \underline{E}_k , as shown in figure 2. The correction \underline{E}_k is the output of an PI-compensator, which acts on the difference between the estimated and the 'real' stator flux linkage vector. The real stator flux linkage vector is approximated by a vector with the same phase as the estimated vector and as magnitude the reference value for the stator flux magnitude.

In [12] it is claimed that the flux estimation is thus forced on

¹If the flux vector phase error is considered more severe than the amplitude error, the measure can be adapted so that deviations between integrator and PCLPF are constrained within an ellipse rather than a circle.

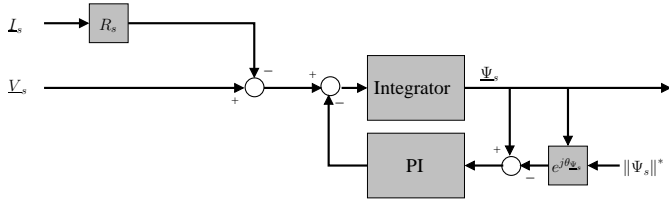


Figure 2: Closed-loop integrator with PI

a circular trajectory with asymptotic phase convergence. This closed-loop integrator clearly cancels out the drift problems and even copes with wrong values for the stator resistance. However, the transient behavior and robustness of the method have to be demonstrated. The possibly long settling time for the flux magnitude and especially the phase are to be considered as drawbacks.

5) *The stator resistance:* Due to skin effect and temperature changes, the stator resistance R_s can have significant variations resulting in large errors in the estimated flux. Even if this is corrected in closed-loop methods, transient errors occur. As such the adaptation of R_s is often needed or desirable. A thermal model of the machine can be used to update the value of the stator resistance. Several techniques for online stator resistance estimation are described in [8], [16] and [17] Most of these methods use either PI or fuzzy logic estimators to obtain a value for the stator resistance.

Although these algorithms can offer good results, clearly:

- flux estimation methods insensitive to changes in R_s are still preferable,
- estimation of R_s adds complexity,
- attention should be given to the parameter dependency (on rotor position and inductances) introduced by these algorithms. For example, the methods in [8], [16] are dependent on L_d and L_q , however the influence of saturation is not discussed.

B. Current model based methods

In order to avoid the problems associated with the integration of the back-emf, the stator flux linkage can be calculated by using the current model. In this section we discuss the open-loop current model as well as two implementations of the current model in an extended Kalman filter.

1) *Open-loop current model:* The open-loop current model is defined by the equations (3)-(4). It is independent from the stator resistance and an offset will not lead to drift in the flux estimation. However it is dependent on the stator inductances and the rotor position. The need to measure or estimate the rotor position is clearly a disadvantage. The stator inductances and rotor flux are also variable as they depend on the magnetic saturation in the machine.

2) *Extended Kalman filter current model:* To reduce the parameter dependence and to perform the rotor position estimation needed in the current model an observer can be used. Several observers have been used, a short overview is given in [7], [8]. Here the extended Kalman filter is selected.

The Kalman filter is a stochastic recursive optimum-state estimator. For nonlinear systems an extended Kalman filter (EKF) has to be used. The EKF is used to obtain unmeasurable states (e.g. speed and rotor position) by using a model for the dynamical system, measured states and statistics of the system and measurement noise. By means of the noise input it is possible to take account of both measuring errors and modelling errors.

Here only the basics of the EKF will be given, a more comprehensive discussion can be found in [7]. The EKF is a two-step method as shown in figure 3. With the measured

inputs u_k and machine model ($f(x, u)$ and $h(x)$) the next state of the machine \hat{x}_{k+1} is predicted (prediction step). From this state the next output is calculated and compared to the measured value. The error on the output, together with the covariance values of measurement noise R and system Q are used to correct the state values (correction or innovation step), in which matrix P and the Kalman gain matrix K are calculated. In this paper two implementations of the EKF are

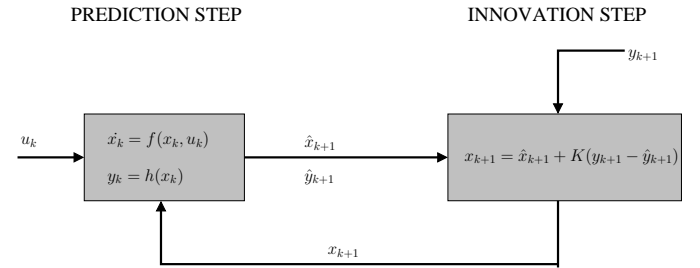


Figure 3: EKF

studied. The difference is based on the selection of the state variables. In EKF1 the current components in the stationary reference frame are selected as state variables, $x = [I_\alpha \ I_\beta \ \omega \ \theta]$ as in [7], [9]. In EKF2 the stator flux linkage components in the stationary reference frame are selected as state variables, $x = [\Psi_\alpha \ \Psi_\beta \ \omega \ \theta]$ as in [10]. In both cases the current components in the stationary reference frame $y = [I_\alpha \ I_\beta]$ are selected as output. The inertia is assumed to be infinite, reducing the mechanical equation. Because speed ω and rotor position θ are in the state vector, a good value for the covariance will ensure the correction of this modelling error by the EKF. Some implementation details are given in appendix. More details about the tuning of EKF's can be found in [18].

IV. HOW TO COMPARE THE DIFFERENT FLUX ESTIMATORS

In order to compare the different estimation methods discussed previously, it is necessary to take a closer look on how errors in the stator flux linkage estimation affect the quality of the torque control.

A. Effects of errors in flux estimations

In [19] a short discussion on this subject is given for the specific case of an LPF as stator flux estimator in a switching-table based DTC. The authors conclude that three effects occur. Firstly, due to the attenuation by the filter the controller will try to increase the actual flux above the reference value. Secondly the phase shift can result in the selection of the wrong sector and thus the wrong voltage vector, leading to a reduced magnitude of the actual flux at sector crossings. Thirdly, also due to the phase shift, sixth harmonic torque ripples are introduced in steady state.

In a more general situation we can make the distinction between two distinct pathways through which the errors in the flux estimation will deteriorate the torque and flux control. A first pathway is where the errors in the flux estimation result in incorrect values for the controlled variables. The estimated stator flux magnitude follows directly from the estimated Ψ_α and Ψ_β , but also the estimated torque value is determined by the stator flux estimation:

$$T = \frac{3}{2} N_p (\Psi_\alpha I_\beta - \Psi_\beta I_\alpha) \quad (12)$$

With this pathway the error on the controlled variables will give rise to an erroneous correction (i.e. selection of the wrong inverter switching state). Both for switching-table based and SVM based DTC, the torque and flux magnitude will be corrected to a wrong value. The second pathway is through

R_s	2.875 Ω	L_s	8.5 mH	J	0.008 kgm ²
Ψ_f	0.175 Wb	N_p	4	F	0.001 Nms

Table I: Parameters of SPMSM used in simulation

the error in the estimated value for the angle of the stator flux linkage vector (θ_Ψ). The mechanism in the second pathway is different for switching-table based and SVM DTC. For switching-table DTC an error in θ_Ψ only has an effect during the crossing from one sector to another. If the estimated flux vector is situated in a different sector than the real stator flux vector, the response of the machine to the selected inverter switching state can result in large excursions from the reference values or even unstable operation (although in certain cases, e.g. at high speeds, the lag of the estimated vector can have a positive influence on the torque at sector crossing). However if the stator voltage is calculated in a corrective or predictive way, the error in the flux vector angle will result in erroneous voltage vectors at all times.

B. Measures of quality for the flux estimation

To discuss and compare the quality of the stator flux linkage estimation different situations have to be considered. In steady state (or situations about steady state) deviations in magnitude and phase of the estimated flux components Ψ_α and Ψ_β should be considered. However in transient operation not only the deviations in magnitude and phase are important, but certainly the reaction speed for the estimation is of great importance. As is clear from the previous discussion, the most relevant variables to compare however are the stator flux linkage vector magnitude and especially the angle θ_Ψ .

One could note that the true test for the quality of the flux estimators lies in the quality of the obtained torque control. While this is true, comparing results for the stator flux estimation is useful in studying the intrinsic capabilities of estimators; showing how erroneous working conditions and instability arise rather than merely demonstrating them.

V. RESULTS IN SIMULATION

The different estimation methods discussed in III have been implemented in Matlab/Simulink. All simulations are performed with the same switching-table based SPMSM DTC drive, where the control is executed with the values for the real stator flux in the machine (calculated). The estimators that we want to compare all use the same measurements and are running in parallel with the calculation. The difference in amplitude and angle between their respective outputs and the calculated stator flux vector amplitude and angle show the capabilities of the estimators. The parameters of the SPMSM are given in Table I.

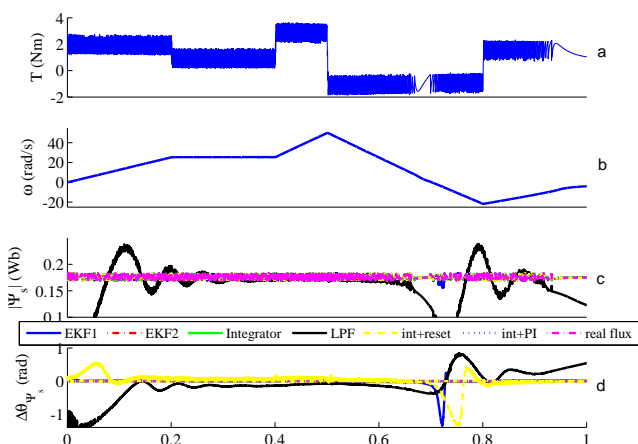


Figure 4: Highly dynamic operation of the estimators with DTC

A. Operation under ideal circumstances

When no parameter deviations or measurement disturbances are present and the estimation methods are correctly initialized, the maximum performance of the methods is obtained. The first results are shown for a very dynamic operation of the drive. Torque and speed trajectories are shown in figure 4a-4b. Also shown are the phase errors for the different methods (4d), as well as the amplitude (4c). It is clear that the LPF does not perform well under these highly dynamic conditions. The other voltage model based methods perform very good, they behave as pure integrators. The only exception is the integrator+PCLPF reset which has serious deviations in the phase estimation. In an experimental implementation it is important to disable the reset action during transients as the PCLPF will give rise to erroneous reset actions (as seen in figure 4d). The EKF estimators have some minor deviations in flux amplitude and angle in the transient areas. Especially the zero-crossing of the speed gives some disturbance for EKF1.

In the remainder of this section a less dynamic operation is selected, so that the methods can be evaluated and compared in steady state as well. The drive starts from stand-still and is accelerated with a constant torque of 2 Nm, then the torque reference is reduced to 1 Nm to maintain a constant speed. In the following comparisons the transient part is defined by the time needed to reach this constant speed (the run-up and the torque change), once the speed is constant (after about 0.5 seconds) steady state is reached. In steady state with ideal conditions the errors with all methods, except for the LPF, are negligible.

B. Influence of measurement disturbances

Voltage and current signal conditioning circuits and analog-to-digital converters can introduce a DC-offset to the measurement. Here the effect of this disturbance on the performance of the estimators is investigated. The DC-offset on the voltage measurements is 1% and -2% of the DC bus voltage for α and β respectively. The offset for the currents is 3% of the steady-state currents.

As is clear from figures 5 and 6, the pure integrator quickly drifts away from the correct value. The LPF does not diverge, but oscillates heavily about the correct values (the errors in magnitude and angle are bounded, in the $\alpha\beta$ reference frame this corresponds with a shift of the center point of the flux locus). The integrator is more or less successfully stabilized by the PCLPF-reset, however large excursions occur, both in magnitude and angle of the stator flux vector. During the run-up of the machine the reset action is suppressed as the PCLPF offers poor performance at low speeds. Thus the PCLPF-reset integrator is identical to the integrator during run-up and performs poorly. The only voltage-model based estimator that gives good results is the PI-stabilized integrator (if appropriate values for K_i and K_p are selected). Although rather large deviations occur in the magnitude during the transient, the steady-state estimation of both the magnitude and angle is very good. During the transient the angle estimation is as good as or better than the estimation with the EKF estimators. The EKF estimators retain a small error in the angle during steady state. In steady state they yield results for the magnitude that correspond with the real flux, only in the run-up transient some minor deviation occurs. The current model is not shown in the figures, as due to the low value of the inductance the estimation error is negligible. Practically only the PI-stabilized integrator, the EKF estimators and the current model are useable under these simulated conditions.

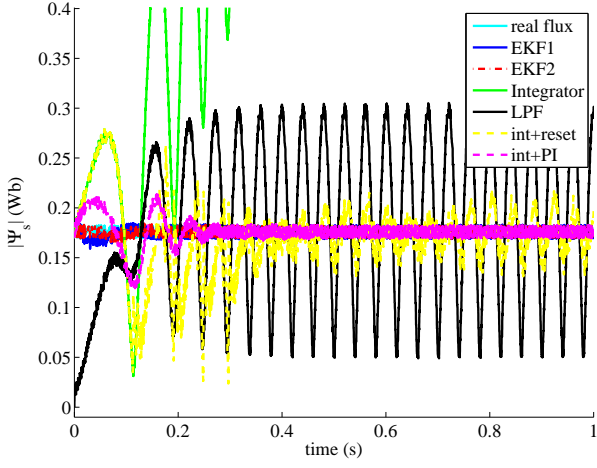


Figure 5: Estimated $|\Psi_s|$ for the different estimation methods

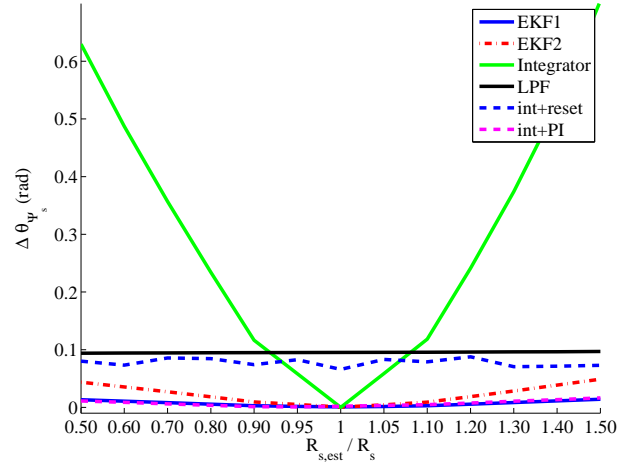


Figure 7: RMS error in the flux vector angle in steady state

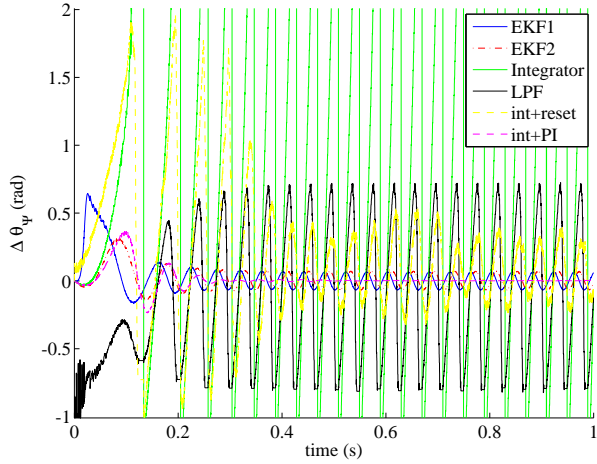


Figure 6: $\Delta\theta_{\Psi_s}$ for the different estimation methods

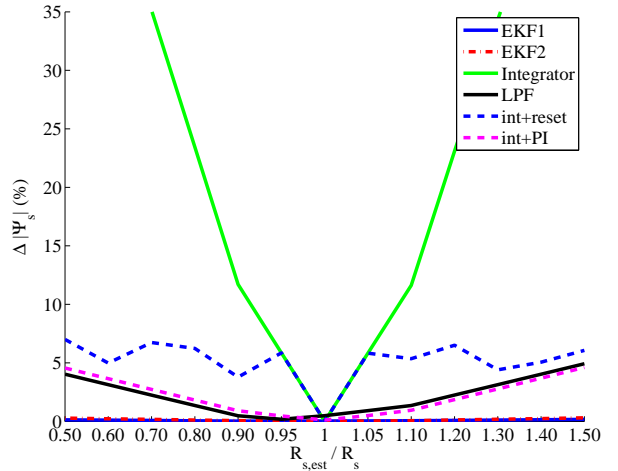


Figure 8: RMS error in the flux vector amplitude in steady state

C. Influence of parameter variations

First the effect of a wrong value for the stator resistance is evaluated. Except for the open-loop current model all estimation methods depend on this parameter. However, there is a large difference in the dependence of the estimator output on R_s as shown in the figure 7 where the RMS error is shown for the stator flux vector phase during steady state where $\frac{R_{s,est}}{R_s}$ is varied. Clearly the RMS angular error of the EKF estimators is very small (below 0.05 electrical radians), even for large deviations of R_s . The angular error introduced by the LPF is, as expected for steady state, constant and corresponds with the phase-lag which is independent of R_s . The pure integrator clearly gives poor performance. If the integrator is stabilized with the PI-compensator it is clear that the canceling of the effect of the incorrect value for R_s is perfectly able to stabilize the integrator and yields results comparable to the EKF estimators. Stabilizing the integrator with a reset based on the PCLPF gives identical results as the pure integrator for low deviations of R_s , as the hysteresis based reset is not yet active. For higher deviations it improves the angular estimation. In steady state the RMS errors of the stator flux amplitude are shown in figure 8. The EKF estimators outperform the other methods as they are the only estimators to have an almost zero RMS error in the magnitude. The PI-stabilized integrator yields results comparable to the LPF.

Until now only steady-state has been considered. The errors on both angle and amplitude for the EKF's are no longer negligible at higher deviations of R_s , however they outperform the other methods in this transient situation (the EKF estimators have an angular error of less than $\frac{\pi}{3}$ at all times and the errors are quickly reduced). The transient performance of the LPF is

very poor (the LPF estimator is known for troublesome starting of the drive). Reset stabilizing barely offers improvements compared to the pure integrator (note that the reset is disabled at low speeds). The PI stabilization of the integrator performs reasonably as far as the magnitude is concerned. However the angular error can be much larger than the value for the pure integrator, the large angular deviations and the time needed to reduce the angular error make it practically inapplicable. Clearly the voltage model based methods need an algorithm to adaptively update R_s in order to give reasonable performance. The EKF estimators (for which additionally estimating the stator resistance would be easily done if needed) are very robust to variations of R_s .

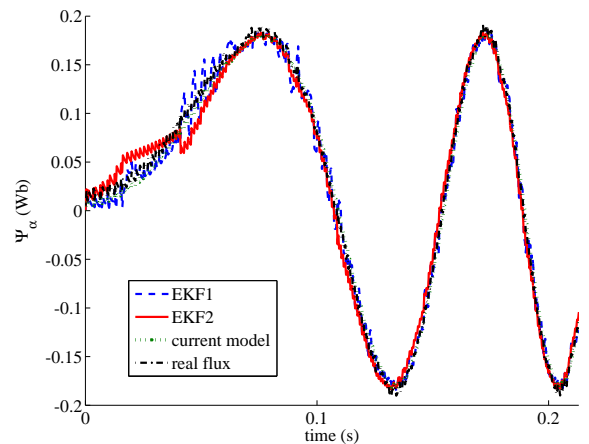


Figure 9: Estimated Ψ_α for EKF1, EKF2 and the current model compared to the 'real' flux. $L_{s,est} = \frac{L_s}{2}$

	EKF1	EKF2	CM
RMS amplitude error (%)	2.475	2.304	2.543
RMS phase error (rad)	0.0657	0.0728	0.0286

Table II: RMS errors of amplitude and phase

The methods based on the current model are however also dependent on the synchronous inductance L_s . In figure 9 the estimated flux component Ψ_α is shown for EKF1, EKF2, the open-loop current model and the correct flux calculation for a situation where the estimated inductance is $L_{s,est} = \frac{L_s}{2}$. Both EKF1 and EKF2 converge quickly to values close to the estimated values with the open-loop current model, but all methods have similar errors when compared to the real flux, as shown in figures 10 and 11. The RMS-values of the errors are found in table II. Clearly the estimation with EKF1 and EKF2 does not yield better results than the open-loop current model. Still the estimation is performed in a sensorless fashion, which is a considerable advantage.

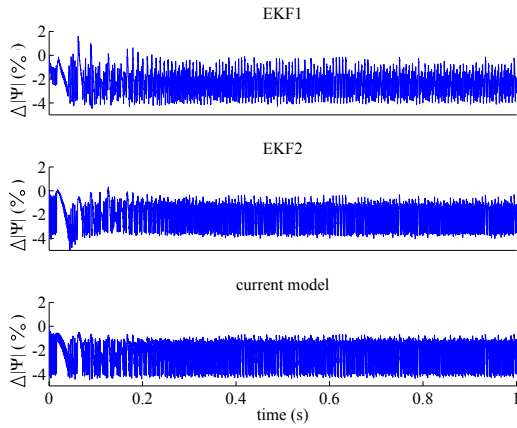


Figure 10: Error in percent of the estimated $|\Psi_s|$ for EKF1, EKF2 and the current model, $L_{s,est} = \frac{L_s}{2}$

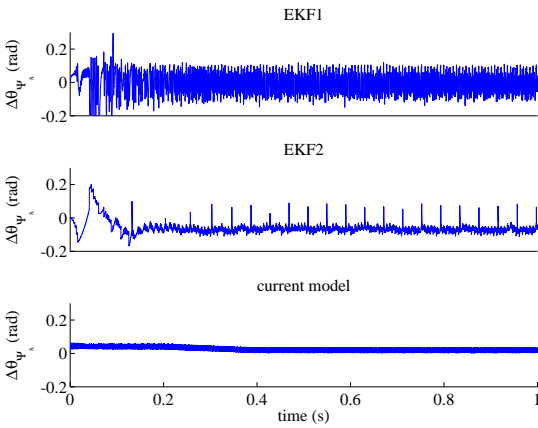


Figure 11: Estimated phase difference $\Delta\theta_{\Psi_s}$ between EKF1, EKF2 and the current model and the real flux respectively, $L_{s,est} = \frac{L_s}{2}$

Clearly the EKF estimators can cope very well with errors in R_s , but variations in L_s result in stable operation with considerable deviations in steady-state (same magnitude as the open-loop current model). When R_s is varied, the EKF estimators can correct for this modelling inaccuracy by the feedback loop. For variations in L_s (and Ψ_f) however this is not the case as the parameter L_s , unlike R_s , is not only used in $f(x)$ but also in $h(x)$ or $o(x)$. For EKF1 the state vector x will converge to the correct values, but due to the use of L_s in determining Ψ_α and Ψ_β from x (function $o(x)$, see appendix) the output is incorrect. For EKF2 L_s is used in $h(x)$ and thus the state vector x will not converge to the right value.

D. Influence of initial conditions

In figure 12 it is assumed that the initial stator flux vector position is correctly identified, however with an incorrect amplitude (half of the correct value). It can be seen in the figure that the LPF is not affected as it does not use the initial condition (but offers poor transient performance as always). The pure integrator, due to the constant offset in one flux component, heavily oscillates about the correct value. When stabilized with a reset by PCLPF, the integrator is rather quickly forced to correct values (the first 0.15 s this estimator follows the pure integrator, once the PCLPF reset becomes active the correct values are quickly obtained). If stabilized by an PI compensator the offset also is quickly reduced and the correct value is obtained. EKF1 is not affected by this change in initial conditions and EKF2 (where the flux components are the state) converges almost immediately.

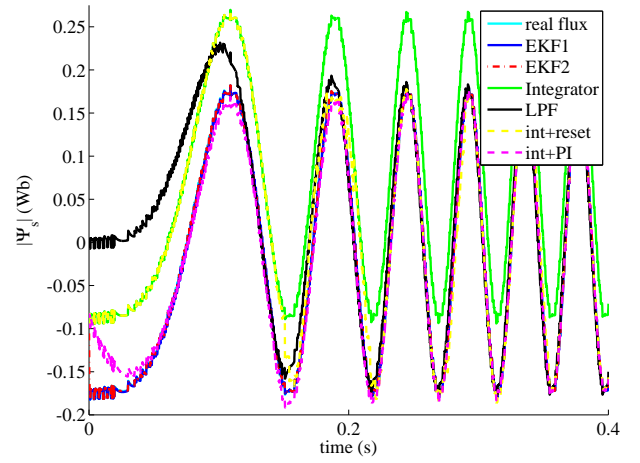


Figure 12: Ψ_β for different methods $\Psi_{\beta,init,est} = \frac{\Psi_{\beta,init}}{2}$

A more severe error in initial conditions is presented when the flux linkage is obtained correctly but the position estimation is wrong. In the simulation of figures 13 and 14 an error of $\frac{\pi}{2}$ radians is chosen. For the voltage model based methods similar conclusions can be drawn as in the previous case: both PI and PCLPF are able to stabilize the integrator, but with large deviations in both the phase and magnitude estimation. Especially the PI-stabilized integrator now offers quite poor results compared to the EKF estimators. The EKF estimators now both have incorrect initial conditions, but converge very quickly (the phase error is negligible within 0.025s and 0.1 s for EKF1 and EKF2 respectively).

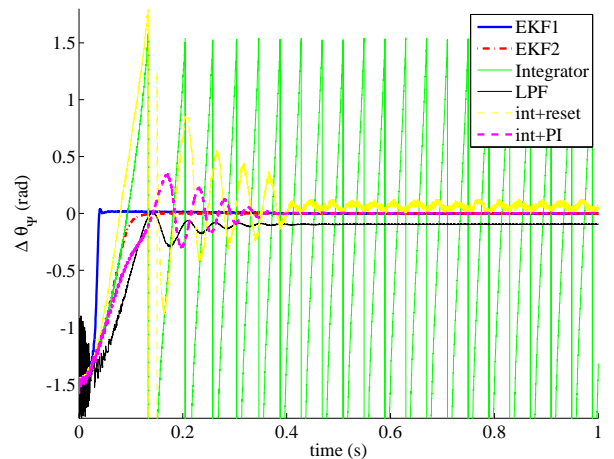


Figure 13: $\Delta\theta_{\Psi_s}$ for different methods, error on $\theta_{\Psi_s,init} = \frac{\pi}{2}$

	integrator	LPF	int+PI	int+reset	current model	EKF1	EKF2	remarks
measurement or separate estimation of θ needed	no / initial	no / initial	no / initial	no / initial *	yes	no / (initial)	no / (initial)	* needs angular frequency
dependence on R_s	very high	high	high	high - very high	none	very low	very low	
dependence on L_s	none	none	none	none	high	high	high	
dependence on Ψ_f	none	none	none	none	high	high	high	
effect of DC-offset	instability	practically unusable	disturbed in transient	recurring deviation-reset sequences	deviation depends on $\frac{L_s}{\Psi_f}$	small	small	
preferred applications	can not be used	almost not usable (very low dynamic, torque-ripple tolerant applications) *	less suited for highly dynamic operation *	- less suited for highly dynamic operation - more attractive if rotor speed is known *	- if rotor position is measured anyway - if very stable and smooth low-speed operation is desired	- high-end highly dynamic with strict torque tolerance - powerful DSP or FPGA	- high-end highly dynamic with strict torque tolerance - powerful DSP or FPGA	* R_s estimation (e.g. temperature sensor available) makes these options more attractive

Table III: Overview of the comparison of estimators,

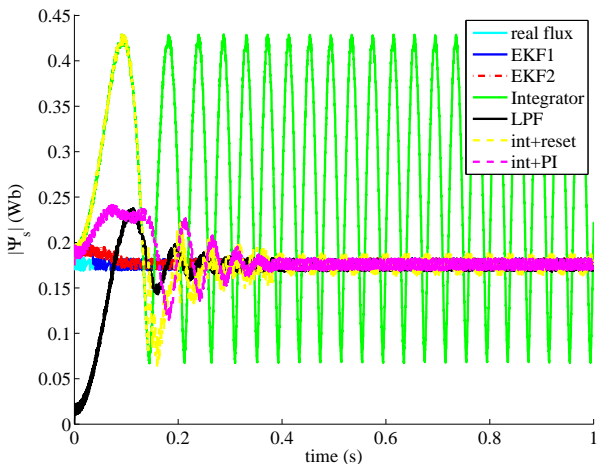


Figure 14: $|\Psi_s|$ for different methods, error on $\theta_{\Psi_s, init} = \frac{\pi}{2}$

VI. PRACTICAL IMPLEMENTATION DETAILS FOR FPGA

For the digital implementation of the stator flux linkage estimators a Spartan 3E Starter Board from Digilent Inc., based on the Xilinx XC3S500EFG320 FPGA (500K gates, 10476 logic cells), is used. Besides the versatility offered by using an FPGA, the test platform allows for a parallel implementation of the estimators so that all estimators could be compared under exactly the same circumstances.

The configuration of the FPGA is programmed in Matlab/Simulink with the System Generator (SG) tool from Xilinx. This tool allows to build up the entire functionality of the FPGA as a typical Simulink graphical block diagram by using elements from the Xilinx Block Set. Once the block diagram is finished and simulated it can be generated by System Generator into a bit stream for the FPGA. Some specific functions (acquiring values from the analog-to-digital convertors (ADCs), user interaction by LCD and sending values to the digital-to-analog convertors (DACs)) are written in VHDL and interfaced with the rest of the block diagram through the Black Box block.

A. Implementation of voltage-model based and open-loop current-model based estimators

As the voltage-model based estimators and the open-loop current model estimator are quite straightforward to implement, this is done by using standard SG blocks (Accumulator, AddSub, Cmult, Delay). The only part which is less straightforward is the PCLPF as it needs the stator voltage angular frequency. In order to avoid the implementation of some algorithm to extract the angular frequency from the measurements, the rotor speed (which is known as the rotor position is measured for the open-loop current model) is used as a good approximation in steady-state (in transient the PCLPF reset is preferably disabled anyway).

B. Implementation of EKF1 and EKF2

The EKF is much harder to implement with the standard SG blocks. A first possibility for the implementation is to instantiate a Microblaze 32-bit RISC soft processor within the FPGA (using logic primitives). The Microblaze soft processor can be instantiated with a Floating Point Unit (FPU), in this way 32 bit floating point variables can be used. This simplifies the implementation of the EKF as the round-off error (which is a common source of degradation or instability of the EKF, see [20]) is less of a concern compared to the situation where fixed-point arithmetics are used. Of course this comes at a certain cost: instantiating the Microblaze with an FPU will take up about half of the FPGA resources, including about 35 % of the 20 18x18 multipliers.

A second interesting implementation is obtained by using the AccelDSP synthesis tool from Xilinx. This is a high-level MATLAB language based tool for designing DSP blocks for Xilinx FPGAs. As such a Matlab implementation of the EKF can be automatically converted from floating- to fixed-point arithmetic, verified (to see if the conversion does not degrade the EKF performance too much), and synthesized into VHDL or a custom SG block. Handing in the luxury of the floating-point arithmetic can result in a serious reduction of needed system resources (if a not too naive implementation, i.e. all the calculations are optimized before the conversion, is used).

Both the implementation using the Microblaze processor as the implementation with AccelDSP can be imported into the total System Generator block diagram.

VII. CONCLUSIONS

In this paper an overview of several techniques for the estimation of the stator flux linkage is given. Due to the large number of estimators proposed in literature no attempt is made to discuss all the possibilities, but a selection of interesting methods is made. The results from the comparison are summarized in Table III.

Clearly the EKF-implementations, although they use more parameters, exhibit a lesser dependence on R_s and initial conditions (as long as the EKF converges) than the voltage model based implementations. Furthermore they are capable of handling measurement offset better. On the other hand they are much more complicated to design correctly.

Closed-loop integrator methods can also be very powerful (especially the PI-stabilization), given their simplicity. However, a good stator resistance estimation scheme, which is still needed or desirable for these methods, also increases the complexity. While the results in steady state are quite accurate, the considerable transient errors are the largest drawback as highly dynamic operation is desired in many applications were DTC drives are used.

Overall it is clear that the EKF, if correctly designed and implemented, can yield the best results. Still the dependence on L_s and Ψ_f should be addressed, as the saturation in the machine otherwise makes the EKF operation very troublesome. Previous papers in literature have not mentioned this problem. Results on this topic will be published in the near future.

All the methods have advantages and disadvantages, but most of them are usable if only the characteristics of the estimator match the application profile. To this end some possible links between application and estimator are discussed in Table III as well. Application aspects that influence the choice of estimator are:

- allowable torque ripple during steady state
- importance of dynamic operation
- motor characteristics (high or low L_s , Ψ_f , ...)
- sensors available (temperature, position, phase voltage,...)
- time to implement the control

Further work includes finalizing the comparison under experimental conditions and expanding the scope of the study to PMSMs with saliency.

ACKNOWLEDGMENT

T. Vyncke wishes to thank the Research Foundation-Flanders for his grant as Ph. D. fellowship of the Research Foundation - Flanders (FWO).

This work was supported by the GOA project BOF 07/GOA/006. The research was performed as part of the Interuniversity Attraction Poles programme IUAP P6/21 financed by the Belgian government.

APPENDIX

Both EKFs are initialized with a zero matrix for P , covariance matrices Q and R for EKF1, EKF2 respectively are:

$$Q = \text{diag}(0.01 \ 0.01 \ 0.05 \ 0.01), \quad R = \text{diag}(10 \ 10)$$

$$Q = \text{diag}(0.0001 \ 0.0001 \ 1000 \ 0.1), \quad R = \text{diag}(10 \ 10)$$

System and output functions for EKF1 ($x = [I_\alpha \ I_\beta \ \omega \ \theta]$):

$$f(x) = \begin{bmatrix} -\frac{R_s}{L_s}x_1 + \frac{\Psi_f}{L_s}x_3 \cos x_4 \\ -\frac{R_s}{L_s}x_2 + \frac{\Psi_f}{L_s}x_3 \sin x_4 \\ 0 \\ x_3 \end{bmatrix} \quad (13)$$

$$h(x) = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad o(x) = \begin{bmatrix} L_s x_1 + \Psi_f \cos x_4 \\ L_s x_2 + \Psi_f \sin x_4 \end{bmatrix} \quad (14)$$

System and output functions for EKF2 ($x = [\Psi_\alpha \ \Psi_\beta \ \omega \ \theta]$):

$$f(x) = \begin{bmatrix} -\frac{R_s}{L_s}x_1 + \frac{R_s}{L_s}\Psi_f \cos x_4 \\ -\frac{R_s}{L_s}x_2 + \frac{R_s}{L_s}\Psi_f \sin x_4 \\ 0 \\ x_3 \end{bmatrix} \quad (15)$$

$$h(x) = \begin{bmatrix} x_1 - \Psi_f \cos x_4 \\ x_2 - \Psi_f \sin x_4 \\ L_s \end{bmatrix} \quad o(x) = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (16)$$

REFERENCES

- [1] I. Takahashi and T. Noguchi, "A new quick-response and high-efficiency control strategy of an induction motor," *IEEE Trans. Ind. Applicat.*, vol. 22, no. 5, pp. 820–827, Sept./Oct. 1986.
- [2] G. S. Buja and M. P. Kazmierkowski, "Direct torque control of PWM inverter-fed AC motors – a survey," *IEEE Trans. Ind. Electron.*, vol. 51, no. 4, pp. 744–757, Aug. 2004.
- [3] L. Zhong, M. F. Rahman, W. Y. Hu, and K. W. Lim, "Analysis of direct torque control in permanent magnet synchronous motor drives," *IEEE Trans. Power Electron.*, vol. 12, no. 3, pp. 528–536, May 1997.
- [4] M. Niemelä, J. Luukko, and J. Pyrhönen, "Position sensorless PMSM DTC-drive for industrial applications," in *Conf. Proc. EPE*, Graz, 2001, p. 10.
- [5] M. F. Rahman, L. Zhong, M. E. Haque, and M. Rahman, "A direct torque-controlled interior permanent-magnet synchronous motor drive without a speed sensor," *IEEE Trans. Energy Conversion*, vol. 18, no. 1, pp. 17–22, Mar. 2003.
- [6] D. Swierczynski and M. P. Kazmierkowski, "Direct torque control of permanent magnet synchronous motor (PMSM) using space vector modulation (DTC-SVM) – simulation and experimental results," in *Conf. Proc. IEEE 28th Annual Conference of the Industrial Electronics Society (IECON'02)*, vol. 1, Nov. 5–8, 2002, pp. 751–755.
- [7] P. Vas, *Sensorless Vector and Direct Torque Control*. New York: Oxford University Press, 1998, pp. 122–178.
- [8] M. F. Rahman, M. E. Haque, L. Tang, and L. Zhong, "Problems associated with the direct torque control of an interior permanent-magnet synchronous motor drive and their remedies," *IEEE Trans. Ind. Electron.*, vol. 51, no. 4, pp. 799–809, Aug. 2004.
- [9] A. Llor, J. Rétif, X. Lin-Shi, and S. Arnalte, "Direct stator flux linkage control technique for a permanent magnet synchronous machine," in *Conf. Rec. IEEE 34th Annual Power Electronics Specialists Conference (PESC'03)*, vol. 1, June 15–19, 2003, pp. 246–250.
- [10] V. Comnac, M. Cernat, F. Moldoveanu, and I. Draghici, "Sensorless speed and direct torque control of surface permanent magnet synchronous machines using an extended kalman filter," in *Conf. Proc. 9th Mediterranean Conference on Control and Automation (MED'01)*, Dubrovnik, Croatia, June 27–29, 2001, p. 6.
- [11] J. Luukko, M. Niemelä, and J. Pyrhönen, "Estimation of the flux linkage in a direct-torque-controlled drive," *IEEE Trans. Ind. Electron.*, vol. 50, no. 2, pp. 283–287, Apr. 2003.
- [12] G. D. Andreescu and A. Popa, "Flux estimator based on integrator with DC-offset correction loop for sensorless direct torque and flux control," in *Proc. 15th Int. Conf. on Electrical Machines ICEM 2002*, Bruges, Belgium, Aug. 2002, p. 6.
- [13] B. Lang, W. Liu, and G. Luo, "A new observer of stator flux linkage for permanent magnet synchronous motor based on kalman filter," in *Proc. 2nd IEEE Conf. on Industrial Electronics and Applications*, Harbin, China, May 2007, pp. 1813 – 1817.
- [14] M. Hinkkanen and J. Luomi, "Modified integrator for voltage model flux estimation of induction motors," *IEEE Trans. Ind. Electron.*, vol. 50, no. 4, Aug. 2003.
- [15] B. K. Bose and N. R. Patel, "A programmable cascaded low-pass filter-based flux synthesis for a stator flux-oriented vector-controlled induction motor drive," *IEEE Trans. Ind. Electron.*, vol. 44, no. 1, pp. 140–143, Feb. 1997.
- [16] L. Tang and M. F. Rahman, "A novel proportional-integral (PI) stator resistance estimator for a direct torque controlled interior permanent magnet synchronous machine drive," in *Conf. Proc. IEEE International Electric Machines and Drives Conference (IEMDC'03)*, vol. 1, June 1–4, 2003, pp. 382–388.
- [17] S. Mir, M. E. Elbuluk, and D. S. Zinger, "PI and fuzzy estimators for tuning the stator resistance in direct torque control of induction machines," *IEEE Trans. Ind. Electron.*, vol. 13, no. 2, pp. 279–287, Mar. 1998.
- [18] S. Bolognani, L. Tubiana, and M. Zigliotto, "Extended Kalman filter tuning in sensorless PMSM drives," *IEEE Trans. Ind. Applicat.*, vol. 39, no. 6, pp. 1741–1747, Nov./Dec. 2003.
- [19] N. R. N. Idris and A. H. M. Yatim, "An improved stator flux estimation in steady-state operation for direct torque control of induction machines," *IEEE Trans. Ind. Applicat.*, vol. 38, no. 1, pp. 110–116, Jan./Feb. 2002.
- [20] M. S. Grewal and A. P. Andrews, *Kalman Filtering, Theory and Practice using MATLAB*. New York: Wiley-Interscience, 2001, pp. Chapter 6, 202–216.