

Fixed point implementation of Kalman filtering for AC drives: a case study using TMS320F24x DSP

Massimiliano Labbate, Roberto Petrella, Marco Tursini

Department of Electrical Engineering, University of L'Aquila

67040 Monteluco di Roio, L'Aquila, Italy

phone: +39-0862434446 – fax: +39-0862434403

e-mail: labmax@tiscalinet.it – petrella@ing.univaq.it – tursini@ing.univaq.it

Abstract

The real-time digital implementation of the Kalman filter requires a very fast signal processor specialised and optimised to perform complex mathematical calculations and manipulate a large amount of data. In fact, the algorithm is computationally intensive, and all of the steps involved require vector and/or matrix operation.

In this paper the analysis of an actual fixed point implementation of the Kalman filter is discussed based on a last generation μ C DSP (TMS320F240). The considered case study refers to rotor speed and position estimation in AC drives when using low resolution position transducers. It is shown that the adopted processor is suitable for an efficient and relatively simple implementation of the filter allowing to increase the resolution in speed calculation with respect to the classical differentiation solution.

Introduction

As known, the Kalman filter is an optimal recursive algorithm which provides the minimum variance state estimation for a time-varying linear system. It is able to tolerate system modelling and measurement errors, which are considered as noise processes in the state estimation. It processes all available measurements regardless of their precision, to provide a quick and accurate estimate of the variables of interest, also achieving a fast convergence. Its extension to non-linear systems, the Extended Kalman Filter (EKF), does not assure the minimum variance estimate and no convergence proof can be given. Nevertheless, the approach behaves well in most situations, as demonstrated by numerous applications (e.g. [1][2]).

For a straightforward application of this algorithm, the non-linear discrete-time state equations of the system are written in the following form:

$$\begin{aligned} \mathbf{x}_{k+1} &= \mathbf{f}(\mathbf{x}_k, \mathbf{u}_k, k) + \mathbf{w}_k & \text{cov}(\mathbf{w}) &= E(\mathbf{w}\mathbf{w}^T) = \mathbf{Q} \\ \mathbf{y}_k &= \mathbf{h}(\mathbf{x}_k, \mathbf{u}_k, k) + \mathbf{v}_k & \text{cov}(\mathbf{v}) &= E(\mathbf{v}\mathbf{v}^T) = \mathbf{R} \end{aligned} \quad (1)$$

where \mathbf{x}_k is the system state vector, \mathbf{y}_k is the system output, \mathbf{u}_k is the system input, \mathbf{w}_k and \mathbf{v}_k are zero-mean white Gaussian additive noises with covariance \mathbf{Q} and \mathbf{R} , respectively (independent from the system state \mathbf{x}_k). The vector \mathbf{w}_k takes into account the system disturbances and model inaccuracies, while \mathbf{v}_k represents the measurement noise. A block diagram of the EKF for the system (1) is given in Fig. 1 together with the list of the steps of a recursive implementation [3]. The filter provides a first estimate of \mathbf{x} ($\tilde{\mathbf{x}}$, prediction) based on the model equations supposing that the model noise is zero.

Then the measurements and noise models are used to generate the sub-optimal estimate $\hat{\mathbf{x}}$. $\tilde{\mathbf{P}}$ and $\hat{\mathbf{P}}$ are, respectively, the prediction and the estimation error covariance matrices.

In the following sections a brief review of the basic applications of the EKF to the estimation of the state variables of electrical drives is reported.

$$\begin{aligned} \tilde{\mathbf{x}} &= \mathbf{f}(\hat{\mathbf{x}}, \mathbf{u}) & \tilde{\mathbf{y}} &= \mathbf{h}(\tilde{\mathbf{x}}, \mathbf{u}) \\ \text{I) } \mathbf{F} &= \left. \frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\tilde{\mathbf{x}}} & \mathbf{H} &= \left. \frac{\partial \mathbf{h}(\mathbf{x})}{\partial \mathbf{x}} \right|_{\mathbf{x}=\tilde{\mathbf{x}}} \\ \text{II) } \tilde{\mathbf{P}} &= \mathbf{F}\hat{\mathbf{P}}\mathbf{F}^T + \mathbf{Q} \\ \text{III) } \mathbf{K} &= \tilde{\mathbf{P}}\mathbf{H}^T [\mathbf{H}\tilde{\mathbf{P}}\mathbf{H}^T + \mathbf{R}]^{-1} \\ \text{IV) } \hat{\mathbf{x}} &= \tilde{\mathbf{x}} + \mathbf{K}(\mathbf{y} - \tilde{\mathbf{y}}) \\ \text{V) } \hat{\mathbf{P}} &= \tilde{\mathbf{P}} - \mathbf{K}\tilde{\mathbf{P}} \end{aligned}$$

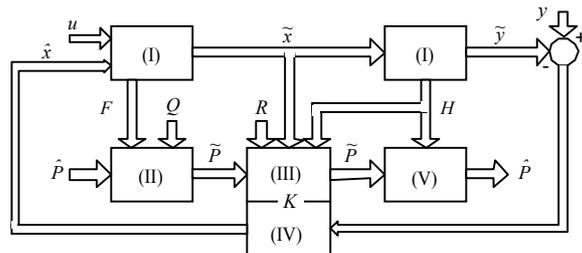


Fig. 1. EKF recursive computational scheme.

Sensorless Field Oriented Control (FOC) of Induction Motors (IM)

Sensorless FOC of IM requires rotor flux and speed information, which are supposed to be provided by the EKF. The filter is usually arranged in the two-phase fixed reference frame and it includes the four electrical (stator and rotor) equations and an additional equation for speed estimation. Stator voltage (v_s) is the filter input and stator current (i_s) is the output (2) (Fig. 2).

$$x = [i_{s\alpha} \ i_{s\beta} \ \varphi_{r\alpha} \ \varphi_{r\beta} \ \omega]^T \quad y = [i_{s\alpha} \ i_{s\beta}]^T \quad u = [v_{s\alpha} \ v_{s\beta}]^T \quad (2)$$

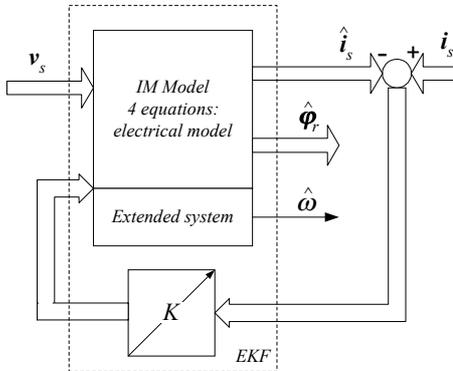


Fig. 2. EKF estimation for Induction Motor.

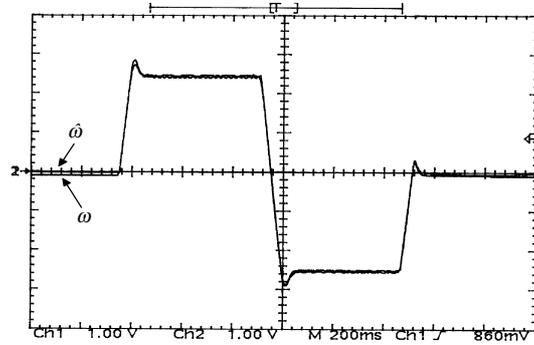


Fig. 3. IM sensorless operation: measured and estimated speed.

An actual implementation of the 5th order filter on the TMS320F240 takes about 150 μ s. The system performance are quite good both during transient and low speed operations, as it is shown in Fig. 3 [1].

Sensorless Field Oriented Control of Permanent Magnet Synchronous Motors (PMSM)

In the case of PMSM, FOC requires the knowledge of the rotor magnet position. In sensorless schemes this information is estimated together with the speed one in order to close the speed control loop. The filter is usually arranged in the two-phase fixed reference frame and it includes the two stator electrical equations and the mechanical ones (3) (Fig. 4).

$$x = [i_{s\alpha} \ i_{s\beta} \ \omega \ \theta]^T \quad y = [i_{s\alpha} \ i_{s\beta}]^T \quad u = [v_{s\alpha} \ v_{s\beta}]^T \quad (3)$$

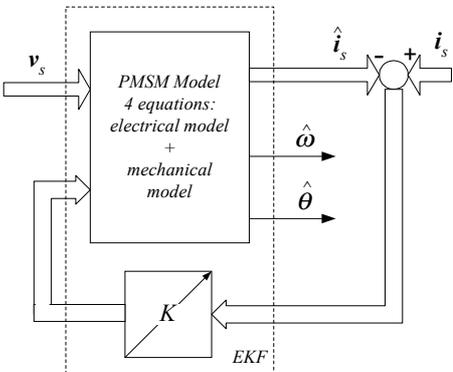


Fig. 4. EKF estimation for PMSM.

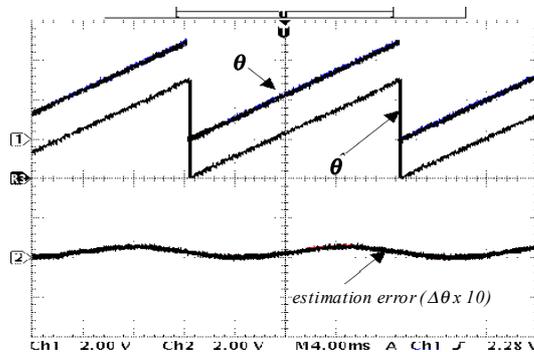


Fig. 5. PMSM sensorless operation: measured and estimated position.

As for the IM, stator voltage is assumed as the system input while stator current as output. An actual implementation of the 4th order filter on the TMS320F240 takes about 120 μ s. The system performance is resumed in Fig. 5 [2].

A case study: speed estimation with low resolution position transducers

Position and rotor speed are required for high performance servo drive systems. Almost all of those systems are provided with optical encoders, but requirements for cost reduction suggest the use of low resolution transducers. This causes a reduction in the system performance, especially at low speed. Several methods have been developed in order to obtain high precision position and speed detection at low speed or when low resolution transducers are employed. In this paper a Kalman filter based solution for this problem is investigated [4]. The main concern is about the best choice of the filter parameters in order to obtain good dynamic and steady-state performance when a fixed point arithmetic is considered for the actual implementation.

Statement of the problem

Speed information is usually provide by the derivative of the encoder position, that is the counted pulses divided by the elapsed time. As a result, one has a quantization error superimposed to the effective speed. If N_p is the number of pulses per revolution, the quantization error $\Delta\omega$ is:

$$\Delta\omega = \frac{60}{N_p T_{sc}} [rpm]. \quad (4)$$

The last equation shows that the quantization error is independent on the operating speed and it depends only on the number of encoder pulses and the speed control period (T_{sc}). In Fig. 6 this situation is resumed in graphical format for some practical cases. In middle-high speed region the quantization error causes a percentage error which is acceptable, but in very low speed region the amplitude of the quantization error is the same as the effective speed and the percentage error becomes intolerable. A way to reduce this error is to increase the speed control period, but this solution reduces the bandwidth of the speed control loop. Furthermore, the increase of the number of encoder pulses means a higher cost of the drive system.

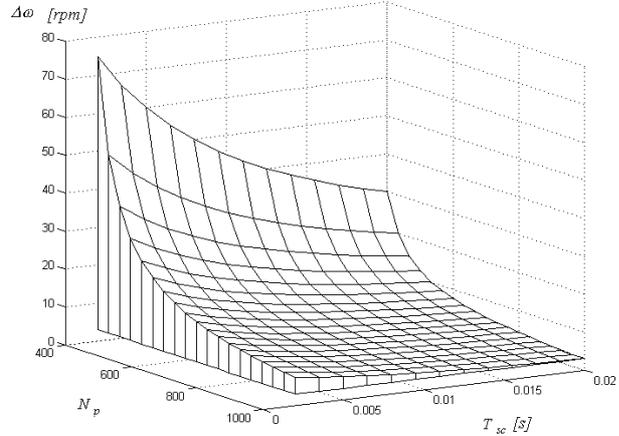


Fig. 6. Speed quantization error.

Solution using EKF

In addition to the application as an observer, which has been resumed in the previous sections, the Kalman filter can be used as a filter itself, provided that a proper state model is arranged for each particular case. The non-linear state model which has been considered in the present paper is based on the orthogonal components (\sin - \cos) representation of a rotating space vector, generated from the encoder position, which is highly affected by the quantization noise (Fig. 7). The filter allows to estimate both the components of the vector position (\sin - \cos) and the frequency (ω) with an accuracy which is independent from the transducers resolution. Moreover one sample of the estimated orthogonal components and the estimated frequency is available at each execution of the EKF.

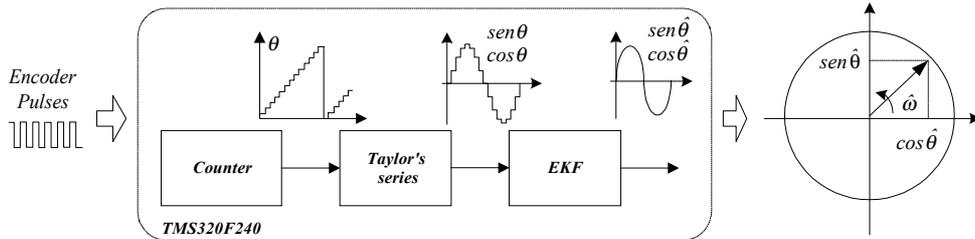


Fig. 7. Configuration for position filtering and speed estimation.

This algorithm is suitable in each application where similar signals are available [5]. The non-linear discrete time state model of the system is shown below, where the scaling factor ω_b has been introduced for the speed.

$$\mathbf{x}_k = [\cos\theta_k \quad \sin\theta_k \quad \omega_k]^T \quad \mathbf{y}_k = [\cos\theta_k \quad \sin\theta_k]^T \quad \mathbf{u}_k = [0 \quad 0]^T \quad (5)$$

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, k) = \begin{pmatrix} x_1 \cdot \cos(\omega_b T_{sc} \cdot x_3) - x_2 \cdot \sin(\omega_b T_{sc} \cdot x_3) \\ x_1 \cdot \sin(\omega_b T_{sc} \cdot x_3) + x_2 \cdot \cos(\omega_b T_{sc} \cdot x_3) \\ x_3 \end{pmatrix} \quad \mathbf{y}_k = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \cdot \mathbf{x}_k \quad (6)$$

$$\mathbf{Q} = q\mathbf{I}_{3 \times 3} \quad \mathbf{R} = r\mathbf{I}_{2 \times 2} \quad (7)$$

Performance analysis

In order to analyse the performance of the considered EKF algorithm, simulation results assuming floating point implementation have been carried out using a Simulink™ model. Then, the same algorithm has been implemented using the TMS320F24x fixed point DSP. The results are presented and compared in order to show the influence of the fixed point implementation with respect to the ‘ideal’ floating point simulation. Particularly, when a fixed point processing architecture is considered, different aspects of the actual implementation have to be addressed. In fact the dynamic range in fixed point arithmetic with a 16-bit word length is $[-1;(1-2^{-15})]$. Therefore, to avoid the overflow and underflow problems, all variables in the filter equations must be scaled to values less than one. The adopted choice of the scaling values influences both the possibility to represent the coefficients of the filter matrices and the performance of the filter itself. In order to achieve a full comparison, these constraints are included in the floating point simulations.

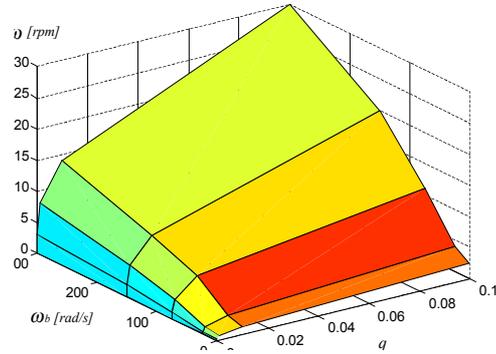


Fig. 8. Steady state estimation error with: $N_p = 480$, $T_{sc} = 150 \mu\text{s}$, $\omega_{ref} = 10 \text{ rpm}$, $r = 1$.

Simulation results

Fig. 8 shows the steady state estimation error with floating point implementation, as a function of the speed scaling factor ω_b and the modelling error covariance matrix parameter q (7). The figure refers to low speed operation ($\omega_{ref} = 10 \text{ rpm}$) where the quantization error is more important. By comparison of the Fig. 8 and Fig. 6, one can notice that the speed error is greatly reduced by the use of EKF

algorithm when certain values of q are considered. Particularly, assuming $q = 0$ the error is null independently on the speed scaling factor (this result is presented in the subsequent Fig. 9). Unfortunately, as far as the transient response is concerned, high values of q should be adopted, which means a large gain K in the filter. Moreover, the choice of $q = 0$ causes the element of the estimation error covariance matrix to overcome the underflow limit, as experimented in the fixed point 16-bit implementation. This conditions is not reached when q is set to the minimum positive value achieved by the 1.15 format, i.e. 2^{-15} . Fig. 9 shows the corresponding behaviour of the estimated speed when two different solutions are adopted for the generation of the $\sin\theta\text{-cos}\theta$ functions, that is a look-up table with 256 elements and a 6th order Taylor’s series. It must be noted the increment of precision obtained by using the Taylor’s series.

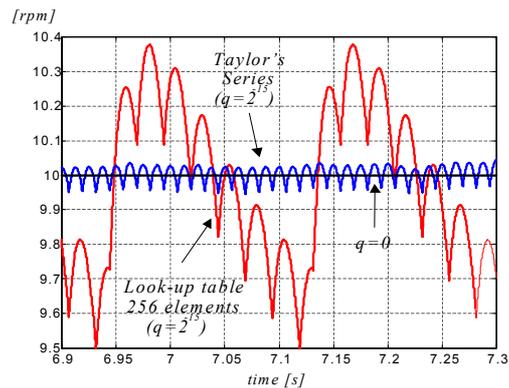


Fig. 9. Steady state estimation with: $\omega_b = 10 \text{ rad/s}$, $N_p = 480$, $T_{sc} = 150 \mu\text{s}$, $\omega_{ref} = 10 \text{ rpm}$, $r = 1$.

Implementation issues

In order to verify the effectiveness of the considered solution in the whole speed range a certain hardware set-up has been adopted which makes uses of a digital function generator supplying the quadrature encoder pulses interface (QEP) of the TMS320F240. This solutions allows to simulate and to accurately define the frequency of the signals from an encoder which in a real system would be mounted on the shaft of the motor.



The algorithm under investigation has been written in assembly language with 16-bit fixed point numerical representation. All the matrix calculations has been implemented by means of dedicated macros. The execution time of the whole algorithm is about $75 \mu\text{s}$ ($72 \mu\text{s}$ for the EKF and $3 \mu\text{s}$ for the Taylor’s series). A host PC has been used to elaborate and display the results of the computation.

Experimental Results

Fig.10 and 11 show the comparison between floating point simulation and TMS320F240 fixed point implementation results in the case of a reference speed of 8 rpm and a 12 pulses/revolution encoder. The parameter q is set to the lower limit of the format (2^{-15}), while the speed scaling value ω_b is set to 1 rad/s in order to reduce the steady-state error (see Fig. 8). The speed control period (T_{sc}) is increased to 16 ms in order to achieve a better correspondence with the low number of pulses/revolution available. It must be noted the good correspondence between the simulation and experimental results. Also, the difference between floating point simulation and fixed point implementation (apart for the discussed limitation on the value of q) is negligible. The responses show slow transient and small steady state error due to the assumed low value of q .

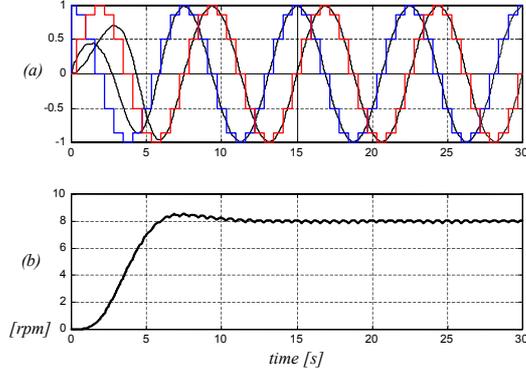


Fig. 10. Floating point simulations with:
 $q = 2^{-15}$, $\omega_b = 1$ rad/s, $\omega_{ref} = 8$ rpm, $N_p = 12$, $T_{sc} = 16$ ms
 (a) actual and estimated sen-cos; (b) estimated speed.

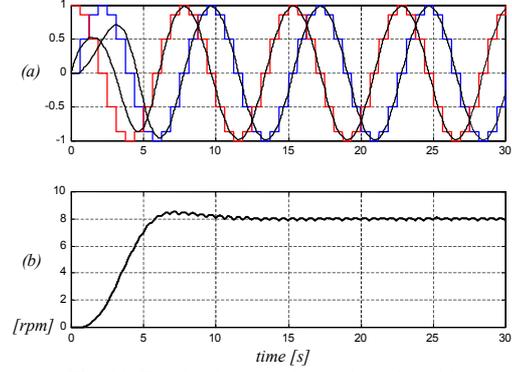


Fig. 11. Fixed point experimental results with:
 $q = 2^{-15}$, $\omega_b = 1$ rad/s, $\omega_{ref} = 8$ rpm, $N_p = 12$, $T_{sc} = 16$ ms
 (a) actual and estimated sen-cos; (b) estimated speed.

Finally, Fig. 12 and Fig. 13 show very low speed operation with different number of pulses for revolution. One can notice the decrease of the steady-state error as the number of pulses for revolution increases.

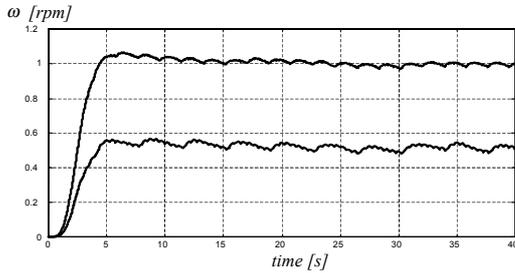


Fig. 12. Fixed point experimental results with:
 $q = 2^{-15}$, $\omega_b = 1$ rad/s, $\omega_{ref} = 0.5/1$ rpm, $N_p = 480$, $T_{sc} = 16$ ms.

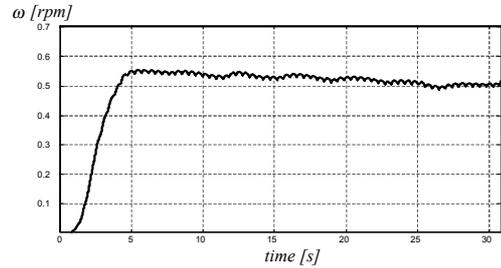


Fig. 13. Fixed point experimental results with:
 $q = 2^{-15}$, $\omega_b = 1$ rad/s, $\omega_{ref} = 0.5$ rpm, $N_p = 2000$, $T_{sc} = 16$ ms.

Conclusions

In this paper a method for speed and position estimation for low resolution position transducers is presented, based on the Extended Kalman Filter. The influence of the scaling factor and the filter parameters in the case of a fixed point implementation with the TMS320F24x is discussed. Particularly, small values of the speed scaling factor reduce the speed error at steady-state but cause the format overflow for increasing speed. A possible solution for this problem, is to adjust this parameter on-line with the change of the speed operating region.

References

- [1] "Sensorless Field Oriented Speed Control of 3-phase AC Induction Motor using TMS320F240 DSP" *Application Note*, SPRA458, Texas Instruments.
- [2] A. Germano, F. Parasiliti, M. Tursini, "Sensorless Speed Control of a PM Synchronous Motor Drive by Kalman Filter," *Proc. of ICEM 94*, vol. 2 pp. 540-544, Paris, 1994
- [3] R. Dhaouadi, N. Mohan, L. Norum, "Design and Implementation of an Extended Kalman Filter for the State Estimation of a Permanent Magnet Synchronous Motor," *IEEE Trans. on Power Electronics*, vol. 6, No. 3, July 1991.
- [4] B.J. Brunsbach, G. Henneberger, Th. Klepsch, "Speed Estimation with Digital Position Sensor," *Proc. of ICEM 92*, pp. 577-581, Manchester, 1992.
- [5] G. Andria, L. Salvatore "Elaborazione di segnali per azionamenti elettrici mediante il filtro di Kalman", *L'Energia Elettrica*, n. 12, pp. 589-600, 1989.