Optimized Method for Speed Estimation Using Incremental Encoder

Alecksey Anuchin, Valentina Astakhova,
Dmitry Shpak, Alexandr Zharkov
Department of Electric Drives
Moscow Power Engineering Institute
Moscow, Russia
anuchin.alecksey@gmail.com

Fernando Briz
Department of Electrical, Electronic, Computers and Systems Engineering
University of Oviedo
Gijón, Spain

***Abstract*—This paper proposes a high-performance speed measurement method for electric drives using incremental encoders. The proposed solution improves the period-based speed estimation method. Such methods suffer from low resolution of time measurement at high speeds. The proposed method solves this problem by processing every speed estimation between the consecutive execution of the speed controller predicting the speed signal in order to achieve maximum performance of the speed loop. The processing algorithm was suggested for the speed estimation. The new method was studied and compared with conventional methods with the modelling experiment. It shows a good step response transient with lower oscillation in comparison to conventional methods for the same tuning of the speed controller.**

Keywords—angular velocity; incremental encoder; speed estimation; variable speed drives; least squares method; prediction.

I. INTRODUCTION

Incremental encoders are widely used for position and speed feedback in electric drives (see Fig. 1). An encoder provides information on the rotor position. This information is enough for the stable operation of the flux observers in the field-oriented control systems or direct torque control. However, many electric drives also implement speed control that requires speed feedback. For high speed spindle drives, that feedback should have high accuracy and a high sampling frequency. The accuracy is required due to the high demands placed on the speed stability, while high sampling frequency is needed to guarantee a rapid reaction to the step loads, for example, when the cutting tool comes in contact with the workpiece.

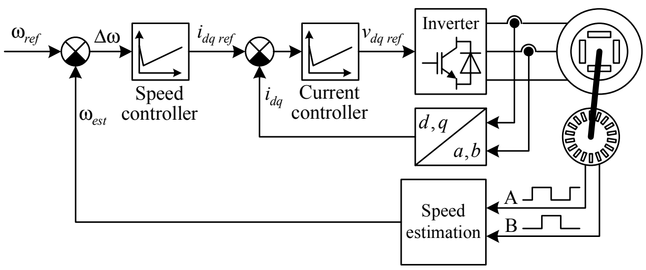

Fig. 1. An electric drive with incremental position encoder.

The research was performed with the support of the Russian Science Foundation grant (Project № 16-19-10618).

Speed estimation from the encoder signals is usually performed within the microcontroller implementing the control. Several methods have been proposed to estimate the rotor speed [1, 2], including frequency-based methods, period-based methods, both with fixed and with variable amounts of rotation, [3, 4] and also, PLL-based methods.

The frequency-based method counts the number of pulses from the incremental encoder over a predefined time. This count is proportional to the actual speed. The problem with the frequency-based method is that its accuracy depends on the desired sampling frequency of the speed estimation. If the sampling frequency is high, the error increases. The method provides the constant absolute error of speed estimation regardless of the motor speed, but its relative error can be unacceptably high on low speeds.

The period-based method measures the elapsed time for an encoder pulse. Its estimation at low speeds is more accurate. With the increase in speed, the frequency of the quadrature signals increases that causes a significant increase in the error. Further improvement in this method is considered in [5] and implemented using FPGA with a high-resolution time-to-digital converter, though this might be inapplicable on microcontroller systems.

A modified period-based method dynamically changes the amount of rotation [3]. It gives accurate results regardless of the current speed of the motor, but it also has a smaller sampling frequency.

The main drawback of the period-based method in comparison to the method with variable amount of rotation is low accuracy, though it provides accurate results on an average. The problem of the method with variable amount of rotation is its asynchronous nature that causes the jitter effect that can degrade the performance of the speed control loop.

This paper proposes a speed estimation method that processes the output data provided by the period-based method between two consecutive executions of the speed controller in order to achieve an accurate result. This processing is to be done with respect to the current speed and to be adapted to the number of samples produced by the speed estimator during the period of the speed control. This not only helps to calculate the average of the signal but also to estimate the derivative of the speed in time in order to predict the speed value in the beginning of the next control cycle to increase the response time of the speed loop.

II. Period-Based Speed Estimation

The period-based method measures the elapsed time for a specified amount of rotation. If the amount of rotation is equal to a single period of the "A" or "B" signal, then the speed in rpm n_{PBM} can be estimated as:

$$n_{PBM} = \frac{60}{\Delta T \cdot K}, \tag{1}$$

where K is the resolution of the encoder in pulses per revolution and ΔT is the elapsed time. As the speed increases the elapsed time becomes smaller, and the accuracy of the measurement reduces due to a higher quantization effect, as shown in the right part of Fig. 2, where there are only few increments of the CPU timer during the speed estimation process.

The relative error (2) depends on the actual speed, resolution of the encoder, and CPU timer frequency (which usually is equal to CPU frequency — f_{CPU}):

$$\delta n_{PBM} = \frac{n_{PBM} \cdot K}{60 f_{CPU}}. \tag{2}$$

The absolute error increases with speed. This error will eventually produce a high frequency noise in the q-axis current reference, as shown in Fig. 3:

$$\Delta n_{PBM} = n_{PBM} \cdot \delta n_{PBM} = \frac{n_{PBM}^2 K}{60 f_{CPU}}. \tag{3}$$

For instance, for a BLAC drive with a 1000 ppr encoder [6] and a maximum speed of 8000 rpm, the duration of a single period of one of the signals is 7.5 us. For a 60 MHz microcontroller, this corresponds to 450 CPU cycles and gives 0.22% error in speed estimation. For a speed of 8000 rpm, this corresponds to an absolute error of 17.7 rpm.

III. Processing the Period-Based Speed Estimations

Large errors of period-based speed estimation at high speed can be avoided by using a moving average filter [7]. If this filter uses the speed estimates obtained over each speed control period, then the maximum delay introduced by the filter will be equal to half of the period of the speed control loop. For instance, if ten sequential estimations are taken and the maximum relative error of each is 0.22% due to the CPU timer quantization effect, then for ten estimations the elapsed time will be ten times greater, thus, the maximum relative error is ten times lesser and below 0.022%.

The frequency of the encoder signals is proportional to the actual speed n_{act} and resolution of the position encoder K:

$$f_{enc} = K \frac{n_{act}}{60}. \tag{4}$$

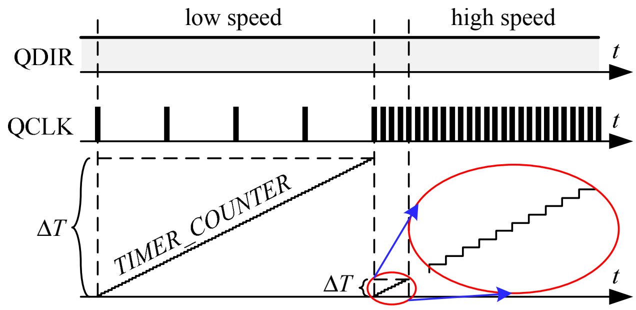

Fig. 2. Period-based method diagram. From top to bottom: QDIR — direction of rotation; QCLK — position counter synchronization signal; and period timer counter.

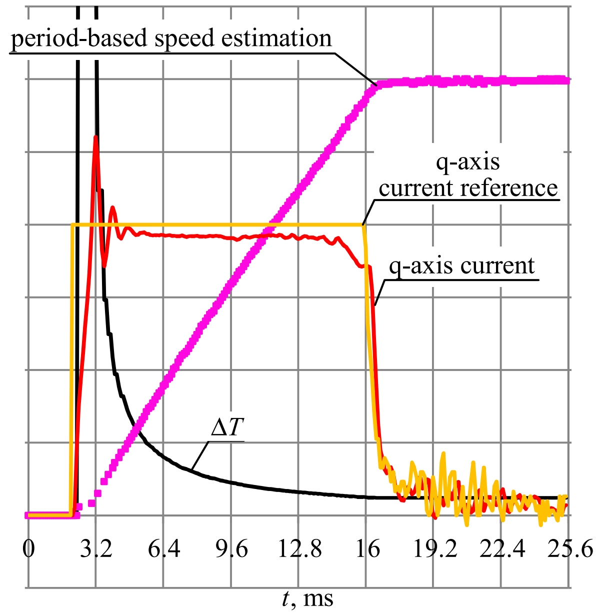

Fig. 3. Start up of the motor with 1000 ppr incremental position encoder and period-based speed estimator on 60 MHz Piccolo MCU from Texas Instruments (ΔT — 5000 CPU cycles per division; q-axis current reference — 3 A per division; speed — 500 rpm per division).

The theoretical value of the CPU cycles for each encoder pulse is given by

$$c_{exact} = \frac{f_{CPU}}{f_{enc}}. \tag{5}$$

As the counter counting CPU cycles can only contain integer values, the fractional part of c_{exact} will be truncated and the elapsed time be estimated as:

$$\Delta T = \mathrm{int}\left(\frac{f_{CPU}}{f_{enc}}\right)\frac{1}{f_{CPU}}, \tag{6}$$

and it should be mentioned that modulo is not discarded but left for the next estimation.

For each period of speed controller sampling, the guaranteed number of speed estimations depends on the encoder frequency and the speed loop sampling frequency f_{SC}:

$$N_{est} = \mathrm{int}\left(\frac{f_{enc}}{f_{SC}}\right). \qquad (7)$$

The speed estimations between speed controller executions may be only integer values, but for any speed it may be N_{est} or $N_{est}+1$, according to the moment of the first estimation due to the asynchronous nature of the period-based method. An example of $N_{est}=3$ during operation at the constant speed is shown in Fig. 4.

If the motor speed is changing, for instance, during acceleration of the drive, then the speed estimations are distributed between the speed controller executions as shown in Fig. 5. In this case, it is possible to not only evaluate the average speed of the drive, but to predict the speed and to estimate its value at that instant of time at which the speed controller applies its output to the input of the current controller. This can help increase passband of the speed loop.

For the last N points of the period-based speed estimation between two consecutive executions of the speed controller, the method of least squares [8] can be implemented. As a result, the equation for the speed as a function of time predicts speed value at that instant of time, when the speed controller applies its reference to the current controller:

$$n_{est} = at + b, \qquad (8)$$

where t is the time from the last speed controller execution,

$$\left.\begin{aligned} a &= \frac{N\sum_{i=1}^{N} t[i]\, n_{PBM}[i] - \sum_{i=1}^{N} t[i] \cdot \sum_{i=1}^{N} n_{PBM}[i]}{N\sum_{i=1}^{N}\left(t[i]\right)^2 - \left(\sum_{i=1}^{N} n_{PBM}[i]\right)^2}, \\ b &= \frac{\sum_{i=1}^{N} n_{PBM}[i] - a\cdot\sum_{i=1}^{N} t[i]}{N}; \end{aligned}\right\} \qquad (9)$$

where $t[i]$ represents the instants of time of speed estimation (from the last speed controller execution) and $n_{PBM}[i]$ represents the speed estimations from the period-based method.

As the speed controller command for the current controller and its command to the inverter and the current response are delayed for at least one half of the PWM cycle (see Fig. 5), the time t from (8) can be set to 1.5 PWM cycles.

Potentially, on a constant speed, the predicted value can suffer due to the quantization effect of the period-based method. For instance, the right part of Fig. 4 contains three estimations: the first value is higher than the rest. The processing of these data by the least squares method gives a positive derivative of speed, while it actually remains constant.

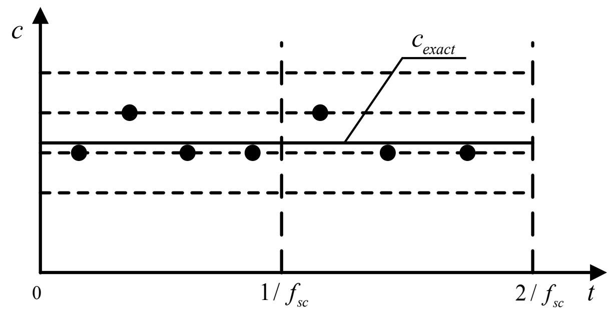

Fig. 4. Estimated elapsed time represented in CPU timer clock cycles for some constant speed (solid line — the exact elapsed time without rounding; dashed lines — integer values; points — estimations of elapsed time).

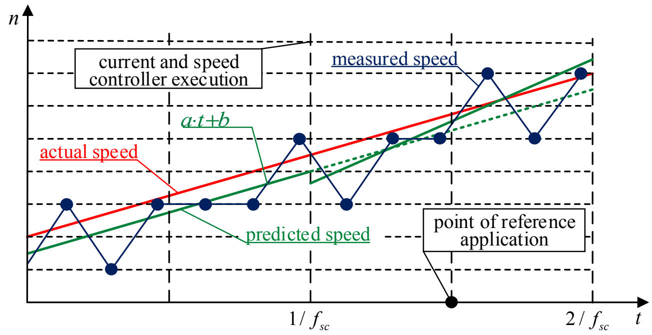

Fig. 5. Illustration for the predictive speed estimation.

In order to reduce this noise, a variable prediction horizon can be implemented. The predictive method should mostly operate during the transient and average — at the settled speed. Equation (8) gives average value with t equal to one half of the PWM cycle. So, depending on the dynamics of the drive, t should vary from 0.5 to 1.5 durations of the PWM cycle. This function can be expressed as:

$$t = 1.0\cdot\left|\frac{T-\hat{T}_{load}}{T_{max}}\right| + 0.5, \qquad (10)$$

where T is the torque produced by the motor, $\hat{T}_{load}$ is the estimation of the load torque, T_{max} is the limit of the speed controller or maximum torque reference.

IV. Modelling Results

The proposed method was implemented in the C++ Builder. The modelled test bench includes a BLAC motor with a 1000 ppr incremental encoder. BLAC motor operates under field oriented control with a PWM frequency of 8 kHz. Speed and currents are sampled in the middle of the PWM cycle, the voltage command being updated in the beginning of each PWM cycle. Fig. 6a shows the final part of the response to a step like speed command of 8150 rpm using the conventional period-based method. The feedback for the speed controller uses the latest speed estimation. Due to the high quantization error of the speed estimator, the unsaturated output of the speed controller exceeds its limits. Thus, the current reference swings between maximum and minimum levels.

The simple average of all speed estimations between consistent executions of the speed controller gives much more accurate results but the speed reaches its reference value with noticeable oscillation (see Fig. 6b). This oscillation is caused by an aggressive tuning of the speed controller for that type of feedback. It can be clearly seen that though the average method gives precise results, the signal suffers from the significant delay of one half of the PWM cycle in measurement and one half in the control.

The proposed predictive method (see Fig. 6c) provides predicted value of the speed for the feedback, and the motor reaches the speed reference rapidly. But in comparison to the average method, the system with predicted feedback suffers from the noise in prediction at the settled speed.

The operation of the drive with the variable prediction horizon for speed estimation is shown in Fig. 6d. The speed reaches its referenced value without oscillations and the transient is very short. This method provides maximum performance of the speed control loop. The motor current is stable at the settled speed that results in smaller current deviations, smaller ohmic losses, and higher efficiency respectively.

V. Conclusions

The algorithm which uses the least square method for processing the estimations of the period-based method was considered in this paper. The least square method helps obtain an average of the speed estimations during the speed control cycle. It can also predict the behaviour of the speed that allows the achievement of better performance of the speed loop.

The simulation results shows the benefits of the proposed method that expels the delay in the speed estimation and provides accurate average speed estimation.

The proposed speed estimation method provides high accuracy together with the prediction of the speed behaviour in dynamics. The method is applicable to any drives with incremental encoders with high dynamics and high encoder resolution. For the drives with low dynamics and low-resolution encoders, this method provides accurate speed estimation, but the prediction should be disabled in order to avoid noises in the feedback.

The proposed method can be implemented on modern microcontrollers. However, in applications requiring high resolution encoders and high sampling frequencies, the use of FPGAs might be required.

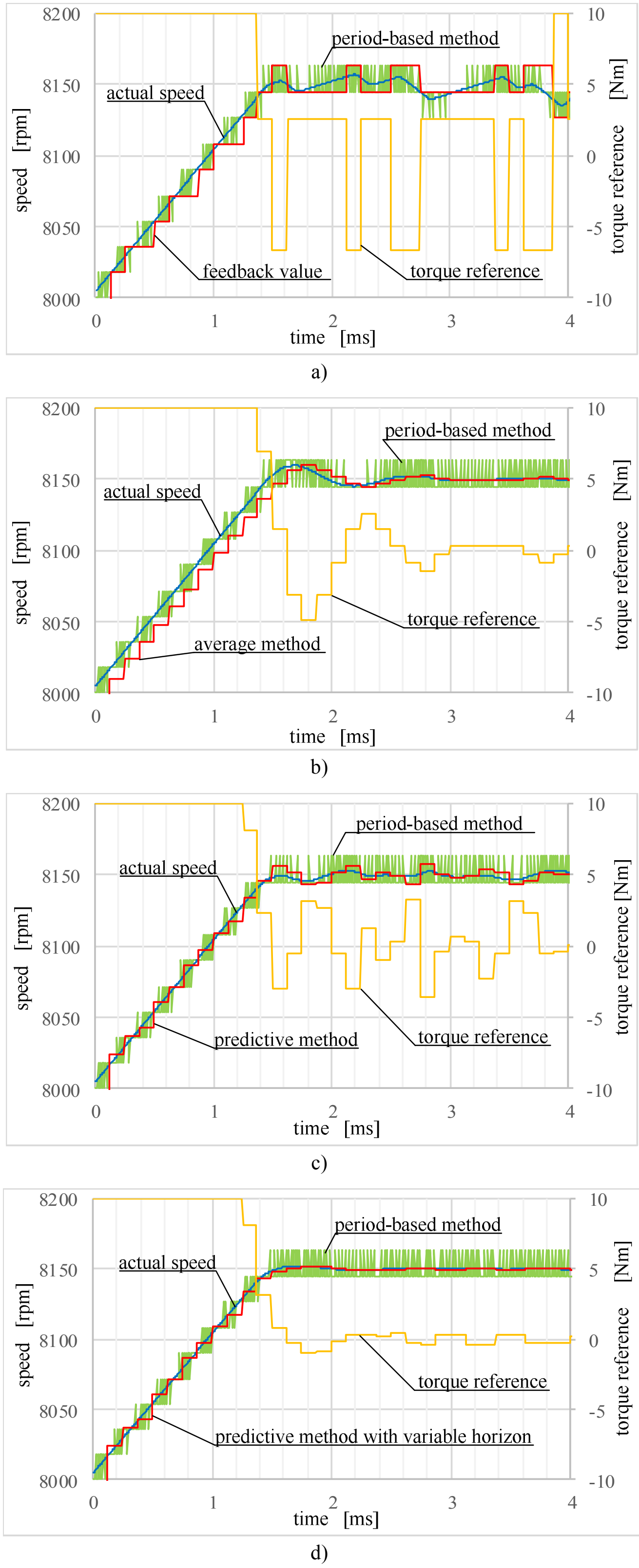

Fig. 6. Transients in the speed loop with various speed estimation methods: a — conventional period-based method; b — period-based method with average filter; c — predictive period-based method; d — predictive period-based method with variable prediction horizon.

REFERENCES

[1] R. Petrella, M. Tursini; L. Peretti; M. Zigliotto, "Speed measurement algorithms for low-resolution incremental encoder equipped drives: A comparative analysis," in Proc. Int. Conf. Electrical Machines and Power Electronics (ACEMP), 2007, pp. 780-787, doi: 10.1109/ACEMP.2007.4510607.

[2] C. A. Negrea, I. I. Incze, M. Imecs, A. V. Pop; C. Szabo, "An improved speed identification method using incremental encoder in electric drives," in Proc. of IEEE Int. Conf. on Automation Quality and Testing Robotics (AQTR), 2012, pp. 536-540, doi: 10.1109/AQTR.2012.6237769.

[3] A. Anuchin, A. Dianov, D. Shpak, V. Astakhova, K. Fedorova, "Speed estimation algorithm with specified bandwidth for incremental position encoder," in Proc. 17th Int. Conf. on Mechatronics - Mechatronika (ME), 2016.

[4] N. Johnson; K J. Mohan; K E. Janson; J. Jose, "Optimization of incremental optical encoder pulse processing," in Proc. of Int. Multi-Conf. on Automation, Computing, Communication, Control and Compressed Sensing (iMac4s), 2013, pp. 769-773, doi: 10.1109/iMac4s.2013.6526510.

[5] J. N. Lygouras, Th. P. Pachidis, K. N. Tarchanidis; V. S. Kodogiannis, "Adaptive high-performance velocity evaluation based on a high-resolution time-to-digital converter," *Transactions on Instrumentation and Measurement*, vol. 57, no. 9, 2008, pp. 2035-2043, doi: 10.1109/TIM.2008.919039.

[6] HEDS-9000/9100, Two Channel Optical Incremental Encoder Modules, Data Sheet, Avago Technologies.

[7] A. F. Ilmiawan, D. Wijanarko, A. H. Arofat; H. Hindersyah; A. Purwadi, "An easy speed measurement for incremental rotary encoder using multi stage moving average method," in Proc. of Int. Conf. on Electrical Engineering and Computer Science (ICEECS), 2014, pp. 363-368, doi: 10.1109/ICEECS.2014.7045279.

[8] http://www.efunda.com/math/leastsquares/lstsqr1dcurve.cfm.