Chapter 4
Rotor Position Error Calculator

Introduction

The rotor position error calculator, labelled θ Error Calculator in the controller structure diagrams of Figures 1.1 and 1.2, calculates the error between the controller's applied rotor position θ' and the actual rotor position using the controller's applied motor currents and the measured motor currents.

The ability of the calculator to find the rotor position error depends on the measured currents being influenced by the motor's back emf to a sufficient level to overcome motor current measurement errors and the controller's applied currents estimation errors. This puts a lower speed limit on the ability of the calculator to estimate the position error.

An important property built into the calculator is to arrange for the error output to gradually drop to zero when the speed drops to values too low for the motor to generate sufficient back emf to obtain usable position error measurements. This allows the position error feedback correction controller to stay active at zero speed, avoiding the need to switch it in and out depending on the speed. This is very important in order to allow the speed to be seamlessly set to any value and to allow maximum controlled acceleration from zero speed. It is particularly important if a position controller is to be added around the speed controller.

As described in Chapter 1, Overview, when the calculator cannot produce a position error value due to the speed being too low, a positive d-axis current is injected into the motor to magnetically lock the rotor into position. This keeps control of the rotor position at speeds down to zero. The positive d-axis current remains on up to a speed called the switch speed, above which this current is gradually reduced and full motor load torque control is available. The setting of the switch speed is described in this chapter.

The output of the error calculator is not in units of degrees or radians. Instead, it is in units of equivalent high-speed q-axis current error, i.e., the expected error in the q-axis current for a given rotor position error due to motor inductance and rotor flux only.

Detecting Angle Error from Currents

Consider the application of a load torque causing the rotor magnetic axis to shift from the applied controller d axis. This change can be detected from the change in the d and q axis currents. The change in angle, δ, can be calculated from the change in currents.

To simplify the calculations, assume initially that the voltages applied to the motor match the back emf and the currents are zero. Then the Feed Forward Converter will set the applied motor voltages to:

$$\begin{bmatrix} v'_\alpha \\ v'_\beta \end{bmatrix} = p \begin{bmatrix} \cos\theta' & -\sin\theta' \\ \sin\theta' & \cos\theta' \end{bmatrix} \begin{bmatrix} \tilde{\lambda}_r \\ 0 \end{bmatrix} = \begin{bmatrix} \cos\theta' & -\sin\theta' \\ \sin\theta' & \cos\theta' \end{bmatrix} \begin{bmatrix} 0 \\ \omega\tilde{\lambda}_r \end{bmatrix} \quad (4.1)$$

where p is the differential operator d/dt and $\tilde{\lambda}_r$ is the estimate of the actual rotor flux linkage λ_r.

It is assumed we are using a perfect pulse width modulator, so the motor voltages are:

$$\begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix} = \begin{bmatrix} \cos\theta' & -\sin\theta' \\ \sin\theta' & \cos\theta' \end{bmatrix} \begin{bmatrix} 0 \\ \omega\tilde{\lambda}_r \end{bmatrix} \quad (4.2)$$

Now, with the load torque applied, assume the resulting back emf phase lag is δ, and the resulting d and q axis motor currents change from 0 to i_d and i_q. The equation for the voltages is now:

$$\begin{bmatrix} v_\alpha \\ v_\beta \end{bmatrix} = \begin{bmatrix} \cos\theta' & -\sin\theta' \\ \sin\theta' & \cos\theta' \end{bmatrix} \begin{bmatrix} R\,i_d \\ R\,i_q \end{bmatrix} + p \begin{bmatrix} \cos\theta' & -\sin\theta' \\ \sin\theta' & \cos\theta' \end{bmatrix} \begin{bmatrix} L\,i_d \\ L\,i_q \end{bmatrix} + p \begin{bmatrix} \cos\theta' & -\sin\theta' \\ \sin\theta' & \cos\theta' \end{bmatrix} \begin{bmatrix} \lambda_r\cos\delta \\ \lambda_r\sin\delta \end{bmatrix} \quad (4.3)$$

Substituting v_α and v_β with Equation 4.2, then expanding and removing the matrix rotation factors:

$$\begin{bmatrix} 0 \\ \omega\tilde{\lambda}_r \end{bmatrix} = \begin{bmatrix} R\,i_d \\ R\,i_q \end{bmatrix} + \begin{bmatrix} L\,p\,i_d - \omega L\,i_q \\ L\,p\,i_q + \omega L\,i_d \end{bmatrix} + \begin{bmatrix} \omega\lambda_r\sin\delta \\ \omega\lambda_r\cos\delta \end{bmatrix} \quad (4.4)$$

For the motor controller, the value we are interested in is sin δ, the approximate angle error when measured in radians, which, from the first line of Equation 4.4, is given by:

$$\sin\delta = \frac{L\,i_q}{\lambda_r} - \frac{R}{\lambda_r\,\omega} i_d - \frac{L}{\lambda_r\,\omega} p\,i_d \quad (4.5)$$

The fast transient term can be removed for the θ Error Calculator, giving:

$$\sin\delta = \frac{L\,i_q}{\lambda_r} - \frac{R}{\lambda_r\,\omega} i_d \quad (4.6)$$

In reality, the initial currents are not zero, and we are looking for changes in current away from their applied values at the input of the Feed Forward Converter, resulting in Equation 4.6 changing to:

$$\sin\delta = \frac{L}{\lambda_r}\left(i_q - i'_q\right) - \frac{R}{\lambda_r\,\omega}\left(i_d - i'_d\right) \quad (4.7)$$

For actual implementation, the output of the θ Error Calculator is in units of equivalent high-speed q-axis current error or λ_r sin δ / L, which is designated i_δ, instead of sin δ. Also, estimated values of L and R and the controller filtered speed ω_f' (see Chapter 1) for ω are used. The actual formula used is:

$$i_\delta = (i_q - i_q') - \frac{\tilde{R}}{\tilde{L}\,\omega_f'}(i_d - i_d') \tag{4.8}$$

Another limitation is that ω cannot be zero. To overcome this, ω'_f is limited in amplitude in software to a value ω_s, called the switch speed, indicating the method cannot be used when the back emf drops to the noise level.

On test in a simulator, it was further found that the term R/$L\omega_s$ (i_d-i'_d) must be multiplied by a ramp function which ramps up from 0 to 1 as speed rises from 0 to ω_s. Otherwise, instability at low speeds could occur.

The resulting formula for i_δ for use in the controller is as follows:

$$\begin{aligned}
i_\delta &= (i_q - i_q') - \frac{\tilde{R}}{\tilde{L}\,\omega_f'}(i_d - i_d') && |\omega_f'| \ge \omega_s \\
i_\delta &= (i_q - i_q') - f_\delta(|\omega_f'|)\frac{\tilde{R}}{\tilde{L}\,\omega_s}(i_d - i_d') && 0 \le \omega_f' < \omega_s \\
i_\delta &= (i_q - i_q') - f_\delta(|\omega_f'|)\frac{\tilde{R}}{\tilde{L}\,(-\omega_s)}(i_d - i_d') && -\omega_s < \omega_f' < 0
\end{aligned} \tag{4.9}$$

where $f_\delta(|\omega_f'|)$ is the ramp function defined as:

$$\begin{aligned}
f_\delta(|\omega_f'|) &= \frac{|\omega_f'|}{\omega_s} && |\omega_f'| < \omega_s \\
f_\delta(|\omega_f'|) &= 1 && |\omega_f'| \ge \omega_s
\end{aligned} \tag{4.10}$$

Setting Switch Speed ω_s

The setting of ω_s depends on two factors: the accuracy of the resistance measurement and the accuracy of the inverter output voltage.

Resistance changes as the motor heats up. Resistance will rise by 40% for a 100°C rise in winding temperature. It is unlikely the winding temperature will exceed this, as the magnets will be in danger of demagnetising.

Assuming Li_q of Equation 4.6 is negligible at the low speeds, the equation of the angle error, Equation 4.7, reduces to:

$$\sin\delta = -\frac{R}{\omega}\frac{(i_d - i_d')}{\lambda_r} \tag{4.11}$$

From zero up to switch speed ω_s, it is expected that the d axis reference current i_d^* will be raised to a holding current level I_h to hold the rotor in sync at zero speed. Because of this, the main cause of an error δ_{err} in the angle δ is an error in (i_d-i_d') due to the error in the resistance estimation. For a given resistance estimation error and I_h holding current value for i_d^*, the resulting value of current error (i_d-i_d') is:

$$(i_d - i_d') = I_h \frac{(R - \tilde{R})}{R} \tag{4.12}$$

for motor resistance R and estimated motor resistance $\tilde{R}$.

The value of the switch speed ω_s for a maximum angle error δ_{err} can be calculated as:

$$\omega_s = \frac{\tilde{R}}{\sin \delta_{err}} \frac{I_h}{\tilde{\lambda}_r} R_{err} \tag{4.13}$$

where:

$$R_{err} = \frac{(R - \tilde{R})}{R} \tag{4.14}$$

The main cause of a resistance error is the change in resistance due to a motor winding temperature rise. Winding temperature can be expected to rise by up to 100°C, which would give a motor resistance rise of 40%, resulting in an R_{err} value of 0.3. A suitable maximum angle error δ_{err} is 0.26 radians (15°).

Switch Speed and Inverter Voltage Error

For many applications, as the speed is reduced, the angle error caused by the inverter voltage error, chiefly cause by the switching dead-time, is greater than the error caused by a resistance value change. This is particularly the case for drone motor controllers which operate at very high PWM frequencies (typically 40kHz) and high currents. Also, the indeterminate voltage at low current and low speed combined with the high amplification of the i_d^* ripple current shown in Equation 4.11 can cause unstable feedback oscillations. The inverter voltage error relative to the back emf depends on the switching dead-time, the DC inverter voltage, the motor Hz/Volt back emf constant, and the PWM frequency. In this controller design, the switch speed is calculated as a constant times the multiplication of all these factors, with the constant determined empirically.

Switch speed can be reduced and thus low speed performance improved by reducing or removing the error caused by the dead-time. The best way to achieve this is by compensating for the dead-time with a carefully designed voltage feed-back system. Traditional current feed-back dead-time compensation schemes will not work for this controller.

Automatic Error Correction

At zero speed, when the function $f_\delta(|\omega_f'|)$ in Equation 4.9 is zero, the current error $(i_d\text{-}i_d')$ is calculated but not used. This can be stored as $(i_d\text{-}i_d')_0$, then subtracted from $(i_d\text{-}i_d')$ in Equation 4.9 at non-zero speeds to correct for resistance error.

In practice, $(i_d\text{-}i_d')$ can be stored as $(i_d\text{-}i_d')_0$ whenever $|\omega_f'|$ is less then, say, $\omega_s/10$, and Equation 4.9 modified to:

$$
\begin{aligned}
&i_\delta=(i_q-i_q')-\frac{\tilde{R}}{\tilde{L}\,\omega_f'}((i_d-i_d')-(i_d-i_d')_0) && |\omega_f'|\geq\omega_s\\
&i_\delta=(i_q-i_q')-f_\delta(|\omega_f'|)\frac{\tilde{R}}{\tilde{L}\,\omega_s}((i_d-i_d')-(i_d-i_d')_0) && 0.1\,\omega_s\leq\omega_f'<\omega_s\\
&i_\delta=(i_q-i_q') && -0.1\,\omega_s<\omega_f'<0.1\,\omega_s\\
&i_\delta=(i_q-i_q')-f_\delta(|\omega_f'|)\frac{\tilde{R}}{\tilde{L}(-\omega_s)}((i_d-i_d')-(i_d-i_d')_0) && -\omega_s<\omega_f'\leq-0.1\,\omega_s
\end{aligned}
\tag{4.15}
$$

With this automatic correction of the resistance error, only expected changes in resistance during times when the speed is not zero need to be considered when determining the value of ω_s. Errors in the estimate of resistance $\tilde{R}$ at zero speed cancel out.

Flux Control

The second line of Equation 4.4 is:

$$\omega\lambda_r^*=R\,i_q+L\,p\,i_q+\omega L\,i_d+\omega\cos\delta\,\lambda_r \tag{4.16}$$

Ignoring the Lpi_q transient term and approximating cos δ to 1, the flux error can be found as:

$$\lambda_r^*-\lambda_r=\frac{R}{\omega}\,i_q+L\,i_d \tag{4.17}$$

To take account of the applied values of current included in the controller, is is modified to:

$$\lambda_r^*-\lambda_r=\frac{R}{\omega}\,(i_q-i_q')+L(i_d-i_d') \tag{4.18}$$

This could be used to automatically adjust the flux setting, eliminating the need to set the flux manually.

In the present design, this is not implemented. The flux never changes for a given motor, so it is felt that better performance can be obtained by manually setting the flux to its correct value.

The automatic adjustment of flux using Equation 4.18 could be useful for an auto-tuning procedure.

Stall Detection

An alternative use for the flux error as calculated in Equation 4.18 is the detection of a motor stall condition.

When the motor is stalled the motor does not produce any rotor flux λ_r. This can be detected using Equation 4.18. In practice, to operate down to low speeds with resistor estimate errors, the following modifications can be incorporated:

$$\begin{aligned}
\lambda_r^* - \lambda_r &= \frac{\widetilde{R}}{\omega_f'}\left(i_q - i_q'\right) + \widetilde{L}\left(i_d - i_d'\right) & \left|\omega_f'\right| \geq \omega_t \\
\lambda_r^* - \lambda_r &= \frac{\widetilde{R}}{\omega_t}\left(i_q - i_q'\right) + \widetilde{L}\left(i_d - i_d'\right) & 0 \leq \omega_f' < \omega_t \\
\lambda_r^* - \lambda_r &= \frac{\widetilde{R}}{\left(-\omega_t\right)}\left(i_q - i_q'\right) + \widetilde{L}\left(i_d - i_d'\right) & -\omega_t < \omega_f' < 0
\end{aligned} \tag{4.19}$$

where the minimum frequency of the divisor ω_t is selected to obtain acceptable errors due to errors in the estimated resistance and in i_q.

The minimum frequency ω_t should be selected so that the error in the measured voltage drop Ri_q is not significant compared with the back emf $\omega\lambda_r$ at this operating frequency.

A way of doing this is to continually adjust ω_t so that the estimated back emf at ω_t operating frequency is a proportion of the estimated Ri_q voltage drop:

$$\omega_t \lambda_r^* = K_n \widetilde{R}\, i_q' \tag{4.20}$$

Frequency ω_t is calculated as:

$$\omega_t = K_n \frac{\widetilde{R}}{\lambda_r^*} i_q' \tag{4.21}$$

A conservative selection of the proportional constant K_n is 1.

The minimum value of ω_t should be limited to, say, ω_s.

In the implementation of stall detection used for the Hunter Motor Controller, the continuous adjustment of is not used. Instead, ω_t is just set to ω_s. This is adequate for most applications.

In the implementation, the drive is tripped off when the flux error λ_r^* - λ_r rises to a proportion of the estimated flux λ_r^*.

Flux Measurement

Another use for the flux error calculation of Equation 4.19 is the measurement of actual flux for use in tuning.

The estimated flux can be set to an approximate value, then the flux error can be found using Equation 4.19 and used to correct the estimated flux. This could be done before stall detection using the measured flux error.

Position Sensor Integration

The controller runs without position sensing but for some applications a rotor position sensor may be required. High accuracy positioning of the rotor is one example. Another is when an accurate fixed torque needs to be applied at standstill.

An obvious and viable way of using a rotor position sensor is to replace the θ Error Calculator in the controller structure diagrams of Figures 1.1 and 1.2 with the difference between the actual rotor flux angle measured with the position sensor and the applied rotor flux angle θ'. With this arrangement, a positive *d* axis current at zero speed to hold the rotor in position is no longer required and the DC gain of the θ Error Calculator no longer needs to be reduced at zero speed. The disadvantage is that the actual rotor position is indeterminate between the sensor measurement resolution steps, causing chattering.

An alternative way of making use of the rotor position sensor measurement is to keep the standard sensorless scheme in operation and to use the position measurement together with the torque versus position error set by the *d*-axis holding current at zero speed to measure the load torque, then adding the equivalent torque current to the *q*-axis current to bring the position error to zero. A digital z-domain controller will probably needed to be included for stability. The advantages of this method are that chattering would be eliminated and position control would automatically revert to full sensorless mode in the event of a position sensor failure. The disadvantage is that an extra motor loss will be incurred from the positive *d*-axis current at zero speed.