AUTOMATIC CALIBRATION OF SINUSOIDAL ENCODER SIGNALS

Silvano Balemi

University of Applied Sciences of Southern Switzerland
Department of innovative technologies (SUPSI-DTI)
CH-6928 Lugano-Manno

Abstract: Precision positioning and manipulation systems often rely on data from sensors delivering a sinusoidal signal pair. These signals can be interpolated to provide a high resolution measurement.

The paper presents a novel method for removing signal differences and drifts by automatic self-calibrating the sinusoidal signal pair. Then, manual calibration is avoided and slow varying effects, like thermal drifts or imprecise sensor mounting, can be effectively compensated.

The proposed method can be implemented on a DSP. It uses an on-line gradient search to minimize a given performance index.   *Copyright © 2005 IFAC*

Keywords: Precision Control, Sinusoidal encoders, Measurement interpolation, Measurement calibration.

1. INTRODUCTION

Precision positioning and manipulation systems often rely on data from sensors delivering a sinusoidal signal pair. From the signal pair, a high resolution measurement can be obtained with resolutions of more than 1000 times finer than the length of the signal period.

Unfortunately, the signal pair is often affected by different gains and unknown offsets which may depend on the individual sensor or even on the mounting. Additionally, these quantities may change during time because of thermal drifts or of dirt affecting the sensor.

The compensation of these effects can be performed either in software or implemented in digital interpolation circuits which deliver a position measurement based on the signal pair. In both cases, gains and offsets are often corrected on-line by measuring and evaluating the extremes of the two signals. This is not satisfactory because the signals may be affected by additional non-linearities or by noise, which influence the computation of the correction.

A different approach (Venema and Hannaford, 1995) makes use of the dynamics of the system to be controlled. It estimates the position by exploiting the known model of the system and by considering the effect of gains and offsets as noise to be filtered out. However, this may deliver unsatisfactory results during slow motions, in particular if the signal pair is affected by unmodeled effects. Moreover, the estimation must be tailored for the system to be controlled, and can not be used in a standalone, generic sensor.

The method presented in this paper relies on an on-line computation of the correction parameters, i.e. it calibrates the sensor during operation. Then, both effects due to the sensor mounting or to the characteristics of the individual sensor and slow varying changes can be compensated.

The paper first presents three different variations of a discrete-time, digital interpolation algorithm which is inspired by an equivalent closed-loop analog circuit. Then it recalls the Heydemann correction method, an off-line parameter compensation based on a least-square minimization. It then proceeds with a variation of the Heydemann correction, which is suitable for on-line use and finally presents a new search method based on the minimization of the residual compensation error.

2. SIGNAL INTERPOLATION

A sinusoidal interface encoder provides two signals: $\sin(\alpha)$ and $\cos(\alpha)$. For a displacement sensor (see Figure 1) the angle α is given by

$$\alpha = k_{\text{sincos}} \cdot x = \frac{2 \cdot \pi}{l_{\text{period}}} \cdot x \qquad (1)$$

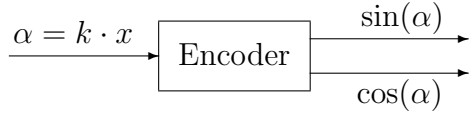where x indicates the displacement and l_{period} the signal period length when increasing x.

Fig. 1. Sinusoidal encoder signals in function of the displacement x

Most digital interpolators first square the sinusoidal signals and deliver the position with a resolution of $l_{\text{period}}/4$ like in quadrature encoders. The estimate within the quarter of period is given by

$$\hat{\alpha} = \arctan\left(\frac{\sin(\alpha)}{\cos(\alpha)}\right)$$

either by computing the trigonometric function or by using a table with the two signal pair values as inputs. Common to all these implementations is the difficulty in obtaining precise estimates in presence of signal non-linearities and/or of noise.

3. THE CLOSED-LOOP INTERPOLATION

Before the advent of digital interpolators, interpolation was performed using an analog control loop circuit having the characteristic of a low-pass filter (see Figure 2). Then, signal non-linearities as well as measurement noise can be filtered out.

The signal e at the input of the filter is given by

$$e = \sin(\alpha) \cdot \cos(\hat{\alpha}) - \cos(\alpha) \cdot \sin(\hat{\alpha}) = \sin(\alpha - \hat{\alpha})$$

and for small values of $\alpha - \hat{\alpha}$ it is possible to write:

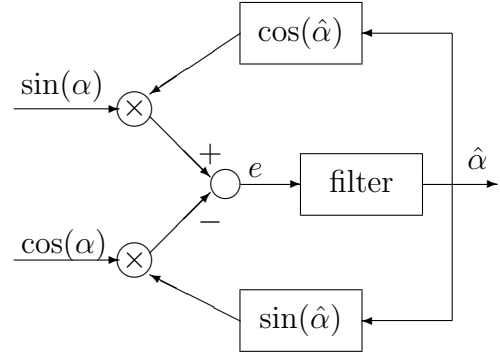$$e = \sin(\alpha - \hat{\alpha}) \approx \alpha - \hat{\alpha}$$

Fig. 2. Closed-loop estimation of $\alpha = k_{\text{sincos}} \cdot x$

showing that the signal e approximates the error between the value of α and its estimate $\hat{\alpha}$. As shown in Figure 3, $\hat{\alpha}$ converges to the value of α for stable closed-loops.
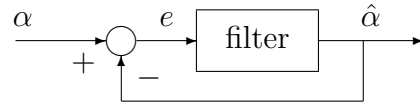

Fig. 3. Simplified interpolation closed-loop system

From $\hat{\alpha}$ it is then possible to obtain the position estimate with $\hat{x} = \frac{\hat{\alpha}}{k_{\text{sincos}}}$.

4. THE DIGITAL FILTER

A digital implentation of the closed-loop circuit above is already object of a DSP application note of AnalogDevices (2000). In this section three possible choices of the discrete-time filter are presented. Our objective is to estimate both the position and the speed of the motion.

4.1 Method 1: Second order filter and position derivative

For the filter shown in Figure 2 the following second order discrete-time realization is proposed.

$$G_{\text{filter}}(z) = \frac{N_{\text{filter}}(z)}{D_{\text{filter}}(z)} = \frac{z \cdot a + b}{z^2 + z \cdot c + d} \qquad (2)$$

Two discrete-time closed-loop poles

$$p_1 = e^{p_{s1} \cdot T_s} = e^{-\omega_0 \cdot t_s \cdot \left(\xi - \sqrt{\xi^2 - 1}\right)}$$
$$p_2 = e^{p_{s2} \cdot T_s} = e^{-\omega_0 \cdot t_s \cdot \left(\xi + \sqrt{\xi^2 - 1}\right)}$$

are chosen with appropriate frequency ω_0 and damping coefficient ξ. Then, the discrete-time desired characteristic polynomial is:

$$D_{\text{cl}}(z) = (z - p_1) \cdot (z - p_2)$$
$$= z^2 - (p_1 + p_2) \cdot z + p_1 \cdot p_2$$

If the filter denominator $z^2 + z \cdot c + d$ is given, it is possible to obtain the desired closed-loop

denominator by choosing an appropriate filter numerator. In fact:

$$G_{\mathrm{cl}}(z) = \frac{N_{\mathrm{cl}}(z)}{D_{\mathrm{cl}}(z)} = \frac{N_{\mathrm{filter}}(z)}{N_{\mathrm{filter}}(z) + D_{\mathrm{filter}}(z)} \quad (3)$$

the comparison of the two denominators $D_{\mathrm{cl}}(z)$ and $N_{\mathrm{filter}}(z) + D_{\mathrm{filter}}(z)$ yields

$$N_{\mathrm{filter}}(z) = D_{\mathrm{cl}}(z) - D_{\mathrm{filter}}(z)$$
$$= z \cdot (-p_1 - p_2 - c) + p_1 \cdot p_2 - d$$

The speed is computed by taking the derivative of the position estimate, thus causing extra on-line computational load (see Figure 4).
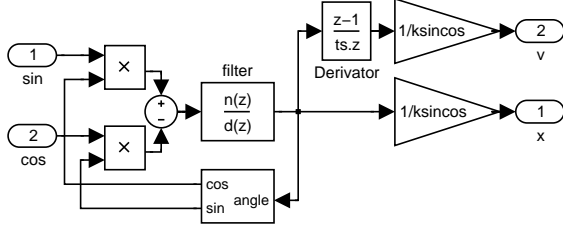

Fig. 4. Closed-loop interpolator with speed estimate from position derivative

Zero steady state error is achieved, if the filter is of type 1, i.e. if at least one of the filter poles is at 1. Choosing both poles at 1 (the filter is then of type 2) also guarantees that a ramp (i.e. caused by a constant motion) can be followed without error.

4.2 Method 2: First order stage and integrator

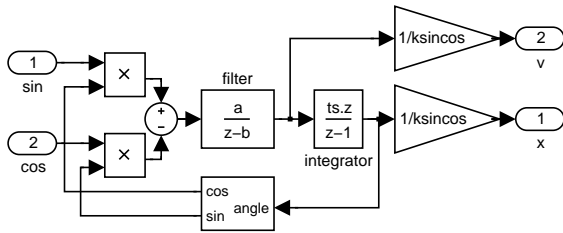The second order filter can be separated into two first order terms as in Figure 5.

Fig. 5. Closed-loop interpolator with speed estimation without derivation

Thanks to the integral term $\frac{t_s \cdot z}{z-1}$, the speed is obtained without having to explicitly taking the derivative of the position estimate.

The two desired closed-loop poles can be chosen as in Method 1. With the filter of the form

$$G_{\mathrm{filter}}(z) = \frac{a}{z-b} \cdot \frac{t_s \cdot z}{z-1}$$

the closed-loop transfer function becomes

$$G_{\mathrm{cl}}(z) = \frac{t_s \cdot z \cdot a}{z^2 + z \cdot (a \cdot t_s - b - 1) + b}$$

Then, comparison of the desired closed-loop denominator with the denominator of the closed-loop transfer function yields the coefficients

$$\begin{cases} a = \dfrac{p_1 \cdot p_2 - p_1 - p_2 + 1}{t_s} \\ b = \qquad p_1 \cdot p_2 \end{cases}$$

Note that the filter pole b is used to place the closed-loop poles. Thus, the second filter pole cannot be freely chosen and it is impossible to implement a filter of type 2.

4.3 Method 3: Second order stage and integrator

Finally, for the filter shown in Figure 2 a second order realization is proposed with the structure shown in Figure 6.
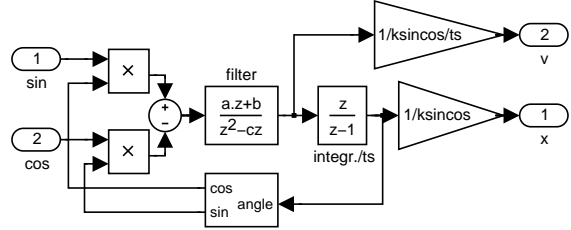

Fig. 6. Closed-loop interpolator with speed estimation without derivation

Apparently, the filter is a third order one. In reality, for the choice of a filter of type 1 or higher, the resulting parameters are the same as in Method 1 with the only difference that an additional pole and a zero at the origin are now present, which cancel out in the filter transfer function.

The advantage of this form is that the most recent estimate of the speed can be extracted while it is also possible to implement a second order type 2 filter.

This form does not cause additional divisions and multiplications. The only price to pay with respect to Method 1 is one additional variable definition and a variable copy per cycle.

5. CALIBRATION

The signals measured in Figure 2 are not ideal but are affected by offsets, unknown gains, phase shift differences between the two signals and by non-linearities.

Considering the effect of gains, offsets and phase shifts only, the signal pair measured can be expressed by

$$\begin{aligned} y_{c,k} &= a_c \cdot \cos \alpha_k + b_c \\ y_{s,k} &= a_s \cdot \sin(\alpha_k - \beta) + b_s \end{aligned} \quad (4)$$

5.1 Heydemann correction

A calibration method has been proposed by Heydemann (1981) (see also (Birch, 1990)). This correction relies on a least-square estimate and on the solution of a set of non-linear equations.

The measurements given by equation (4) can be rearranged as

$$\frac{1}{a_c} \cdot y_{c,k} + \frac{-b_c}{a_c} = \cos(\alpha_k)$$

$$\frac{1}{a_s} \cdot y_{s,k} + \frac{-b_s}{a_s} = \sin \alpha_k \cdot \cos \beta - \cos \alpha_k \cdot \sin \beta$$

After replacement of $\cos(\alpha_k)$ in the second expression and introduction of the notation $g_c = 1/a_c$ and $g_s = 1/a_s$ the equations above can be rewritten as

$$g_c \cdot (y_{c,k} - b_c) = \cos(\alpha_k)$$

$$\frac{g_s \cdot (y_{s,k} - b_s) + (g_c(y_{c,k} - b_c) \cdot \sin \beta)}{\cos \beta} = \sin(\alpha_k)$$

The equality $\sin^2 \alpha + \cos^2 \alpha = 1$ gives

$$\theta_1 \cdot y_c^2 + \theta_2 \cdot y_s^2 + \theta_3 \cdot y_c \cdot y_s + \theta_4 \cdot y_c + \theta_5 \cdot y_s = 1$$

with

$$\theta_1 = \frac{g_c^2}{K \cdot \cos^2 \beta}$$

$$\theta_2 = \frac{g_s^2}{K \cdot \cos^2 \beta}$$

$$\theta_3 = \frac{2 g_c g_s \sin \beta}{K \cdot \cos^2 \beta}$$

$$\theta_4 = -\frac{2 g_c}{K \cdot \cos^2 \beta}(g_c b_c + g_s b_s \sin \beta) \quad (5)$$

$$\theta_5 = -\frac{2 g_s}{K \cdot \cos^2 \beta}(g_s b_s + g_c b_c \sin \beta)$$

$$K = 1 - \frac{g_c^2 b_c^2 + g_s^2 b_s^2 + 2 g_c b_c g_s b_s \sin \beta}{\cos^2 \beta}$$

or equivalently

$$\underbrace{[y_c^2, \; y_s^2, \; y_c \cdot y_s, \; y_c, \; y_s]}_{Y} \cdot \underbrace{[\theta_1, \; \theta_2, \; \theta_3, \; \theta_4, \; \theta_5]'}_{\Theta} = 1$$

For a set of measurement pairs the equation extends to the equation set

$$\mathbf{Y} \cdot \Theta = \mathbf{1}$$

Then the least square estimate

$$\hat{\Theta} = (\mathbf{Y}' \cdot \mathbf{Y})^{-1} \cdot \mathbf{Y}' \cdot \mathbf{1}$$

can be computed. Unfortunately, the parameters cannot be used to directly compensate the measurements y_c and y_s. Given $\hat{\Theta}$ the set of non-linear equations (5) must be solved for the five compensation parameters g_c, b_c, g_s, b_s and β. The necessary computational complexity is high and makes this method unsuitable for on-line use.

5.2 Gradient search: correction of gains and offsets

A first idea would be to use a gradient search giving directly the correction parameters. Considering only the effect of gains and offset, equation (4) reduces to

$$y_{c,k} = a_c \cdot \cos \alpha_k + b_c$$
$$y_{s,k} = a_s \cdot \sin \alpha_k + b_s$$

This expression can be rewritten as

$$\theta_1 \cdot y_{c,k} + \theta_2 = \cos \alpha_k$$
$$\theta_3 \cdot y_{s,k} + \theta_4 = \sin \alpha_k$$

where $\theta_1 = 1/a_c$, $\theta_2 = -b_c/a_c$, $\theta_3 = 1/a_s$ and $\theta_4 = -b_s/a_s$. The trigonometric equality used before can be rewritten as

$$\|M \cdot \Theta\|_2^2 = 1$$

where $\Theta = [\theta_1, \theta_2, \theta_3, \theta_4]'$ and

$$M = \begin{pmatrix} y_{c,k} & 1 & 0 & 0 \\ 0 & 0 & y_{s,k} & 1 \end{pmatrix} \quad (6)$$

The minization of the error $\|M_k \cdot \Theta\|_2^2 - 1$ leads to the objective function

$$J = \sum_k \epsilon_k^2 = \sum_k (\|M_k \cdot \Theta\|_2^2 - 1)^2$$

to be minimized. The corresponding gradient is

$$\nabla J = \sum_k 4 \cdot M_k' \cdot M_k \cdot \Theta \cdot (\Theta' \cdot M_k' \cdot M_k \cdot \Theta - 1)$$

which can be used off-line for estimating Θ and thus the parameters a_c, b_c, a_s and b_s. This is performed by iterating $\Theta \to \Theta - \gamma \cdot \nabla J$ with a suitably small γ. Note that explicit estimate of the offsets and of the gains is not needed. The estimate of the parameter vector Θ is sufficient for the calibration of our measurements.

An on-line version can be implemented in various ways, e.g. by calculating the gradient over a number of steps before updating Θ or also by computing a time-smoothed version of the gradient with a forgetting factor λ as in the following formula

$$\nabla J_k = 4 \cdot M_k' \cdot M_k \cdot \Theta \cdot (\Theta' \cdot M_k' \cdot M_k \cdot \Theta - 1)$$
$$+ \lambda \cdot \nabla J_{k-1}$$

5.3 Gradient search: correction of gains, offsets and phase shifts

The phase difference between the two sinusoidal signals in the measurement is not always exactly 90 degrees. The gradient search with elimination

of the angle α can be extended, like in the Heydemann correction, to include the estimate of the phase deviation β from the ideal 90 degrees. Then the equations for the compensation of the two measurements (4) can be rewritten as

$$\begin{aligned} \theta_1 \cdot y_{c,k} + \theta_2 &= \cos(\alpha_k) \\ \theta_3 \cdot y_{s,k} + \theta_4 + \theta_5 \cdot y_{c,k} &= \sin(\alpha_k) \end{aligned}$$

where $\theta_1 = \frac{1}{a_c}$, $\theta_2 = -\frac{b_c}{a_c}$, $\theta_3 = \frac{1}{a_s \cos\beta}$, $\theta_4 = -\frac{b_s}{a_s \cos\beta} - \frac{b_c \cdot \tan\beta}{a_c}$ and $\theta_5 = \frac{\tan\beta}{a_c}$. Similarly to the previous gradient search, the new matrix

$$M = \begin{pmatrix} y_{c,k} & 1 & 0 & 0 & 0 \\ 0 & 0 & y_{s,k} & 1 & y_{c,k} \end{pmatrix} \qquad (7)$$

can be defined together with a five element vector Θ which delivers the five compensation parameters. Again, no explicit estimate of offsets, gains and phase difference has to be computed.

5.4 Gradient search: correction of non-linearities

The sinusoidal signals may be affected by non-linearities given by the general expression

$$y_{c,k} = f_c(\cos\alpha_k) \qquad y_{s,k} = f_s(\sin\alpha_k)$$

Assuming that the non-linearities are invertible (which is reasonable in practice) the inverse functions

$$f_c^{-1}(y_{c,k}) = \cos\alpha \qquad f_s^{-1}(y_{s,k}) = \sin\alpha$$

can be defined. For the functions f_c^{-1} and f_s^{-1} a third order polynomial approximation can be chosen, which leads to

$$\begin{aligned} \theta_1 \cdot y_{c,k} + \theta_2 \cdot y_{c,k}^2 + \theta_3 \cdot y_{c,k}^3 + \theta_4 &= \cos\alpha_k \\ \theta_5 \cdot y_{s,k} + \theta_6 \cdot y_{s,k}^2 + \theta_7 \cdot y_{s,k}^3 + \theta_8 &= \sin\alpha_k \end{aligned}$$

The steps defined in sections 5.2 and 5.3 can be re-applied here with the only difference that the matrix M is now of the form

$$M = \begin{pmatrix} y_{c,k} & y_{c,k}^2 & y_{c,k}^3 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & y_{s,k} & y_{s,k}^2 & y_{s,k}^3 & 1 \end{pmatrix}$$

and that there are 8 compensation parameters θ_i.

5.5 Gradient based on residual

The calibration methods above require complex computations. Moreover, static measurements (constant x) may bias the estimates. Another method considers the error residuals

$$r_k = \begin{pmatrix} \theta_1 \cdot y_{c,k} + \theta_2 - \cos\hat{\alpha}_k \\ \theta_3 \cdot y_{s,k} + \theta_4 - \sin\hat{\alpha}_k \end{pmatrix} \qquad (8)$$

i.e. the error between the estimated sine and cosine values and the compensated measurements.

Consider the objective function $J = \sum_k \|r_k\|^2$. The first order approximation of J with respect to a parameter variation $\delta\Theta$ is

$$\sum_k \|r_k + \nabla r_k \delta\Theta\|^2 \approx \sum_k \|r_k\|^2 + 2\cdot\sum_k r_k^T \cdot \nabla r_k \cdot \delta\Theta$$

The objective function value can be reduced with the following choice of the parameter update:

$$\delta\Theta = -\gamma \cdot \sum_k (\nabla r_k)^T \cdot r_k$$

where the small scalar value γ controls the speed of convergence of the parameter vector Θ.

The gradient ∇r_k is

$$\nabla r_k = \begin{pmatrix} y_{c,k} + \sin\hat{\alpha}_k \cdot \frac{\partial\hat{\alpha}_k}{\theta_1} & -\cos\hat{\alpha}_k \cdot \frac{\partial\hat{\alpha}_k}{\theta_1} \\ 1 + \sin\hat{\alpha}_k \cdot \frac{\partial\hat{\alpha}_k}{\theta_2} & -\cos\hat{\alpha}_k \cdot \frac{\partial\hat{\alpha}_k}{\theta_2} \\ \sin\hat{\alpha}_k \cdot \frac{\partial\hat{\alpha}_k}{\theta_3} & y_{s,k} - \cos\hat{\alpha}_k \cdot \frac{\partial\hat{\alpha}_k}{\theta_3} \\ \sin\hat{\alpha}_k \cdot \frac{\partial\hat{\alpha}_k}{\theta_4} & 1 - \cos\hat{\alpha}_k \cdot \frac{\partial\hat{\alpha}_k}{\theta_4} \end{pmatrix}^T$$

$$= \underbrace{\begin{pmatrix} y_{c,k} & 1 & 0 & 0 \\ 0 & 0 & y_{s,k} & 1 \end{pmatrix}}_{M_k} + \begin{pmatrix} \sin\hat{\alpha}_k \\ -\cos\hat{\alpha}_k \end{pmatrix} \cdot \nabla\hat{\alpha}_k$$

where M_k is again the matrix of equation (6). For $\nabla\hat{\alpha}_k$ take the recursive equation

$$\hat{\alpha}_{k+1} = -c \cdot \hat{\alpha}_k - d \cdot \hat{\alpha}_{k-1} + a \cdot e_k + b \cdot e_{k-1} \quad (9)$$

from the filter (2). Then

$$\nabla\hat{\alpha}_{k+1} = -c\cdot\nabla\hat{\alpha}_k - d\cdot\nabla\hat{\alpha}_{k-1} + a\cdot\nabla e_k + b\cdot\nabla e_{k-1}$$

Consider now the error

$$e_k = (\theta_3 \cdot y_{s,k} + \theta_4)\cdot\cos\hat{\alpha}_k - (\theta_1 \cdot y_{c,k} + \theta_2)\cdot\sin\hat{\alpha}_k$$

The derivative of e_k with respect to θ_1 is

$$\frac{\partial e_k}{\partial\theta_1} = -y_{c,k} \cdot \sin\hat{\alpha}_k$$

$$-(\theta_3 \cdot y_{s,k} + \theta_4)\cdot\sin\hat{\alpha}_k \cdot \frac{\partial\hat{\alpha}_k}{\partial\theta_1}$$

$$-(\theta_1 \cdot y_{c,k} + \theta_2)\cdot\cos\hat{\alpha}_k \cdot \frac{\partial\hat{\alpha}_k}{\partial\theta_1}$$

and considering that $\theta_3 \cdot y_{s,k} + \theta_4 \approx \sin\alpha_k$ and that $\theta_1 \cdot y_{c,k} + \theta_2 \approx \cos\alpha_k$ the gradient

$$\frac{\partial e_k}{\partial\theta_1} = -y_{c,k} \cdot \sin\hat{\alpha}_k - \cos(\alpha_k - \hat{\alpha}_k) \cdot \frac{\partial\hat{\alpha}_k}{\partial\theta_1}$$

is obtained. Finally, for $\hat{\alpha}_k \approx \alpha_k$

$$\frac{\partial e_k}{\partial\theta_1} = -y_{c,k} \cdot \sin\hat{\alpha}_k - \frac{\partial\hat{\alpha}_k}{\partial\theta_1}$$

Proceeding in the same way for the other variables the gradient ∇e_k can be obtained

$$\nabla e_k = \underbrace{\begin{pmatrix} -y_{c,k} \cdot \sin\hat{\alpha}_k \\ -\sin\hat{\alpha}_k \\ y_{s,k} \cdot \cos\hat{\alpha}_k \\ \cos\hat{\alpha}_k \end{pmatrix}^T}_{w_k} -\nabla\hat{\alpha}_k \qquad (10)$$

Now, with this equation

$$\nabla\hat{\alpha}_{k+1} = -(c+a)\cdot\nabla\hat{\alpha}_k - (d+b)\cdot\nabla\hat{\alpha}_{k-1}$$
$$+ a\cdot w_k + b\cdot w_{k-1} \qquad (11)$$

Then the parameter update can be expressed as

$$\delta\Theta = -\gamma \cdot \sum_k \left(M_k + \begin{pmatrix} \sin\hat{\alpha}_k \\ -\cos\hat{\alpha}_k \end{pmatrix} \cdot \nabla\hat{\alpha}_k \right)^T \cdot r_k$$

where w_k comes from (10) and $\nabla\hat{\alpha}_k$ from (11).

5.5.1. Approximation of the gradient The gradient ∇r_k can be simplified by noting that equation (11) corresponds to the equation of the closed-loop (3) with w as input and $\nabla\hat{\alpha}$ as ouput.

The closed-loop system is stable and well damped (by design). Because the filter is of type 2, the steady-state amplification is 1. Assuming that the dynamics of the measured system are slower than the dynamics of the interpolator closed-loop, $\nabla\hat{\alpha}_k$ can be approximated by w_k. Then the parameter update becomes

$$\delta\Theta = -\gamma \cdot \sum_k \left(M_k + \begin{pmatrix} \sin\hat{\alpha}_k \\ -\cos\hat{\alpha}_k \end{pmatrix} \cdot w_k \right)^T \cdot r_k$$
$$= -\gamma \cdot \sum_k \left(M_k^T \cdot r_k + w_k^T \cdot \begin{pmatrix} \sin\hat{\alpha}_k \\ -\cos\hat{\alpha}_k \end{pmatrix}^T \cdot r_k \right)$$

and with the definition (8) for r_k

$$\delta\Theta = -\gamma \cdot \sum_k \left(M_k^T \cdot r_k - w_k^T \cdot e_k \right)$$

Further noting that

$$w_k = \begin{pmatrix} -\sin\hat{\alpha}_k & \cos\hat{\alpha}_k \end{pmatrix} \cdot M_k$$

the final form

$$\delta\Theta = -\gamma \cdot \sum_k M_k^T \cdot \left(r_k + \begin{pmatrix} \sin\hat{\alpha}_k \\ -\cos\hat{\alpha}_k \end{pmatrix} \cdot e_k \right)$$

is obtained. Because the value of e_k is already needed for the closed-loop interpolation, the calculation of $\delta\Theta$ only involves 6 additional multiplications and 6 additions per cycle.

Of course, the idea above can be extended to the compensation of the phase difference or to the compensation of a general non-linearity as shown previously for the first search method.

The simulation of Figure 7, where the parameter update at time k is $\delta\Theta_k = -\gamma \cdot (\nabla r_k)^T \cdot r_k$, shows how the compensation parameters for offsets, gains and phase difference converge approximately to the corresponding parameter values from the Heydemann correction (for measurements affected by additional distortions, the estimates are necessarily identical, because the objectives functions are different).
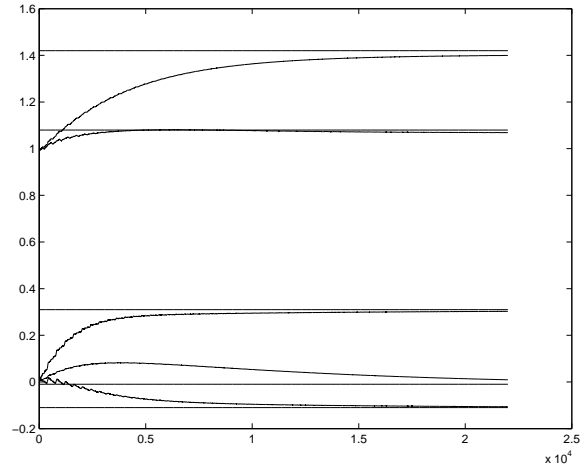

Fig. 7. Simulation of the 5 parameter update for compensation of offsets, gains and phase difference.

6. CONCLUSIONS

In this papers a novel method for calibrating online the compensation parameters for measurements from sinusoidal encoders is discussed. It limits the on-line computation burden, thus allowing its implementation on standard hardware components like DSPs.

The implementation on a fixed-point DSP will follow soon.

REFERENCES

AnalogDevices (2000). Closed-loop position estimation with signal compensation for sinusoidal encoders with the AMD401. Technical Report AN401-23. Analog Devices Inc.

Birch, K. P. (1990). Optical fringe subdivision with nanometric accuracy. *Precision Engineering* **12**(4), 195–198.

Heydemann, Peter L. M. (1981). Determination and correction of quadrature fringe measurement errors in interferometers. *Applied Optics* **20**(19), 3382–3384.

Venema, Steven and Blake Hannaford (1995). Kalman filter based calibration of precision motor control. In: *Proceedings of IROS, Pittsburgh, PA, USA.*