A SPEED LOOP AUTOTUNING METHOD BASED ON SIGNAL INJECTION FOR ELECTRICAL DRIVES

A. Costabeber, P. Mattavelli**, L. Peretti**, M. Zigliotto***

**University of Padova, Dept. of Information Engineering, Padova, Italy (email: alessandro.costabeber@dei.unipd.it)*

***University of Padova, Dept. of Technique and Management of Industrial Systems, Vicenza, Italy
(email: paolo.mattavelli@unipd.it, luca.peretti@unipd.it, mauro.zigliotto@unipd.it)*

Keywords: AC drives, Closed-loop systems, Self-tuning regulators, Harmonic analysis.

Abstract

This paper presents an auto-tuning technique for the speed controller in electrical drives. The method is based on the model reference approach, where a perturbation frequency is injected into the speed control loop and superimposed to the torque reference signal. The tuning is performed by a simple signal processing elaboration before and after the injection point and the controller parameters are adjusted so as to satisfy the specified bandwidth and phase. The method is based on the Goertzel single-tone Direct Fourier Transform (DFT), enabling low computational complexity, ease of application and high immunity to noise uncorrelated to the perturbation frequency. Thus, the amplitude of the injected signal can be very small, practically comparable to the sensor resolution, without relevant interference of the normal drive operation, enabling break-through applications of the on-line controller auto-tuning. The effectiveness and validity of the proposed approach is extensively verified by simulations and experiments on a 3-kW÷1500-rpm induction motor drive.

1 Introduction

In the literature, there exist an impressive quantity of PI autotuning methods and new ones are continuously proposed. However, the industrial scenario shows distaste towards new solutions. Nevertheless, there are some new elements that can overturn that tendency. They are related to the increasing cost of human intervention in the start-up and commissioning stage. This pushes towards automatic tuning methods that lead, without compromises, to performances as close as possible to the standard manual optimisation. Open loop steps, closed loop steps and relay feedback are certainly the most popular techniques for industrial control autotuning. In particular, the success of relay-based autotuning is witnessed by a vast literature [7, 13] and the reasons are that it is not necessary to open the loop, the tuning rules are simple and the specifications may be expressed with well understood indexes, such as the cut-off frequency and the phase margin. Many variations have been proposed for better autotuning, including a saturation relay [13], relay with a P control preload [11] and a two-level relay [10]. Relay methods have anyway some disadvantages [2]. First, in noisy environments, pure relay feedback methods can suffer from chattering, which is not allowed in some processes. As a countermeasure,

a relay with hysteresis can be used. However, its switching period fluctuates and the size of hysteresis should be adjusted iteratively for a given phase angle. A simple and precise model of the oscillations in the relay feedback system is needed [1]. And it is worth to recall that the process is put through oscillation, anyway. Out of the cited relay methods, there exist several other proposals, testifying the live interest for the subject. There exist non-conventional solutions, for example based on fuzzy logic [12] or genetic algorithms [4]. But the methods are quite involved and often proved by simulations only. Moreover, their approach is far from the common industrial practice, and they require the weights of the fitness function, or fuzzy sets shaping and definition, all things that may represent a hard obstacle to the practical implementation. Alternative solutions are for example in [5], (for dc-dc converters), where an adaptive prediction error filter (PEF) is used for an on-line system identification and controller definition. The main disadvantage consists in assuming a repetitive disturbance to be present and in the fact that only Proportional-Derivative (PD) controllers are considered. In [6] a sophisticated tuning procedure for switched mode power supplies (SMPS) is proposed based on Pseudo-Random-Binary-Sequence (PRBS) injection in the control loop. A cross-correlation algorithm and a DFT are then performed to obtain the loop gain frequency response.

This paper offers a simple and effective alternative to all of the aforementioned solutions. As proposed in [3, 8] for digitally controlled SMPS, a perturbation is injected in the control loop at the desired crossover frequency and the signals before and after the injection node are measured and processed by a Goertzel algorithm to derive the specified loop gain and the phase margin. In the paper, the PI regulator parameters are then derived by direct calculation, achieving the desired closed loop performance.

Thanks to the Goertzel single-tone FFT, the injected signal can be very small, almost comparable with the current sensor resolution, and the auto tuning can be performed online, without affecting the normal closed loop operation. The key features are the low computational complexity, the ease of application, the high immunity to uncorrelated noise, the achievement of good accuracy in the identification of the process model and short duration of the tuning phase. All of these features have been tested on a laboratory prototype, and both simulation and experimental evidences are included in the paper.

2 Principle of the proposed technique

The proposed method focuses on the automatic tuning of the speed PI controller of an electrical drive for induction motors, assuming a tuned internal current control loop. The scheme follows to the general model reference approach depicted in Figure 1.

Figure 1: Model reference tuning approach.

The signal $y^*(s)$ represents the reference of the system. $T_d(s)$ is the desired open loop gain transfer function, composed by the desired controller $R_d(s)$ and the desired process $P_d(s)$. $T(s)$ is the actual open loop gain transfer function, composed by the actual regulator $R(s)$ and the actual process $P(s)$. In this work, $R(s)$ is the speed PI regulator, while $P(s)$ accounts for the inner current loop (with the inverter and the motor) and the mechanical load. Consequently, the input of $P(s)$ is the current reference coming from $R(s)$, while $y^*(s)$, $y_d(s)$ and $y(s)$ are respectively the reference speed, the desired speed and the actual speed of the drive. Assuming an unknown process $P(s)$ and a given reference $y^*(s)$, the purpose of the tuning algorithm is to modify the parameters of $R(s)$ (proportional gain $K_{p\omega}$ and time constant $\tau_{i\omega}$) until the dynamic behaviour of $y(s)$ follows that of $y_d(s)$. The signal $y^*(s)$ can be chosen as needed. As a matter of fact, the most important item for open loop gain transfer functions is represented by the crossover frequency f_B , for which a phase margin m_ϕ is also usually specified. In this way, the closed loop transfer function will have the desired dynamical performances, along with the required stability. Thus, $y^*(s)$ can be chosen as a periodic signal with frequency equal to f_B , and the tuning algorithm should modify $K_{p\omega}$ and $\tau_{i\omega}$ in order to ensure the specified magnitude and phase delay for $y(s)$. In particular, at the crossover frequency the magnitude of the open loop gain transfer function shall be equal to one, with a phase delay of $-\pi + m_\phi$ and the same applies to the magnitude and phase delay of $y_d(s)$. The situation depicted in Figure 1, that is, the open loop gain transfer function excited without feedback control, is not acceptable in most cases, due to the lack of robustness and precision. In real applications, closed loop control will be adopted, and the open loop gain transfer function should be evaluated from the closed loop transfer function by means of an indirect measurement. Figure 2 reports the proposed measurement scheme for the derivation of the open loop gain transfer function of the speed control loop.

Figure 2: Speed loop gain measurement.

A sinusoidal disturbance $d(t)$ with frequency f_B and amplitude A_d is added to the output of $R(s)$, which represents the torque-producing current reference. The transfer functions between $d(s)$ and $d_1(s)$ and $d_2(s)$ respectively are the following:

$$\frac{d_1(s)}{d(s)} = -\frac{R(s)P(s)}{1 + R(s)P(s)}, \quad \frac{d_2(s)}{d(s)} = \frac{1}{1 + R(s)P(s)} \quad (1)$$

From (1), it follows:

$$R(s)P(s) = T(s) = -\frac{d_1(s)}{d_2(s)} \quad (2)$$

From (2), the gain and phase of $T(s)$ are derived:

$$\begin{aligned} |T(j\omega)| &= |d_1(j\omega)/d_2(j\omega)| \\ \angle T(j\omega) &= -\pi + \angle d_1(j\omega) - \angle d_2(j\omega) \end{aligned} \quad (3)$$

The actual speed open loop gain transfer function $T(s)$ is evaluated by measuring the sinusoidal signals $d_1(s)$ and $d_2(s)$. As mentioned, the desired open loop gain transfer function $T_d(s)$ shall have the following features at the bandwidth frequency $\omega_B = 2\pi f_B$:

$$|T_d(j\omega_B)| = 1, \quad \angle T_d(j\omega_B) = -\pi + m_\phi \quad (4)$$

Comparing (3) with (4), the tuning target is fulfilled when the signals $d_1(t)$ and $d_2(t)$ have the same amplitude and $d_1(t)$ leads $d_2(t)$ by m_ϕ radians.

The measurement of the amplitude and phase of $d_1(s)$ and $d_2(s)$ are post-processed by a Goertzel algorithm, which will be described in Section 3. In short, the Goertzel algorithm is a single-tone version of the Fast Fourier Transform (FFT), suitable for on-line harmonic analysis when one or few frequencies have to be analysed. It is equivalent to a high-order digital band-pass filter. Figure 3 shows the complete scheme of the proposed tuning solution.

Figure 3: Tuning architecture.

The tuning algorithm is composed by the following steps. At start-up, the speed controller is initialized by a proportional gain $K_{p\omega_{in}}$ only. The sinusoidal disturbance $d(t)$ is then injected, and amplitude and phase of $T(j\omega_B)$ are evaluated by (3). The tuning algorithm calculates the speed PI regulator parameters solving the following system:

$$\begin{cases} P(j\omega_B)R_{in}(j\omega_B) = T_{in}(j\omega_B) \\ P(j\omega_B)R(j\omega_B) = T_d(j\omega_B) \end{cases} \quad (5)$$

where $R_{in}(j\omega_B) = K_{p\omega_{in}}$ and $T_{in}(j\omega_B)$ is the measured speed open loop gain transfer function in the initial step of the procedure, while $R(j\omega_B)$ is a conventional PI regulator:

$$R(s) = K_p \omega \frac{(1 + s\tau_{i\omega})}{s\tau_{i\omega}} \quad (6)$$

Equations (5) can be rewritten as:

$$\frac{T_{in}(j\omega_B)}{R_{in}(j\omega_B)} = \frac{T_d(j\omega_B)}{R(j\omega_B)} \quad (7)$$

From (4), (6) and (7), the PI regulator transfer function in ω_B is derived:

$$R(j\omega_B) = \frac{e^{j(-\pi+m\phi)-\angle T_{in}(j\omega_B)}}{|T_{in}(j\omega_B)|} K_{p\omega, in} \quad (8)$$

Comparing the phase of (6) and (8), $\tau_{i\omega}$ is soon obtained:

$$\begin{aligned} -\pi + m\phi - \angle T_{in}(j\omega_B) &= \arctan(\omega_B \tau_{i\omega}) - \frac{\pi}{2} \\ \Rightarrow \tau_{i\omega} &= \frac{1}{\omega_B} \tan\left(-\frac{\pi}{2} + m\phi - \angle T_{in}(j\omega_B)\right) \end{aligned} \quad (9)$$

By substituting (9) into (6), and evaluating the module of (8), the value of $K_{i\omega}$ is derived:

$$\begin{aligned} \frac{K_{p\omega, in}}{|T_{in}(j\omega_B)|} &= \frac{K_p \omega}{\omega_B \tau_{i\omega}} \sqrt{1 + (\omega_B \tau_{i\omega})^2} \\ \Rightarrow K_{p\omega} &= \frac{\omega_B \tau_{i\omega} K_{p\omega, in}}{|T_{in}(j\omega_B)|} \frac{1}{\sqrt{1 + (\omega_B \tau_{i\omega})^2}} \end{aligned} \quad (10)$$

The tuning technique has high immunity to the measurement noise, due to the high selectivity of the Goertzel algorithm. A disturbance at frequency $f \neq f_B$ does not affect the result of the tuning process significantly. However, any possible uncorrelated disturbance with $f = f_B$ affects both $d_1(t)$ and $d_2(t)$, leading to possibly incorrect results as the amplitude of the disturbance is big enough with respect to $d(t)$.

There are different ways to solve the problem, according to the way the tuning process is used. If the algorithm is executed as a part of a start-up procedure, the signal-to-noise ratio can be improved by increasing the amplitude of the injected signal $d(t)$. Conversely, when the tuning algorithm is exploited as an online procedure, a low amplitude of the perturbation $d(t)$ is preferred, to avoid interferences with the normal drive operations. In the latter case, the noise effect is overcome with a measurement of $d_1(t)$ and $d_2(t)$ before $d(t)$ is injected. The amplitude and phase of $d_1(t)$ and $d_2(t)$ (here due to the disturbances only), will be compensated shortly afterwards, during the actual tuning process. These countermeasures give, the proposed technique an impressively high noise immunity, as testified by several experimental tests. As a side effect of the frequency selectivity of the Goertzel algorithm, it has been found that $d(t)$ has no longer to be strictly sinusoidal being, for example, a simpler square wave signal, with a sterling low-amplitude.

3 The Goertzel algorithm

Discrete Fourier Transform (DFT) or Fast Fourier Transform (FFT) algorithms are used for the detection of the frequency

spectrum of a signal. When a single harmonic detection is required, DFT and FFT algorithms are not computationally efficient, and a single-tone of the DFT, known as the Goertzel algorithm, ensures better performances [9]. The normalised definition of DFT is the following:

$$X(k) = \sum_{r=0}^{N-1} x(r) W_N^{kr}, \quad 0 \leq k, r \leq N-1 \quad (11)$$

where N is the number of samples, $W_N = e^{-j2\pi/N}$, $x(r)$ and $X(k)$ are the sampled signal (sample time $T=1$) and its frequency-domain transformation (sample frequency $F=1/N$), respectively. The Goertzel algorithm is obtained by manipulating (11), considering the following identity:

$$W_N^{-kN} = e^{(j2\pi/N)Nk} = e^{j2\pi k} = 1 \quad (12)$$

Using (12) into the first of (11), it follows:

$$X(k) = W_N^{-kN} \sum_{r=0}^{N-1} x(r) W_N^{kr} = \sum_{r=0}^{N-1} x(r) W_N^{-k(N-r)} \quad (13)$$

Based on (13), the following sequence can be introduced,:

$$y(n) = \sum_{r=0}^{N-1} x(r) W_N^{-k(n-r)} \quad (14)$$

The sequence (14) is equal to $X(k)$ when $n=N$. It represents a convolution between $x(n)$ and W_N^{-kn} , or equivalently the output of a linear system whose impulse response is equal to W_N^{-kn} . The Z-transform of W_N^{-kn} is:

$$H(z) = \underbrace{\frac{1}{(1 - 2\cos(2\pi k/N)z^{-1} + z^{-2})}}_{\text{IIR filter}} \underbrace{(1 - W_N^k z^{-1})}_{\text{FIR filter}} \quad (15)$$

Expression (15) represents the transfer function between the sequence $y(n)$ and the input $x(n)$, split into an IIR filter and a FIR filter. As regards the former, let $s(n)$ be the output of the IIR filter and $S(z)$ its Z-domain value. It follows:

$$X(z) = (1 - 2\cos(2\pi k/N)z^{-1} + z^{-2})S(z) \quad (16)$$

Back into the time domain, it is:

$$s(n) = x(n) + 2\cos(2\pi k/N)s(n-1) - s(n-2) \quad (17)$$

With the input $x(n)$ being sampled during real-time control, the sequence $s(n)$ is computed online. As regards the FIR filter, its output $y(n)$ is computed only for $n=N$, provided that $s(N)$ and $s(N-1)$ have been previously saved:

$$X(k) = y(n)|_{n=N} = s(N) - W_N^k s(N-1) \quad (18)$$

Since W_N^k is a complex number, expression (18) is complex, too. The real and imaginary parts are:

$$\begin{aligned} \text{Re}[X(k)] &= s(N) - \cos(2\pi k/N)s(N-1) \\ \text{Im}[X(k)] &= \sin(2\pi k/N)s(N-1) \end{aligned} \quad (19)$$

Since $x(n)$ is a real signal, (17) shows that the computation of $s(n)$ requires one multiplication (as $2\cos(2\pi k/N)$ is stored as a coefficient) and two additions. Thus, $s(N)$ is obtained with N multiplications and $2N$ additions. The calculation of $\text{Re}[X(k)]$ and $\text{Im}[X(k)]$ requires two multiplications and one addition. Altogether, the Goertzel algorithm then requires $N+2$ multiplications and $2N+1$ additions. This is a net saving, if compared to the DFT in (11), which needs $2N$ multiplications and $2N$ additions to compute a real and imaginary parts of $X(k)$ from a real $x(k)$ signal [9]. However, the real advantage of the Goertzel algorithm for single harmonic analysis resides in its recursive definition (17). The $N+2$ multiplications and $2N+1$ additions are spread over the whole time window, and $s(n)$ is updated with one multiplication and two additions, each sample period. In this way, the algorithm can be effectively used in real time without affecting the execution time of control routines. Moreover, the value of N can be increased as needed with no effect on the computational effort, obtaining a very high DFT selectivity.

4 Simulation results

An intensive investigation using simulation models were carried out in order to test the proposed technique. The drive parameters used for the simulations, which include motor, mechanical and control parameters, are reported in Table 1.

Pole pairs p	2
Stator resistance R_s	3,37 Ω
Rotor resistance referred to the stator R_{sr}	2,97 Ω
Magnetising inductance L_ϕ	0,3962 H
Transient inductance L_t	0,0337 H
Mechanical inertia J_m	10^{-3} kgm ²
Viscous friction B_m	10^{-2} Nms/rad
PWM & current loop sample frequency	10 kHz
Speed loop sample frequency	2,5 kHz

Table 1: Drive parameters for the simulation.

First, the simulations focused on the injection of sinusoidal and non-sinusoidal disturbances, in order to verify that the Goertzel-based tuning algorithm was not affected by undesired injected harmonics. Figure 4 and Figure 5 shows the profile of $d_1(t)$ and $d_2(t)$ in case of a sinusoidal injection of 1 A and a square-wave injection of 0,01 A, respectively. The crossover frequency was of 30 Hz with 70° of phase margin, and an initial $K_{p\omega}=0,01$ A/(rad/s). In both cases, the tuning process returned the same results for the PI regulator parameters, as shown in Figure 6, proving both the sensibility and the selectivity of the technique. Figure 7 shows the Bode diagram of the speed open loop gain transfer function before and after the tuning process. The dashed vertical line at 30 Hz points out a unity gain and a 70° phase margin, as expected.

As long as quantisation effects on the currents are not considered, the amplitude of $d(t)$ does not affect the tuning results. However, as soon as they are introduced, the amplitude of $d(t)$ becomes an important design item. Figure 8 shows the profile of the $K_{p\omega}$ percentage error as function of the ratio (amplitude of $d(t)$)/quantisation. Amplitudes lower than two times the current quantisation step value lead to

sensible errors, which affect the tuning results, and therefore this can be considered a lower bound for the $d(t)$ amplitude.

Figure 4: $d_1(t)$ and $d_2(t)$ signals with sinusoidal injection.

Figure 5: $d_1(t)$ and $d_2(t)$ signals with square-wave injection.

Figure 6: Calculation of the $K_{p\omega}$ and $\tau_{i\omega}$ parameters.

Figure 7: Speed open loop Bode diagram before and after tuning.

Figure 8: $K_{p\omega}$ error as function of the amplitude/quantisation ratio.

5 Experimental results

Extensive experiments were carried on a laboratory test bench composed by a 3-kW/1500-rpm induction motor and a fast control prototyping board. The software for the speed/current cascade control was implemented and debugged in C language. The voltage PWM and current sample frequency was set at 10 kHz, while the speed loop was executed at 2,5 kHz. The quantisation of the current signals was 24,4 mA.

At drive start-up, the tuning procedure was executed fixing f_b to 10 Hz and m_ϕ to $\pi/3$ rad. Figure 9 shows the d_1 and d_2 signals before and after the tuning, when a 3A sinusoidal signal was injected. The signals after the tuning perfectly fulfil the conditions expressed by (3) and (4).

The same tuning procedure was executed during drive operation, with a very low amplitude of $d(t)$ (0,05 A). Figure 10 shows the speed steps before and after the tuning, which is performed in the interval [0 s; 0,5 s], updating the regulator parameters at 0,5 s. Once again, the speed rise time and profile after tuning were in accordance with the constraints on f_b and m_ϕ imposed by the tuning procedure.

The tuning procedure was evaluated taking as terms of comparison the amplitude of $d(t)$, the frequency resolution and the Nyquist frequency of the Goertzel algorithm. Figure 11 reports the $K_{p\omega}$ percentage error as function of the $d(t)$ amplitude (normalised to the quantisation interval, solid line), and $K_{p\omega}$ percentage error as function of the $d(t)$ frequency (normalised to the Goertzel resolution frequency, dashed line). It is confirmed that consistent results are obtained as long as the amplitude of the injected signal is at least two times the quantisation value. Lower values are meaningless, since distorted or no torque oscillations are induced into the mechanical load, and it is unworthy to seek any information from the measurements. It was also observed that a frequency resolution of about 6÷10 times smaller than the observed frequency is enough for reliable results. For a 10-Hz injected signal, this translates into a Goertzel time window of 0,6÷1 s.

Figure 12 reports the $K_{p\omega}$ percentage error as function of the Nyquist frequency of the Goertzel algorithm. When the latter is executed at the same rate of the current of the speed loop, results are quite reliable. The Goertzel algorithm could be

executed even at a lower rate, but it is not recommended to go below 40÷50 times the frequency of $d(t)$.

Figure 9: $d_1(t)$ and $d_2(t)$ before (left) and after (right) tuning.

Figure 10: On-line tuning with low-amplitude $d(t)$ injection.

Figure 11: Quantisation and frequency resolution effects on $K_{p\omega}$ error.

Figure 12: Nyquist frequency effect on $K_{p\omega}$ error.

The robustness of the tuning procedure against other system disturbances, not injected on purpose, was also verified by forcing the injection of a signal $d_3(t)$ before the signal $d_1(t)$ of Figure 2. It is expected that $d_3(t)$ has no effect on the tuning results when its frequency is different to that of $d_1(t)$, while negative effects are expected when frequency is the same. Figure 13 reports the measured $K_{p\omega}$ percentage error when $d_3(t)$ is a sinusoidal signal with a frequency of f_b and the tuning procedure is executed.

Figure 13: Effect of deterministic $d_3(t)$ injection before $d_1(t)$.

The tuning deviates from the correct results, proportionally with the amplitude of $d_3(t)$. The worst case occurred when $d_3(t)$ was out of phase with respect to $d(t)$. However, when the phase of $d_3(t)$ is randomly changing, tuning results are much more reliable, as shown in Figure 14. Thus, the tuning procedure is robust against random noise, while the precautions mentioned in Section 2 should be considered in case of deterministic disturbances at the same frequency of f_b .

Figure 14: Effect of random $d_3(t)$ injection before $d_1(t)$.

6 Conclusions

A model-reference based tuning method for the speed regulator in electrical drives was presented. The controller tuning is obtained by injecting a perturbation frequency in the closed-loop speed control and by measuring the amplitude and phase of the signals before and after the injection points. The most relevant features of the proposed solution are the excitation at the desired cross-over frequency and the use of Goertzel algorithm for the signal processing elaboration. Using these provisions, the frequency identification is reduced to a few elementary operations at each sampling instant (i.e. a combination of a IIR second-order filter and a

recursive FIR filter). The tuning of the PI controller, which ensures the desired cross-over frequency and phase margin, is obtained using a single direct calculation. The resulting approach is also suitable for on-line tuning since the excitation signal can be made very small, being the algorithm highly immune to noise uncorrelated to the perturbation frequency. Simulation and experimental results have proven the effectiveness and accuracy of the proposed method, in spite of the small amplitude of the excitation signal.

References

- [1] I. Boiko, "Autotune Identification via the Locus of a Perturbed Relay System Approach", *IEEE Trans. on Control Systems Tech.*, vol. 16, no. 1, pp. 182-185, 2008.
- [2] J. Byeon, J-S. Kim, D. Chun, S. W. Sung, J. Lee, "Third quadrant Nyquist point for the autotuning of PI controllers", *ICCAS-SICE*, pp. 3283-3286, 2009.
- [3] L. Corradini, P. Mattavelli, W. Stefanutti, S. Saggini, "Simplified Model Reference-Based Autotuning for Digitally Controlled SMPS", *IEEE Trans. Power Electron.*, vol. 23, no. 4, pp. 1956-1963, 2008.
- [4] L. Fan, E. M. Joo, "Design for auto-tuning PID controller based on genetic algorithms", *IEEE Conf. on Industrial Electronics and Applications*, pp. 1924-1928, 2009.
- [5] A. Kelly, K. Rinne, "A self-compensating adaptive digital regulator for switching converters based on linear prediction", *21th Ann. IEEE Applied Power Electronics Conference and Exposition (APEC)*, Dallas, USA, pp. 712-718, 2006.
- [6] B. Miao, R. Zane, D. Maksimovic, "System identification of power converters with digital control through cross-correlation methods", *IEEE Trans. Power Electron.*, vol. 20, no. 5, pp. 1093-1099, 2005.
- [7] N. Miskovic, Z. Vukic, M. Barisic, "Transfer function identification by using self-oscillations", pp.1-6, *15th Medit. Conf. on Control and Automation*, Athens, 2007.
- [8] J. Morroni, R. Zane, D. Maksimovic, "Design and Implementation of an Adaptive Tuning System Based on Desired Phase Margin for Digitally Controlled DC-DC Converters", *IEEE Trans. Power Electron.*, vol. 24, no. 2, pp. 559-564, 2009.
- [9] A. V. Oppenheim, R. W. Schaffer, *Discrete-Time Signal Processing, Third Edition*, Prentice-Hall, Inc., Upper Saddle River, NJ, 2009.
- [10] S. W. Sung, I-B. Lee, J. Lee, "Modified Proportional-Integral Derivative (PID) Controller and a New Tuning Method for the PID Controller", *Ind. Eng. Chem. Res.*, vol. 34, no. 11, pp. 4127-4132, Nov. 1995.
- [11] K. K. Tan, T. H. Lee, S. Huang, K. Y. Chua, R. Ferdous, "Improved critical point estimation using a preload relay", *Journal of Process Control*, vol. 16, no. 5, pp. 445-455, June 2006.
- [12] J. Yoshitsugu, K. Inoue, M. Nakaoka, "Fuzzy autotuning scheme based on α -parameter ultimate sensitivity method for AC speed servo system", *IEEE Trans. on Industry Applications*, vol. 36, no. 2, pp. 492-499, 2000.
- [13] C. C. Yu, *Autotuning of PID Controllers - A Relay Feedback Approach*, ISBN: 978-1-84628-036-8, 2006.