Self-tuning PLL. A new, easy, fast and highly efficient phase-locked loop algorithm

Reyes Sánchez-Herrera[1], José M. Andújar[2], *Senior, IEEE*, Marco Márquez[3], Andrés Mejías[4], Gabriel Gómez-Ruiz[5]

***Abstract*— A phase-locked loop or phase lock loop (PLL) is, essentially, a closed loop control system, where its output signal (regulated) maintains a direct (chosen) relationship to the input signal (unregulated). If the phases of the input and output signals remain the same, their frequencies will also match. In its usual way, a PLL is an electronic circuit with filters and controllers that must be tuned each time, according to the input signal parameters. As a contribution and novelty, this paper introduces a new, easy, fast and highly efficient PLL algorithm, that it does not need to adjust every time the input signal changes. This makes it independent of the input signal it receives and, therefore, and in a certain way, universally applicable. In addition, the proposal is implemented exclusively by software, housed in a microcontroller, which also represents another novelty. As another remarkable feature, the proposed algorithm is very simple, with a very low computational cost, which makes it very stable, practically insensitive to noise, and very fast. These characteristics and the performance shown over a wide range of input signal frequencies, make the proposed algorithm suitable for use in different applications, from the electricity grid synchronization to the demodulation of frequency modulation, DC motor drives, etc. In addition to frequency and phase tracking, the developed algorithm generates the signal corresponding to the fundamental frequency of the input signal. Furthermore, the algorithm behavior is not affected by the input signal distortion and is independent of its initial phase. The performance and excellent behavior of the developed algorithm are evaluated through simulations and experimental tests.**

***Index Terms*— phase-locked loop, PLL algorithm, self-tuning PLL, synchronization, demodulation, modulation.**

I. Introduction

PHASE-LOCKED loop (PLL) is required for applications where a signal whose phase is directly related to that of input must be generated, as for example in communication systems, microprocessors, dc motor drives, [1] or mechanical encoders, [2]. Furthermore, the determination of the phase and frequency of the supply voltage is vital in the synchronization of the DC/AC converters to the electricity grid, [3]–[6], in electric power quality analysis, [7], [8], electrical measurements, [9], etc.

In its simplest form, a PLL is made up of 3 blocks, Fig. 1. A variable frequency controlled oscillator (VCO) generates a periodic signal whose phase is compared, using a phase detector (PD), with the phase of the input periodic signal, *InpS*(*t*). The produced error signal is filtered by the loop filter (LF), removing unwanted components of the phase comparison frequencies, generating a control signal to tune the oscillator to keep the phases matching.

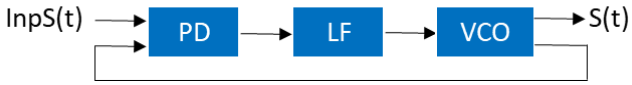


Fig. 1. PLL basic structure.

Despite the developed algorithm can be of application in practically all the PLL uses, this article will focus on the application to the electricity grid. In the electricity grid synchronization field, the way of implementing the PD (the pivotal element in the PLL) gives rise the different technologies present in the literature, as the synchronous reference frame PLL (SRF-PLL), [5], the second-order generalized integrator PLL (SOGI-PLL), [7], [10], or the enhanced PLL (EPLL), [11], [12], among others, as presented in [13]-[16]. In [13], an improved demodulation technique is presented, which uses double demodulation without recreating the double frequency component for rejection purpose. In [14], a demodulation-based technique is proposed that is less affected by the dc offset. In [15] an open loop approach is proposed for the selective detection of harmonics in a variable frequency environment. Finally, [16] presents a modified demodulation-based technique that has been improved/modified to function as PLL. All [13] to [16] PLLs are electronic circuits constituted by elements that must be tuned to the input signal frequency. Due on the one hand to this (the proposed algorithm in this paper is self-tuning), and on the other hand that they are not commonly used in the electricity grid, it has not been considered illustrative to compare them with the algorithm developed in this research.

SRF-PLL is based on creating a quadrature signal by delaying the original single-phase signal by one fourth of the electricity grid fundamental period, [17]. There are as many kinds of SRF-PLL as different methods to create the quadrature signal, [18], [19]. SRF-PLL extracts the phase, frequency, and amplitude of the fundamental component of the input signal but not as a waveform at the output of the algorithm.

Affiliation: Research Centre for Technology, Energy and Sustainability, University of Huelva. [1]reyes.sanchez@die.uhu.es, [2]andujar@uhu.es, [3]marcoa@pi.uhu.es, [4]mjias@uhu.es, [5]gabriel.gomez936@alu.uhu.es,

This paper is framed in the project "Construction, commissioning and testing of a remotely programmable DC / AC inverter prototype that can be used connected to the grid or to power isolated loads", funded by the Andalucía government, PAIDI 2017, call for technology transfer projects.

EPLL is also very popular in single-phase applications. It uses an adaptive notch filter to improve the PD performance. EPLL extracts the fundamental component of the input signal, and estimates its phase, frequency, and amplitude accurately [11], although it does not show rejection of the disturbances in any case [20].

Regarding SOGI-PLL, it provides the phase, frequency, and amplitude of the fundamental component of the input signal, and presents good disturbance rejection, fast transient response, and robust performance [21]. However, the cost to pay is its higher complexity [19], [22], [23]. SOGI-based frequency-locked loop (SOGI-FLL) is one of the most popular second-order PLL, [24], because it presents good performance when the electricity grid voltage presents nonconformities as voltage sag/swell, or imbalance, among other disturbances [25].

Practically, most of the PLL techniques published in the literature require fixed or adaptive filters in the PD block, [26]–[28]. They also include the LF and the VCO. Thus, they are constituted by elements that have to be tuned according to the input signal parameters. Also, PLL lock time increases due to the complexity of its configuration and computation time.

This work presents a new PLL algorithm, called self-tuning (ST-PLL), which is not, as usual, an electronic device, but a software algorithm. As another of the main novelties, the proposed PLL does not need to be tuned. This fact makes the ST-PLL algorithm adapt to any type of signal, from the electricity grid voltage (400/230 V RMS –root mean square– and 50/60 Hz) to the output of a frequency modulator (1 V RMS or less, and variable frequency in the range of kHz-MHz), going through many other applications with intermediate characteristics.

Furthermore, the ST-PLL algorithm is simple, which makes the PLL lock time very short. This feature is mandatory to be able to track the signal of a frequency modulator, as well as the electricity grid voltage transient non-conformities, such as voltage dip/surge or frequency variations, among others. Another very important feature of the ST-PLL algorithm is its immunity to input signal distortion. This last characteristic is pivotal because it helps to reduce the increasing harmonic distortion of the frequency of the voltage and the current in the electricity grid, due to the increasing use of power electronic and communication devices. Finally, the ST-PLL algorithm is very easy to implement and, in short, it can make an important contribution to the field of electrical power quality and electrical measurements, among others.

Table I summarizes the main characteristics of the reviewed literature on PLL proposals. To clearly distinguish the contributions and novelty of this work, the authors' proposal is presented in the last row. The fact that it can be fully implemented in a microcontroller (see section IV) opens the possibility of making the proposed PLL accessible remotely, among other advantages.

The rest of the paper is organized as follows: in section II, the ST-PLL algorithm theoretical foundation is presented, which, as will be shown, is very easy and intuitive. In section III, different and extreme situations are simulated to test the efficiency and robustness of the algorithm. The results are compared and discussed with those obtained by the EPLL and SOGI-PLL algorithms (SRF-PLL does not provide a waveform to its output). In section IV, the excellent experimental results of the ST-PLL algorithm acting on a real system (the electricity grid) are shown. Finally, in section V, some conclusions are drawn.

TABLE I PLL COMPARATIVE

	SRF-PLL [5]	EPLL [11]	SOGI-PLL[10]	ST-PLL
After being tuned in, tracking the input signal phase and frequency.	x	x	x	x
Non affected by the input signal distortion.	x		x	x
Simplicity and low computational cost.	x	x		x
Provides the waveform of the fundamental component of the input signal.		x	x	x
Follows the phase and frequency of the input signal without needing to be tuned each time.				x
It can be easily integrated into a microcontroller because it is completely software.				x

II. FOUNDATION OF THE ST-PLL ALGORITHM

The developed algorithm is based on the well-known adaptive filter based on the Fourier series, whose objective is to obtain the fundamental harmonic or fundamental component at the fundamental frequency of a signal. This type of filter is widely used in power converters applications for the synchronization to the electricity grid. The fundamental reason is that the frequency of the electricity grid has a constant and well-known value.

Then, given a signal $InpS(t)$, it can be expressed as a sum of components with frequencies multiple of the fundamental (harmonic components). Therefore, $InpS(t)$ can be expressed mathematically as

$$InpS(t) = \frac{a_0}{2} + \sum_{n=1}^{\infty} A_n \cos(n\omega_0 t + \varphi_n) \quad (1)$$

Where $a_0/2$ is the constant term, or DC component, of $InpS(t)$. $A_n\cos(n\omega_0 t + \varphi_n)$ is the *n*th harmonic that, for $n = 1$, represents the first harmonic, or fundamental, with fundamental frequency ω_0. From it, for $n = 2,3,\ldots$, the rest of harmonics can be obtained. Then, the goal of the adaptive filter is to obtain the fundamental harmonic $A_1\cos(\omega_0 t + \varphi_1)$ from (1). To do this, the first step is to multiply (1) by the unitary sine and cosine at the fundamental frequency, and then calculate the mean value of the obtained expressions, (2) and (3).

$$SinS(t) = \frac{1}{T}\int_{t_0}^{t_0+T} \frac{a_0}{2}\sin\omega_0 t\, dt + \frac{1}{T}\sum_{n=1}^{\infty}\int_{t_0}^{t_0+T} A_n\cos(n\omega_0 t + \varphi_n)\sin\omega_0 t\, dt \quad (2)$$

$$CosS(t) = \frac{1}{T}\int_{t_0}^{t_0+T} \frac{a_0}{2}\cos\omega_0 t\, dt + \frac{1}{T}\sum_{n=1}^{\infty}\int_{t_0}^{t_0+T} A_n\cos(n\omega_0 t + \varphi_n)\cos\omega_0 t\, dt \quad (3)$$

As is well known, the mean value of a sinusoidal signal in one period is zero, as well as the mean value of the product of sinusoidal signals with different frequencies. So, (2) and (3) only have nonzero for $n = 1$, giving rise to two time-independent expressions, i.e., the two constant expressions (4) and (5).

$$SinS(t) = -\frac{A_1}{2}\sin\varphi_1 \tag{4}$$

$$CosS(t) = \frac{A_1}{2}\cos\varphi_1 \tag{5}$$

Now, multiplying (4) and (5) by 2 and, respectively, by the unitary sine and cosine at the fundamental frequency, it allows, adding the two resulting expressions, to obtain the complete fundamental harmonic (amplitude, frequency and phase), $FundS(t)$, expression (6).

$$FundS(t) = A_1\cos(\omega_0 t + \varphi_1) \tag{6}$$

The way to practically implement the mathematical operations performed is shown in Fig. 2. The mean values of the sinusoidal signals are obtained by low pass filters (LPF), tuned to a cutoff frequency that, for applications in power converters for synchronization to the electricity grid, are within a maximum of 10 or 12 Hz (depending on the fundamental frequency of the electricity grid, 50 or 60 Hz), which implies a very slow dynamic response.

The key to obtaining the clean signal shown in (6) is to know the fundamental frequency of the original signal $InpS(t)$. This is necessary to choose the frequency of the unitary sine and cosine that allows eliminating the harmonics of the original signal. But what if the fundamental frequency of the original signal $InpS(t)$ is not known? It so happens that the adaptive filter does not work correctly and, in most cases, expression (6) is not achieved.

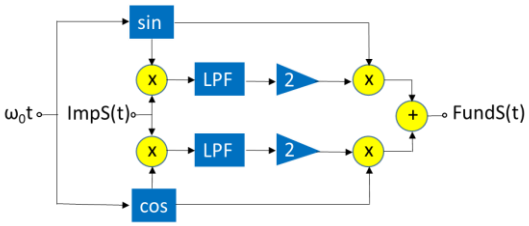


Fig. 2. Adaptive filter based on the Fourier series.

Suppose that fundamental frequency ω_0 of the input signal $InpS(t)$ is unknown; then the frequency that is chosen for the unitary sine and cosine will generally not match it. Assume this is $\omega_c = \omega_0 - \Delta\omega$, where $\Delta\omega$ can be positive or negative. So, considering T_c the sampling time (the subscript c indicates since, logically, it must be related to ω_c), the expression (2) can now be written as (7).

$$SinS\Delta(t) = \frac{1}{T_c}\int_{t_0}^{t_0+T_c}\frac{a_0}{2}\sin(\omega_0 t - \Delta\omega t)\,dt + \frac{1}{T_c}\sum_{n=1}^{\infty}\int_{t_0}^{t_0+T_c}A_n\cos(n\omega_0 t + \varphi_n)\sin(\omega_0 t - \Delta\omega t)\,dt \tag{7}$$

Reasoning as from (2) to (4),

$$SinS\Delta(t) = \frac{1}{T_c}\int_{t_0}^{t_0+T_c}A_1\cos(\omega_0 t + \varphi_1)\sin(\omega_0 t - \Delta\omega t)\,dt \tag{8}$$

$$= -\frac{A_1}{T_c}\int_{t_0}^{t_0+T_c}(\cos^2\omega_0 t\cos\varphi_1\sin\Delta\omega t + \sin^2\omega_0 t\sin\varphi_1\cos\Delta\omega t)dt + \cdots$$

Ellipsis in equation (8) represent the rest of the integrals, all null. Now, taking into account the relationships $\cos^2\omega_0 t = \frac{1+\cos 2\omega_0 t}{2}$ and $\sin^2\omega_0 t = \frac{1-\cos 2\omega_0 t}{2}$, it is easy to obtain (9).

$$SinS\Delta(t) = -\frac{A_1}{2T_c}\int_{t_0}^{t_0+T_c}(\cos\varphi_1\sin\Delta\omega t + \sin\varphi_1\cos\Delta\omega t)dt + \cdots = \frac{A_1}{2\Delta\omega T_c}\cos(\Delta\omega t + \varphi_1) \tag{9}$$

Where, again, ellipsis represent the rest of the integrals, all null.

Operating the same way with expression (3), (10) is obtained.

$$CosS\Delta(t) = \frac{A_1}{2\Delta\omega T_c}\sin(\Delta\omega t + \varphi_1) \tag{10}$$

At this point, it is important to show how the developed algorithm works. Well, if $\Delta\omega$ is brought to zero, $\omega_c = \omega_0 - \Delta\omega$ is brought to ω_0. Once this is achieved, the adaptive filter topology (Fig. 2) based on the Fourier series is applicable. So, the first objective of the algorithm is to get $\Delta\omega \to 0$.

With this objective in mind, from (9) and (10) it is easy to show (11).

$$|\Delta\omega| = \frac{1}{t}\left|\tan^{-1}\left(\frac{CosS\Delta(t)}{SinS\Delta(t)}\right)\right| \tag{11}$$

Boundaries of function tan are shown in [29], where a PLL is presented with wide hold-in, pull-in, and lock-in ranges. In addition, it must be noticed that if $\Delta\omega$ goes to zero, $SinS\Delta(t)$, equation (9), is positive, and $CosS\Delta(t)$, equation (10), also goes to zero.

From here, the reasoning is the following. If $\Delta\omega < 0 \Rightarrow \omega_c > \omega_0$, so, ω_c must be decremented by the amount (11). On the contrary, if $\Delta\omega > 0 \Rightarrow \omega_c < \omega_0$, so, ω_c must be increased by the amount (11). The sign of $\Delta\omega$ is easily detected by the sign of $CosS\Delta(t)$, equation (10), as mentioned above. The explained way of proceeding is reflected in equation (12).

$$\begin{cases} if\ \Delta\omega < 0 \Rightarrow \omega_c(T_u + 1) = \omega_c(T_u) - |\Delta\omega| \\ if\ \Delta\omega > 0 \Rightarrow \omega_c(T_u + 1) = \omega_c(T_u) + |\Delta\omega| \end{cases} \tag{12}$$

Where T_u is the update time of $\Delta\omega$ which does not have to be equal to T_c (this is clarified a bit later). The flow chart of the developed algorithm, which provides the complete fundamental harmonic (amplitude, frequency and phase) of an unknown input signal is shown in Fig. 3. As can be seen, it is very simple.

The term *accuracy* (allowed frequency deviation threshold) in figure 3 represents one of the two parameters which makes the algorithm stable and robust. In fact, *accuracy* avoids the problems associated with possible instability caused by low values (close to zero) of $\Delta\omega$ in the denominators of equations (9) and (10). To compute these equations, T_C is automatically adapted to the chosen ω_c. The second parameter (n_u) is the number of times per cycle of the input signal that is updated $\Delta\omega$. That is, the update of $\Delta\omega$ is not performed for each sampling time (this unnecessarily slows down the algorithm). Our experience tells us that two updates per cycle are sufficient, although obviously if for a specific case it does not work well,

this value can be easily changed in the algorithm. So for example, for $n_u = 2$ the algorithm will automatically update $\Delta\omega$ every 10 ms (T_u in (12) is 10 ms) if the frequency of $InpS(t)$ is 50 Hz, but it will update every T_u = 5 ns if it is 100 MHz. This makes the algorithm independent of the frequency of the input signal, allowing it to automatically self-adjust to a wide range of input signal frequencies.

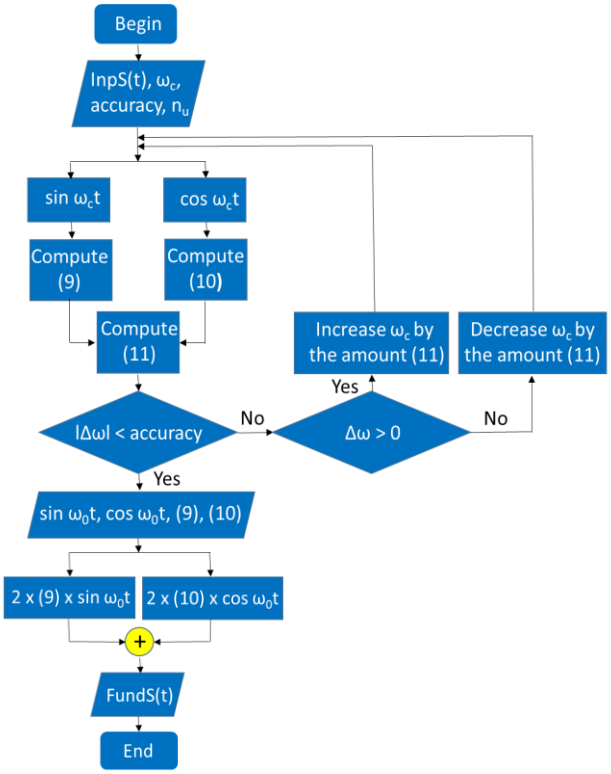


Fig. 3. Flowchart of the developed ST_PLL algorithm, which provides the complete fundamental harmonic (amplitude, frequency and phase) of an unknown input signal.

Continuing with Figure 3, once it is fulfilled that $\Delta\omega <$ accuracy, ω_0 and ω_c are, in practice, equals, so, the last computed values of (9) and (10) are, respectively, (4) and (5). In the same way, $\sin \omega_c t$ and $\cos \omega_c t$ are, respectively, $\sin \omega_0 t$ and $\cos \omega_0 t$. Then, multiplying (9) and (10) by 2 and, respectively, by the unitary sine and cosine at the fundamental frequency ω_0, it allows, adding the two resulting expressions, to obtain the complete fundamental harmonic (amplitude, frequency and phase), $FundS(t)$. Finally, regarding the meaning of the t of equation (11) in the algorithm, it is the simulation time.

III. Performance of the ST-PLL Algorithm

In this section, the results provided by the developed ST-PLL algorithm are presented. To evaluate the algorithm, a distorted signal ($InpS(t)$) has been used as input, the parameters of which are presented in Table 2 (obviously these data are unknown to the algorithm). Fig. 4 shows a period of the input signal, where its distortion can be clearly observed. The simulation time has been set to 1 s and the sampling time to 5 µs. Of course, if necessary the algorithm can work with shorter sampling times. The accuracy (Fig. 3) has been set in 0.03 rad/s.

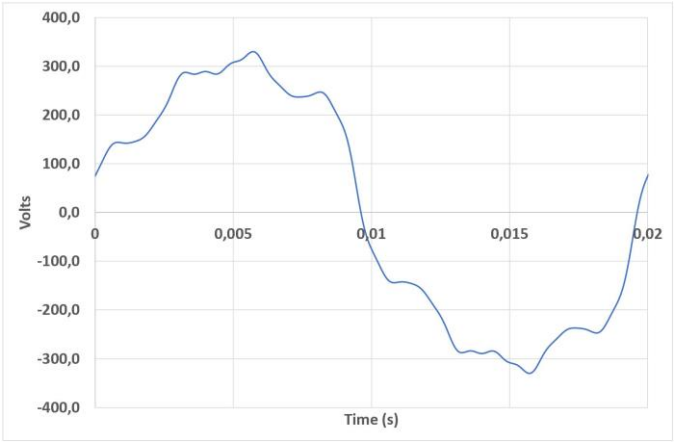


Fig. 4. Input signal, $InpS(t)$, waveform.

TABLE 2 Input signal, $InpS(t)$, parameters

Harmonic order	FC	H3%	H5%	H7%	H11%
RMS value (V)	230	10	10	10	1
Frequency (Hz)	50	150	250	350	550
Phase (degrees)	0	30	50	70	110

Harmonic order	H13%	H17%	H19%	H23%
RMS value (V)	1	1	1	1
Frequency (Hz)	650	850	950	1150
Phase (degrees)	130	130	190	230

As can be seen in Fig. 4, the input signal has a non-zero value at t = 0, although the phase of the fundamental component is zero (it is due to the phases of the harmonics). The chosen input signal complicates the calculation of its parameters, so it can be a good test signal to evaluate the performance of the ST-PLL algorithm.

In what follows, from the chosen input signal, changes will be made to its parameters to show the performance of the developed algorithm under different conditions, but always preserving the waveform (Fig. 4).

A. ST-PLL algorithm performance under different conditions

Case 1. This is considered the basic case; the input signal is exactly the one in Fig. 4. The estimated (chosen) frequency (ω_c) is 100π rad/s (the same as ω_0). The complete fundamental harmonic of the input signal (amplitude, frequency, and phase) provided by the ST-PLL algorithm is presented in Fig. 5.

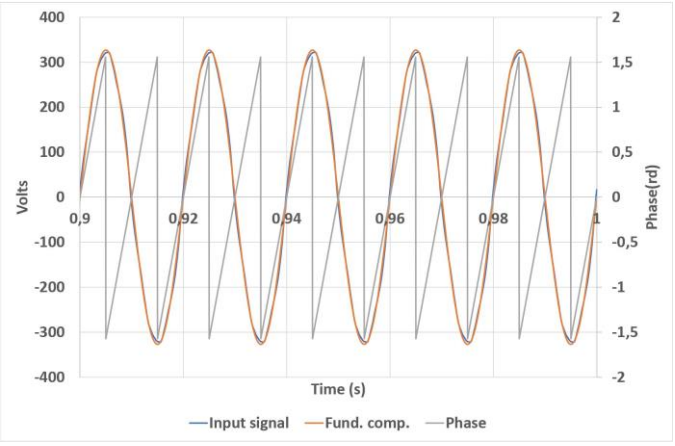


Fig. 5. The ST-PLL algorithm accurately calculates the fundamental harmonic of the input signal. Its frequency, phase, and RMS values are 50 Hz, 0º, and 230 V, respectively.

Case 2. As most of the algorithms presented in the literature are tuned to the electricity grid values (230 V / 50 Hz or 120 V / 60 Hz) and, in general, there is no evidence of their performance with other frequencies and RMS values, this test will show the adaptability of the algorithm developed. For that, now $\omega_c = \omega_0 = 196\pi$ rad/s, and the RMS value of the fundamental harmonic is 15 V. The complete fundamental harmonic of the input signal (amplitude, frequency, and phase) provided by the ST-PLL algorithm is presented in Fig. 6.

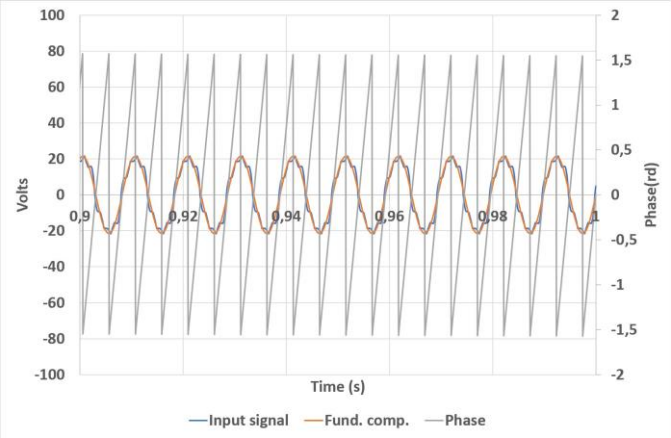


Fig. 6. The ST-PLL algorithm accurately calculates the fundamental harmonic of the input signal. Its frequency, phase and RMS values are 98 Hz, 0º and 15 V, respectively.

Case 3. In this case, the input signal fundamental harmonic frequency, RMS, and phase values are now 48π rad/s, 450 V, and 80º, respectively. Again $\omega_c = \omega_0 = 48\pi$ rad/s, and again the tracking is as good as in cases 1 and 2 (see Fig. 7).

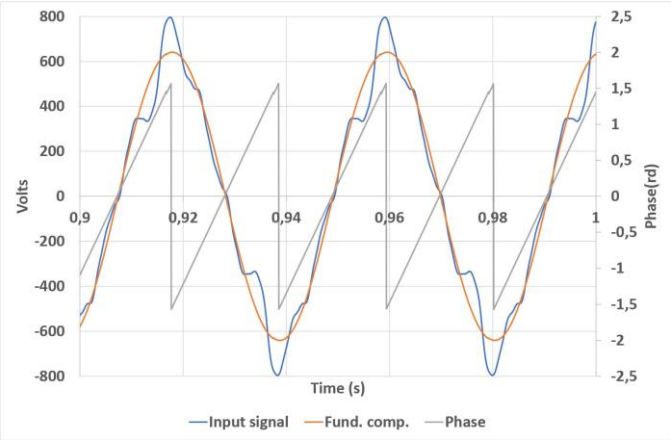


Fig. 7. The ST-PLL algorithm accurately calculates the fundamental harmonic of the input signal. Its frequency, phase, and RMS values are 24 Hz, 80º, and 450 V, respectively.

B. Performance comparison of ST-PLL algorithm versus EPLL and SOGI-PLL algorithms at the electricity grid frequency

Now, the performance of the developed algorithm will be compared with those of the most popular algorithms presented in the literature, EPLL and SOGI-PLL. SRF-PLL is discarded because it does not provide the fundamental component waveform. To facilitate the reproducibility of the comparison of the three algorithms performed in this work, Figures 8 and 9 present the EPLL [12] and SOGI-PLL [10] used block diagrams, respectively.

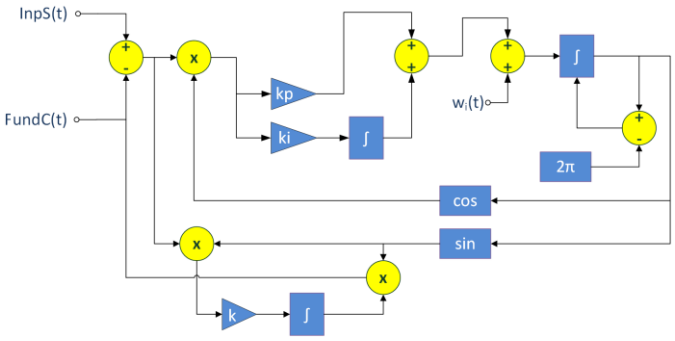


Fig. 8. EPLL block diagram.

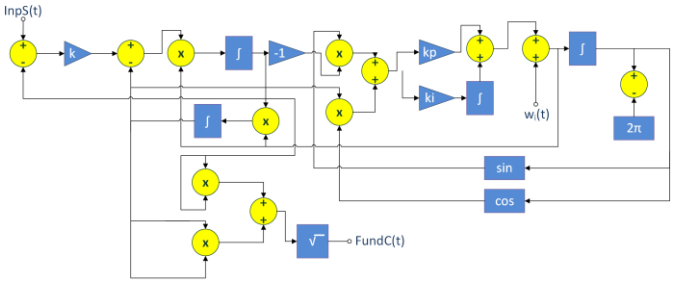


Fig. 9. SOGI-PLL block diagram.

The algorithms are tested under the most unfavorable condition: instantaneous change of frequency, phase, and RMS value of the input signal. In particular, an abrupt change of the fundamental component has been imposed in t = 0.2 s: the frequency increases from 100π to 110π rad/s (note that just after the instant of change, $\omega_0 = 110\pi$ rad/s, but $\omega_c = 100\pi$ rad/s, so there is an $\Delta\omega = 10\pi$ rad/s), the phase changes from 30 to -90º and RMS value increases from 230 to 500 V.

Figs. 10 to 12 show the output of EPLL, SOGI-PLL, and ST-PLL algorithms, respectively, to the described abrupt change of the fundamental component. As can be seen, the EPLL (Fig. 10) and ST-PLL (Fig. 12) algorithms quickly achieve frequency and phase tracking of the fundamental component; measured in detail in just 60 ms. However, ST-PLL is considerably faster to capture the RMS value. Measured in detail, the EPLL algorithm takes just over 3 s, while the ST-PLL algorithm takes just 1 s.

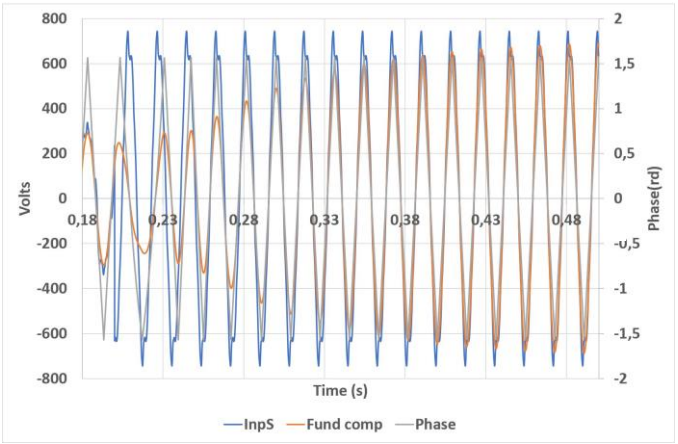


Fig. 10. Results obtained applying the EPLL algorithm when an abrupt change of the fundamental component has been imposed in t = 0.2 s: the frequency increases from 100π to 110π rad/s, the phase changes from 30 to -90º and RMS value increases from 230 to 500 V.

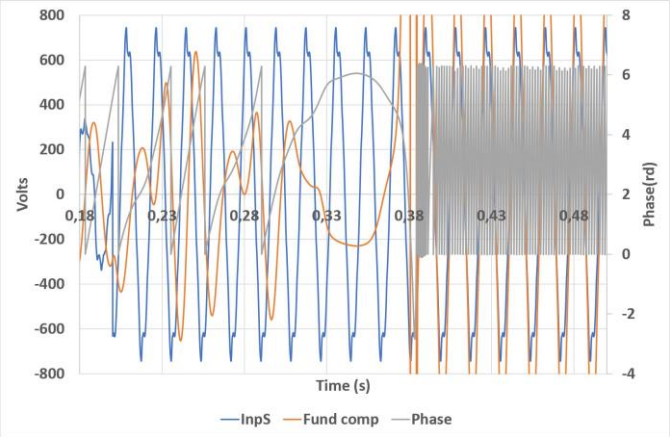


Fig 11. Results obtained applying the SOGI-PLL algorithm when an abrupt change of the fundamental component has been imposed in t = 0.2 s: the frequency increases from 100π to 110π rad/s, the phase changes from 30 to -90º and RMS value increases from 230 to 500 V.

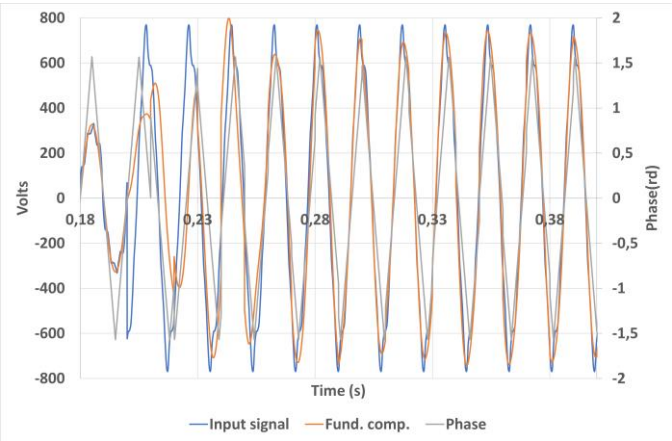


Fig. 12. Results obtained applying the ST-PLL algorithm when an abrupt change of the fundamental component has been imposed in t = 0.2 s: the frequency increases from 100π to 110π rad/s, the phase changes from 30 to -90º and RMS value increases from 230 to 500 V.

Regarding the SOGI-PLL algorithm (Fig. 11), it is clearly seen that it cannot track the abrupt change in the input signal. But why? Obviously, because it needs to tune its LPF which is intrinsic to its structure. In fact, tuning the bandwidth of its LPF from 100π rad/s (twice that of the electricity grid as usual) to 400π rad/s, the result is shown in Fig. 13. Now, the behavior of the algorithm is similar to ST-PLL. Notwithstanding, this is not a practical solution for the SOGI-PLL algorithm because the perturbations are unexpected.

As a result of this comparison, the ST-PLL algorithm is at least three times faster than the EPLL algorithm at capturing abrupt changes in the input signal, and with the enormous advantage over the SOGI-PLL of not having to re-tune every time.

Regarding the behavior of the developed algorithm in the face of phase jumps and voltage dips, Fig. 12 presents the result when simultaneous abrupt frequency, phase, and rms values jump happen in a distorted input signal. As can be seen and has already been commented, ST-PLL adapts quickly to the new conditions. However, to further highlight the excellent behavior of the developed algorithm against disturbances in the grid, Fig. 14 presents the results of its application when a voltage dip occurs. As can be seen, the rms voltage is tracked in just one cycle, and the phase and frequency are not affected.

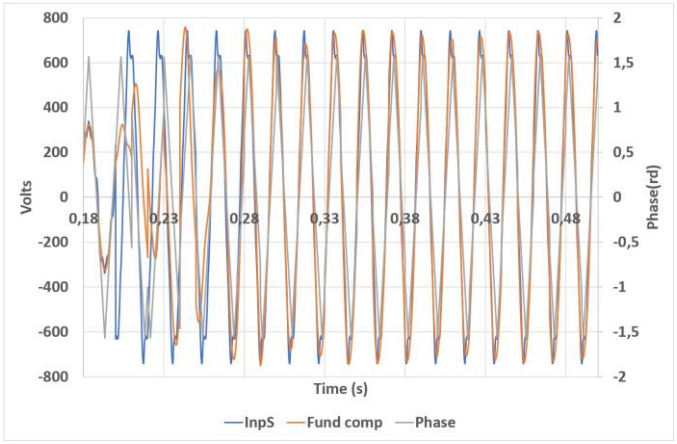


Fig. 13. Results obtained applying the SOGI-PLL algorithm, tuning previously the bandwidth of its LPF to 400π rad/s, when an abrupt change of the fundamental component has been imposed in t = 0.2 s: the frequency increases from 100π to 110π rad/s, the phase changes from 30 to -90º and RMS value increases from 230 to 500 V.

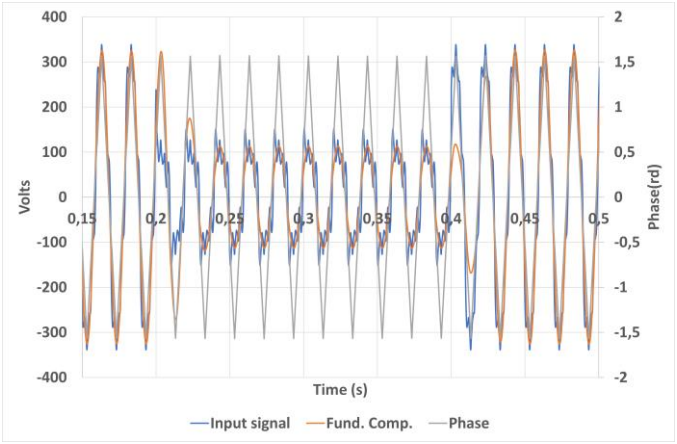
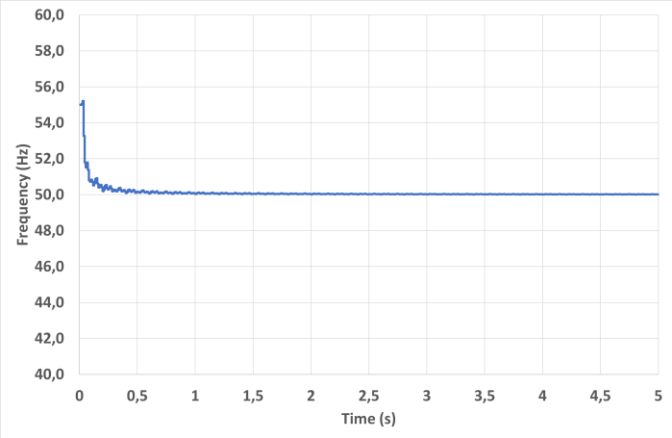


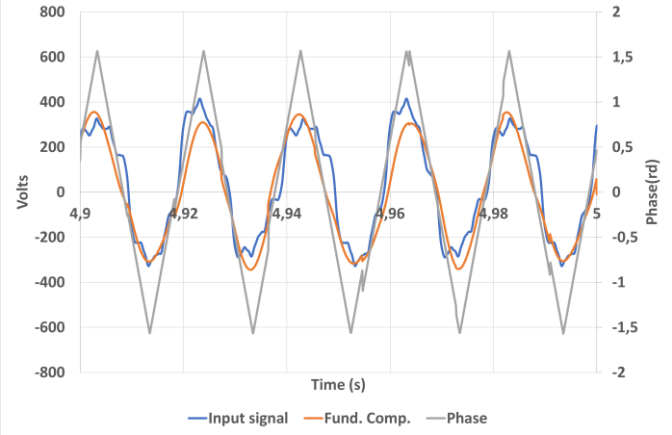
Fig. 14. Results obtained by applying the ST-PLL algorithm when there is a voltage dip in the electricity grid.

Finally, it must be considered that in practical supply systems, there may be inter-harmonics caused by loads operating in a transient or saturated state, switched converters not synchronized to the fundamental frequency or, recently, smart meters that use the same electricity grid, its wiring, as a physical means of communication. To appreciate the behavior of the ST-PLL algorithm in practical supply systems, the following test has been carried out: two new components have been added to the input signal shown in Fig. 4 (which parameters are in table 2); a dc component with a value of 32.53 V and a 75 Hz inter-harmonic with 15% rms value. Fig. 15a) shows that the ST-PLL algorithm achieves to track the input signal fundamental frequency in a few cycles, from an estimated initial value of 55 Hz. Fig. 15b) shows that the phase of the fundamental component of the input signal is also tracked but, however, the fundamental component is not perfectly sinusoidal, ¿why? Figure 16a) shows the spectrum of the output signal of the ST-PLL algorithm when the input signal is that of Figure 4 without dc component or inter-harmonic. As can be seen, the fundamental component provided by the algorithm is

sinusoidal, that is, it can cancel the harmonic components. However, Fig. 16b), when the input signal also contains dc and inter harmonics, the ST-PLL algorithm still eliminates harmonic components and dc, but does not eliminate inter-harmonic component. In any case, the function as PLL of the algorithm is perfectly satisfied because captures both the fundamental frequency and phase of the input signal.



(a)



(b)

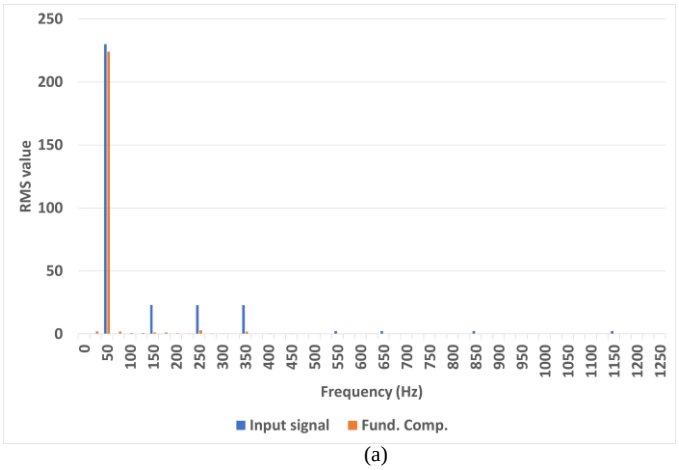
Fig. 15. Results provided by the ST-PLL when applied to a signal with inter-harmonics. (a) Frequency, and (b) fundamental component and phase.

C. *Performance of ST-PLL algorithm at high frequency*

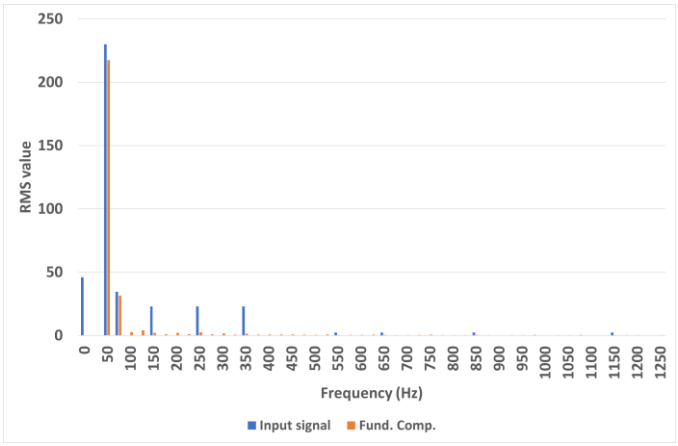
To show the true possibilities of the developed algorithm in high frequency need to be implemented in a more efficient code that the used in this research (MATLAB®). However, as a first approximation to explore the excellent possibilities of the algorithm, the following test will be carried out: the objective is to track an unknown ω_0 = 180000π rad/s (f_0 = 90 kHz), 1 V RMS input signal that jumps down 20 KHz every 0.005 s (Fig. 17). As a test of the algorithm's operating speed, it is assumed that the estimated frequency is 10 times higher than the actual one, i.e., f_c = 900 kHz.

Fig. 17 shows that the ST-PLL algorithm hunts for the unknown frequency right away.

Fig. 18 shows, for the area detailed in Fig. 17, how the ST-PLL algorithm quickly captures the entire input signal in one of the jumps.



(a)



(b)

Fig. 16. Spectra corresponding to the signal provided as fundamental component by the ST-PLL. (a) Input signal contains harmonics but not inter-harmonics. (b) Input signal contains dc component, harmonics and inter-harmonics.

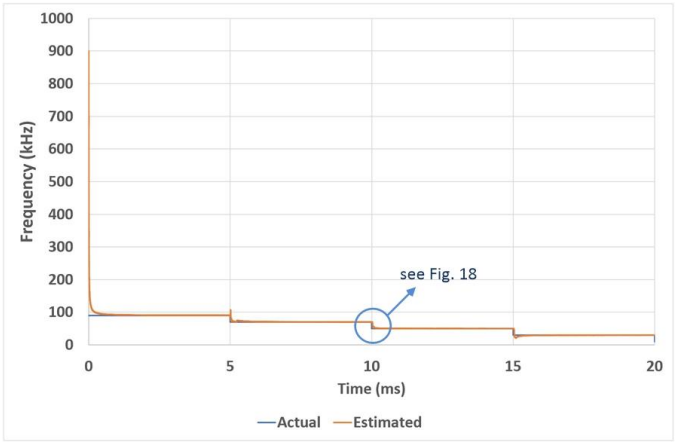


Fig. 17. ST-PLL algorithm tracking speed of an unknown frequency.

IV. Experimental Results

After checking by simulations, the excellent behavior of the ST-PLL algorithm, now, in this section, the results of experimental tests designed and carried out to test its performance in actual situations will be shown. For this, the authors have designed and implemented a workbench with all the necessary functionalities (Figure 19).

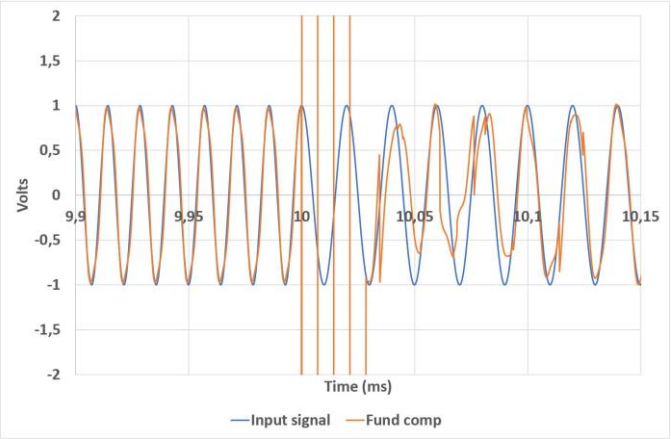


Fig. 18. ST-PLL algorithm tracks quickly the jump indicated in Fig. 16.

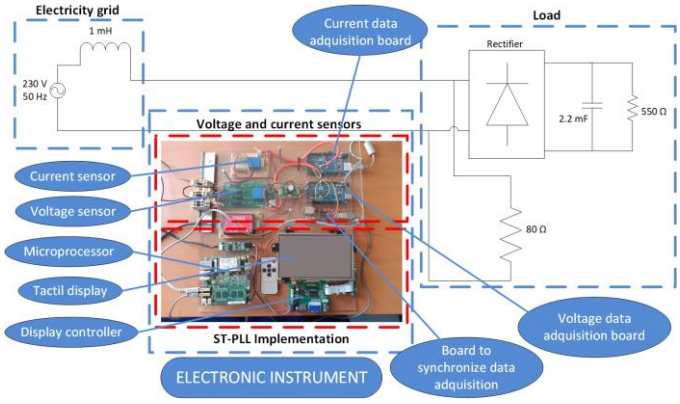


Fig. 19. Experimental workbench to test the ST-PLL algorithm.

In the first place, the developed workbench will be briefly presented, then two tests will be carried out: the first directly on the electricity grid and the second on the current provided by a grid-connected inverter.

In the bottom part of the developed electronic instrument (Figure 19), the ST-PLL is executed implemented in JavaScript using the EJsS application (easy Java/JavaScript simulations), [30], an open tool written in Java (although compatible with the limitations of the new generation of browsers), which overcomes the constraints of proprietary software. EJsS can be used both to carry out simulations and to obtain the data from physical sensors, process, and evaluate them. In this case, this second utility has been used. The upper part of the instrument presented in Fig. 19 contains the measuring devices. Specifically, two LEMTM sensors, [31], one for voltage, and the other for current. They are controlled by means of two Arduino microcontrollers, [32], and synchronized between them by an FPGA (field-programmable gate array), [33].

Now, a first experimental test is described. It is also presented in Fig. 19. In the figure can be seen as the instrument has been connected to the electricity grid by means of a 1 mH inductance that acts as the electricity grid impedance to weaken it. As a load, an uncontrolled rectifier with capacitive impedance on the DC side in parallel to an 80 Ω resistance on the AC side has been used. The capacitive impedance is made up of a 550 Ω resistor in parallel with a 2200 μF capacitor. As a consequence of the mounted circuit, the current at the input of the load is a very distorted signal and, due to inductance, also the voltage signal, but not as much as the current. In addition, current and voltage RMS values are very different.

The developed electronic instrument has been used to track voltage and current frequency and phase as well as to provide their fundamental components. In both cases, the estimated (chosen) frequency (ω_c) has been set at 90π rad/s ($f_c = 45$ Hz) compared to that of the electricity grid, which is $\omega_0 = 100\pi$ rad/s, ($f_0 = 50$ Hz) (see Fig. 3). The sampling time has been set to 0.156 ms.

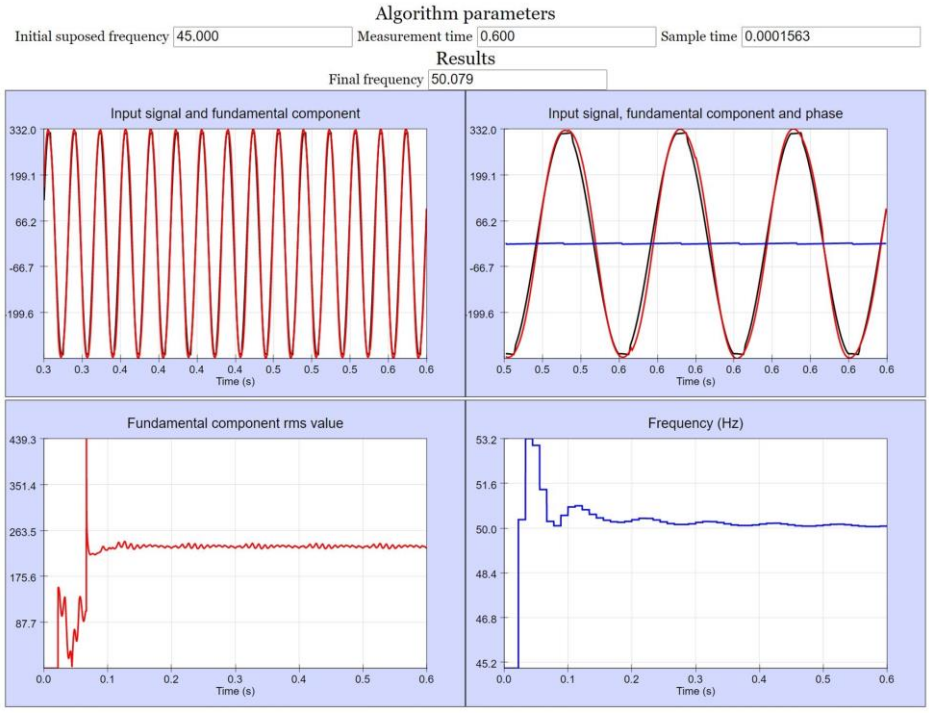


Fig. 20. Results of experimental test: Voltage (V). ST-PLL algorithm parameters: estimated frequency = 45 Hz, measurement time = 0.6 s, sample time = 0.0001563 s. The input signal in black, the fundamental component in red and the phase in blue.

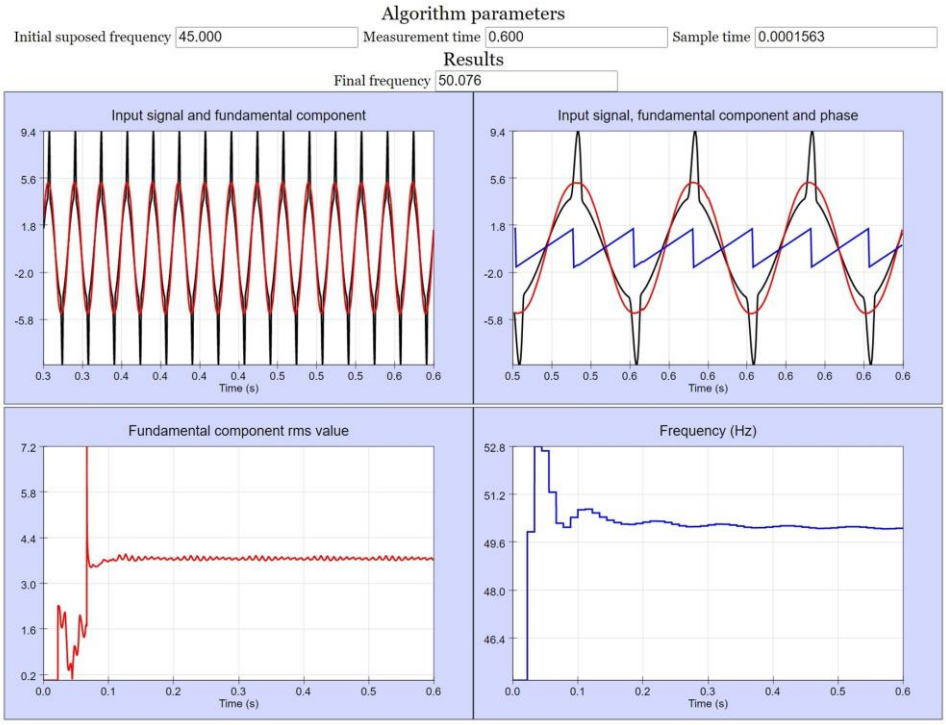


Fig. 21. Results of experimental test: Current (A). ST-PLL algorithm parameters: estimated frequency = 45 Hz, measurement time = 0.6 s, sample time = 0.0001563 s. The input signal in black, the fundamental component in red and the phase in blue.

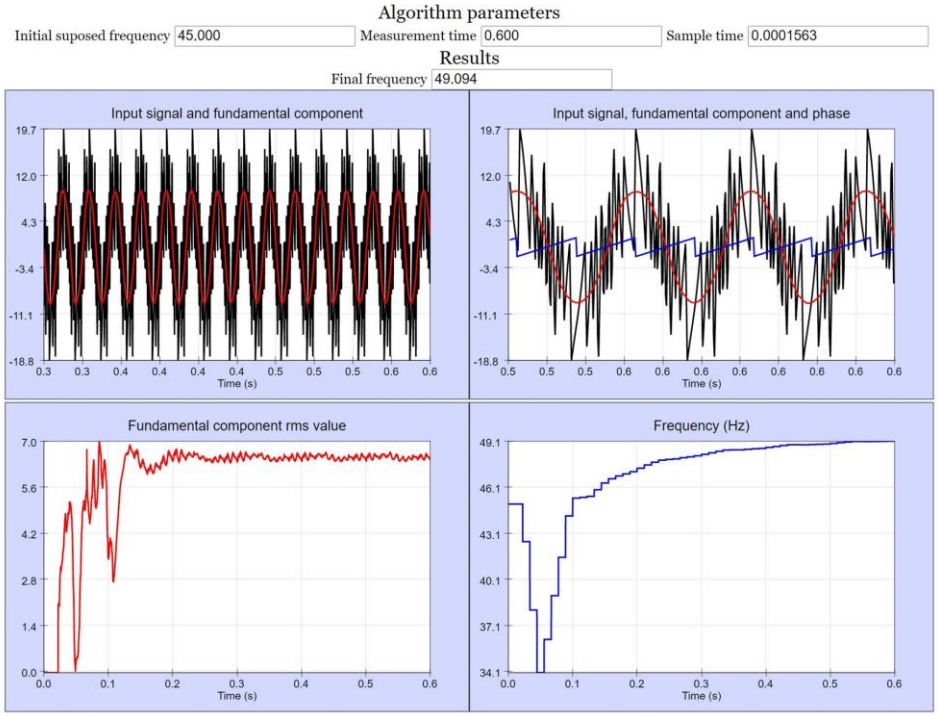


Fig. 22. Results of applying the ST-PLL to the current provided by a grid-connected inverter.

The results of the experimental test are presented in Figs. 20 (voltage) and 21 (current). As can be seen in both figures, the electronic instrument developed, thanks to the built-in ST-PLL algorithm, fixes the frequency and phase of both waveforms and provides their corresponding fundamental components with high precision. In addition, it is easy to make an idea of the speed of the algorithm; thus, if the sampling time is 0.1563 ms and the rms value of the fundamental component is tracked in less than 100 ms, the algorithm has been executed more than 650 times in this short period of time.

Finally, the ST-PLL has been applied to the current provided by a grid-connected inverter (of course, using the same experimental workbench of Figure 19) and the results are presented in Fig. 22. It is a real waveform with high and low

harmonic components. As can be seen in Fig. 22, the ST-PLL algorithm manages to trace the input signal and provides its fundamental component in a few cycles.

CONCLUSIONS

In this article, a new phase-locked loop algorithm has been presented, which, being self-tuning, has been called (ST-PLL). It is not an electronic device, but a software algorithm implementable in a virtual instrument. In fact, the ST-PLL algorithm is much easier to implement in practice than other PLLs published in the literature and requires much less computational effort.

Regardless that its mathematical foundation is very simple, which also leads to a very simple practical implementation, perhaps two of its most important qualities are that it is self-adjusting and also very fast. The first quality allows it to work with a wide frequency range in real-time because no tuning is necessary; the second allows the algorithm to work in a wide range of different applications, from electricity grid (low frequency) to communications (high frequency).

The performance of the ST-PLL algorithm has been tested both by simulation and experimentally. Furthermore, its performance has been compared with those of the most cited in the literature in the field of electricity grids, specifically in single-phase systems. The results have been excellent in terms of speed and, most of all, with regard to a feature that makes it unique, it tunes itself automatically based on the input signal it receives.

The results have also shown that the ST-PLL algorithm can track the input signal after abrupt and simultaneous changes in its frequency, phase, and rms value, even if the input signal is distorted. But even more so, the ST-PLL algorithm eliminates harmonic components and DC offsets and, in the case that voltage dips can occur in the electricity grid, the rms voltage is tracked in just one cycle, and the phase and frequency are not affected.

In order to check the actual performance of the ST-PLL algorithm in practice, an experimental electronic instrument has been developed and implemented. It has been applied to the power grid, weaken by means of a grid impedance. The algorithm performance has been, again, excellent and has confirmed the simulation results.

The experimentation has been carried out on a single-phase electricity grid, but, of course, it is obvious that, for the three-phase grid, a single-phase PLL like the one developed is enough to track the frequency (it will be the same in the other two phases). To generate the three-phase fundamental waveform, the ST-PLL algorithm should be applied to the input signal positive sequence waveform. Computing the three-phase waveform from the fundamental component obtained from the algorithm, the three-phase input signal fundamental positive sequence component would be obtained.

It might be thought that the confirmation of the results would have concluded this investigation. Nothing is further from reality, precisely now, it is pending for further research to explore the true possibilities of the algorithm, which, in the authors' opinion, go far beyond what is shown in this work, especially in the field of communications. For this, it will be necessary to implement the algorithm in a more efficient and faster code, and to carry out specific experimental circuits for communications.

Although the ST-PLL algorithm was conceived and designed to solve synchronization problems in the electricity grid, it has shown possibilities that go much further and that deserve to be explored in future works.

ACKNOWLEDGMENT

This paper is framed in the project “Construction, commissioning and testing of a remotely programmable DC / AC inverter prototype that can be used connected to the grid or to power isolated loads” funded by the Andalucía government, PAIDI 2017, call for technology transfer projects.

REFERENCES

[1] X. Sun, J. Cao, G. Lei, Y. Guo and J. Zhu, “Speed Sensorless Control for Permanent Magnet Synchronous Motors Based on Finite Position Set”, *IEEE Trans. Ind. Electron.*, vol. 67, n. 7, pp. 6089-6100, Jul. 2020.

[2] H. X. Nguyen, T. N.-C. Tran, J. W. Park and J. W. Jeon, “An Adaptive Linear-Neuron-Based Third-Order PLL to Improve the Accuracy of Absolute Magnetic Encoders”, *IEEE Trans. Ind. Electron.,* vol. 66, n. 6, pp. 4639-4649, Jun. 2019.

[3] L. Huang et al., “Grid-Synchronization Stability Analysis and Loop Shaping for PLL-Based Power Converters with Different Reactive Power Control”, *IEEE Trans. Smart Grid*, vol. 11, n. 1, pp. 501-516, Jan. 2020.

[4] H. Wu and X. Wang, “Design-Oriented Transient Stability Analysis of PLL-Synchronized Voltage-Source Converters”, *IEEE Trans. Power Electron.*, vol. 35, n. 4, pp. 3573-3589, Apr. 2020.

[5] Z.-X. Zou and M. Liserre, “Modeling Phase-Locked Loop-Based Synchronization in Grid-Interfaced Converters”, *IEEE Trans. Energy Convers.*, vol. 35, n. 1, pp. 394-404, Mar. 2020.

[6] D. Zhu, S. Zhou, X. Zou and Y. Kang, “Improved Design of PLL Controller for LCL-Type Grid-Connected Converter in Weak Grid”, *IEEE Trans. Power Electron.*, vol. 35, n. 5, pp. 4715-4727, May 2020.

[7] F. Chishti, S. Murshid and B. Singh, “Robust Normalized Mixed-Norm Adaptive Control Scheme for PQ Improvement at PCC of a Remotely Located Wind–Solar PV-BES Microgrid”, *IEEE Trans. Ind. Inform*., vol. 16, n. 3, pp. 1708-1721, Mar. 2020.

[8] M. Chen, L. Peng, B. Wang and W. Wu, “Accurate and Fast Harmonic Detection Based on the Generalized Trigonometric Function Delayed Signal Cancellation”, *IEEE Access*, vol. 7, pp. 3438-3447, Jan. 2019.

[9] L. Bartolomei, A. Mingotti, L. Peretto, R. Tinarelli and P. Rinaldi, “Performance Evaluation and Comparison of a Low-Cost, PLL-Based Acquisition System Under Off-Nominal Conditions”, *IEEE Trans. Instrum. Meas*., vol. 69, n. 5, pp. 2048-2056, May 2020.

[10] S. Golestan, J. M. Guerrero, F. Musavi and J. C. Vasquez, “Single-Phase Frequency-Locked Loops: A Comprehensive Review”, *IEEE Trans. Power Electron.*, vol. 34, n. 12, pp. 11791-11812, Dec. 2019.

[11] S. Golestan, J. M. Guerrero and J. C. Vasquez, “Single-Phase PLLs: A Review of Recent Advances”, *IEEE Trans. Power Electron.*, vol. 32, n. 12, pp. 9013-9030, Dec. 2017.

[12] S. Silwal, S. Taghizadeh, M. Karimi-Ghartemani, M. J. Hossain and M. Davari, “An Enhanced Control System for Single-Phase Inverters Interfaced with Weak and Distorted Grids”, *IEEE Trans. Power Electron.*, vol. 34, n. 12, pp. 12538-12551, Dec. 2019.

[13] H. Ahmed, S. Biricik, and M. Benbouzid, “Enhanced Frequency Adaptive Demodulation Technique for Grid-Connected Converters”, *IEEE Trans. Ind. Electron.*, vol. 68, n. 11, pp. 11053-11062, nov. 2021.

[14] M. Shamim Reza and V. G. Agelidis, “A Demodulation-Based Technique for Robust Estimation of Single-Phase Grid Voltage Fundamental Parameters”, *IEEE Trans. Ind. Inform*., vol. 13, no. 1, pp. 166-175, Feb. 2017.

[15] M. S. Reza, M. M. Hossain, and A. M. Y. M. Ghias, “Open-Loop Approach for Robust Detection of Selective Harmonic in Single-Phase System”, *IEEE Trans. Ind. Inform.*, vol. 15, no. 12, pp. 6260-6269, Dec. 2019.

[16] S. Golestan and J. M. Guerrero, "An Analysis of Modified Demodulation-Based Grid Voltage Parameter Estimator", *IEEE Trans. Power Electron.*, vol. 30, no. 12, pp. 6528-6533, Dec. 2015.

[17] Lamo, F. López, A. Pigazo and F. J. Azcondo, "An Efficient FPGA Implementation of a Quadrature Signal-Generation Subsystem in SRF PLLs in Single-Phase PFCs", *IEEE Trans. Power Electron.*, vol. 32, n. 5, pp. 3959-3969, May 2017.

[18] Y. Han, M. Luo, X. Zhao, J. M. Guerrero and L. Xu, "Comparative Performance Evaluation of Orthogonal-Signal-Generators-Based Single-Phase PLL Algorithms—A Survey", *IEEE Trans. Power Electron.*, vol. 31, n. 5, pp. 3932-3944, May 2016.

[19] S. Golestan, J. M. Guerrero, J. C. Vasquez, A. M. Abusorrah and Y. Al-Turki, "All-Pass-Filter-Based PLL Systems: Linear Modeling, Analysis, and Comparative Evaluation", *IEEE Trans. Power Electron.*, vol. 35, n. 4, pp. 3558-3572, Apr. 2020.

[20] F. Wu, D. Sun, L. Zhang and J. Duan, "Influence of Plugging DC Offset Estimation Integrator in Single-Phase EPLL and Alternative Scheme to Eliminate Effect of Input DC Offset and Harmonics", *IEEE Trans. Ind. Electron.*, vol. 62, n. 8, pp. 4823-4831, Aug. 2015.

[21] K. Al Hosani, T. H. Nguyen and N. Al Sayari, "An improved control strategy of 3P4W DVR systems under unbalanced and distorted voltage conditions", *Int. J. Electr. Power Energy Syst.*, vol. 98, pp. 233-242, Jun. 2018.

[22] H. K. Yada and M. S. R. Murthy, "An improved control algorithm for DSTATCOM based on single-phase SOGI-PLL under varying load conditions and adverse grid conditions", *IEEE International Conference on Power Electronics, Drives and Energy Systems (PEDES)*, Dec. 2016.

[23] M. Xie, H. Wen, C. Zhu and Y. Yang, "DC Offset Rejection Improvement in Single-Phase SOGI-PLL Algorithms: Methods Review and Experimental Evaluation", *IEEE Access*, vol. 5, pp. 12810-12819, Jun. 2017.

[24] H. Yi, X. Wang, F. Blaabjerg and F. Zhuo, "Impedance Analysis of SOGI-FLL-Based Grid Synchronization", *IEEE Trans. Power Electron.*, vol. 32, n. 10, pp. 7409-7413, Oct. 2017.

[25] K. R. Patil and H. H. Patel, "Modified dual second-order generalised integrator FLL for synchronization of a distributed generator to a weak grid", *IEEE 16th International Conference on Environment and Electrical Engineering (EEEIC)*, Jun. 2016.

[26] H. Pan, Z. Li and T. Wei, "A Novel Phase-Locked Loop with Improved-Dual Adaptive Notch Filter and Multi-Variable Filter", *IEEE Access*, vol. 7, pp. 176578-176586, Dec. 2019.

[27] H. Ahmed, S.-A. Amamra and M. Bierhoff, "Frequency-Locked Loop-Based Estimation of Single-Phase Grid Voltage Parameters", *IEEE Trans. Ind. Electron.*, vol. 66, n. 11, pp. 8856-8859, Nov. 2019.

[28] J. Hazarika and P. Sumathi, "Moving Window Filter Based Frequency-Locked Loop for Capacitance Measurement", *IEEE Trans. Ind. Electron.*, vol. 62, n. 12, pp. 7821-7823, Dec. 2015.

[29] M. V. Blagov et al., "Hold-in, Pull-in and Lock-in Ranges for Phase-locked Loop with Tangential Characteristic of the Phase Detector", *Procedia Comput. Sci.*, vol. 150, 2019, pp. 558-566.

[30] "Easy Java Simulations Wiki | Main / EJS Home Page" [Online], Available: https://www.um.es/fem/EjsWiki/?userlang=es.

[31] "LEM – current transducer, voltage transducer, sensor, power measurement" [Online], Available: https://www.lem.com/en.

[32] "Arduino – Home" [Online], Available: https://www.arduino.cc/.

[33] "What is an FPGA? Programming and FPGA Basics - INTEL® FPGAS" [Online], Available: https://www.intel.com/content/www/es/es/products/programmable/fpga/new-to-fpgas/resource-center/overview.html.

Reyes Sánchez-Herrera was born in Huelva, southwest of Spain. She received the Industrial Engineering degree in 1995 and Ph. D. in Electrical Engineering with award in 2007. Currently, she is professor with the department of Electrical Engineering and Director of Research Transfer at the University of Huelva. She has 5 patents. She has 36 publications, 20 of them published in indexed journals in the ISI Journal Citation Reports. Particularly, she has 12 publications included in the first/second quartile. She has worked on 14 research projects funded by public institutions and companies, leading 5 of them. Her research includes electrical power quality, power conversion systems and renewable energy systems.

José M. Andújar became a Member of IEEE in 2010, and Senior Member in 2012. He received his Ph.D. degree in 2000, at the University of Huelva, Spain. He is currently a Full Professor of systems engineering and automatic control with the University of Huelva. Throughout his professional life, he has received 28 awards and academic honors. He has supervised 14 Doctoral Theses with 9 awards. He has 17 international patents. He has more than 420 publications, among them more than 125 articles published in indexed journals in the ISI Journal Citation Reports. Particularly, he has 63 publications included in the first quartile in 26 different journals; most of these are among the top ten in their categories, and several are number one. He has led or co-led 71 research projects funded by public institutions and companies. His main research interests are control engineering, renewable energy systems, and precision agriculture.

Marco Márquez received the industrial engineering Phd from the University of Huelva in 2015. He was an Associate Professor with the Department of Electronic at the University of Huelva, from 1994 to 2002, and is now a Full Professor with the Department of Electronics, Pedro Gómez Institute. His research interests include new e-learning technologies and the communications aspects in remote labs.

Andrés Mejías received the Industrial Engineering degree, the Dipl. Computing degree, and the Master's degree in production systems engineering from the University of Seville, Seville, Spain, in 1989, 1993, and 2005, respectively. And the Ph.D. degree in industrial engineering from the University of Huelva, Huelva, Spain, in 2012. He received the award for the best doctoral thesis of the IEEE Education Society in 2013 (Spanish chapter). He is currently a Full Professor with the Department of Electronic Engineering at the University of Huelva. His research interests include new e-learning technologies, mainly those related to remote labs.

Gabriel Gomez Ruiz is research assistant in University of Huelva (Spain). Currently he works in the project "Construction, commissioning and testing of a remotely programable DC / AC inverter prototype that can be used connected to the grid or to power isolated loads." In 2020 he obtained Industrial Electronic Engineering degree and now he's studying a Master's Degree in Industrial Engineering, both in University of Huelva.