Naji Rajai Nasri AMA[1], Wilson KOMATSU[1], Fuad KASSAB JUNIOR[1], Lourenco MATAKAS JUNIOR[1]

Polytechnic School of the University of Sao Paulo (1)

Adaptive single phase moving average filter PLLs: analysis, design, performance evaluation and comparison

Abstract. *This work analyzes the single-phase Phase Locked Loops using adaptive Moving Average filter and proposes a discrete time tuning procedure, resulting in faster transient response when compared with previous methods, with a good tradeoff between transient (settling time) and steady state (phase ripple) characteristics. An extensive experimental performance evaluation and comparison between the proposed PLL, pPLL and DFAC-pPLL is carried out showing that the proposed PLL presents similar or better performance than more sophisticated ones like DFAC-pPLL.*

Streszczenie. *W artykule opisano jednofazowy układ PLL z filtrem adaptacyjnym i zaproponowano procedurę strojenia zapewniającą krótszy stan nieustalony i dobre przejście między stanem przejściowym a ustalonym. Porównano eksperymentalnie różne układy PLL, pPLL i DFAC-pPLL.* ***Układ PLL z filtrem adaptacyjnym o ruchomej średniej – analiza, projektowanie, właściwości***

Keywords: PLL; Synchronization; Adaptive moving average filter; distributed generation.
Słowa kluczowe: układ PLL, ruchoma średnia, filtr adaptacyjny.

doi:10.12915/pe.2014.05.43

I. Introduction

Our proposal involves single-phase Phase Locked Loops (PLL) using adaptive Moving Average (MAV) filters, suitable for harmonics distorted and frequency changing AC systems such as micro-grids including low power renewable energy source, such as solar and wind. It is shown that simple PLL structures, as those proposed in this paper, can present similar or better performance (fast transient response and low output signal phase ripple) than more sophisticated ones found in the literature. To demonstrate this claim, an experimental comparison of some single-phase PLL strategies is made, and especially, the proposed MAMAV-PLL (Modified Adaptive MAV-filter based PLL) is compared with the recently published DFAC-pPLL [1] based on the pPLL [2]. The MAMAV-PLL derives from conventional adaptive moving average filter PLL (AMAV-PLL) [1] [3-5], which is based on the non-adaptive moving average filter PLL (MAV-PLL) [3-6]. This paper shows that pPLL, and MAV filter-based PLLs (MAV-PLL, AMAV-PLL and MAMAV-PLL) have the same structure as the well known classical multiplier-type PLL structure [3-11], and differs only in the implementation of the phase detector and its associated low pass filter. The linearized model of the classical multiplier-type PLL structure is the same for all PLLs discussed herein.

The controller tuning employed for the aforementioned PLLs is reviewed and a discrete time controller tuning procedure is proposed for the MAV filter-based PLLs.

The relative performance between PLLs is obtained with experimental results comparison. In order to obtain accurate measurements, a software and hardware-based test bench is presented and applied.

The paper is organized as follows. Section II briefly reviews the classical multiplier-type PLL. The features of the pPLL, its improved version DFAC-pPLL, the MAV-PLL and the adaptive AMAV-PLL are highlighted. To improve the AMAV-PLL performance during input variation, an amplitude estimator was added and the resulting structure is called modified adaptive MAV-PLL (MAMAV-PLL). Section III briefly describes the tuning of the pPLL and the DFAC-pPLL used in their original papers, and proposes a discrete tuning strategy for the MAV filter-based PLLs. Section IV compares and discusses the experimental results for all analyzed PLLs.

[1] In this work, a conventional MAV filter uses a fixed window with N samples and an adaptive MAV changes the window size according to the mains frequency.

II. PLL comparison

The basic PLL structure is shown in Fig. 1 [7].

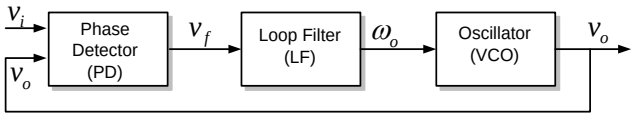

Fig. 1. Basic PLL structure [7].

All the PLLs studied in this paper share the same basic PLL structure, composed of a phase detector (PD), a loop filter (LF) and a voltage controlled oscillator (VCO). The PD output v_f depends on the displacement angle between input and output signals v_i and v_o. The LF, also known as the controller (Fig. 2), is responsible for imposing the desired PLL dynamic and steady state performance. The VCO generates an output signal v_o with frequency ω_o. In steady state, the PLL output signal v_o is synchronized in phase and frequency with the fundamental component of the reference signal v_i.

A. Classical multiplier-type PLL implementation

The classical multiplier-type implementation of the basic PLL structure of Fig. 1 is shown in Fig. 2.

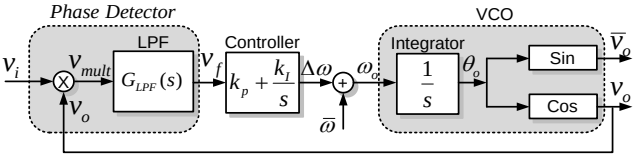

Fig. 2.Classical multiplier-type PLL implementation [5-6].

Consider harmonics distorted input voltage v_i to be represented by:

$$v_i = A_1 \sin(\omega_1 t + \phi_1) + A_3 \sin(3\omega_1 t + \phi_3) + + A_5 \sin(5\omega_1 t + \phi_5) + ..., \quad (1)$$

where A_1 is the fundamental component peak value, ω_1 the fundamental component frequency, $\bar{\omega}$ is the grid nominal frequency, and ϕ_1 is the fundamental phase angle.

In steady state, the PLL output signal $\bar{v}_o$ is in phase and v_o is in quadrature with the fundamental component of the input signal v_i. The output signal v_o is given by:

$$v_o = \cos(\omega_o t + \phi_o) = \cos(\theta_o). \quad (2)$$

Using (1), (2) and by considering that $\omega_o=\omega_1$, the output of the PD is given by:

(3)
$$v_{mult}=\frac{1}{2}\begin{bmatrix}A_1\sin(\phi_1-\phi_o)+A_1\sin(2\omega_1 t+\phi_1+\phi_o)+\\ A_3\sin(2\omega_1 t+\phi_3-\phi_o)+A_3\sin(4\omega_1 t+\phi_3+\phi_o)+\\ A_5\sin(4\omega_1 t+\phi_5-\phi_o)+A_5\sin(6\omega_1 t+\phi_5+\phi_o)+...\end{bmatrix}$$

The oscillating terms of v_{mult} are called n according to (4).

(4)
$$n=\frac{1}{2}\begin{bmatrix}A_1\sin(2\omega_1 t+\phi_1+\phi_o)+\\ A_3\sin(2\omega_1 t+\phi_3-\phi_o)+A_3\sin(4\omega_1 t+\phi_3+\phi_o)+\\ A_5\sin(4\omega_1 t+\phi_5-\phi_o)+A_5\sin(6\omega_1 t+\phi_5+\phi_o)+...\end{bmatrix}$$

Equation (3) can be rewritten as:

(5) $$v_{mult}=0.5A_1\sin(\phi_d)+n,$$

where:

(6) $$\phi_d=\phi_1-\phi_o.$$

Using (5) and Fig. 2, the nonlinear mathematically equivalent model of the PLL is constructed as shown in Fig. 3 where:

(7) $$\theta_i=\omega_1 t+\phi_1.$$

Its linearized model for small values of ϕ_d is presented in Fig. 4.

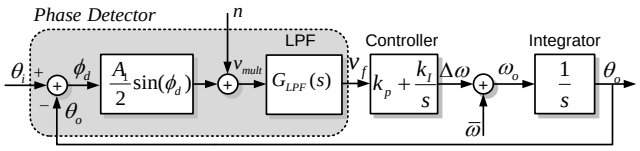

Fig. 3.Nonlinear model for the single-phase PLL.

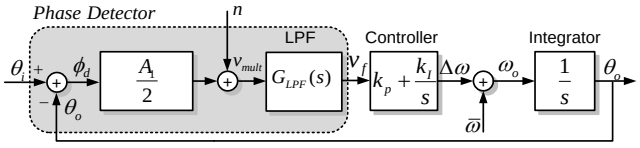

Fig. 4.Linearized model for the single-phase PLL.

The interpretation of the functionality of the classical multiplier-type PLL implementation (Fig. 2) does not depend on the instantaneous power theory as stated by [2,12-13] but only on the mathematical equations discussed above.

For the multiplier-type PLL (Fig. 2) any kind of low pass filter (LPF) can be used, such as Butterworth LPF, MAV LPF, etc.

B. pPLL

The three phase pPLL was originally proposed by [13], and is explained by using the instantaneous power theory approach. Its single phase version, proposed by [2], is presented in Fig. 5 and its linearized model in Fig 6.

In the pPLL, the input signal v_i is considered as a fictitious voltage signal and the output signal i (output of the Sin block of Fig. 5) is considered as a fictitious current. The product $p=v_i\ i$ in Fig. 5 is called fictitious instantaneous power. According to [2], if the mean value of the fictitious power p is zero, the pPLL is in lock condition ($\theta_i=\theta_o$).

By comparing the linearized models (Figs.4 and 6), they can be observed to be identical, as their PDs have the same transfer function, according to (8). For better understanding, this paper explicitly includes the effect of noise n (Fig. 4).

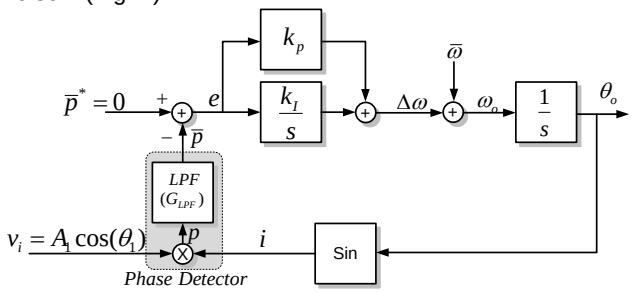

Fig. 5.Original pPLL structure [1-2].

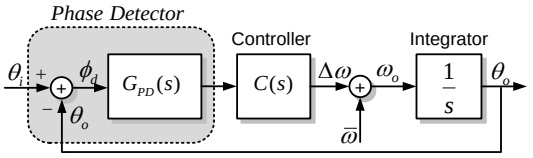

Fig. 6.Linearized model of the original pPLL structure [1-2].

(8) $$G_{PD}(s)=\frac{A_1}{2}G_{LPF}(s).$$

C. DFAC-pPLL

According to sections II.A and II.B, the output of both, the multiplier-type and the pPLL type, contain even harmonics (2nd, 4th, 6th, etc.) if v_i contains odd harmonics. For proper PLL operation, these harmonics should be properly attenuated. In the pPLL proposed by [2], the attenuation is achieved by a 4th order Butterworth LPF.

By noticing that the major component of n that should be attenuated is the 2nd harmonic generated by the PD, [1] proposes the use of the Double Frequency cancelation strategy, as shown in Fig. 7 and in Fig. 8. This strategy achieves the double frequency harmonic cancelation generated by the PD by subtracting the double frequency signals $\bar{v}_d,\bar{v}_q$ from v_d, v_q as shown in Fig. 8.

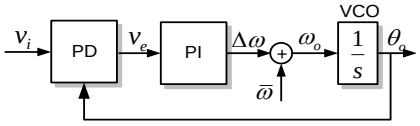

Fig. 7.DFAC-pPLL structure [1] The PD block is detailed in Fig. 8

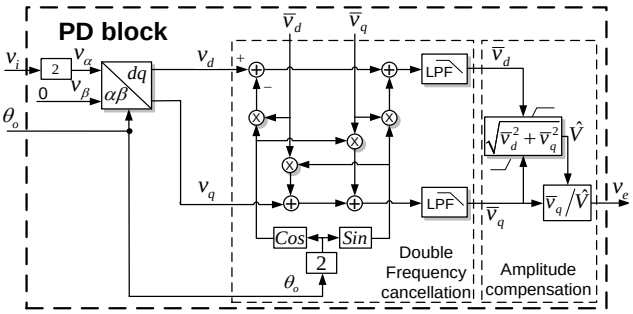

Fig. 8. DFAC-pPLL phase detector (PD) structure [1].

In steady state conditions, the output of the first block of the PD ($\bar{v}_q$) does not contain the second harmonic of the multiplier type PD, but is proportional to the input voltage fundamental component A_1.

To compensate the dependency of the PLL open loop gain on the input voltage amplitude, [1] proposes dividing the $\bar{v}_q$ signal by the estimated amplitude of the fundamental component ($\hat{V}=A_1$), as shown in Fig. 8.

D. PLLs based on moving average filters

1) Non-adaptive MAV-PLL structure and modeling

Fig. 9 shows discrete time block diagram for digital implementation of a single phase PLL, using a multiplier type PD and a MAV filter [5], based on the continuous time implementation of Fig. 2.

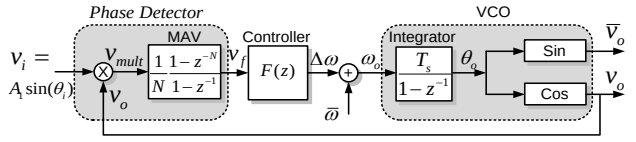

Fig. 9.Single phase PLL with non-adaptive MAV filter (MAV-PLL).

Discrete time modelling and implementation was used because the MAV filter has a linear time invariant transfer function in the discrete domain, but in the continuous time domain only approximate transfer functions are available. Based on the procedure described in section II.A, a discrete time linearized model is developed and presented in Fig. 10.

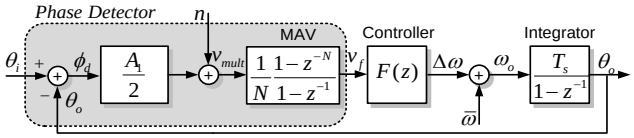

Fig. 10. Discrete time linearized model for the single phase PLL.

The transfer function of a N^{th} order MAV filter is given by [6]:

$$MAV(z) = \frac{1}{N}\frac{1-z^{-N}}{1-z^{-1}}. \tag{9}$$

This filter computes the arithmetic average of a moving window composed by the last N samples. An ideal N^{th} order MAV filter eliminates all the frequencies that are integer multiples of f_n (where f_n is the first MAV notch frequency) as shown in Fig. 11. For a sampling frequency f_s, N is given by:

$$N = f_s / f_n . \tag{10}$$

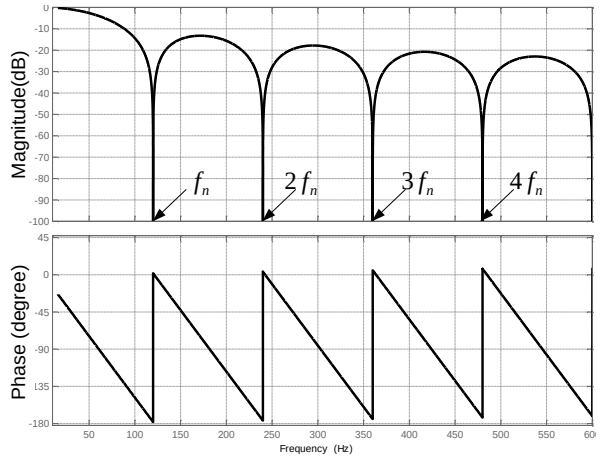

Fig. 11. Bode plot of the MAV filter $N = 100$, $f_n = 120\text{Hz}$, $f_s = 12\text{kHz}$

The MAV filter should be designed to appropriately attenuate all the oscillating terms of v_{mult}. Thus, if n contains only even harmonics[2], f_n must satisfy

$$f_n = 2 f_1 , \tag{11}$$

and (10) can be rewritten as:

$$N = f_s / 2 f_1 . \tag{12}$$

If ω_1 is not equal to the nominal frequency $\bar{\omega}$, MAV filter attenuation performance will be degraded as shown in Fig. 11. In this case, the even harmonics of v_{mult} will not be cancelled and leads to: i) distorted PLL output signal v_o and ii) phase and frequency ripple. This motivates the use of adaptive moving average filter, presented in the next section.

2) Adaptive MAV PLL (AMAV-PLL)

For real applications, the grid frequency f_1 can vary in a limited range $f_1 \pm \Delta f$ defined by the corresponding standards [14-17]. To achieve the appropriate attenuation of the signal n for v_i with variable frequency, two conditions must be provided:

1) Equation (12) should be satisfied;

2) N should be an integer value.

There are two possible solutions: i) the online variation of window size N with fixed sampling frequency f_s and ii) the variation of the sampling frequency f_s with constant window size N.

a) Online variation of the MAV filter window size N.

This solution was originally proposed by [3-4,12]. The discrete time AMAV-PLL (Fig. 12) is implemented by adding a N calculating block in the MAV-PLL (Fig. 11) [5]. The window size N, calculated by (12), is modified to (13), assuring that N is an integer number, followed by a limiter block to restrict the variation range of N. The function 'ROUND' in (13) rounds its argument to the nearest integer.

$$N = \text{ROUND}\left(f_s / 2 f_1 \right) \tag{13}$$

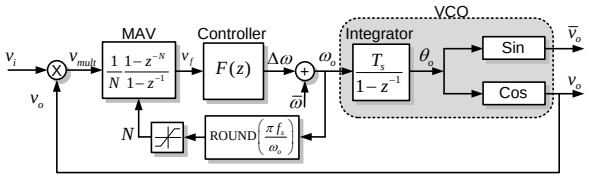

Fig. 12. AMAV-PLL structure [5].

The following analysis shows AMAV filter assured attenuation, and the effect of the choice of f_s (e.g. 6, 12 and 24k*Hz*). As N should be integer, the frequency f_n of the first notch of the MAV filter can only achieve discrete values defined by f_s / N. Therefore, variation of f_n implies in the real time change of N. An example to demonstrate this effect is shown in Fig.13, where f_s=12k*Hz*, nominal f_n=120*Hz*, resulting in nominal N=100.

In Fig. 13, N is varied from 98 to 102, and the individual MAV filtering characteristics are shown in Fig 13(a). According to Fig. 13(a) and to its detailed version Fig. 13(b), the attenuation of this adaptive filter in the region 2 ($f_1 \pm \Delta f$) is always higher than 46dB (assured attenuation).

Table 1 presents the values of the assured attenuation for three sampling frequencies f_s (6, 12 and 24k*Hz*), MAV window size N and the first MAV notch frequency f_n. It also shows the relationship between f_s, N, and Δf_n (f_n varies in steps of size Δf_n as a function of N according to (10)) for the three values of f_s.

[2] If n contains odd harmonics, $f_n = f_1$

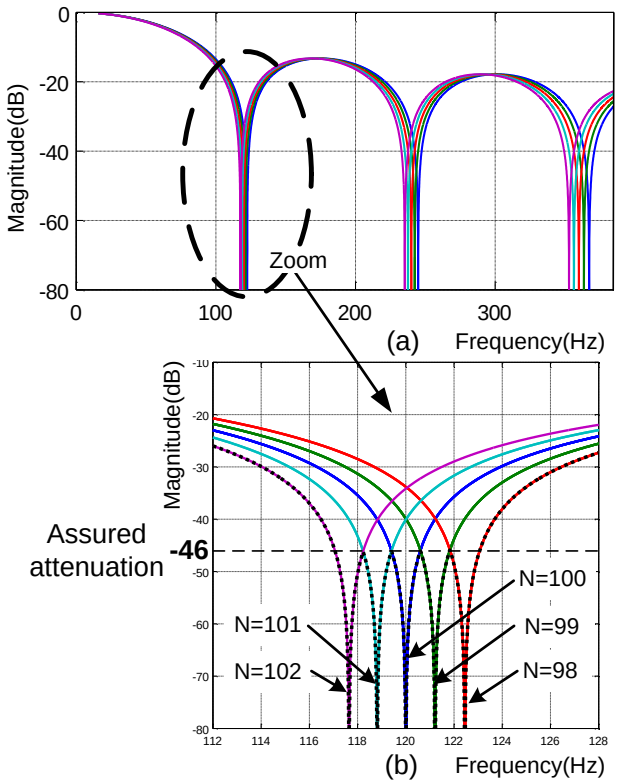

Fig. 13. (a): individual MAV filter frequency response for 98≤N≤102; (b): detailed version around $2f_1$ Continuous lines are related to individual MAV filters, and dotted line to the adaptive filter.

Table 1. Effect of the choice of sampling frequency f_s

$f_s = 6KHz$			$f_s = 12KHz$			$f_s = 24KHz$		
N	f_n	Δf_n	N	f_n	Δf_n	N	f_n	Δf_n
50	120	0	100	120	0	200	120	0
51	117.65	2.35	101	118.81	1.19	201	119.40	0.60
52	115.39	2.26	102	117.65	1.165	202	118.81	0.59
Assured attenuation: 40dB			Assured attenuation: 46dB			Assured attenuation :52dB		

The increase of f_s yields the following tradeoffs:

- f_n presents higher resolution;
- Higher attenuation is obtained;
- Larger memory size is needed to implement the MAV filter.

b) Online variation of the sampling frequency f_s with constant window size N.

The second AMAV solution is to keep constant window N and to vary the sampling frequency f_s according to (12) [18]. This solution could present better attenuation, at the expense of requiring real time:

i) recalculation of the parameters of the filters and controller or robust design of filters and controller so as to not degrade overall system performance and stability margins with varying f_s,

ii) change of the converter sampling and switching frequency.

This method is out of this papers scope. Its trade-offs are further discussed in [5].

3) Modified Adaptive MAV Filter PLL (MAMAV-PLL)

Both MAV and AMAV PLLs performance depends on the amplitude of the fundamental component of v_i. Lower amplitudes (e.g. during voltage sags) reduce the open loop gain, resulting in slower transient response. Amplitude normalization can be used to improve transient response for this situation [1]. This work proposes an amplitude normalization strategy where the amplitude estimator shown in Fig. 14 is based on the fundamental amplitude estimator, part of the Fundamental Wave Detector (FWD) proposed by [4].

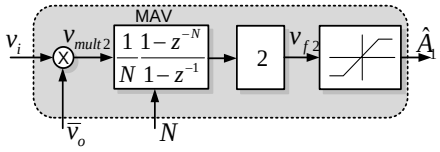

Fig. 14. Amplitude estimator.

Using v_i defined by (1) and defining $\bar{v}_o$ as

(13) $$\bar{v}_o = \sin(\omega_o t + \phi_o),$$

results in v_{mult2}:

(14)
$$v_{mult2} = \frac{1}{2}\left[\begin{array}{l} A_1\cos(\phi_1-\phi_o) - A_1\cos(2\omega_1 t+\phi_1+\phi_o)+ \\ A_3\cos(2\omega_1 t+\phi_3-\phi_o) - A_3\cos(4\omega_1 t+\phi_3+\phi_o)+ \\ A_5\cos(4\omega_1 t+\phi_5-\phi_o) - A_5\cos(6\omega_1 t+\phi_5+\phi_o)+\ldots \end{array}\right]$$

Assuming that the AMAV filter attenuates the even harmonics of v_{mult2}, result in:

(15) $$v_{f2} = A_1\cos(\phi_1-\phi_o).$$

And assuming the PLL is in lock condition, i.e. $(\phi_1=\phi_o)$,

(16) $$v_{f2} = A_1.$$

The limiter is added to the amplitude estimator of Fig. 14 (e.g. in the range of 0.1 to 1.5) to avoid abnormal values of v_{f2} that may happen during transient operation in unlocked condition.

By adding the amplitude estimator of Fig. 14 to the AMAV-PLL structure of Fig. 12, the proposed MAMAV-PLL is obtained in Fig. 15.

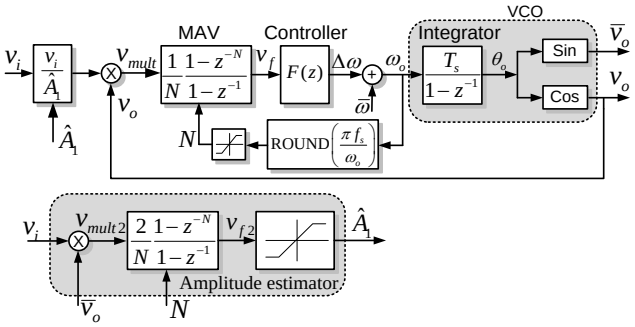

Fig. 15. MAMAV-PLL structure.

III. Controller Tuning

A fair PLL performance comparison between the discussed PLLs requires the use of the PLL controller parameters employed by the original references for the pPLL and the DFAC-pPLL, as it is supposed that the original authors applied the best tuning.

This paper proposes a discrete time tuning procedure for the MAV filter based PLLs. The MAV based PLLs tuned using this procedure present transient performance similar to other PLLs with more sophisticated structures (e.g. DFAC-pPLL) and with better steady state performance.

A. pPLL controller design review

The original pPLL project tuning [2] consists of finding the PI controller parameters and the low pass filter (type,

order and cutoff frequency) based on a trial-and-error approach to satisfy the project specifications of: fundamental frequency and 2nd harmonic attenuation of -58db, phase margin of 34° and open loop crossover frequency of 10*Hz*, for a grid voltage amplitude of 0.8pu. This resulted in k_p=160, k_I=*3600* and a 4th order Butterworth LPF[3]. These parameters led to good steady state performance at the cost of slow transient response.

B. DFAC-pPLL controller design review

The DFAC-pPLL of Fig. 7 and 8 [1] uses a PI controller and two first order LPF. The controller tuning is based on the symmetrical optimum method to ensure maximum phase margin. The original paper controller parameter are k_P=155.26, k_I=10044, ω_c=24.71*Hz*, ω_p=59.3 *Hz*.

C. Previous MAV based PLLs controller design review

Previous PI controller tuning methods for MAV filter based PLLs can be found in [3-4] and later in [6].

References [3-4] use AMAV filter and PI controller. The AMAV filter transfer function is approximated to a unity gain and the closed loop transfer function is reduced to the second order system canonical form. Therefore, the PI parameters are determined in terms of closed loop crossover frequency and damping factor. The AMAV filter approximation leads to unsatisfactory results when a fast transient response is required (e.g., 3-cycle settling time for phase jump transient).

Reference [6] controller design is based on the inspection of the open loop transfer function Bode plot, where: i) the stability conditions (positive gain margin *GM* and phase margin *PM*) are met, ii) the damping factor ζ is calculated using $\zeta=PM/100$ for $PM < 70$, iii) the 2% settling time is estimated by $t_s=4/\zeta\omega_n \approx 4/\zeta\omega_c$ This design results in settling time figures of less than 3 cycles for 45° phase jump, and more than 20 cycles for 2*Hz* frequency jump.

Due to these slow transient results, references [3-4,6] controller designs are not used in this paper.

D. Proposed MAV based PLLs controller design

The controller design proposed herein is using a discrete PI controller of the form:

$$F(z) = K\frac{z-\alpha}{z-1}, \tag{17}$$

which consists from a DC gain (K) and a real zero located at (α). This form can be written in the common PI controller form as:

$$F(z) = k_P + \frac{k_I}{1-z^{-1}}. \tag{18}$$

where $k_p=\alpha K$ and $k_I=K(1-\alpha)$.

Initially, the controller parameters are set to $K=1$ and α=-1 to be able to ignore the effect of the controller zero (non-dominant zero). The resulting Bode plot of the PLL open loop transfer function $G_{OL}(z)$ is shown in Fig. 16. The system is unstable due to the negative phase margin of *PM*=-25.2°. The next step is to readjust the controller zero α and the DC gain K to: i) obtain positive phase and gain margins that guarantee the PLL stability ii) good transient response, this step may need several iterations before a satisfying results can be obtained. A good initial value for placing the controller zero is slightly higher than ω_c obtained from Fig. 16.

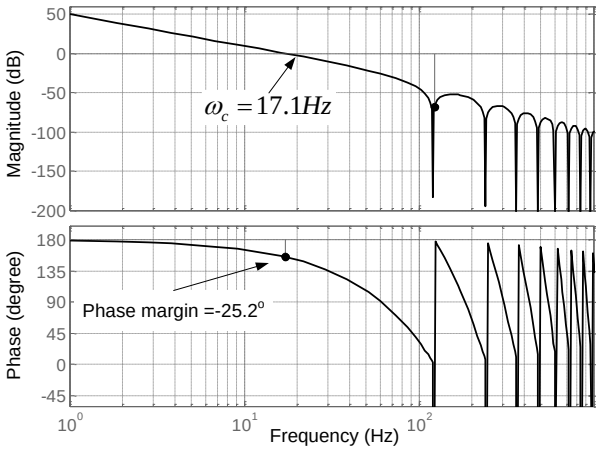

Fig. 16. Bode plot of the proposed PLL open loop transfer function with $K=1$ and $\alpha=-1$.

Using the proposed graphical tuning method, the controller parameters α=0.9956, K=319.18 was obtained. The resulting tuned MAV based PLL has faster transient response when compared with all previous methods [3-4,6].

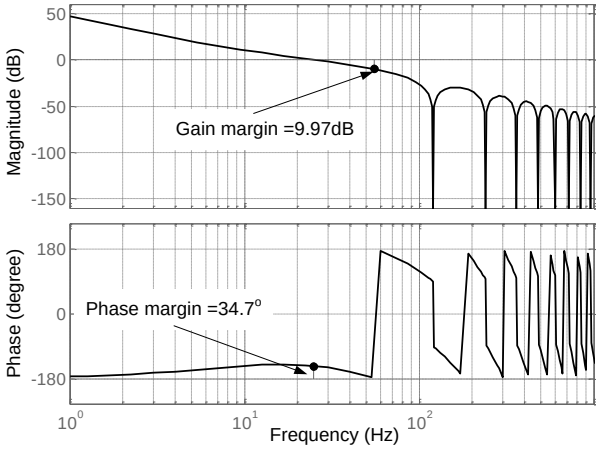

Fig. 17. Bode plot of the proposed PLL open loop transfer function with the final controller parameters K=319.18 and α=0.9956.

IV. Experimental Results

A. Experimental Hardware Test Bench

Fig. 18 presents the block diagram of the experimental hardware test bench. Two acquisition boards NI PCI 6221 were used within a personal computer running MATLAB Simulink with Real Time Windows Target.

The software for the experimental setup consists of three independent blocks implemented in the Simulink environment: i) signal generator block: synthesizes typical test signals (e.g. phase and frequency jump, voltage sag and harmonics) and the fundamental component angle θ_i, ii) PLL block: contains the PLL under test, iii) supervisor block: performs real time calculation of the phase error ϕ_d between the generator angle θ_i and the PLL angle θ_o. The generator angle θ_i is used only by the supervisor to verify the tracking performance of the PLL under test. All the signals of Fig. 18 are available in the Simulink environment and up to four analog signals can be made available for measurement using oscilloscope.

To consider the effect of the analog to digital converter (ADC) performance on the PLL under test, the signal generator output is converted into an analog signal using a 16 bits digital to analog converters (DACs), and fed into a 16-bit ADC as the input v_i to the PLL.

[3] The LPF cutoff frequency (f_{cutoff}), not provided in the original paper, was found here by a trial and error process, resulting in f_{cutoff}=43.2*Hz*.

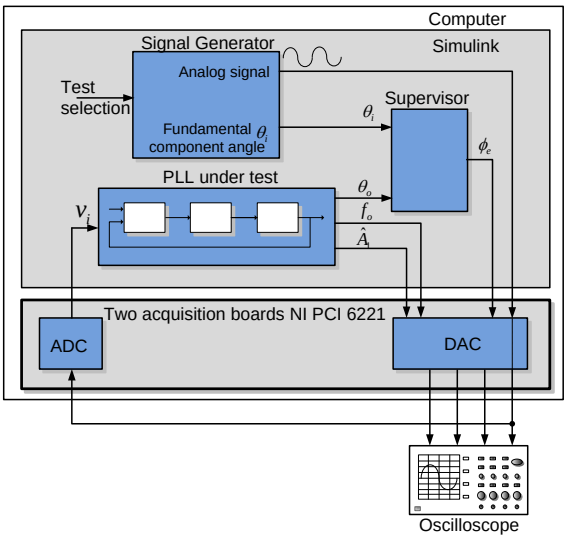

Fig. 18. Block diagram of the experimental hardware test bench.

B. Transient tests

This section presents the experimental results for transient tests including phase and frequency jump and voltage sag. Only the waveforms for the DFAC-pPLL and MAMAV-PLL are provided because they present the best results amongst the analyzed PLLs. The resulting experimental figures were extracted from the Simulink environment.

Fig 19 presents the experimental waveforms for a 40° phase jump occurring at positive zero crossing of the input signal v_i, for the MAMAV-PLL and for the DFAC-pPLL.

Fig 20 presents experimental waveforms for a 5*Hz* frequency jump of the input signal v_i.

Fig 21 presents experimental waveforms for a 30% voltage sag occurring at the positive peak of the input signal v_i.

At first glance, figures 19, 20 and 21 show good results for both PLLs, but these waveforms do not provide precise measurement of the PLLs dynamic response parameters (settling time, overshoot and peak phase error)[4]. Thus, to achieve the best accuracy, all the PLL signals were stored in real time in the MATLAB environment and offline processed to calculate these three parameters. Table 2 presents the settling time, overshoot and peak phase error of the five PLLs analyzed in this paper. Values from original papers (column a) are compared with values obtained from experimental results using the measured internal signals (column b). The settling time criterion is defined for each test: i) 40° phase jump test: the elapsed time (measured in cycles) necessary for the phase angle ϕ_o to enter and to remain in the ±2% band around 40°; ii) 5*Hz* frequency jump test: the elapsed time (measured in cycles) necessary for the PLL frequency f_o to enter and remain in the ±2% band around 65*Hz*; iii) voltage sag test: the elapsed time (measured in cycles) necessary for the phase angle ϕ_o to enter and to remain in the 0.3° band.

The experimental results measured in this paper (column b of Table 2) are close to the ones obtained in the original papers (column a of Table 2) except for the pPLL, where some values are slightly different, but better than the original ones. This can be explained by the different implementation platform, parameter measurement method and absence of the original LPF cutoff frequency which was estimated in section III.A.

[4] This accuracy problem also appears when using an oscilloscope.

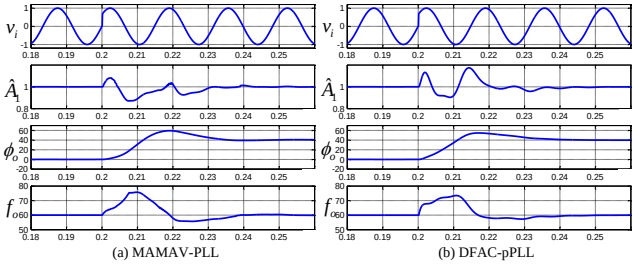

Fig. 19. Experimental waveforms for 40° phase jump. Input signal v_i with 40° phase jump, estimated amplitude $\hat{A}_1$, PLL output phase angle ϕ_o and PLL frequency f_o.

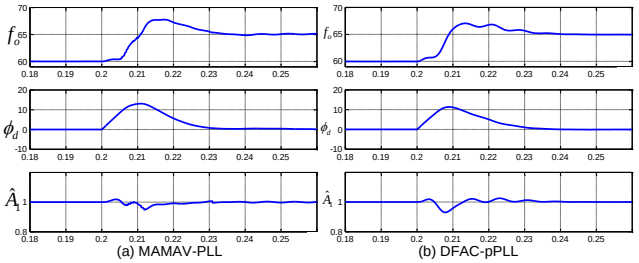

Fig. 20. Experimental waveforms for 5*Hz* frequency jump. PLL frequency f_o, PLL output phase angle ϕ_o and estimated amplitude $\hat{A}_1$.

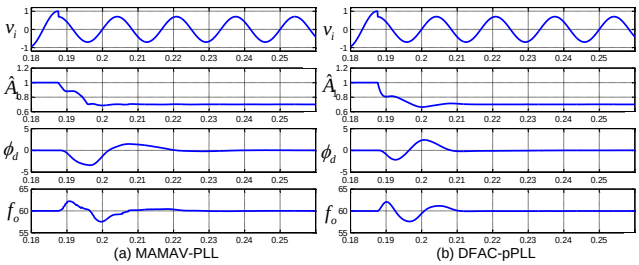

Fig. 21. Experimental waveforms 30% voltage sag. Input signal v_i, estimated amplitude $\hat{A}_1$, PLL output phase angle ϕ_o and PLL frequency f_o.

Table 2. Experimental Transient Tests

	pPLL		DFAC -pPLL		MAV	AMAV -PLL	MAMAV -PLL
	(a)	(b)	(a)	(b)	(b)	(b)	(b)
+40° Phase jump							
2% Settling time (cycles)	7	5.14	2.4	2.50	2.12	2.14	2.09
Overshoot (degree)	23	20.12	15	14.65	18.84	17.24	19.34
+5*Hz* Frequency step							
2% Settling time (cycles)	7	5.00	2.4	2.34	2.06	2.14	2.13
Peak Error (degree)	30	31.04	10	11.35	12.80	12.84	13.07
30% Voltage sag							
0.3° Settling time (cycles)	5	4.96	1.2	1.17	2.82	2.81	1.85
Peak Error (degree)	2	1.81	2	2.39	4.08	4.14	3.41

(a) results from original papers. (b) measured in this paper.

The DFAC-pPLL has a faster transient response as compared to the pPLL and as good as the non-adaptive MAV-PLL. The non-adaptive MAV-PLL transient response improvement was achieved by only changing the PI controller design as compared with the one from [3-4,6,12]. It presents lower settling time than the DFAC-pPLL for phase and frequency jump, but the use of non-adaptive MAV filter degrades PLL steady state performance for operation out of the nominal frequency. This problem can be partially solved by using the AMAV-PLL that will be discussed in section IV.C.1.

AMAV-PLL also offers low settling time for phase and frequency jump, but both the MAV-PLL and AMAV-PLL have a higher settling time in the case of voltage sag. This settling time is improved in the case of MAMAV-PLL due to

the use of amplitude normalization. The MAV-based PLLs present slightly higher overshoot and peak phase error than DFAC-pPLL.

C. Steady state tests with input harmonics distortion

The steady state performance of the adaptive MAV-based PLLs (AMAV-PLL and MAMAV-PLL) for varying input frequency f_1 is analyzed in section IV.C.1. Section IV.C.2 compares all studied PLLs steady state performance using the pPLL and DFAC-pPLL original controller parameters.

In spite of its low settling time, the DFAC-pPLL offers lower attenuation for the input voltage harmonics, resulting in higher peak to peak phase ripple $\Delta\phi_o$. Thus, to have a fair comparison of the transient response based on the same steady state performance, section IV.C.3 redesigns the DFAC-pPLL so as to present a peak to peak phase ripple equal to the worst case shown by the MAMAV-PLL, and then the transient response is compared.

1) Steady state performance of the adaptive MAV based PLLs (AMAV-PLL and MAMAV-PLL) for varying input frequency f_1

Fig. 22 presents the experimental behavior of $\Delta\phi_o$ over the range ($59 \leq f_1 \leq 65$) *Hz*, considering input signal with A_1=100% and A_3=15%, for the MAMAV-PLL. $\Delta\phi_o$ was evaluated with step resolution of $\Delta f_1 = 0.01Hz$.

From Fig. 22, the higher peak to peak phase error is $\Delta\phi_o \approx 0.15^o$ at $f_1 \approx 64.15Hz$. Fig. 22 behavior can be explained by the presence of multiple notches, characteristic of adaptive MAV filter frequency response (Figs. 13). The lowest peak to peak phase ripple $\Delta\phi_o \approx 0$ (corresponding to the maximum MAV filter attenuation) occurs at the input frequencies given by (19):

$$f_1 = f_s / (2.N) \text{ ,} \tag{19}$$

derived from (12). Fig. 22 was obtained with $f_s = 12kHz$ and integer values of N varying in the range $99 \leq N \leq 107$ samples.

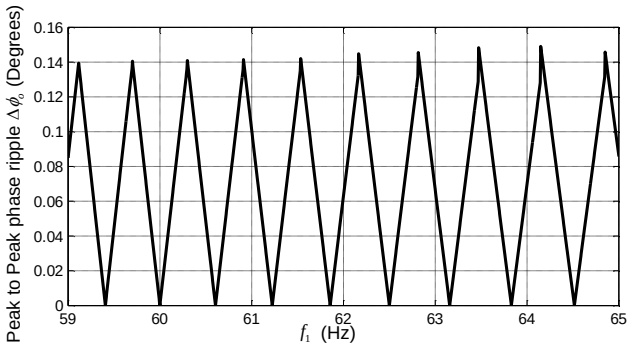

Fig. 22 Experimental MAMAV-PLL peak to peak phase ripple $\Delta\phi_o$ for $59 \leq f_1 \leq 65$ *Hz* range with 15% of the 3rd harmonic.

2) Comparison between pPLL and DFAC-pPLL using original parameters with proposed MAV based PLLs

Table 3 presents the steady state peak to peak phase ripple error $\Delta\phi_o$ for the PLLs studied. The controller and filter parameters are the same used in section IV.B. As explained in section IV.C.1, the adaptive MAV-based PLLs performance depends on the input signal frequency f_1 as shown in Fig. 22. Therefore, the following frequencies f_1=60*Hz*, f_1=60.3*Hz* and f_1=60.606*Hz* are applied for evaluation. To allow comparison, distorted input signal cases include 15% of 3rd harmonic as used by [1-2].

Table 3. Experimental steady state tests – peak to peak phase ripple error (degrees)

	pPLL		DFAC -pPLL		MAV	AMAV -PLL	MAMAV -PLL
	(a)	(b)	(a)	(b)	(b)	(b)	(b)
60*Hz* pure sine		0.16		0.01	0.00	0.00	0.00
60*Hz*+15% 3rd harmonic	≈0	0.18	1.7	1.62	0.00	0.00	0.00
60.3*Hz* pure sine		0.16		0.01	0.13	0.13	0.13
60.3*Hz*+15% 3rd harmonic		0.18		1.61	0.14	0.14	0.14
60.606*Hz* pure sine		0.15		0.01	0.25	0.00	0.00
60.606*Hz*+15% 3rd harmonic		0.18		1.59	0.28	0.00	0.00

(a) results from original papers. (b) measured in this paper

The comparison between experimental results in this paper and from the original ones show good agreement, as verified before in the transient tests. The pPLL has good filtering performance under all steady state conditions, but as discussed in section IV.B (Table 2), it has a slow transient response. The DFAC-pPLL has good steady state performance for variable frequency non-distorted grid input voltage, but its performance is degraded with the presence of the 3rd harmonic.

The non-adaptive MAV-PLL performance is degraded as the mains frequency f_1 deviates from the nominal frequency ($f_1 = 60Hz$). For a limited frequency deviation, in the range of $56.44 \leq f_1 \leq 63.97Hz$, the non-adaptive MAV-PLL offers a better steady state (peak to peak phase ripple $\Delta\phi_o \leq 1.62^o$) for harmonic polluted mains when compared with the DFAC-pPLL. The AMAV-PLL and MAMAV-PLL offer a better steady state performance for harmonic polluted mains when compared with the DFAC-pPLL, for all the measured cases.

3) Comparison between readjusted DFAC-pPLL and proposed MAV based PLLs

In this section, the DFAC-pPLL is readjusted using the original paper design guidelines [1] to have a similar input harmonics rejection as the MAMAV-PLL worst case (maximum peak to peak phase ripple $\Delta\phi_o \approx 0.15^o$, as discussed in section IV.C.1), with distorted input voltage (A_1=100% and A_3=15%). The new parameter values are k_P=43.35, k_I=783.15, ω_c=6.9*Hz*, ω_p=16.56*Hz*.

Tables 4 and 5 present transient and steady state experimental results for the readjusted DFAC-pPLL. In order to facilitate the comparison, the MAV filter based PLLs and the original DFAC-pPLL results are also shown.

The frequency $f_1 \approx 64.150Hz$ was included in Table 4 in order to obtain maximum peak to peak phase ripple $\Delta\phi_o \approx 0.15^o$, for the AMAV-PLL and MAMAV-PLL, as discussed in section IV.C.1.

Table 4 shows that the readjusted DFAC-pPLL met the specified steady state performance for distorted input signals. Table 5 presents the transient performance of the redesigned DFAC-pPLL together with the original DFAC-pPLL and the MAV filter based PLLs.

The steady state improvement of the redesigned DFAC-pPLL, achieved by PLL bandwidth reduction, led to the increase of the settling time for all the transient tests.

For harmonic polluted systems, for instance, and smart grids with low power distributed generation, the MAV filter based PLLs presented a good tradeoff between transient and steady state performance. Additionally, it presents a computationally efficient implementation (MAV filter plus multiplier type phase detector), even using frequency adaptive MAV filters.

Table 4. Steady state Tests with Redesigned DFAC-PPLL – peak to peak phase error (degrees)

	Redesigned DFAC -pPLL	Original DFAC -pPLL	MAV	AMAV -PLL	MAMAV -PLL
60*Hz* pure sine	0.00	0.01	0.00	0.00	0.00
60*Hz*+15% 3rd harmonic	0.15	1.62	0.00	0.00	0.00
60.3*Hz* pure sine	0.00	0.01	0.13	0.13	0.13
60.3*Hz*+15% 3rd harmonic	0.15	1.61	0.14	0.14	0.14
60.606*Hz* pure sine	0.00	0.01	0.25	0.00	0.00
60.606*Hz*+15% 3rd harmonic	0.15	1.59	0.28	0.00	0.00
64.150*Hz* pure sine	0.00	0.00	1.46	0.12	0.12
64.150*Hz*+15% 3rd harmonic	0.13	1.43	1.68	0.15	0.15

Table 5. Transient Tests with Redesigned DFAC-pPLL

	Redesigned DFAC -pPLL	Original DFAC -pPLL	MAV	AMAV -PLL	MAMAV -PLL
+40° Phase jump					
2% Settling time (cycles)	8.55	2.50	2.12	2.14	2.09
Overshoot (degree)	13.59	14.65	18.84	17.24	19.34
+5*Hz* Frequency step					
2% Settling time (cycles)	8.33	2.34	2.06	2.14	2.13
Peak Error (degree)	36.37	11.35	12.80	12.84	13.07
30% Voltage sag					
0.3° Settling time (cycles)	1.8	1.17	2.82	2.81	1.85
Peak Error (degree)	1.23	2.39	4.08	4.14	3.41

D. Amplitude estimator performance for the DFAC-pPLL and MAMAV-PLL

Section IV.C compared the peak to peak phase ripple $\Delta\phi_o$ performance for the DFAC-pPLL and the MAMAV-PLL. This section analyses the peak to peak amplitude ripple $\Delta\hat{A}_1$ of the amplitude estimators of the DFAC-pPLL and the MAMAV-PLL, because the increase of $\Delta\hat{A}_1$ value results in distorted PLL output signals (ϕ_0, $\bar{v}_o$, v_o, θ_o etc.). For the DFAC-pPLL, $\Delta\hat{A}_1$ increases with the distortion of the input signal v_i, as shown in the experimental result of Fig. 23, where A_1=100% and the 3rd harmonic A_3 varies in the range from 0 to 20%, keeping constant input frequency f_1 =60*Hz*.

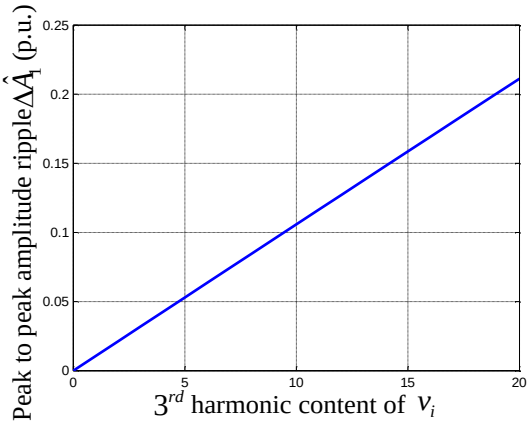

Fig. 23. Experimental DFAC-pPLL amplitude estimator peak to peak amplitude ripple $\Delta\hat{A}_1$ performance evaluation under input voltage distortion variation

The DFAC-pPLL presents only small variation in $\Delta\hat{A}_1$ with the input f_1 frequency increase, depending instead on the input v_i voltage distortion. Conversely, the MAMAV-PLL peak to peak amplitude ripple $\Delta\hat{A}_1$ depends mainly on input frequency f_1 variation but not significantly on the input signal v_i distortion. Fig. 24 experimentally evaluates $\Delta\hat{A}_1$ in the range of $59\leq f_1 \leq 65Hz$, using A_1=100% and A_3=15%. From Fig. 24, the worst case for the MAMAV-PLL is $\Delta\hat{A}_1 < 0.01$ p.u. This result is nearly ten times lower than the result for the DFAC-pPLL, from Fig. 23, at about $\Delta\hat{A}_1 \approx 0.16$ p.u.

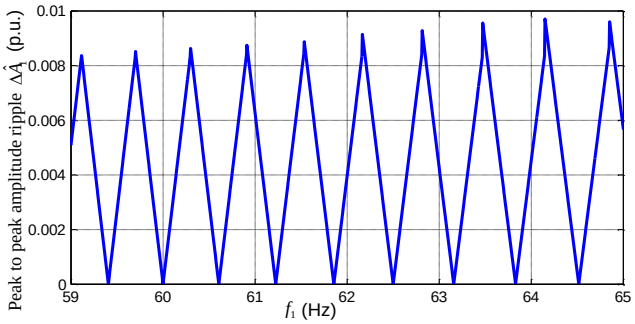

Fig. 24. Experimental MAMAV-PLL peak to peak amplitude ripple $\Delta\hat{A}_1$ for $59\leq f_1 \leq 65Hz$ range with 15% of the 3rd harmonic.

V. Conclusions

An analysis of some PLLs was presented, including the pPLL and the DFAC-pPLL, as well as three MAV filter-based PLLs, followed by an experimental comparison in transient and steady state conditions. A hardware and software-based experimental workbench is proposed to obtain precise measurements and accurate comparison.

The pPLL provides good phase ripple attenuation, at the expense of slow transient performance (bandwidth BW=24*Hz*[5]). it presents phase ripple caused by the second harmonic generated by the phase detector as well as the input signal v_i harmonics. The DFAC-pPLL solves the phase ripple problem caused by the second harmonic generated by the phase detector, using a Double Frequency cancellation strategy, allowing simultaneously low phase ripple and good transient response (BW=42*Hz*). Notwithstanding the cancelation of the second harmonic, input signal v_i harmonics cause significant phase ripple. The MAV filter-based PLLs have good attenuation of the second harmonic as well as the harmonics caused by distorted input signal v_i. The controller project of MAV filter-based PLLs ignoring MAV filter dynamics usually results in slow transient response PLLs. This work proposes the use of a discrete controller design procedure, taking into account the complete PLL open loop transfer function with N+2 order (including the MAV filter, VCO and PI controller). The resulting MAV filter based PLLs present good transient response (BW=46.7*Hz*). It is faster than all previously reported MAV filter-based PLLs with PI controllers, presenting better transient and steady state tradeoff than the recently published DFAC-pPLL. The non-adaptive MAV filter has a well known frequency variation problem, solved with the adaptive MAV filter. The assured attenuation for the adaptive MAV filter is quantified, which depends on the adopted fixed sampling frequency of the PLL (the higher the sampling frequency, the better the adaptive MAV filter attenuation).

The DFAC-pPLL and the MAMAV-PLL use input signal amplitude normalization to avoid degraded performance due to voltage sags. By considering a PLL project to operate within a frequency range $59\leq f_1 \leq 65Hz$ and third harmonic (A_3=15%) distorted input voltage v_i[6], it is shown that, with the proposed amplitude estimator, there is lower amplitude ripple in steady state than with that of the DFAC-pPLL.

[5] -3dB closed loop bandwidth, measured using the linearized model.

[6] These figures were used in the original papers of the DFAC-pLL and the pPLL and adopted here for comparative analysis.

For a fair comparison, the DFAC-pPLL was readjusted in order to provide the same steady state peak to peak phase ripple compared to the AMAV-PLL and MAMAV-PLL for the same distorted input. The resulting settling time was higher than that from the original DFAC-pPLL as well as the adaptive MAV filter-based PLLs.

Computationally efficient MAV filter-based PLL implementation lead to better tradeoff between transient and steady state performance amongst the analyzed PLLs, even using a simple PI controller and conventional multiplier type phase detector. These adaptive MAV filter-based PLLs are good alternatives for operation with variable frequency and distorted voltages, as those found in weak systems, such as small-scale smart grids with distributed generation.

Acknowledgment: This work was supported by CNPq and FDTE (postdoctoral fellowship at University of Sao Paulo).

REFERENCES

[1] S. Golestan, M. Monfared, F.D. Freijedo, and J.M. Guerrero, "Design and Tuning of a Modified Power-Based PLL for Single-Phase Grid-Connected Power Conditioning Systems," Power Electronics, IEEE Transactions on, vol. 27,pp. 3639-3650, 2012.

[2] R. M. Santos Filho, P. F. Seixas, P. C. Cortizo, L. A. B. Torres, and A. F. Souza, "Comparison of Three Single-Phase PLL Algorithms for UPS Applications," Industrial Electronics, IEEE Transactions on, vol. 55, pp. 2923-2932, 2008.

[3] F.P. Marafao, S.M. Deckmann, J.A. Pomílio, and R.Q. Machado, "A Software-Based PLL Model: Analysis and Applications," in Brazilian Automatic Conference (CBA), 2004.

[4] M. S. Padua, S. M. Deckmann, and F. P. Marafao, "Frequency-Adjustable Positive Sequence Detector for Power Conditioning Applications," in Proc. IEEE 36th Power Electronics Specialists Conf. PESC '05, 2005, pp. 1928-1934.

[5] Naji Ama, Rayra Destro, Wilson Komatsu, Fuad Kassab, and Lourenco Matakas, "PLL performance under frequency fluctuation-compliance with standards for distributed generation connected to the grid," in Innovative Smart Grid Technologies Latin America (ISGT LA), 2013 IEEE PES Conference On, 2013, pp. 1-6.

[6] F.D. Freijedo, J. Doval-Gandoy, O. Lopez, and E. Acha, "Tuning of Phase-Locked Loops for Power Converters Under Distorted Utility Conditions," Industry Applications, IEEE Transactions on, vol. 45, pp. 2039-2047, 2009.

[7] Remus Teodorescu, Marco Liserre, and Pedro Rodríguez, Grid Converters for Photovoltaic and Wind Power Systems.: John Wiley & Sons, Ltd, 2011.

[8] R.E. Best, Phase Locked Loops.: McGraw-Hill, 2007.

[9] F.M. Gardner, Phaselock Techniques.: Wiley, 2005.

[10] Naji Rajai Nasri Ama, F.O. Martinz, L. Matakas, and F. Kassab, "Phase-Locked Loop Based on Selective Harmonics Elimination for Utility Applications," Power Electronics, IEEE Transactions on, vol. 28, no. 1, pp. 144-153, jan. 2013.

[11] Andrzej Mondzik, Marcin Baszynski, Stanislaw Piróg, Adam Penczek and Milosz Szarek, "Phase-Locked Loop for Grid-Connected Power Electronics Converters" Przeglad Elektrotechniczny, no. 2, p. 335, 2010.

[12] S.M. Deckmann, F.P. Marafao, and M. S. Padua, "Single and Three-Phase Digital PLL Structures based on Instantaneous Power Theory," in Brazilian Power Electronics Conference (COBEP), Fortaleza, Brazil., 2003.

[13] L.G.B. Rolim, D.R. da Costa, and M. Aredes, "Analysis and Software Implementation of a Robust Synchronizing PLL Circuit Based on the pq Theory," Industrial Electronics, IEEE Transactions on, vol. 53,pp. 1919-1926, 2006.

[14] IEC 61727, Characteristics of the utility interface for photovoltaic (pv) systems.

[15] IEEE Std. 1547 - Standard for Interconnecting Distributed Resources with Electric Power Systems.

[16] VDE-AR-N 4105, "Technical requirements for the connection to and parallel operation with low-voltage distribution networks", 2011.

[17] VDE 0126-1-1:automatic disconnection device between a grid parallel power-generating system and the public low voltage grid.,VDE Verlag GmbH, Berlin: 2006.

[18] I. Carugati, P. Donato, S. Maestri, D. Carrica, and M. Benedetti, "Frequency Adaptive PLL for Polluted Single-Phase Grids," Power Electronics, IEEE Transactions on, vol. 27, pp. 2396-2404, 2012.

Authors: *Naji Rajai Nasri Ama, Polytechnic School of the University of Sao Paulo, Electrical Energy and Automation Dept. PEA, Av. Prof. Luciano Gualberto, tr. 3, n.158, 05508-900 Sao Paulo, SP, Brazil, E-mail: naji@pea.usp.br.*

Wilson Komatsu, Polytechnic School of the University of Sao Paulo, Electrical Energy and Automation Dept. PEA, Av. Prof. Luciano Gualberto, tr. 3, n.158, 05508-900 Sao Paulo, SP, Brazil, E-mail: wilsonk@usp.br.

Fuad Kassab Junior, Polytechnic School of the University of Sao Paulo, Dep. de Engenharia de Telecomunicações e Controle PTC, Av. Prof. Luciano Gualberto, tr. 3, n.158, 05508-900 Sao Paulo, SP, Brazil, E-mail: fuad@lac.usp.br.

Lourenco Matakas Junior, Polytechnic School of the University of Sao Paulo, Electrical Energy and Automation Dept. PEA, Av. Prof. Luciano Gualberto, tr. 3, n.158, 05508-900 Sao Paulo, SP, Brazil, E-mail: matakas@pea.usp.br.