Abstract—The analysis and design of a three-phase Phase Locked Loop system to synchronize to the electrical grid is presented. The whole phase locked loop model is obtained and a Proportional Integral, a Proportional Integral Derivative and a novel Fractional Proportional Integral controllers in continuous time domain are proposed to solve the phase tracking by improving time response, phase error and overshoot. The performance of the three controllers is analyzed under distorted utility conditions, such as imbalance, harmonics and phase and/or frequency variations. The phase locked loop is totally implemented by software, and tested by simulation via SIMULINK and experimentally via xPCTarget, in a three phase Autoadjustable Synchronous Reference Frame application. A comparison in the dynamic and steady state of the phase locked loop parameters like accurate, robustness and fastness is carried out for the three different controllers.

Index Terms—Phase locked loops, modeling, PI control, PID control, fractional control, power systems control.

I. INTRODUCTION

Nowadays, the electric power quality is hardly affected by the increase of non-linear loads connected to the grid. To assure minimum specifications in the quality, reliability and use of the electric magnitudes, the utility interface operation of power converters such as ac/dc converters, active filters, cycloconverters, etc. is very common.

The positive-sequence fundamental phase angle of the utility voltage is a critical piece of information for this kind of devices in the operation of most power conversion and conditioning applications such as active power filters, quality meters of electrical magnitudes, uninterrupted power supplies and grid connected photovoltaic or wind power generation systems, so they need an accurate phase tracking system to work properly.

The phase obtained is used to construct a reference carrier wave to synchronize the on/off commutations of power devices, calculate and control the flow of active/reactive power or transform the feedback variables to a reference frame suitable for control purposes.

Some methods have been developed to track the phase, which in a general way can be divided into two groups [1]: open-loop methods, such as low pass filters, space vector filters or extended Kalman filters, which directly estimate the phase angle of the PCC voltage from its alfa-beta coordinates, obtained by the Clarke transformation [2][3]; and closed-loop methods, mainly Phase Locked Loop (PLL) methods, in which the phase angle estimation is adaptively updated by a closed-loop mechanism whose objective is to track the real value of the angle [4]-[11]. These methods have, however, the shortcomings of not operating properly under imbalance, harmonic distortion or frequency variations. Most recently, Adaptive PLL [12] and Enhanced PLL [13]-[14] have been developed with the objective of getting frequency adaptation and unaffected robust response under harmonics or imbalance in the input signals. However, these methods propose three-phase structures formed by four single-phase modules, which complicate the control stage and require heavy computation.

In electric power systems, the instantaneous angle and frequency information are typically recovered using a time basis PLL technique. The quality of the PLL is the key to obtain the maximum performance of the application, so it must lock the phase of the utility voltage at the Point of Common Coupling (PCC), which is distorted due to imbalance, harmonic, and phase or frequency variations, as quickly and accurately as possible. Because of that, it is essential to develop a controller which provides fast time response, zero error in the steady state and validity under any input signal for the PLL.

The simplest controller that fits these specifications is the Proportional Integral (PI). It could be designed in continuous-time or discrete-time domain [8]. Then, the regulator gains could be calculated through different control strategies such as symmetrical optimum [7], Wiener optimization [8], second order system pattern [7][16], etc.

This paper focuses on the design and operation comparison of three types of different controllers for a PLL: Proportional Integral (PI), a Proportional Integral Derivative (PID), and a Fractional Proportional Integral (FPI), in order to find the one with best performance. The paper is structured as follows. Firstly, the explanation of the PLL applied to an Autoadjustable Synchronous Reference Frame (ASRF) [14] application is exposed. Then, a PLL model is obtained. With the model, an open loop transfer function is attained and the PI and PID controllers are designed by using the root locus method. The FPI is designed in the frequency domain by improving the frequency response of the previous two designed controllers. Afterwards, the closed loop model is simulated and the PLL is implemented by software to carry out an ASRF application, which consists on the extraction of the positive-sequence fundamental three phase component of an unbalanced three-phase signal with harmonics. The ASRF is checked by simulation using SIMULINK, and then, it is used in experimental environment with the xPCTarget platform. Finally, the results of the three controllers are summarized in a comparison table.

Comparison of Controllers for a Three-phase Phase Locked Loop System under Distorted Conditions

V. Miñambres1,2, M.I. Milanés1,2, B. Vinagre2, E. Romero1,2

1Power Electrical & Electronic Systems (PE&ES)
2School of Industrial Engineering. University of Extremadura. Badajoz, Spain
II. THEORETICAL EXPLANATION OF THE PLL AND THE ASRF APPLICATION

The PLL and the ASRF application used to compare the controllers proposed in this paper are shown in Fig. 1. In the following sections, a theoretical explanation of the PLL under sinusoidal input voltages is firstly presented. Afterwards, the adaptation of the model necessary in case of distorted input voltages is detailed, and finally the ASRF application is explained.

A. Tracking Method. PLL under sinusoidal input voltage

The instantaneous three-phase voltage at the PCC \( u_{PCC}(a,b,c) \) could be converted to the \( 0-d-q \) dynamic reference frame \( u_{PCC}(0,d,q) \) using the Park transformation:

\[
\begin{align*}
\mathbf{P} & = \begin{bmatrix}
\frac{2}{3} & \frac{1}{3} & \frac{1}{3} \\
\cos \theta & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta + \frac{2\pi}{3}) \\
\sin \theta & -\sin(\theta - \frac{2\pi}{3}) & -\sin(\theta + \frac{2\pi}{3})
\end{bmatrix}, \\
\mathbf{u}_{PCC}(0,d,q) & = \mathbf{P} \cdot \mathbf{u}_{PCC}(a,b,c),
\end{align*}
\]

where \( \mathbf{P} \) is the transformation matrix, defined as:

\[
\begin{align*}
\mathbf{u}_{PCC}(0,d,q) & = \mathbf{P} \cdot \mathbf{u}_{PCC}(a,b,c), \\
\mathbf{P} & = \begin{bmatrix}
\frac{2}{3} & \frac{1}{3} & \frac{1}{3} \\
\cos \theta & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta + \frac{2\pi}{3}) \\
\sin \theta & -\sin(\theta - \frac{2\pi}{3}) & -\sin(\theta + \frac{2\pi}{3})
\end{bmatrix}.
\end{align*}
\]

If the input voltage is sinusoidal with amplitude \( U \), then:

\[
\begin{align*}
\mathbf{u}_{PCC}(a,b,c) & = \begin{bmatrix}
U \sin(\omega_a t + \phi) \\
U \sin(\omega_b t + \phi - \frac{2\pi}{3}) \\
U \sin(\omega_c t + \phi + \frac{2\pi}{3})
\end{bmatrix},
\end{align*}
\]

Denoting \( \omega_a t + \phi \) by \( \theta_a \), (3) and (1) could be substituted in (2) and, after applying some trigonometric relations and rejecting the homopolar component (since it will not be used), the following expression is obtained:

\[
\mathbf{u}_{PCC}(d,q) = \begin{bmatrix}
\mathbf{u}_{PCC,d} \\
\mathbf{u}_{PCC,q}
\end{bmatrix} = \begin{bmatrix}
U \sin(\theta_a - \theta) \\
-\cos(\theta_a - \theta)
\end{bmatrix},
\]

When the difference angle \( \Delta \theta = \theta_a - \theta \) is big, the angle \( \theta \) is locked soon due to the feedback. Then, \( \Delta \theta \) will be small, and the Taylor approximation for the sine and cosine around zero could be applied [16]:

\[
\mathbf{u}_{PCC}(d,q) = \begin{bmatrix}
\mathbf{u}_{PCC,d} \\
\mathbf{u}_{PCC,q}
\end{bmatrix} = \begin{bmatrix}
\frac{3}{2}U \sin(\theta_a - \theta) \\
\frac{3}{2}U (-\cos(\theta_a - \theta))
\end{bmatrix}.
\]

Finally, the voltage division is calculated to work only with the angle, that is, \( u_d \) is normalized:

\[
u_d / u_q = -\Delta \theta.
\]

With this approach, the PLL does not depend of the level of the voltage input.

B. Adaptation of the PLL to be used under distorted input voltage

Taking into account that the input voltage is distorted, the expressions exposed in the previous section are only valid if the positive sequence fundamental component is obtained from the input. This is carried out by a Butterworth Low Pass Filter (LPF) in the \( d \) and \( q \) components. With a cut-off frequency of \( f_c = 13 \text{Hz} \) the LPF gets their mean values \( U_d \) and \( U_q \), which assures no harmonics in each one. The first one is used for the PLL control, and both of them for the ASRF reconstruction.

C. PLL and ASRF application

The aim is to make zero \( U_d / U_q \) through the PLL controller \( C \), which is possible when \( \Delta \theta = 0 \), as can be extracted from (6) . The division achieves the controller to be valid for any level of the input voltage. On the one side, the angle \( \theta \) will correspond to the angle of the \( q \) axis and, on the other side, \( U_q \) will carry on with the positive-sequence fundamental signal.

The PLL control loop consists on a \( C \) controller which takes its error signal \( U_d / U_q \) and gives the amount of pulsation \( \Delta \omega \) necessary to add or subtract to a reference pulsation \( \omega_0 \) to catch the input voltage pulsation \( \omega = 2\pi f \). The LPF makes the controller to be immune to the input harmonics and imbalance. Obviously, \( \omega_0 \) is calculated as the nominal value of the input voltage frequency (50Hz in electric systems) which helps the algorithm to improve the convergence. The result is the fundamental pulsation \( \omega = 2\pi f \), so the positive sequence fundamental angle \( \theta \) is obtained by integrating it. The phase will be locked to the \( q \) axis rotation when the error signal \( U_d / U_q = 0 \), that is, \( \omega = \omega_0 \) and \( \theta = \theta_0 \), so \( \Delta \theta = 0 \).

The ASRF reconstruction of the positive-sequence fundamental voltage \( u_{PCC,1}(a,b,c) \) is possible by applying the inverse Park transformation \( \mathbf{P}^{-1} \) to \( U_q \) with the angle \( \theta \). The component \( U_d \) does not affect in the steady state, but using it a softer dynamic state is achieved, and the homopolar is rejected.

III. MODEL OF THE PLL

The model of the designed PLL is presented in Fig. 2. It is obtained from the PLL system after being linearized (5) and normalized (6). Obviously, the LPF is taking into account to design properly the controller \( C \). An approximation delay is modeled too, to simulate the sample rate \( T_s \) of the algorithm.

From the model, the open loop transfer function without \( C \) is derived \((T_s = 1e-4 \text{s})\):

\[
\begin{align*}
\mathbf{PLL}_{\text{LPF}}(s) & = \mathbf{LPF}(s) \frac{1}{s} \frac{1}{s+1} \\
& = \frac{(2\pi f_c)^2}{s^2 + 2\sqrt{2} f_c s + (2\pi f_c)^2} \frac{1}{s+1} \\
& = \frac{6672}{s^2 + 1.012 s + 166.2} \frac{1}{s+1}.
\end{align*}
\]

(8)

The root locus is shown in Fig. 3.
Now, by tuning the controller $C$, the poles of the system could be located where the system response is stable and fast.

IV. DESIGN OF THE CONTROLLER FOR THE PLL

The controller block $C$ for the PLL must be, at least, a PI, to obtain a zero steady state error, because the model is similar to the position control of a servo with a ramp input. In this paper it will be analyzed the classical PI and PID controllers and, as a novelty, a FPI controller is designed and compared with the usual ones.

A. PI controller

The PI controller is designed by using the root locus tool from the MATLAB toolbox. The pole introduced by the PI is located at the origin, and the zero is located where the root locus shows the best stable situations (-14.33). The criteria selected then to locate the four poles is based on the closeness to the $x$ axis. Fig. 4a shows the result: Two pairs of complex conjugate poles at the same position and one real pole with a high negative value.

The PI controller responds to the next expression:

$$C_{PI}(s) = K_p + \frac{K_i}{s} = 28.8526 + \frac{412.18}{s}. \quad (9)$$

B. PID controller

The PID controller is designed like the PI. The difference is that the PID introduces two zeros (both at -57.59 in the design), that is, an added degree of freedom, so the root locus has more stable situations, plus a faster transitory. Fig. 4b displays the result: Two pairs of complex conjugate poles at the same position and one real pole with a high negative value.

The PID controller responds to the next expression:

$$C_{PID}(s) = K_p + \frac{K_i}{s} + K_d s = 56.382 + \frac{1658.3}{s} + 0.479 s. \quad (10)$$

C. FPI controller

Fractional calculus is a generalization of the integration and differentiation to the non-integer (fractional) order fundamental operators $D^\alpha$, where $a$ and $t$ are the limits and $\alpha (\alpha \in \mathbb{R})$ is the order of the operation. Among many different definitions, one of the most commonly used for the general fractional integro-differential operation is the Grünwald–Letnikov (GL):

$$\bigtriangleup D^\alpha f(t) = \lim_{h \to 0} h^{-\alpha} \sum_{j=0}^{[t-a]/h} (-1)^j \binom{\alpha}{j} f(t-jh), \quad (11)$$

where $[\cdot]$ means the integer part.

Podlubny [17] proposed a generalization of the PID controller, namely the PI$^D$ controller, involving an integrator of order $\lambda$ and a differentiator of order $\mu$. He also demonstrated the better response of this type of controller, in comparison with the classical PID controller, when used for the control of fractional order systems. A frequency domain approach by using fractional order PID controllers was also studied [18].

In power electronics, fractional order controllers have been successfully applied in [19] and [20]. In the first one, a Power Electronic Buck Converter is controlled by using different controllers: with an order of integration less than one (PI), the FPI could maintain zero error on steady state by introducing less phase lag.

The generalized PI$^D$ controller has the next form (see [21] and references therein):

$$C_{PI^D}(s) = K_p + \frac{K_i}{s^\lambda} + K_d s^\mu. \quad (12)$$

The FPI controller is designed in the frequency domain for improving the phase margin obtained with the other two controllers: with an order of integration less than one (PI), the FPI could maintain zero error on steady state by introducing less phase lag.

The final three Bode plots of the open loop model systems for each controller are shown in Fig. 5. As it can be seen, the stable conditions are met in a very narrow margin of frequencies for the PI and PID, so there is another possible improvement here for the FPI: The phase margin of the model with the FPI is larger than the others models (see the values listed in TABLE. I), so a more stable and robust system is
expected. The disadvantage is the DC gain response, which is smaller too, and the steady state will be reached slowly. The phase margin reveals that the PID is able to adapt to higher changes.

Finally, the FPI controller responds to the next expression:

\[ C_{FPI}(s) = K_p + \frac{K_i}{s} = K_p + \frac{K_i}{s^{1-j}} = 20 + \frac{100}{s^{0.3}}. \quad (13) \]

By defining an integrator, and then, the complementary fractional term, the error in the steady state is forced to zero.

The fractional controller is implemented by the “fractional PID” block from the “ninteger” [22] MATLAB toolbox with the fractional parameters \( crone, melttime, frac\ n = 10 \) and \( bandwidth = 0.1-100 \) [22].

V. SIMULATION ANALYSIS OF THE MODEL

The PLL model (see Fig. 2) has been simulated with an input angle \( \theta_n \) modeled by a ramp with a slope \( \omega_n = 2\pi f_m \). If the input voltage is sinusoidal and balanced, the plot of the voltage vector will rotate with constant frequency and the phase changes linearly.

Fig. 6 shows the results \( U_d/U_q \) vs \( \theta_n = f_m - f \) and \( \theta_n = \Delta\theta \) in a simulation test with \( f_m = 51\ Hz \). One observes that the FPI controller is the slowest one with a small steady state error that is going close to zero. The PID is the fastest and the PI stays as an intermediate solution. The PI and PID offer quasi-zero steady state error. The controllers are well designed.

VI. ASRF ALGORITHM SIMULATION

The ASRF algorithm (see Fig. 1) has been simulated with a three-phase PCC voltage input with harmonics and imbalance. In Fig. 7 the PLL variables are shown, while Fig. 8 and Fig. 9 displays the ASRF inputs and outputs in transient and steady state, with the next input voltage parameters: phase \( a \) 230 V, phase \( b \) 180 V, phase \( c \) 280 V, all the phases with \( 1/7 \) of \( 7^{th} \) harmonic and 51 Hz. The ASRF obtains the positive sequence fundamental component of the input voltage, at 51 Hz. The sample rate was \( 1e-4 \) s, that is, a frequency of \( 10\ kHz \).

The PLL algorithm response is similar to the PLL model simulation. The differences are mainly in case of the PID controller, which applies an excess effort in the beginning of the transient state and has some oscillations around zero in the steady state. These unexpected effects do not happen in the model simulation because of the linearization.

The phase error, which has been measured in the steady state, is close to zero (see the values listed in TABLE. I). The rapidness under the described situation (like a step frequency input) is also summarized in TABLE. I.

The input margin has been measured by changing the frequency of a sinusoidal and balanced input while the phase was not locked. Obviously, the PID is the best one with regards to this parameter because of its higher efforts, and consequently its rapidness. The PI and FPI are similar at this point. The values are in TABLE. I.

Harmonic Distortion (THD) of the output voltage for the phase \( a \). One can observe from TABLE. I that the THD is close to zero and equal in all the cases. This is because the LPF block is responsible for filtering the harmonic content and it has been used the same filter for all the cases.

The steady state has been analyzed to calculate the Total Harmonic Distortion (THD) of the output voltage for the phase \( a \). One can observe from TABLE. I that the THD is close to zero and equal in all the cases. This is because the LPF block is responsible for filtering the harmonic content and it has been used the same filter for all the cases.
VII. ASRF EXPERIMENTAL TESTS

A. Hardware

The hardware to implement the ASRF is based on the xPCTarget from MATLAB. It proposes a x86 based PC, called Target, to run the algorithm, which is controlled by another PC, called Host, via Ethernet. Obviously, to measure the PCC voltages it is necessary to install a xPCTarget compatible DAQ board. The Advantech embedded PC PPT-154T with the National Instruments PCI-6071E presented in Fig. 10 is the equipment used as Target.

The HP-6834B AC power source, which can be seen in Fig. 10, has been utilized to generate the PCC voltage. These voltages have been measured using HALL effect sensors.

B. Software

The ASRF algorithm (Fig. 1) has been implemented with the xPCTarget tool from MATLAB toolbox via SIMULINK. The code is obtained by compiling the ASRF block scheme and saved into a “dlm” format archive. To do that, a third party compiler is required (the selected was VISUAL BASIC).

The “dlm” archive is loaded in the Target through the “xpcexplorer” application in the Host via Ethernet. With this application, all the options and parameters of the Target loaded code could be controlled. The sample rate was 10 kHz.

The Target has its own real time OS. It is created through “xpcexplorer” by setting the proper configuration.

The block used to get the measurements has been the xPCTarget A/D of the DAQ board and then, the measured values are corrected according to the voltage sensor attenuation. The outputs are registered through the xPCScope block which allows saving the data into the Target hard disk.

C. Results

The experimental test has been carried out by setting a three-phase voltage input with the next parameters: Phase $a$ 230 V, phase $b$ 200 V, phase $c$ 260 V, and all the phases with 1/5 of 5th and 1/7 of 7th harmonics and 51 Hz, which represents a situation where the PCC voltage has imbalance and harmonics. After making the experiment, the saved data are analyzed. The results in the transient and in the steady state are displayed in Fig. 11 and Fig. 12, demonstrating that the ASRF works with the three controllers.
Fig. 11. Experimental tests of the ASRF algorithm in the transient state with different controllers. Input (distorted) and output (sinusoidal and balanced) signals.

Fig. 12. Experimental test of the ASRF algorithm in the steady state with different controllers. Input (distorted) and output (sinusoidal and balanced) signals.

According to the amplitudes of the three outputs, one can conclude from Fig. 13 that the most accurate controller is the PID.

VIII. COMPARISON BETWEEN THE CONTROLLERS

The results obtained in the comparison of the controllers carried out in section IV. C. and section VI. under specific simulation conditions, are summarized in TABLE. I. From this table, one can conclude that the selection of the controller depends on the kind of application and its necessities, because each one has their own pros and cons.

If the robustness is the priority, the FPI is at the top. The fastest is the PID. In the search of the widest input frequency range, the PID must be the selected again. The THD reveals that every controller works equally well under distorted conditions. The minimum phase error is reached by the PI.

The phase is tracked by avoiding the harmonic and imbalance of the input, so the output is the positive sequence fundamental three-phase voltage. This is demonstrated in Fig. 13 by showing the Individual Harmonic Distortion (IHD) ratio of the phase a for the three controllers (taking as base value the fundamental component of the phase a input voltage). As it can be seen, only the fundamental voltage remains so the THD is 0%, except in the PI case, which has a THD of 0.8%.

TABLE. I

<table>
<thead>
<tr>
<th></th>
<th>Robustness.</th>
<th>Settling time (s)</th>
<th>Input frequency margin (Hz)</th>
<th>THD (%)</th>
<th>Phase error (rad)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PI</td>
<td>32.8</td>
<td>0.22</td>
<td>39-61</td>
<td>0.8</td>
<td>3e-6</td>
</tr>
<tr>
<td>PID</td>
<td>26.1</td>
<td>0.16</td>
<td>12-103 (oscillating)</td>
<td>8.6e-3</td>
<td>0 (oscillating)</td>
</tr>
<tr>
<td>FPI</td>
<td>42.9</td>
<td>2.5 (±0.01rad)</td>
<td>40-59</td>
<td>8.6e-3</td>
<td>3.59e-4</td>
</tr>
</tbody>
</table>
IX. CONCLUSIONS

A full working three-phase PLL has been designed. Three different controllers have been tested to lock the phase: PI, PID and a novel FPI. Due to the use of LPFs and the normalization in the PLL model, each controller is immune to harmonics and imbalance, respectively, so all the controllers fit under any distorted condition. Every controller has been designed using well known control strategies, despite of the complex PLL system model.

The performance of each controller has been analyzed and summarized in a table in order to compare them. The result is a wide variety, because each controller has its own advantages and drawbacks, so one concludes that the selection must be done depending on the application specifications.

The PLL has also been tested experimentally for an ASRF application, confirming the performance with every controller.

X. REFERENCES


