A synchronverter-based magnitude phase-locked loop

Pietro Lorenzetti, Florian Reissner, George Weiss

A synchronverter-based magnitude phase-locked loop

Pietro Lorenzetti, Florian Reissner, and George Weiss, *Member, IEEE*

***Abstract*— A magnitude phase-locked loop (MPLL) is a system that synchronizes its output signal in frequency, phase, and magnitude with the dominant sinusoidal component of its input signal. We propose a novel MPLL design based on the model of a synchronverter (i.e., an inverter that behaves towards the power grid like a synchronous generator). The synchronverter model is detached from its usual three-phase power electronics environment, and transformed into a (single phase) MPLL with a wide pull-in range and great noise rejection properties. We prove synchronization under reasonable conditions. Extensive simulation results are provided to validate its performance, and to compare it with existing solutions.**

***Index Terms*— Magnitude phase locked loop, synchronverter, virtual synchronous machine, synchronization, singular perturbations, pendulum equation.**

I. Introduction

A *phase-locked loop* (PLL) is a system that synchronizes its output signal in frequency, as well as in phase, with the dominant sinusoidal component of the input signal [15]. When a PLL is able to synchronize also in magnitude, it is referred to as a *magnitude phase-locked loop* (MPLL). MPLLs and PLLs find applications in many fields, such as communications (adaptive notch filters), estimation problems, signal processing, control, and power electronics, see, e.g., [7], [14], [16], [20], [26], [42]. Most of the PLL architectures presented in the literature share a common design (see Fig. 1), which includes three main blocks: a phase detector (PD), a loop filter (LF), and a voltage-controlled oscillator (VCO) [7], [19], [20]. In its simplest implementation, the PD consists of a multiplier, which generates a signal containing the phase difference between the input signal and its estimated phase. This signal is then low-pass filtered through the LF, and used as input to the VCO. (More details on this simple model are in [20, Chap. 1]). Depending on the requirements (e.g., accuracy, robustness, noise rejection), more involved PD schemes can be implemented. We refer the reader to [14], [16], [36], [48] for an overview of the most recent techniques.

When the PLL input signal is three-phase, as it usually is in power system applications, it is common to use the Park transform (see Sec. II) to formulate the synchronization problem in the dq plane [45], [48]. Indeed, the Park transform maps (purely) sinusoidal signals with a certain reference frequency to constants, making the design and the analysis more straightforward. A similar approach can be used also for single phase signals. Nevertheless, the design becomes more involved due to the need (in the PD block) of an *orthogonal signal generator* (OSG) [4], [5], [13]. An ideal OSG is a system that receives a single phase signal as input and outputs a signal orthogonal to its input (in the sense that $\cos \omega t$ becomes $\sin \omega t$ for all $\omega > 0$). This system can be used to compute the dq components of single phase signals. Most of the single phase PLLs proposed in the literature are equipped with an OSG (an approximation of the ideal one), see, for instance, [11], [14]–[16], [43].

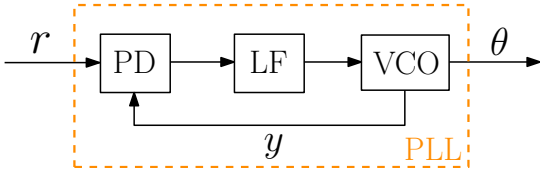


Fig. 1: Block diagram of a standard PLL [7], [19], [20], with input r and output θ, composed of a phase detector (PD), a loop filter (LF), and a voltage-controlled oscillator (VCO).

Rigorous stability results for PLLs can be found in [38], where density functions (see [37]) are used to assess stability for a simplified PLL, in [48] where passivity-based arguments are employed, in [36], where Lyapunov techniques are used, and in [26], [39], where more involved mathematical tools are used. Linearized models are studied [14], [15].

Recently, in [45] the structural resemblance between droop controllers [18], [23] and PLLs has been pointed out. This property has been exploited to operate a (slightly modified) classical droop controller for inverters as a PLL, without a dedicated synchronization unit. (Related findings are in [46].) Moreover, it has been shown how the droop-based PLL recovers the popular *enhanced* PLL (EPLL) from [19], [20] when the inverter output impedance is resistive, see [45, Sec. IV-B]. Using their words: "*Droop controllers and PLLs can be improved and further developed via adopting advancements in the other field.*" In this spirit, we aim at designing a novel MPLL with a wide pull-in range (in the sense of [25]), and great noise immunity properties, based on synchronverters.

Synchronverters [24], [27], [33], [35], [47] are a class of virtual synchronous machines: inverters controlled to imitate

This research is supported by the Israel Science Foundation (ISF), grant no. 2802/21. F. Reissner was a member in the ITN network WinGrid, funded by the European Union's Horizon 2020 research and innovation program under the Marie Sklodowska-Curie grant no. 861398.

P. Lorenzetti is with Université de Lorraine, CNRS, CRAN, F-54000 Nancy, France (e-mail: pietro.lorenzetti@univ-lorraine.fr). F. Reissner and G. Weiss are with the School of Electrical Engineering, Tel Aviv University, Ramat Aviv, Israel (e-mail: reissner@tauex.tau.ac.il, gweiss@tauex.tau.ac.il).

the behavior of synchronous generators (SG). Their control algorithm implements several SG-related features (e.g., droop and inertia), making them quite popular for power systems applications [1], [3], [10], [12], [41], [44]. A detailed stability analysis of a grid-connected synchronverter can be found in [6], [34] for constant excitation current, and in [27] for variable excitation current. An important step in the evolution of the synchronverter algorithm is in [46], where it is shown that if the synchronverter is not injecting any active or reactive power into the electric grid, its internal synchronous voltage is able to synchronize with the grid voltage (in phase, frequency, and amplitude), without the need for a dedicated synchronization unit. Thus, the synchronverter is by itself a three-phase MPLL (with a narrow pull-in range). When it injects power, the synchronverter still synchronizes, but the amplitude and phase of the synchronous internal voltage will, in general, be different from the amplitude and phase of the grid voltage.

In this paper, we propose a (single phase) MPLL formed by an OSG combined with a modified fourth order synchronverter model S, adapted from [24], [27], [34], [35], [47]. Further, we design "jumping" subsystems that bring the MPLL frequency and amplitude close to the dominant input signal frequency and amplitude, dramatically enlarging the pull-in range of the proposed MPLL. These blocks are connected as shown in Fig. 2. As remarked in [45], the main intuition behind the idea of using a synchronverter as a MPLL, is to regard the reference signal as the grid voltage, and the MPLL output as the synchronous internal voltage. Then, thanks to the self-synchronizing property of the synchronverter model, the MPLL output will track the dominant sinusoidal component of the reference. Contrary to usual grid-connected synchronverters, here the reference is single-phase, so that an OSG is needed to generate its dq components.

Our design is advantageous compared to others in the following aspects: rapid convergence to synchronism (thanks to "jumping features" described in Subsect. III-C), wide pull-in range, and strong noise immunity. The *pull-in range* of a PLL (also called capture range) is the maximum deviation of the initial frequency guess $\omega(0)$ from the dominant reference frequency ω_R such that for any initial phase $\theta(0)$ and initial amplitude of the PLL output signal, in the absence of noise, the PLL will synchronize its output to the reference signal.

Results related to this work were presented in our conference papers [28], [29], where a nonlinear internal model, based on our MPLL, was used to solve the reference tracking problem for a stable linear and uncertain plant $\mathbf{P}$. In [29], a rudimentary OSG has been employed and the "jumping" feature (see Subsect. III-C) was not yet developed. For these reasons, in [29] tracking is ensured only for ω_R in a narrow range. In [28] the pull-in range is dramatically increased, thanks to the OSG algorithm from Subsect. III-A and a rudimentary version of the "frequency jumping" feature from Subsect. III-C. (Amplitude jumping was not considered in [28].) However, no stability analysis is provided in [28], [29]. Here, we show that by taking $\mathbf{P} = 1$, a robust MPLL with a very large pull-in range can be obtained. Under (reasonable) simplifying assumptions, we perform a stability analysis of the proposed (nonlinear) MPLL, using results on the pendulum equation, together with tools from singular perturbation theory. Furthermore, we present a frequency-adaptive tuning, which guarantees (at least in theory) the convergence of the estimate for any possible reference frequency. (More details on the connections between this work and [28], [29] are in Sect. VI.)

The paper is organized as follows. In Sect. II we recall the third order self-synchronizing synchronverter model. In Sect. III we describe the architecture of the proposed MPLL. In Sect. IV, under a series of simplifying assumptions, we rewrite the closed-loop system as an interconnection of a pendulum-type "fast" system with a "slow" system comprising an integrator and a low-pass filter, as in singular perturbation theory [22], to study the closed-loop system stability. In Sect. V we explain how to adaptively tune the controller parameters. In Sect. VI we show how to connect multiple copies of the proposed MPLL to track several components of the input signal. Moreover, we extend the stability analysis of the MPLL to the internal model-based controller from [28]. Finally, in Sect. VII we present extensive simulation results to show the performance of the proposed MPLL.

II. A Third Order Synchronverter Model

We briefly present here a third order self-synchronizing synchronverter model that is obtained by simplifying (eliminating two fast state variables) the fifth order model in [27], [34], [35], [46], [47]. This third order synchronverter model is the core of the proposed MPLL. As already mentioned, a synchronverter is an inverter controlled such that it behaves like an SG. Thus, its mathematical model resembles that of an SG, but the variables (rotor inertia, rotor angle, rotor angular velocity, excitation flux) are virtual. We will briefly explain here how to derive a (simplified) synchronverter model.

Let us denote by θ the virtual rotor angle and by $\omega = \dot{\theta}$ its angular velocity (these are state variables of our model). We introduce the (unitary) *Park transformation*, see, e.g., [34],

$$U(\theta) = \sqrt{\frac{2}{3}} \begin{bmatrix} \cos\theta & \cos\left(\theta - \frac{2\pi}{3}\right) & \cos\left(\theta + \frac{2\pi}{3}\right) \\ -\sin\theta & -\sin\left(\theta - \frac{2\pi}{3}\right) & -\sin\left(\theta + \frac{2\pi}{3}\right) \\ 1/\sqrt{2} & 1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix}. \quad (1)$$

Given a vector $v \in \mathbb{R}^3$, e.g., a three-phase grid voltage, the first two components of $U(\theta)v$ are called the dq coordinates of v, denoted, respectively, by v_d, v_q. For all the signals v that we consider, the last component of $U(\theta)v$ is zero. By taking $\theta = 0$ and selecting the first two components of $U(0)v$, we obtain the α, β components of v:

$$v_\alpha = \sqrt{\frac{2}{3}} \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \end{bmatrix} v, \;\; v_\beta = \sqrt{\frac{2}{3}} \begin{bmatrix} 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} v. \quad (2)$$

We denote by m the *excitation flux* (the rotor field) and by E the *synchronous internal voltage*, whose dq components are

$$E_d = 0, \qquad E_q = -m\omega,$$

see, for instance, [34, eq. (2.5)]. Let v_d, v_q be the dq coordinates of the grid voltage v, and i_d, i_q be the dq coordinates of the stator current i. Then, assuming purely inductive impedance $j\omega L$, $j = \sqrt{-1}$, for the stator windings,

$$(E_d + j\,E_q) - (v_d + j\,v_q) = j\omega L(i_d + j\,i_q).$$

The above is obtained from [34, eq. (2.3)], with $R_s = 0$, by setting $\dot{i}_d = 0, \dot{i}_q = 0$.

Let T_e denote the electric torque acting on the rotor due to the stator currents. From conservation of energy considerations, it follows that

$$T_e = -mi_q.$$

From Newton's second law, we obtain the *swing equation*

$$J\dot{\omega} = T_m - T_e - D_p(\omega - \omega_{\text{nom}}),$$

where $J > 0$ is the *virtual inertia* of the rotor, $T_m > 0$ is the nominal active mechanical torque from the prime mover, ω_{nom} is the *nominal grid frequency*, and $D_p > 0$ is the *frequency droop constant*. Let $Q = v_q i_d - v_d i_q$ denote the *reactive power* flowing to the grid, then m is regulated by

$$\dot{m} = k(Q_{\text{set}} - Q),$$

where Q_{set} is the reactive power reference and $k > 0$. (This corresponds to (11) in [24], with $D_q = 0$.)

Combining the above equations with the "self-synchronizing conditions" introduced in [46] (see also [29, Sect. III]), i.e., $T_m = 0$ and $Q_{\text{set}} = 0$, we obtained the reduced self-synchronizing synchronverter model:

$$\dot{\theta} = \omega, \tag{3a}$$
$$Q = v_q i_d - v_d i_q, \tag{3b}$$
$$\dot{m} = -kQ, \tag{3c}$$
$$J\dot{\omega} = mi_q - D_p(\omega - \omega_{\text{nom}}), \tag{3d}$$
$$\omega L i_d = -m\omega - v_q, \qquad \omega L i_q = v_d. \tag{3e}$$

This is not yet the internal model **S** from Fig. 2. **S** will be introduced in Sect. III by modifying (3) to suit our needs.

Remark 2.1: The reduced self-synchronizing synchronverter model (3) is based on the models from [27], [34], [47] with $T_m = 0$ and $Q_{set} = 0$. The main differences are:

a) The algebraic computations of the stator currents i_d, i_q using (3e), which, unlike in [27], [34], [35], [47], are not treated as states here. For this reason, we regard (3) as a *reduced* (third order) synchronverter model.
b) We have set the stator resistance to zero and the voltage droop coefficient also to zero. We have eliminated all the safety limitations, as they are not needed in our context.

According to our experience (acquired through extensive simulations), using a reduced model in place of a "full order model" does not affect the performance of the proposed MPLL. Moreover, it greatly simplifies the stability analysis of the closed-loop system in Fig. 2, presented in Sect. IV.

III. The Architecture of the Proposed MPLL

The intuition of regarding (3) as an internal model stems from the following observation. Assume that the synchronverter described in (3) is connected to the electric grid with a fundamental frequency (the grid frequency) ω_R. Then, given an initial guess $\omega(0)$ (usually $\omega(0) = \omega_{\text{nom}}$) "sufficiently close" to ω_R, *the synchronous internal voltage E will (after a synchronization process) follow in amplitude, frequency and phase the fundamental component of the grid voltage v*. Our idea is to exploit this synchronization property, by replacing

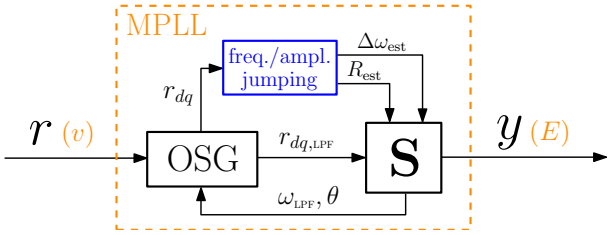


Fig. 2: The proposed MPLL formed by an OSG, a synchronverter model **S**, and a frequency (and amplitude) jumping block. The signal r is the reference. The orange letters refer to the analogy with the synchronverter signals: E is the synchronous internal voltage, and v is the grid voltage.

the grid voltage v with the reference signal r and by regarding the synchronverter internal voltage E as the output signal of an MPLL. In the following, we describe in detail the building blocks of the proposed MPLL from Fig. 2.

A. The OSG algorithm

In a usual three-phase power grid application, the self-synchronizing synchronverter algorithm receives (among its inputs) v_d and v_q, which can be instantaneously computed from v using the Park transformation (1). This cannot be done here, since r is a single-phase (scalar) signal. This is a common problem in single-phase power systems applications, and it can be solved by means of an OSG. Several OSG-implementations can be found in the literature [4], [5], [11], [13], [16], [42], [43]. Among them, we have opted for the one shown in Fig. 3, which resembles the OSG based on the second order generalized integrator proposed in [11]. (Strictly speaking, the term OSG is used in the literature to denote the transformation from single-phase to $\alpha\beta$ components. For convenience, we have included here also the $\alpha\beta$ to dq transformation (9) and the output filter in our OSG.)

Consider a reference signal r given by

$$r(t) = R\sin(\omega_R t + \gamma_R) + N\sin(\omega_N t + \gamma_N), \tag{4}$$

where $R, N, \omega_R, \omega_N > 0$ and $\gamma_R, \gamma_N \in [-\pi, \pi)$. We think of the larger sinusoid with frequency ω_R as the signal to be tracked, while the other component of r represents the noise. For the "jumping" subsystem to be able to work correctly, we will assume that the noise is "small" in the following sense:

$$\frac{N}{R} < \frac{\omega_N}{\max\{\omega_R, \omega_N\}}. \tag{5}$$

Denote $\theta_R = \omega_R t + \gamma_R$ and $\theta_N = \omega_N t + \gamma_N$. An ideal OSG should provide the following $\alpha\beta$ components of r:

$$r_\alpha(t) = R\sin(\theta_R) + N\sin(\theta_N), \tag{6a}$$
$$r_\beta(t) = -R\cos(\theta_R) - N\cos(\theta_N). \tag{6b}$$

These correspond to applying (2) to the three-phase signal

$$\tilde{r} = \sqrt{\frac{2}{3}}R\begin{bmatrix}\sin(\theta_R)\\ \sin(\theta_R - \frac{2\pi}{3})\\ \sin(\theta_R + \frac{2\pi}{3})\end{bmatrix} + \sqrt{\frac{2}{3}}N\begin{bmatrix}\sin(\theta_N)\\ \sin(\theta_N - \frac{2\pi}{3})\\ \sin(\theta_N + \frac{2\pi}{3})\end{bmatrix}.$$

While obtaining r_α is trivial, obtaining the correct r_β for any ω_R, ω_N would require using the Hilbert transform, which

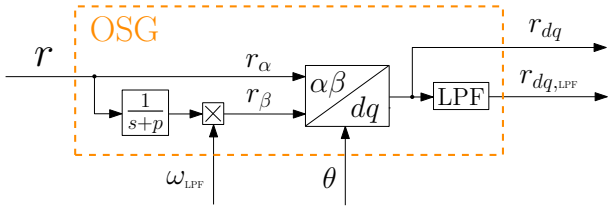


Fig. 3: The block diagram of the OSG from Fig. 2.

is not causal [40, p. 191]. Thus, we compromise for a good approximation of r_β using the procedure described below.

With reference to Fig. 3, the following happens.

1) r_β is obtained by using a quasi-integrator with transfer function $\frac{1}{s+p}$, with $p > 0$ small compared to ω_R. The quasi-integrator introduces a phase shift of approximately $-\frac{\pi}{2}$, and the stable pole $-p$ prevents the presence of unwanted DC-bias in r_β. (A pure integrator could introduce DC-bias that would not decay.) A further scaling of r_β is needed to ensure that the amplitudes of r_α and r_β are equal. This is done by multiplying the output of the filter $\frac{1}{s+p}$ with ω_{LPF}, which is our current estimate for ω_R (see Subsect. III-B). This results in

$$r_\beta(t) \approx -\frac{\omega_{\text{LPF}}}{\omega_R} R\cos(\theta_R) - \frac{\omega_{\text{LPF}}}{\omega_N} N\cos(\theta_N). \quad (7)$$

If $\omega_{\text{LPF}} \approx \omega_R$, which is a reasonable assumption when the "jumping" subsystem in Subsect. III-C has already performed its duty (a short time after starting the system), then the first component in (7) is correct with a good approximation, while the second component (the noise) is wrongly scaled, but we shall see that this is usually not a problem. Thus, after the "jumping" subsystem has obtained a good approximation of ω_R and R, we will often assume (based on (7)) that

$$r_\beta(t) = -R\cos(\theta_R) - \frac{\omega_{\text{LPF}}}{\omega_N} N\cos(\theta_N). \quad (8)$$

2) The dq components of r, i.e., the vector $r_{dq} = [r_d, r_q]^\top$ is obtained from $r_{\alpha\beta} = [r_\alpha\, r_\beta]^\top$ employing the standard $\alpha\beta$ to dq transformation:

$$\begin{bmatrix} r_d \\ r_q \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} r_\alpha \\ r_\beta \end{bmatrix}. \quad (9)$$

We introduce the angles

$$\delta := \theta - \theta_R, \quad \delta_N := \theta - \theta_N \pmod{2\pi},$$

so that

$$\dot\delta = \omega - \omega_R \qquad \dot\delta_N = \omega - \omega_N.$$

Applying (9) to the signals $r_\alpha = r$ and r_β from (8) yields, after a routine computation,

$$r_d = -R\sin\delta - \frac{N}{2}\left(1 + \frac{\omega_{\text{LPF}}}{\omega_N}\right)\sin\delta_N + \frac{N}{2}\left(1 - \frac{\omega_{\text{LPF}}}{\omega_N}\right)\sin(\theta + \theta_N), \quad (10)$$

$$r_q = -R\cos\delta - \frac{N}{2}\left(1 + \frac{\omega_{\text{LPF}}}{\omega_N}\right)\cos\delta_N + \frac{N}{2}\left(1 - \frac{\omega_{\text{LPF}}}{\omega_N}\right)\cos(\theta + \theta_N). \quad (11)$$

3) The above signal $r_{dq} = \left[\begin{smallmatrix} r_d \\ r_q \end{smallmatrix}\right]$ is low-pass filtered to remove undesired oscillations, as shown in Fig. 3, obtaining $r_{dq,\text{LPF}} = \left[\begin{smallmatrix} r_{d,\text{LPF}} \\ r_{q,\text{LPF}} \end{smallmatrix}\right]$. The idea is that the first terms in the above expressions for r_d and r_q (which depend on δ) are very low frequency terms, practically constant, because the "jumping" subsystem ensures that, after a short initial transient, $\dot\delta = \omega - \omega_R$ is very small (when compared to ω_R). The other terms in (10), (11) oscillate at the frequencies $\omega - \omega_N$ and $\omega + \omega_N$, so that the LPF (which has a low corner frequency when compared to ω_R) will attenuate them (unless $\omega - \omega_N$ is very small). Thus, (10) and (11) imply that

$$r_{d,\text{LPF}} = -R\sin\delta + n_d, \quad r_{q,\text{LPF}} = -R\cos\delta + n_q, \quad (12)$$

where n_d and n_q are small noise signals containing the frequencies $|\omega - \omega_N|$ and $\omega + \omega_N$. Here, by "small" we mean that $|n_d|$ and $|n_q|$ are much smaller than R.

Our simulation results in Sect. VII confirm the effectiveness of the OSG shown in Fig. 3.

B. The synchronverter-based internal model S

Using the notation introduced above, we adjust the reduced synchronverter model (3) to suit our needs. For this, we define

$$x_\rho^{\frac{1}{2}} = \frac{x}{\sqrt[4]{x^2 + \rho^2}}, \quad (13)$$

where $\rho \geq 0$. When $\rho > 0$ is "sufficiently small", the above function is a good approximation of the function

$$x_0^{\frac{1}{2}} = \frac{x}{\sqrt{|x|}}.$$

With this notation, the internal model **S** is given by

$$\dot\theta = \omega, \quad (14a)$$

$$Q = r_{q,\text{LPF}} i_d - r_{d,\text{LPF}} i_q, \quad (14b)$$

$$\dot m = -kQ_\rho^{\frac{1}{2}}, \quad (14c)$$

$$J\dot\omega = mi_q - D_p(\omega - \omega_{\text{LPF}}), \quad (14d)$$

$$\omega_{\text{LPF}} L i_d = -m\omega - r_{q,\text{LPF}}, \qquad \omega_{\text{LPF}} L i_q = r_{d,\text{LPF}} \quad (14e)$$

$$\tau\dot\omega_{\text{LPF}} = -\omega_{\text{LPF}} + \omega, \quad (14f)$$

with $\rho > 0$. The inputs to this model are $r_{d,\text{LPF}}$ and $r_{q,\text{LPF}}$, while k, J, D_p are positive parameters. Note that we have added the last equation (14f), with $\tau > 0$, to describe a very simple low-pass filter (LPF) used to generate ω_{LPF} from ω.

Remark 3.1: In principle, we would like to use $Q_0^{\frac{1}{2}}$ in (14c). However, this function has infinite derivative at $Q = 0$. This lack of smoothness causes trouble in the stability analysis in Sect. IV. By contrast, $x_\rho^{\frac{1}{2}}$ is infinitely differentiable for $\rho > 0$. According to our experience, ρ should be chosen a few orders of magnitude less than $\frac{R^2}{\omega_R L}$ and then our system works just as for $\rho = 0$. In our simulations in Sect. VII we have chosen

$$\rho = 10^{-3}\frac{R^2}{\omega_R L}.$$

We define the output y of our MPLL to be the following signal coming from **S**:

$$y = m\omega\sin\theta. \quad (15)$$

This output should track the dominant sinusoidal component of r. The block diagram of $\mathbf{S}$ is shown in Fig. 4.

We now explain the modifications that led from (3) to (14), keeping in mind that the model (3) is meant for a synchronverter working under the following assumptions: the amplitude of v from Sect. II (which corresponds to R here) is in a certain known range, and $\omega_R \approx \omega_{\text{nom}}$. This is not the case in our MPLL, where (at least in principle) R can be of any order, there is no nominal frequency, and $\omega(0) = \omega_{\text{LPF}}(0)$ (i.e., our initial guess on ω_R) can be rather far from ω_R.

The intuition behind (14c) is as follows: in steady state m is proportional to R. To ensure that our MPLL can work with a wide range of possible values R, we would like that in a transient, after a sudden change of R, $\dot{m}$ should also depend linearly on R. In other words, regardless of the amplitude of R, we want the transition of m to a new steady state value to take approximately the same amount of time. Since Q depends quadratically on R, this can be achieved by replacing (3c) with (14c). We have noticed that this modification allows our MPLL to work correctly even when R changes several orders of magnitude.

The effect of ω_{nom} in the swing equation (3d) is eliminated by high-pass filtering the term $\omega - \omega_{\text{nom}}$. This results in (14d), which can be written also as $J\dot{\omega} = mi_q - D_p\omega_{\text{HPF}}$, where $\omega_{\text{HPF}} = \omega - \omega_{\text{LPF}}$, see the block diagram in Fig. 4. Thus, the stabilizing effect of the droop is still present during transients, but it disappears at steady-state since $\omega = \omega_{\text{LPF}}$ there.

Finally, (14e) has been modified from (3e) by replacing ω with the filtered ω_{LPF}, which yields better results in simulations.

C. The "jumping" subsystems

To considerably enlarge the pull-in range of our MPLL, we have added two subsystems: the "frequency jumping" and the "amplitude jumping". The inputs of these subsystems are r_d and r_q, while the outputs are the frequency correction estimate $\Delta\omega_{\text{est}}$ and the amplitude estimate R_{est}.

1) The frequency jumping: By periodically estimating the frequency deviation $\omega_R - \omega$ using the signals r_d and r_q, we *instantaneously* update the controller internal frequency ω if $\omega_R - \omega$ is large. The details:

Using (6) and (7), we define the vector

$$r_{\alpha\beta}^{\text{mod}} = \begin{bmatrix} r_\alpha \\ \frac{\omega_R}{\omega_{\text{LPF}}} r_\beta \end{bmatrix} = r_{\alpha\beta,R}^{\text{mod}} + r_{\alpha\beta,N}^{\text{mod}}, \tag{16}$$

where $r_{\alpha\beta,R}^{\text{mod}}$ is the dominant component and $r_{\alpha\beta,N}^{\text{mod}}$ is the noise component. The component $r_{\alpha\beta,R}^{\text{mod}} = R\begin{bmatrix}\sin(\theta_R) & -\cos(\theta_R)\end{bmatrix}^\top$ rotates (counter-clockwise) on a circle with radius R, with angular velocity ω_R. We have

$$r_{\alpha\beta,N}^{\text{mod}} = N\begin{bmatrix}\sin(\theta_N) & -\frac{\omega_R}{\omega_N}\cos(\theta_N)\end{bmatrix}^\top.$$

If the assumption (5) holds, then

$$\begin{aligned}\|r_{\alpha\beta,N}^{\text{mod}}\| &= N\sqrt{\sin^2(\theta_N) + \frac{\omega_R^2}{\omega_N^2}\cos^2(\theta_N)} \\ &\le N\max\left\{1, \frac{\omega_R}{\omega_N}\right\} < R. \end{aligned}\tag{17}$$

We introduce the vector signals r_{dq}^{mod}, $r_{dq,R}^{\text{mod}}$ and $r_{dq,N}^{\text{mod}}$ by applying the $\alpha\beta$ to dq transformation from (9) to the vectors $r_{\alpha\beta}^{\text{mod}}$, $r_{\alpha\beta,R}^{\text{mod}}$ and $r_{\alpha\beta,N}^{\text{mod}}$, respectively. Thus,

$$r_{dq}^{\text{mod}} = r_{dq,R}^{\text{mod}} + r_{dq,N}^{\text{mod}}.$$

The norms of these vectors are not affected by the $\alpha\beta$ to dq transformation, so that from (17)

$$\|r_{dq,N}^{\text{mod}}\| < \|r_{dq,R}^{\text{mod}}\| = R.$$

It follows from here that the angles of the vectors r_{dq}^{mod} and $r_{dq,R}^{\text{mod}}$ differ by less than 90°:

$$\left|\arg r_{dq}^{\text{mod}} - \arg r_{dq,R}^{\text{mod}}\right| < \frac{\pi}{2}. \tag{18}$$

By simple geometric arguments it can be shown that the scaling used in (16) to define $r_{\alpha\beta}^{\text{mod}}$ will change the angle of $r_{\alpha\beta}^{\text{mod}}$ by at most 90°:

$$\left|\arg r_{\alpha\beta} - \arg r_{\alpha\beta}^{\text{mod}}\right| < \frac{\pi}{2}. \tag{19}$$

This difference will be preserved when applying the $\alpha\beta$ to dq transformation (9). Combining (18) and the dq version of (19) we obtain that

$$\left|\arg r_{dq} - \arg r_{dq,R}^{\text{mod}}\right| < \pi. \tag{20}$$

In Fig. 5 the vector $r_{dq,R}^{\text{mod}}$ is represented. The dq frame is rotating at the angular velocity $-\omega$ with respect to the $\alpha\beta$ frame, see (9). $r_{\alpha\beta,R}^{\text{mod}}$ rotates (counter-clockwise) at an angular velocity ω_R. As a consequence, $r_{dq,R}^{\text{mod}}$ has angular velocity $\Delta\omega = \omega_R - \omega$. We want to estimate this $\Delta\omega$, but the signal $r_{dq,R}^{\text{mod}}$ is not available, only the noisy and unscaled signal r_{dq} is available. We claim that this is almost as good as having $r_{dq,R}^{\text{mod}}$, if we use the right algorithm.

Let us denote by $n_{\text{cross},R}^{\text{mod}}$ the number of times that $r_{dq,R}^{\text{mod}}$ crosses the d or q axes in a given interval of time T_{jump} (to be chosen later). We count each counter-clockwise crossing as $+1$, and each clockwise crossing as -1. (This is like four times the winding number of the plot of r_{dq}^{mod} around zero in the time period T_{jump}.) We have that

$$\frac{1}{4}n_{\text{cross},R}^{\text{mod}} \approx \frac{\Delta\omega}{2\pi}T_{\text{jump}}, \tag{21}$$

with an approximation error of at most $\frac{1}{4}$. But we cannot count $n_{\text{cross},R}^{\text{mod}}$, because the signal $r_{dq,R}^{\text{mod}}$ is not available, so we have to work with n_{cross}, which is the number of times that r_{dq} crosses the d or q axes. Due to (20) we have $\left|n_{\text{cross}} - n_{\text{cross},R}^{\text{mod}}\right| < 2$. Combining this with (21) we get that $\frac{1}{4}n_{\text{cross}} \approx \frac{\Delta\omega}{2\pi}T_{\text{jump}}$, with an approximation error of at most $\frac{3}{4}$. From here, the difference $\Delta\omega$ can be estimated by

$$\Delta\omega_{\text{est}} = n_{\text{cross}}\frac{\pi}{2T_{\text{jump}}}.$$

According to the earlier stated error estimates, $|\Delta\omega_{\text{est}} - \Delta\omega| < \frac{\pi}{T_{\text{jump}}}$. To make this error small, we choose T_{jump} to be much larger than our current estimate for the period of r, i.e., $T_{\text{jump}} >> 2\pi/\omega$.

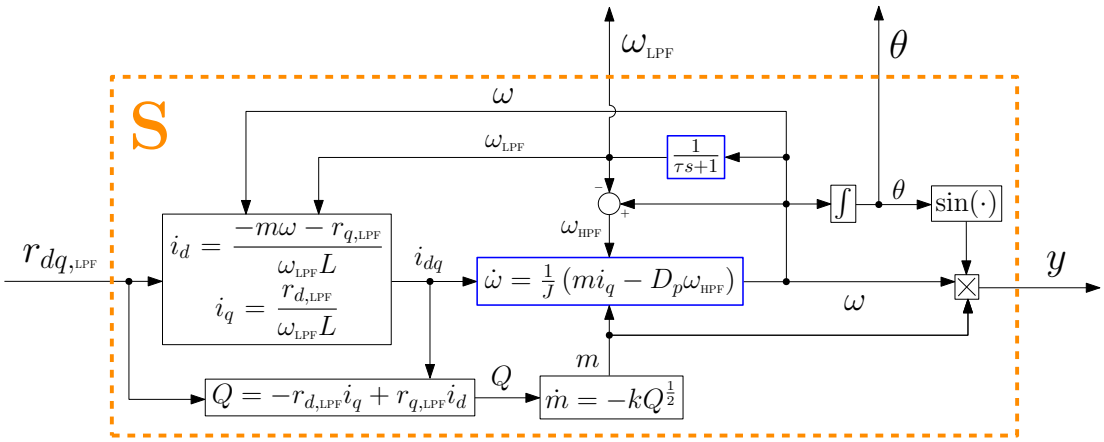


Fig. 4: The detailed block diagram of the internal model **S** from Fig. 2, given by (14), (15). The blocks with blue frame can have their state reset to a new value by the jumping subsystems through the inputs $\Delta\omega_{\text{est}}$ and R_{est} (not shown here).

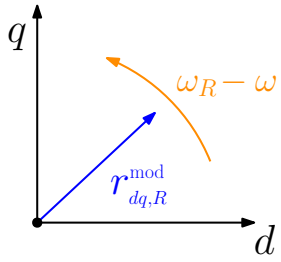


Fig. 5: The vector $r^{\text{mod}}_{dq,R}$ rotates on the circle of radius R, with angular velocity $\omega_R - \omega$. The vector r^{mod}_{dq} describes a complicated trajectory, but the number of times it crosses the positive d axis is almost the same as for $r^{\text{mod}}_{dq,R}$.

If $|\Delta\omega_{\text{est}}|$ is larger than a certain threshold $\varepsilon > 0$, then the MPLL internal frequency ω and the LPF state ω_{LPF} (blocks with a blue frame in Fig. 4) are updated according to

$$[\omega]_{new} = \omega + \Delta\omega_{\text{est}}, \qquad [\omega_{\text{LPF}}]_{new} = \omega_{\text{LPF}} + \Delta\omega_{\text{est}}. \quad (22)$$

This update takes place after each time interval of length T_{jump} (unless $|\Delta\omega_{\text{est}}| < \varepsilon$). In addition, the internal frequency ω also jumps if no jump has occurred for at least 5s and $|n^{\text{mod}}_{\text{cross}}| > 10$. This second condition, resulting from experience acquired through simulations, helps the system to recover faster when, as a result of an earlier jump, ω is driven to low frequencies. As shown in our simulations, the "jumping" subsystem increases tremendously the pull-in range of our MPLL. (In our experience, after at most three jumps, with $\varepsilon = 0.01\omega_R$, no more jumps are needed.) We always assume that $\varepsilon << \omega_R$.

Remark 3.2: Our "jumping" subsystem is remotely related to the discrete-time identifier in [8]. This identifier consists of a linear regressor, which, at any jump time, updates the internal model parameters, depending on the new samples.

2) The amplitude jumping: Using r_d and r_q, the amplitude of the reference signal r can be estimated as

$$R_{\text{est}} = LPF\left(\sqrt{r_d^2 + r_q^2}\right), \quad (23)$$

where $LPF(\cdot)$ denotes a first order low-pass filter with time constant τ_r. In order to decrease the influence of noise on the OSG tuning, an amplitude jump is performed if $R_{\text{est}}/(m\omega) > 1.3$ or $R_{\text{est}}/(m\omega) < 0.75$, i.e., if the estimated amplitude from (23) is more than 130% or less than 75% of the amplitude of y. When a jump is executed, m is updated as

$$m = \frac{R_{\text{est}}}{\omega}. \quad (24)$$

The parameters k and L are tuned according to (49) in Sect. V.

IV. MPLL Stability Analysis

We assume that the "frequency jumping" subsystem has achieved its aim of bringing ω and ω_{LPF} into an ε-neighborhood of ω_R, and we shall do the stability analysis for what happens afterwards. Inspired by singular perturbation theory, we will decompose the closed-loop system from Fig. 2 (ignoring the "jumping" subsystems) into two subsystems: a fast pendulum-like system, governed by (14a) and the swing equation (14d), and a slow reduced model, containing the integrator (14c) and the LPF (14f), as shown in Fig. 6. The algebraic equations (14b) and (14e) belong to the slow system. As we will discuss later, such a decomposition is justified by choosing τ large enough in (14f) and k sufficiently small in (14c). Given the high complexity involved in the model, e.g., the non-linearity of **S** and of the OSG, we will work under some (reasonable) simplifying assumptions.

A. Derivation of a simplified model for our MPLL

We derive here a simplified model for the MPLL shown in Fig. 2. This model will then be used for the stability analysis.

Assumption 1: The reference signal r is composed of two sinusoidal signals, as in (4), satisfying the assumption (5).

Assumption 2: The "frequency jumping" subsystem has achieved its aim, so that for some specified $\varepsilon > 0$,

$$\omega, \omega_{\text{LPF}} \in (\omega_R - \varepsilon, \omega_R + \varepsilon), \quad \varepsilon << \omega_R. \quad (25)$$

Assumption 3: The OSG performs as expected, so that r_d, r_q are given by (10), (11) and, hence, (12) holds.

This assumption implies that we can ignore the dynamics of the OSG from Fig. 3 in our stability analysis.

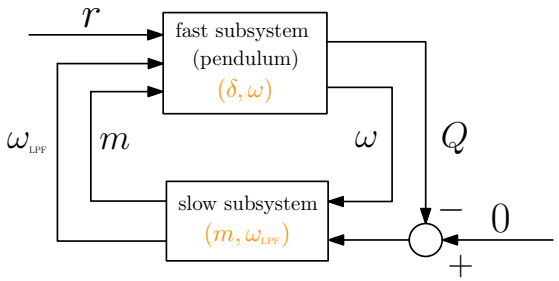


Fig. 6: The decomposition of our MPLL (without frequency and amplitude jumping) into a fast and a slow subsystem.

Using (12), we rewrite i_d and i_q from (14e):

$$\begin{aligned} i_d &= \frac{1}{\omega_{\text{LPF}} L}\left[-\omega m + R\cos\delta - n_q\right], \\ i_q &= \frac{1}{\omega_{\text{LPF}} L}\left[-R\sin\delta + n_d\right]. \end{aligned} \tag{26}$$

Similarly, we can rewrite Q from (14b) as

$$Q = \frac{1}{\omega_{\text{LPF}} L}\left[\omega m R\cos\delta - R^2 + \mathbf{n}_Q\right], \tag{27}$$

where $\mathbf{n}_Q$ is a small noise signal containing the frequencies 0, $|\omega \pm \omega_N|$ and $2|\omega \pm \omega_N|$. To obtain the statement about $\mathbf{n}_Q$, we have assumed that δ is practically constant (frequency zero) and by "small" we mean that $|\mathbf{n}_Q| << R^2$.

Using the above expressions, we can rewrite the internal model $\mathbf{S}$ from (14) as a set of differential equations:

$$\dot{m} = -k\left(\frac{1}{\omega_{\text{LPF}} L}\left[\omega m R\cos\delta - R^2 + \mathbf{n}_Q\right]\right)_\rho^{\frac{1}{2}}, \tag{28a}$$

$$\tau\dot{\omega}_{\text{LPF}} = -\omega_{\text{LPF}} + \omega, \tag{28b}$$

$$\dot{\delta} = \omega - \omega_R, \tag{28c}$$

$$J\dot{\omega} = -\frac{mR}{\omega_{\text{LPF}} L}\sin\delta + \frac{m n_d}{\omega_{\text{LPF}} L} - D_p(\omega - \omega_{\text{LPF}}). \tag{28d}$$

Here, we have used the notation from (13) and the differential equation (28c) is in the tangent space of the unit circle $\mathbb{T}$, since δ is measured modulo 2π. The state of the system (28) is $[m\ \omega_{\text{LPF}}\ \delta\ \omega]^\top$, and the state space is

$$X = (0,\infty)\times(\omega_R-\varepsilon,\omega_R+\varepsilon)\times\mathbb{T}\times(\omega_R-\varepsilon,\omega_R+\varepsilon). \tag{29}$$

Indeed, the frequency jumping feature from Subsection III-C will bring ω and ω_{LPF} into the interval $(\omega_R-\varepsilon,\omega_R+\varepsilon)$. If for all $t \geq t_0$, ω is in this interval and $\omega_{\text{LPF}}(t_0)$ is in this interval, then clearly $\omega_{\text{LPF}}(t)$ cannot escape from it for $t \geq t_0$. If m starts in $(0,\infty)$, then it cannot escape from $(0,\infty)$, because, for very small m, $\dot{m}$ becomes positive (since $|\mathbf{n}_Q| < R^2$).

B. The system (28) as a singular perturbation model

In order to study the stability of the system (28) in the absence of the noise ($N = 0$), we rewrite it as a standard singular perturbation model, according to the guidelines of [21, Chapt. 11], [22, Chapt. 7], [31, App. A], and [32, Sect. III]. The first step is to find the equilibrium points of (28).

Proposition 4.1: If $N = 0$, then the system (28), with state space X from (29), has a unique equilibrium point, given by

$$(m^e, \omega^e_{\text{LPF}}, \delta^e, \omega^e) = \left(\frac{R}{\omega_R}, \omega_R, 0, \omega_R\right). \tag{30}$$

Moreover, at the equilibrium point we have $y = r$, i.e., the MPLL output y is exactly tracking the reference r.

Proof: If $N = 0$, then also $n_d = 0$ and $\mathbf{n}_Q = 0$. It is obvious from (28b), (28c) that at an equilibrium point $(m^e, \omega^e_{\text{LPF}}, \delta^e, \omega^e)$ we must have

$$\omega^e = \omega^e_{\text{LPF}} = \omega_R.$$

Substituting this into (28d), we get that $\sin(\delta^e) = 0$, so that either $\delta^e = 0$ or $\delta^e = \pi$ and, accordingly, $\cos(\delta^e) = 1$ or -1. The second option leads to $\dot{m} > 0$ (according to (28a)), so that it can be eliminated. Thus, we conclude that $\delta^e = 0$.

Finally, from (28a) we see that at equilibrium we have $\omega^e m^e\cos(\delta^e) - R = 0$, whence

$$m^e = \frac{R}{\omega_R}. \tag{31}$$

Summarizing, we have found that for $N = 0$, (28) has a unique equilibrium point in X, given by (30).

The last statement of the proposition follows by comparing (4) and (15), with $m = m^e$, $\delta = \delta^e = 0$, $\omega = \omega^e = \omega_R$. □

Following [21, Chap. 11], [32, Sect. III], we introduce the deviations from equilibrium values:

$$\tilde{m} = m - m^e, \quad \tilde{\omega}_{\text{LPF}} = \omega_{\text{LPF}} - \omega^e, \quad \tilde{\omega} = \omega - \omega^e, \tag{32}$$

the scalars $M > \mu > 0$, and the sets

$$U = \left\{\left[\begin{smallmatrix} m \\ \omega_{\text{LPF}}\end{smallmatrix}\right] \in \mathbb{R}^2 \,|\, \mu \leq m \leq M,\ \omega_R - \varepsilon \leq \omega_{\text{LPF}} \leq \omega_R + \varepsilon\right\},$$

$$\tilde{U} = U - \left[\begin{smallmatrix} m^e \\ \omega^e_{\text{LPF}}\end{smallmatrix}\right] = \left[\mu - \frac{R}{\omega_R}, M - \frac{R}{\omega_R}\right] \times [-\varepsilon, \varepsilon], \tag{33}$$

chosen such that $m^e \in (\mu, M)$, assuming that a rough initial estimate of R and ω_R exists, giving an estimate of m^e based on (31). Then $\left[\begin{smallmatrix} m^e \\ \omega_R\end{smallmatrix}\right] \in U$. We also introduce the functions

$$\tilde{h}\left(\left[\begin{smallmatrix} \tilde{m} \\ \tilde{\omega}_{\text{LPF}}\end{smallmatrix}\right], \left[\begin{smallmatrix} \delta \\ \tilde{\omega}\end{smallmatrix}\right]\right) = \left[\begin{smallmatrix} -\left(\frac{R}{(\tilde{\omega}_{\text{LPF}}+\omega_R)L}\left[(\tilde{\omega}+\omega_R)\left(\tilde{m}+\frac{R}{\omega_R}\right)\cos\delta - R\right]\right)_\rho^{\frac{1}{2}} \\ -\tilde{\omega}_{\text{LPF}} + \tilde{\omega}\end{smallmatrix}\right],$$

$$\tilde{f}\left(\left[\begin{smallmatrix} \tilde{m} \\ \tilde{\omega}_{\text{LPF}}\end{smallmatrix}\right], \left[\begin{smallmatrix} \delta \\ \tilde{\omega}\end{smallmatrix}\right]\right) = \left[\begin{smallmatrix} \tilde{\omega} \\ -\frac{D_p}{J}(\tilde{\omega} - \tilde{\omega}_{\text{LPF}}) - \frac{R\left(\tilde{m}+\frac{R}{\omega_R}\right)}{J(\tilde{\omega}_{\text{LPF}}+\omega_R)L}\sin\delta\end{smallmatrix}\right],$$

with the function $x_\rho^{\frac{1}{2}}$ as in (13).

Assumption 4: We assume that $k, \frac{1}{\tau}$ are small and of the same order of magnitude. In other words, there exists $\gamma > 0$ (independent of k) such that

$$\frac{1}{\tau} = \gamma k.$$

This assumption matters when we study (28) as $k \to 0$.

Under Assumption 4, for $N = 0$, we can rewrite the system (28) using the new "slow" timescale $\upsilon := kt$ as

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}\upsilon}\left[\begin{smallmatrix} \tilde{m} \\ \tilde{\omega}_{\text{LPF}}\end{smallmatrix}\right] &= \left[\begin{smallmatrix} 1 & 0 \\ 0 & \gamma\end{smallmatrix}\right]\tilde{h}\left(\left[\begin{smallmatrix} \tilde{m} \\ \tilde{\omega}_{\text{LPF}}\end{smallmatrix}\right], \left[\begin{smallmatrix} \delta \\ \tilde{\omega}\end{smallmatrix}\right]\right), \\ k\frac{\mathrm{d}}{\mathrm{d}\upsilon}\left[\begin{smallmatrix} \delta \\ \tilde{\omega}\end{smallmatrix}\right] &= \tilde{f}\left(\left[\begin{smallmatrix} \tilde{m} \\ \tilde{\omega}_{\text{LPF}}\end{smallmatrix}\right], \left[\begin{smallmatrix} \delta \\ \tilde{\omega}\end{smallmatrix}\right]\right). \end{aligned} \tag{34}$$

The above is a standard singular perturbation model for small $k > 0$, see [21, Sect. 11.5]. Thus, we consider $\tilde{m}$ and $\tilde{\omega}_{\text{LPF}}$ to be "slow" variables, while δ and $\tilde{\omega}$ to be "fast".

Following [21, Sect. 11.5], [32, Sect. III], we are searching for a function $\tilde{\Xi} \in C^1(\tilde{U}; \mathbb{R}^2)$ such that

$$\tilde{f}\left(\left[\begin{smallmatrix}\tilde{m}\\ \tilde{\omega}_{\text{LPF}}\end{smallmatrix}\right], \tilde{\Xi}\left(\left[\begin{smallmatrix}\tilde{m}\\ \tilde{\omega}_{\text{LPF}}\end{smallmatrix}\right]\right)\right) = 0.$$

Thus, $\tilde{\Xi}$ gives the temporary equilibrium points of the fast variables, as a function of the slow variables considered to be "frozen". It can be easily checked that $\tilde{\Xi}$ must be given by

$$\tilde{\Xi}\left(\begin{bmatrix}\tilde{m}\\ \tilde{\omega}_{\text{LPF}}\end{bmatrix}\right) = \begin{bmatrix}\lambda\\ 0\end{bmatrix}, \qquad \lambda \in \left(-\frac{\pi}{2}, \frac{\pi}{2}\right), \tag{35}$$

where

$$\sin\lambda := \frac{D_p(\tilde{\omega}_{\text{LPF}} + \omega_R)\tilde{\omega}_{\text{LPF}} L}{R\left(\tilde{m} + \frac{R}{\omega_R}\right)} = \frac{D_p \omega_{\text{LPF}} \tilde{\omega}_{\text{LPF}} L}{Rm}. \tag{36}$$

(We have excluded the endpoints $\lambda = \pm\pi/2$, because there Ξ would not be differentiable.) A necessary and sufficient condition for the existence of $\tilde{\Xi}$ is

$$\left|\frac{D_p(\tilde{\omega}_{\text{LPF}} + \omega_R)\tilde{\omega}_{\text{LPF}} L}{R\left(\tilde{m} + \frac{R}{\omega_R}\right)}\right| < 1 \qquad \forall \begin{bmatrix}\tilde{m}\\ \tilde{\omega}_{\text{LPF}}\end{bmatrix} \in \tilde{U}. \tag{37}$$

Remark 4.2: A sufficient condition for (37) to hold is

$$D_p(\omega_R + \varepsilon)\varepsilon L < R\mu, \tag{38}$$

with $\varepsilon > 0$ from Assumption 2.

Assumption 5: We assume that the parameters of our problem satisfy (38). Hence, $\tilde{\Xi} \in C^1(\tilde{U}, \mathbb{R}^2)$ is well-defined.

We define the shifted fast variables

$$\begin{bmatrix}\delta_f\\ \tilde{\omega}_f\end{bmatrix} = \begin{bmatrix}\delta\\ \tilde{\omega}\end{bmatrix} - \tilde{\Xi}\left(\begin{bmatrix}\tilde{m}\\ \tilde{\omega}_{\text{LPF}}\end{bmatrix}\right) = \begin{bmatrix}\delta - \lambda\\ \tilde{\omega}\end{bmatrix}. \tag{39}$$

With this notation, we reformulate (34) (which corresponds to $N = 0$) like [21, eqs. (11.35),(11.36)]:

$$\frac{\mathrm{d}}{\mathrm{d}v}\begin{bmatrix}\tilde{m}\\ \tilde{\omega}_{\text{LPF}}\end{bmatrix} = \begin{bmatrix}1 & 0\\ 0 & \gamma\end{bmatrix}\tilde{h}\left(\begin{bmatrix}\tilde{m}\\ \tilde{\omega}_{\text{LPF}}\end{bmatrix}, \begin{bmatrix}\delta_f\\ \tilde{\omega}_f\end{bmatrix} + \tilde{\Xi}\left(\begin{bmatrix}\tilde{m}\\ \tilde{\omega}_{\text{LPF}}\end{bmatrix}\right)\right), \tag{40a}$$

$$k\frac{\mathrm{d}}{\mathrm{d}v}\begin{bmatrix}\delta_f\\ \tilde{\omega}_f\end{bmatrix} = \tilde{f}\left(\begin{bmatrix}\tilde{m}\\ \tilde{\omega}_{\text{LPF}}\end{bmatrix}, \begin{bmatrix}\delta_f\\ \tilde{\omega}_f\end{bmatrix} + \tilde{\Xi}\left(\begin{bmatrix}\tilde{m}\\ \tilde{\omega}_{\text{LPF}}\end{bmatrix}\right)\right)$$
$$- k\frac{\partial\tilde{\Xi}\left(\begin{bmatrix}\tilde{m}\\ \tilde{\omega}_{\text{LPF}}\end{bmatrix}\right)}{\partial\begin{bmatrix}\tilde{m}\\ \tilde{\omega}_{\text{LPF}}\end{bmatrix}}\begin{bmatrix}1 & 0\\ 0 & \gamma\end{bmatrix}\tilde{h}\left(\begin{bmatrix}\tilde{m}\\ \tilde{\omega}_{\text{LPF}}\end{bmatrix}, \begin{bmatrix}\delta_f\\ \tilde{\omega}_f\end{bmatrix} + \tilde{\Xi}\left(\begin{bmatrix}\tilde{m}\\ \tilde{\omega}_{\text{LPF}}\end{bmatrix}\right)\right), \tag{40b}$$

which has an equilibrium point in $\left(\left[\begin{smallmatrix}0\\ 0\end{smallmatrix}\right], \left[\begin{smallmatrix}0\\ 0\end{smallmatrix}\right]\right)$.

Using standard arguments, see [21, Ch. 11], [22, Ch. 7] or [31, Sect. III], we identify the reduced model and the boundary-layer system associated to (40). Since $\tilde{\omega}_f = \tilde{\omega}$, we will stop using the notation $\tilde{\omega}_f$. The *reduced (slow) system* is obtained by taking $\left[\begin{smallmatrix}\delta_f\\ \tilde{\omega}\end{smallmatrix}\right] = \left[\begin{smallmatrix}0\\ 0\end{smallmatrix}\right]$ in (40a), yielding

$$\frac{\mathrm{d}}{\mathrm{d}v}\begin{bmatrix}\tilde{m}\\ \tilde{\omega}_{\text{LPF}}\end{bmatrix} = \begin{bmatrix}-\left(\frac{R}{(\tilde{\omega}_{\text{LPF}} + \omega_R)L}\left[\omega_R\left(\tilde{m} + \frac{R}{\omega_R}\right)\cos\lambda - R\right]\right)_\rho^{\frac{1}{2}}\\ -\gamma\tilde{\omega}_{\text{LPF}}\end{bmatrix}, \tag{41}$$

where λ is as in (36). The state $[\tilde{m}\ \ \tilde{\omega}_{\text{LPF}}]^\top$ of the reduced model is in the set $\tilde{U}$ defined in (33).

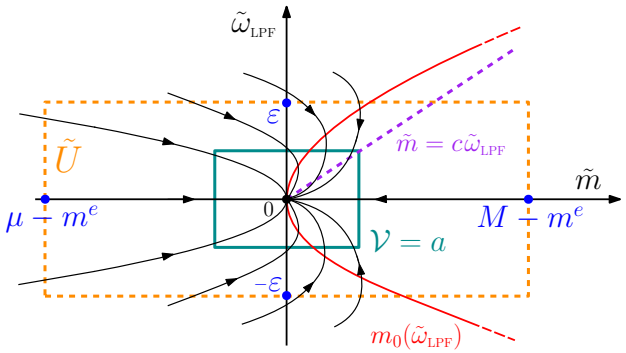


Fig. 7: Graphical representation of the proof of Prop. 4.3. The green rectangle is one where $\mathcal{V}$ is constant. Its right edge is to the right of the curve $\tilde{m} = m_0(\tilde{\omega}_{\text{LPF}})$ and its left edge is to the left of the curve. The black lines are trajectories of (41).

The *boundary-layer (fast) system* associated to (40) is obtained by rewriting (40b) in the original fast time scale t and taking $k = 0$, which yields

$$\frac{\mathrm{d}}{\mathrm{d}t}\begin{bmatrix}\delta_f\\ \tilde{\omega}\end{bmatrix} = \begin{bmatrix}\tilde{\omega}\\ -\frac{D_p}{J}(\tilde{\omega} - \tilde{\omega}_{\text{LPF}}) - \frac{R\left(\tilde{m} + \frac{R}{\omega_R}\right)}{J(\tilde{\omega}_{\text{LPF}} + \omega_R)L}\sin(\delta_f + \lambda)\end{bmatrix}, \tag{42}$$

where $\left[\begin{smallmatrix}\tilde{m}\\ \tilde{\omega}_{\text{LPF}}\end{smallmatrix}\right] \in \tilde{U}$ are treated as fixed parameters.

C. Stability analysis of the reduced (slow) model (41).

According to the material in the previous subsection, the state of the reduced model (41) is in $\tilde{U} \subset \mathbb{R}^2$, as defined in (33), and the differential equation of $\tilde{m}$ is

$$\frac{\mathrm{d}\tilde{m}}{\mathrm{d}v} = -\left(\frac{R}{(\tilde{\omega}_{\text{LPF}} + \omega_R)L}\left[\omega_R(\tilde{m} + m^e)\cos\lambda - R\right]\right)_\rho^{\frac{1}{2}},$$

where λ is a function of $\tilde{m}$ and $\tilde{\omega}_{\text{LPF}}$, see (36).

Proposition 4.3: Under Assumption 5, the reduced model (41) is globally asymptotically stable (GAS) on its state space $\tilde{U}$ from (33).

Proof: A key observation is that $\frac{\mathrm{d}\tilde{\omega}_{\text{LPF}}}{\mathrm{d}v} = -\gamma\tilde{\omega}_{\text{LPF}}$ is a GAS autonomous system, independently of $\tilde{m}$. Thus, for each $\tilde{\omega}_{\text{LPF}} \in [-\varepsilon, \varepsilon]$, we can compute the unique $m_0(\tilde{\omega}_{\text{LPF}}) \in [\mu - m^e, M - m^e]$ at which $\frac{\mathrm{d}\tilde{m}}{\mathrm{d}v}(\tilde{m}, \tilde{\omega}_{\text{LPF}}) = 0$, from

$$(m_0 + m^e)\cos\lambda = m^e. \tag{43}$$

By algebraic manipulations, using (36), this reduces to

$$R^2(m_0 + m^e)^2 - D_p^2(\tilde{\omega}_{\text{LPF}} + \omega_R)^2\tilde{\omega}_{\text{LPF}}^2 L^2 = R^2(m^e)^2,$$

whence

$$m_0(\tilde{\omega}_{\text{LPF}}) = \sqrt{\frac{D_p^2(\tilde{\omega}_{\text{LPF}} + \omega_R)^2 L^2}{R^2}\tilde{\omega}_{\text{LPF}}^2 + (m^e)^2} - m^e.$$

(Notice that $m_0(0) = 0$, as expected, since $(0, 0)$ must be an equilibrium point.) We choose a number $c > 0$ such that

$$c\tilde{\omega}_{\text{LPF}} > m_0(\tilde{\omega}_{\text{LPF}}) \qquad \forall\, \tilde{\omega}_{\text{LPF}} \in (0, \varepsilon].$$

Let us define the Lyapunov-like function

$$\mathcal{V}\left(\left[\begin{smallmatrix}\tilde{m}\\ \tilde{\omega}_{\text{LPF}}\end{smallmatrix}\right]\right) = \max\{|\tilde{m}|, c\,|\tilde{\omega}_{\text{LPF}}|\}.$$

(This is not a Lyapunov function in the usual sense, because it is not continuously differentiable.) For any $a > 0$, the set of states $(\tilde{m}, \tilde{\omega}_{\text{LPF}})$ where $\mathcal{V} \leq a$ is a rectangle centered at zero. It is easy to check that at any boundary point of such a rectangle, the flow is pointing strictly inwards. Indeed, on the top and the bottom edge, this is clear because $\tilde{\omega}_{\text{LPF}}$ is decaying exponentially. The left edge of the rectangle is to the left of the curve m_0 and the right edge is to the right of the curve m_0, see Fig. 7. Hence, $\tilde{m}$ must increase on the left edge, and it must decrease at the right edge. Thus, any trajectory starting on the left or the right edge of the rectangle will enter into the rectangle. It follows that $\frac{\mathrm{d}}{\mathrm{d}v}\mathcal{V} < 0$ everywhere in $\tilde{U}$.

It follows from the above argument that for any initial state of the system (41) in $\tilde{U}$, $\mathcal{V}$ converges to a limit $v_0 \geq 0$. Since $|\tilde{\omega}_{\text{LPF}}| \to 0$, it follows that $|\tilde{m}| \to v_0$. Thus, the state $(\tilde{m}, \tilde{\omega}_{\text{LPF}})$ converges to either $(v_0, 0)$ or to $(-v_0, 0)$. The limit point must be an equilibrium point of (41). This implies that $v_0 = 0$. Thus, the equilibrium state $(0, 0)$ is GAS. □

D. Stability analysis of the boundary-layer system (42).

Recall that $\delta_f = \delta - \lambda$. To study its stability, we rewrite the boundary-layer (fast) system (42) as a pendulum-like equation:

$$\ddot{\delta}_f + \frac{D_p}{J}\dot{\delta}_f + \frac{Rm}{J\omega_{\text{LPF}}L}\sin(\delta_f + \lambda) = \frac{D_p}{J}\tilde{\omega}_{\text{LPF}}, \tag{44}$$

where $\left[\begin{smallmatrix} m \\ \omega_{\text{LPF}} \end{smallmatrix}\right] \in U$ and λ are fixed parameters.

To rewrite the above as a pendulum equation, we introduce the (dimension-less) timescale σ:

$$\sigma := t\sqrt{\frac{Rm}{J\omega_{\text{LPF}}L}}. \tag{45}$$

Then, using the expression (36) for $\sin\lambda$, we can rewrite (42) as a damped pendulum equation given by

$$\frac{\mathrm{d}^2\delta}{\mathrm{d}\sigma^2} + \alpha\frac{\mathrm{d}\delta}{\mathrm{d}\sigma} + \sin\delta = \sin\lambda, \tag{46}$$

where

$$\alpha = D_p\sqrt{\frac{\omega_{\text{LPF}}L}{JRm}}. \tag{47}$$

For a comprehensive study of pendulum-like dynamical systems and their connection to PLLs and synchronous machines, we recommend the book by Reitmann et al. [39].

A dynamical system is said to be almost globally asymptotically stable (aGAS) if, for almost every initial state, the corresponding system trajectory converges to an equilibrium point (there may be many), see, e.g., [2], [37]. This is the best global stability property that we can hope to hold for a system that has more than one equilibrium point.

The following observations about equation (46) will be needed. This equation is usually studied in the phase space $\mathbb{R}^2$ consisting of all the pairs $(\delta, \frac{\mathrm{d}\delta}{\mathrm{d}\sigma})$. It is easy to see that there is an infinite set of equilibrium points for (46), and they are of the form $(\delta^e, 0)$, where $\sin\delta^e = \sin\lambda$. By linearization, it is easy to see that the equilibrium points with $\cos\delta^e > 0$ are exponentially stable, and the equilibrium points with $\cos\delta^e < 0$ are unstable. The linearization cannot have eigenvalues with zero real part, because $|\sin\delta| < 1$ and $\alpha > 0$. In addition to equilibrium points, the equation (46) may also exhibit limit cycles in the phase space. Hayes [17] has shown that this cannot happen if the damping is strong enough, namely

$$\alpha > 2\sin\left(\frac{\lambda}{2}\right). \tag{48}$$

It is most convenient to study the stability of (42) in the state space $\mathbb{T} \times \mathbb{R}$, i.e., to consider the angle δ_f modulo 2π.

Proposition 4.4: Under Assumption 5, if the constant α from (47) satisfies (48), then for almost every initial state $(\delta_{f0}, \tilde{\omega}_0) \in \mathbb{T} \times \mathbb{R}$, the trajectories of the boundary-layer (fast) system (42) converge to the stable equilibrium point $(\delta_f, \tilde{\omega}) = (0, 0)$.

Proof: From the Böhm-Hayes theorem [39, Theorem 5.3.3], under our assumptions, every state trajectory of (46) in the phase space $\mathbb{R}^2$ converges to one of the system's equilibrium points (stable or unstable). According to the observations before the proposition (the absence of imaginary eigenvalues), we can now argue exactly as in the proof of [34, Lemma 3.5] to conclude that, for almost every initial state in $\mathbb{R}^2$, the state trajectory of (46) converges to a stable equilibrium point.

There is a simple, smooth, and invertible transformation (consisting of the scaling (45) and the shift $\delta = \delta_f + \lambda$) by which the state trajectories of (42), regarded with the state space $\mathbb{R}^2$, are mapped into the state trajectories of (46). When we consider (42) with the angle δ_f measured modulo 2π, all of its stable equilibrium points (described before the proposition, in terms of the corresponding stable equilibria of (46)) collapse into $(0, 0)$. Similarly, all the unstable equilibria of (42) collapse into $(\delta_u - \lambda, 0)$, where δ_u is the unique angle with $\sin\delta_u = \sin\lambda$ and $\delta_u \in \left(\frac{\pi}{2} - \lambda, \frac{3\pi}{2} - \lambda\right)$ (considered as an arc on the circle $\mathbb{T}$). Hence, almost all the state trajectories of (42) converge to the stable equilibrium point $(0, 0)$. □

E. Stability analysis of the model (28).

Recall that under Assumptions 1–3, the equations (28) represent our internal model **S** and, in fact, our entire MPLL. We present here our main stability result for this system.

Theorem 4.5: Under Assumptions 1–5, there exists $k^* > 0$ such that for all $k \in (0, k^*]$ the following holds:

If $N = 0$, then the (unique) equilibrium point from (30) is locally exponentially stable. Moreover, when the system is at this equilibrium point, then $r = y$, i.e., the output of the MPLL tracks the reference signal r.

Proof: Our plan is to use [21, Theorem 11.4] (here, f and g do not depend on t and ε), concerning the exponential stability of a singularly perturbed system. We have already recast our system (28) as a singularly perturbed system in (40) in the spirit of [21, Chapt. 11]. For this, we have introduced new state variables in (32), (39), so that the equilibrium point of (40) is at the origin. Our functions $\left[\begin{smallmatrix} 1 & 0 \\ 0 & \gamma \end{smallmatrix}\right]\tilde{h}$, $\tilde{f}$ and $\tilde{\Xi}$ correspond to what is denoted in [21, Chapt. 11] by f, g, and h, respectively, while our small parameter k corresponds to ε in [21, Chapt. 11].

An elementary computation shows that the linearization of (41) around the equilibrium point $(0, 0)$ is of the form

$$\frac{\mathrm{d}}{\mathrm{d}v}\begin{bmatrix}\tilde{m} \\ \tilde{\omega}_{\text{LPF}}\end{bmatrix} = \begin{bmatrix}-\frac{R}{\sqrt{\rho L}} & * \\ 0 & -\gamma\end{bmatrix}\begin{bmatrix}\tilde{m} \\ \tilde{\omega}_{\text{LPF}}\end{bmatrix},$$

where $*$ denotes an irrelevant entry. Thus, it follows that the reduced model (41) is (locally) exponentially stable. Besides, from the observations written before Proposition 4.4, the boundary-layer system (42) is (locally) exponentially stable, and it is easy to see that this holds uniformly in $(\tilde{m}, \tilde{\omega}_{\text{LPF}})$ when $(\tilde{m}, \tilde{\omega}_{\text{LPF}})$ are in a small neighborhood of the origin. Hence, all the conditions in [21, Theorem 11.4] are satisfied, and we obtain that the origin is an exponentially stable equilibrium point for (40). Going back to the original state variables m, ω_{LPF}, δ, and ω, we conclude that the system (28) has an exponentially stable equilibrium point given by (30). □

Remark 4.6: Theorem 4.5 is a local result, which is not satisfactory for an MPLL. Indeed, the jumping subsystems from Subsect. III-C may bring ω, ω_{LPF} and m close to the equilibrium value, but the initial value of the angle δ is random. This is why Proposition 4.4 is important: it tells us that if the slow variables m and $\tilde{\omega}_{\text{LPF}}$ are practically constant (this is the case if Assumption 4 holds) and if the parameters satisfy the Hayes condition (48), then for almost every initial condition the fast variables δ and ω will converge to the correct values and, hence, the system state will get very close to its equilibrium point. After this, thanks to Theorem 4.5, the system will remain near the stable equilibrium point, i.e., it will be synchronized in frequency, phase, and amplitude. The small deviations will be due to noise in the input signal.

Remark 4.7: Unfortunately, there are no available results in the literature for aGAS singularly perturbed systems, which could strengthen the stability result from Theorem 4.5. Preliminary results on this class of systems can be found in [30], but the theory is not yet mature enough to be applied here.

V. Adaptive tuning of the MPLL parameters

In usual synchronverter applications, the operating frequency is in a narrow interval around 50 Hz (in Europe) and the amplitude of the line voltage is around $230\sqrt{3}$ Volts [27, Sect. II]. Given these two values, the parameters of the synchronverter model can be tuned so that stability is guaranteed, as described in [27, Sect. VI.A]. However, the synchronverter-based MPLL proposed in this paper is expected to operate over a wide range of frequencies/amplitudes, which are not known a priori. Therefore, in order to guarantee its stability, we propose here an adaptive tuning of its parameters.

Based on our previous experience with the synchronverter algorithm [24], [27], [34], [35], [47], we choose the nominal parameters for a sine wave with amplitude 300 and frequency 50Hz. The nominal values for J, D_p, k, L, τ are:

$$J_n = 0.02, \quad D_{p,n} = 1.21, \quad k_n = 0.2, \quad L_n = 0.05, \quad \tau_n = 0.5.$$

We mention that, with respect to usual synchronverter applications, the inertia J is lower. The frequency droop D_p is chosen such that the linearized version of (44) has two real poles, which coincide. With this choice, we guarantee critical damping, and thus, the fastest convergence without overshoot.

We introduce a frequency-scaling term ω_{sc} and an amplitude-scaling term r_{sc} given by

$$\omega_{\text{sc}} = \frac{\omega}{2\pi \cdot 50}, \qquad r_{\text{sc}} = \frac{R_{\text{est,LPF}}}{300},$$

where ω (in $\frac{\text{rad}}{\text{sec}}$) is the internal frequency of $\mathbf{S}$ and $R_{\text{est,LPF}}$ is obtained as in (23), but using $r_{dq,\text{LPF}}$ in place of r_{dq}. Recall the jumping subsystems from Subsect. III-C. Whenever $\omega, \omega_{\text{LPF}}$ and m are updated (according to (22) and (24)), the parameters of (28) are scaled as follows:

$$J = \frac{J_n}{\omega_{\text{sc}}^4}, \quad D_p = \frac{D_{p,n}}{\omega_{\text{sc}}^3}, \quad k = k_n \cdot \sqrt{\omega_{\text{sc}}} \cdot r_{\text{sc}}, \quad L = L_n \cdot r_{\text{sc}}^2, \quad \tau = \frac{\tau_n}{\omega_{\text{sc}}}. \tag{49}$$

In (24) we often use $R_{\text{est,LPF}}$ in place of R_{est}.

The parameters of the OSG from Fig. 3 also need to be updated. Denoting the time constant of the low-pass filter at the output of the OSG by τ_r, we choose as nominal values

$$p_n = 2, \qquad \tau_{r,n} = 0.05,$$

and we update p and τ_r according to the law:

$$p = p_n \cdot \omega_{\text{sc}}, \qquad \tau_r = \frac{\tau_{r,n}}{\omega_{\text{sc}}}.$$

Further, in case of a frequency jump, the state of the quasi integrator inside the OSG is updated as

$$r_\beta = r_{\beta,pr}\omega_{\text{sc}},$$

where $r_{\beta,pr}$ is the value before the frequency jump. Finally, the time constant for the jumping subsystem is chosen (and updated) according to: $T_{\text{jump}} = 0.6/\omega_{\text{sc}}$.

VI. Some applications of the proposed MPLL

A. Using the MPLL for estimating several fundamental components of the input signal

In some applications, it might be relevant to estimate several sinusoidal components of r, where

$$r(t) = \sum_{i=1}^{n} R_i \sin(\omega_{R,i} t + \gamma_{R,i}),$$

with rapidly decreasing amplitudes. Suppose that we want to track the first n_R components of r. To solve this problem, n_R copies of the MPLL from Fig. 2 can be connected as shown in Fig. 8, where, for simplicity, we have chosen $n_R = 3$. Each MPLL receives the reference r minus a signal $\bar{h}_j$ which removes those components from r that are already tracked by another MPLL: $\bar{h}_j = \sum_{k \neq j} y_k s_k$ where $s_k = 1$ if the k-th MPLL is synchronized and $s_k = 0$ otherwise. The output of the multi-MPLL is given by $y_\Sigma = \sum y_k s_k$. The algorithm works as follows. First, MPLL_1 is connected and the upper switch is open. Then, once y_1 has converged to the first sinusoidal component of r, $\bar{h}_2 = \bar{h}_3 = y_1$. The upper switch is closed. In this way, MPLL_2 is connected and, at steady-state, y_2 converges to the second sinusoidal component of r such that $\bar{h}_1 = y_2$, $\bar{h}_2 = y_1$ and $\bar{h}_3 = y_1 + y_2$. Finally, MPLL_3 is connected and synchronizes.

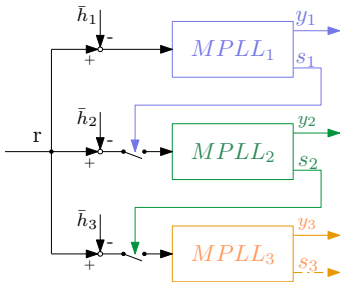


Fig. 8: The connection of three copies of the MPLL from Fig. 2 when $n_R = 3$ in Subsect. VI-A.

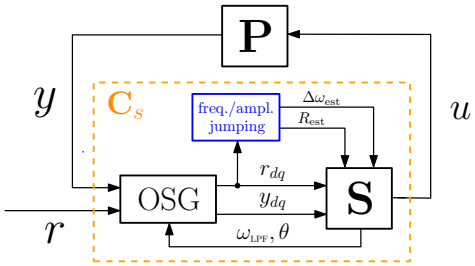


Fig. 9: Closed-loop system formed by the linear stable plant $\mathbf{P}$ in feedback with the controller $\mathbf{C}_s$. The signal r is the reference to be tracked by the output y of $\mathbf{P}$. There are two copies of the OSG from Fig. 3: one for r and one for y. In $\mathbf{S}$ here (14d), (14e) are replaced with (51), (50) respectively.

B. Using the proposed MPLL as an internal model controller for reference tracking

The MPLL from Fig. 2 can be also employed to solve the reference tracking problem for a stable unknown linear plant $\mathbf{P}$, when the reference r is a sinusoid with uncertain frequency, as described in [28], [29]. For this application, the closed-loop system has to be formed as shown in Fig. 9, where y is the output of the plant. In this configuration, y_d and y_q need to be computed using an additional OSG. The idea here is that, thanks to the robustness of $\mathbf{S}$ to amplitude changes and phase shifts in y, caused by $\mathbf{P}$, the closed-loop system from Fig. 9 behaves similarly to the MPLL from Fig. 2. The stability analysis becomes more involved in this case, due to the presence of the unknown plant $\mathbf{P}$. In these settings, the currents i_d, i_q in $\mathbf{S}$ have to be computed from

$$\omega_{\text{LPF}} L i_d = y_q - r_q, \qquad \omega_{\text{LPF}} L i_q = r_d - y_d, \tag{50}$$

(instead of (14e)) and the swing equation (14d) in $\mathbf{S}$ becomes

$$J\dot{\omega} = -\frac{y_d i_d + y_q i_q}{\omega_{\text{LPF}}} - D_p(\omega - \omega_{\text{LPF}}). \tag{51}$$

We present an outline of how the arguments from Sect. IV can be modified to study the stability of the system from Fig. 9. The signal y is now the output of $\mathbf{P}$, given by

$$y = |\mathbf{P}(j\omega)| m\omega \sin(\theta + \arg \mathbf{P}(j\omega)), \tag{52}$$

where $|\mathbf{P}(j\omega)|, \arg \mathbf{P}(j\omega)$ are, respectively, the magnitude and the phase shift of the plant transfer function $\mathbf{P}(s)$ at $s = j\omega$. Since $\mathbf{P}$ is unknown, we cannot recover the dq components of y from the states of $\mathbf{S}$, and we need an additional OSG.

Assumption 6: We assume that the dynamics of $\mathbf{P}$ can be replaced with its steady-state input-output map. We further assume that $\mathbf{P}(s) \neq 0$ for $s = \pm j\omega_R$.

Under Assumptions 2, 3, and 6, for y from (52) we have $y_\alpha = y$ and $y_\beta = -|\mathbf{P}(j\omega)| m\omega \cos(\theta + \arg \mathbf{P}(j\omega))$. Applying (9) to the above signals yields

$$\begin{aligned} y_d &= |\mathbf{P}(j\omega)| m\omega \sin(\arg \mathbf{P}(j\omega)), \\ y_q &= -|\mathbf{P}(j\omega)| m\omega \cos(\arg \mathbf{P}(j\omega)). \end{aligned} \tag{53}$$

Assumption 7: We assume that for $\omega, \omega_{\text{LPF}} \in (\omega_R - \varepsilon, \omega_R + \varepsilon)$, with $\varepsilon > 0$ small (as in Assumption 2), we have

$$\begin{aligned} |\mathbf{P}(j\omega)| &\approx |\mathbf{P}(j\omega_R)| = A_r, \\ \arg \mathbf{P}(j\omega) &\approx \arg \mathbf{P}(j\omega_R) = \varphi_r. \end{aligned}$$

Remark 6.1: Assumptions 2 and 7 can be found also in [9], even though no "jumping" feature is employed there.

Under the above assumption, we can rewrite (53) as follows:

$$y_d = A_r m\omega \sin \varphi_r, \qquad y_q = -A_r m\omega \cos \varphi_r. \tag{54}$$

The sinusoidal reference r is as in (4), with $N = 0$. Thus, r_d and r_q are as in (12), with, respectively, $n_d = 0$ and $n_q = 0$. (There is no need to low-pass filter the outputs of the OSG here.) Following Sect. IV, we can derive the closed-loop system equations, similar to (28), with the difference that here y_d and y_q are as in (54) (replacing, respectively $y_d = 0$ and $y_q = -m\omega$), and T_e has to be computed using

$$P = \omega T_e = r_d i_d + r_q i_q.$$

Therefore,

$$\begin{aligned} i_d &= \frac{1}{\omega_{\text{LPF}} L} \left(A_r m\omega \cos \varphi_r + R \cos \delta\right), \\ i_q &= \frac{1}{\omega_{\text{LPF}} L} \left(-A_r m\omega \sin \varphi_r - R \sin \delta\right), \\ Q &= \frac{1}{\omega_{\text{LPF}} L} \left[R A_r m\omega \cos(\delta + \varphi_r) - R^2\right], \\ T_e &= \frac{R A_r m}{\omega_{\text{LPF}} L} \sin(\delta + \varphi_r). \end{aligned}$$

Using the above, we obtain the following (simplified) description of the closed-loop system from Fig. 9:

$$\begin{cases} \dot{m} = -k \left(\dfrac{R A_r m\omega \cos(\delta + \varphi_r) - R^2}{\omega_{\text{LPF}} L} \right)_\rho^{\frac{1}{2}}, & \text{(55a)} \\ \tau \dot{\omega}_{\text{LPF}} = -\omega_{\text{LPF}} + \omega, & \text{(55b)} \\ \dot{\delta} = \omega - \omega_R, & \text{(55c)} \\ J\dot{\omega} = -D_p(\omega - \omega_{\text{LPF}}) - \dfrac{R A_r m}{\omega_{\text{LPF}} L} \sin(\delta + \varphi_r). & \text{(55d)} \end{cases}$$

From here, using the same singular perturbation arguments of Sect. V, the stability of the reduced (slow) model and of the boundary-layer (fast) system can be studied (as in Subsect. IV-C and Subsect. IV-D) and conclusions on the overall stability of (55) can be obtained when k and $\frac{1}{\tau}$ are "sufficiently" small.

Remark 6.2: Extensive simulation results for adaptive internal models with an architecture similar to Fig. 9 are in [28], [29]. With respect to [28], [29], the novelty of the architecture presented here is the amplitude jumping feature described in Sect. III-C and the adaptive tuning described in Sect. V.

VII. Simulation results

We present simulation results for the full system from Fig. 2 in Simulink, for different choices of ω_R, ω_N, R and N. In all simulations, the MPLL has been initially tuned for $R_{\text{est}}(0) = R_{\text{est,LPF}}(0) = 300$ and $\omega(0) = 100 \cdot 2\pi$ rad/s. To compare the tracking quality of the MPLL, we define the *normalized tracking error* $\varepsilon =: \sqrt{2}\frac{y-r}{R}$. For convenience, we show frequencies in Hz and not in rad/s, and we denote $f_R = \omega_R/(2\pi)$ and $f_N = \omega_N/(2\pi)$. We use the noise to signal ratio $\eta = N/R$ and the ratio of the noise frequency to the signal frequency $\nu = \omega_N/\omega_R$.

A. Convergence towards a reference signal

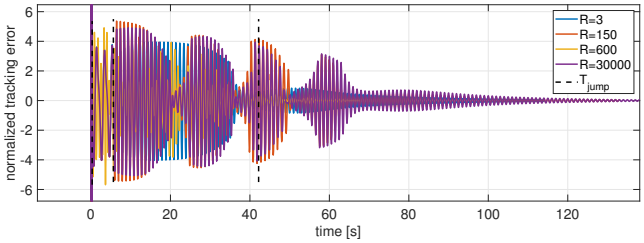
We demonstrate here the capability of the MPLL to track reference signals with $R \in [3; 30{,}000]$ and $f_R \in [1; 10{,}000]$ Hz. Fig. 10 shows the normalized tracking error ε (a), the MPLL frequency $f = \omega/(2\pi)$ (b) and the amplitude scaling r_{sc} (c) for $f_R = 1$ Hz. The MPLL performs 2 or 3 jumps in all cases before converging and, apart from the simulation where $R = 600$, the system initially jumps to an even lower frequency of 0.5Hz, before again jumping back to a frequency close to 1Hz. After 70 seconds, the system tracks f_R closely. Fig. 10(a) shows that after frequency convergence, another 50 cycles are required for the MPLL to adapt to the correct amplitude of r. Fig. 11 shows results for $f_R = 50$Hz.

Fig. 12 shows results for $f_R = 10{,}000$Hz. Interestingly, only one jump is performed for the system to be sufficiently close to f_R. After the jump, occurring at $t = 0.3$s, both frequency and normalized tracking error quickly converge.

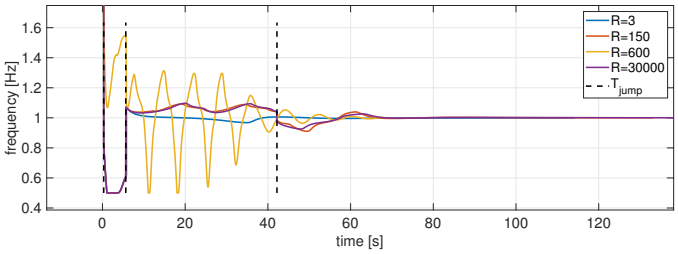
Remark 7.1: We emphasize that the adaptive tuning presented in Sect. V has been employed in all our simulations. Without it, the pull-in range of the proposed MPLL would drastically decrease, and the system may become unstable when operated far from its "nominal tuning" range. Indeed, without adaptive tuning, we have shown in our conference paper [28] that synchronization is possible only for $\omega_R \in [0.25\omega_n, 4\omega_n]$, where $\omega_n = \omega(0)$ indicates the nominal value according to which the algorithm in [28] is tuned. As shown above, the algorithm proposed in this paper has a much wider pull-in range. For instance, the above simulations show convergence for $\omega_R \in [0.01\omega(0), 100\omega(0)]$.

B. Noise rejection

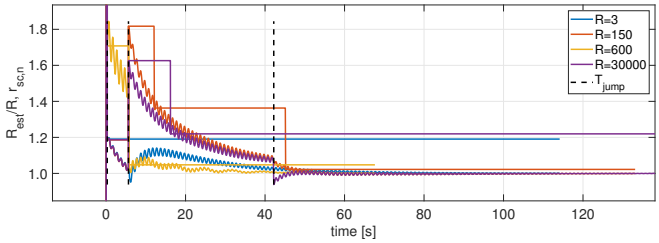
In order to investigate the robustness of the MPLL to noise, we have simulated the system for various combinations of N and ω_N when the reference signal is given by $R = 300$ and $f_R = 100$Hz. In Fig. 13 we plot a pixel for each choice of η and ν. The color of a pixel is defined by the rms-value of the normalized tracking error $E_{rms} = \sqrt{LPF(\varepsilon^2)}$ that the system achieves after converging, where $LPF(\cdot)$ denotes a low pass filter with time constant $\frac{1}{\omega_R}$. A purple line indicates (5). We note that the for the values of η and ν that fulfill (5), namely pixels that are on the left of the purple line, E_{rms} is low. When η is much larger than what is defined by (5), the tracking error becomes large. At $\nu = 1$, meaning $\omega_R = \omega_N$, the amplitude of the MPLL adapts to $R+N$ because noise and



(a) The normalized tracking error ε. We see that ε decays exponentially once $f \approx f_R$, i.e., from $t = 80$s (approximately).

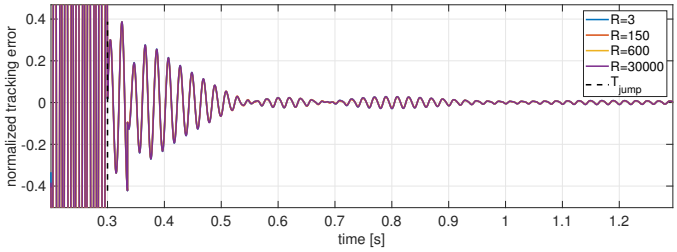


(b) The frequency f of the MPLL when tracking a 1Hz signal. The system jumps 2 to 3 times to reach $f_R = 1$Hz.

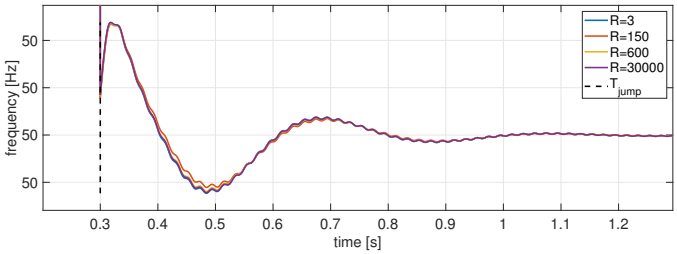


(c) The estimated amplitude R_{est} (continuous line) and the scaling factor r_{sc} (step function) used for tuning of the MPLL.

Fig. 10: MPLL performance for tracking a sine wave with $f_R = 1$Hz and $R \in [3; 30{,}000]$, without noise.

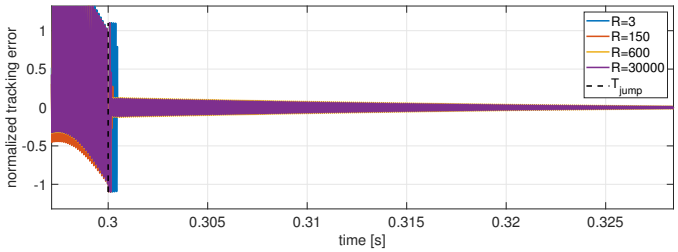


(a) The normalized tracking error ε (for $f_R = 50$Hz) converges, almost independently of R, within 25 oscillations to a low value.

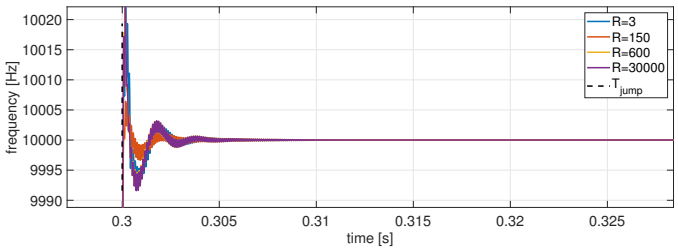


(b) After an initial jump, the frequency $f = \omega/(2\pi)$ converges within about 50 cycles.

Fig. 11: MPLL performance for tracking a sine wave with $f_R = 50$Hz and $R \in [3; 30{,}000]$, without noise.



(a) The normalized tracking error ε decays exponentially for $t > 0.3$s.



(b) The frequency f of the MPLL when tracking a 10kHz signal. The system jumps once and then after ≈ 50 cycles, $f \approx f_R$.

Fig. 12: MPLL performance for tracking a signal of 10kHz and $R \in [3; 30{,}000]$, without noise.

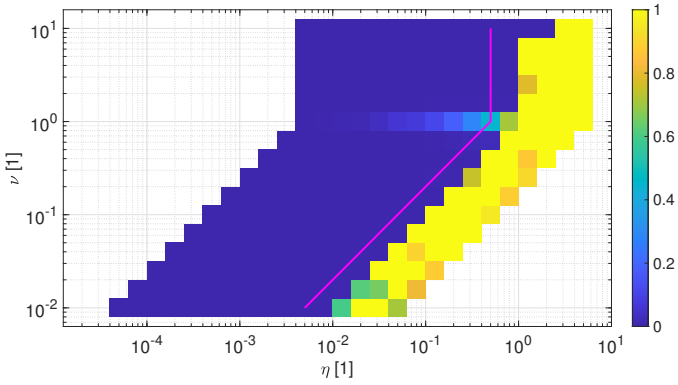


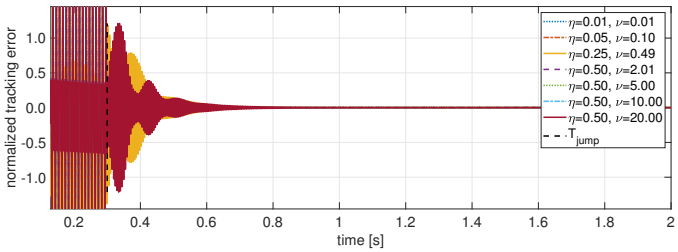
Fig. 13: A map where the color of each pixel denotes the low-pass filtered rms-value of the normalized tracking error after convergence. The purple line indicates the right boundary of the region satisfying condition (5).

reference signal cannot be distinguished, such that the actual tracking error is proportional to N.

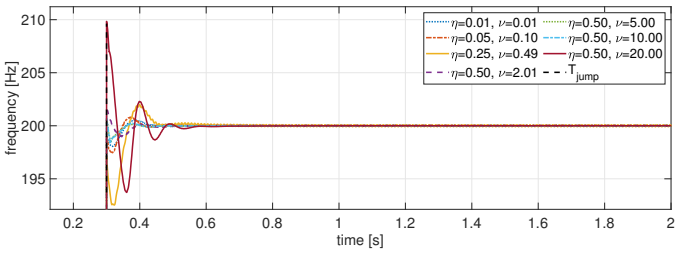
In order to further illustrate performance of our MPLL under the presence of noise, Fig. 14 shows some results with $f_R = 200$Hz and for different f_N where (5) holds. The system is stable and converges quickly after one initial jump in all cases. As already seen in Subsect. VII-A, frequency convergence is faster than amplitude convergence. This follows from the inherent two time-scales behavior of the proposed MPLL: the boundary-layer (fast) system (42) represents the frequency dynamics, while the reduced (slow) model (41) represents the amplitude dynamics of the MPLL output signal.

C. MPLL and QPLL compared

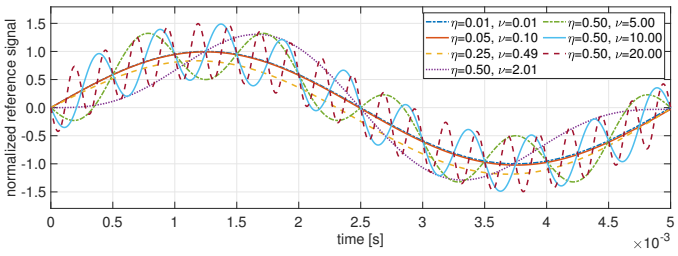
We give here a comparison of our MPLL's performance with another example from the literature: the quadrature PLL (QPLL) from [19]. Since the QPLL does not use adaptive



(a) The normalized tracking error ε when noise is present.



(b) The frequency of the MPLL quickly converges to $f_R = 200$Hz.



(c) Zoom of one oscillation period of r for the selected η and ν.

Fig. 14: MPLL performance for $f_R = 200$ Hz and various amplitudes and frequencies of the noise, all satisfying (5).

tuning features such as the jumping mechanism detailed in this paper, we do not expect it to work far away from its tuning point. In Fig. 15 we show the obtained tracking results for the QPLL when tuning was done for $R = 300$ and $f_R = 100$Hz. In this case, the relevant parameters (see [19]) are: $\mu_f = 200\pi$, $\mu_s = \mu_c = 300$. The way the QPLL is constructed, the QPLL frequency $f = \omega/(2\pi)$ is very volatile. If only signal tracking is of interest, f is not a good indicator for the QPLL's performance: The frequency f may have very large fluctuations (by a multiple of the reference frequency) while r is still relatively well tracked. We show in Fig. 15 both indicators: We plot E_{rms} (left) and W_{rms} for $R \in [30; 30{,}000]$ and $f_R \in [10; 1{,}000]$ Hz, where $W_{rms} = \sqrt{LPF\left(\frac{f-f_R}{f_R}\right)^2}$, similar to E_{rms}. It can be seen that the QPLL is able to track most signals relatively well, while interestingly there are some combinations of R and f_R that lead to very large tracking errors. The indicator W_{rms} performs poorer. We did the same tests with the MPLL, however we do not plot these results here, since both graphs for E_{rms} and W_{rms} were entirely blue (meaning that the MPLL performed much better).

Similar results can be obtained if noise is added to r: Fig. 16 shows the performance of the QPLL when in each simulation $\omega_N = 0.75\omega_R$ and $N = 0.5R$. Clearly, the system behavior deteriorates when compared to the noiseless scenario. The same simulations conducted with our MPLL give the results

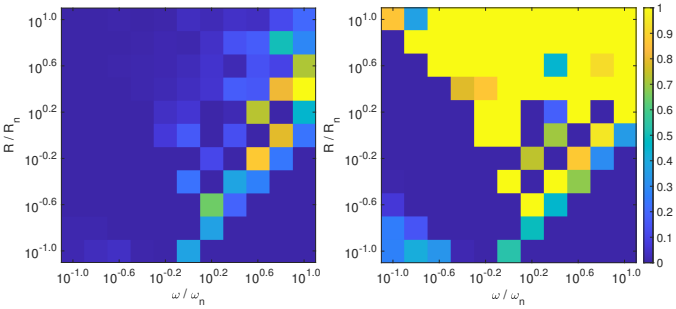


Fig. 15: Tracking performance of the QPLL in [19] without noise: E_{rms} (left) and W_{rms} (right) for different R and ω. The error E_{rms} is small, even if the frequency is fluctuating.

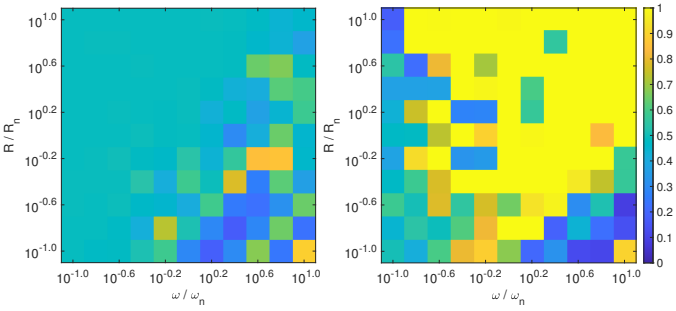


Fig. 16: Tracking performance of the QPLL in [19] with noise: E_{rms} (left) and W_{rms} (right) for different R and ω.

shown in Fig. 17. Due to the system inertia, the frequency error W_{rms} (right) is very low in all cases. A very light tracking error E_{rms} is distinguishable for lower frequencies (left). We show plots of ω, r and y when $f_R = 160$Hz and $R = 475.5$ in Fig. 18. The top plots of Fig. 18a and 18b show the performance of the MPLL and the bottom parts show the performance of the QPLL. The QPLL actually reproduces r including the noise signal.

D. Tracking of several frequencies

We show here an example when r is not composed of a single frequency. We took $r = \sum_1^3 R_j \sin(\omega_j t)$ where $\omega = [2\ 3\ 4] \cdot 200\pi$, $R = [2\ 1.5\ 1] \cdot 300$, $\omega_N = 600 \cdot 2\pi$ and $N = 150$. Fig. 19a shows $e = r - y_\Sigma$ in gray, with y_Σ as defined in Sect. VI-A. Fig. 19b shows the frequency of the three MPLLs as a solid line and their expected target

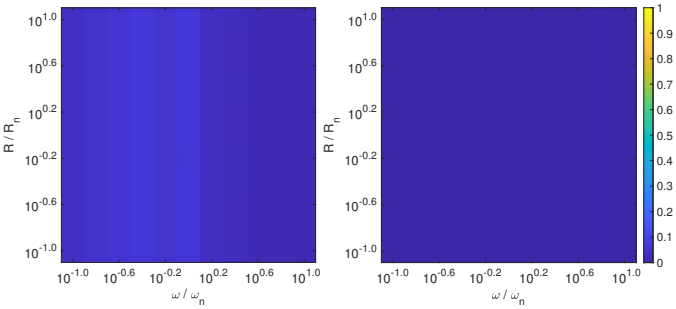
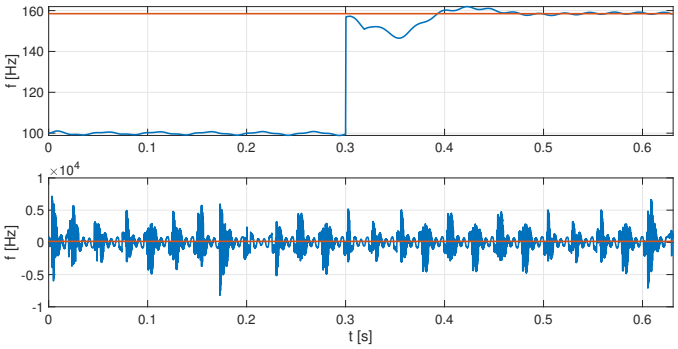
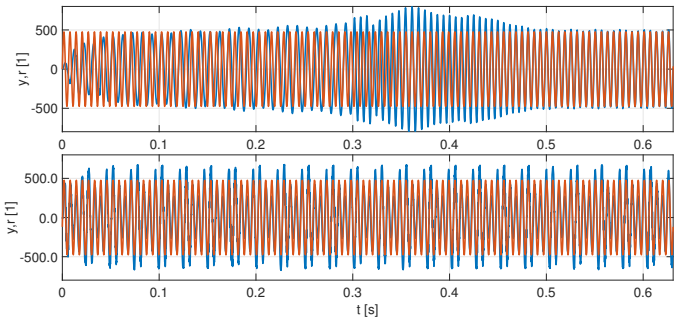


Fig. 17: Tracking performance of the proposed MPLL with noise: E_{rms} (left) and W_{rms} (right) for different R and ω.



(a) System frequency f (blue) and reference frequency $f_R = 160$Hz (red) of the MPLL (top) and the QPLL (bottom). The MPLL jumps at $t = 0.3$s (top). The QPLL shows a highly fluctuating f.



(b) r (red) and y (blue) for the MPLL (top) and the QPLL (bottom). After jumping at $t = 0.3$s, y converges to r in the case of the MPLL, while the QPLL produces a more noisy output.

Fig. 18: Comparison of the MPLL and the QPLL for $f_R = 160$Hz, $R = 475.5$, $N = R/2$, $f_N = 120$ Hz.

frequency as a dashed line. In both graphs, a vertical dash-dotted line indicates the start of operation of a MPLL and a dashed vertical line indicates a frequency jump. We see that each MPLL converges correctly and that the noise signal decreases each time a new MPLL has synchronized.

VIII. Conclusion

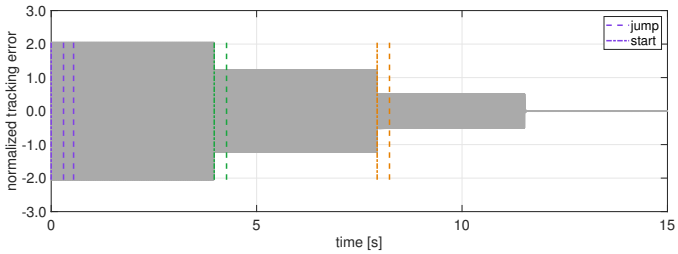
A novel synchronverter-based MPLL has been proposed, with a wide pull-in range and great noise rejection properties. Using singular perturbation methods, we have derived sufficient conditions on the MPLL parameters guaranteeing stability. These conditions, together with the performance of the proposed MPLL, have been validated through extensive simulations. Future work will be devoted at using the proposed MPLL to build an adaptive internal model-based controller, to solve the output regulation and disturbance rejection problem when the plant is uncertain and the reference frequency is unknown, improving the results from [28], [29].

References

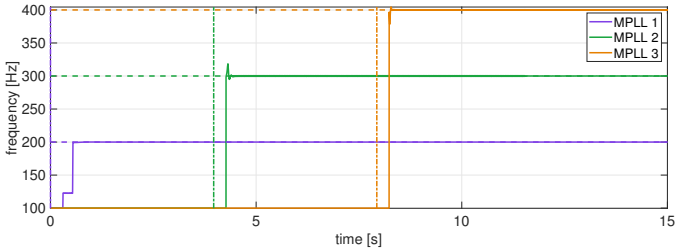
[1] J. Alipoor, Y. Miura and T. Ise, "Distributed generation grid integration using virtual synchronous generator with adaptive virtual inertia," *Proc. IEEE Energy Conversion Congress and Exposition (ECCE)*, Denver, CO, Sept. 2013, pp. 4546–4552, 2013.

[2] D. Angeli, "An almost global notion of input-to-state stability," *IEEE Trans. on Automatic Control*, vol. 49, pp. 886–874, 2004.

[3] R. Aouini, B. Marinescu, K. Ben Kilani and M. Elleuch, "Synchronverter-based emulation and control of HVDC transmission," *IEEE Trans. Power Systems*, vol. 31, pp. 278–286, 2015.



(a) The normalized tracking error $r - y_\Sigma$ (grey) and the subsequent start of synchronization of each MPLL (vertical dash-dotted lines) and the moment the MPLL jumped (dashed lines).



(b) Frequency of the three MPLLs (solid lines), the reference frequency component in r (dashed lines). A vertical dash-dotted line marks the start of operation of each MPLL.

Fig. 19: Tracking of a signal with 4 frequency components of which one is considered noise. After sequential synchronization of the three MPLLs the three components of r are well tracked and the error is low.

[4] A. Bamigbade and V. Khadkikar, "Benchmarking of different orthogonal signal generator configurations for SOGI PLL applications," *IEEE Ind. Appl. Soc. Annual Meeting (IAS)*, pp. 1–6, 2021.

[5] A. Bamigbade, V. Khadkikar, H.H. Zeineldin, M.S.E. Moursi and M.A. Hosani, "A novel power-based orthogonal signal generator for single-phase systems," *IEEE Trans. on Power Delivery*, vol. 36, pp. 469–472, 2021.

[6] N. Barabanov, J. Schiffer, R. Ortega and D. Efimov, "Conditions for almost global attractivity of a synchronous generator connected to an infinite bus," *IEEE Trans. Automatic Control*, vol. 62, pp. 4905–4916, 2017.

[7] R.E. Best, *Phase-Locked Loops: Design, Simulation, and Applications. 6th edit.*, McGraw-Hill Education, New York, 2007.

[8] M. Bin, L. Marconi and A.R. Teel, "Adaptive output regulation for linear systems via discrete-time identifiers," *Automatica*, vol. 105, pp. 422–432, 2019.

[9] M. Bodson and S.C. Douglas, "Adaptive algorithms for the rejection of sinusoidal disturbances with unknown frequency," *Automatica*, vol. 33, pp. 2213–2221, 1997.

[10] R. Bojoi, F. Mandrile, F. Reissner and G. Weiss, "Virtual synchronous machines - inverters for a stable and well-damped grid," tutorial at *Energy Conversion Congress and Exposition* (ECCE 2022), Detroit, USA, Oct. 2022.

[11] M. Ciobotaru, R. Teodorescu and F. Blaabjerg, "A new single-phase PLL structure based on second order generalized integrator," *37th IEEE Power Electronics Specialists Conf. (PESC)*, pp. 1–6, 2006.

[12] S. Dong, Y.-N. Chi and Y. Li, "Active voltage feedback control for hybrid multi-terminal HVDC system adopting improved synchronverters," *IEEE Trans. on Power Delivery*, vol. 31, pp. 445–455, 2016.

[13] S. Gautam, W. Hassan, A. Bhatta, D.D.-C. Lu and W. Xiao, "A comprehensive study of orthogonal signal generation schemes for single phase systems," *1st Intern. Conf. on Power Electronics and Energy (ICPEE)*, pp. 1–8, 2021.

[14] S. Golestan, J.M. Guerrero, F. Musavi and J.C. Vasquez, "Single-phase frequency-locked loops: A comprehensive review," *IEEE Trans. on Power Electronics*, vol. 34, pp. 11791–11812, 2019.

[15] S. Golestan, M. Monfared, F.D. Freijedo and J.M. Guerrero, "Dynamics assessment of advanced single-phase PLL structures," *IEEE Trans. on Industrial Electronics*, vol. 60, pp. 2167–2177, 2013.

[16] 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. on Power Electronics*, vol. 31, pp. 3932–3944, 2016.

[17] W.D. Hayes, "On the equation for a damped pendulum under constant torque," *Journal of Applied Mathematics and Physics*, vol. 4, pp. 398-401, 1953.

[18] T. Hornik, Q.-C. Zhong. *Control of Power Inverters in Renewable Energy and Smart Grid Integration*. John Wiley & Sons, 2012.

[19] M. Karimi-Ghartemani, H. Karimi and M.R. Iravani, "A magnitude/phase-locked loop system based on estimation of frequency and in-phase/quadrature-phase amplitudes," *IEEE Trans. on Industrial Electronics*, vol. 51, pp. 511–517, 2004.

[20] M. Karimi-Ghartemani, *Enhanced Phase-Locked Loop Structures for Power and Energy Applications.* IEEE Press and John Wiley & Sons, Hoboken, NJ, 2014.

[21] H.K. Khalil, *Nonlinear Systems; 3rd ed.* Prentice-Hall, Upper Saddle River, NJ, 2002.

[22] P. Kokotovic, H.K. Khalil, and J. O'Reilly, *Singular Perturbation Methods in Control: Analysis and Design*. SIAM, 1999.

[23] P.S. Kundur, O.P. Malik, *Power System Stability and Control; 2nd ed.* McGraw Hill, New York, 2022.

[24] Z. Kustanovich, S. Shivratri, Y. Hang, F. Reissner, G. Weiss, "Synchronverters with fast current loops", *IEEE Trans. on Industrial Electronics*, vol 70, pp.11357–11367, 2023.

[25] N.V. Kuznetsov, G.A. Leonov, M.V. Yuldashev, R.V. Yuldashev, "Rigorous mathematical definitions of the hold-in and pull-in ranges for phase-locked loops," *IFAC-PapersOnLine*, vol. 48, pp. 710–713, 2015.

[26] G.A. Leonov and N.V. Kuznetsov, *Nonlinear Mathematical Models of Phase-Locked Loops. Stability and Oscillations.* Combrisge University Press, UK, 2014.

[27] P. Lorenzetti, Z. Kustanovich, S. Shivratri, and G. Weiss, "The equilibrium points and stability of grid-connected synchronverters," *IEEE Trans. Power Systems*, vol. 37, pp. 1184–1197, 2022.

[28] P. Lorenzetti, F. Reissner and G. Weiss, "An internal model for tracking a sinusoidal reference with unknown frequency and uncertain plant," *2022 IEEE 61st Conference on Decision and Control, Cancun, Mexico*, pp. 2525–2531, 2022.

[29] P. Lorenzetti and G. Weiss, "A variable frequency internal model controller inspired by synchronverter theory," *2022 American Control Conference, Atlanta, GA, USA*, pp. 715–720, 2022.

[30] P. Lorenzetti and G. Weiss, "Almost global stability results for a class of singularly perturbed systems," *2023 22nd IFAC World Congress, Yokohama, Japan*, vol. 56, pp. 809–814, 2023.

[31] P. Lorenzetti and G. Weiss, "PI control of stable nonlinear plants using projected dynamical systems," *Automatica*, vol. 146, pp. 110606, 2022.

[32] P. Lorenzetti and G. Weiss, "Saturating PI control of stable nonlinear systems using singular perturbations," *IEEE Trans. on Automatic Control*, vol. 68, pp. 867–882, 2023.

[33] V. Mallemaci, F. Mandrile, S. Rubino, A. Mazza, E. Carpaneto, R. Bojoi, "A comprehensive comparison of virtual synchronous generators with focus on virtual inertia and frequency regulation," *Electric Power Systems Research*, vol. 201, pp. 107516, 2021.

[34] V. Natarajan and G. Weiss, "Almost global asymptotic stability of a grid-connected synchronous generator," *Math. of Control, Signals and Systems*, vol. 30, pp. 1–43, 2018.

[35] V. Natarajan and G. Weiss, "Synchronverters with better stability due to virtual inductors, virtual capacitors and anti-windup," *IEEE Trans. on Industrial Electronics*, vol. 64, pp. 5994–6004, 2017.

[36] G. Pin, B. Chen, G. Fedele, and T. Parisini, "Robust frequency-adaptive quadrature phase-locked-loops with Lyapunov-certified global stability," *IEEE Trans. on Control Systems Technology*, vol. 31, pp. 467–474, 2023.

[37] A. Rantzer, "A dual to Lyapunov's stability theorem," *Systems & Control Letters*, vol. 42, pp. 161-168, 2001.

[38] A. Rantzer, "Almost global stability of phase-locked loops," *Proceedings of the 40th IEEE Conference on Decision and Control*, pp. 899–900 vol. 1, 2001.

[39] V. Reitmann, V.B. Smirnova and G.A. Leonov, *Non-Local Methods for Pendulum-Like Feedback Systems*, Wiesbaden: Springer, 1992.

[40] W. Rudin, *Functional Analysis*. McGraw-Hill, New York, 1973.

[41] Z. Shuai, W. Huang, Z.J. Shen, A. Luo and Z. Tian, "Active power oscillation and suppression techniques between two parallel synchronverters during load fluctuations," *IEEE Trans. on Power Electronics*, vol. 35, pp. 4127–4142, 2020.

[42] S.M. Silva, B.M. Lopes, B.J.C. Filho, R.P. Campana and W.C. Bosventura, "Performance evaluation of PLL algorithms for single-phase grid-

connected systems," *IEEE Industry Applications Conf. 2004, 39th IAS Annual Meeting*, pp. 2259–2263 vol. 4, 2004.
[43] D. Stojić, N. Georgijević, M. Rivera, S. Milić, "Novel orthogonal signal generator for single phase PLL applications", *IET Power Electronics*, vol. 11, pp. 427–433, 2018.
[44] K.R. Vasudevan, V.K. Ramachandaramurthy, T.S. Babu and A. Pouryekta, "Synchronverter: A comprehensive review of modifications, stability assessment, applications and future perspectives," *IEEE Access*, vol. 8, pp. 131565–131589, 2020.
[45] Q.-C. Zhong and D. Boroyevich, "Structural resemblance between droop controllers and phase-locked loops," *IEEE Access*, vol. 4, pp. 5733–5741, 2016.
[46] Q.-C. Zhong, P.-L. Nguyen, Z. Ma and W. Sheng, "Self-synchronized synchronverters: Inverters without a dedicated synchronization units," *IEEE Trans. Power Electronics*, vol. 29, pp. 617–630, 2014.
[47] Q.-C. Zhong and G. Weiss, "Synchronverters: Inverters that mimic synchronous generators," *IEEE Trans. on Industr. Electronics*, vol. 58, pp. 1259–1267, 2011.
[48] D. Zonetti, A. Bobtsov, R. Ortega, N. Nikolaev and O. Gomis-Bellmunt: "An almost globally stable adaptive phase-locked loop for synchronization of a grid-connected voltage source converter," *arXiv*, 2022.

Pietro Lorenzetti received the MEng degree in Mechatronic Engineering from Politecnico di Torino, and in Automation and Control Engineering from Politecnico di Milano in 2017, with honours, thanks to the double-degree program "Alta Scuola Politecnica". He was an Early Stage Researcher within the Marie Curie ITN project "ConFlex" in the Control and Power Electronics group in Tel Aviv University. He completed his PhD in control engineering at Tel Aviv University, in 2023, and he is now with Université de Lorraine, Nancy. He is the recipient of the IFAC Young Author Award of the MICNON2021 conference. His research interests include nonlinear systems, nonlinear control, and power systems stability.

Florian Reissner received the BSc and MSc degree from Technical University Berlin, Germany in 2015. Between 2015 and 2020 Florian worked in project management at Vinci Energies, Lyon and as innovation consultant in incubators in Frankfurt and Berlin. In 2020, he was selected as an Early Stage Researcher in the Marie Curie ITN project "WinGrid" and has started working towards the PhD in the Control and Power Electronics group in Tel Aviv University. He likes to build complex systems from scratch, overseeing every detail.

George Weiss received the MEng degree in control engineering from the Polytechnic Institute of Bucharest, Romania, in 1981, and the PhD degree in applied mathematics from Weizmann Institute, Rehovot, Israel, in 1989. He was with Virginia Tech, Blacksburg, VA, Ben-Gurion University, Beer Sheva, Israel, the University of Exeter, UK, and Imperial College London, UK. His research interests include distributed parameter systems, operator semigroups, power electronics, and the grid integration of distributed energy sources.