ISSUE: April 2026

Stabilizing The Loops Of A Three-Phase Vienna Rectifier—Part 2

by Christophe Basso, Future Electronics, Toulouse, France

In the first part of this series,[1] I have shown how to assemble two PWM switch models to predict the ac response of a single-phase T-type converter. I have then extended the analysis to the three-phase Vienna rectifier, driven with a *dq0* algorithm. After several ac sweep sequences, the three loops have been stabilized with classical analog type 2 compensators. The transient simulations have confirmed stable input currents with a low harmonic content in low- and high-line conditions.

In this second part, we will continue simulating the Vienna rectifier with a cycle-by-cycle (switching) model for checking the total harmonic distortion (THD) but also the stability under various loading conditions. Specifically, the transient response of the three loops in the Vienna rectifier will be verified using the switching model under low-line and high-line inputs, and these simulations will be compared against those obtained with the averaged models.

After these points have been verified, I will show how to close the loops with a digital proportional integral (PI) compensation block. This involves an explanation of how the analog type 2 compensator, which was generated with a macro in part 1, operates, followed by an explanation of the proportional-integral or PI compensator. The latter is the digital equivalent of the analog type 2a compensator, which is like the type 2 but lacks the high-frequency pole. As we'll see, the addition of a zero-order hold to the digital compensator achieves the same effect as the missing high-frequency pole.

This discussion also covers how to build the PI compensator in LTspice using the backward Euler transform, modeling the filter with delay blocks and automating generation of filter coefficients with a macro. Issues of implementing the digital compensator such as scaling of the output voltage, and the lack of a virtual ground are dealt with on the way to verifying the digital compensator design.

The digital PI compensator is then applied to compensate the three *dq0* control loops in the Vienna rectifier. Despite a simple implementation in LTspice, the averaged models can still offer tremendous help in determining what PI coefficients are suited for stabilizing the loops. In contrast, with switching models, it simply wouldn't be practical to extract the ac responses of the control loops in the Vienna rectifier.

Finally, PLECS simulations will be used to confirm my approach with the averaged models. This simulator employs a piece-wise linear approach that speeds simulation of switching models. It also has pre-defined blocks for modulators and compensators, which ease modeling of the circuit.

Cycle-By-Cycle Simulations Of The Vienna Rectifier

Average models are unbeatable in terms of simulation speed because they have no internal switching parts. They not only lend themselves well to running ac analyses but, owing to their large-signal capabilities, you can also explore large-signal transient responses.

However, when it comes down to assessing switching losses, current and voltage stresses or efficiency, you need to resort to a slower cycle-by-cycle model. I usually start with simple voltage-controlled switches for the power transistors, and, when all is working as expected, I add more comprehensive transistor or diode models to obtain reliable simulation results. That way, when a non-convergence issue occurs, I know that the latest addition is causing problems, and I can easily backtrack to test if a different model fixes the issue.

The complete circuit appears in Fig. 1, showing the six voltage-controlled switches, directly driven by the PWM section in the lower right corner. I have used two ideal switches (S_{a1} to S_{a6}) supplemented with a diode (D_7 to D_{12}) to approach the real implementation made of back-to-back transistors. The modulator is made of subcircuits X_7 to X_9 which compare the current setpoints reconstructed by the inverse *dq0* transform block X_3 with a 30-kHz 6-V (-3 V/3 V) peak-to-peak sawtooth voltage.

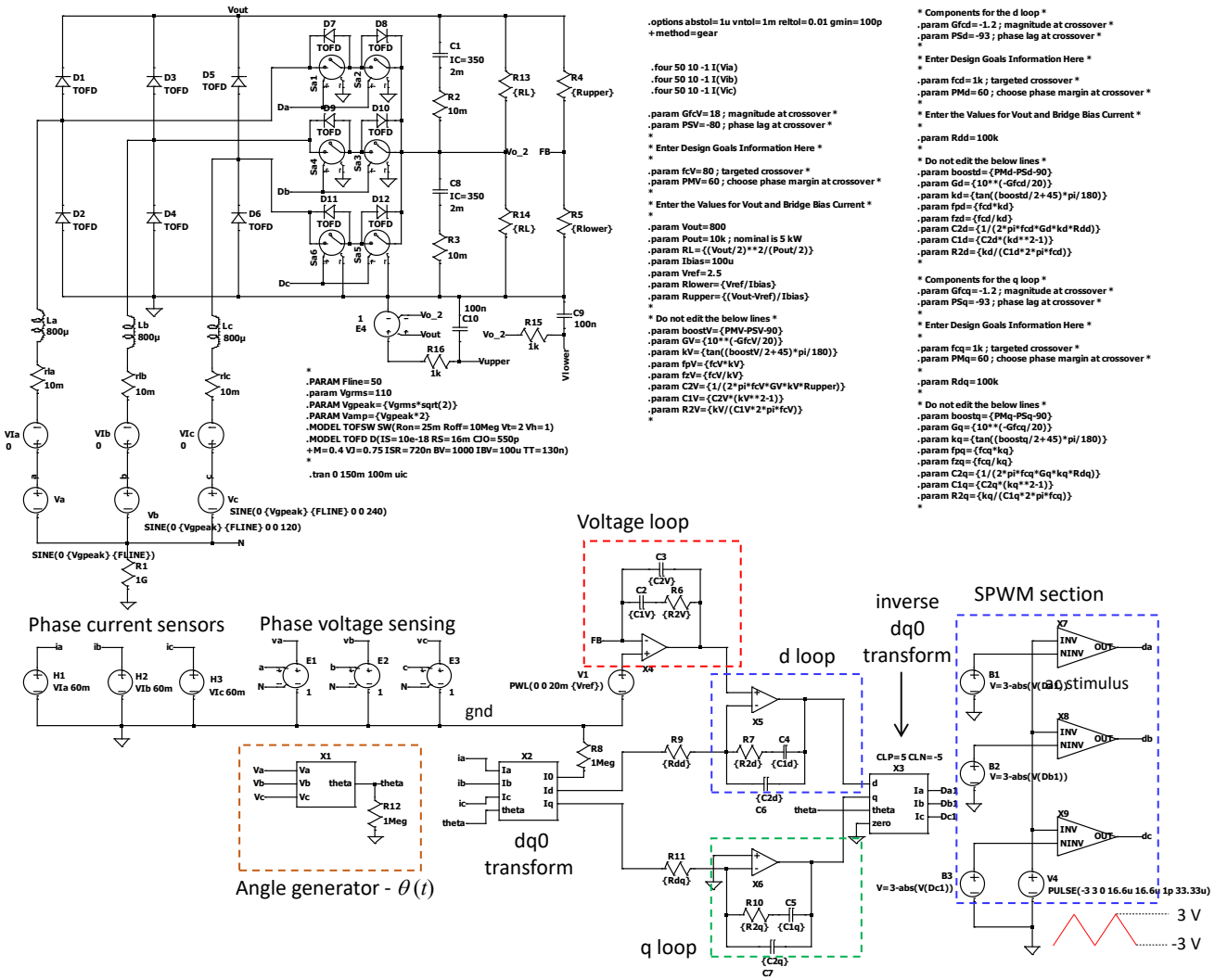


Fig. 1. The cycle-by-cycle simulation time for this three-phase rectifier with ideal switches is around 80 s which is good for such a complex power factor correction circuit.

The simulation results are available in Fig. 2 for operation at both input line extremes, with a 10-kW output power. The two rails are well balanced at 400 V each and the current absorbed from the grid is sinusoidal. The THD is around 3% in low-line operation and remains below 6% when the input voltage increases to 230 V (400 V rms line-to-line). The low-line performance obtained in Fig. 2 shows reasonably close agreement with that obtained in part 1 Fig. 18 for simulation using the averaged model. In that case, THD was 1.3%.

When the basic implementation works as expected, it becomes possible to use component models for the switching elements like transistors and diodes. For example, voltage-controlled switches can be replaced by SiC transistor subcircuits should you like to assess conduction and switching losses. However, before running long simulation cycles with unverified models, I recommend you thoroughly test the subcircuit you have identified as a potential model of the selected SiC transistor.

Reference 2 shows how reproducing the double-pulse test documented in the transistor datasheet represents a good way to validate the model or reject it. By assessing the energy dissipated during calibrated switching cycles, you can check whether the imported model reliably predicts losses with sufficient accuracy or, on the contrary, delivers unacceptable results.

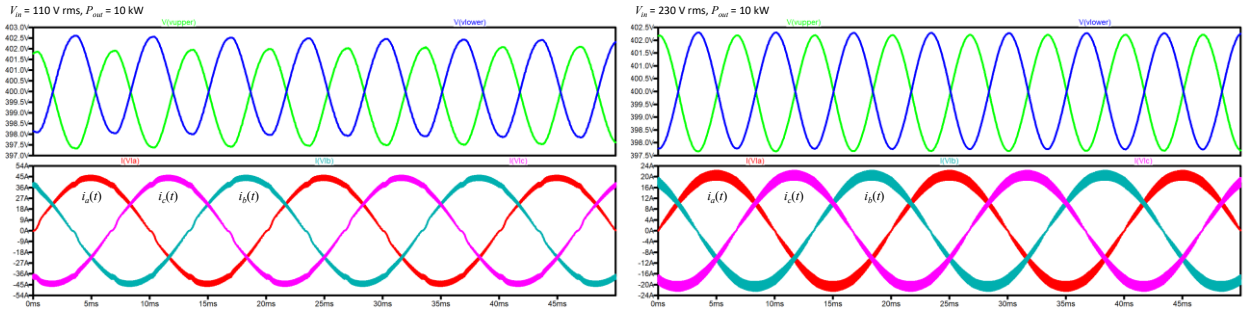


Fig. 2. The input currents in low-line operation (shown on the left) look good and exhibit a THD of 3.2%. In high-line conditions (shown on the right), the THD increases to 5.8%.

Nevertheless, having a good model on hand is not enough to ensure accurate simulation results as switching performance heavily relies on the power distribution network (PDN) and the way parasitics are distributed in the wiring environment. For instance, there's the equivalent series resistance and inductance of the dc-link capacitors. Failure to faithfully identify and include these parasitics can make your simulation useless.

However, extracting these terms from a PCB layout or a power module assembly can be extremely tedious and may hamper simulation time or convergence. You must therefore keep a good balance between circuit complexity and acceptable results which, in the end, will have to be verified on the bench.

Transient Response Of The Vienna Rectifier

Now that our averaged and switching circuits operate as expected in terms of input currents THD and voltage regulation, we can study the response to a load step. The transient response test is part of the stability analysis package but does not replace the loop measurement that we performed with the averaged model. It is complementary and will tell us if the adopted compensation strategy keeps the regulated variables (the two 400-V dc bus voltages) within adequate limits.

Fig. 3 gathers the transient responses of the Vienna rectifier operated at low line. The output power is stepped from 5 to 10 kW in a few microseconds, a very rapid transition. The left-side plots correspond to the averaged circuit implementing the six PWM switch models. The simulation is fast and very handy for checking different compensation strategies like changing crossover frequencies of the loops, adding or reducing phase margins and so on.

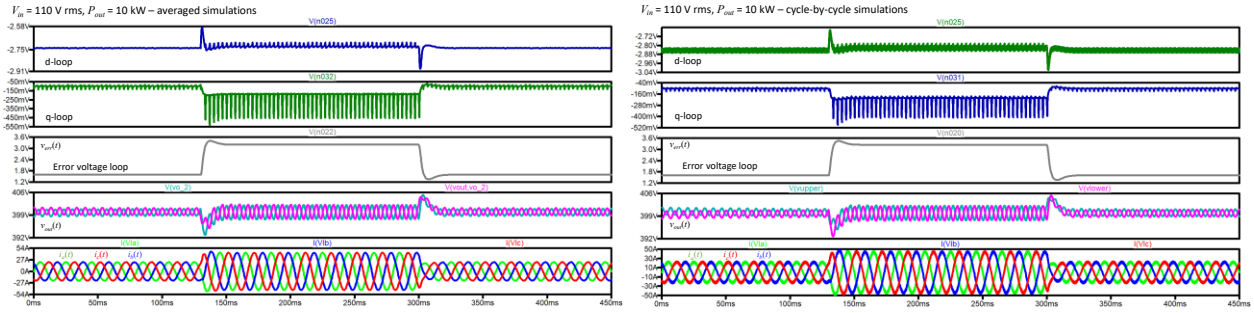


Fig. 3. Transient response simulation is an important exercise in checking loop stability. Here, you can see how stable the three loops are. You can also appreciate the good correspondence between the waveforms produced by the averaged models on the left and the switched models on the right.

As you can see in the waveforms, the voltage response is very smooth, with an error voltage loop response without significant overshoot. The *d*-loop compensator output exhibits a faster transient response, which is normal as its crossover frequency is 1 kHz, way above that of the voltage loop.

The *q*-loop does not significantly vary in this configuration where the reactive power is purposely kept to zero. The situation could be different in applications in which, rather than looking for a pure resistive emulation, you would impose a particular inductive or capacitive component, for grid-forming purposes as with a six-pack structure for example.

In Fig. 3, the right-side waveforms are collected on the switched model operated under identical conditions. LTspice does not provide practical tools for copy/pasting waveforms coming from different simulations but you can clearly see that the matching between averaged and switched models is excellent, confirming the validity of the averaged circuit.

The comparison between switched and averaged models is key when designing control loops. Before trusting the ac response and the adopted compensation strategy, it is important to validate the average model with cycle-by-cycle simulations but also by bench measurements. Once the model is validated, you can assess and trust the robustness of your design via worst-case analyses or Monte Carlo sweeps, to cite a few tools well adapted to averaged models.

To complete this comparison, Fig. 4 shows the same transient test but run in high-line conditions (400 V line-to-line) this time. Again, you can note an excellent relationship between the two plots.

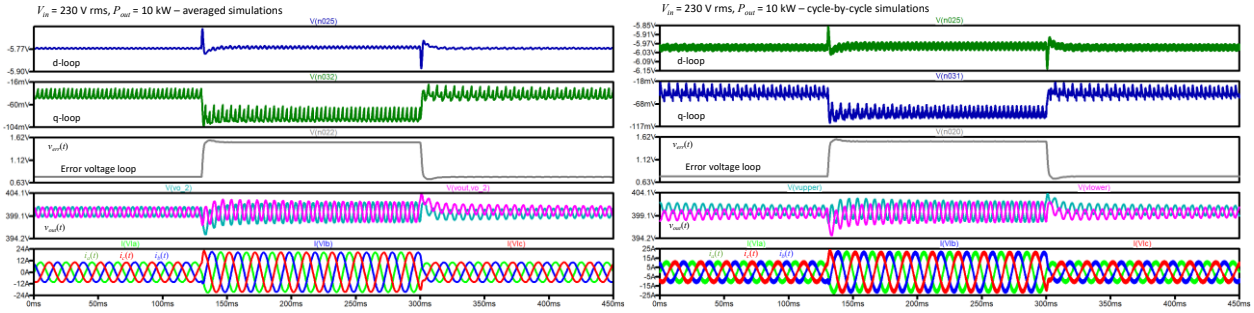


Fig. 4. When subjected to a load step in high-line conditions, the responses of the averaged models (left waveforms) and switched models (right waveforms) are almost identical.

Stabilizing The Loops—Analog Vs. Digital

In part 1, the three loops in the Vienna rectifier were stabilized with analog type 2 compensators using a macro to generate the filter coefficients. This form of active filter involves a pole at the origin and a pole-zero pair. The pole at the origin provides a near-infinity gain at dc—limited by the op-amp's open-loop gain A_{OL}—while the distance between the zero and the pole sets the amount of needed phase boost. The pole-zero pair is therefore adjusted to offer adequate phase margin while the mid-band gain forces the wanted crossover frequency.

Fig. 5 shows the typical response of this filter which, in this example, provides 50° of phase boost at 1 kHz with a 10-dB mid-band gain. The circuit is self-biased to adjust the op-amp dc output in a linear zone (e.g. ≈2 V for a 5-V supply voltage) and uses the classical *LoL*/*CoL* filter to inject an ac stimulus: when LTspice computes the dc operating point, it shorts all inductors and opens the capacitors. When the inductance is shorted, the loop involving E_1 biases the resistive divider R_{upper}/R_{lower} to ensure an adequate op-amp output. Considering the high open-loop gain of the op-amp, the voltage on the divider is adjusted at the microvolt level: for a 12-V output, E_1 biases the R_{upper} terminal at 11.999695 V for an ≈2-V output.

Needless to say, manually tweaking a dc source to stabilize the op-amp output at the desired level, would have been extremely tedious and long. With the approach used in part 1 with the auto-bias circuit, the loop is conveniently closed in dc but becomes open in ac, owing to the extremely low cutoff frequency set by the *LC* filter which excludes E_1 from the analysis. By plotting the voltage at node V_{out}, you obtain the expected Bode plot. You could also insert the ac source in series with E_1—then remove *LoL*/*CoL*—and you would obtain identical results but by probing V_{out}/V_{in} this time.

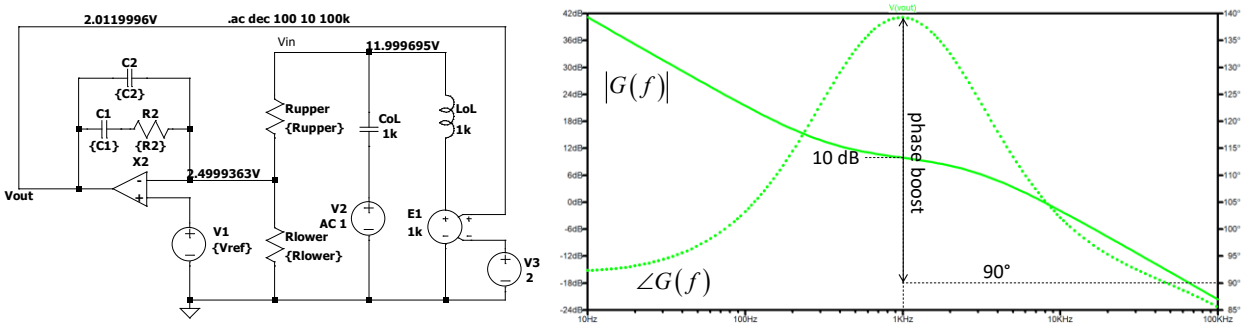


Fig. 5. A type 2 compensator built around an operational amplifier offers a potential phase boost up to 90°.

In a digitally-controlled system, a proportional-integral or PI compensator is usually adopted for stabilizing the *dq0* loops of our example. The transfer function of such filter is given below:[3]

$$G(s) = k_p\left(\frac{1+s\tau_i}{s\tau_i}\right) = \frac{1+s\tau_i}{s\frac{\tau_i}{k_p}} = \frac{1+\frac{s}{\omega_z}}{\frac{s}{\omega_{po}}} \qquad (1)$$

If you factor the zero in the denominator and implement an inverted zero, the above expression transforms into a more familiar shape:

$$G(s) = G_0\left(1+\frac{\omega_z}{s}\right) \qquad (2)$$

in which you have

$$G_0 = \frac{\omega_{po}}{\omega_z} \qquad (3)$$

$$\omega_z = \frac{1}{\tau_i} \qquad (4)$$

$$\omega_{po} = \frac{k_p}{\tau_i} \qquad (5)$$

The transfer function obtained in equation (2) is very close to that describing the type 2 compensator from Fig. 5 except that it does not feature a high-frequency pole. As a matter of fact, equation (2) describes a so-called type 2a compensator which is the analog PI compensator. The typical application circuit is shown in Fig. 6.

Please note the right-side macro in Fig. 6 which automates all the calculations for the poles and zero but also for the PI parameters, k_p and τ_i. In this example, the gain deficit exhibited by the power stage at 1 kHz is 20 dB and a 60° phase boost is needed. The macro places a zero at 577.35 Hz while the 0-dB crossover pole f_{po} is located at 5 kHz. If we extract the magnitude from (2), we find the desired 20 dB of gain brought by the compensator at 1 kHz, the selected crossover frequency f_c:

$$\left|G(f_c)\right| = \frac{f_{po}}{f_z}\sqrt{1+\left(\frac{f_z}{f_c}\right)^2} = \frac{5k}{577.35}\sqrt{1+\left(\frac{577.35}{1k}\right)^2} = 10 \text{ or } 20 \text{ dB} \qquad (6)$$

The phase response of this compensator at 1 kHz is also easily calculated:

$$\arg G(s) = 180° - \tan^{-1}\left(\frac{f_z}{f_c}\right) = 180° - \tan^{-1}\left(\frac{577.35}{1k}\right) = 150° \qquad (7)$$

The pole at the origin in the type 2 or type 2a combined with the inverting configuration provides a phase lag of -270° or 90° at dc. The phase boost at 1 kHz is thus 150 - 90 = 60° as we wanted.

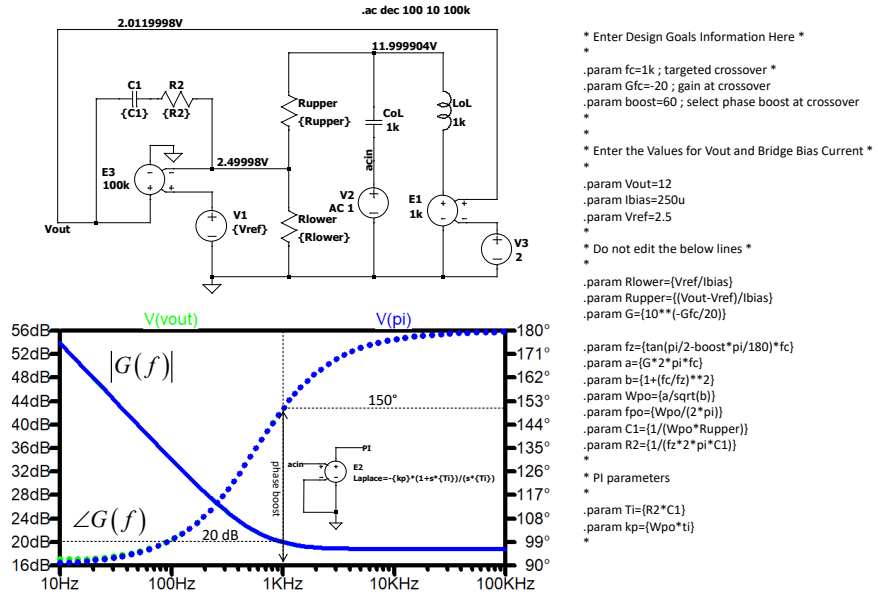


Fig. 6. A type 2a compensator does not feature a high-frequency pole as does its type 2 cousin.

Digital Compensators

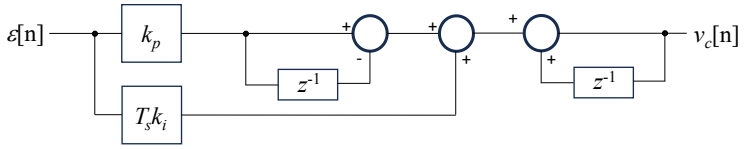
In three-phase active power factor correction projects, the control section usually relies on full digital implementation, including compensators. To build a digital filter like the PI, you start from the transfer function in the Laplace-domain and replace *s* by its equivalent using the backward Euler transform. You then obtain an expression in the *z*-domain that you convert into a difference equation, detailing how the stimulus ε is processed to form the sampled response v_c.

Fig. 7 shows a possible process applied to the original transfer function expressed in the Laplace-domain. Please note that other implementations are possible, e.g. resorting to the bilinear transformation rather than the backward Euler but it won't be covered here.

Once you have the final expression in the *z*-domain, modeling with a SPICE simulator can be done in different ways. One approach is to describe the z^{-1} function with a delay line as illustrated in reference 4. With this implementation, you can now assemble the delay blocks by following the flow. You end up with the circuit shown in Fig. 8.

You can see the z^{-1} blocks arranged with a few adders. Parameters k_p and k_i are computed by the left-side macro which asks for a 20-dB attenuation at 1 kHz. There is no boost computation in this simple example and the zero is arbitrarily placed at 100 Hz.

$$\frac{V_c(s)}{\varepsilon(s)} = k_p + \frac{k_i}{s} \quad \text{Backward-Euler} \quad s = \frac{1-z^{-1}}{T_s}$$



$$V_c(z) = k_p\varepsilon(z) + T_s k_i \varepsilon(z)\cdot z^{-1} - k_p\varepsilon(s)\cdot z^{-1} + V_c(z)\cdot z^{-1}$$
$$v_c[n] = k_p\varepsilon[n] + T_s k_i \varepsilon[n] - k_p\varepsilon[n-1] + v_c[n-1]$$

Fig. 7. The backward-Euler transform is applied to a PI Laplace transfer function to obtain the difference equation $v_c[n]$.

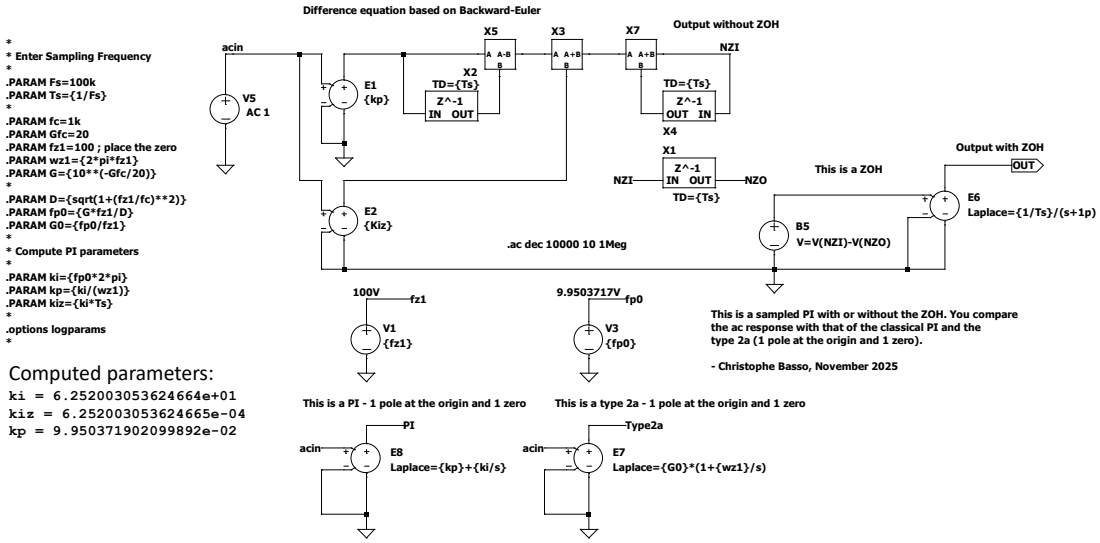


Fig. 8. You construct this simplified digital filter by assembling the blocks as described by the difference equation.

The parameters are passed to a Laplace expression with voltage-controlled source E_7 and E_8: the first one produces the frequency response of a type 2a described by equation (2) while the second determines the ac response of the PI controller with k_p and k_i. These two voltage-controlled sources are there to check if the digital PI transfer function matches the time-continuous expressions.

For the digital part, k_p is passed as a simple gain while k_i is scaled by the sampling period T_s to become k_{iz}. The model from Fig. 8 includes a zero-order hold (ZOH) for reconstructing the time-domain signal, e.g. if you drive an analog PWM or for observation purposes. Node NZI represents the direct output, before the ZOH.

If you ac-sweep the input of this filter and compare all three responses, you obtain the graph shown in Fig. 9. The 20-dB gain correctly translates into a 20-dB attenuation at 1 kHz. The three frequency responses are identical in magnitude up to 10 kHz while the digital phase starts diverging at around 1 kHz.

These are the effects of the ZOH which brings a delay of $T_s/2$ and a $\sin(x)/x$-shaped frequency response. Owing to the attenuation occurring at high frequencies, the sampled PI filter frequency response resembles that of a type 2 compensator despite the absence of a second pole in equation (2). We will take advantage of this characteristic in the PLECS implementation described later in the article.

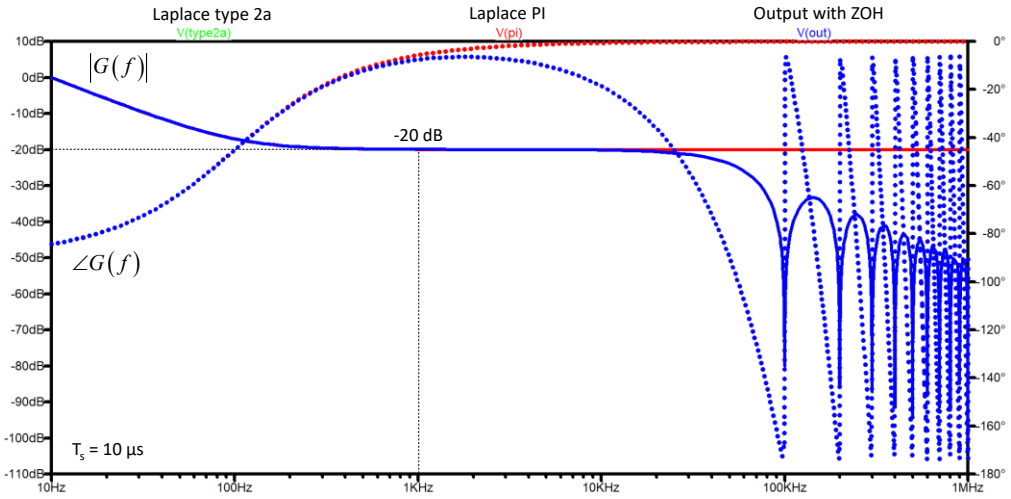


Fig. 9. You construct this simplified digital filter by assembling the blocks as described by the difference equation.

Once the subcircuit is encapsulated behind a symbol, we can capture a slightly different test circuit as represented in Fig. 10. In this illustration, you see a resistive divider sensing the 800-V bus voltage. In reality, specific fully isolated circuits exist to sense the voltage and safely bring a scaled down version for entering the microcontroller digital-to-analog converter (DAC). Nevertheless, the division ratio will be sensibly the same as here, 2.5/800 = 3.125m.

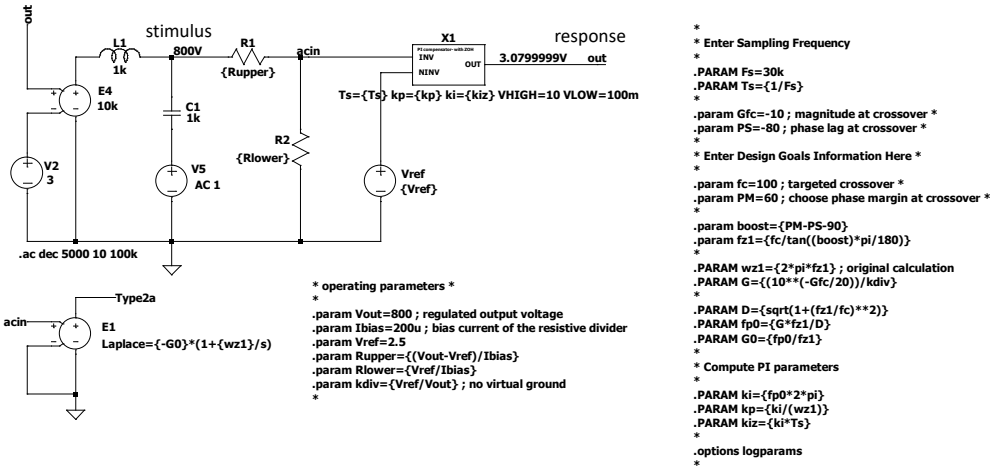


Fig. 10. The PI digital filter is now associated with a resistive divider for regulation purposes.

In the compensators shown in Figs. 5 and 6, there is a virtual ground brought by the op-amp operated in closed loop. Considering the high open-loop gain of the active element, both inverting and non-inverting pins share the same dc potential set by V_{ref}. If we see the 2.5-V reference voltage as a perfect dc source, any modulation applied at the compensator input will not affect the reference voltage.

As a consequence, if there is no ac modulation seen over V_{ref}, the ac potential of the inverting pin is also zero volts. Having 0 V ac across R_{lower} implies no ac current flowing into it, naturally excluding it from the poles-zeros calculation process: the resistive divider ratio does not affect the transfer function for *s* other than dc (when the local feedback is inactive).

In the digital PI circuit from Fig. 10, we do not have a virtual ground and the ac modulation entering the

subcircuit—node *acin*—is attenuated by the division ratio brought by our resistive divider or any external circuit scaling down the 800-V rail. The typical division ratio of 3.125m thus translates into a 50.1-dB attenuation we need to account for. This is the k_{div} parameter found in the macro located in the middle of the figure. Therefore, when measuring the control-to-output transfer function of the power stage, you either probe the V_{out} node and include the aforementioned attenuation or you probe after the resistive divider which naturally involves the division ratio.

The macro in Fig. 10 includes this particular point and now determines the zero position based on the required phase boost. In this example, a 10-dB gain is needed with a phase boost of 50° at 100 Hz. The sampling period is 33.3 µs and corresponds to the Vienna rectifier switching frequency of 30 kHz. The attenuation captured for the G_{fc} parameter assumes a measurement at the bus level (not after the resistive divider) and the macro will translate the needed gain accounting for k_{div} in the calculation of parameter *G*.

The PI filter will thus provide a total gain of around 60 dB at the selected 100-Hz crossover frequency. These 60 dB will undergo the 50-dB attenuation from the divider and, in the end, the gain at 100 Hz should be the expected at 10 dB. This is confirmed by the plot from Fig. 11.

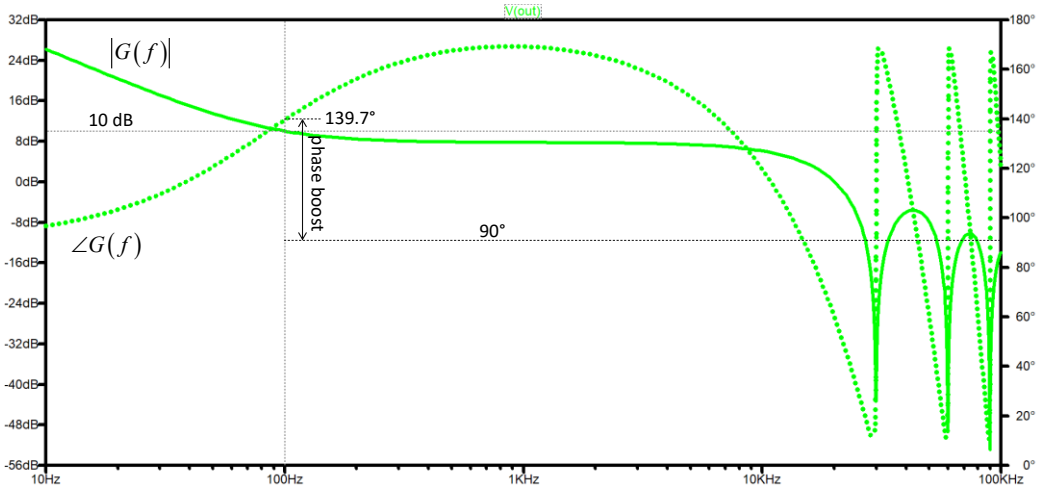


Fig. 11. The ac response matches what we entered in the macro: a 10-dB gain with a phase boost of 50°.

Implementation In The Vienna Rectifier

We now have a valid PI compensator that we can integrate into our averaged model. There is one thing to note with this delay line-based approach; the cycle-by-cycle transient simulation can be a challenge in terms of speed, convergence and integrity of results. That is the drawback of using many delay lines in these filters in a switching circuit.

An alternative consists of using a second-order Padé approximant, built with passive elements. It would have probably led to a more robust subcircuit but I did not explore this option and kept the actual subcircuit which worked well with averaged models.

Fig. 12 shows the whole circuit now equipped with digital filters. I did not show the automated macros, but they are part of the schematic diagram. The voltage loop is compensated for an 80-Hz crossover frequency while the *d*- and *q*-loops are stabilized for a 1-kHz crossover. The rest of the circuit is unchanged. We can now run a transient test with digital compensation and check if the whole converter is still stable.

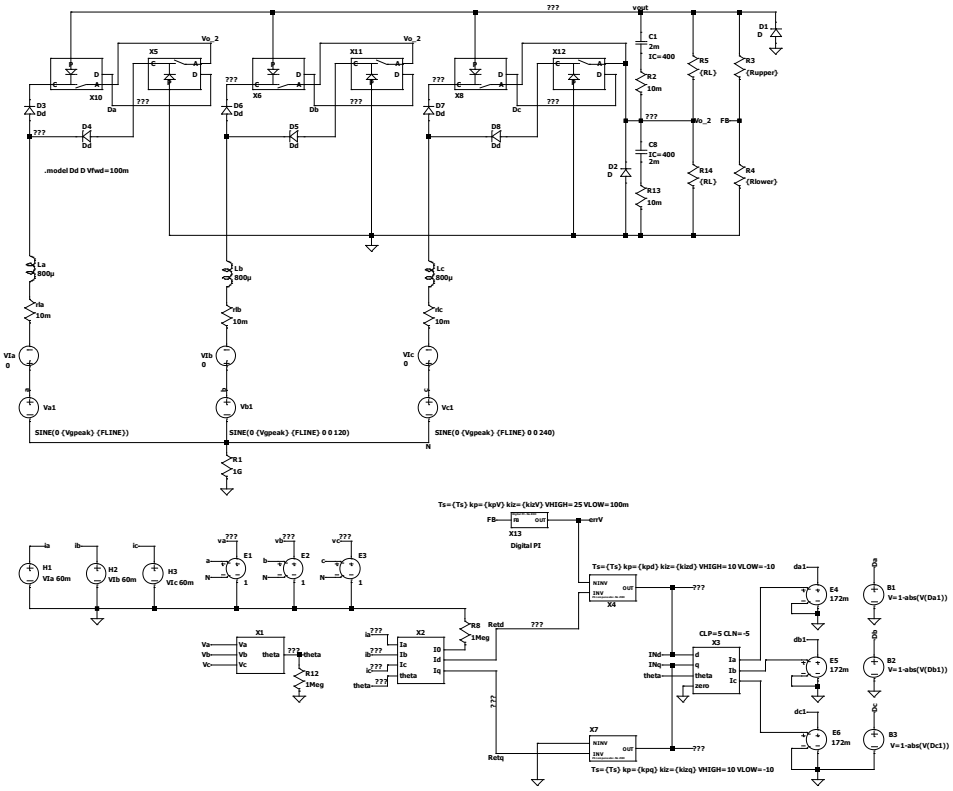


Fig. 12. The filters simply replace the op-amp and their passive components.

As confirmed in Fig. 13, the transient response is well stable at the two input levels, 110 (190 V line-to-line) and 230 V (400 V line-to-line). For this particular design, the macro did compute the following coefficients, displayed by pressing CTRL+L after LTspice has finished the simulation:

kpv = 3.072774086791324e+01 ; k_p for the voltage loop
kizv = 2.972477698697325e-01 ; k_{iz} for the voltage loop
kpd = 1.235433173601355e+00 ; k_p for the *d*-loop
kizd = 1.434266460277525e-01 ; k_{iz} for the *d*-loop
kpq = 1.039487470095485e+00 ; k_p for the *q*-loop
kizq = 1.206784831502161e-01 ; k_{iz} for the *q*-loop

These coefficients could now be entered into your digital compensator graphical user interface (GUI), provided the complete architecture and the gains involved in the adopted modulator were well modeled in the averaged circuit.

Reference 5 is an interesting video in which the speaker uses the auto-tuning capability of the program to automatically calibrate PI compensators for the best possible transient response. If the obtained results look promising, I prefer to first plot the control-to-output responses of the three loops then understand and drive the compensation strategy for the given characteristics. Linking the poles and zeros with analog or digital coefficients used in a PI or PID compensator, gives you the full control of the compensation strategy and lets you tweak knowingly in case fine-tuning is necessary.

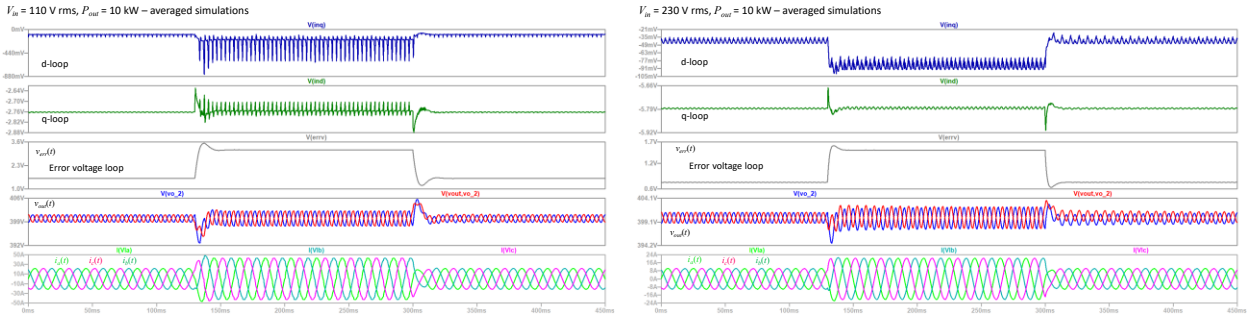


Fig. 13. The transient response is obtained in less than 30 s and confirms the stability of the converter at both input line levels.

Simulations With PLECS

It is possible to also run a similar analysis with PLECS, the power electronics simulation program from Plexim. Rather than wiring components pin-to-pin as in LTspice, the user will place ready-made blocks describing a given function like space vector modulation (SVM), PI or PID compensators in analog or discrete mode, transistors with a piece-wise linear (PWL) model involving switching losses and so on. Simulating my circuit with PLECS offers a good way to check that my conclusions on stability are correct.

Beat Arnet who is chief embedded engineer at Plexim U.S., kindly spent time to capture my LTspice Vienna rectifiers operated with analog and sampled compensators. Fig. 14 shows how to implement the three-phase rectifier with analog filters. Like in my LTspice macros, the component values around the compensator are automatically calculated.

PLECS also offers the ability to automate the parameters, giving you the neat option to change crossover or phase margin values. That way, you can immediately see the impact on the transient response and shape the loops to meet your design goals.

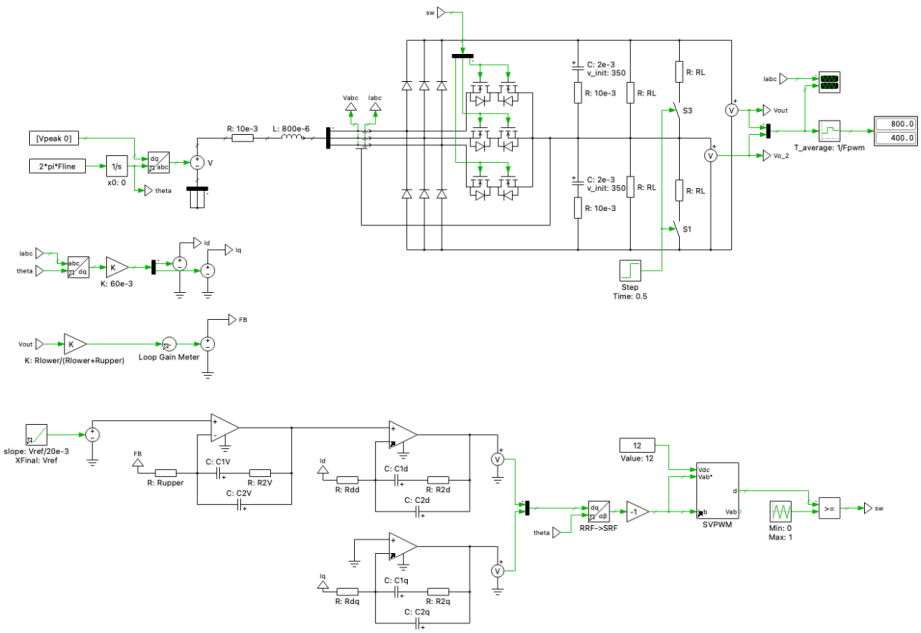


Fig. 14. This first version of the Vienna rectifier in PLECS implements analog compensators.

Fig. 15 shows the ac analysis performed by PLECS on the voltage loop. The compensation was set to meet 80 Hz with a 60° phase margin, which the left-side plot confirms. The two other loops, *d* and *q*, were stabilized for a 1-kHz crossover frequency.

On the right-side of the figure, the transient response is excellent, confirming a sound compensation strategy for the three active loops. In this experiment, the output power is instantaneously stepped from 50 to 100% of the nominal 10-kW output power. The simulation is fast and lasts a few seconds on my computer.

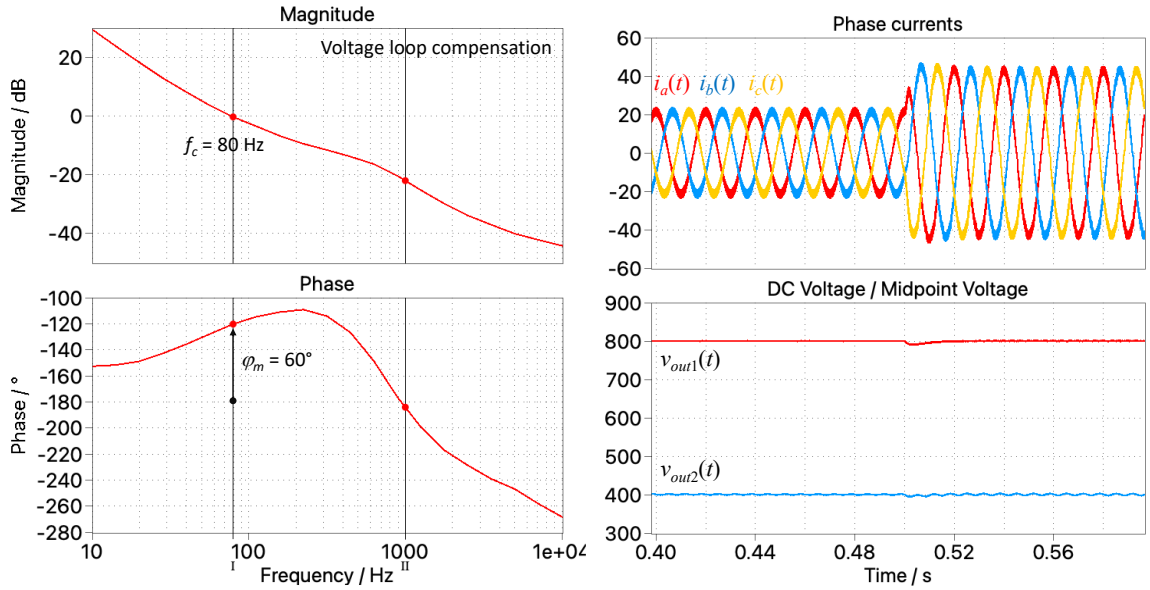


Fig. 15. The voltage loop is compensated for an 80-Hz crossover frequency with a 60° phase margin. In the waveforms on the right, the transient response is excellent with nice sinusoidal input currents.

The sampled version of the compensators appears in Fig. 16. The original PI compensator did not lead to a good transient response considering its flat high-frequency response. By involving a ZOH in the modulator section, the response in the upper spectrum was naturally rolled off and it cured the issue.

Reference 6 offers an interesting description of the various modulator schemes implemented in PLECS with their impact on the frequency response.

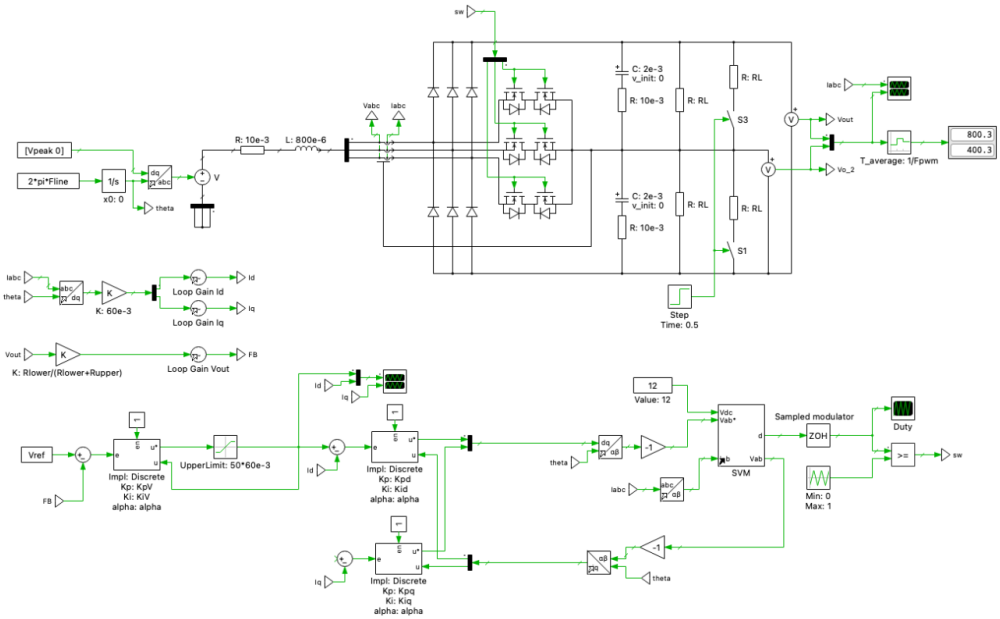


Fig. 16. The compensators in PLECS are now using sampled versions of a PI or type 2a compensator. An anti-windup scheme has been added in the voltage-loop block.

The original PI compensator implemented in Fig. 8 includes an integral term prone to saturation during start-up or transient events. An anti-windup structure is usually added to prevent this type of issue and a possible implementation is described in Fig. 17. In normal operation, the integrator operates in its linear region, away from the positive and negative supply rails. As a result, the input and output levels of the limiter are identical. By subtracting the two values, you obtain zero volts, making the circuit silent in steady-state operation.

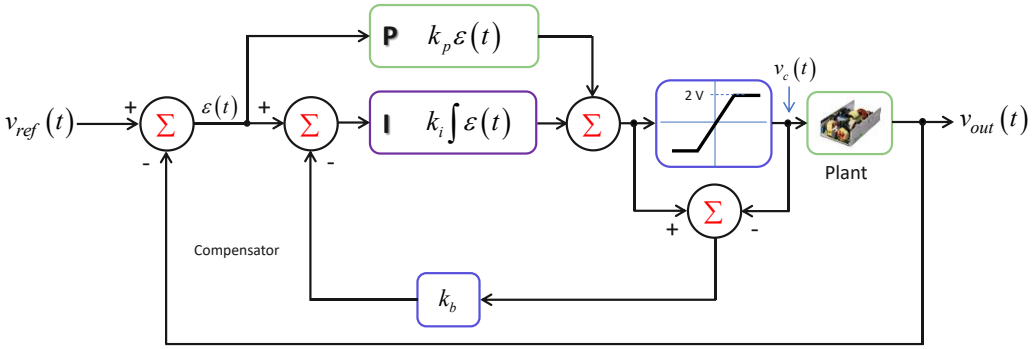


Fig. 17. An anti-windup system activates to prevent the integral term from saturating in a large transient event.

During the start-up sequence or if a large-signal transient event occurs, the integral term will quickly saturate as it integrates or accumulates the error signal over time. When its output exceeds a certain threshold, the limiter will start activating, naturally making the input-output difference a non-zero value.

The difference voltage will then be routed back to the integrator input to counteract the increasing error voltage at the integral block input. This compensation voltage is adjusted by block k_b whose value shall be properly adjusted to not degrade the overall response when the limiter is active.

Once the anti-windup block is added, we can check the start-up sequence. This is what Fig. 18 shows, with a

smooth ramp up of the output rails from zero to 400 V on the split rails. The input voltage is 110 V rms for the phase-neutral sources and the output power is fixed at 10 kW.

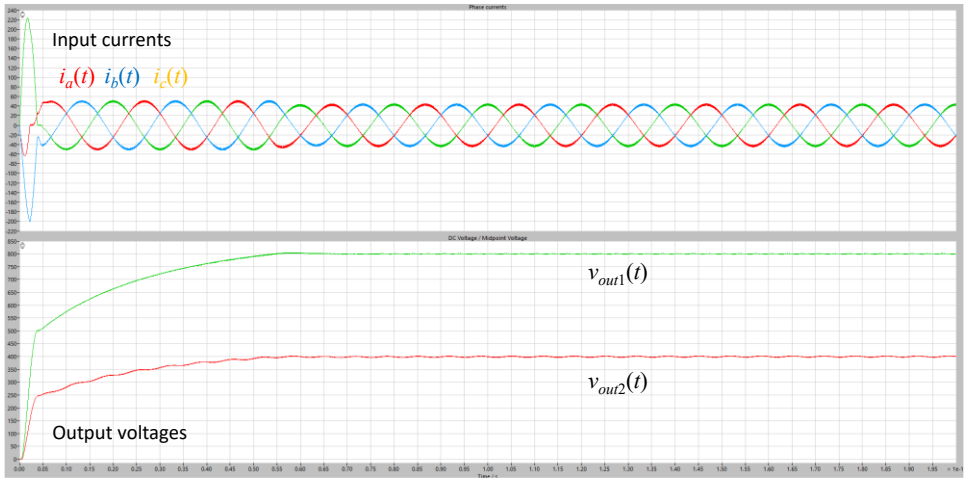


Fig. 18. The start-up sequence of the Vienna rectifier in PLECS is smooth without overshoot.

In a three-level inverter, the dc bus theoretically equally splits between the two output capacitors. However, dc imbalance can occur linked to unequal load currents, various drops in the power meshes, but also timing delays in the control algorithm. An extra loop is often implemented to prevent voltage runaway across the capacitors.

This can be done in different ways but I found the solution described in reference 7 figure 2-23 interesting: a low-gain regulator drives the zero input of the *dq0* inverse-transform block. The regulator adds a small modulation on the current setpoints to react to any discrepancy found between the low and high output rails. This option is not implemented here but I have tested in my LTspice 2025 collection of files[8] and it does the job very well, without significantly affecting the input currents THD figures.

Now that the simulation runs smoothly, we can check the transient response by stepping the output from 5 to 10 kW. The results shown in Fig. 19 confirm the good stability of the system with the digital coefficients determined in the LTspice simulations..

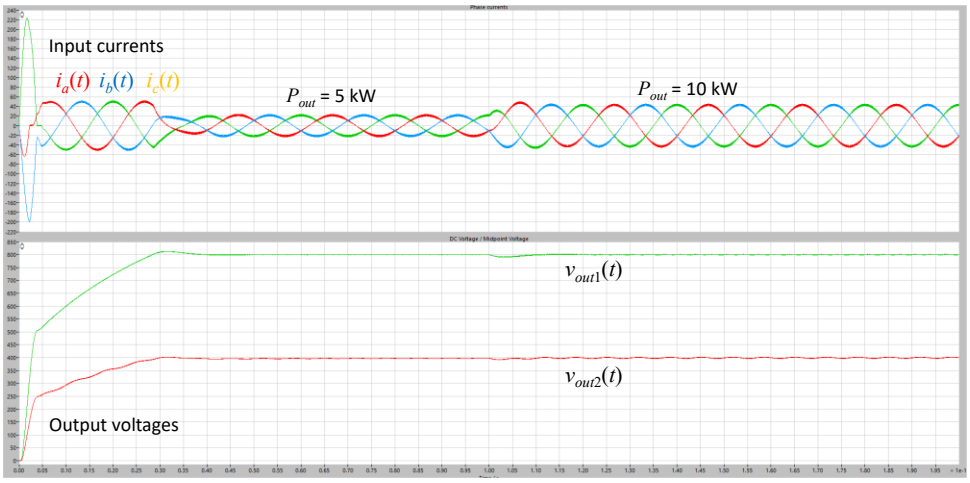


Fig. 19. The transient response of the digitally compensated Vienna rectifier simulated in PLECS is also excellent, with a well-controlled undershoot in the bus voltages.

Conclusion

This article closes this two-part story on how average models can help characterize the ac response of the three-phase Vienna rectifier. By first displaying the individual loop responses, I have stabilized the system with analog compensators, later replaced by digital versions.

The digital PI resorts to a delay line for the z^{-1} operator and delivers the expected response very quickly. Once coefficients are well calibrated, transient simulations confirm the converter is stable at the two input voltage extremes.

To further validate the modeling approach, I have shown that PLECS simulations using my automated macros, also deliver a stable response with both analog and digital circuits. This modeling technique can be extended to other PFC structures like the six-pack for instance, also available as a ready-made template in my summer 2025 file collection.

Acknowledgement

I would like to warmly thank Beat Arnet from Plexim for the friendly discussions we've had around these simulations and the circuits he kindly built in PLECS. Merci!

References

1."Stabilizing The Loops Of A Three-Phase Vienna Rectifier—Part 1" by Christophe Basso, How2Power Today, March 2026.

2. "Double-Pulse Test Helps Validating SiC SPICE Models" by Christophe Basso, PCIM News Platform, February 2025.

3. *Designing Control Loops for Linear and Switching Power Supplies* by Christophe Basso, Artech House, 2012.

4."Generic Average Modeling and Simulation of Discrete Controllers" by Sam Ben-Yaakov and Danny Adar, Applied Power Electronics Conference, Anaheim, CA, 2001.

5. "PID Control of a Vienna Rectifier-Based Power Factor Corrector," Mathworks training video, March 2021.

6. "PWM Modulator Delay: Analysis verified by PLECS" by B. Arnet, Plexim knowledge base, March 2026.

7. "Three-Phase 30-kW Vienna PFC Reference Design," Microchip, 2020.

8. Summer 2025 LTspice Files Collections, free 250+ simulation templates from the author.

About The Author

Christophe Basso is a business development manager with Future Electronics, a member of the power team and covering EMEA. Previously, he was a technical fellow with onsemi for 24 years where he originated numerous integrated circuits. SPICE simulation is also one of his favorite subjects and he has authored two books on the subject. Christophe's latest work is "An Intuitive Guide to Compensating Switching Power Supplies". Christophe received a BSEE-equivalent from the Montpellier University, France and an MSEE from the Institut National Polytechnique de Toulouse, France. He holds 25 patents on power conversion and often publishes papers in conferences and trade magazines.

For further reading on compensating power supplies, see the How2Power Design Guide, locate the "Design Area" category and select "Stability". Also, see "Modeling and Simulation". For more on PFC circuits, see the "Popular Topics" category and select "Power Factor Correction".