Autotuning Techniques for Digitally-Controlled Point-of-Load Converters with Wide Range of Capacitive Loads

M. Shirazi, R. Zane, D. Maksimovic  
Colorado Power Electronics Center  
University of Colorado at Boulder  
[mariko.shirazi, zane, maksimov]@colorado.edu

L. Corradini  
DEI  
University of Padova - Italy  
luca.corradini@dei.unipd.it

P. Mattavelli  
DTG  
University of Padova – Italy  
mattavel@dei.unipd.it

Abstract—This paper addresses auto-tuning of digital controllers for point-of-load (POL) switching converters with wide range of capacitive loads. Two auto-tuning methods are considered with particular attention given to robustness and feasibility. The first method is derived from the well known relay-feedback autotuning technique, where specific frequencies are excited to gain information on the power stage. In the second, system-identification based method, compensator parameters are computed based on on-line identification of the power stage frequency response. The tuning techniques proposed in this paper have been specifically developed to handle wide capacitance and ESR range, and important extensions of the basic algorithms are implemented in order to face practical issues such as limit cycling conditions, output voltage tolerance specification, closed-loop bandwidth maximization and phase margin constraints. Simulation and experimental results on a 12-to-1.5 V, 9 A, 200 kHz POL converter are provided to show the effectiveness and to compare the considered techniques.

I. INTRODUCTION

Various performance or system gains using digital control of high-frequency switched-mode power supplies (SMPS) have recently received increased attention [1-5]. In this paper, we focus on an important practical problem in the design and deployment of point-of-load (POL) converters: the impedance of the capacitive load is often unknown at design time, and can furthermore vary significantly due to component tolerances or temperature variations. Given very wide power-stage variations, standard analog-controlled POL converters may require manual re-design or re-tuning at the time of deployment, or accept penalties in closed-loop dynamic responses in order to meet acceptable stability margins. In a digitally controlled converter, a block diagram of which is shown in Fig. 1, a key advantage is the possibility of autotuning the controller parameters to adapt to the specific power stage. Efficient and robust implementation of digital regulators with embedded tuning capabilities could be a significant breakthrough for digital control in power electronics. The purpose of this paper is to describe the operation, implementation, experimental performance results and comparison of two techniques addressing autotuning for POL converters with wide range of capacitive loads: (1) a relay-based autotuning method based on [6-12], and (2) an auto-tuning method based on system-identification (system-ID) techniques [13-16]. Autotuning objectives and the design space are introduced in Section II. Sections III and IV describe application of the two autotuning techniques along with experimental results. Section V discusses the effectiveness of each of the autotuning methods, and conclusions are presented in Section VI.

II. AUTOTUNING OBJECTIVES

Figure 1 shows a synchronous buck point-of-load converter with a voltage-mode digital controller. In the power stage, we can assume that nominal values of the input voltage \( V_{\text{in}} \) and the filter inductance \( L \) are known. The output filter capacitance \( C \) and the equivalent series resistance (ESR) can vary in wide ranges. Upon start-up, after the output voltage reaches regulation using a slow default compensator, a short auto-tuning process is initiated. Once the tuning algorithm updates the compensator parameters, a “power good” signal is generated to indicate that the POL is ready for normal operation. The goal of the tuning algorithm is to determine the compensator parameters: the zeros \( z_1, z_2 \), and the gain \( K \), to achieve high-performance voltage regulation over all values of \( C \) and ESR of interest. Specifically, the performance goals are as follows:

- Closed loop operation with specified phase margin \( \Phi_{\text{spec}} \)
- As high closed-loop bandwidth as possible, i.e. as high loop gain cross-over frequency \( f_c \) as possible
- Output voltage within specified tolerance band \( \pm \Delta V/V \) during auto-tuning process
- Operation without limit cycling oscillations, including satisfying a necessary condition in terms of the compensator integral gain [17,18],

\[
K_{I} V_{\text{in}} \leq 1.
\]
To illustrate the auto-tuning objectives in the presence of wide C and ESR variations, we consider the representative experimental POL example (12-to-1.5 V, 9 A, 200 kHz) shown in Fig. 1. The digital controller is implemented on a Xilinx FPGA development system with a 25 MHz clock. The A/D converter samples the output voltage once per switching period with LSB resolution as low as 1 mV.

To represent possible C and ESR values of interest, it is useful to consider the C-ESR design space plot shown in Fig. 2. Two sets of boundary lines define the design space, i.e. the ranges of C and ESR of interest. The first set of boundaries is related to the output voltage ripple of the power stage. The magnitude of the ripple, which is a function of both C and ESR, cannot be influenced by the controller. This means that limits must be placed in order to make sure that the design meets the ripple specifications. Two dashed lines in Fig. 2 define the 45 mV peak-to-peak ripple boundary, which corresponds to half of a ± 45 mV, or ± 3% allowed tolerance band. The second set of boundaries are related to the filter resonant frequency \( f_0 \) and the ESR associated with the ESR zero of the power stage. We assume that \( f_0 \) cannot be greater than \( f/20 \); this places the maximum allowed \( f_0 \) at half the usual bandwidth target of approximately \( f/10 \); the corresponding boundary for this assumption is represented by a dashed horizontal line in Fig. 2. Finally, it is reasonable to add a lower limit for the possible values of \( f_{esr} \), \( f_{esr} > 2f_0 \) for example, which is conservative for the types of filter capacitors commonly used in POL applications. The corresponding boundary is the oblique dashed line in Fig. 2. The intersection of the areas defined by the boundaries discussed above defines the C-ESR design space shown in Fig. 2. Within the design space, the C-ESR area “covered” by a given auto-tuning method will be a direct measure of its effectiveness.

### III. RELAY-BASED AUTOTUNING METHOD

The first autotuning approach discussed here is based on the relay-feedback [6-10], which employs a relay block in the feedback loop as shown in Fig. 3a. The relay block is essentially a 1-bit quantizer, and its implementation in the digital controller is very simple: the output is \( +D_r \) for all positive error signals and \( -D_r \) otherwise. Since the relay block output cannot be equal to zero, a limit cycle oscillation at frequency \( f_{osc} \) will arise in the system, with \( f_{osc} \) satisfying the following condition:

\[
\frac{4D_r}{\pi A_{osc}} T(f_{osc}) = -1. 
\]  

In (2) the term \( 4D_r/(\pi A_{osc}) \) represents the describing function of the relay, \( A_{osc} \) being the output voltage oscillation amplitude, while \( T \) is the linear part of the system’s loop gain, i.e. \( T = G_c G_{vd} \). The key point in (2) is that for a given compensator \( G_c \), \( f_{osc} \) is determined by the power stage transfer function \( G_{vd} \). The oscillating frequency thus carries information about the process and can be used to identify some of its properties for tuning purposes. By iteratively adjusting the PID parameters and measuring the corresponding oscillating frequencies, the tuning algorithm shapes the loop gain in order to achieve the desired closed loop specifications. Other limit cycling-based tuning techniques were recently proposed in literature [11,12]. In [6] the autotuning objective was to meet a target closed-loop bandwidth \( f_c \) and a target phase margin \( m \) specification. In order to properly handle as wide area as possible in the C-ESR design space, we have implemented several important modifications: 1) More precise and robust setting of the closed-loop system bandwidth; 2) Dynamic selection of the closed-loop bandwidth, which is now maximized while still satisfying the no-limit cycling requirements (1) for proper system operation; 3) Efficient limitation of the output voltage
oscillation amplitude during the tuning process, thus allowing the tuning to be performed within the output tolerance band specifications.

A. Relay-feedback based tuning algorithm

The flow chart of the tuning algorithm is illustrated in Fig. 3b. The tuning process consists of three phases, denoted as phase A, B and C, the purpose of which is to determine the three PID parameters \( z_1, z_2 \) and \( K \), respectively. For each phase the appropriate structure of \( G_c(z) \) is selected in order to induce the system oscillation towards particular frequencies.

During phase A the power stage is forced to oscillate at the resonant frequency \( f_0 \) of the output filter by letting \( G_c(z) \) be the transfer function of a digital integrator. As the phase lag of the integrator equals \(-90^\circ\), (2) implies:

\[
\angle G_{vd}(f_{osc}) = -90^\circ, \tag{3}
\]

meaning that \( f_{osc} = f_0 \). The corner frequency of the power stage can thus be measured, and the first PID zero is tuned so that \( f_{z1} = f_0 \). At this point, it is important to note that a practical auto-tuning algorithm must limit the amplitude of the output voltage oscillation so that the output voltage always stays within the specified tolerance band, set here to \( \pm 45 \text{ mV} \), or \( \pm 3\% \). Equation (2) suggests that \( A_{osc} \) is high when the loop gain is also large, so the worst-case condition for \( A_{osc} \) happens during phase A, when the system is oscillating at the filter resonant frequency. From (2) it is also clear that for a given loop gain \( T \), \( A_{osc} \) is proportional to the amplitude \( D_r \) of the relay output. From these worst-case considerations the maximum allowed value for \( D_r \) can be set to limit \( A_{osc} \) to a maximum value \( A_{osc,max} \):

\[
D_{r,max} = \frac{\pi}{4V_{in,max}} \frac{A_{osc,max} (ESR_{min} + R_L)}{f_s L} \tag{4}
\]

If \( D_r < D_{r,max} \) the condition \( A_{osc} < A_{osc,max} \) is satisfied everywhere in the design space, allowing the tuning to be safely performed under all circumstances.

After completion of phase A, a maximum target bandwidth \( f_{c,max} \) is then determined according to the following criterion:

\[
f_{c,max} = \min\left(\frac{f_s}{10}, 4 f_0\right). \tag{5}
\]

A first upper-bound for \( f_{c,max} \) is placed at \( f_s/10 \), which is the usual target in conventional analog voltage-mode control. Equation (5) also sets the initial target bandwidth \( f_c \) that will be used for the first tuning iteration. Consequently, a second limit, proportional to \( f_0 \), is introduced in (5) in an attempt to satisfy the no-limit-cycling condition (1) in the first iteration over the target bandwidth. With \( f_c \) so determined, phase B is started, the purpose of which is to tune the second PID zero \( z_2 \) to meet the phase margin specification \( \alpha_{m} \) at \( f = f_c \). This task is addressed by iteratively adjusting the position of \( z_2 \), using the bisection method, until the proper phase relationship is satisfied within the loop. The iterative \( z_2 \) tuning requires only frequency measurements of the induced limit-cycling oscillations. After \( z_2 \) is properly positioned phase C is executed to determine the overall PID gain \( K \) that achieves the selected target bandwidth \( f_c \). The PID gain \( K \) is determined by computation,

\[
K = K_{corr} \frac{2\pi f_c f_s}{V_{in} f_0 f_s}, \tag{6}
\]

which follows from asymptotic approximation of the system’s loop gain magnitude Bode diagram. The coefficient \( K_{corr} \) is used as a correction term whenever poles and zeros are very close in frequency, affecting the asymptotic approximation. The frequency \( f_c \) is a function of the loop gain poles and zeros and their relative locations. An estimation of the frequency \( f_{est} \) associated with the ESR zero is thus required in order to evaluate \( K \). Information about the actual location of \( f_{est} \) can be inferred from the position of \( z_2 \) after phase B: the lower \( f_{est} \) is, the higher \( z_2 \) is placed in frequency since a smaller phase boost is needed to achieve the target phase margin. This basic observation can be expressed as the following estimation rule:

\[
f_{ESR, est} = f_c \cot(m_{\alpha} + 2\pi f_c \Delta t - \theta_{PID}(f_c)) . \tag{7}
\]

where \( \Delta t \) represents a delay time that accounts for the A/D conversion time, the PID computational delay and the DPWM delay; \( \theta_{PID}(f_c) \) is the PID phase lag at the selected target bandwidth \( f_c \). Equation (7), though approximated, is able to predict whether \( f_{est} \) is close to or far from \( f_c \) and if its value has to be considered in (6). The computation of the gain \( K \) in (6) differs significantly from the approach presented in [6], which relies on the measurement of the output voltage oscillation amplitude \( A_{osc} \). Due to the filtering action of the power stage, \( A_{osc} \) can easily drop close to the ADC LSB resolution, thus heavily affecting the measurement. With the proposed approach the determination of \( K \) is not affected by quantization effects at all.

At the end of phase C the PID compensator is fully tuned, subject to checking the no-limit-cycling condition (1) as follows:

\[
\alpha_{min} < K_{i} V_{in} < \alpha_{max} \tag{8}
\]

The safety range \([\alpha_{min}, \alpha_{max}]\) represents a trade-off between a fast closed-loop response and the need to avoid limit-cycling oscillations [17,18]: as the integral term \( K_i \) is directly related to the overall PID gain, (8) can be forced through an iteration that re-starts phases B and C with a different target bandwidth \( f_c \) if (8) is not satisfied. The tuning process is re-started with a lower target \( f_c \) if \( K_{i} V_{in} > \alpha_{max} \), while \( f_c \) is increased whenever \( K_{i} V_{in} < \alpha_{min} \). Simulation results indicate \([0.4,0.5]\) as a suitable safety range for \( K_{i} V_{in} \). When \( \alpha_{min} < K_{i} V_{in} < \alpha_{max} \) the tuning process terminates, the relay block is removed from the feedback loop and the system goes into regulation.
B. Relay feedback based experimental results

The relay-based autotuner was realized with Xilinx System Generator for DSP. This tool integrates itself in the Matlab/Simulink environment and consists of a Simulink blockset that can be used to build a block-diagram description of the system, run fixed-point arithmetic simulations and generate the corresponding HDL code for subsequent FPGA synthesis and implementation. The algorithm has been experimentally tested in the POL prototype described in Section II, with two different sets of capacitive loads: $C = 3 \, \mu F$ ($2.2 \, \mu F$ electrolytic, $400 \, \mu F$ ceramic, $200 \, \mu F$ tantalum) and $C = 600 \, \mu F$ ($400 \, \mu F$ ceramic, $200 \, \mu F$ tantalum).

The experimental output voltage waveform during a typical tuning process (target $m_0=50^\circ$) is shown in Fig. 4. The entire tuning process takes about 27 ms, after which the system goes into regulation. The output voltage perturbation entirely lies within the specified tolerance band. Figure 5 shows experimental post-tuning 0 A → 9 A load step-ups transients for the 3 mF case and for the 600 μF case respectively: a high performance transient behavior is observed for both capacitive loads, and the closed-loop bandwidth is correctly maximized according to the resonant frequency of the power stage, as defined by (5). The experimental PID parameters were employed to evaluate the system loop gain, using an average model for the power converter. The loop gain plot, shown in Fig. 6 for the 3 mF case, indicates good agreement between the target parameters and the experimental values.

IV. SYSTEM-ID BASED AUTOTUNING METHOD

The second auto-tuning approach discussed in this paper is based on the system-identification approach [13-16]. The converter frequency response, in real and imaginary form, is obtained by injecting a Pseudo Random Binary Signal (PRBS) to the control input of the converter, cross-correlating the output voltage perturbations with the input PRBS using the Fast Walsh Hadamard Transform (FWHT), and then taking the FFT of the resulting impulse response. The amplitude of the output voltage perturbations during PRBS injection can be directly controlled by the amplitude of the duty cycle perturbations.

As opposed to the computationally intensive direct digital design algorithms described in [16], a much simpler autotuning of PID parameters is proposed in this paper. The proposed tuning algorithm operates directly on the real/imagninary frequency response data and requires significantly less complex computations, thus allowing a practical implementation on an FPGA.

A. System-ID based autotuning algorithm

The flow chart of the tuning algorithm is illustrated in Fig. 7. The first steps are to obtain the real/imaginary frequency response data through PRBS injection, FWHT and FFT, as described above. The next step is to identify the corner frequency $f_0$ by searching the real/imaginary data for the frequency where the phase crosses $-90^\circ$, i.e., where the real part becomes negative. As in the relay-based autotuner, the first compensator zero is placed at $f_c$.

The algorithm next identifies the maximum possible crossover frequency $f_{c,max}$ as the minimum of $f_c/10$ and the frequency beyond which identification results become
significant corrupted by quantization noise, $f_{sysID,max}$. The latter can be estimated using Bode straight-line asymptotes:

$$f_{sysID,max} = f_0 \sqrt{\frac{\Delta d_{PRBS}}{q_{AD}}},$$  \hspace{2cm} (9)

where $G_{vdo}$ is the DC gain measured from the frequency response, $\Delta d_{PRBS}$ is the amplitude of the duty cycle perturbations, and $q_{AD}$ is the A/D LSB resolution. Once $f_{c,max}$ is known, the target crossover frequency $f_c$ is initialized here and the algorithm enters the loop to iterate $f_c$. For each value of $f_z$ the algorithm computes the phase margin at $f_c$ and compares this to the phase margin specification. A bisection method is used to iterate the location of the second zero if the specification is not met. If it is impossible to achieve the specified phase margin, then $f_c$ is decreased and the process is repeated. Once an achievable $f_c$ is found, the PID gain $K$ is determined from the magnitude of the loop transfer function at that frequency.

In the search for $f_z$, the algorithm first determines whether the second zero must be placed above or below the target crossover frequency. This is determined by computing the phase margin if the second zero were placed exactly at the target crossover frequency. The phase margin is tested using the resulting loop transfer function data. If the phase margin is low then the second zero must be placed lower than the target crossover frequency and vice-versa. The appropriate maximum and minimum allowed locations for the second zero are then computed for each case. Practical limits are placed on the maximum zero location for the case of $f_z > f_c$ and the minimum zero location for the case of $f_z < f_c$. The maximum zero location for the case of $f_z < f_c$ is subject to no-limit-cycling restrictions that follow from (1) using Bode straight-line asymptotes. Look-up tables are used in the computations for the discrete zeros $z_1$ and $z_2$ and for the complex exponential $e^{j2\pi f_z T_s}$. To improve noise immunity, the measured frequency response data is filtered by averaging over a window centered at $f_c$ before being used in computations.

### B. System-ID based experimental results

The system-ID based autotuner was fully coded in Verilog HDL and implemented on the Xilinx Virtex4 FPGA with a 25 MHz clock. Table I shows the equivalent gate count and kilobytes of memory required to implement the system-ID based autotuner. It is interesting to note that hardware requirements are relatively modest in spite of the computations involved and the fact that further algorithm and code optimizations are possible.

The system-ID based autotuner has been tested on the experimental POL prototype described in Section II using the same 3 mF and 600 µF capacitive loads as described in Section III.B. The A/D LSB resolution is 7.8 mV. The output voltage tolerance band is ±65 mV, or ±4.3% The target phase margin is $\phi_m=50^\circ$.

Figure 8 shows the output voltage during PRBS injection. Two 1024-point PRBS periods are injected, lasting a total of 10 ms. The remainder of the autotuning consists of computations that do not require any further measurements. With the 25 MHz clock, the system-ID portion takes 3 ms while the tuning algorithm itself takes less than 100 µs.

### Table I

<table>
<thead>
<tr>
<th>Module</th>
<th>Total equivalent gate count</th>
<th>RAM</th>
<th>ROM</th>
</tr>
</thead>
<tbody>
<tr>
<td>System-ID</td>
<td>9600</td>
<td>10 kB</td>
<td>0.5 kB</td>
</tr>
<tr>
<td>Tuning</td>
<td>31600</td>
<td>2.5 kB</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>41200</td>
<td>10 kB</td>
<td>3.0 kB</td>
</tr>
</tbody>
</table>

![Fig. 7 - System-ID method: Algorithm flowchart](image)

![Fig. 8 - System-ID method: Output voltage during PRBS injection](image)

![Fig. 9 - System-ID method: Experimental frequency response](image)
Figure 9 shows experimental frequency responses along with the curve-fit average model for the two capacitive loads. Table II shows the results of the autotuning – PID parameters, target crossover frequency, and predicted phase margin. In addition, the table shows the crossover frequency and stability margins obtained by applying the tuned PID to the curve-fit average model. It can be seen that for both capacitive loads, the crossover frequency and phase margin from the curve-fit averaged models match quite closely to the target crossover frequency and predicted phase margin from the System-ID data. The 0-to-9 A step load transient responses are shown in Fig. 10.

The System-ID algorithm relies on computations based on frequency-response data, which can be corrupted by quantization noise, especially at higher frequencies, as shown in Fig. 9. To examine robustness and repeatability of the autotuning results, the autotuner was run for each capacitive load 100 times. Tables III.A and III.B list the mean, maximum, and standard deviation for identified resonant frequency \( f_0 \), target crossover frequency \( f_c \), predicted phase margin \( m_0 \), and PID parameters \( K, f_z1, \) and \( f_z2 \) for \( C = 3 \) mF and \( C = 600 \) μF respectively. In both cases, the algorithm selects the same \( f_0 \) and close to the same target \( f_c \) every time. Differences in \( f_z2 \) and consequently \( m_0 \) arise when \( m_0 \) is close to the target for a given \( f_z2 \), meeting the target at that \( f_z2 \) for some data sets but requiring a smaller \( f_z2 \) for others.

**TABLE II**

<table>
<thead>
<tr>
<th>SYSTEM-ID METHOD: AUTOTUNING RESULTS</th>
<th>( C = 3 ) mF</th>
<th>( C = 600 ) μF</th>
</tr>
</thead>
<tbody>
<tr>
<td>PID Gain ( K )</td>
<td>2.1299</td>
<td>0.9631</td>
</tr>
<tr>
<td>PID ( f_z1 )</td>
<td>2.7 kHz</td>
<td>6.1 kHz</td>
</tr>
<tr>
<td>PID ( f_z2 )</td>
<td>10.5 kHz</td>
<td>3.5 kHz</td>
</tr>
<tr>
<td>Target ( f_c )</td>
<td>10.5 kHz</td>
<td>16.0 kHz</td>
</tr>
<tr>
<td>Predicted ( m_0 )</td>
<td>59.8°</td>
<td>57.2°</td>
</tr>
<tr>
<td>Average model ( f_c )</td>
<td>9.8 kHz</td>
<td>16.4 kHz</td>
</tr>
<tr>
<td>Average model ( m_0 )</td>
<td>61.5°</td>
<td>54.5°</td>
</tr>
<tr>
<td>Average model ( GM )</td>
<td>10.0 dB</td>
<td>16.2 dB</td>
</tr>
</tbody>
</table>

\( C = 3 \) mF, TARGET \( m_0 = 50° \)

\( C = 600 \) μF, TARGET \( m_0 = 50° \)

**TABLE IIIA**

<table>
<thead>
<tr>
<th>SYSTEM-ID METHOD: STATISTICAL ANALYSIS OF RESULTS</th>
<th>( C = 3 ) mF, TARGET ( m_0 = 50° )</th>
<th>( C = 600 ) μF, TARGET ( m_0 = 50° )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Identified ( f_0 ) [kHz]</td>
<td>2.73</td>
<td>2.73</td>
</tr>
<tr>
<td>Target ( f_c ) [kHz]</td>
<td>10.49</td>
<td>10.16</td>
</tr>
<tr>
<td>Predicted ( m_0 ) [°]</td>
<td>58.4</td>
<td>54.1</td>
</tr>
<tr>
<td>PID Gain ( K )</td>
<td>2.2641</td>
<td>2.9340</td>
</tr>
<tr>
<td>PID ( f_z1 ) [kHz]</td>
<td>2.73</td>
<td>2.73</td>
</tr>
<tr>
<td>PID ( f_z2 ) [kHz]</td>
<td>10.49</td>
<td>10.16</td>
</tr>
</tbody>
</table>

\( C = 3 \) mF, TARGET \( m_0 = 50° \)

\( C = 600 \) μF, TARGET \( m_0 = 50° \)

V. EFFECTIVENESS OF AUTOTUNING TECHNIQUES

Sections III and IV described operation and successful experimental verification of the two considered autotuning techniques for two capacitive loads. To further evaluate effectiveness of the autotuners, we performed simulation tests over the C-ESR design space in Fig. 2. In addition to the target phase margin \( m_0 = 50° \), the success criteria included a target gain margin (10 dB) and the absence of limit cycling. Figure 11 shows that both algorithms are able to successfully tune the PID parameters to meet the stability margins over the entire C-ESR design space. This includes two orders of magnitude in the capacitance variation (\( 1 \) mF < \( C < 100 \) mF) and more than three orders of magnitude in the ESR range. In terms of the time constant \( \tau = ESR \cdot C \) of the capacitive load, the proposed tuning algorithms are capable of handling about five orders of magnitude, from nanoseconds to hundreds of microseconds.

Performance for capacitances greater than \( 100 \) mF was not investigated. Both methods have difficulties near the \( f_0 = f_s / 20 \) boundary. Such unusually high-\( \tau \) cases are not properly handled by the relay-based method mainly because of the poor phase boost achievable from the first PID zero \( z_1 \), which is always tuned at \( f_z1 = f_0 \). These points thus fail to meet the phase margin specification. A different tuning strategy for \( z_1 \), not discussed here, could be investigated to extend the design space coverage further for the relay-feedback approach. The System-ID method, on the other hand, places \( f_z1 \) low enough so

![Fig. 10 – System-ID method: 9 A load steps](image)

![Fig. 11 – Effectiveness of autotuning techniques: design space coverage](image)
that the phase margin specification is met, but instead results in a loop gain that exhibits multiple 0 dB crossings due to \( f_c \) being placed too close to the high-Q-factor resonant peak at \( f_0 \).

The main advantages of the relay-based method include targeted frequency measurements, as well as robustness and computational simplicity of the iterative tuning process for a PID compensator. Experimental results demonstrate very high-performance closed-loop operation.

The System-ID approach instead starts by finding the overall frequency response and then proceeds to tune the compensator parameters by computations only. Quantization noise effects in identifying the frequency response at relatively high frequencies may limit the achievable bandwidth.

In hardware implementations, we found that higher-level tools (such as Xilinx System Generator, which was used to implement the relay-based method) can result in significant overheads compared to direct HDL coding. As an example, in spite of more involved computations, modest hardware requirements have been achieved for the System-ID based autotuner, which was implemented directly in Verilog HDL.

VI. CONCLUSIONS

This paper addresses auto-tuning of digital controllers for point-of-load (POL) switching converters with wide range of capacitive loads. Two auto-tuning methods are considered: a modified relay-feedback autotuning technique, and a system-identification (System-ID) based autotuner. Simulation and experimental results on a 12-to-1.5 V, 9 A, 200 kHz POL converter show that both methods are capable of achieving target stability margins and high performance closed-loop responses over a design space that includes orders of magnitude variations in the load capacitance and ESR. Main limitations of the proposed techniques were discussed along with hardware requirements. Both autotuning methods can be considered good candidates for practical POL and other DC-DC applications.

REFERENCES


