Circuit noise evaluation by Padé approximation based model-reduction techniques

Peter Feldmann

Bell Laboratories
Murray Hill, NJ 07974

Roland W. Freund

Bell Laboratories
Murray Hill, NJ 07974

Abstract

This paper introduces a new circuit noise analysis and modeling method. The noise analysis method computes an analytic expression of frequency, in rational form, which represents the Padé approximation of the noise power spectral density. The approximation can be carried out efficiently, to the required accuracy, using a variant of the PVL [1] or MPVL [2] algorithms. The new method is significantly more efficient than traditional methods for noise computation at numerous frequency points. In addition, it allows for a compact and cascadable modeling of noise that can be used in system level simulations.

1 Introduction

Noise is a fundamental phenomenon in electronic circuits caused by the small fluctuations in currents and voltages that occur within the devices in the circuit. The fluctuations are due mainly to the discontinuous nature of electric charge. Determining the effects of noise is very important, as noise often represents the fundamental limit of circuit or system performance.

Noise analysis algorithms for circuits in DC steady-state have been available for a long time in programs such as SPICE [6]. The results of such programs is the noise power over a range of frequencies in tabulated form. Circuit or system designers, typically reduce the information contained in the noise spectrum to one single number such as the *noise figure* [3]. While such compact representations offer good insight, and are very convenient for back-of-the-envelope calculations, CAD tools at both the circuit and the system levels can take advantage of the more accurate and complete information available in the noise spectrum, and, in return, offer more accurate analysis.

In this paper we introduce an algorithm which computes the noise power spectral density as a closed form rational expression. More specifically, the algorithm computes the Padé approximation of the noise power spectral density using the numerically robust and efficient Lanczos [4] method. The spectrum is computed to the required accuracy over the frequency range of interest. This method is significantly more efficient than repeatedly evaluating the noise power over a fine frequency grid. However, the real advantage of the new approach consists in the compact noise models that get produced. These compact noise models with the spectral density specified as rational expressions can be accepted by the algorithm as input noise sources. In other words, the algorithm can consume its own output, thus lending itself to a hierarchical analysis methodology.

For example, the circuit designer, designs an amplifier, analyzes it, and produces a high-level model, say, of its transfer function and of its output noise spectrum. The system designer uses models for all the components to performs, using the same noise analysis algorithm, a system-level simulation. No information is lost due to a too narrow interface between circuit-level and system-level models.

The paper is organized as follows: In the next section we review the circuit noise analysis problem. In Section 3 we reformulate the noise spectral density expression in a form compatible with Padé via Lanczos, (PVL), model reduction. Section 4 discusses the application of the PVL algorithm to this particular problem. Finally, in Section 5, we illustrate the noise analysis problem with a few circuit examples and then present a few concluding remarks.

2 Review of circuit noise analysis

The principal mechanisms of noise in integrated circuits are [3]: thermal, shot, and flicker noise. Mathematically, device noise is modeled by stochastic processes [5]. Stochastic processes, represent ansambles of functions of time, $\mathbf{n}(t)$, and are characterized in terms of statistical averages, such as the mean and autocorrelation, in the time domain, or the power spectral density in the frequency domain.

Thermal noise, is caused by the thermal agitation of the electrons and occurs in almost all devices. Thermal noise is modeled by a parallel current source, the value of which is a zero mean stochastic process with a frequency independent (*white noise*) spectral density equal to

$$S_{\mathrm{th}}(\omega) = 4kTG, \tag{1}$$

where k is Boltzman's constant, T the absolute temperature, and G the conductance.

Shot noise, is due to the fact that the current through a PN junction consists of discrete charge carriers randomly crossing a potential barrier. Shot noise in a junction is also modeled by a parallel, white noise current source. The spectral density of shot noise is

$$S_{\mathrm{sh}}(\omega) = 2qI_d, \tag{2}$$

where q is the electron charge and I_d the average junction current.

Flicker (or $\frac{1}{f}$) noise, occurs in all active devices and even in some resistors due to mechanisms that are not well understood. Flicker noise is modeled by a stochastic process with a frequency dependent spectral density

$$S_{\mathrm{fl}}(\omega) = K_1 \frac{I^a}{(2\pi\omega)^b}, \tag{3}$$

where I is the average direct current, K_1 is a technology dependent constant characterizing a particular device and process, a is a constant in the range $0.5 - 2.0$, and b is a constant of about one, (hence the name $\frac{1}{f}$ noise).

In this paper we only consider circuits with a constant excitation in steady-state (DC). The noise analysis methods presented below can also be applied to circuits with time varying bias conditions [7], but such an extension is beyond the scope of the present paper. Moreover, noise is assumed to represent a "small" perturbation to the circuit.

The circuit equations under these assumptions are

$$f(x(t), \frac{d}{dt}x(t), b_0, n(t)) = 0. \tag{4}$$

Here, $x(t)$ is the vector of circuit variables, typically currents and voltages, b_0 is the constant (DC) excitation, and $n(t)$ is a vector of "small" perturbations caused by the noise sources.

Moreover, we assume that the circuit is stable and therefore the solution of the noiseless circuit x_0, is constant in time and satisfies

$$f(x_0, 0, b_0, 0) = 0, \tag{5}$$

since, obviously, $\frac{d}{dt}x_0 = 0$.

The response of the circuit in the presence of the perturbation perturbation, $z(t)$, of the DC solution

$$f(x_0 + z(t), \frac{d}{dt}z(t), b_0, n(t)) = 0. \tag{6}$$

Assuming that the noise perturbation is "small", the first order Taylor expansion of the circuit equations (6) around the DC solution is sufficiently accurate

$$f(x_0, 0, b_0, 0) + Gz(t) + C\frac{d}{dt}z(t) - Bn(t) = 0 \tag{7}$$

where $G = \left.\frac{\partial f}{\partial x}\right|_{x_0,0,b_0,0}$, $C = \left.\frac{\partial f}{\partial \dot{x}}\right|_{x_0,0,b_0,0}$, and $B = -\left.\frac{\partial f}{\partial n}\right|_{x_0,0,b_0,0}$. Subtracting (5), we are left with just the linear, differential equations for the perturbation signals

$$Gz(t) + C\frac{d}{dt}z(t) = Bn(t). \tag{8}$$

The vector stochastic process $\mathbf{n}(t)$ is specified in terms of its frequency domain cross-spectral density matrix where $S_{xx}(\omega)$. The diagonal elements in $S_{xx}(\omega)$ represent the power spectral density of each noise source, and the off-diagonal elements describe statistical coupling of noise signals. In practical cases, S_{xx} will almost always be a diagonal matrix.

The noise analysis problem reduces to that of the propagation of a stochastic process through a linear system. The general expression of the noise power spectral density at the output of the linear system, S_{yy} is given by the well known formula [5]

$$S_{yy}(j\omega) = H(j\omega)S_{xx}(j\omega)H^{\mathrm{H}}(j\omega) \tag{9}$$

When only one output is analyzed, $S_{yy}(j\omega)$ is just a scalar function of frequency. For more than one output, $S_{yy}(j\omega)$ is a full matrix, the dimension of which is the number of outputs. The diagonal elements of $S_{yy}(j\omega)$ represent the power spectral density of the noise at each output and the off-diagonal elements represent the cross-spectral density.

The many-to-one vector transfer function of the linear system from the noise sources to an output port of interest is

$$H(j\omega) = l^{\mathrm{T}}(G + j\omega C)^{-1}B \tag{10}$$

where l denotes the incidence vector that corresponds to the output port of interest. More generally, when more than one output is considered, we have a many-to-many matrix-transfer-function from noise sources to the outputs

$$H(j\omega) = L^{\mathrm{T}}(G + j\omega C)^{-1}B, \tag{11}$$

where L is the incidence matrix of the output ports.

From the formula (9), using (10), we obtain the following expression for the noise power spectral density at the output of the system

$$S_{yy}(j\omega) = l^{\mathrm{T}}(G + j\omega C)^{-1}BS_{xx}(j\omega)B^{\mathrm{T}}(G + j\omega C)^{-\mathrm{H}}l. \tag{12}$$

The noise analysis method implemented in SPICE [6], evaluates this expression efficiently, for a given ω, using the solution of the adjoint system

$$x_a(j\omega) = (G + j\omega C)^{-\mathrm{H}}l.$$

thus reducing (12) to

$$S_{yy}(j\omega) = x_a^{\mathrm{H}}(j\omega)BS_{xx}(j\omega)B^{\mathrm{T}}x_a(j\omega), \tag{13}$$

The SPICE noise computation algorithm computes the noise power only at a given frequency. When we are interested in the spectrum of the noise we must repeat the procedure for a large number of frequencies.

We now introduce a novel noise analysis method which computes a closed form rational expression for the noise power spectral density over a wide frequency range. This method is more efficient than the classical method, when noise needs to be computed over a frequency grid. Moreover, the closed form expression represents a compact model of the noise spectrum, and can be used hierarchically in system simulations.

3 Reformulated noise spectral density

For the following, it will be convenient to introduce the new variable $s := j\omega$. Note that, in order to be "physically" meaningful, the variable s has to be purely imaginary. For now, we thus assume that s is purely imaginary. Later on, we will drop this constraint and treat s as a general complex variable.

The new noise analysis and modeling algorithm relies on the computation of the Padé approximation to the noise power spectral density expression (12). The Padé approximation of a general transfer function expression of the form

$$F(s) = \tilde{l}^{\mathrm{T}}(\tilde{G} + s\tilde{C})^{-1}\tilde{r}, \tag{14}$$

where $\tilde{r}$ and $\tilde{l}$ are vectors of length $\tilde{N}$, and $\tilde{G}$ and $\tilde{C}$ are $\tilde{N} \times \tilde{N}$ matrices, can be computed efficiently with the PVL (Padé via Lanczos) algorithm [1]. At first glance, it appears that noise-type transfer functions (12) are very different from (14). However, we will show that there are vectors $\tilde{r}$, $\tilde{l}$ and matrices $\tilde{G}$, $\tilde{C}$ so that the functions (12) and (14) agree for all purely imaginary s, i.e., for all physically meaningful values of s.

First, consider the case when the noise sources are all white. Then S_{xx} is not a function of the frequency, and thus the function (12) reduces to

$$F(s) = l^{\mathrm{T}}(G + sC)^{-1}BS_{xx}B^{\mathrm{T}}(G + sC)^{-\mathrm{H}}l. \tag{15}$$

Here, we have used the new variable $s = j\omega$. We rewrite (15) by introducing two new vectors, u, and v

$$\begin{aligned} F(s) &= l^{\mathrm{T}}u, \\ v &= (G + sC)^{-\mathrm{H}}l, \\ u &= (G + sC)^{-1}BS_{xx}B^{\mathrm{T}}(G + sC)^{-\mathrm{H}}l \\ &= (G + sC)^{-1}BS_{xx}B^{\mathrm{T}}v \end{aligned} \tag{16}$$

The vectors u and v represent, therefore, the solution of a system of linear equations

$$\begin{bmatrix} 0 & (G + sC)^{\mathrm{H}} \\ G + sC & -BS_{xx}B^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} l \\ 0 \end{bmatrix} \tag{17}$$

From (16) and (17) we obtain

$$\begin{aligned} F(s) = \begin{bmatrix} l^{\mathrm{T}} & 0 \end{bmatrix} & \left(\begin{bmatrix} 0 & G^{\mathrm{T}} \\ G & -BS_{xx}B^{\mathrm{T}} \end{bmatrix} + \right. \\ & \left. + s \begin{bmatrix} 0 & -C^{\mathrm{T}} \\ C & 0 \end{bmatrix} \right)^{-1} \begin{bmatrix} l \\ 0 \end{bmatrix} \end{aligned} \tag{18}$$

exactly of the form (14), amenable to PVL reduction.

Unfortunately, as shown in the previous section, not all noise sources are white. In order to be able to treat more general noise sources, we actually consider a more general class of noise-type transfer functions. More precisely, we study functions of the form

$$F(s) = l^{\mathrm{T}}(G + sC)^{-1}BP^{-1}(s)B^{\mathrm{T}}(G + sC)^{-\mathrm{H}}l. \tag{19}$$

Here, l is a real vector of length N, G and C are real $N \times N$ matrices, B is a real $N \times M$ matrix, and $P(s)$ is a *matrix polynomial*

$$P(s) = P_0 + P_1 s + P_2 s^2 + \cdots + P_L s^L, \tag{20}$$

whose coefficients P_i, $i = 0, 1, \ldots, L$, are $M \times M$ matrices. We assume that P_L is not a zero matrix, so that L is the *degree* of the matrix polynomial $P(s)$. The form (19) can express practically all interesting noise power spectral densities. The degree L itself can be arbitrary; however, the cases of low degree such as $L = 0, 1, 2$ are the most important ones. For example, for $L = 0$ and $P_0 = S_{xx}^{-1}$, the function (19) reduces to the case (15) of white noise. The flicker noise frequency dependent power spectral density (3) can also be well approximated by an expression of form (19) by expanding the denominator into a power series

$$S_{\mathrm{fl}}(\omega) = K_1 I^a \left(c_0 + c_1 s + c_2 s^2 + \ldots\right)^{-1} \qquad (21)$$

Rewriting the noise-type transfer function $F(s)$ given by (19) in the form (14) then allows us to compute Padé-based reduced-order models for $F(s)$ by simply applying the PVL algorithm to the representation (14) of $F(s)$.

Next, we show how to transform (19) to form (14). Consider the linear system

$$\begin{bmatrix} 0 & (G+sC)^{\mathrm{H}} & 0 & 0 \cdots & 0 \\ G+sC & 0 & B & 0 \cdots & 0 \\ 0 & B^{\mathrm{T}} & P_0+sP_1 & sP_2 \cdots & sP_L \\ 0 & 0 & sI & -I \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots \; \ddots & \vdots \\ 0 & \cdots & \cdots & 0 \;\; sI & -I \end{bmatrix} \begin{bmatrix} x \\ y \\ z_1 \\ z_2 \\ \vdots \\ z_L \end{bmatrix} = \begin{bmatrix} l \\ 0 \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix} \qquad (22)$$

From the last $L-1$ blocks of equations in (22), it follows that

$$z_i = s z_{i-1} \quad \text{for all} \quad i = 2, 3, \ldots, L, \qquad (23)$$

and thus

$$z_i = s^{i-1} z_1 \quad \text{for all} \quad i = 2, 3, \ldots, L. \qquad (24)$$

Using the third block of equations in (22), together with (24) and (20), we get

$$\begin{aligned} B^{\mathrm{T}} y &= -(P_0 + sP_1) z_1 + sP_2 z_2 + \cdots + sP_L z_L \\ &= -\left(P_0 + sP_1 + s^2 P_2 + \cdots + s^L P_L\right) z_1 \\ &= -P(s) z_1. \end{aligned} \qquad (25)$$

By the first two blocks of equations in (22), we have

$$\begin{aligned} y &= (G+sC)^{-\mathrm{H}} l, \\ x &= -(G+sC)^{-1} B z_1. \end{aligned} \qquad (26)$$

Combining (25) and (26), we get

$$x = (G+sC)^{-1} B (P(s))^{-1} B^{\mathrm{T}} (G+sC)^{-\mathrm{H}} l. \qquad (27)$$

Next, we observe that, for purely imaginary s, the linear system (22) can be rewritten in the form

$$\left(\tilde{G} + s\tilde{C}\right) \tilde{x} = \tilde{l}. \qquad (28)$$

Here, $\tilde{x}$ and $\tilde{l}$ are vectors of length $\tilde{N}$ d defined by

$$\tilde{x} = \begin{bmatrix} x \\ y \\ z_1 \\ \vdots \\ z_L \end{bmatrix} \quad \text{and} \quad \tilde{l} = \begin{bmatrix} l \\ 0 \\ 0 \\ \vdots \\ 0 \end{bmatrix}, \qquad (29)$$

and $\tilde{G}$ and $\tilde{C}$ are $\tilde{N} \times \tilde{N}$ matrices given by

$$\tilde{G} = \begin{bmatrix} 0 & G^{\mathrm{T}} & 0 & 0 & \cdots & 0 \\ G & 0 & B & 0 & \ddots & 0 \\ 0 & B^{\mathrm{T}} & P_0 & 0 & \ddots & \vdots \\ 0 & 0 & 0 & -I & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & 0 & -I \end{bmatrix} \qquad (30)$$

and

$$\tilde{C} = \begin{bmatrix} 0 & -C^{\mathrm{T}} & 0 & 0 & \cdots & 0 \\ C & 0 & 0 & 0 & \ddots & 0 \\ 0 & 0 & P_1 & P_2 & \ddots & P_L \\ 0 & 0 & I & 0 & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & \cdots & 0 & I & 0 \end{bmatrix} \qquad (31)$$

Using (19), and (28) to (31), it follows that

$$\begin{aligned} F(s) &= l^{\mathrm{T}} (G+sC)^{-1} B (P(s))^{-1} B^{\mathrm{T}} (G+sC)^{-\mathrm{H}} l \\ &= l^{\mathrm{T}} x = \tilde{l}^{\mathrm{T}} \tilde{x} \\ &= \tilde{l}^{\mathrm{T}} \left(\tilde{G} + s\tilde{C}\right)^{-1} \tilde{l}. \end{aligned} \qquad (32)$$

This shows that, for purely imaginary s, the noise-type transfer function (19) is indeed of the form (14) with $\tilde{l} = \tilde{r}$, $\tilde{G}$, and $\tilde{C}$ defined in (31).

Of particular interest are several special cases

- The case $L = 1$

  Here $\tilde{l}$, $\tilde{G}$, and $\tilde{C}$ reduce to

$$\begin{aligned} &\tilde{l} = \begin{bmatrix} l & 0 & 0 \end{bmatrix}^{\mathrm{T}}, \\ &\tilde{G} = \begin{bmatrix} 0 & G^{\mathrm{T}} & 0 \\ G & 0 & B \\ 0 & B^{\mathrm{T}} & P_0 \end{bmatrix}, \tilde{C} = \begin{bmatrix} 0 & -C^{\mathrm{T}} & 0 \\ C & 0 & 0 \\ 0 & 0 & P_1 \end{bmatrix}. \end{aligned} \qquad (33)$$

- The case $L = 0$

This is the case (15) that all noise sources are white. It is covered by (33) with $P_0 = S_{xx}^{-1}$ and $P_1 = 0$. However, in this case, we eliminate the third block rows and columns in (33) and obtain

$$\tilde{l} = \begin{bmatrix} l \\ 0 \end{bmatrix}, \tilde{G} = \begin{bmatrix} 0 & G^{\mathrm{T}} \\ G & -B^{\mathrm{T}} S_{xx}^{-1} B \end{bmatrix}, \tilde{C} = \begin{bmatrix} 0 & -C^{\mathrm{T}} \\ C & 0 \end{bmatrix} \tag{34}$$

This is exactly the form arrived at in (18).

4 Application of PVL

Now that we have shown how to reformulate noise-type transfer function $F(s)$ given by (19) in the "PVL" form (14), it is straightforward to employ PVL to generate reduced-order models. Recall that, in our case, $\tilde{l} = \tilde{r}$ in (14).

First, we choose a real expansion point s_0, and compute, $\mathcal{L}$ and $\mathcal{U}$ as the factorization

$$\tilde{G} + s_0 \tilde{C} = \mathcal{L} \cdot \mathcal{U}. \tag{35}$$

Then, setting $s = s_0 + \sigma$, we rewrite (14) as follows:

$$\begin{aligned} F(s_0 + \sigma) &= \tilde{l}^{\mathrm{T}} \left(\tilde{G} + s_0 \tilde{C} + \sigma \tilde{C} \right)^{-1} \tilde{l} \\ &= \left(\mathcal{U}^{-\mathrm{T}} \tilde{l} \right)^{\mathrm{T}} \left(I + \sigma \mathcal{L}^{-1} \tilde{C} \mathcal{U}^{-1} \right)^{-1} \left(\mathcal{L}^{-1} \tilde{l} \right). \end{aligned} \tag{36}$$

We then apply the Lanczos process to the matrix $A = \mathcal{L}^{-1} \tilde{C} \mathcal{U}^{-1}$, using $b = \mathcal{L}^{-1} \tilde{l}$ and $c = \mathcal{U}^{-\mathrm{T}} \tilde{l}$ as the right, respectively left, starting vector. After running the Lanczos process for n iterations, we obtain an $n \times n$ tridiagonal matrix, T_n, such that the function

$$F_n(s_0 + \sigma) = \left(c^{\mathrm{T}} b \right) \cdot e_1^{\mathrm{T}} \left(I + \sigma T_n \right)^{-1} e_1 \tag{37}$$

where e_1 represents the first unit vector of length n, is just an n-th *Padé approximant* to $F(s_0 + \sigma)$. More precisely, $F_n(s_0 + \sigma)$ is a rational function of σ with numerator polynomial of degree at most $n - 1$ and denominator polynomial of degree at most n such that

$$F_n(s_0 + \sigma) = F(s_0 + \sigma) + \mathcal{O}(n^{q(n)}), \tag{38}$$

where $q(n)$ is maximal. In the generic case, $q(n) = 2n$. Note that (38) just states that the Taylor expansions of F_n and F about the expansion point s_0 agree in as many leading Taylor coefficients as possible. We note that all quantities involved in the Lanczos process are real, as long as the coefficient matrices $P_0, P_1, \ldots, P_L$ of (20) are real, which is usually the case.

We observe that the reduced-order model for the noise spectral density of a circuit module will always have the form in (37), which results from the PVL algorithm. If the reduced order models of circuit modules are used in higher level simulations, expressions of the form in (37) appear in the $S_{xx}(s)$ noise source spectral density matrix of the system simulation. The resulting output noise spectral density of the system will have the form

$$F(s) = l^{\mathrm{T}} (G + sC)^{-1} B (P_0 + sP_1)^{-1} B^{\mathrm{T}} (G + sC)^{-\mathrm{H}} l, \tag{39}$$

therefore, compatible with PVL, as shown for the special case $L = 1$ in (33).

Finally, we must make some comments, regarding properties of the PVL algorithm specific to its application to "noise"-type problems. So far, we have made no assumptions on the matrix polynomial

$$P(s) = P_0 + P_1 s + P_2 s^2 + \cdots + P_L s^L. \tag{40}$$

If the function $F(s)$ describes the noise power spectral density of a circuit, then $P(s)$ needs to be such that

$$F(j\omega) > 0 \quad \text{for all} \quad \omega > 0. \tag{41}$$

Ideally, the PVL algorithm is stopped as soon as the Padé approximant F_n has converged to F in the frequency case of interest, i.e., if

$$|F(j\omega) - F_n(j\omega)| \le \mathtt{tol} \quad \text{for all} \quad \omega \in [\omega_{\min}, \omega_{\max}]. \tag{42}$$

Together with (41), this ensures that the Padé-based reduced-order model

$$F_n(s_0 + \sigma) = \left(c^{\mathrm{T}} b \right) \cdot e_1^{\mathrm{T}} \left(I + \sigma T_n \right)^{-1} e_1 \tag{43}$$

satisfies

$$F_n(j\omega) \gtrsim 0 \quad \text{for all} \quad \omega \in [\omega_{\min}, \omega_{\max}]. \tag{44}$$

This observation is important if we want to use the reduced-order model as noise sources in a high-level simulation.

As a final note, we remark that the discussion in this and previous sections can be generalized for the computation of the cross-spectral density matrix of multiple outputs. In the multiple output case the l and $\tilde{l}$ vectors become matrices and MPVL will be used instead of PVL.

5 Examples

We applied the noise computation algorithm on a number of circuits. The first example is the 741 operational amplifier. The size of the problem is 55 variables. Figure 1 shows the exact transfer function of the

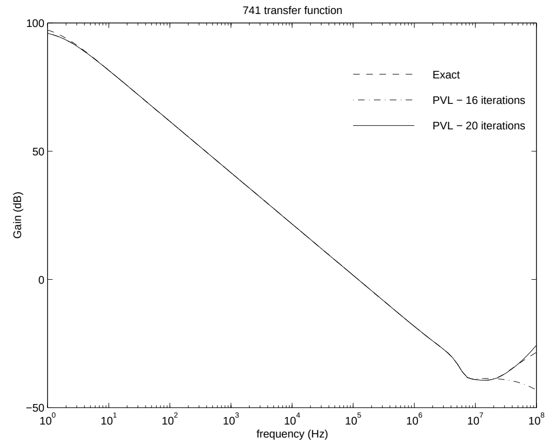

Figure 1: 741 gain

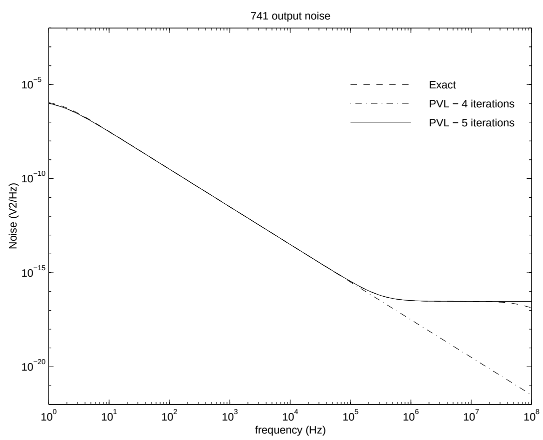

Figure 2: 741 noise

amplifier compared to the PVL reduced order models of orders 16 and 20. The order 20 approximation captures the behavior of the amplifier almost exactly. Figure 2 shows the amplifier output noise power spectral density over the same frequency range. Here, a Padé approximations of order 5 is already sufficient to capture the noise spectrum.

The next example is a 5-th order Cauer filter which uses ten 741 opamps as building blocks. The total size of the problem is 463 variables. Figures 3 and 4 show the transfer function and the output noise spectrum computed exactly, and with PVL. We observe that we need roughly the same number of iterations to obtain an almost perfect match of both the transfer function and the noise spectrum.

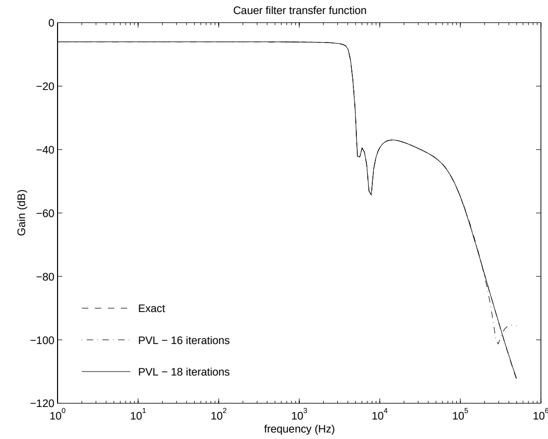

Figure 3: Cauer filter transfer characteristic

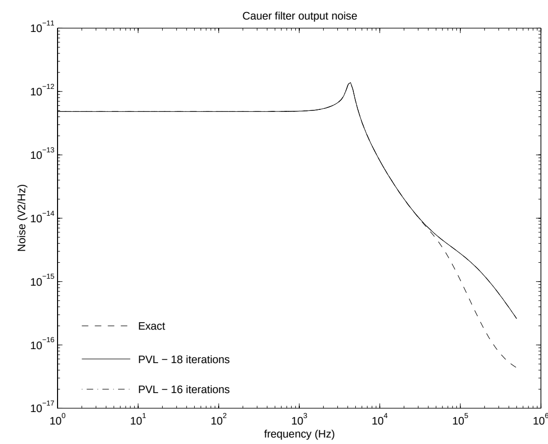

Figure 4: Cauer filter noise

The final example is a bandpass filter derived from a 3-rd order Chebyshev low-pass prototype, and implemented with single amplifier biquads. It also uses the 741 opamp as a building block. The problem size is 147. Figures 5 and 6 show the transfer function and the output noise spectrum computed exactly, and with PVL. We observe that we need 18 iterations to match the transfer function and only 14 to match the noise spectrum.

6 Conclusions

In this paper we have introduced a new noise analysis method which computes the noise power spectral density of a circuit node or the cross-spectral density of a number of nodes. The results are presented in the form of a closed form polynomial rational function of frequency which represents a Padé approximation of the true noise spectral density. This method is significantly more efficient than the classical noise analysis method for predicting noise over a range of frequencies. The main advantage of the method, however, is the fact that it produces a reduced-order model of the noise generated by the circuit under analysis. This model can then be employed, using the same algorithm in a system-level analysis. The noise analysis algorithm accepts the noise source power spectral density in a rational polynomial form. This form covers practically all possible noise sources of interest.

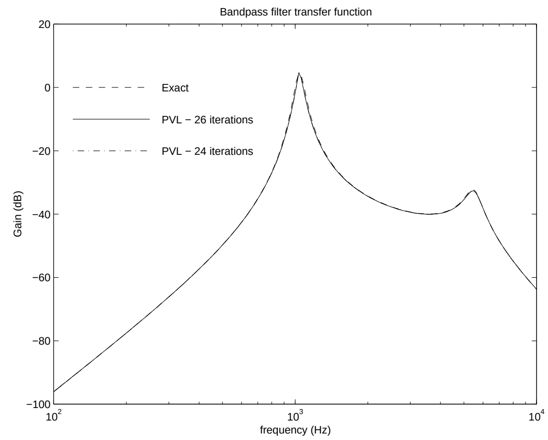

Figure 5: Bandpass filter transfer characteristic

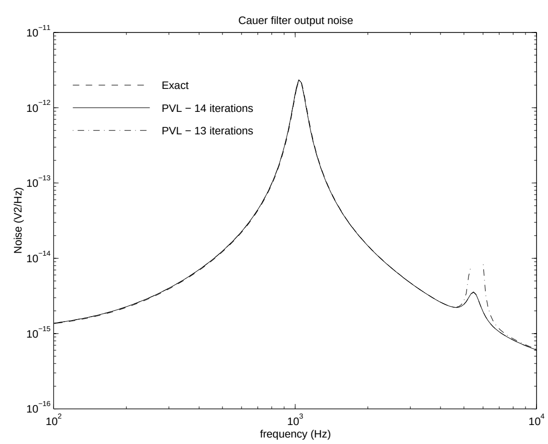

Figure 6: Bandpass filter noise

Acknowledgments

We wold like to thank our colleagues: Peter Kinget, David Lee, Jaijeet Roychwdhury, and Lászlo Tóth, for useful discussions and help with examples.

References

[1] P. Feldmann and R. W. Freund, *Efficient linear circuit analysis by Padé approximation via the Lanczos process*, IEEE Trans. Computer-Aided Design **14** (1995), 639–649.

[2] P. Feldmann and R. W. Freund, *Reduced-order modeling of large linear sub-circuits via a block Lanczos algorithm*, Proc. 32nd Design Automation Conference, ACM, New York, 1995, pp. 474–479.

[3] P.R. Gray and R.G. Meyer, *Analysis and Design of Analog Integrated Circuits* 3rd. edition, New York, N.Y.: John Wiley, 1993.

[4] C. Lanczos, *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators*, J. Res. Nat. Bur. Standards **45** (1950), 255–282.

[5] A. Papoulis, *Probability, Random Variables, and Stochastic Processes*, 3rd. edition, New York, N.Y.: McGraw-Hill, 1991.

[6] Rohrer, R.A., Nagel L., Meyer, R. and Weber, L., *Computationally efficient electronic-circuit noise calculations*, IEEE J. Solid State Circuits, vol. SC-6, pp. 204-213, 1971.

[7] J. Roychowdhury and P. Feldmann. "A New Linear-Time Harmonic Balance Algorithm for Cyclostationary Noise Analysis in RF Circuits," Proc. Asia and South-Pacific Design Automation Conference, Jan 1997.