

Modelling high-order filters in a transient microwave circuit simulator

F.P. Hart, S.R. Luniya, J. Nath, A. Victor, A. Walker and M.B. Steer

Abstract: Transient simulation of narrowband bandpass filters used in microwave circuits is challenging because of matrix ill-conditioning. Here, such filters are modelled as the equivalent discrete-time form developed using a bilinear z -transform. The technique has been implemented in a general purpose transient circuit simulator and validated using a 1.7 GHz 5-section coaxial filter with a 0.9% bandwidth.

1 Introduction

Transient circuit simulation is becoming increasingly important in modelling RF and microwave circuits where pulsed operation, large transistor count circuits and composite signals including multi-channel and chirped signals, tight specifications and high-level noise and interference are increasingly prevalent. The ability to model RF and microwave systems in conventional transient circuit simulation is limited by the difficulty of modelling high-order, narrow-band filters. In transient circuit simulators filters are traditionally modelled using Laplace inverse pole-zero descriptions or using compact models of circuit elements. Accurate modelling typically requires small time-steps and the simulation problem is analogous to capturing small differences of large numbers [1].

The range of filters that can be modelled in a typical commercial transient simulator is shown in Fig. 1 for a Butterworth bandpass filter modelled using all-pole elements. This figure indicates regions where valid results are obtained with results valid for lower orders and higher bandwidths. When driven by a linear FM chirp source [2], a generic radar signal, for example, a convergent solution can only be obtained for a filter with a fractional bandwidth $>3\%$ and when the order of the low-pass prototype of the filter order is <10 or so. The valid operating region is somewhat wider for sinusoidal excitation. The root cause of the problem is ill conditioning of the modified nodal admittance matrix (MNAM) developed from the associated discrete model during iterative simulation [3]. The problem is exacerbated by the large range of voltage and current levels in high Q filters.

Even though higher-frequency operation of modern circuits increases the possibility for ill-conditioned MNAMs, the potential for a given circuit to have an ill-conditioned MNAM was known to the creators of Spice [4]. For

example, consider the simple two-stage ladder circuit shown in Fig. 2. Here, a 50 MHz sinusoidal voltage source drives the ladder, which is terminated in a 50-ohm load. In this circuit, the first stage has element values $L_1 = 1 \mu\text{H}$ and $C_1 = 1 \text{ pF}$, and the second stage element values are multiplied by a scale factor, η , which ranges over several orders of magnitudes. The frequency-domain MNAM for this illustrative circuit is easily obtained using the techniques described in [5]. Fig. 3 shows (the logarithm of) the condition number of the MNAM for this circuit as a function of (the logarithm of) the scaling factor η . For large scaling factors – indicating especially stiff circuits – the condition number grows without bound leading to invalid simulation results. Note that the character of the curve here does not change significantly with a change in driving point frequency; the knee of the curve simply moves to the left or right.

The matter of ill-conditioning is arguably worse in the time domain. This can be seen by considering how the construction of the time-domain MNAM entries unavoidably involves entries with largely varying orders of magnitude. To see this, consider a single second-order factor of a tenth-order low-pass prototype Butterworth filter

$$H_{\text{LP}}(s) = \frac{k_0 s + k_1}{s^2 + b_0 s + 1} \quad (1)$$

where k_0 and k_1 are the residues of each of the rational fractions resulting from a partial fraction expansion and the b_0 factors can be found in [6] for a variety of filter types. This filter is subsequently converted to a bandpass filter through the substitution $H_{\text{BP}}(S) = H_{\text{LP}}((S^2 + \omega_0^2)/BS)$, where B is the 3-dB bandwidth and ω_0 is the centre frequency. When this substitution is made, the result is

$$H_{\text{BP}}(S) = \frac{k_0 B S^3 + B^2 k_1 S^2 + k_0 B \omega_0^2 S}{S^4 + b_0 B S^3 + (2\omega_0^2 + B^2) S^2 + b_0 B \omega_0^2 S + \omega_0^4} \quad (2)$$

Creation of the time-domain MNAM entries for the filter is accomplished via analytic Laplace transform inversion of (2) and the addition of state variables to account for the extra time-domain derivatives. As a result, the unit coefficient of S^4 in (2) and the very large factor ω_0^4 occupy separate MNAM entries with no way of resolving large disparities in their magnitudes. Thus, the ill-conditioning within the time-domain MNAM is induced by the intrinsic

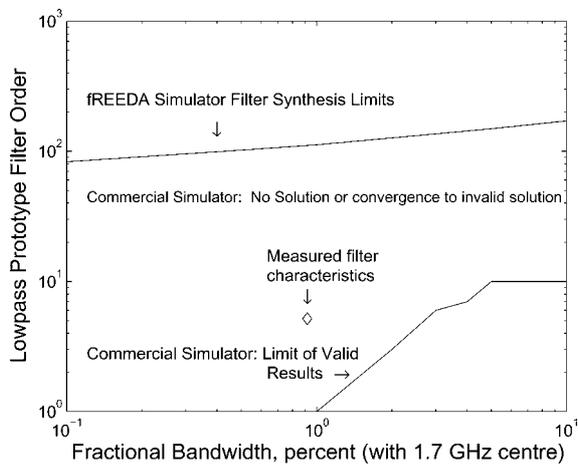


Fig. 1 Synthesis limitations of a leading commercial simulator compared with the model developed here and implemented in *fREEDA*TM

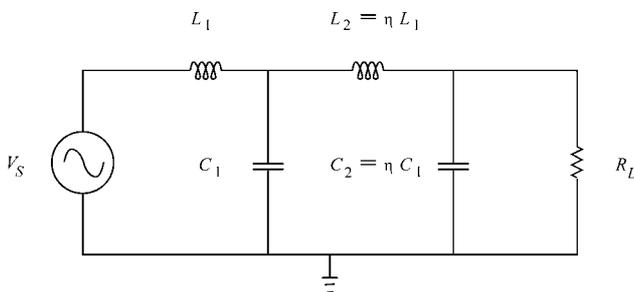


Fig. 2 Two-stage ladder circuit with ill-conditioned MNAM

centre frequency of operation and is unrelated to whether or not the overall system is stiff because of the interaction of other components. As an example of the atrocious ill-conditioning that can occur, a conventional implementation in *fREEDA*TM of a tenth-order Butterworth bandpass filter with a centre frequency of 1 MHz and a passband bandwidth of 100 kHz (i.e. 10%) yielded an MNAM with a condition number of 2.9×10^{43} !

In this paper, a strategy is presented for modelling filters of high order (up to order 83) and fractional bandwidths as low as 0.1%. The central concept is using a (pre-warped) *z*-domain approximation. The filter model makes only unit entries in the MNAM while filling in the right-hand side

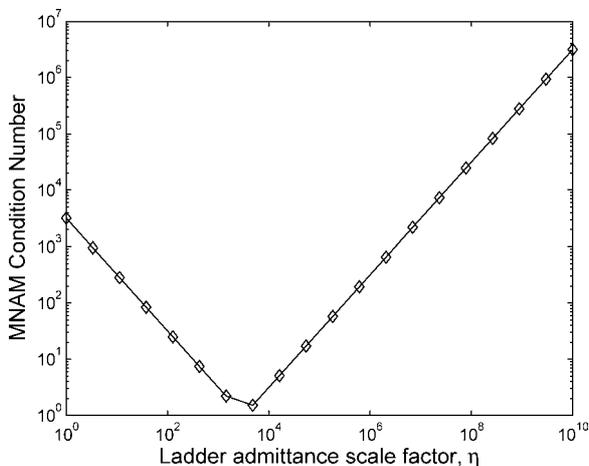


Fig. 3 Condition number of MNAM for the two-stage ladder circuit for various ladder scale factors at 50 MHz

known vector of the matrix-vector circuit equation, thus avoiding the creation (or exacerbation) of MNAM ill-conditioning. This technique was implemented in an open-source general purpose circuit simulator [7] modified to accommodate Spice-like transient circuit simulation for some elements and *z*-domain simulation for others. The procedure is described for a bandpass filter and results presented for transient simulation.

2 Z-Domain filter transformation

With the Bilinear *z*-Transform (BZT) [8] and uniform time discretisation (i.e. equal time-steps), a linear continuous time filter can be replaced by a discrete-time [9] equivalent filter provided that the Nyquist criterion [10] is observed. Although transient circuit simulation is intended to model continuous-time systems, simulation actually proceeds using discretised time models [11]. Thus, the behavioural modelling of analogue filters using equivalent discrete-time filters is a natural fit for transient simulation environments.

The process of modelling a bandpass filter begins with the specification of passband and stopband frequencies, passband flatness, stopband attenuation and often insertion loss. The bandpass specifications are then transformed to low-pass specifications, and well-defined [12] steps using BZT yield an equivalent discrete-time bandpass filter.

1. Determine the pre-warped frequency specifications from the analogue filter specifications.
2. Determine the low-pass transfer function in the *s*-domain using the pre-warped frequency specifications, then apply the lowpass-to-bandpass frequency transformation

$$s \leftarrow \frac{S^2 + \omega_o^2}{S} \quad (3)$$

which translates the filter behaviour into a passband in *S* geometrically centred at ω_o .

3. Perform the bilinear transformation

$$S \leftarrow \frac{z - 1}{z + 1} \quad (4)$$

to convert the filter from the *S*-domain to the discrete-time frequency domain (i.e. *z*-domain) and algebraically reduce the transfer function to a form that can be realised.

Steps 2 and 3 will be described in detail in Section 3. Step 1, frequency pre-warping, converts a continuous-time frequency ω_a to its discrete-time equivalent ω_p through the application of the formula

$$\omega_p = \tan\left[\frac{\omega_a \tau}{2}\right] \quad (5)$$

Here τ is the fixed step time in circuit simulation. However, from the discrete-time filtering perspective, τ also functions as the sampling interval and is related to the sampling frequency ω_s of the discrete-time filter by

$$\tau = \frac{2\pi}{\omega_s} \quad (6)$$

Substituting (6) into (5) leads to

$$\omega_p = \tan\left[\pi \frac{\omega_a}{\omega_s}\right] \quad (7)$$

Application of the Nyquist criterion requires that $\max(\omega_a) < \omega_s/2$, and after applying the criterion it can be seen that a one-to-one relationship exists between ω_a and ω_p .

3 Bandpass filter model

Construction of a particular instance of a discrete-time filter begins with the netlist parameters passed to the simulator engine. These parameters specify frequencies corresponding to the s -domain, that is, frequencies that are not pre-warped. The parameters may explicitly specify the centre frequency and number of poles of the filter; otherwise the passband and stopband edge frequencies implicitly specify the centre frequency and number of poles, which are then pre-warped and used to compute the number of poles, N , by following steps described in [12, 13]. From here forward, all frequencies will be assumed to be pre-warped.

The transfer function $H(s)$ of an N -pole low-pass filter is given by

$$H(s) = \begin{cases} A \prod_{k=0}^{(N/2)-1} \frac{1}{(s-p_k)(s-\bar{p}_k)}, & \text{for } N \text{ even} \\ \frac{A}{(s+p_r)} \prod_{k=0}^{((N-1)/2)-1} \frac{1}{(s-p_k)(s-\bar{p}_k)}, & \text{for } N \text{ odd} \end{cases} \quad (8)$$

In (8), p_k and \bar{p}_k are complex conjugate pole pairs, p_r is the magnitude of a pole that occurs on the negative real axis and A is a constant that sets the filter gain. For unity gain

$$A = \begin{cases} \prod_{k=0}^{(N/2)-1} |p_k|^2, & \text{for } N \text{ even} \\ p_r \prod_{k=0}^{((N-1)/2)-1} |p_k|^2, & \text{for } N \text{ odd} \end{cases} \quad (9)$$

The pole locations p_k are given in [14] for Butterworth and Chebychev filters.

The process is illustrated by considering the transfer function of the pole pair of a low-pass filter

$$H_k(s) = \frac{1}{a_k s^2 + b_k s + c_k} \quad (10)$$

Now, applying the frequency transformation (3) to (10) the bandpass $H_k(S)$ response is derived from $H_k(s)$

$$H_k(S) = \frac{S^2}{a_k S^4 + b_k S^3 + c_k S^2 + D_k S + E_k} \quad (11)$$

where $C_k = 2\omega_0^2 a_k + c_k$, $D_k = b_k \omega_0^2$ and $E_k = a_k \omega_0^4$. Next, the BZT $S \Leftarrow (z-1)/(z+1)$ is applied to (11) yielding

$$H_k(z) = \frac{1 - 2z^{-2} + z^{-4}}{F_k + G_k z^{-1} + H_k z^{-2} + K_k z^{-3} + M_k z^{-4}} \quad (12)$$

with F_k, \dots, M_k in Table 1. A low-pass pole singleton transformation can be derived similarly with a_k set to zero

$$H_k(z) = \frac{1 - z^{-2}}{F_k + G_k z^{-1} + H_k z^{-2} + K_k z^{-3} + M_k z^{-4}} \quad (13)$$

with $F_k \dots M_k$ in Table 1. Realisation of (12) and (13) in canonical form [15] requires normalisation of the F_k , and

Table 1: Transfer function denominator coefficients

	Low-pass pole pair	Low-pass pole singleton
new coefficient	k, N even or $k \neq K, N$ odd	$k = K, N$ odd
F_k	$a_k + b_k + C_k + D_k + E_k$	$b_k + c_k + b_k \omega_0^2$
G_k	$-4a_k - 2b_k + 2D_k + 4E_k$	$2b_k(\omega_0^2 - 1)$
H_k	$6a_k - 2C_k + 6E_k$	$b_k - c_k + b_k \omega_0^2$
K_k	$-4a_k + 2b_k - 2D_k + 4E_k$	0
M_k	$a_k - b_k + C_k - D_k + E_k$	0

so for purposes of realising the filter, the canonical form of $H_k(z)$ is given by $H_{ck}(z) = H_k(z)/F_k$ so that

$$H_{ck}(z) = \frac{\alpha_{k0} + \alpha_{k1} z^{-2} + \alpha_{k2} z^{-4}}{1 + \beta_{k1} z^{-1} + \beta_{k2} z^{-2} + \beta_{k3} z^{-3} + \beta_{k4} z^{-4}} \quad (14)$$

The coefficients in (14) are given in Table 2 and are obtained by collecting the scaled terms from (12) and (13). The $1/F_k$ scaling factors are combined into a *discrete-time multiplier coefficient* A_d , where

$$A_d = \begin{cases} A \prod_{k=0}^{(N/2)-1} \frac{1}{F_k}, & \text{for } N \text{ even} \\ A \prod_{k=0}^{((N-1)/2)-1} \frac{1}{F_k}, & \text{for } N \text{ odd} \end{cases} \quad (15)$$

The discrete-time form of the filter, in a block cascade implementation, is shown in Fig. 4. Following the development given in [15], the canonical form of a filter block that implements (14) is the form that requires the fewest storage elements to implement. This form is created by introducing an intermediate function $W_k(z)$. If $X_k(z)$ and $Y_k(z)$ are the input and output, respectively, of the k th cascaded block of the filter, then the intermediate function $W_k(z)$ is introduced by

$$H_{ck}(z) = \frac{Y_k(z)}{X_k(z)} = \frac{Y_k(z)}{W_k(z)} \cdot \frac{W_k(z)}{X_k(z)} \quad (16)$$

Table 2: Bandpass filter block coefficient definitions

	Low-pass pole pair	Low-pass pole singleton
canonical coefficient	k, N even or $k \neq K, N$ odd	$k = K, N$ odd
α_{k0}	1	1
α_{k1}	-2	-1
α_{k2}	1	0
β_{k1}	G_k/F_k	G_k/F_k
β_{k2}	H_k/F_k	H_k/F_k
β_{k3}	K_k/F_k	0
β_{k4}	M_k/F_k	0

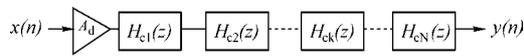


Fig. 4 Discrete-time bandpass Butterworth filter in cascade form

where, by substitution of the numerators and denominators of (14), the following relationships are obtained

$$\frac{Y_k(z)}{W_k(z)} = \alpha_{k0} + \alpha_{k1}z^{-2} + \alpha_{k2}z^{-4} \quad (17)$$

$$\frac{W_k(z)}{X_k(z)} = \frac{1}{1 + \sum_{i=1}^4 \beta_{ki}z^{-i}} \quad (18)$$

Rearranging (17) and (18), the final z -domain form of the equations implementing the canonical form of the filter block is given by

$$Y_k(z) = [\alpha_{k0} + \alpha_{k1}z^{-2} + \alpha_{k2}z^{-4}]W_k(z) \quad (19)$$

$$W_k(z) = X_k(z) - \sum_{i=1}^4 \beta_{ki}z^{-i}W_k(z) \quad (20)$$

and, after inversion from the z -domain to the time-domain, the final form of the block filter equations are obtained

$$y_k(n) = \alpha_{k0}w_k(n) + \alpha_{k1}w_k(n-2) + \alpha_{k2}w_k(n-4) \quad (21)$$

$$w_k(n) = x_k(n) - \sum_{i=1}^4 \beta_{ki}w_k(n-i) \quad (22)$$

In (21) and (22), the index variable n refers to the discretised time increment, that is, $t = n\tau$. In order to facilitate Newton iteration at a given time step, it is necessary to subject the present values of the variables $x_k(n)$, $w_k(n)$, and $y_k(n)$ to Newton iteration. Let a superscripted (j) denote the iterates, then (21) and (22) are modified as follows

$$y_k^{(j)}(n) = \alpha_{k0}w_k^{(j)}(n) + \alpha_{k1}w_k^{(j)}(n-2) + \alpha_{k2}w_k^{(j)}(n-4) \quad (23)$$

$$w_k^{(j)}(n) = x_k^{(j)}(n) - \sum_{i=1}^4 \beta_{ki}w_k^{(j)}(n-i) \quad (24)$$

The block diagram of the k th cascaded block of the filter, reflecting (23) and (24) and in canonical [15] form, is shown in Fig. 5. One new element in Fig. 5 (and not present in [15]) is the storage element Δj required to store values of the Newton iterates of $w_k^{(j)}(n)$ within a time step. Upon convergence at a particular time step, the Z^{-1} storage elements holding $w_k(n-1), \dots, w_k(n-4)$ are updated at the initial iterate of the next time step.

One drawback to the discrete-time filter implementation concerns entries that must be made in the Jacobian matrix to facilitate Newton iteration [16]. In conventional implementations of device models, Jacobian entries [17, 18] are determined through the evaluation of equations resulting from the analytic partial differentiation of continuous-time constitutive equations. Given the inherently discrete-time nature of the filter described here, that method is not available. Instead, the Jacobian entries are formed from a ratio of the difference of the current and previous output iterates to the current and previous input iterates – effectively an implementation of the Secant method. This has the effect of reducing the rate of

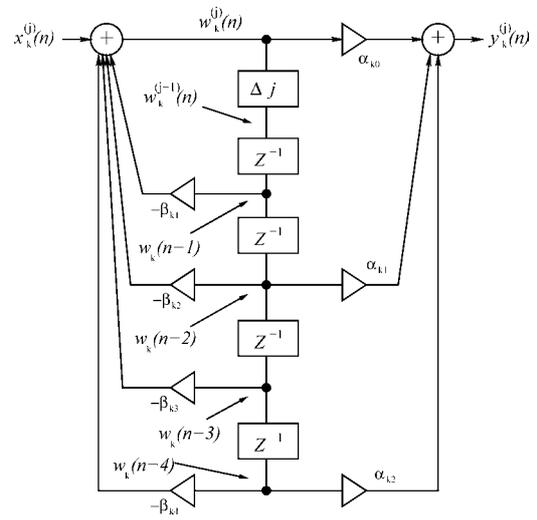


Fig. 5 Typical discrete-time filter block in canonical form

convergence of the Newton iterations from quadratic (i.e. error reduced at a quadratic rate) to superlinear with a resulting rate of $(1 + \sqrt{5})/2 \approx 1.62$ [19].

4 Experimental results and discussion

A discrete-time Butterworth bandpass filter was implemented in fREEDATM [7] using the methodology of the previous section. Validation used a 5-section (having a fifth order low-pass prototype) filter with coaxial resonators having a maximally flat transfer characteristic. The filter has a -3 dB passband of 15.6 MHz, a centre frequency of 1.7 GHz (a fractional bandwidth of 0.9%), and an insertion loss of 1.2 dB. The passband and stopband edge frequencies and insertion loss were extracted from the S_{21} magnitude of the measured filter. These were used as model netlist parameters.

The first step in validation is verifying the frequency response of the simulated filter. This was accomplished by running a series of transient simulations with sinusoidal sources at 0 dBm power levels, then transforming the results to the frequency domain and collecting the filter response for the series at the excitation frequencies. Simulations were run for a time period long enough to assure that initial transients had died out, and the fast Fourier transform (FFT) was taken on 8192 points of ‘trailing data.’ Fig. 6 compares the results achieved by

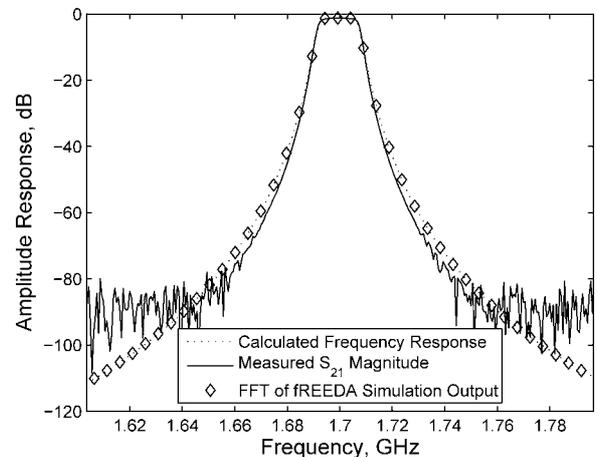


Fig. 6 Simulated (referenced to 0 dBm) and measured frequency response results for a 5-section coaxial filter

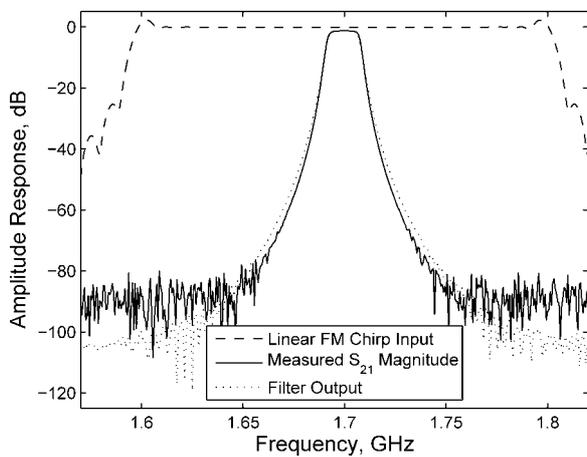


Fig. 7 Response of simulated filter to a linear FM chirp signal

f REEDATM with analytic frequency response calculations and the measured S_{21} magnitude data for the coaxial filter. Agreement is seen in the vicinity of the passband and down the stopband edges to about -80 dBm where the noise floor of the measuring equipment dominates.

Next, the filter was excited by a linear FM chirp signal with a 200 MHz chirp range centred at 1.7 GHz, and the FFT of the result was taken. The power level of chirp source was normalised such that the flat portion of the chirp was at 0 dBm. The linear FM chirp signal is the type of waveform used in radar [2], a system that is difficult to characterise using harmonic balance and traditional circuit simulators. Fig. 7 shows that excellent agreement is seen between the response of the simulated filter to the simulated chirp signal and the measured S_{21} magnitude data for the coaxial filter. The filter order and fractional bandwidth are noted in Fig. 1 where it is seen that the filter characteristics cannot be simulated using a conventional transient circuit simulation technique. Furthermore, the high-dynamic range [20] of f REEDATM enables the filter response to be modelled over a very wide dynamic range.

5 Conclusion

A simulation model for a maximally flat discrete-time band-pass filter suitable for transient circuit simulation was presented. The model is effectively implemented in the z -domain so that a much higher-order filter can be modelled than that achievable with a more conventional implementation based on a pole-zero s -domain transfer function. Owing to the robustness of the f REEDATM transient simulator engine, Newton iteration convergence is achieved using a fixed time step, making the simulation of a discrete-time behavioural filter model possible. In order to deploy

discrete-time filter models in other transient simulator environments, it is necessary to use a fixed time step and avoid the default adaptive time stepping of the Spice engine [21]. More information on f REEDATM is available at <http://www.freeda.org>.

6 Acknowledgment

This material is based upon the work supported in part by the U.S. Army Research Laboratory and the U.S. Army Research Office on Multifunctional Adaptive Radio Radar and Sensors (MARRS) under grant number DAAD19-01-1-0496. The authors thank Harris Corporation for their assistance.

7 References

- 1 Brambilla, A., Premoli, A., and Storti-Gajani, G.: 'Recasting modified nodal analysis to improve reliability in numerical circuit simulation', *IEEE Trans. Circuits Syst. I, Regul. Pap.*, 2005, **52**, (3), pp. 522–534
- 2 Peebles, P.Z. Jr.: 'Radar principles' (Wiley-Interscience, 1998)
- 3 Rudd, E.P.: 'The solution of ill-scaled circuit equations', *IEEE Trans. Circuits Syst.*, 1984, **31**, (7), pp. 671–672
- 4 Nagel, L.W.: 'SPICE2, A computer program to simulate semiconductor circuits'. Technical Report ERL-M520 (University of California at Berkeley, 1975)
- 5 Gottling, J.E.: 'Node and mesh analysis by inspection', *IEEE Trans. Educ.*, 1995, **38**, (4), pp. 312–316
- 6 Su, K.: 'Analog filters' (Kluwer Academic Publishers, 2002, 2nd edn.)
- 7 Christoffersen, C.E., Mughal, U.A., and Steer, M.B.: 'Object oriented microwave circuit simulation', *Int. J. RF Microw. Comput. Aided Eng.*, 2000, **10**, (3), pp. 164–182
- 8 Steiglitz, K.: 'The equivalence of digital and analog signal processing', *Inf. Control*, 1965, **8**, (5), pp. 455–467
- 9 Oppenheim, A.V., and Shafer, R.W.: 'Discrete-time signal processing' (Prentice-Hall, 1989)
- 10 Steiglitz, K.: 'An introduction to discrete systems' (John Wiley & Sons, 1974)
- 11 Chua, L.O., and Lin, P.M.: 'Computer-aided analysis of electronic circuits: algorithms and computational techniques' (Prentice-Hall, 1975)
- 12 Ifeachor, E.C., and Jervis, B.W.: 'Digital signal processing: a practical approach' (Prentice-Hall, 2002, 2nd edn.)
- 13 Stearns, S.D., and Hush, D.R.: 'Digital signal analysis' (Prentice-Hall, 1990, 2nd edn.)
- 14 Parks, T.W., and Burrus, C.S.: 'Digital filter design' (Wiley-Interscience, 1987)
- 15 Bozic, S.M.: 'Digital and Kalman filtering' (Edward Arnold, 1994, 2nd edn.)
- 16 Kelley, C.T.: 'Solving nonlinear equations with Newton's method' (SIAM, 2003)
- 17 McCalla, W.J.: 'Fundamentals of computer-aided circuit simulation' (Kluwer Academic Publishers, 1988)
- 18 Pillage, L.T., Rohrer, R.A., and Visweswariah, C.: 'Electronic circuit system simulation methods' (McGraw-Hill, 1995)
- 19 Ralston, A., and Rabinowitz, P.: 'A first course in numerical analysis' (Dover, 2001, 2nd edn.)
- 20 Luniya, S.R., Steer, M.B., and Christoffersen, C.E.: 'High dynamic range transient simulation of microwave circuits'. Proc. IEEE Radio and Wireless Conf. (RAWCON) 2004, Atlanta, GA, USA, September 2004, pp. 487–490
- 21 Vladimirescu, A.: 'The spice book' (John Wiley & Sons, 1994)