Reduced-Order Wide-Band Interconnect Model Realization using Filter-Based Spline Interpolation

Arthur Nieuwoudt Mehboob Alam Yehia Massoud

Department of Electrical and Computer Engineering
Rice University
Houston, TX 77005
Tel: 713-348-6706
Fax: 713-348-6196
e-mail: {abnieu,alam,massoud}@rice.edu

Abstract— In the paper, we develop a systematic methodology for modeling sampled interconnect frequency response data based on spline interpolation. Through piecewise polynomial interpolation, we are able to avoid the numerical problems associated with global polynomial fitting and generate higher order systems to model simulated or measured wideband frequency response data. We reduce the complexity of the generated systems using a data point pruning algorithm and by applying model order reduction based on balanced truncation. The methodology provides substantially greater accuracy than global polynomial approximation while only having O(n) growth in model complexity.

I. Introduction

As the complexity of on-chip mixed-signal systems increases, the accurate and efficient modeling of interconnect is vital for estimating and controlling delay, cross-talk noise, and power consumption [1–3]. In today's digital design flow, interconnect is extracted using 2D or 3D field-solvers or analytical models. Frequency domain field solvers produce interconnect resistance, capacitance, and inductance values at a particular frequency [4–6]. Analytical models for interconnect characteristics also typically produce frequency dependent data [1–3, 7]. Consequently, for interconnect characterization, systematic methods are required to produce accurate wideband models that are needed for time-domain simulation.

Previous research on the wideband modeling of sampled data has primarily focused on fitting the frequency response using a single polynomial interpolant. Many popular techniques approximate the frequency response of the system using linear least squares regression to obtain the transfer function coefficients [8, 9]. This requires the solution to a polynomial fitting problem with the Vandermonde matrix as its basis, which can become ill-conditioned for wideband rational interpolation problems. Another common approach is to fit the real and complex poles directly [10, 11], but this requires suitable estimates for the poles to serve as start points for the least squares algorithm. In [12], the Krylov subspace generated by the Arnoldi process is utilized as the basis set for interpolation, but for higher order systems, these basis vectors can lose orthogonality and may be computationally expensive unless other numerical techniques are applied [13].

The primary limitation of all of the aforementioned algorithms is that they attempt to approximate the transfer function data globally across a wide-range of frequencies with only one rational polynomial realization. Fundamental limits exist on the degree of the rational function that can be generated due to the oscillatory behavior of high-order fitted polynomials and ill-conditioning at higher frequencies [14]. While the global behavior of polynomial or other interpolants can be problematic, they are well-suited for local approximation across a narrow-range of frequencies. Therefore, piecewise polynomial interpolation using *splines* is an attractive alternative for modeling complex transfer function data across a wide-range of frequencies.

In this paper, we present a systematic methodology for generating stable wideband models of sampled transfer function data for interconnect applications. The approach is based on smoothing spline interpolation, which generates a series of piecewise polynomials to individually model narrow frequency ranges of the simulated data in order to avoid the ill-conditioning of global polynomial approximation techniques. We reduce the complexity of the generated systems using a data point pruning algorithm and by applying model order reduction based on balanced truncation [15]. We compare the spline interpolation methodology with the global polynomial fitting techniques presented in [9] and [10] on several complex interconnect related design problems. The results indicate that the methodology has substantially greater accuracy when compared with global polynomial fitting techniques while only having $O(n)$ growth in system generation time, system evaluation time, and memory as the number of data samples (n) increases.

II. Transfer Function Approximation Problem

A. Problem Formulation

The input for the interpolation problem is a series of sampled points for a particular linear time-invariant system's transfer function, $H(s_1)$, $H(s_2)$, ... , $H(s_n)$, for n frequencies. The objective is to create a system with a response that approximates the sampled data at every frequency. The system can be modeled using its rational polynomial function representation,

$$H(s) = \frac{a_0 + a_1 s + a_2 s^2 + ... + a_J s^J}{b_0 + b_1 s + b_2 s^2 + ... + b_K s^K} \quad (1)$$

where the coefficients a_0, a_1, ..., a_J and b_0, b_1, ..., b_K determine the system's zeros and poles. Since (1) contains higher powers of s, this formulation can lead to ill-conditioned systems. Equivalently, the transfer function can be expressed as

$$H(s) = C(sI - A)^{-1}B + D \tag{2}$$

for a system that is represented in state space form,

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \tag{3}$$

where $A \in \Re^{nxn}$, $B \in \Re^{nxm}$, $C \in \Re^{pxn}$, and $D \in \Re^{pxm}$ define the linear maps between the system's state variables, inputs, and outputs. The variable x represents the internal system state, and u and y are the input and output vectors, respectively.

B. Polynomial Fitting of Frequency Response Data

Following the formulation from [9], consider the transfer function described at each frequency point,

$$H(s_i) = \frac{a_0 + a_1 s_i + \cdots + a_J s_i^J}{b_0 + b_1 s_i + \cdots + b_K s_i^K}. \tag{4}$$

Multiplying both sides of (4) by the denominator yields

$$H(s_i)(b_0 + b_1 s_i + \cdots + b_K s_i^K) = a_0 + a_1 s_i + \cdots + a_J s_i^J. \tag{5}$$

By combining (5) for each frequency point, s_i, into matrix form, we form the following system of equations

$$[N, \; -D] \begin{bmatrix} a \\ b \end{bmatrix} = 0 \tag{6}$$

where

$$\begin{aligned} N &= \begin{bmatrix} 1 & s_1 & \cdots & s_1^J \\ 1 & s_2 & \cdots & s_2^J \\ \vdots & \vdots & \ddots & \vdots \\ 1 & s_n & \cdots & s_n^J \end{bmatrix} \\ D &= \begin{bmatrix} H(s_1) & H(s_1)s_1 & \cdots & H(s_1)s_1^J \\ H(s_2) & H(s_2)s_2 & \cdots & H(s_2)s_2^J \\ \vdots & \vdots & \ddots & \vdots \\ H(s_n) & H(s_n)s_n & \cdots & H(s_n)s_n^J \end{bmatrix} \\ a &= \begin{bmatrix} a_0 & a_1 & \cdots & a_J \end{bmatrix}^T \\ b &= \begin{bmatrix} b_0 & b_1 & \cdots & b_J \end{bmatrix}^T. \end{aligned} \tag{7}$$

To model the complex valued frequency response data, the matrices in (7) must be segmented into their real and imaginary components. To solve the system of equations described in (6) exactly, we must have $J + K + 2$ sampled data points. If we have more than $J + K + 2$ sampled data points, we have an overdetermined system of equations, in which case we can apply linear least squares regression. To guarantee that the system generated by solving (6) is stable, we must remove any positive real poles from the formulation. The poles of the generated system can be determined by solving the eigenvalue problem $B_{eig}x = \Lambda x$, where Λ is a matrix with diagonal entries that contain the poles of the approximate system, and B_{eig} is defined in [9]. After removing the poles with positive real parts, the denominator of (1) can then be reconstructed and the numerator regenerated to match the frequency response data [9].

Note that if the order of the numerator or denominator of the transfer function is too large, then the system will become ill-conditioned since the difference between s^J or s^K and s^0 will increase. A common approach to reduce the impact of this problem is to normalize the frequency using a factor such as $\omega_i = \sqrt{\omega_{min}\omega_{max}}$ where ω_{min} and ω_{max} are the minimum and maximum angular frequencies of the sampled frequency response data [8]. For narrow-band applications with a sample frequency spread of 1 or 2 orders of magnitude, the ill-conditioning is largely avoided for approximate systems up to 10th order. However, for larger frequency ranges or higher order system approximations, normalization cannot prevent matrix ill-conditioning, which can lead to numerical errors during matrix inversion.

III. Spline Interpolation of Sampled Data

In this section, we present the spline interpolation methodology for generating systems to model sampled frequency response data. Initially, we re-sample, smooth, and prune data points in order to reduce the complexity of the system as described in Section III-A. Using the pruned and re-sampled data points, we then create a piecewise polynomial transfer function for each interval between the re-sampled frequency points. We also generate a filter bank for the frequency point intervals. Each piecewise polynomial transfer function is then convolved with each filter in the filter bank as described in Section III-B. When the convolved piecewise transfer function-filter bank systems are combined, we have a model for the system's wide-band response. Finally, by judicially applying model order reduction techniques such as balanced truncation, we are able to generate lower order systems as explained in Section III-C. Overall, our methodology provides a flexible means for generating stable approximations of frequency response data.

A. Re-Sampling, Smoothing and Pruning using Splines

To reduce the complexity of the generated models, we utilize cubic spline interpolation to re-sample, smooth, and prune the frequency response data. In this work, we use cubic polynomial smoothing splines to describe the transfer function between sample points, which are described as

$$S(\omega) = \left\{ \begin{array}{ccc} S_1(S(\omega)) & \forall & \omega \in [\omega_1, \omega_2] \\ S_2(S(\omega)) & \forall & \omega \in [\omega_2, \omega_3] \\ \vdots & \vdots & \vdots \\ S_{n-1}(S(\omega)) & \forall & \omega \in [\omega_{n-1}, \omega_n] \end{array} \right\} \tag{8}$$

where S_i are piecewise cubic spline functions that describe the transfer function on a particular interval.

To decrease the number of data points required to interpolate the data, we employ a recursive algorithm to re-sample and prune the data points utilized for interpolation. First, we generate a smoothing cubic spline interpolation function for the original frequency response data, which we use as the baseline

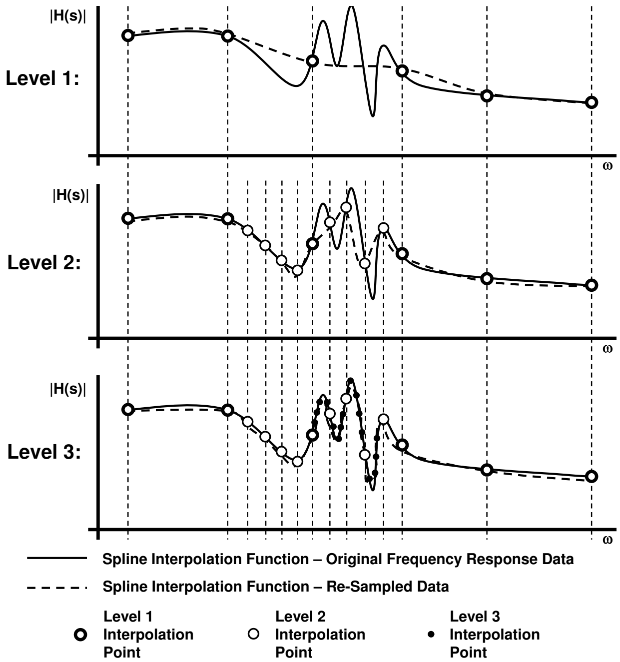

Fig. 1. Graphic illustration of the smoothing, re-sampling, and pruning algorithm. The algorithm recursively re-samples the spline interpolants for the sampled data based on the smoothness of a particular interval.

model to determine the accuracy of the re-sampled and pruned data. This initial baseline spline function also serves to smooth the noise that can inherently exist in the frequency response data. We then recursively re-sample the data and apply a spline interpolation function to determine the accuracy of the approximate spline interpolant on each sub-interval as compared to the original data's spline interpolated response function. To ensure the accuracy of the approximate spline interpolant, we evaluate both the original spline function and the approximated spline function for 1000 data points on the given interval, and if the maximum relative error exceeds 0.25 percent, we perform finer grain sampling on the sub-region. Figure 1 depicts this recursive process. On relatively smooth sub-intervals, a spline interpolant created with only a few data points closely matches the spline interpolation function generated using the original frequency response data. On intervals with more complex behavior, we provide more fine grain sampling to more closely match the sampled data.

B. Filter Bank and Piecewise Transfer Function Construction

Once we construct the spline approximation of the sampled data, we must generate a state space representation of the transfer function to utilize the interpolated system for time domain simulations. Each piecewise approximation of the transfer function is convolved with a bandpass filter with a passband that encompasses each frequency interval. A block diagram of the final system is displayed in Figure 2. In order to generate the analog filter bank, we create a series of bandpass filters for the frequency ranges prescribed by the re-sampled interpolation points. Since the response of each filter is added to obtain the final system's frequency response, the stop-band attenuation must be as great as possible to eliminate noise in regions of the transfer function with low magnitude. In this paper, we use elliptical filters with 0.001 dB ripple in the passband and 120 dB attenuation in the stop-band. For applications where the phase response of the system is important, filters with more linear phase such as Bessel or Butterworth filters can be employed in the filter bank [16]. While the re-sampling process ensures that the original frequency data points are not on the filter bank boundaries, we would still like to generate the smoothest frequency response possible. In order to minimize the impact of the overlap error, we follow two approaches: (1) equalize the normalized roll-off bandwidth of each filter and (2) optimize the position of each filter to reduce overlap error.

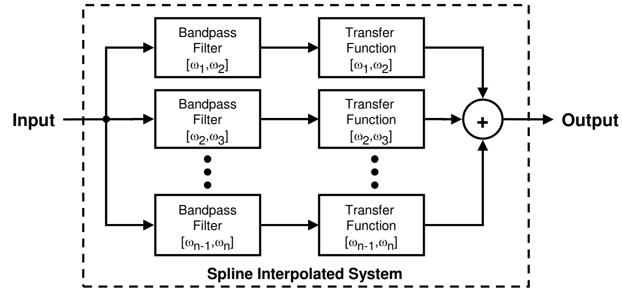

Fig. 2. Block diagram of system generated using our methodology.

The normalized steepness of the filter's roll-off bandwidth is primarily determined by its total bandwidth, passband ripple, stop-band attenuation, and filter order. Since the total bandwidth, passband ripple, and stop-band attenuation are determined by the frequency range for each spline interpolant and the required accuracy of the transfer function in the passband, the filter order is the only degree of freedom available for adjusting the normalized roll-off bandwidth. We adjust the filter order using bisection until the given order filter has a response with the desired normalized steepness in roll-off bandwidth. Since the required order is not likely to significantly vary for adjacent filters in the filter bank, we first try to locate the required filter order over a small range near the previous filter's order before performing bisection on the entire range of possible filter order values.

To further improve the frequency response of the filter bank at the boundaries between the bandpass filters, we utilize nonlinear constrained optimization in order to adjust the filter's bandwidth to minimize the overlap error. When constructing the filter bank, we iteratively optimize the low frequency cutoff of each new filter added to the filter bank based on the following optimization problem formulation:

$$\begin{array}{ll} \text{Minimize} & \| \overrightarrow{F_{fb}(\overrightarrow{s_s})} - 1 \|_2^2 \\ \text{Subject to} & \left(\begin{array}{c} \overrightarrow{F_{fb}(\overrightarrow{s_s})} \leq 1 \\ \overrightarrow{s_{olmin}} \leq \overrightarrow{s_s} \leq \overrightarrow{s_{olmax}} \end{array} \right) \end{array} \quad (9)$$

where F_{fb} is the frequency response of the filter bank, $\overrightarrow{s_s}$ is a vector of frequency points near the filter bank overlap frequency, and s_{olmin} and s_{olmax} are the minimum and maximum frequencies where overlap distortion is possible. The first constraint limits the filter bank's overlap distortion to be less than the nominal 0 dB response. In this paper, we utilize Sequential Quadratic Programming [17] to solve the problem in (9).

Combining each of the convolved piecewise transfer functions together produces an A matrix with a block diagonal sparsity pattern. This stems from the fact that the addition of two systems in state space representation is

$$A = \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix}, \; B = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix}, \\ C = [C_1 \; C_2]\,, \; D = D_1 + D_2. \tag{10}$$

While the overall system dimension grows as the number of interpolated frequency response points increases, when evaluating the system using sparse block diagonal routines, the complexity of generating the system and evaluating its response is only $O(n)$ for n sample points in system generation time, system evaluation time, and memory. This is due to the fact that at most, each frequency point requires one local transfer function generation and one additional bandpass filter in the filter bank. Therefore, we can efficiently generate systems with complex frequency responses based on many sampled frequency data points.

C. Block Model Order Reduction using Balanced Truncation

Given the relatively smooth frequency response of the final system, model order reduction (MOR) techniques have the potential to significantly reduce the order of the generated system. The methods used for the MOR of linear systems fall into two main categories: Singular Value Decomposition (SVD) methods [15] and Krylov subspace projection methods [18]. The benefits of SVD-based methods such as balanced truncation include a guaranteed error bound and system stability. For large scale systems, moment-matching methods provide a tractable alternative to SVD methods. However, unlike SVD methods, there is no guarantee of stability or an error bound.

Both classes of MOR techniques destroy the sparsity pattern of the original system. Therefore, applying MOR to our full interpolated system is inefficient and in many cases intractable. However, we can apply MOR to individual or sets of system sub-blocks in the A matrix. Given that the order of each sub-block will typically be less than 100, we utilize balanced truncation due to its guaranteed error bound and stability. We employ the algorithm for balanced truncation described in [15] to avoid matrix inversion and ill-conditioned solutions. In addition to providing stable approximations with bounded error, the balanced truncation approach can also be modified to preserve the passivity of the original system [15].

D. Stability Analysis

The spline interpolation methodology consists of additions and multiplications of the rational transfer function (R_i) and bandpass filter (F_i) for a particular interval, i, as displayed in Figure 2. The final system (H_f) can be expressed as

$$H_f = \sum_{i=1}^{n-1} R_i \cdot F_i \tag{11}$$

for a spline interpolated system with $n-1$ intervals. Since we employ the unstable pole removal technique on the individual

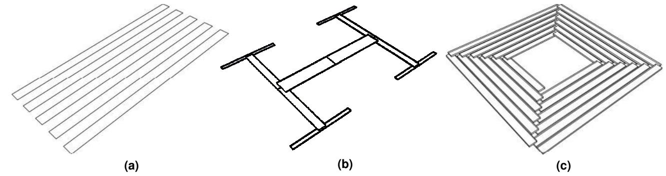

Fig. 3. Interconnect bus (a), H-tree clock network (b), and integrated spiral inductor (c) design examples for the spline interpolation methodology.

rational transfer functions, the individual systems R_i are stable. For a state space system of the form presented in (3), if all of the eigenvalues of the matrix A have negative imaginary parts, then the system is stable.

Since the individual R_i and F_i systems are stable, H_f is stable. To prove this point, we need to show that the stable eigenvalues of the constituent systems (R_i and F_i) are preserved in constructing H_f using the formulation in (11). For all of the intervals in the system, R_i and F_i are multiplied together in the frequency domain. Multiplying two state space systems together produces a new system defined as

$$A = \begin{bmatrix} A_1 & B_1C_2 \\ 0 & A_2 \end{bmatrix}, \; B = \begin{bmatrix} B_1D_2 \\ B_2 \end{bmatrix}, \\ C = [C_1 \; D_1C_2]\,, \; D = D_1D_2. \tag{12}$$

Note that A is a block upper diagonal system with A_1 and A_2 on the diagonal. Therefore, the eigenvalues of A are simply the eigenvalues of A_1 and A_2. Consequently, the multiplication of R_i and F_i produces a stable system. Using a similar argument for the addition of two state space systems, which is defined in (10), if the two state space systems being added are stable, then their summation is stable. Since the formation of H_f consists of multiplications and additions of stable systems, then the final system produced by the spline interpolation methodology must be a stable system. Furthermore, the balanced truncation model order reduction technique is also guaranteed to preserve system stability [15].

While the spline interpolation methodology and balanced truncation guarantees the generated system's stability, it does not necessarily ensure passivity. However, to obtain a passive system, the convex optimization techniques described in [19] can be applied to each of the state space sub-blocks generated by our methodology. Since these matrices have a relatively low order, typically less than 100, the numerical complexity of using the method in [19] will be relatively low. Consequently, the proposed method provides a tractable solution for generating stable wideband models for a large number of frequency response data samples with complex behavior.

IV. Results

To illustrate the efficacy of the spline interpolation methodology, we approximate the overall system response from sampled frequency domain data collected from several interconnect structures including an interconnect bus, H-tree clock network, and spiral inductor, which are depicted in Figure 3. Table I lists the interpolation problem characteristics and generated system

TABLE I
DESIGN PROBLEM CHARACTERISTICS

	Inductor	Bus	H-Tree
Data Points	100	500	2000
Minimum Frequency (Hz)	10^8	10^2	10^2
Maximum Frequency (Hz)	10^{11}	10^{15}	10^{12}
Filter Bank Size	23	95	611
Non-zero Entries in A	11514	47944	273339
Equivalent System Order	108	296	523
Equivalent Balanced Truncation System Order	62	174	300
System Generation Time (s)	37.2	131.4	503.4

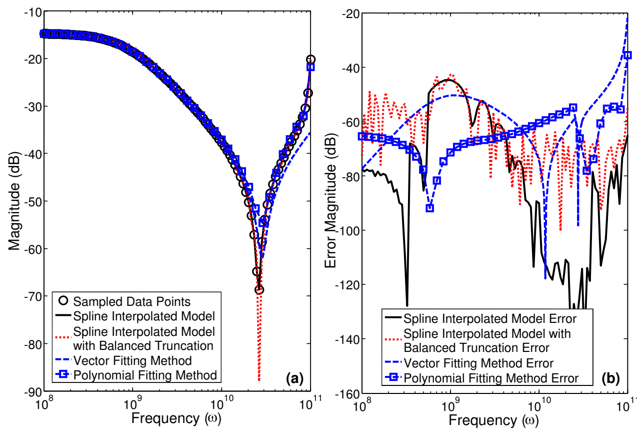

Fig. 4. Frequency response (a) and error (b) of the systems generated to model the spiral inductor. Since the sampled data points provide a relatively smooth response over the narrow-band of frequencies simulated, each of the techniques provides a relatively accurate approximation of the sampled data.

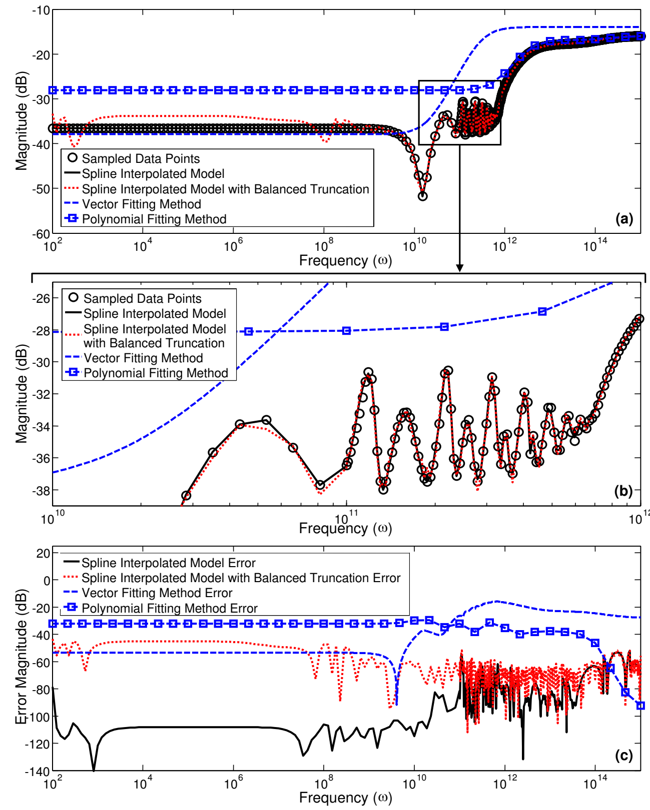

Fig. 5. Frequency response (a, b) and error (c) of the systems generated to model the interconnect bus data. The models generated using the spline interpolation technique closely match the fine detail in the sampled data while the vector and polynomial fitting methods only provide coarse-grained approximate models.

statistics. We create the models for each of these structures using the field solvers FastCap and FastHenry for capacitance and inductance extraction [5, 6] and combine the results using modified nodal analysis [18]. From this RLC model, we then generate the sampled frequency points for each of the interconnect structures. Given the sampled transfer function data, we then utilize the spline interpolation methodology to create a state space system that approximates the sampled response. We compare the accuracy of our technique with the vector fitting method presented in [10] and the polynomial fitting method described in [9].

We generated a system from data points obtained from a six turn spiral inductor, which is crucial in RF applications [20,21]. For this example, we simulated 100 frequency points over a frequency range of 10^8 through 10^{11} radians per second. Figure 4 displays the frequency response of the system generated using the spline interpolation method. The systems generated using the proposed method with and without balanced truncation have maximum errors of -44.7 dB and -42.7 dB, respectively. The balanced truncation version of the algorithm reduces the overall system order by 42 percent. The systems generated using the vector and polynomial fitting techniques have errors of -21.7 dB and -35.5 dB, respectively. Since the sampled data points have a smooth frequency response over a narrow-band of frequencies, each of the techniques provide a relatively accurate approximation.

To demonstrate the accuracy of the spline interpolation methodology for more complex sets of sampled data, we generated systems from data points obtained from both an interconnect bus and a H-tree clock network. Figures 5 and 6 display the frequency responses of the models generated using the spline interpolation methodology and the vector and polynomial fitting techniques for the bus and H-tree structures, which have 500 and 2000 sample data points, respectively. For these structures, many sample data points are required to capture the complex frequency response. The sample data points are distributed over a large frequency range and exhibit significantly greater complexity than the spiral inductor example. For the bus example, the spline interpolation method closely matches the frequency response of the sampled data points with a maximum error of -51.9 dB, while the method with balanced truncation only increases the maximum error to -44.6 dB. In contrast, the vector and polynomial fitting techniques fail to generate models that match the frequency response of the sampled data and have maximum errors of -21.5 dB and -28.9 dB, respectively. As depicted in Figures 5b and 6b, the models generated using the spline interpolation method closely match the fine detail of the sampled data for the bus and H-tree examples, while the vector and polynomial fitting techniques only provide

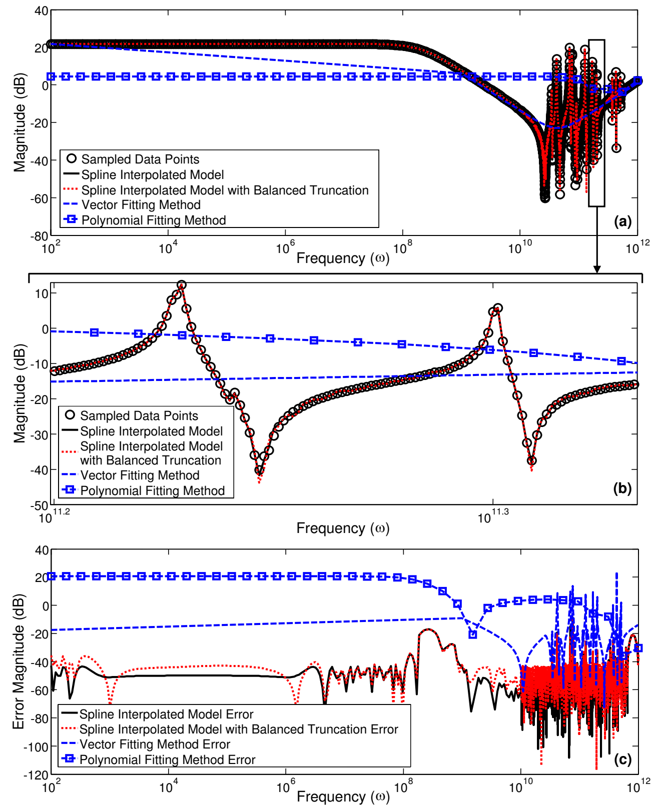

Fig. 6. Frequency response (a, b) and error (c) of the systems generated to model the H-tree data. Note that the systems generated using the spline interpolation methodology almost completely match the sampled data.

coarse-grained approximate models. Therefore, the spline interpolation methodology provides a tractable solution for generating accurate state space models from sampled frequency response data with complex wideband behavior.

V. Conclusion

In the paper, we developed a methodology for interpolating sampled frequency response data based on smoothing spline interpolation, which is vital for the accurate and efficient modeling of on-chip interconnect. By using piecewise polynomial interpolation, we are able to avoid the numerical problems associated with global polynomial approximation and generate higher order systems. The methodology has substantially greater accuracy when compared with global polynomial fitting while only having $O(n)$ complexity. The methodology's flexibility, computational efficiency, and accuracy will enable the creation of wideband interconnect models suitable for time-domain simulation, which will facilitate the characterization and mitigation of delay, cross-talk noise, and power consumption for tomorrow's nanoscale integrated circuits.

References

[1] A. Nieuwoudt, M. S. McCorquodale, R. T. Borno, and Y. Massoud, "Efficient Analytical Modeling Techniques for Rapid Integrated Spiral Inductor Prototyping," in *Proc. IEEE Custom Int. Cir. Conf.*, Sept. 2005, pp. 274–277.

[2] M. Mondal and Y. Massoud, "Reducing Pessimism in RLC Delay Estimation Using an Accurate Analytical Frequency Dependent Model for Inductance," in *Proc. IEEE Intl. Conf. on CAD*, Nov. 2005, pp. 691–696.

[3] Y. Cao, X. Huang, D. Sylvester, H. Wan, T.-J. King, and C. Hu, "Impact of On-Chip Interconnect Frequency Dependent R(f) L(f) on Digital and RF Design," *IEEE Trans. VLSI Sys.*, vol. 13, no. 1, pp. 158–162, Jan. 2005.

[4] Y. Massoud and J. White, "FastMag: a 3-D Fast Inductance Extraction Program for Structures with Permeable Materials," in *Proc. IEEE Intl. Conf. on CAD*, Nov. 2002, pp. 478–484.

[5] M. Kamon, M. J. Tsuk, and J. White, "Fasthenry: A Multipole-Accelerated 3-D Inductance Extraction Program," *IEEE Trans. Micro. Theory and Tech.*, vol. 42, no. 9, pp. 1750–1758, Sept. 1994.

[6] K. Nabors and J. White, "Fastcap: A Multipole Accelerated 3-D Capacitance Extraction Program," *IEEE Trans. CAD*, vol. 10, no. 11, pp. 1447–1459, Nov. 1991.

[7] C. S. Amin, F. Dartu, and Y. I. Ismail, "Weibull Based Analytical Waveform Model," *IEEE Trans. CAD*, pp. 1156–1168, vol. 24, no. 8, Aug. 2005.

[8] M. Elzinga, K. L. Virga, L. Zhao, and J. L. Prince, "Pole-Residue Formulation for Transient Simulation of High-Frequency Interconnects using Householder LS Curve-Fitting Technique," *IEEE Trans. Adv. Pack.*, vol. 23, no. 2, pp. 142–147, May 2000.

[9] B. Young, *Digital Signal Integrity*. Prentice Hall, 2001.

[10] B. Gustavsen and A. Semlyen, "Rational Approximation of Frequency Domain Responses by Vector Fitting," *IEEE Trans. Power Delivery*, vol. 14, no. 3, pp. 1052–1061, July 1999.

[11] D. Saraswat, R. Achar, and M. S. Nakhla, "A Fast Algorithm and Practical Considerations for Passive Macromodeling of Measured/Simulated Data," *IEEE Trans. Adv. Pack.*, vol. 27, no. 1, pp. 57–70, Feb. 2004.

[12] C. P. Coelho, J. Phillips, and L. M. Silveira, "Robust Rational Function Approximation Algorithm for Model Generation," in *Proc. IEEE Design Automation Conf.*, June 1999, pp. 207–212.

[13] R. B. Lehoucq and D. C. Sorensen, "Deflation Techniques for an Implicitly Restarted Arnoldi Iteration," *SIAM J. Matrix Anal. Appl.*, vol. 17, no. 4, pp. 789–821, Oct. 1996.

[14] L. Tang and J. D. Baeder, "Uniformly Accurate Finite Difference Schemes for p-Refinement," *SIAM J. Sci. Comp.*, vol. 20, no. 3, pp. 1115–1131, May 1999.

[15] J. Phillips, L. Daniel, and L. M. Silveira, "Guaranteed Passive Balancing Transformations for Model Order Reduction," *IEEE Trans. CAD*, vol. 22, no. 8, pp. 1027–1041, Aug. 2003.

[16] R. Schaumann and M. E. van Valkenburg, *Design of Analog Filters*. Oxford U.P., 2001.

[17] A. Nieuwoudt and Y. Massoud, "Multi-level Approach for Integrated Spiral Inductor Optimization," in *Proc. IEEE Design Automation Conf.*, June 2005, pp. 648–651.

[18] A. Odabasioglu, M. Celik, and L. T. Pileggi, "PRIMA: Passive Reduced-Order Interconnect Macromodeling Algorithm," *IEEE Trans. CAD*, vol. 17, no. 8, pp. 645–654, Aug. 1998.

[19] C. P. Coelho, J. Phillips, and L. M. Silveira, "A Convex Programming Approach for Generating Guaranteed Passive Approximations to Tabulated Frequency-Data," *IEEE Trans. CAD*, vol. 23, no. 2, pp. 293–301, Feb. 2004.

[20] A. Nieuwoudt and Y. Massoud, "Robust Automated Synthesis Methodology for Integrated Spiral Inductors with Variability," in *Proc. IEEE Intl. Conf. on CAD*, Nov. 2005, pp. 502–508.

[21] A. Nieuwoudt, T. Ragheb, and Y. Massoud, "SOC-NLNA: Synthesis and Optimization for Fully Integrated Narrow-Band CMOS Low Noise Amplifiers," in *Proc. IEEE Design Automation Conf.*, July 2006, pp. 879–884.