

A Barycentric Vector Fitting Algorithm for Efficient Macromodeling of Linear Multiport Systems

Dirk Deschrijver, *Member, IEEE*, Luc Knockaert, *Senior Member, IEEE*, and Tom Dhaene, *Senior Member, IEEE*

Abstract—Common-pole modeling of frequency responses can be time consuming for systems with highly resonant responses. This letter presents a Barycentric Vector Fitting algorithm that improves the running time of existing Vector Fitting algorithms. The effectiveness of the approach is demonstrated by applying it to coupled microstrip lines. Numerical results confirm the accuracy of the model and the savings in computation time.

Index Terms—Barycentric interpolation, macromodeling, system identification, transfer functions, vector fitting (VF).

I. INTRODUCTION

RATIONAL modeling of frequency responses has received a lot of attention over the past decade. Such models are very important for efficient time domain and frequency domain simulation of microwave systems. When compared to other rational fitting methods, the Vector fitting (VF) algorithm was found to be a highly robust and efficient method, applicable to both smooth and sharply resonant responses of high order and wide frequency bands [1], [2]. Over the past decade, a lot of improvements to VF have been proposed. In [3], a fast QR step was proposed to reduce the computational complexity of the algorithm when fitting multi-port systems with many ports and poles. It was shown in [4] that also the convergence properties can be enhanced in the presence of noise by adding an additional relaxation constraint. This combined approach has become the de-facto standard algorithm, available on [5].

This letter presents a barycentric vector fitting algorithm that is able to compute models with comparable accuracy at a reduced computational cost. In particular, the computation time of the pole-identification step is reduced by exploiting the specific form of the barycentric interpolation formula. Numerical results confirm that this leads to significant savings in terms of computation time, while preserving a good accuracy. Another key difference with the existing VF algorithms is that the new approach does not require an a-priori choice of starting poles, which is often based on a heuristic procedure [1].

II. OUTLINE OF THE ALGORITHM

The goal of the algorithm is to compute a macromodel that approximates all scattering elements $H_v(s)$ (for $v = 1, \dots, V$)

Manuscript received June 26, 2012; revised November 07, 2012; accepted January 03, 2013. Date of publication January 18, 2013; date of current version February 08, 2013. This work was supported by Interuniversity Attraction Poles Programme BESTCOM initiated by Belgian Science Policy Office, and also by the Research Foundation Flanders (FWO-Vlaanderen).

The authors are with the Department of Information Technology, Ghent University-iMinds, Gent, Belgium (e-mail: dirk.deschrijver@intec.ugent.be).

Color versions of one or more of the figures in this letter are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/LMWC.2013.2239282

of a transfer matrix with a common set of P poles over a given frequency range of interest $[s_{\min}, s_{\max}]$. All data samples are split in 2 disjoint sets \bar{S} and S , such that $\bar{S} \cap S = \emptyset$

$$\bar{S} = \{\bar{s}_m, H_v(\bar{s}_m)\}_{m=1}^M, S = \{s_k, H_v(s_k)\}_{k=1}^K. \quad (1)$$

\bar{S} contains a small amount of $M = P/2$ samples that are equally spaced over the frequency range [6], whereas S comprises the K remaining data samples (usually $K \gg M$). First, the algorithm computes a causal rational model that interpolates all data in \bar{S} (Section III). Then, the coefficients of this model are iteratively calculated such that the model also approximates the data S in a least-squares (LS) sense (Section IV). In each step of the iteration, the starting poles of the model are relocated to a better position. This process is called pole-identification. Once the accuracy of the model has converged or stalled, the iteration is terminated and the poles of the final model are found by solving an eigenvalue problem. Once these poles are calculated, all available data in $S \cup \bar{S}$ can be used to identify the corresponding residues as a linear LS problem, see [1].

III. RATIONAL MODEL REPRESENTATION

A. Barycentric Interpolation Formula

A rational function $R_v(s)$ that interpolates all the data in \bar{S} can be obtained by applying the barycentric interpolation formula. This formula yields an interpolating transfer function for any choice of non-zero barycentric weights ($w_m \neq 0$) [7]

$$R_v(s) = \frac{N_v(s)}{D(s)} = \frac{\sum_{m=1}^M \frac{w_m}{s - \bar{s}_m} H_v(\bar{s}_m)}{w_0 + \sum_{m=1}^M \frac{w_m}{s - \bar{s}_m}}. \quad (2)$$

Although (2) resembles the VF model representation, there are some differences: 1) both $N_v(s)$ and $D(s)$ share the same set of coefficients w_m and 2) the common set of starting poles \bar{s}_m are purely imaginary instead of complex-valued. The starting poles \bar{s}_m are equally spaced over the frequency range (Section II) and located on the positive part of the imaginary axis. Unfortunately, the causality of model (2) is not guaranteed.

B. Causal Model Representation

As to obtain a real-rational causal model ($R_v^*(s) = R_v(s^*)$), both $\{\bar{s}_m, H_v(\bar{s}_m)\}$ and $\{\bar{s}_m^*, H_v^*(\bar{s}_m^*)\}$ are interpolated and a linear combination of the basis functions is formed as shown in [1, App. A]. This modification leads to a transfer function with

$P = 2M$ complex conjugate, purely imaginary starting poles $\bar{s} = \{\bar{s}_m, \bar{s}_m^*\}_{m=1}^M$ having real coefficients c_p

$$R_v(s) = \frac{N_v(s)}{D(s)} = \frac{\sum_{p=1}^P c_p \Phi_{p,v}^N(s, \bar{s})}{c_0 + \sum_{p=1}^P c_p \Phi_p^D(s, \bar{s})}. \quad (3)$$

The basis functions $\Phi_{p,v}^N(s, \bar{s})$ and $\Phi_{p+1,v}^N(s, \bar{s})$ are defined for each complex conjugate pole-pair $\{\bar{s}_m, \bar{s}_m^*\}$ of $N_v(s)$ as

$$\Phi_{p,v}^N(s, \bar{s}) = \frac{H_v(\bar{s}_m)}{s - \bar{s}_m} + \frac{H_v^*(\bar{s}_m)}{s - \bar{s}_m^*} \quad (4)$$

$$\Phi_{p+1,v}^N(s, \bar{s}) = \frac{iH_v(\bar{s}_m)}{s - \bar{s}_m} - \frac{iH_v^*(\bar{s}_m)}{s - \bar{s}_m^*} \quad (5)$$

and similarly, $\Phi_p^D(s, \bar{s})$ and $\Phi_{p+1}^D(s, \bar{s})$ in $D(s)$ are defined for each complex conjugate pole-pair $\{\bar{s}_m, \bar{s}_m^*\}$ of $D(s)$ as

$$\Phi_p^D(s, \bar{s}) = \frac{1}{s - \bar{s}_m} + \frac{1}{s - \bar{s}_m^*} \quad (6)$$

$$\Phi_{p+1}^D(s, \bar{s}) = \frac{i}{s_k - \bar{s}_m} - \frac{i}{s_k - \bar{s}_m^*} \quad (7)$$

where $p = 2m - 1$ and $m = 1, \dots, M$. Hence, for any choice of real-valued coefficients $c_p \neq 0$, the transfer function model $R_v(s)$ in (3) will be causal and interpolates the data in \bar{S} .

IV. BARYCENTRIC VECTOR FITTING (B-VF)

The flexible choice of coefficients c_p allows one to impose extra properties to the model [8]. Here, coefficients c_p are calculated iteratively such that the model also approximates the data in S in a LS sense. A linear formulation is obtained by multiplying both sides of (3) with $D(s)$ and by minimizing

$$\arg \min_{c_p} \sum_{v=1}^V \sum_{k=1}^K |N_v(s_k) - H_v(s_k)D(s_k)|^2. \quad (8)$$

Hence, if the following definitions are introduced ($c_0 = 1$):

$$H'_v = [H_v(s_1) \dots H_v(s_K)] \quad (9)$$

$$H_v = \text{diag}([\Re(H'_v) \ \Im(H'_v)]) \quad (10)$$

$$\hat{1} = (2K \times 1) \text{ column vector of ones} \quad (11)$$

then the coefficients c_p of (3) can be calculated by solving the following overdetermined set of linear equations

$$\begin{bmatrix} X_v^N - H_1 X^D \\ X_v^N - H_2 X^D \\ \dots \\ X_v^N - H_V X^D \end{bmatrix} [C] = \begin{bmatrix} H_1 \hat{1} \\ H_2 \hat{1} \\ \dots \\ H_V \hat{1} \end{bmatrix} \quad (12)$$

where the matrices X_v^N and X_D are defined as follows:

$$X_v^N = \begin{bmatrix} \Re(X_v^{N'}) \\ \Im(X_v^{N'}) \end{bmatrix}, \quad X^D = \begin{bmatrix} \Re(X_1^{D'}) \\ \Im(X_1^{D'}) \end{bmatrix} \quad (13)$$

$$X_v^x = \begin{bmatrix} \Phi_{1,v}^x(s_1, \bar{s}) & \dots & \Phi_{P,v}^x(s_1, \bar{s}) \\ \dots & \dots & \dots \\ \Phi_{1,v}^x(s_K, \bar{s}) & \dots & \Phi_{P,v}^x(s_K, \bar{s}) \end{bmatrix}. \quad (14)$$

Here, x is a generic variable that denotes either N or D . Once the coefficients c_p in solution vector C are found, one can simplify (3) by cancelling out the starting poles \bar{s} . The relocated poles $\theta = \{\theta_1, \dots, \theta_P\}$ of the transfer function are then the zeros of $D(s)$. These zeros are calculated by solving an eigenvalue problem that is based on the minimal state space realization (A, B, C, D) of $D(s)$, as shown in [1, App. B]

$$\theta = \text{eig}(A - BD^{-1}C). \quad (15)$$

The estimated poles can iteratively be refined by applying the Sanathanan–Koerner (SK) iteration with explicit weighting [2]

$$\arg \min_{c_p^{(t)}} \sum_{k=1}^K \left| \frac{N_v^{(t)}(s_k) - H(s_k)D^{(t)}(s_k)}{D^{(t-1)}(s_k)} \right|^2. \quad (16)$$

So, instead of replacing the initial poles \bar{s} with the relocated poles θ , an inverse weighting function ($1/D^{(t-1)}(s)$) is applied to each row of the LS matrix and updated values for the coefficients $c_p^{(t)}$ are computed in each iteration step $t = 1, \dots, T$.

Once the accuracy of the model has converged, any unstable poles are flipped into the left half plane and the standard residue identification of VF is applied to compute a full LS solution using all data in $S \cup \bar{S}$, see [1] for details.

V. RELAXATION

In order to improve the convergence properties of the SK iteration [9], the high frequency asymptotic constraint on $D^{(t)}(s)/D^{(t-1)}(s)$ can be removed. This is achieved by making c_0 a free variable, and adding an additional relaxation condition to the LS equations as shown in [4]. The implementation of relaxation [4] is completely analogous, and can easily be combined with the new presented methodology.

VI. COMPLEXITY ANALYSIS

The computational complexity of the new algorithm can be analyzed by counting the number of floating point operations for the linear algebra problems that are involved. Following the same procedure as outlined in [10, Sect. III], it is shown that the number of flops for the QR-based VF algorithm is

$$\begin{aligned} \text{Fl}_{\text{VF}}(V, K, M, P, T) &= T \cdot [V \cdot \text{Fl}_{\text{QR}}(2(K+M), 2P) \\ &\quad + \text{Fl}_{\text{LSP}}(PV, P, 1) + \text{Fl}_{\text{eig}}(P) \\ &\quad + \text{Fl}_{\text{LSP}}(2(K+M), P, V) \\ &\quad \simeq 16VKP^2T. \end{aligned} \quad (17)$$

A similar analysis on the Barycentric VF algorithm shows that the number of flops can be further reduced by a constant factor

$$\begin{aligned} \text{Fl}_{\text{B-VF}}(V, K, M, P, T) &= T \cdot \text{Fl}_{\text{LSP}}(2KV, P, 1) + \text{Fl}_{\text{eig}}(P) \\ &\quad + \text{Fl}_{\text{LSP}}(2(K+M), P, V) \\ &\quad \simeq 4VKP^2T. \end{aligned} \quad (18)$$

For both algorithms, the pole-identification step is the most expensive part. The VF algorithm performs a QR decomposition for each single matrix element (see [3, (10)]), and then it solves a compound least-squares problem (see [3, (11)]). The B-VF algorithm can reduce this cost because it only needs to solve

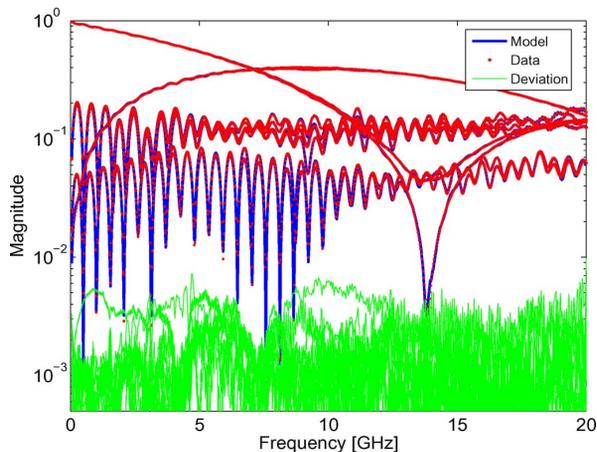


Fig. 1. Response of model calculated with B-RVF ($P = 100$).

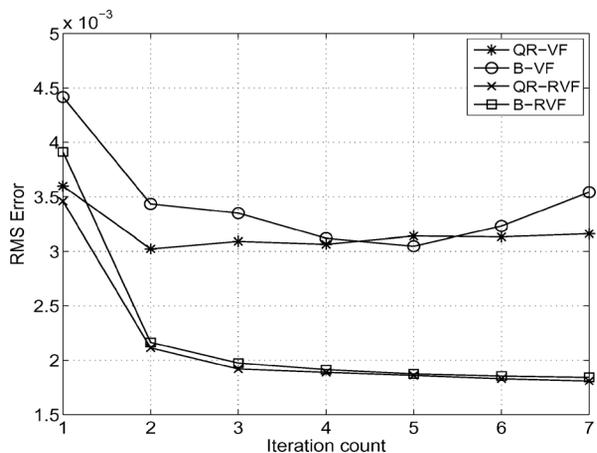


Fig. 2. Evolution of RMS error in terms of iteration count t .

TABLE I
OVERVIEW OF VECTOR FITTING ALGORITHMS

Name	Description
QR-VF	QR-based Vector Fitting
QR-RVF	QR-based Relaxed Vector Fitting
B-VF	Barycentric Vector Fitting (new)
B-RVF	Barycentric Relaxed Vector Fitting (new)

TABLE II
TOTAL COMPUTATION TIME FOR SINGLE ELEMENT FITTING ($V = 1$)

P	QR-VF	B-VF	Gain	QR-RVF	B-RVF	Gain
50	0.349 s	0.248 s	1.40×	0.348 s	0.246 s	1.41×
100	0.631 s	0.404 s	1.56×	0.620 s	0.397 s	1.56×
200	1.576 s	0.671 s	2.34×	1.450 s	0.697 s	2.08×
300	2.916 s	1.069 s	2.72×	3.164 s	1.037 s	3.05×
400	4.857 s	1.276 s	3.80×	4.661 s	1.333 s	3.49×
500	6.968 s	1.606 s	4.33×	7.224 s	1.738 s	4.15×

a least-squares problem (12), albeit with a matrix on the left-hand-side that is somewhat taller in size.

VII. NUMERICAL RESULT: COUPLED MICROSTRIP LINES

This example demonstrates the modeling of a 6 inch coupled microstrip lines on an FR4 PC board [11]. The noisy S-parameters of the 4×4 transfer matrix were measured at 799 frequencies over the range [0.05 – 20 GHz]. They are modeled using the different modeling approaches outlined in Table I. A macro-model with $P = 100$ poles was computed using B-RVF and the result is shown in Fig. 1. In Fig. 2, the evolution of the RMS error is compared in terms of iteration count, and it is found

TABLE III

TOTAL COMPUTATION TIME FOR MULTIPLE ELEMENT FITTING ($V = 16$)

P	QR-VF	B-VF	Gain	QR-RVF	B-RVF	Gain
50	2.76 s	1.38 s	1.99×	2.96 s	1.57 s	1.88×
100	5.76 s	3.58 s	1.61×	5.56 s	3.76 s	1.47×
200	12.96 s	7.94 s	1.63×	14.10 s	8.26 s	1.70×
300	24.63 s	12.77 s	1.92×	25.46 s	13.08 s	1.94×
400	40.26 s	17.71 s	2.27×	39.61 s	18.07 s	2.19×
500	60.18 s	22.36 s	2.69×	59.85 s	22.74 s	2.63×

TABLE IV

COMPUTATION TIME FOR POLE-IDENTIFICATION: $P = 400$. & $V = 16$

Algorithm	Pole Identification	Operations	Time
QR-(R)VF	QR decompositions (10) in [3]	272 Gflop	19.25 s
	LS matrix (11) in [3]	20 Gflop	1.42 s
B-(R)VF	LS matrix (12)	61 Gflop	7.95 s

that B-VF and B-RVF have a similar accuracy and convergence behavior as QR-VF and QR-RVF, respectively. In Tables II and III, also the total computation time is compared for the modeling of a single matrix element ($V = 1$) or all the matrix elements ($V = 16$). It turns out that the new barycentric algorithms are significantly faster, especially as the number of poles increases. Table IV shows that this speed-up is due to the faster pole identification step, as discussed in Section VI¹.

VIII. CONCLUSION

A fast approach for the macromodeling of frequency responses was presented. The method exploits the barycentric interpolation formula in order to speed-up the pole identification step. Numerical results show that the method delivers accurate results, even when the data is contaminated with noise. It improves the running time of QR-based VF algorithms, for the fitting of single/multiple elements of a transfer matrix.

REFERENCES

- [1] B. Gustavsen and A. Semlyen, "Rational approximation of frequency domain responses by vector fitting," *IEEE Trans. Power Del.*, vol. 14, no. 3, pp. 1052–1060, Jul. 1999.
- [2] D. Deschrijver, B. Gustavsen, and T. Dhaene, "Advancements in iterative methods for rational approximation in the frequency domain," *IEEE Trans. Power Del.*, vol. 22, no. 3, pp. 1633–1642, Jul. 2007.
- [3] D. Deschrijver, M. Mrozowski, T. Dhaene, and D. De Zutter, "Macromodeling of multiport systems using a fast implementation of the vector fitting method," *IEEE Microw. Wirel. Comp. Lett.*, vol. 18, no. 6, pp. 383–385, Jun. 2008.
- [4] B. Gustavsen, "Improving the pole relocating properties of vector fitting," *IEEE Trans. Power Del.*, vol. 21, no. 3, pp. 1587–1592, Jul. 2006.
- [5] *Public Domain Vectfit3.m Software Package, Vector Fitting Website*, [Online]. Available: <http://www.energy.sintef.no/Produkt/VECTFIT/>
- [6] A. Semlyen and B. Gustavsen, "Vector fitting by pole relocation for the state equation approximation of nonrational transfer matrices," *Circ., Syst. and Sign. Proc.*, vol. 19, no. 6, pp. 549–566, 2000.
- [7] C. Schneider and W. Werner, "Some new aspects of rational interpolation," *Math. Comput.*, vol. 47, no. 175, pp. 285–299, Jul. 1986.
- [8] L. Knockaert, "A simple and accurate algorithm for Barycentric rational interpolation," *IEEE Signal Process. Lett.*, vol. 15, pp. 154–157, 2008.
- [9] C. Sanathanan and J. Koerner, "Transfer function synthesis as a ratio of two complex polynomials," *IEEE Trans. Autom. Control*, vol. AC-8, no. 1, pp. 56–58, Jan. 1963.
- [10] A. China and S. Grivet-Talocia, "On the parallelization of vector fitting algorithm," *IEEE Trans. Comp. Packag. Manufact. Technol.*, vol. 1, no. 11, pp. 1761–1773, Nov. 2011.
- [11] W. Beyene and C. Yuan, "An accurate transient analysis of high-speed package interconnects using convolution technique," *Analog Integr. Circuits Sign. Processing*, vol. 35, no. 2–3, pp. 107–120, 2003.

¹All timing results are for $T = 10$ iterations in MATLAB on 64-bit operating system with Intel(R) Core(TM) i7-2760 QM CPU at 2.4 GHz and 8 GB RAM. Timing results QR-VF and QR-RVF are based on implementation [5]. The starting poles are always chosen in an optimal way as discussed in [1], [2]