

Macromodeling of Multiport Systems Using a Fast Implementation of the Vector Fitting Method

Dirk Deschrijver, Michal Mrozowski, *Fellow, IEEE*, Tom Dhaene, *Senior Member, IEEE*, and Daniel De Zutter, *Fellow, IEEE*

Abstract—Broadband macromodeling of large multiport systems by vector fitting can be time consuming and resource demanding when all elements of the system matrix share a common set of poles. This letter presents a robust approach which removes the sparsity of the block-structured least-squares equations by a direct application of the QR decomposition. A 60-port printed circuit board example illustrates that considerable savings in terms of computation time and memory requirements are obtained.

Index Terms—Broadband macromodeling, numerical techniques, system identification, vector fitting (VF).

I. INTRODUCTION

COMPACT rational macromodels, based on measurements or full-wave electromagnetic (EM) simulations, are very important for efficient time domain and frequency domain simulation of high-frequency/high-speed interconnection structures, components and systems. The vector fitting (VF) technique is a robust macromodeling tool that circumvents the ill-conditioning and unbalanced weighting problems which usually occur in a rational approximation problem [1], [2].

In the past few years, various enhancements like orthonormalization [3] and relaxation [4] have been proposed to improve the accuracy and convergence of this technique [5]. Unfortunately, VF still suffers from computationally inefficiency if the device under study has a relatively large number of ports, although a splitting strategy has been considered [6].

This letter introduces a new improvement, which performs the identification of multiport systems using a common set of poles in a significantly reduced amount of time. This goal is achieved by exploiting the sparsity of the least-squares (LS) equations that are solved during the pole-identification step. The presented method is based on the QR decomposition, which does not impose specific restrictions to the data or the modeling process. Numerical results illustrate that significant savings are obtained in terms of computation time and memory resources, while preserving accuracy of the macromodel.

Manuscript received November 23, 2007; revised February 6, 2008. This work was supported by the Polish Ministry of Science and Higher Education, the Science and Innovation Administration Flanders (Bilateral Scientific Cooperation project), and the Fund for Scientific Research Flanders (FWO-Vlaanderen).

D. Deschrijver, T. Dhaene, and D. De Zutter are with the Department of Information Technology, Ghent University-IBBT, Ghent 9000, Belgium, (e-mail: dirk.deschrijver@intec.ugent.be; tom.dhaene@intec.ugent.be; daniel.dezutter@intec.ugent.be)

M. Mrozowski is with the Department of Electronics, Technical University of Gdansk, Gdansk 80-952, Poland (e-mail: m.mrozowski@ieee.org).

Digital Object Identifier 10.1109/LMWC.2008.922585

II. VECTOR FITTING METHODOLOGY

VF is a robust macromodeling algorithm that calculates the rational function approximation $R_v(s)$ of a linear multiport structure, based on simulated or measured frequency responses $\{s_k, H_v(s_k)\}_{k=1}^K$

$$R_v(s) = \frac{(\sigma H)_v(s)}{\sigma(s)} = \frac{\sum_{p=1}^P c_{p,v} \Phi_p(s, a)}{\check{c}_0 + \sum_{p=1}^P \check{c}_p \Phi_p(s, a)}. \quad (1)$$

The objective of the macromodeling technique is to identify the unknown coefficients $c_{p,v}$ and \check{c}_p of the transfer function such that the least-squares distance between $R_v(s_k)$ and the data samples $(s_k, H_v(s_k))$ corresponding to each matrix element $v = 1, \dots, V$ is minimized over some predefined frequency range of interest $[s_1, \dots, s_K]$. The basis functions $\Phi_p(s, a)$ are chosen to be rational functions that are based on a prescribed set of poles $a = \{-a_1, \dots, -a_P\}$. These poles are selected according to a heuristical scheme [1].

A. Parameter Estimation

A direct calculation of the model coefficients in (1) leads to a non-linear identification problem that can be hard to solve using standard optimization techniques. It was shown in [1] that accurate approximations can be found by solving a linear approximation of the problem in an iterative procedure

$$(\sigma H)_v(s) = \sigma(s) H_v(s). \quad (2)$$

If the following definitions are introduced ($\check{c}_0 = 1$):

$$H'_v = [H_v(s_1) \dots H_v(s_K)] \quad (3)$$

$$H_v = \text{diag}([\Re(H'_v) \Im(H'_v)]) \quad (4)$$

$$\hat{1} = (2K \times 1) \text{ column vector of ones} \quad (5)$$

and the matrix X is defined as

$$X' = \begin{bmatrix} \Phi_1(s_1, a) & \dots & \Phi_P(s_1, a) \\ \dots & \dots & \dots \\ \Phi_1(s_K, a) & \dots & \Phi_P(s_K, a) \end{bmatrix} \quad (6)$$

$$X = \begin{bmatrix} \Re(X') \\ \Im(X') \end{bmatrix} \quad (7)$$

then the solution of (2) corresponds to solving the following overdetermined set of equations

$$\begin{bmatrix} X & 0 & 0 & 0 & -H_1 X \\ 0 & X & 0 & 0 & -H_2 X \\ 0 & 0 & \dots & 0 & \dots \\ 0 & 0 & 0 & X & -H_V X \end{bmatrix} \begin{bmatrix} C_1 \\ C_2 \\ \dots \\ C_V \\ \tilde{C} \end{bmatrix} = \begin{bmatrix} H_1 \hat{1} \\ H_2 \hat{1} \\ \dots \\ H_V \hat{1} \end{bmatrix} \quad (8)$$

where C_v is a vector that contains the residues $c_{p,v}$ of $(\sigma H)_v(s)$, and \tilde{C} is a vector that contains the residues \tilde{c}_p of $\sigma(s)$. Once the coefficients are solved, it is clear that (1) can be simplified by cancelling out the prescribed poles a . Then, the relocated poles $\theta = \{\theta_1, \dots, \theta_P\}$ of the transfer function are in fact the zeros of $\sigma(s)$. These zeros are easily calculated by solving an eigenvalue problem that is based on the minimal state space realization $(A_\sigma, B_\sigma, C_\sigma, D_\sigma)$ of $\sigma(s)$

$$\theta = \text{eig}(A_\sigma - B_\sigma D_\sigma^{-1} C_\sigma). \quad (9)$$

This procedure is repeated in an iterative fashion ($t = 0, \dots, T$), by replacing the prescribed poles a by the relocated poles θ until convergence. Once the final poles are found, the residues of the transfer function can be solved as a linear problem by setting $\sigma(s)$ in (1) equal to 1. The rational function approximation (1) can then easily be realized as a state-space model. Details about this procedure are well documented [1].

B. Relaxation

It is known that the iterative procedure of VF is equivalent to the Sanathanan-Koerner iteration [7], [8] using an implicit weighting scheme [5]. This iteration is known to have good convergence properties if the signal-to-noise ratios are sufficiently high. However, it was shown in [9] that the convergence may stall or diverge if the data samples are contaminated with simulation or measurement noise. An efficient solution to this problem is presented in [4], where the high frequency asymptotic constraint on $\sigma(s)$ is removed. This is achieved by making \tilde{c}_0 a free variable, and adding an additional relaxation condition to the LS equations. The implementation of relaxation [4] is completely analogous, and can easily be combined with the new presented methodology.

III. FAST FITTING

Even if the structure under study has a moderate amount of ports, the size of the corresponding LS matrix (8) may become prohibitively large. A closer investigation reveals that a lot of computational effort is wasted on the calculation of the residues C_v , which are discarded by the VF algorithm. It will be shown that an application of the QR decomposition to the single-element LS equations leads to a simplified set of equations which depend only on \tilde{C} . These equations are computed for each matrix element sequentially, and considered at once to identify a common pole set [10].

A. Pole Identification

First, a QR decomposition is applied to the LS equations, which correspond to the pole-identification of each single matrix element v

$$[X \quad -H_v X] = [Q_v] \begin{bmatrix} R_v^{11} & R_v^{12} \\ 0 & R_v^{22} \end{bmatrix}. \quad (10)$$

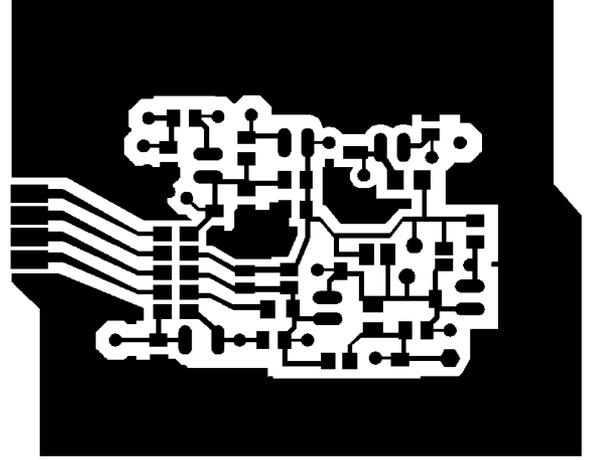


Fig. 1. Physical layout of the 60-port PCB board.

Combining the factorization for all matrix elements v yields a reduced set of equations, where the unknowns are only the shared coefficients \tilde{C} of $\sigma(s)$

$$\begin{bmatrix} R_1^{22} \\ \dots \\ R_V^{22} \end{bmatrix} [\tilde{C}] = \begin{bmatrix} Q_1^T H_1 \hat{1} \\ \dots \\ Q_V^T H_V \hat{1} \end{bmatrix}. \quad (11)$$

It is observed that the LS matrices on the left half side of (10) and (11) are both significantly smaller than the sparse matrix in (8), which leads to a significant reduction in terms of memory storage and computation time. In some cases, e.g., if the frequency response is resolved very densely, the computation time can be further reduced by applying the QR decomposition to the normal equations instead [5]. Although the normal equations are more compact, the conditioning of the problem may degrade and result in some loss of accuracy.

B. Residue Identification

Once these coefficients \tilde{C} of $\sigma(s)$ are computed, the common poles of the transfer function are found by solving an eigenvalue problem (9), and the corresponding residues of the macromodel can easily be calculated as in [1].

IV. NUMERICAL EXAMPLE: 60 PORT PCB BOARD

This section illustrates the CPU time and the overall memory requirements that are needed to compute an accurate macromodel for a 60-port printed circuit board (PCB) board. The board is simulated with Agilent's Momentum RF EM simulator over a frequency range from dc up to 5 GHz, at 101 equidistant frequency samples. The layout of the PCB board is shown in Fig. 1. In a practical application of the VF algorithm, one can take advantage of the fact that its scattering matrix is symmetric. This can easily be done by considering only the upper or the lower triangular part of the transfer matrix. Therefore a total amount of 1830 matrix elements needs to be modeled.

Based on the simulated data samples, a proper rational function approximation of the transfer matrix is calculated using 10 common poles. All the timing results are computed in a Matlab environment on a HP Notebook computer with an Intel Centrino processor (1.73 GHz clock frequency) and 1 GB of RAM. This

computer has a 32 b architecture, and matrix entries are stored as double precision floating point values.

A. Vector Fitting

Three implementations of VF are considered.

- A VF implementation which does not exploit the sparsity of the LS matrices (Standard VF).
- A VF implementation which makes use of sparse matrix storage/operations in Matlab (Sparse VF).
- A VF implementation which uses the new Fast Fitting approach as described in this letter (Fast VF).

To improve the accuracy and convergence speed of the results, relaxation (see Section II-B) has also been enabled.

B. Computational Results

It is found that Standard VF (available on [11]) is unable to perform the computations because the storage of matrix (8) consumes a lot of memory resources. It can be shown that at least 54 GB of RAM memory is required, which is computationally infeasible on most desktop computers [12].

Fortunately, the Matlab environment offers some sparse matrix data structures, which represents matrices in an amount of space that is proportional to the number of non-zero entries. The Sparse VF implementation (available on [11]) exploits the sparsity of the LS matrix (8), and is able to reduce the storage requirements considerably to approximately 88 MB of RAM memory, which is acceptable. An accurate macromodel can be calculated in 1 h and 22 min, using three iterations. It turns out that the maximum absolute error over all 1830 matrix elements is bounded by -62 dB, which is quite satisfactory.

The fast fitting approach using QR, as described in Section III of this letter (Fast VF), is able to compute a macromodel with the same accuracy in only 44 s using the same number of iterations. The amount of memory resources that is required to store (10) and (11) is at least 32 KB and 1.46 MB of RAM memory, respectively. Since (11) has a block-triangular structure, it can be considered as a sparse matrix. Therefore, the storage of this matrix is further reduced to 805 KB of RAM memory. A similar accuracy was obtained by solving the normal equations, which takes only 25 s.

These results indicate that Fast VF is the preferable approach, as it significantly reduces the overall macromodeling time and memory consumption, while retaining the accuracy of the results. Fig. 2 shows a subset of the matrix elements, which confirms that an excellent agreement is observed between the 101 frequency samples (marked with dots), and the response of the macromodel (solid line).

V. CONCLUSION

Rational fitting of broadband frequency-domain responses using VF becomes computationally expensive and resource demanding if a multiport system is identified using a common set of transfer function poles. A very efficient implementation

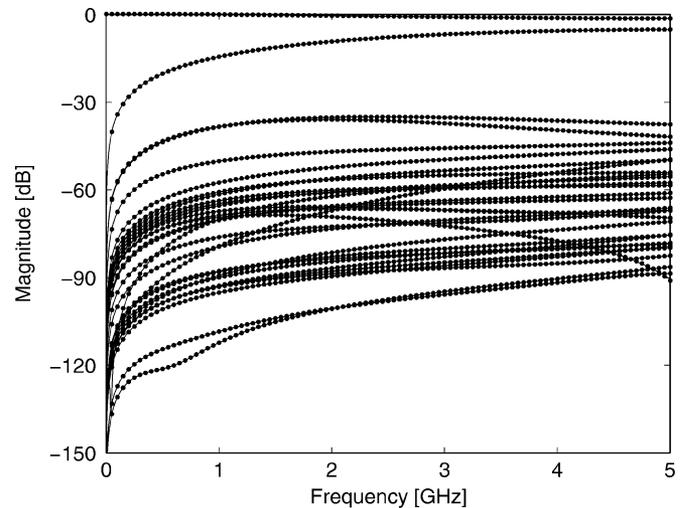


Fig. 2. Subset of scattering parameters : model (solid line), data (dots).

of the VF technique is presented which exploits the sparsity of the corresponding LS equations using a QR decomposition. This approach significantly reduces the overall computation time and memory consumption.

REFERENCES

- [1] 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, Aug. 1999.
- [2] R. Gao, Y. S. Mekonnen, W. T. Beyene, and J. E. Schutt-Ainé, "Black-box modeling of passive systems by rational function approximation," *IEEE Trans. Adv. Packag.*, vol. 28, no. 2, pp. 209–215, May 2005.
- [3] D. Deschrijver, B. Haegeman, and T. Dhaene, "Orthonormal vector fitting: A robust macromodeling tool for rational approximation of frequency domain responses," *IEEE Trans. Adv. Packag.*, vol. 30, no. 2, pp. 216–225, May 2007.
- [4] B. Gustavsen, "Improving the pole relocating properties of vector fitting," *IEEE Trans. Power Delivery*, vol. 21, no. 3, pp. 1587–1592, Aug. 2006.
- [5] D. Deschrijver, B. Gustavsen, and T. Dhaene, "Advancements in iterative methods for rational approximation in the frequency domain," *IEEE Trans. Power Delivery*, vol. 22, no. 3, pp. 1633–1642, Aug. 2007.
- [6] S. Grivet Talocia and A. Ubollu, "On the generation of large passive macromodels for complex interconnect structures," *IEEE Trans. Adv. Packag.*, vol. 29, no. 1, pp. 39–54, Feb. 2006.
- [7] C. K. Sanathanan and J. Koerner, "Transfer function synthesis as a ratio of two complex polynomials," *IEEE Trans. Automatic Control*, vol. AC-8, no. 1, pp. 56–58, Jan. 1963.
- [8] W. Hendrickx, D. Deschrijver, and T. Dhaene, "Some remarks on the vector fitting iteration," in *Proc. Progress Ind. Math. Ind. (ECMI'04)*, 2006, vol. 8, pp. 134–138.
- [9] S. Grivet-Talocia and M. Bandinu, "Improving the convergence of vector fitting for equivalent circuit extraction from noisy frequency responses," *IEEE Trans. Electromagn. Compat.*, vol. 48, no. 1, pp. 104–120, Feb. 2006.
- [10] D. Traina, G. Macchiarella, and T. K. Sarkar, "Robust formulations of the cauchy method suitable for microwave duplexer modeling," *IEEE Trans. Microw. Theory Tech.*, vol. 55, no. 5, pp. 974–982, May 2007.
- [11] Vector Fitting [Online]. Available: <http://www.energy.sintef.no/Produkt/VECTFIT/index.asp>
- [12] The Mathworks, "Avoiding Out of Memory Errors," Tech. Note 1107 [Online]. Available: <http://www.mathworks.com/support/tech-notes/1100/1107.html>