The Application of Preconditioned Jacobi-Davidson Methods in Pole-zero Analysis

J. Rommes[1], C.W. Bomhof[2], H.A. van der Vorst[1], and E.J.W. ter Maten[3]

[1] Utrecht University, The Netherlands, jrommes@cs.uu.nl
[2] Plaxis BV, Delft, The Netherlands,
[3] Philips Research Laboratories, Eindhoven University of Technology, The Netherlands

Abstract. The application of Jacobi-Davidson style methods in electric circuit simulation will be discussed in comparison with other iterative methods (Arnoldi) and direct methods (QR, QZ). Numerical results show that the use of a preconditioner to solve the correction equation may improve the Jacobi-Davidson process, but may also cause computational and stability problems when solving the correction equation. Furthermore, some techniques to improve the stability and accuracy of the process will be given.

1 Introduction

Pole-zero analysis is used in electrical engineering to analyse the stability of electric circuits [10,11,14]. For example, if a circuit is designed to be an oscillator, pole-zero analysis is one of the ways to verify that the circuit indeed oscillates. Another application is the verification of reduced order models over a wide frequency range [9]. Because the complexity of the circuits designed nowadays grows, together with the frequency range of interest, there is need for faster algorithms, not neglecting the accuracy. In this paper, Sect. 2 introduces the pole-zero problem. Section 3 describes the Jacobi-Davidson style methods as alternatives of conventional methods for the pole-zero problem. In Sect. 4, some typical numerical problems and techniques are discussed. In Sect. 5 the methods will be compared by numerical results, concluding with some future research topics.

2 Pole-zero Analysis in Circuit Simulation

The Kirchhoff Current Law and the Kirchhoff Voltage Law describe the topology of an electric circuit. Together with the Branch Constitutive Relations, which reflect the electrical properties of the branches, the two Kirchhoff Laws result in a system of differential algebraic equations [10]:

$$\frac{\mathrm{d}}{\mathrm{d}t}\boldsymbol{q}(t,\boldsymbol{x}) + \boldsymbol{j}(t,\boldsymbol{x}) = 0, \tag{1}$$

where $\boldsymbol{x} \in \mathbb{R}^n$ contains the circuit state and $\boldsymbol{q}, \boldsymbol{j} : \mathbb{R} \times \mathbb{R}^n \to \mathbb{R}$ are functions representing the reactive and resistive behaviour, respectively. The way (1) is solved depends on the kind of analysis (DC-analysis, AC-analysis, transient analysis, pole-zero analysis). In every analysis, the capacitance matrix $C \in \mathbb{R}^{n \times n}$ and the conductance matrix $G \in \mathbb{R}^{n \times n}$ appear:

$$C(t, \boldsymbol{x}) = \frac{\partial \boldsymbol{q}(t, \boldsymbol{x})}{\partial \boldsymbol{x}}, \qquad G(t, \boldsymbol{x}) = \frac{\partial \boldsymbol{j}(t, \boldsymbol{x})}{\partial \boldsymbol{x}}.$$

Both matrices are real, non-symmetric and sparse.

Starting from a linearization round the DC-operating point, the time-domain formulation is as follows:

$$\begin{cases} C\frac{\mathrm{d}\boldsymbol{x}(t)}{\mathrm{d}t} + G\boldsymbol{x}(t) = \boldsymbol{e}(t) \\ \boldsymbol{x}(0) = 0, \end{cases} \tag{2}$$

where $\boldsymbol{e}(t)$ models the excitation. Because not all properties can be computed in the time domain, the problem is transformed to the frequency domain by applying a Laplace transform:

$$(sC + G)\mathcal{X}(s) = \mathcal{E}(s), \tag{3}$$

where $\mathcal{X}, \mathcal{E}$ are the Laplace-transforms of the variables $\boldsymbol{x}, \boldsymbol{e}$ and s is the variable in the frequency domain. The response of the circuit to a variation of the excitation is given by the transfer function

$$\mathcal{H}(s) = (sC + G)^{-1}. \tag{4}$$

The elementary response of circuit variable $\mathcal{X}_o$ to excitation $\mathcal{E}_i$ is given by

$$\mathcal{H}_{oi}(s) = \boldsymbol{e}_o^T (sC + G)^{-1} \boldsymbol{e}_i. \tag{5}$$

The poles are the values $p_k \in \mathbb{C}$ that satisfy $\det(p_k C + G) = 0$, hence $(G + p_k C)\boldsymbol{x} = 0$ for some $\boldsymbol{x} \neq 0$, which leads to a generalised eigenproblem ($\lambda = -p_k$):

$$G\boldsymbol{x} = \lambda C\boldsymbol{x}, \qquad \boldsymbol{x} \neq 0. \tag{6}$$

Because the problem of computing the zeroes is similar to the problem of computing the poles, the rest of this paper will consider the problem of computing the poles.

Especially for large circuits ($n > 10^4$), robust, iterative methods for the generalised eigenvalue problem (6) with sufficient accuracy and acceptable computational costs are needed. Furthermore, all right half-plane poles and no false right half-plane poles are desired. The dominant poles and zeroes must be accurate enough to produce correct Bode-plots for the frequency range of interest.

Two kinds of pole-zero methods are known in literature [10]: combined and separate pole-zero computation. This paper focuses on separate pole-zero computation.

3 Jacobi-Davidson Style Methods

Because of its accuracy and robustness, the full-space QR-method (and the QZ-method to a less degree) is a popular choice as solver for the eigenproblem (6). However, the total costs of $O(n^3)$ and the necessity of an LU-decomposition, which destroys the sparsity of G and causes numerical inaccuracies and maybe instabilities, become unacceptable for larger problems. Iterative methods like the implicitly restarted Arnoldi method also need the LU-decomposition and are designed to compute only a few ($m \ll n$) eigenvalues [11].

The Jacobi-Davidson method [13], on the other hand, is designed to converge fast to a few selected eigenvalues. Based on the Jacobi-Davidson method, the JDQR-method [8], which computes a partial Schur form, and the JDQZ-method [8], which computes a partial generalised Schur form, are developed. Without going into much detail, the basic idea behind the Jacobi-Davidson methods is as follows. For the problem $A\boldsymbol{x} = \lambda\boldsymbol{x}$, given the eigenpair approximation $(\theta_k, \boldsymbol{u}_k)$:

- Search a correction $\boldsymbol{v} \in \boldsymbol{u}_k^{\perp}$ for $\boldsymbol{u}_k$ such that

$$A(\boldsymbol{u}_k + \boldsymbol{v}) = \lambda(\boldsymbol{u}_k + \boldsymbol{v}).$$

- Solve $\boldsymbol{v}$ from the *correction equation*, with $\boldsymbol{r}_k = A\boldsymbol{u}_k - \theta_k\boldsymbol{u}_k$:

$$(I - \boldsymbol{u}_k\boldsymbol{u}_k^*)(A - \theta_k I)(I - \boldsymbol{u}_k\boldsymbol{u}_k^*)\boldsymbol{v} = -\boldsymbol{r}_k.$$

- Orthogonally expand the current basis V with $\boldsymbol{v}$.

The Ritz-vector $\boldsymbol{u}_k = V\boldsymbol{s}$ is obtained by applying a full-space method, for instance the QR-method, to the projected matrix V^*AV, resulting in the eigenpair $(\theta_k, \boldsymbol{s})$. The Jacobi-Davidson method satisfies a Ritz-Galerkin condition [13].

The correction equation needs more attention. For the JDQR-method, it is

$$(I - QQ^*)(A - \theta_k I)(I - QQ^*)\boldsymbol{v} = -\boldsymbol{r}_k, \tag{7}$$

where $Q \in \mathbb{R}^{n\times k}$. If the correction equation is solved exactly, the convergence of the Jacobi-Davidson method is quadratic [13]. Besides solving the correction equation exactly, one can use linear iterative methods, like GMRES, with or without preconditioning. Because exact solvers are often not feasible in practice, the focus is on iterative methods with preconditioning. Using a preconditioner, however, is not as easy as it seems. Consider a preconditioner $K \approx A - \theta_k I$. There are three major issues. Firstly, the preconditioner is projected afterwards ($\tilde{K} = (I - QQ^*)K(I - QQ^*)$). Secondly, $A - \theta_k I$ becomes more and more ill conditioned as the approximations θ_k become near the eigenvalue λ. Thirdly, this θ_k changes every iteration, and so does $A - \theta_k I$.

4 Numerical Problems and Techniques

In [5], a technique is described to reduce the problem size of the ordinary eigenproblem. The idea is to remove the columns (and corresponding rows) from $G^{-1}C$ which are equal to zero, as well as the rows (and corresponding columns) equal to zero. This is justified because rows and columns equal to zero have corresponding eigenvalues of value zero, and removing these rows and columns does not influence the other eigenvalues. Because the product $G^{-1}C$ is not available explicitly, for computational reasons, the k rows and columns to keep are administrated in a matrix $S = [e_{i_1}, \ldots, e_{i_k}]$, where e_j is the j-th unit vector of length n. The reduced matrix is then defined by $S^T G^{-1} C S$.

Reduction of the problem in this way has a number of advantages. Firstly, Jordan blocks may be removed, thereby improving the stability and accuracy of the computations. Secondly, the computational costs will be reduced. Table 1 shows the degree of reduction for some example problems.

Table 1. The size, reduced size and degree of reduction for some example problems. The data is extracted from [5] and [10].

Problem	Size	Size (reduced)	Reduction
pz_09	504	365	28%
pz_28	177	74	58%
pz_36_osc	120	86	28%
pz_agc_crt	114	96	16%
jr_1	815	681	16%

Note, however, that the spectral properties of the reduced problem do not differ from the spectral properties of the original problem. As a consequence, the speed of convergence of the eigenmethod used will not be improved significantly. This technique is not applicable directly to the generalised eigenproblem.

Considering the computation of the preconditioner (for the operator in the correction equation), one has to cope with two problems, i.e. the near singularity of the operator to precondition and the continuous change of the operator. These two problems appear in both Jacobi-Davidson QR and Jacobi-Davidson QZ. For computational reasons, a new preconditioner should be computed at most once for each new eigenvalue target. This is possible in practice, because the search space and corresponding Ritz-values in general contain good approximations for new eigenvalues. Some additional advantage can be gained by only recomputing the preconditioner for changes in θ larger than a certain threshold. Singularity problems can be dealt with by replacing zero diagonal elements by a small threshold value, a common technique

for incomplete LU-decompositions. The costs and fill-in can be controlled by using a drop-tolerance for non-diagonal elements, resulting in ILUT [12] decompositions.

5 Numerical Results and Conclusions

The data for the test problems was generated by the in-house analog electric circuit simulator Pstar of Philips Research [14]. Both full-space and iterative methods have been used to solve the ordinary and generalised eigenproblem. A small selection of the results presented in [10] has been made to identify the problems which are typical for the different approaches. Implementations of the Jacobi-Davidson methods are based on the algorithms in [4] and are available on the Internet: refer to [1] for JDQR and to [2] for JDQZ. Experiments have been done in Matlab 5.3 [3].

The transformation of the generalised eigenproblem to the ordinary eigenproblem may introduce inaccuracies, as has been mentioned before. Bode-plot (**a**) in Fig. 1 shows an example of this. Before computing the eigenvalues, the problem has been reduced from size 30×30 to 12×12 with the method described in [5]. The solution computed by QR differs significantly on two points from the exact solution, which is computed by using (5) for several frequencies s. The two notches are caused by non-cancelling poles and zeroes, which do cancel in the original problem. It is conceivable that this is caused by the inversion of G. The iterative methods Arnoldi and JDQR suffer even more from inaccuracies. Bode-plot (**b**) in Fig. 1 shows the computed solutions for the generalised eigenproblem. In this case, the QZ-method nearly resembles the exact solution, while the iterative schemes still suffer from inaccuracies. The fact that QZ performs better than QR, while both methods in theory compute the same eigenvalues, strengthens the argument that the inversion of G introduces critical inaccuracies. A general remark can be made

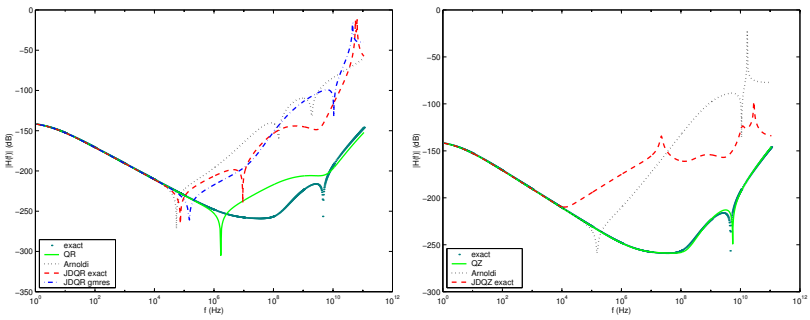

Fig. 1. (**a**) Bode-plot computed from the ordinary eigenproblem; (**b**) Bode-plot computed from the generalised eigenproblem.

about the interpretation of Bode-plots. It is not clear how accurate the original data of the circuit is. As a consequence, one may argue that the resulting Bode-plots are only representative up to a certain frequency, depending on the accuracy of the original data. For applications in the RF-area, this issue and the accuracy of eigenvalues near zero play even a more important role.

Using preconditioners when solving the correction equation of the JDQR method does indeed improve the speed of convergence, as can be seen in Fig. 2, where graph (**a**) shows the convergence history when using GMRES as solver, and graph (**b**) when using GMRES with an ILUT preconditioner ($t = 10^{-8}$). The quality of the improvement strongly depends on the accuracy

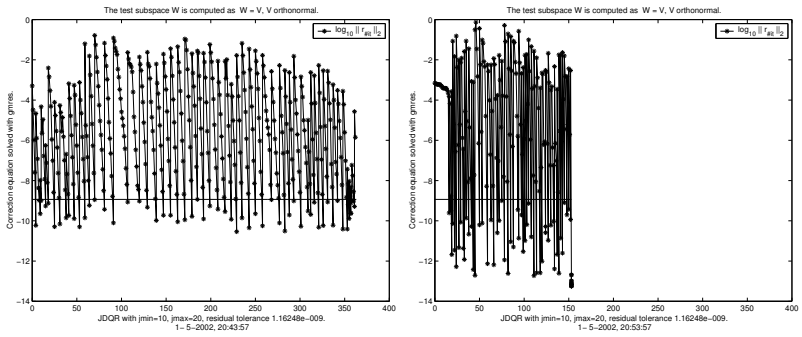

Fig. 2. (**a**) Convergence history for JDQR with GMRES; (**b**) Convergence history for JDQR with ILUT-preconditioned GMRES ($t = 10^{-8}$). A convergence history plots the residual against the Jacobi-Davidson iteration number; each drop below the tolerance means an accepted eigenvalue.

of the preconditioner. When using an ILUT preconditioner, a drop-tolerance of maximal $t = 10^{-6}$ is acceptable. This shows also one of the difficulties: the preconditioner has to be rather accurate, and in the case of ILU based preconditioners this means in general high costs. Apart from that, the ILU based preconditioners experience problems for singular matrices, and the matrix $A-\theta_k I$ becomes more and more singular. This last problem has appeared to be more severe for the JDQZ method. The fact that the preconditioner is projected afterwards has not a significant influence on the quality.

The reduction technique for the ordinary eigenproblem does lead to lower computational costs for both the direct and iterative methods. However, the gain depends on the degree of reduction, and is more pronounced for the iterative methods, because of the dominating matrix-vector products. The computational gained varied from 15% for the QR-method to 30% for the iterative methods, with top gains of 50%. Also, the condition number of the problem is improved, and if Jordan blocks are removed, the computed eigenvalues are more accurate. The number of iterations needed to computed

the eigenvalues is not decreased, as expected. For more information and result of this reduction technique, refer to [5,11].

The observations launch ideas for future work. One can think of efficient updates for preconditioners [6], model reduction techniques and combinations of Jacobi-Davidson with other iterative methods like Arnoldi or combined pole-zero methods, such as Padé via Lanczos [7].

References

1. http://www.math.uu.nl/people/sleijpen/JD_software/JDQR.html.
2. http://www.math.uu.nl/people/sleijpen/JD_software/JDQZ.html.
3. http://www.mathworks.com.
4. Bai, Z., Demmel, J., Dongarra, J., Ruhe, A., and van der Vorst, H., Eds. *Templates for the Solution of Algebraic Eigenvalue Problems: a Practical Guide.* SIAM, 2000.
5. Bomhof, C. Jacobi-Davidson methods for eigenvalue problems in pole zero analysis. Nat.Lab. Unclassified Report 012/97, Philips Electronics NV, 1997.
6. Bomhof, C. *Iterative and parallel methods for linear systems, with applications in circuit simulation.* PhD thesis, Utrecht University, 2001.
7. Feldmann, P., and Freund, R. W. Efficient linear circuit analysis by Padé approximation via the Lanczos process. *IEEE Trans. CAD 14* (1995), 639–649.
8. Fokkema, D. R., Sleijpen, G. L., and van der Vorst, H. A. Jacobi-Davidson style QR and QZ algorithms for the reduction of matrix pencils. *SIAM J. Sc. Comp. 20*, 1 (1998), 94–125.
9. Heres, P. J., and Schilders, W. H. Reduced order modelling of RLC-networks using an SVD-Laguerre based method. In *SCEE 2002 Conference Proceedings* (2002).
10. Rommes, J. Jacobi-Davidson methods and preconditioning with applications in pole-zero analysis. Master's thesis, Utrecht University, 2002.
11. Rommes, J., van der Vorst, H. A., and ter Maten, E. J. W. Jacobi-Davidson Methods and Preconditioning with Applications in Pole-zero Analysis. In *Progress in Industrial Mathematics at ECMI 2002* (2002).
12. Saad, Y. *Iterative methods for sparse linear systems.* PWS Publishing Company, 1996.
13. Sleijpen, G. L., and van der Vorst, H. A. A Jacobi-Davidson Iteration Method for Linear Eigenvalue Problems. *SIAM Review 42*, 2 (2000), 267–293.
14. ter Maten, E. J. W. Numerical methods for frequency domain analysis of electronic circuits. *Surv. Maths. Ind.*, 8 (1998), 171–185.