A Parallelizable GMRES-type Method for *p*-cyclic Matrices, with Applications in Circuit Simulation

Wim Bomhof[1,2]* and Henk A. van der Vorst[1]**

[1] Mathematical Institute, University of Utrecht, Budapestlaan 6, 3584 CD, Utrecht, The Netherlands
[2] Philips Research Laboratories, WAY41, Prof. Holstlaan 4, 5656 AA, Eindhoven, The Netherlands

Abstract. In this paper we propose a GMRES-type method for the solution of linear systems with a p-cyclic coefficient matrix. These p-cyclic matrices arise in the periodic steady state simulation of circuits, assuming that the DAE is discretized in the time domain. The method has similarities with existing GMRES approaches for p-cyclic matrices, but in contrast to these methods the method is efficiently parallelizable, even if the p-cyclic matrix has a small block size. However, the serial costs of the method may be somewhat higher. Numerical experiments demonstrate the effectiveness of the method.

1 Introduction

Periodic differential algebraic equations of the form

$$\begin{aligned} \tfrac{d}{dt}q(t,x(t)) + j(t,x(t)) = 0 \in \mathbb{R}^n \ , \\ x(0) = x(T) \ , \end{aligned} \tag{1}$$

arise in the periodic steady state analysis of a circuit. The sources of the circuit are included in function $j(t,x(t))$. We assume that all explicitly time-dependent coefficients and sources are periodic with period T. Transient analysis of $\frac{d}{dt}q(t,x(t)) + j(t,x(t)) = 0$ for $t \to \infty$ often converges very slowly to the periodic solution of (1). This leads to an excessive amount of CPU-time for the simulation process. Therefore, it may be more efficient to solve the periodic problem (1) with a method that exploits the periodic structure, for instance, the multiple shooting method or the finite difference method, see [1,5]. We will briefly describe the finite difference method. Time discretization of (1) at the discrete time points $t_0 = 0 < t_1 < \ldots < t_{M-1} < t_M = T$ leads to a system of nonlinear algebraic equations with Mn unknowns: $x_i = x(t_i),\ i = 1,\ldots,M$. Applying Newton's method to this nonlinear system gives a linear

* e-mail: bomhof@math.uu.nl
** e-mail: vorst@math.uu.nl

system of the form

$$\begin{bmatrix} F_1 & 0 & 0 & -E_M \\ -E_1 & F_2 & 0 & 0 \\ 0 & \ddots & \ddots & 0 \\ 0 & 0 & -E_{M-1} & F_M \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ \vdots \\ y_M \end{bmatrix} = \begin{bmatrix} c_1 \\ \vdots \\ \vdots \\ c_M \end{bmatrix} \quad . \tag{2}$$

For a more detailed derivation of (2) we refer to [5]. Note that both the multiple shooting method and the finite difference method lead to a linear system of the form (2).

The E_i and F_i in (2) are sparse n by n matrices. The matrix in (2) is called an M-cyclic matrix. We will assume that the diagonal blocks F_i are nonsingular; the off-diagonal blocks E_i may be singular. If one y_i is known, the others can be computed by the forward recursion

$$y_j = \begin{cases} {F_j}^{-1}(c_j + E_{j-1}y_{j-1}) & \text{if } j = i+1, \ldots, M \ , \\ {F_j}^{-1}(c_j + E_M y_M) & \text{if } j = 1 \ , \\ {F_j}^{-1}(c_j + E_{j-1}y_{j-1}) & \text{if } j = 2, \ldots, i-1 \ . \end{cases} \tag{3}$$

Block diagonal scaling of (2) with $M = \text{diag}({F_1}^{-1}, \ldots, {F_M}^{-1})$ leads to a linear system that is easier to handle:

$$\begin{bmatrix} I & 0 & 0 & -C_1 \\ -C_2 & I & 0 & 0 \\ 0 & \ddots & \ddots & 0 \\ 0 & 0 & -C_M & I \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ \vdots \\ y_M \end{bmatrix} = \begin{bmatrix} c'_1 \\ \vdots \\ \vdots \\ c'_M \end{bmatrix} \quad , \tag{4}$$

with $c'_i = {F_i}^{-1} c_i$ and

$$C_i = \begin{cases} {F_i}^{-1} E_M & \text{if } i = 1 \ , \\ {F_i}^{-1} E_{i-1} & \text{if } i = 2, \ldots, M \ . \end{cases} \tag{5}$$

The M-cyclic linear system (4) can be reduced by block Gaussian elimination to a p-cyclic one, with $p \leq M$. In order to eliminate blocks, we define a partition of the integers $1, \ldots, M$:

$$(1, 2, \ldots, M) = (\underline{q}_1, \ldots, \overline{q}_1, \ldots, \ldots, \underline{q}_p, \ldots, \overline{q}_p) \ . \tag{6}$$

A reduced system is obtained by eliminating the unknowns $y_{\underline{q}_i}, \ldots, y_{\overline{q}_i - 1}$, $i = 1, \ldots, p$ from (4). Hence, the unknowns $y_{\overline{q}_1}, \ y_{\overline{q}_2}, \ \ldots, \ y_{\overline{q}_p}$ are not eliminated. The reduced system is

$$Ax = b \ , \quad \text{with} \quad A = \begin{cases} I - B_1 \ , & \text{if } p = 1 \ , \\ \begin{bmatrix} I & 0 & 0 & -B_1 \\ -B_2 & I & 0 & 0 \\ 0 & \ddots & \ddots & 0 \\ 0 & 0 & -B_p & I \end{bmatrix} , & \text{if } p \geq 2 \ , \end{cases} \tag{7}$$

and

$$B_i = C_{\overline{q}_i} \cdot \ldots \cdot C_{\underline{q}_i},\ i = 1, \ldots, p \ , \tag{8}$$

$x = \left[x_1^T, \ldots, x_p^T\right]^T$, $b = \left[b_1^T, \ldots, b_p^T\right]^T$. The vectors b_i are defined by the following recursion:

$$\begin{aligned} & w_{\underline{q}_i} = c'_{\underline{q}_i} \ , \\ & w_k = c'_k + C_k w_{k-1},\ k = \underline{q}_i + 1, \ldots, \overline{q}_i \ , \\ & b_i = w_{\overline{q}_i} \ . \end{aligned} \tag{9}$$

The x_i and y_j are related via $y_{\overline{q}_i} = x_i$, for $i = 1, \ldots, p$. The other y_j, $j = 1, \ldots, M$, can be computed by the forward recursion (3), after (7) has been solved.

Block Gaussian elimination of $x_1, \ldots, x_{p-1}$ in (7) gives a linear system

$$(I - B_p \cdot \ldots \cdot B_1) x_p = \hat{b}_p \ , \tag{10}$$

where $\hat{b}_p$ is defined by the recurrence relation

$$\begin{aligned} & \hat{b}_1 = b_1 \ , \\ & \hat{b}_j = b_j + B_j \hat{b}_{j-1} \ \text{ for } j = 2, \ldots, p \ . \end{aligned}$$

The linear system (10) is equal to the linear system obtained by block Gaussian elimination of $y_1, \ldots, y_{M-1}$ in (4), this is the case $p = 1$ in (7). LU factorization of $I - B_p \cdot \ldots \cdot B_1$ in (10) leads to a direct method for solving (2). This method requires $2nM(\mathrm{nnz}(E_i) + \mathrm{nnz}(L_i) + \mathrm{nnz}(U_i))$ flops for explicitly forming $I - B_p \cdot \ldots \cdot B_1$ and $2/3n^3$ flops for the LU factorization of $I - B_p \cdot \ldots \cdot B_1$. Here $\mathrm{nnz}(\,\cdot\,)$ is the number of nonzeros of a matrix and $L_i U_i$ is the LU factorization of F_i. The other costs are relatively minor. We refer to [1] for other direct methods for (2).

A Krylov subspace method may be attractive for solving (10), because these methods do not require to form $I - B_p \cdot \ldots \cdot B_1$ explicitly. For example, GMRES [4] only needs matrix vector products $(I - B_p \cdot \ldots \cdot B_1)v$. Here B_i is defined by (5) and (8), and sparse LU decompositions of the diagonal blocks F_i with partial pivoting are used for F_i^{-1}. The costs of m matrix vector products are $2mM(\mathrm{nnz}(E_i) + \mathrm{nnz}(L_i) + \mathrm{nnz}(U_i))$ flops and the other costs are relatively minor. Usually $m \ll n$, so then there is a gain compared with the direct method.

Instead of applying GMRES to (10), it can be applied to (7). Then each matrix vector product Av requires the matrix vector products $B_i v_i, i = 1, \ldots, p$, which are independent of each other. So, the computation of Av is efficiently parallelizable. Note that the matrix vector product $(I - B_p \cdot \ldots \cdot B_1)v$ is hard to parallelize because of its serial nature. In practice E_i and F_i are often too small for parallelization on a matrix level. Unfortunately, GMRES converges usually much better for (10) than for (7), and even in a parallel environment there is no gain for (7), if $p \neq 1$.

In Sect. 2 we will introduce an efficiently parallelizable iterative method for solving the p-cyclic linear system (7). We will not discuss all the details of the method, a more comprehensive and precise description of the method will appear later in [2]. The method is not restricted to periodic steady state simulation of circuits, because linear systems of the form (2) may arise in the simulation of other periodic phenomena as well, for example, in periodic AC analysis of RF circuits [6].

Independently and simultaneously with our work, Dekker has developed the P-GMRES method [3]. For the special case $p = 2$ our method is equivalent with this P-GMRES method. For $p = 1$ our method is equivalent with the standard GMRES method [4].

Telichevesky, et al., [5] propose to solve (2) with preconditioned GMRES applied to (2). The preconditioner they use is the block lower triangular part of the matrix in (2). In this approach the matrix B_1 of the reduced system (7), with $p = 1$, arises as a block matrix on the diagonal of the (block upper triangular) preconditioned linear system. Therefore, solving (2) by applying our method (or standard GMRES) to the reduced system (7), with $p = 1$, is approximately as efficient as the approach of [5]. Similar to the $p = 1$ case, this method is hard to parallelize because of the serial nature of the matrix-vector recursions.

In Sect. 3 we present numerical experiments for solving linear systems of the form (2) arising from periodic steady state simulation of circuits.

Superscripts are used frequently in this paper. We will use the following convention for it: X^{-1} denotes the inverse of a nonsingular matrix X. The i in X^i is an index number if X is a vector or matrix and $i \geq 0$. Throughout this paper we use $\| \cdot \|$ for the 2-norm of a vector or a matrix.

2 *p*-cyclic GMRES

The GMRES method applied to (7) finds an approximate solution x^m in the Krylov subspace

$$\mathcal{K}^m = \text{span}(b, Ab, A^2, \ldots, A^{m-1}b) \ , \tag{11}$$

with $x^m = \underset{x^m \in \mathcal{K}^m}{\text{argmin}} \|b - Ax^m\|$. For example, $p = 3$ and $m = 4$ leads to

$$\mathcal{K}^4 = \text{span}(\begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} , \begin{bmatrix} B_1 b_3 \\ B_2 b_1 \\ B_3 b_2 \end{bmatrix} , \begin{bmatrix} B_1 B_3 b_2 \\ B_2 B_1 b_3 \\ B_3 B_2 b_1 \end{bmatrix} , \begin{bmatrix} B_1 B_3 B_2 b_1 \\ B_2 B_1 B_3 b_2 \\ B_3 B_2 B_1 b_3 \end{bmatrix}) \ .$$

The idea of the p-cyclic GMRES method is to decouple the Krylov subspace (11) into p independent subspaces $\mathcal{K}_i^m$, each with dimension less than or equal to m. We illustrate this idea for the special case $p = 3$ and $m = 4$. The generalization to arbitrary p and m is straightforward. We seek an

approximate solution x^4 of (7) in the search space

$$\hat{\mathcal{K}}^4 = \left\{ x^4 = \begin{bmatrix} x_1^4 \\ x_2^4 \\ x_3^4 \end{bmatrix} \middle| \begin{array}{l} x_1^4 \in \mathcal{K}_1^4 = \mathrm{span}(b_1, B_1 b_3, B_1 B_3 b_2, B_1 B_3 B_2 b_1) \\ x_2^4 \in \mathcal{K}_2^4 = \mathrm{span}(b_2, B_2 b_1, B_2 B_1 b_3, B_2 B_1 B_3 b_2) \\ x_3^4 \in \mathcal{K}_3^4 = \mathrm{span}(b_3, B_3 b_2, B_3 B_2 b_1, B_3 B_2 B_1 b_3) \end{array} \right\} . \tag{12}$$

Algorithm 1: p-cyclic GMRES

1. Initialization:
   for $i = 1 : p$
   Define $H_i^m = \mathrm{zeros}(m+1, m)$
   $v_i^1 = b_i$
   $[v_i^1, \mathrm{dummy}, u_i^1] = \mathrm{house\_orth}(v_i^1, [\,])$
   $U_i^1 = u_i^1, \;\; V_i^1 = v_i^1$
   $\delta_i = b_i^T v_i^1$
   end
2. Iterate:
   for $j = 1 : m$
   $$\begin{bmatrix} \hat{v}_1^j \\ \hat{v}_2^j \\ \vdots \\ \hat{v}_p^j \end{bmatrix} = \begin{bmatrix} v_p^j \\ v_1^j \\ \vdots \\ v_{p-1}^j \end{bmatrix}$$
   for $i = 1 : p$ (parallelizable for loop)
   $[v_i^{j+1}, H_i^m(1 : j+1, j), u_i^{j+1}] = \mathrm{house\_orth}(B_i \hat{v}_i^j, U_i^j)$
   $U_i^{j+1} = [U_i^j \;\; u_i^{j+1}], \;\; V_i^{j+1} = [V_i^j \;\; v_i^{j+1}]$
   end
   end
3. Finish:
   Solve the least squares problem:
   $$\min_{y_1,\ldots,y_p} \| \begin{bmatrix} \tilde{I} & 0 & 0 & -H_1^m \\ -H_2^m & \tilde{I} & 0 & 0 \\ 0 & \ddots & \ddots & 0 \\ 0 & 0 & -H_p^m & \tilde{I} \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ \vdots \\ y_p \end{bmatrix} - \begin{bmatrix} \delta_1 e_1 \\ \vdots \\ \vdots \\ \delta_p e_1 \end{bmatrix} \|$$
   $$x^m = \begin{bmatrix} V_1^m y_1 \\ \vdots \\ V_p^m y_p \end{bmatrix}$$

The approximate solution $x^m \in \hat{\mathcal{K}}^m$ is chosen such that the norm of the residual $\|b - Ax^m\|$ is minimal. It is easily seen that $\mathcal{K}^m \subset \hat{\mathcal{K}}^m$, and we have $\dim(\hat{\mathcal{K}}^m) \leq mp$. Usually the dimension of $\hat{\mathcal{K}}^m$ is p times larger than the dimension of $\mathcal{K}^m$ and therefore we hope that p-cyclic GMRES achieves a much faster convergence than GMRES. Since $\mathcal{K}^m \subset \hat{\mathcal{K}}^m$, the following relation holds for the norm of the residual: $\|r^{m,\, p\text{-cyclic GMRES}}\| \leq \|r^{m,\, \mathrm{GMRES}}\|$.

An orthogonal basis for $\hat{\mathcal{K}}^m$ is desirable, for a practical and accurate method. Fortunately, we only need an orthogonal basis for each $\mathcal{K}_i^m$, $i =$

$1, \ldots, p$, which immediately leads to an orthogonal basis for $\hat{\mathcal{K}}^m$. In example (12), $B_1 b_3$ has to be orthogonalized with respect to b_1, and $B_1 B_3 b_2$ has to be orthogonalized with respect to b_1 and $B_1 b_3$, etc. Step 1 and step 2 of Alg. 1 construct an orthogonal basis in an Arnoldi-like way. In step 3 the approximate solution with minimal residual norm is computed. For $p = 1$ step 1 and 2 of Alg. 1 reduce to the standard Arnoldi method and Alg. 1 is equivalent to standard GMRES. The Householder method is used for the orthogonalization of the vectors. The function

$$[v_i^{j+1}, H_i^m(1 : j+1, j), u_i^{j+1}] = \text{house\_orth}(B_i v_{\tilde{i-1}}^j, U_i^j)$$

of Alg. 1, with $\tilde{i-1}$ defined as

$$\tilde{i-1} \equiv \begin{cases} i-1 \text{ if } i = 2, \ldots, p \ , \\ p \quad\quad \text{if } i = 1 \ , \end{cases}$$

computes v_i^{j+1} and $H_i^m(1 : j+1, j)$ such that

$$B_i v_{\tilde{i-1}}^j = [V_i^j \ \ v_i^{j+1}] H_i^m(1 : j+1, j) \ \text{ and } \ [V_i^j \ \ v_i^{j+1}]^T [V_i^j \ \ v_i^{j+1}] = I \ . \quad (13)$$

The Householder vectors are stored in U_i^{j+1}.

Equation (13) can be written in matrix form:

$$B_i V_{\tilde{i-1}}^m = V_i^{m+1} H_i^m \ .$$

Here H_i^m is an $m+1$ by m Hessenberg matrix. This leads to the following block matrix relation:

$$\begin{bmatrix} I & 0 & 0 & -B_1 \\ -B_2 & I & 0 & 0 \\ 0 & \ddots & \ddots & 0 \\ 0 & 0 & -B_p & I \end{bmatrix} \begin{bmatrix} V_1^m & 0 & \ldots & 0 \\ 0 & V_2^m & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \ldots & 0 & V_p^m \end{bmatrix} =$$
$$\begin{bmatrix} V_1^{m+1} & 0 & \ldots & 0 \\ 0 & V_2^{m+1} & & \vdots \\ \vdots & & \ddots & 0 \\ 0 & \ldots & 0 & V_p^{m+1} \end{bmatrix} \begin{bmatrix} \tilde{I} & 0 & 0 & -H_1^m \\ -H_2^m & \tilde{I} & 0 & 0 \\ 0 & \ddots & \ddots & 0 \\ 0 & 0 & -H_p^m & \tilde{I} \end{bmatrix} ,$$

with the $m+1$ by m matrix $\tilde{I} = \begin{bmatrix} I \\ 0 \ldots 0 \end{bmatrix}$, or shortly

$$AV^m = V^{m+1} H^m \ . \quad (14)$$

It is possible to prove that the subspace $\hat{\mathcal{K}}^m = \text{range}(V^m)$, if and only if $\{H_i^m\}_{j+1,j} \neq 0$, $j = 1, \ldots, m-1$, $i = 1, \ldots, p$ and $b_i \neq 0$, $i = 1, \ldots, p$, but the proof is beyond the scope of this paper (see [2]).

Step 2 is the most expensive part of Alg. 1. Step 2 is efficiently parallelizable on a p processor computer system, because only the $\hat{v}_i^j = v_{\tilde{i-1}}^j$ operations require communication.

In step 3 of Alg. 1 the approximate solution of $Ax = b$ is computed in a similar way as in GMRES. We seek an approximate solution x^m that solves the least squares problem

$$\min_{x^m \in \text{range}(V^m)} \|b - Ax^m\| \ . \tag{15}$$

Since $x^m \in \text{range}(V^m)$, we may substitute $x^m = V^m y$. Now formula (14) can be used to reduce the least squares problem (15) to the much smaller least squares problem of Alg. 1, step 3, see [2]. A common way to solve this reduced least squares problem is by QR factorization of H^m: $Q^m R^m = H^m$, see also [2]. Computing the QR factorization of H^m is relatively inexpensive if a suitable reordering is applied to H^m.

3 Numerical Experiments

In this section we demonstrate the effectiveness of the p-cyclic GMRES method for a number of periodic steady state test problems, arising in circuit simulation. The periodic DAE is solved by the procedure explained in Sect. 1. Table 1 shows the sizes of the test problems (2). The matrices are taken from the first Newton step of the nonlinear system solution process.

Table 1. Convergence of p-cyclic GMRES for periodic steady state test problems. M is the number of time points, n is the number of unknowns per time point. The number of iterations needed to satisfy the stopping criterion is displayed in the last 5 columns, for different values of p.

				iterations				
problem	M	n	Mn	$p=1$	$p=2$	$p=4$	$p=8$	$p=16$
pss_1	64	22	1408	5	5	6	7	8
pss_2	128	22	2816	9	10	11	11	11
pss_3	64	54	3456	12	12	12	12	12
pss_4	32	904	28928	1	2	2	4	6
pss_5	128	424	54272	34	36	43	52	67

In Sect. 1 we have described how to reduce (2) to a p-cyclic system. The integer partition (6) is chosen such that $\underline{q}_i - \overline{q}_i = \text{constant}$. This is possible since both p and M are powers of 2 here. Note that in a parallel environment it is desirable to have $\underline{q}_i - \overline{q}_i = \text{constant}$, for a good load balance. We apply p-cyclic GMRES to the reduced p-cyclic system, for different values of p.

The stopping criterion is based on the true residual of (2), which makes a fair comparison possible between the different values of p. The iterative process is stopped if the true residual of (2) is smaller than $10^{-10}\|c\|$, with $c = \left[c_1^T, \ldots, c_M^T\right]^T$.

The convergence results of p-cyclic GMRES are in Table 1. We see that usually the speed of convergence decreases with increasing p, but it decreases only slowly relative to the increase in p (see also Table 1).

The costs of recursion (9), which defines b_i, and the costs of recursion (3), which solves the missing y_i, are not negligible. For both recursions $M - p$ matrix vector products $C_i v$ are needed. Each p-cyclic GMRES iteration costs M matrix vector products $C_i v$. The overhead due to (9) and (3) amounts to approximately $2(M - p)/M \approx 2$ p-cyclic GMRES iterations. The p-cyclic GMRES iterations, as well as recursions (9) and (3) are efficiently parallelizable on a p processor computer system.

Recall that for $p = 1$ the computational costs of our method are approximately equal to the costs of the sequential method proposed in [5]. We conclude that for $p > 1$ our method is somewhat more expensive than the method of [5], but in contrast with this method, our method is efficiently parallelizable.

Acknowledgements

I would like to thank Jan ter Maten, Wil Schilders and Jaap Fijnvandraat (Philips Research) for posing the problem and for being critical on the ideas suggested by the author.

References

1. Ascher, U.M., Mattheij, R.M.M., Russell, R.D.: Numerical solution of boundary value problems for ordinary differential equations. SIAM, Philadelphia, PA, 1995.
2. Bomhof, C.W.: A parallelizable GMRES-type method for p-cyclic matrices. In: PhD thesis, Mathematical Institute, Utrecht University, to appear.
3. K. Dekker, *Parallel GMRES and Domain Decomposition.* Report 00-05, Department of Applied Mathematical Analysis, Delft University of Technology, 2000.
4. Saad, Y., Schultz, M.: GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM J. Sci. Statist. Comput. **7** (1986), pp. 856-869.
5. Telichevesky, R., Kundert, K., Elfadel, I., White, J.: Fast simulation algorithms for RF circuits. In proc. IEEE 1996 Custom Integrated Circuits Conf., San Diego, pp. 437-444, 1996.
6. Telichevesky, R., Kundert, K., White, J.: Efficient AC and noise analysis of two-tone RF Circuits. In proc. 33rd Design Automation Conference (DAC'96), Las Vegas, pp. 292-297, 1996.