# A structure preserving FGMRES method for solving large Lyapunov equations

Matthias Bollhöfer and André K. Eppler

**Abstract** We present a structure-preserving Krylov subspace method for solving large-scale Lyapunov equations where the (approximate) solution is of low rank. This problem arises, e.g., from model order reduction techniques based on Balanced Truncation for large-scale descriptor systems such as those in the simulation of large electrical circuits. The method presented here uses a low-rank approach based on the FGMRES method. For preconditioning the **L**ow **R**ank **C**holesky **F**actor-**A**lternating **D**irect **I**mplicit is applied which turns out to preserve the low-rank structures and allows for the use of inner approximate factorizations.

## 1 Introduction

In very large system integrated (VLSI) technology advances in size and speed leads to differential-algebraic equations (DAE) with several hundred million elements. For verification of the model a full simulation is often impossible. To do this efficiently model order reduction (MOR) methods have been recognized as a key technology in constructing reduced order models, in particular those methods that inherit the central properties of the underlying circuit like stability and passivity. The original linear circuit can be modelled by a DAE of type

$$\begin{aligned} E\dot{x}(t) &= Ax(t) + Bu(t) \\ y(t) &= Cx(t) + Du(t) \end{aligned} \tag{1}$$

Matthias Bollhöfer
TU Braunschweig, *Inst. Computational Mathematics*, Pockelsstr. 14, 38106 Braunschweig, e-mail: m.bollhoefer@tu-bs.de

André K. Eppler
TU Braunschweig, *Inst. Computational Mathematics*, Pockelsstr. 14, 38106 Braunschweig, e-mail: a.eppler@tu-bs.de

where $A,E \in \mathbb{R}^{n,n}, B \in \mathbb{R}^{n,m}, C \in \mathbb{R}^{p,n}, D \in \mathbb{R}^{p,m}$ with $m,p \ll n$. The reduced model approximates the system by smaller matrices $\tilde{E},\tilde{A},\tilde{B},\tilde{C},\tilde{D}$ such that the dimension $n$ is replaced by some $l \ll n$. For circuit equations Positive Real Balanced Truncation (BT) is a suitable method to reduce the dimension and preserve passivity at the same time as shown in [6, 7]. The main idea behind this method consists of balancing the solutions of associated projected Lur'e equations. For special cases these simplify to projected algebraic Riccati equations. For solving these second order matrix equations with Newton's method (see [5]) in every step projected, generalized Lyapunov equations of the following type need to be solved

$$A_k X_k E^T + E X_k A_k^T = -P_l B_k B_k^T P_l^T, \text{ where } X_k = P_r X_k P_r^T, \tag{2}$$

$$A^T Y_k E + E^T Y_k A_k = -P_r^T C_k^T C_k P_r, \text{ where } Y_k = P_l^T Y_k P_l. \tag{3}$$

Here $P_l, P_r$ refer to the left and right projection of the matrix pencil $\lambda E - A_k$ to the subspace of finite eigenvalues. This is necessary because the matrix $E$ is not invertible when dealing with electrical circuits. The matrix $A_k$ and the matrix of the right hand side $B_k$ may change in every outer Newton step, while the matrix $E$ remains unchanged from the descriptor system.

In the case of MOR we are dealing with a small number of input and output signals which in turn yields low rank matrices $B$ (resp. $C$). Moreover, assuming that the system is passive, we obtain symmetric and positive semidefinite solutions $X_k, Y_k$ of (2), (3) which are approximately of low rank, i.e., $X_k = L_k L_k^T$, $Y_k = R_k R_k^T$. It should be noted that for linear RCI networks the descriptor system can symmetrized, meaning that it is necessary only to solve one Riccati equation and therefore only one Lyapunov equation (2) is required at each Newton step, see [6].

In sections 2 and 3 we are going to explain how we combine the two methods to a new solver. First we comment on the LRCF-ADI method and then we state how this can be used within Krylov subspace methods. In section 4 we compare our numerical approach with the LRCF-ADI method for one example circuit and also investigate the use of inner approximate factorizations.

## 2 Low rank Cholesky factor-ADI (LRCF-ADI)

We now discuss solving generalized, projected Lyapunov equations of the form

$$AXE^T + EXA^T = -P_l B B^T P_l^T, \text{ where } X = P_r X P_r^T. \tag{4}$$

One of the most popular methods for solving Lyapunov equations is the alternating direction implicit (ADI) method which was first demonstrated in [10]. The transfer to the generalized case as part of BT can be found in [9]. Formally, for ADI one has to solve a sequence of pairs of equations

$$(E+\tau_j A)X_{j-\frac{1}{2}} = -P_l BB^T P_l^T - X_{j-1}(E-\tau_j A)^T$$
$$X_j(E+\tau_j A)^T = -P_l BB^T P_l^T - (E-\tau_j A)X_{j-\frac{1}{2}}$$

for $j=1,2,3,\ldots,t$. $X_j$ can be computed more efficiently using its representation as low-rank Cholesky factors $X_j = L_j L_j^T$, see [4]. The most efficient way, usually referred to as LRCF-ADI [3] uses elegant way of computing $L_j$ from $L_{j-1}$ such that the computational work grows at most linearly with the number of shifts $t$. In this simplest case $L_j$ can be computed from $L_{j-1}$ via

$$F_j = \sqrt{-\mathrm{Re}\tau_j}/\sqrt{-\mathrm{Re}\tau_{j-1}}\left(F_{j-1} - (\tau_j + \bar{\tau}_{j-1})(E+\tau_j A)^{-1}AF_{j-1}\right), L_j = \left[L_{j-1}, F_j\right],$$

$j=2,\ldots,t$, where $F_1 = \sqrt{-\mathrm{Re}\tau_1}(E+\tau_1 A)^{-1}B$, $L_1 = F_1$.

In practice it is by far too expensive to compute the residual $\|AL_jL_jE^T + EL_jL_j^TA^T + P_lBB^TP_l^T\|_F$. One can bypass this problem via the updated QR-factorization as shown in [4]. Here one computes and updates $QR$ factorizations of the form $Q_1R_1 = [AL_j, EL_j, P_lB]$, $Q_2R_2 = [EL_j, AL_j, P_lB]$. Then the norm reduces to $\|R_1R_2^T\|_F$. For the performance of the ADI method it is essential to solve systems $(E+\tau_jA)$ efficiently. In case of circuit equations these matrices become very large and sparse so the LU-decomposition can be used to solve the systems. The other advantage is, that once these factorizations are computed they can be reused in the next cycle of the ADI method.

## 3 Low rank Krylov subspace methods

For the solution of generalized Lyapunov we discuss the use of Krylov subspace methods. Here we will concentrate on the FGMRES [8] method. In contrast to the standard preconditioned GMRES procedure this allows for changing preconditioners in every iteration step. The elementary operations needed in Krylov subspace methods are matrix vector multiplications, scalar products as well as linear combinations of vectors. These templates will be reformulated in a low-rank pseudo-arithmetic that uses a representation $VZV^T$ similar to the Cholesky-factor representation. It is easy to see that a linear combination of symmetric low rank matrices leads again to a symmetric matrix of lower rank. Furthermore the “matrix-vector” multiplication by the Lyapunov operator applied to $VZV^T$

$$AVZV^TE^T + EVZV^TA^T = [AVEV]\begin{bmatrix} 0 & Z \\ Z & 0 \end{bmatrix}[AVEV]^T,$$

can be rewritten as a symmetric matrix of lower rank as well. As for the linear combination and the “matrix-vector” product after the concatenation of columns $[AVEV]$ the rank may increase. Suppose we have a low rank matrix $\mathscr{X} = VZV^T$. The two main steps in our rank truncation strategy are using an rank revealing QR and an eigenvalue decomposition. First we compute $V = QR\Pi$ where $\Pi$ is just column pivoting (see [2]) and cut off the rank according to $R$ based on a relative tolerance. We denote the remainders by $\hat{Q},\hat{R}$. After that we calculate a symmetric eigenvalue decomposition of $\hat{R}Z\hat{R}^T = U\Sigma U^T$ and discard small values of $\Sigma$ to get $\hat{U}$ and $\hat{\Sigma}$. Here we use the same tolerance again. As a result we have $\mathscr{X} = \hat{V}\hat{\Sigma}\hat{V}^T$ with a unitary matrix $\hat{V} = \hat{Q}\hat{U}$ and a diagonal matrix $\hat{\Sigma}$. Beside the symmetric low-rank pseudo-arithmetic we need to use preconditioning that is consistent with this representation. E.g., diagonal preconditioning can be implemented fairly easy but would violate the symmetric low-rank representation. We therefore propose to use the LRCF-ADI method instead. Suppose that at step $i$, the FGMRES method computes Arnoldi vectors $\mathscr{V}_i = V_i Z_i V_i^T$ represented as symmetric low rank matrices. Then one preconditioning step using LRCF-ADI consists of one cycle $j = 1,\ldots,t$ starting with right hand side $B = V_i$ (compare (4)) in the symmetric low-rank format. LRCF-ADI returns the Cholesky factor $L_t$ which induces a symmetric low rank matrix $\mathscr{W}_i = L_t D_t L_t^T$ for $i = 1,\ldots,m$, where $D_t = I_t \otimes Z_i$. As mentioned earlier, computing the residual during ADI iteration is quite costly and is usually replaced by the updated QR-factorization [4]. For our algorithm this can be avoided to save time and we can work with the residual norm computed by the FGMRES algorithm.

## 4 Numerical experiments

We have tested our method on several examples from NEC Europe and compared the solution with the state of the art method LRCF-ADI as stand-alone solver for the associated generalized, projected Lyapunov equations [9]. Here we only consider a RC-chain with 2002 capacitors, 2003 resistors and 3 voltage sources yielding state dimension $n = 2007$ and rank 3 of the right hand side. Since we concentrate on developing a new Lyapunov solver we set $A_j = A$ and $B_j = B$. We want to solve the Lyapunov equation up to a estimated residual norm of $10^{-10}$. We start the FGMRES with initial guess zero. As stated before it is essential for the performance of the LRCF-ADI method to solve the shifted systems $E + \tau_j A$ fast and reliable. We investigate the influence of approximative solves of these linear systems to both methods. For this purpose we use the software package ILUPACK, see [1]. A preconditioner for each of the systems $E + \tau_j A$ is computed and we solve for 3 different tolerances for the inner iterative solver, $10^{-12}$ , $10^{-8}$ and $10^{-4}$ respectively. To allow for such variable preconditioning techniques it is necessary to use the FGMRES method instead of regular GMRES. 15 different real shifts are chosen as explained in [4]. All computations were done on a workstation with Intel(R) Xeon(TM) MP CPU 3.66GHz and 16GB RAM using Matlab©Version 7.7.0.471 (R2008b).

In Fig. 1 the convergence history with respect to the number of ADI steps is plotted. The black lines refer to the proposed LR-FGMRES while the grey lines belong to the LRCF-ADI. It can be seen that the LRCF-ADI method stagnates for the tolerances $10^{-8}$ and $10^{-4}$ at a residual norm of about the same order of magnitude. The values in Fig. 1 are in fact $6.6e-5$ and $4.3e-9$. We stop the LRCF-ADI iteration after a maximum of 100 steps but only the first 60 ones are plotted here. With our

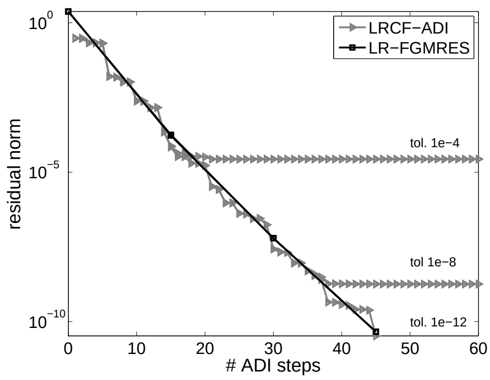


**Fig. 1** Convergence history of residual norms

LR-FGMRES we still get convergence even if the systems are only approximately solved. So our method is almost not affected by this. One can not distinguish the 3 different curves for the LR-FGMRES in Fig.1. With 3 cycles of FGMRES each of them using 15 ADI iterations we are at the same number of steps as the LRCF-ADI method.

| tolerance | LRCF-ADI | LR-FGMRES |
|---|---|---|
| $10^{-12}$ | 2.7 | 6.4 |
| $10^{-8}$ | - | 15.7 |
| $10^{-4}$ | - | 33.8 |

**Table 1** Running time [s] dependent on tolerance

In Table 1 one can see the cpu-time needed to solve the Lyapunov equations according to the related tolerance for solving the shifted systems inside the LRCF-ADI iteration. Table 2 shows the rank development of the right hand sides $B = V_{i-1}$ from equation (4) during the preconditioning step. Initially the rank is always 3. We observe for our method that the ranks of the right hand side increase faster as we raise the tolerance. This explains the longer computational time.

| FGMRES step $i$ | $10^{-12}$ | $10^{-8}$ | $10^{-4}$ |
|---|---|---|---|
| 1 | 3 | 3 | 3 |
| 2 | 8 | 19 | 21 |
| 3 | 25 | 52 | 93 |

**Table 2** Rank development dependent on tolerance

While the LRCF-ADI method is fast it only converges about the same tolerance as one solves the shifted linear systems. The FGMRES turned out to be more robust with respect to this tolerance, this effect was demonstrated by the use of the Multilevel-ILU.

## 5 Conclusions

In this paper we proposed a new method for solving large generalized, projected Lyapunov equations. Using Krylov subspace methods as outer iteration we were able to improve robustness of the LRCF-ADI method when using incomplete factorizations. This is intended to use where direct solvers can not be applied. For applications where larger rank does not contribute too much to the total computation time both methods are suitable. The increase of the ranks in our method will be subject of future work. Another advantage of our solver is the cheaply computable convergence criterion delivered by FGMRES. Other preconditioned Krylov subspace methods like CG, QMR or BiCGStab can be considered within this framework. This work can easily be transfered and applied to the case of standard Lyapunov equations.

**Acknowledgements** The authors would like to thank Carsten Neff and Achim Basermann formerly from NEC Europe for providing the circuit examples. This work is supported by the German Federal Ministry of Education and Research (BMBF) grant no. 03BOPAE4.

## References

1. Bollhöfer, M., Saad, Y.: Multilevel preconditioners constructed from inverse-based ILUs. SIAM J. Sci. Comput **27**, 1627–1650 (2006)
2. Golub, G.H., Van Loan, C.F.: Matrix Computations (Johns Hopkins Studies in Mathematical Sciences)(3rd Edition), 3rd edn. The Johns Hopkins University Press (1996).
3. Li, J.R., White, J.: Low rank solution of Lyapunov equations. SIAM J. Matrix Anal. Appl. **24**(1), 260–280 (2002).
4. Penzl, T.: A cyclic low-rank smith method for large sparse Lyapunov equations. SIAM J. Sci. Comput. **21**(4), 1401–1418 (2000).
5. Peter Benner Jing-Rebecca Li, T.P.: Numerical solution of large-scale Lyapunov equations, Riccati equations, and linear-quadratic optimal control problems. Numerical Linear Algebra with Applications **15**, 755777 (2008).
6. Reis, T., Stykel, T.: PABTEC: Passivity-preserving balanced truncation for electrical circuits. Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on **29**(9), 1354 –1367 (2010).
7. Reis, T., Stykel, T.: Positive real and bounded real balancing for model reduction of descriptor systems. Int. J. Control **83**(1), 74–88 (2010).
8. Saad, Y.: A flexible inner-outer preconditioned GMRES algorithm. SIAM Journal on Scientific Computing **14**(2), 461–469 (1993).
9. Stykel, T.: Low-rank iterative methods for projected generalized Lyapunov equations. Electron. Trans. Numer. Anal. **30**, 187–202 (2008).
10. Wachspress, E.L.: Iterative solution of the Lyapunov matrix equation. Appl. Math. Lett. **1**(1), 87–90 (1988).