Local Fourier Analysis for Tensor-Product Multigrid

Bart Vandereycken and Stefan Vandewalle

Katholieke Universiteit Leuven, Department of Computer Science
Celestijnenlaan 200A, 3001 Leuven, Belgium

Abstract. We present a new formulation of multigrid, the so-called tensor-product multigrid method, which can be used to solve Lyapunov equations. These matrix equations are of considerable importance in control theory and model reduction. Since they are formulated on a tensor product space, they are of possibly very large dimension and one needs an efficient solver like multigrid with optimal chosen components. We show that this can be done by computing the convergence factors with Local Fourier Analysis adapted for this tensor-product multigrid method.

Keywords: partial differential equations, Fourier analysis, iterative methods, multigrid, Lyapunov matrix equation
PACS: 02.60.-x, 02.60.Cb, 02.70.Bf, 02.70.Dh, 02.30.Nw, 02.30.Jr, 02.70.Bf

LARGE-SCALE LYAPUNOV EQUATIONS

Consider the following linear time-invariant system

$$\frac{dx_h}{dt} = A_h x_h(t) + B_h u_h, \quad x_h(0) = x_h^0, \tag{1}$$

with system matrix $A_h \in R^{n\times n}$, input matrix $B_h \in R^{n\times p}$, state vector $x_h \in R^n$ and input vector $u_h \in R^p$. In this paper, matrix A_h will be a finite difference approximation of a system of partial differential equations (PDE) on a grid G_h. The mesh-size of this discretization is emphasized by the subscript $_h$.

The controllability Gramian of system (1) can be found as the matrix $X_h \in R^{n\times n}$ in

$$A_h X_h + X_h A_h^T = C_h, \quad \text{with } C_h = B_h B_h^T. \tag{2}$$

Matrix equation (2) is called a Lyapunov equation and is of great importance in control theory [1] and model reduction [2]. Let $\mathrm{vec}(\cdot)$ be the operator that makes a vector from a matrix by column-wise stacking. By means of a tensor product $\otimes$, we can write (2) as the linear system

$$\mathcal{A}_h \,\mathrm{vec}(X_h) = \mathrm{vec}(C_h), \quad \text{with } \mathcal{A}_h = A_h \otimes I_h + I_h \otimes A_h, \tag{3}$$

where we denote the n-by-n identity matrix by I_h.

The subject of this paper is solving this Lyapunov equation for large-scale systems by multigrid. Large-scale Lyapunov equations arise naturally when system (1) is obtained by a semi-discretization of a PDE. Matrix A_h is then sparse but very large; $n > 10^6$ is not uncommon. A major challenge when solving (2) for such a large-scale problem is that the n^2-by-n^2 matrix X_h is almost always dense.

The dimension of a Lyapunov equation is n^2 which can become very large even for a moderate mesh-width. This shows the need for an efficient iterative method like multigrid to solve (2). In fact, for many PDEs, multigrid is known to be an optimal solver: the amount of work and memory scales linearly with the number of unknowns. In case of a Lyapunov equation, we will show that the multigrid algorithm can be formulated as an iterative method that operates on a tensor-product space. In this manner we obtain a so-called *tensor-product multigrid* method.

We can show that the typical multigrid optimality and efficiency caries over for this tensor-product multigrid method. In [3], this has been proved qualitatively for a specific instance of tensor-product multigrid that solves the two-dimensional heat equation on a square. By means of Local Fourier Analysis, we can show the optimality for more general formulations of the multigrid algorithm and for a wider class of elliptic PDEs. Furthermore, we are able to compute tight estimates of the convergence factor, both norm-wise and asymptotically.

TENSOR-PRODUCT MULTIGRID

We will first explain how a tensor-product multigrid method that solves (2) can be built from standard multigrid components. Let us state the following standard two-grid cycle for solving $A_h x_h = b_h$, see also [4, Ch. 2.2.3].

$$\begin{array}{lcl} x_h^1 & \leftarrow & \texttt{smooth}^{\nu_1}(x_h^0, A_h, b_h) = S^{\nu_1} x_h^0 + T b_h \\ d_H & \leftarrow & I_h^H (b_h - A_h x_h^1) \\ e_H & \leftarrow & \texttt{cgc}(A_H, d_H) = A_H^{-1} d_H \\ x_h^2 & \leftarrow & x_h^1 + I_H^h e_H \\ x_h^3 & \leftarrow & \texttt{smooth}^{\nu_2}(x_h^2, A_h, b_h) = S^{\nu_2} x_h^2 + T b_h \end{array}$$

The algorithm uses a fine grid G_h and a coarse grid G_H with $H = 2h$ and is based on ν_1 pre-smoothing steps, restriction of the defect by I_h^H, interpolation by I_H^h of the coarse grid correction e_H and ν_2 post-smoothing steps. In *multi*-grid the `cgc` function is a recursive call to a two-grid cycle on the coarser grid G_H. The error $e^3 = x_h - x_h^3$ after one total step of the two-grid cycle obeys $\|e^3\| \leq \|M\| \|e^i\|$ where M is the error-amplification matrix of the algorithm and $e^0 = x_h - x_h^0$.

In order to use this two-grid cycle for the Lyapunov equation (3), we need to elaborate the previous multigrid components. Based on the observation that a lot of these components are already based on tensor products of lower dimensional variants, e.g., bilinear interpolation, we can specify the previous operations as acting on the tensor-product space $R^n \times R^n$. In the following, we will denote grids and operators that belong to this tensor-product space by a calligraphic symbol, e.g., $\mathcal{G}_h$.

Hierarchy of grids. Multigrid is a cycling between a hierarchy of grids. For the tensor-product multigrid method we will simply take tensor products of the hierarchy of grids from the original multigrid solver, i.e. $\mathcal{G}_h = G_h \otimes G_h$ for all meshes h. It will be convenient to partition a point $x \in \mathcal{G}_h$ on the grid as $x = x_1 \otimes x_2$ with $x_1, x_2 \in G_h$.

Once the grids are defined, we need a discrete representation of the PDE on these grids. It can be shown that in case of a finite difference scheme, one obtains $\mathcal{A}_h = A_h \otimes I_h + I_h \otimes A_h$, like eq. (3). Besides the fine grid operator $\mathcal{A}_h$ we also need a coarse grid operator $\mathcal{A}_H$. We will only consider direct coarsened operators A_H and the corresponding tensor-product operator $\mathcal{A}_H = A_H \otimes I_H + I_H \otimes A_H$. A Galerkin operator has the disadvantage that the stencils become larger and it has to be computed recursively, both of which may become problematic in higher dimensions.

Intergrid transfers. Prolongation and restriction for the tensor grid are easily constructed as tensor products of the operators on G_h. Indeed, suppose $I_h^H : G_h \to G_H$ then $I_h^H \otimes I_h^H$ defines a suitable restriction $\mathcal{I}_h^H : G_h \otimes G_h \to G_H \otimes G_H$. Prolongation is analogous. From a practical point of view, these operators can be applied directly to the matrix, e.g. for the restriction we get

$$\mathcal{I}_h^H : u_h = \mathrm{vec}(U_h) \mapsto (R \otimes R)\,\mathrm{vec}(U_h) = \mathrm{vec}(R U_h R^T).$$

This means we apply the restriction on each column and each row of the matrix U_h.

Smoothing. Smoothers in multigrid are often based on a splitting $A_h = A_h^+ - A_h^-$ to obtain the relaxation $A_h^+ x_h^{i+1} = A_h^- x_h^i + b$, possibly combined with partial relaxations on a colored grid. This splitting can be directly carried over on $\mathcal{G}_h$, which gives $\mathcal{A}_h^+ = A_h^+ \otimes I_h + I_h \otimes A_h^+$. From this, we get a smoother on $\mathcal{G}_h$ with error amplification matrix $\mathcal{S}$.

LOW-RANK TENSOR-PRODUCT MULTIGRID

For Lyapunov equations, the n^2 unknowns may be too large to store in computer memory. Therefore, one usually approximates the solution matrix X_h itself. A popular choice is approximating X_h by a low-rank matrix of rank $k \ll n$, which only requires $O(n)$ parameters. In [5], Grasedyck and Hackbusch developed a multigrid solver that solves for this low-rank matrix directly without ever forming the whole solution matrix X_h explicitly. This way, one can combine multigrid as an optimal solver for PDEs and benefit from the reduced dimension of the low-rank approximation. The low-rank multigrid solver uses a Jacobi smoother, and standard prolongation and restriction operators.

Now, this low-rank multigrid algorithm can be seen as a perturbation of the tensor-product multigrid method. After one step of low-rank multigrid, one can show that the error obeys $\|e^3\| \leq \|\mathcal{M}\| \|e^0\| + \epsilon$ with $\epsilon \leq \|\mathcal{S}^{\nu_1}(I_h - \mathcal{I}_H^h \mathcal{A}_H^{-1} \mathcal{I}_h^H \mathcal{A}_h)\| \epsilon_{s1} + \|\mathcal{S}^{\nu_1} \mathcal{I}_H^h \mathcal{A}_H^{-1} \mathcal{I}_h^H\| \epsilon_d + \|\mathcal{S}^{\nu_1} \mathcal{I}_H^h\| \epsilon_{cs} + \|\mathcal{S}^{\nu_1}\| \epsilon_{cc} + \epsilon_{s2}$. These ϵ_* are (small) errors due to the low-rank approximations and can be controlled by choosing the rank adaptively in the algorithm. In addition, tight estimates of the norms of the operators in front of these ϵ_* can be computed by the analysis of the next section. So, by keeping the total perturbation ϵ small, we are able to recover the efficiency of an unperturbed multigrid algorithm.

LOCAL FOURIER ANALYSIS

The convergence of a multigrid method can be proved in several ways. One way is by means of Local Fourier Analysis (LFA) [4, Ch. 4], which assumes that the PDE has constant coefficients and is defined on an infinite domain. We show how LFA can be used to analyze the tensor-product multigrid method.

Classic LFA

Let us introduce the standard LFA notation [4, 6] adapted to a general d-dimensional setting. We associate a fixed mesh width $h = (h_1, \ldots, h_d)$ with an infinite grid $G_h = \{x = (x_1, \ldots, x_d) = \kappa h = (\kappa_1 h_1, \ldots, \kappa_d h_d), \kappa \in \mathbb{Z}^d\}$. On this grid the discrete operator A_h corresponds to a difference stencil $[s_\kappa]_h$, i.e.

$$A_h w_h(\boldsymbol{x}) = \sum_\kappa s_\kappa (x + \kappa h).$$

The formal eigenfunctions or Fourier modes of this operator are given by $\varphi(\theta, x) = e^{\mathrm{i}\theta x h} = e^{\mathrm{i}\theta_1 x_1/h_1} \cdots e^{\mathrm{i}\theta_1 x_d/h_d}$ for $x \in G_h$ with formal eigenvalue or symbol

$$\widetilde{A}_h(\theta) = \sum_\kappa s_\kappa e^{\mathrm{i}\theta\kappa}.$$

The frequency $\theta \in \mathbb{R}^d$ varies continuously in the analysis with $\theta \in [-\pi, \pi)^d$.

In addition, we assume a coarse grid $G_H = \{x = \kappa H, \kappa \in \mathbb{Z}^d\}$. The mesh-width H depends on the type of coarsening, e.g. standard coarsening gives $H_i = 2h_i$ for all $i = 1, \ldots, d$. Based on the aliasing on this coarsened grid, one can classify the Fourier modes into high and low frequency components on G_h. This results in a space for the Fourier modes, called the $2h$-harmonics, which are indistinguishable on the coarse grid G_H.

It is well known that the two-grid cycle is invariant for this space of $2h$-harmonics for a wide range of smoothers, and restriction and prolongation operators. This results in an error amplification matrix M of the two-grid cycle that is similar to a block-diagonal matrix $\widetilde{M}$ with blocks $\widetilde{M}_i$ of size 2^d. Convergence factors can then be easily computed by iterating over all the blocks, or equivalently over all the low frequency Fourier modes:

$$\rho(M) = \max_{\theta^{\text{low}}} \rho(\widetilde{M}(\theta)) = \max_i \rho(\widetilde{M}_i) \quad \text{and} \quad \|M\| = \max_{\theta^{\text{low}}} \sqrt{\rho(\widetilde{M}(\theta)^T \widetilde{M}(\theta))} = \max_i \sqrt{\rho(\widetilde{M}_i^T \widetilde{M}_i)}. \tag{4}$$

Here $\rho(M)$ denotes the asymptotic convergence factor of M, or equivalently the spectral radius of M.

LFA for tensor-product multigrid

The Fourier modes on the tensor-product grid $\mathcal{G}_h$ are simply the tensor product of the Fourier modes on G_h, $\varphi(\theta, x) = \varphi(\theta_1, x_1)\varphi(\theta_2, x_2)$, where have used the partition $\theta = \theta_1 \otimes \theta_2$ with $\theta \in [-\pi, \pi)^{2d}$ and $\theta_1, \theta_2 \in [-\pi, \pi)^d$. Since the components of the tensor-product multigrid are tensor products of the corresponding components of a classic multigrid, one can compute the Fourier symbols accordingly. So, for the Fourier symbol of $\mathcal{A}_h$ we get

$$\begin{aligned}
\mathcal{A}_h \varphi(\theta, x) &= \left\{ \sum_\kappa s_\kappa \varphi(\theta_1, x_1 + \kappa h) \right\} \varphi(\theta_2, x_2) + \varphi(\theta_1, x_1) \left\{ \sum_\kappa s_\kappa \varphi(\theta_2, x_2 + \kappa h) \right\} \\
&= \left(\widetilde{A}_h(\theta_1) + \widetilde{A}_h(\theta_2) \right) \varphi_1(\theta_1, x_1)\varphi_2(\theta_2, x_2) = \widetilde{\mathcal{A}}_h(\theta)\varphi(\theta, x)
\end{aligned}$$

The other tensor-product multigrid operators, i.e., the smoother, prolongation and restriction, are analogous. We remark that the Fourier symbols of a smoother on a colored grid, like RB-GS, are slightly more tedious to compute. However, there is a systematic way of deriving the symbols which can be implemented in a symbolic software package like Maple.

The space of $2h$-harmonics on $\mathcal{G}_h$ is again the tensor product of that on G_h. As a result, the error amplification matrix $\mathcal{M}$ of the two-grid cycle is similar to a block-diagonal matrix $\widetilde{\mathcal{M}}$ with blocks of size 2^{2d} instead on 2^d. Convergence factors are computed similarly to (4). Note however, that since $\theta \in [-\pi, \pi)^{2d}$ the total work to compute these estimates is squared compared to that of (4).

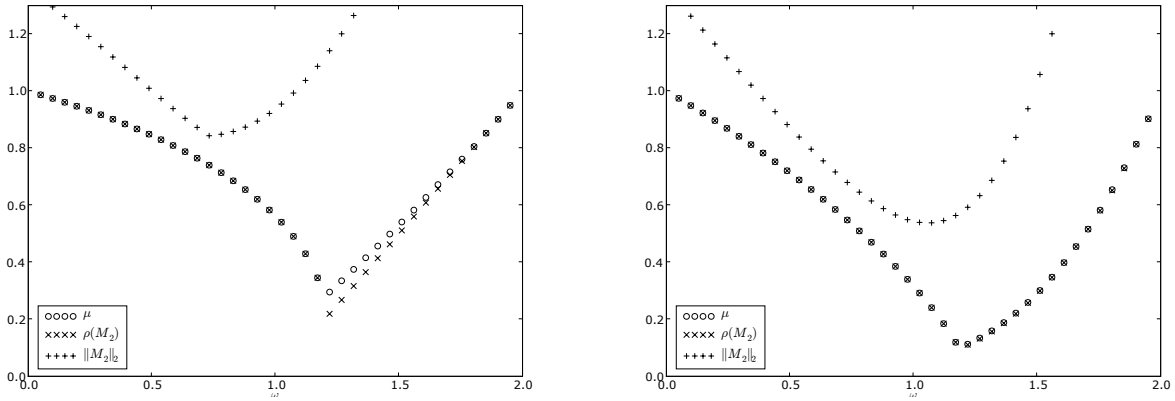

FIGURE 1. Smoothing factor μ, convergence factor $\rho(M)$ and spectral norm $\|M\|$ of the two-grid cycle with respect to the dampening factor ω. Left 1 step pre-smoothing and no post smoothing, right 1 step pre- and post-smoothing.

Results

We present a typical result of LFA for a Lyapunov equation coming from an isotropic Poisson operator. For this kind of operators, Red-Black Gauss-Seidel (RB-GS) turns out to be a very cost-effective smoother. It was pointed out in [7] that for higher dimensional systems, RB-GS with an over-relaxation parameter ω can greatly benefit the smoothing factor, much more than for classic two- or three-dimensional systems. However, these results only consider the smoothing factor and they lack a complete analysis of the two-grid cycle.

In Figure 1 we present the convergence factors in function of this parameter ω for a 4-dimensional system resulting from LFA with RB-GS, full weighting and bilinear interpolation. This 4-dimensional system was obtained as a Lyapunov equation with a two-dimensional Poisson operator. It is clear that while the smoothing factor gives indeed a good indication of the asymptotic convergence factor $\rho(M)$, this is not so for the norm-wise reduction $\|M\|$. The reduction factor $\|M\|$ is considerably larger than the smoothing factor μ and they both have a different minimizer for ω. Remark that for some choices of ω, we have that $\|M\| > 1$, which means that the error can grow after one iteration of multigrid. Since the *asymptotic* convergence factor $\rho(M)$ is smaller than one, this is normally not an issue for classic multigrid. However, for low-rank multigrid, the truncation errors due to the low-rank approximation should stay small. When $\|M\| > 1$, the total truncation error can possibly blow up, which may hinder convergence.

ACKNOWLEDGMENTS

Bart Vandereycken is a Research Assistant of the Research Foundation–Flanders (FWO). Both authors have been partially supported by the Research Council K.U. Leuven, CoE EF/05/006 Optimization in Engineering (OPTEC) and present results of the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office. The scientific responsibility rests with its authors.

REFERENCES

1. P. Benner, *Control Theory*, Handbook of Linear Algebra, Chapman & Hall/CRC, 2006.
2. A. C. Antoulas, *Approximation of Large-Scale Dynamical Systems*, Adv. Des. Control, SIAM, Philadelphia, 2005.
3. T. Penzl, A Multi-Grid Method for Generalized Lyapunov Equations, Tech. Rep. SFB393/97-24, Technische Universität Chemnitz (1997).
4. U. Trottenberg, C. Oosterlee, and A. Schüller, *Multigrid*, Academic Press, 2001.
5. L. Grasedyck, and W. Hackbusch, *SIAM J. Numer. Anal.* **29**, 870–894 (2006).
6. R. Wienands, and W. Joppich, *Practical Fourier Analysis for Multigrid Methods*, CRC-Press, 2004.
7. I. Yavneh, *SIAM J. Numer. Anal.* **32**, 1126–1138 (1995).