Filtering Techniques for Complex Geometry Fluid Flows

Julie S. Mullen[1] and Paul F. Fischer[2]

Abstract

We develop a class of filters based upon the numerical solution of high-order elliptic problems in $\mathbb{R}^d$ that allow for independent determination of order and cut-off wave number and that default to classical Fourier-based filters in homogeneous domains. However, because they are based on the solution of a PDE, the present filters are not restricted to applications in tensor-product-based geometries as is generally the case for Fourier-based filters. The discrete representation of the filtered output is constructed from a Krylov space generated in solving a well-conditioned system arising from a low-order PDE.

1 Introduction

The need for low-pass filters arises in the numerical solution of partial differential equations (PDE's) in many areas of science and engineering. The ability to control high-frequency content is an essential ingredient for many applications in computational fluid dynamics [1, 7, 10]. In particular, recently developed dynamic subgrid scale models for large eddy simulations (LES) require a two-level filter to extrapolate the leading-order effects of unresolved scales of motion [5, 8, 12]. For over two decades, the majority of LES calculations have employed Fourier spectral discretizations in one or more of the spatial directions and have generally filtered the solution *in the Fourier directions only.* The literature concerning filtering in the nonhomogeneous direction is almost nonexistent. As computing hardware and algorithms have reached the level of efficiency where we can now consider employing LES in complex three-dimensional geometries that lack homogeneity in any direction, the need for more general filtering techniques arises.

In the present work we are considering spectral element discretizations for primitive variable formulations of the Navier-Stokes equations (e.g., [4, 9]). Spectral elements are a natural extension of spectral methods in which Nth-order (polynomial) bases are used to represent the velocity and pressure within each of K subdomains or elements. Inter-element continuity requirements vary with the particular implementation, but for second-order PDE's, C^0 continuity is generally sufficient to retain spectral accuracy. It is tempting to filter such approximations in a fashion analogous to global spectral methods, that is, through truncation of high-modes within each subdomain. However, the interelement continuity constraint implies that such a local approach can potentially lead to global contamination of low-wave number modes, even in the one-dimensional case and is therefore

[1] Division of Applied Mathematics, Brown University, Providence, RI 02912

[2] Mathematics and Computer Science Division, Argonne National Laboratory, Argonne, IL 60439

unsuitable as a low-pass filter.

An alternative to the purely local filtering process of truncating modes within each subdomain is to solve a global Helmholtz problem of the form

$$\begin{aligned} -\nabla^2\bar{u} + \alpha\bar{u} &= \alpha u \quad \text{in} \quad \Omega \\ \bar{u} &= u \quad \text{on} \quad \partial\Omega \quad , \end{aligned} \tag{1}$$

where $u(\mathbf{x})$ is the input function and $\bar{u}(\mathbf{x})$ is the desired filtered output. Note that if we take $G(\mathbf{x},\mathbf{x}')$ to be the Green's function associated with (1), then solving for $\bar{u}$ corresponds to the usual notion of filtering via convolution with the kernel G and characteristic length scale $\Delta \sim 1/\sqrt{\alpha}$,

$$\bar{u}(\mathbf{x}) \;=\; F(u(\mathbf{x})) \;=\; \int_{\Omega} G(\mathbf{x},\mathbf{x}')\, u(\mathbf{x}')\, \mathbf{dx}' \quad .$$

The effect of such a filter on the input is clear if we consider the one-dimensional case with $\Omega = [0,1]$ and $u(0) = u(1) = 0$. For sufficiently smooth $u(x)$, we can express u via a sine expansion, $u(x) = \sum_k \hat{u}_k \sin(k\pi x)$, resulting in

$$\bar{u}(x) \quad = \quad \sum_k \frac{\alpha}{\alpha + \pi^2 k^2}\, \hat{u}_k \sin(k\pi x) \quad . \tag{2}$$

Defining the cut-off wave number, k_c, as the point where the transfer function is diminished by a factor of two (3dB-down point) leads to the choice of $\alpha = \pi^2 k_c^2$. It is clear that the filtered modes corresponding to $k \gg k_c$ are largely damped out, while those for $k \ll k_c$ are less affected.

In higher space dimensions, the above filter can be implemented by discretizing (1) in $\mathbb{R}^d$ and solving the resultant linear system using routines readily available in most CFD codes. For general three-dimensional geometries the resultant (well-conditioned) system is best solved iteratively, for example, via conjugate gradient iteration if the discretization leads to a symmetric positive definite (SPD) matrix.

Unfortunately, as illustrated by the transfer function shown in Fig. 1, the second-order filter (2) does not have a very sharp decay. While there has been some study of the potential of second-order filters in LES (e.g., [12]), most LES calculations to date have employed sharp cut-off filters. These are readily implementable in simple geometries that admit the use of Fourier bases. For general geometries, one might try to sharpen the transfer function through m repeated applications of the filter (1), requiring a sequence of m Helmholtz solves, each with scaling factor $\alpha = \pi^2 k_c^2/(2^{\frac{1}{m}} - 1)$ to maintain the 3dB-down point at $k = k_c$ in the final output. The ultimate limit of such a process as $m \longrightarrow \infty$ is a Gaussian filter with transfer function $e^{-\ln 2\,(k/k_c)^2}$. While this filter does provide more rapid decay for the high wave number components, it does little to sharpen the transfer function for $k < k_c$, as seen in Fig. 1. Clearly, the family of rational polynomials obtained through repeated Helmholtz solves of the type (1) is not rich enough to capture the behavior of the sharp cut-off filter, and the potential of such a straightforward approach is limited.

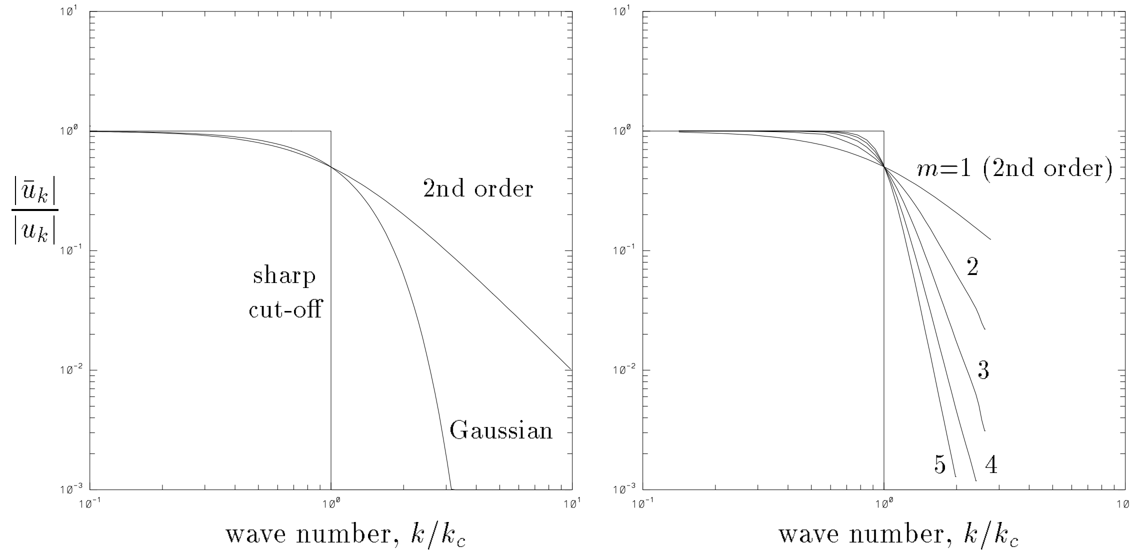


Figure 1: Transfer function vs. wave number, commonly used filters (left) and higher-order filters (right).

A simple class of rational polynomials in k that does converge to the sharp cut-off filter is given by $\left[1+(\frac{k}{k_c})^{2m}\right]^{-1}$. The corresponding PDE has the form

$$-\nabla^{2m}\bar{u} + \alpha\bar{u} \quad = \alpha u \quad \text{in} \quad \Omega \qquad , \tag{3}$$

and the resultant transfer functions for various m are shown in Fig. 1 (right). It is quite clear that for higher values of m, the filter has diminished impact on the amplitude of the low modes, $k < k_c$, as desired.

While (3) is not readily solvable with standard techniques, we demonstrate in the remaining sections that it is possible to construct a good approximation through numerical solution of a much simpler problem, namely, the Poisson equation

$$\begin{aligned} -\nabla^2 v &= u \quad \text{in} \quad \Omega \\ v &= 0 \quad \text{on} \quad \partial\Omega \quad . \end{aligned} \tag{4}$$

Consequently, one can generate high-order filters by employing solver technology that already exists in most three-dimensional general geometry Navier-Stokes codes.

2 Filtering for the Discrete Problem

As our point of departure, we assume that the discretization of (4) leads to a linear system of the form

$$A\underline{v} = \underline{u} \tag{5}$$

where underscore denotes vectors of basis coefficients and $A \in \mathbb{R}^{n\times n}$ is the SPD discrete Laplacian as would be obtained, for example, using second-order finite differences. For general-geometry problems in three space dimensions, A is typically very sparse, and in this case (5) is most effectively solved by iterative methods. The convergence rate of these methods is largely dependent upon the condition number of system, which for SPD matrices is the ratio of the maximum to minimum eigenvalues, $\kappa_A = \lambda_n/\lambda_1$.

The system for the filtered quantities is derived from the discrete analog of (3),

$$[\,A^m \;+\; \alpha I\,]\,\underline{\bar{u}} \;\;=\;\; \alpha\,\underline{u} \quad . \tag{6}$$

The conditioning of (6) is dependent upon the values of α and m. If we denote by P_m the polynomial

$$P_m(A) \;\;=\;\; \frac{1}{\alpha}A^m + I\,, \tag{7}$$

then the corresponding condition number is $\kappa_{P_m} = (\lambda_n^m + \alpha)/(\lambda_1^m + \alpha)$. If $\alpha \simeq \lambda_n^m$, then $\kappa_{P_m} \simeq 2$. However, it is not clear *a priori* how λ_n will relate to a particular choice of α and m. In fact, it is clear that P_m will generally become very ill-conditioned with increasing m. This follows from the fact that, as $m \longrightarrow \infty$, P_m^{-1} is a projector onto the low wave number modes and P_m is consequently undefined in this limit. In practice, we have found P_m to be so ill-conditioned as to defy any attempt to solve (6) via standard iterative procedures that require matrix-vector products in P_m. For even moderate values of m (e.g., $m = 6$), just one application of P_m to a vector can lead to unacceptable round-off error.

Fortunately, P_m and A share the same set of eigenvectors, and we can use this fact to generate good approximate solutions to (6). In [11], van der Vorst proposed a method for approximating the solution to $f(A)\underline{w} = \underline{u}$ using a conjugate gradient/Lanczos procedure based upon the solution of $A\underline{v} = \underline{u}$. Consider the following conjugate gradient (CG) algorithm preconditioned by the SPD matrix M [6]:

```
$\underline{v}_0 = 0$; $\underline{r}_0 = \underline{u}$; $\beta_0 = 0$
for $i = 1, \ldots, j$
    solve $M\underline{z}_{i-1} = \underline{r}_{i-1}$
    $\rho_{i-1} = \underline{z}_{i-1}^T \underline{r}_{i-1}$
    if $(i > 1)$  $\beta_{i-1} = \frac{\rho_{i-1}}{\rho_{i-2}}$
    $\underline{p}_i = \underline{z}_{i-1} + \beta_{i-1}\underline{p}_{i-1}$
    $\gamma_i = \underline{p}_i^T A \underline{p}_i$
    $\alpha_i = \frac{\rho_{i-1}}{\gamma_i}$                    (8)
    $\underline{v}_i = \underline{v}_{i-1} + \alpha_i \underline{p}_i$
    $\underline{r}_i = \underline{r}_{i-1} - \alpha_i A \underline{p}_i$
end
```

Following van der Vorst, we begin by considering the unpreconditioned case where $M \equiv I$. Let $R = R_j = (\underline{\tilde{r}}_0, \underline{\tilde{r}}_1, \ldots, \underline{\tilde{r}}_{j-1})$ be the matrix containing the residual vectors generated by the above algorithm, each normalized according to $\underline{\tilde{r}}_i \;=\; \frac{1}{\sqrt{\rho_i}}\underline{r}_i$. The $\underline{\tilde{r}}_i$s form an orthonormal basis for the Krylov subspace $\mathcal{K}_j(A;\underline{u}) = \mathcal{K}_j(A;\underline{r}_0) \equiv span\{\underline{r}_0, A\underline{r}_0, \ldots, A^{j-1}\underline{r}_0\}$, and $\underline{v}_j$ is the projection of $\underline{v}$ onto this subspace; that is, the final residual $\underline{r}_j \;\equiv\; \underline{u} - A\underline{v}_j$ is orthogonal to $\mathcal{K}_j(A;\underline{u})$. The tridiagonal matrix

$$T \;=\; \begin{bmatrix} d_1 & e_1 & & & \\ e_1 & d_2 & e_2 & & \\ & e_2 & d_3 & & \\ & & & \ddots & e_{j-1} \\ & & & e_{j-1} & d_j \end{bmatrix} \tag{9}$$

$$d_i \;=\; \frac{1}{\alpha_i} + \frac{\beta_{i-1}}{\alpha_{i-1}}\,, \qquad\qquad e_i \;=\; \frac{\sqrt{\beta_i}}{\alpha_i}\,,$$

is the projection of A onto R given by $T = R^T A R$, and the above CG method produces the approximate solution, $\underline{v}_j \approx \underline{v}$, given by

$$\underline{v}_j \;=\; R\,(R^T A R)^{-1} R^T \underline{u} \;\;=\;\; R\,T^{-1} R^T \underline{u}\,.$$

The CG/Lanczos procedure (see, e.g., [6]) employs T and R to obtain estimates of the eigenpairs of A as follows. Let $SQS^T = T$ be the similarity transformation for T. The matrix $Q = diag(q_i)$ contains the eigenvalues of T that are taken as approximations to the j extremal eigenvalues of A. The matrix S contains the eigenvectors of T, and RS contains the corresponding approximate eigenvectors of A. In [11], van der Vorst showed that these approximate eigenpairs could be used to compute an approximate solution, $\underline{\hat{w}} \approx \underline{w}$, to problems of the form $f(A)\underline{w} = \underline{u}$ by simply computing

$$\underline{\hat{w}} \;\equiv\; RS[f(Q)]^{-1} S^T R^T \underline{u}\,. \tag{10}$$

The $j \times j$ matrix $f(Q) = diag(f(q_i))$ is trivially inverted. As noted in [11], this approach is identical to applying CG to $f(A)$ in the case where $f(A) = A + \alpha I$, since A and $A + \alpha I$ share the same Krylov space. However, in general, $f(A)$ does not share the same Krylov space as A, nor is $\underline{\hat{w}}$ the projection of $\underline{w}$ onto R. Nonetheless, (10) provides a very inexpensive means to compute approximations to $\underline{w}$, and one that might in fact be better (because of conditioning considerations) than straightforward application of CG to $f(A)$.

The application of van der Vorst's method to the filtering problem is straightforward when the discretization of (3) yields a polynomial $P_m(A)$ (7). One simply solves $A\underline{v} = \underline{u}$, computes S and Q from T, sets $f(q_i)^{-1} = 1/P_m(q_i)$, and computes $\underline{\bar{u}} \equiv \underline{\hat{w}}$ using (10).

If a Galerkin procedure is used to discretize the second-order filter (1), one obtains

$$[\,A \;+\; \alpha\,B\,]\,\underline{\bar{u}} \;\;=\;\; \alpha\,B\underline{u}\;\;.$$

This is the case encountered in the finite or spectral element formulation and differs from the finite difference case by the presence of the SPD mass matrix, B. In this case the discrete Laplacian is no longer A, but rather $B^{-1}A$, and the analogous discrete form of the high-order filter (3) is

$$[\,(B^{-1}A)^m \;+\; \alpha\,I\,]\,\underline{\bar{u}} \;\;=\;\; \alpha\,\underline{u}\;\;. \tag{11}$$

It is therefore necessary to solve an equation of the form $f(B^{-1}A)\underline{w} = \underline{u}$.

To extend van der Vorst's procedure to this more general case, we retain the preconditioned form of the CG algorithm, setting $M \equiv B$. (In the spectral element method B is diagonal, so its inversion is trivial. In the finite element method it probably suffices to employ a (diagonal) lumped mass matrix in place of the standard mass matrix.) If we define $Z = M^{-1}R$, with R containing the normalized residual vectors as defined above, then the columns of Z are orthonormal with respect to the M inner product, that is, $Z^TMZ = I$. The tridiagonal matrix (9) is now $T = Z^TAZ$, and the eigenpairs of $T = SQS^T$ yield approximate eigenvalues (q_i) and eigenvectors (the columns of ZS) for the generalized eigenvalue problem $A\underline{x} = \lambda M\underline{x}$. The resultant extension of van der Vorst's method to $f(B^{-1}A)\underline{w} = \underline{u}$ is

$$\hat{\underline{w}} = B^{-1}RS[f(Q)]^{-1}S^TR^T\underline{u}\,, \tag{12}$$

where R, Q, and S result from the Lanczos/CG procedure (8) applied to $A\underline{v} = B\underline{u}$, preconditioned with $M = B$.

To summarize, the filtering problem in the Galerkin case is solved by setting $f = P_m$ (7) and using (12) to compute the filtered output, $\bar{\underline{u}} = \hat{\underline{w}}$. Note that if the approximation space (given by the range of $B^{-1}R$) is to contain $\underline{u}$, then the initial residual in the CG procedure must be $\underline{r}_0 = B\underline{u}$ rather than just $\underline{u}$.

At this point, the issue of inhomogeneous boundary conditions has not been addressed. A standard procedure (e.g., [4]) for implementing inhomogeneous Dirichlet boundary conditions in the numerical solution of PDEs is to split the solution into $v = v_h + v_b$, where v_h satisfies homogeneous boundary conditions and v_b is any known function that satisfies the desired boundary conditions on v, and to then subtract the inhomogeneous boundary term from both sides of the equation. For example, for Poisson's equation, we have

$$\begin{aligned} -\nabla^2 v_h &= f + \nabla^2 v_b \quad \text{in} \quad \Omega \\ v_h &= 0 \qquad\qquad\; \text{on} \quad \partial\Omega \quad . \end{aligned}$$

The advantage of this approach is that the trial and test spaces in the Galerkin formulation are now coincident and the resultant discrete operator on the left will be symmetric. Straightforward application of a similar decomposition for the filter problem (3) would yield

$$P_m\bar{\underline{u}}_h \quad = \quad \underline{u} - \bar{P}_m\bar{\underline{u}}_b\,, \tag{13}$$

where $\bar{P}_m$ is given by (7), appropriately augmented to incorporate the effects of the boundary conditions upon the interior nodal point values.

Unfortunately, formation of the right-hand side of (13) requires evaluation of $(B^{-1}A)^m\bar{\underline{u}}_b$ which, unless $\bar{\underline{u}}_b$ is the solution to Laplace's equation, is numerically unstable because of the conditioning problems noted earlier. A suitable alternative is to set $\underline{u}_b = \underline{u}$ and rewrite the filter in the following equivalent form:

$$\begin{aligned} \left(B^{-1}A \;+\; \alpha(B^{-1}A)^{1-m}\right)\bar{\underline{u}}_h \quad &= \quad -B^{-1}A\underline{u} \\ \bar{\underline{u}} \quad &= \quad \bar{\underline{u}}_h \;+\; \underline{u} \;\; . \end{aligned} \tag{14}$$

Once again we have an equation of the form $f(B^{-1}A)\underline{x} = \underline{b}$, which is solved using the techniques outlined above. Note that this approach actually corresponds to subtracting off the high wave number components of $\underline{u}$ to obtain the low pass complement.

3 Results and Discussion

We present initial tests of the PDE/Krylov-based filtering strategy for two- and three-dimensional problems discretized with spectral elements. Similar results have also been obtained in the somewhat easier case of finite difference approximations.

We begin with a two-dimensional test on the unit square, $\Omega = [0, 1]^2$ with the asymmetric input function

$$u(x, y) \quad = \quad = \sum_{l=2}^{20} \sum_{k=2}^{20} \left(\frac{2}{k}\right)^{2.0} \left(\frac{2}{l}\right)^{0.3} \sin(k\pi x)\sin(l\pi y) \ , \qquad (15)$$

which is shown in Fig. 2a. The discretization consists of a 5×5 array of 9th-order spectral elements, for a total resolution of 46×46. Figure 2b shows the result of applying the filter (11-12) to (15), with $k_c = 12$, $m = 8$, and Krylov subspace dimension $j = 100$. The difference between the exact and Krylov-based filter outputs is shown in Fig. 2c. The computed two-dimensional transfer function, $|\hat{\hat{u}}_{kl}|/|\hat{u}_{kl}|$, is shown in Fig. 2d. The transfer function captures the correct modal decay rate down to the the level of the error resulting from the Krylov subspace approximation, that is, 10^{-3} in the present case.

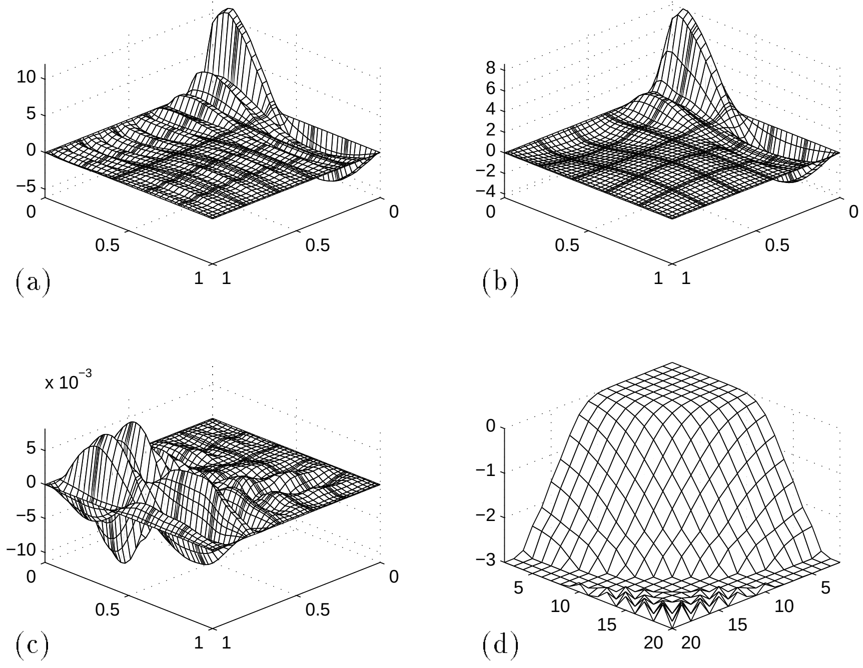


Figure 2: Input function (15) for filter test (a), output of Krylov-based filter (b), filtering error (c), and transfer function (d).

We demonstrate the versatility of the Krylov-based filter by applying it in the complex three-dimensional domain shown in Fig. 3. For this test, the input is the velocity field associated with the horseshoe vortex flow arising from a flat-plate boundary layer interacting with an end mounted cylinder of height $H = D/2$ and diameter D. The incoming boundary layer thickness is $\delta_{99} \approx D/21$, and the Reynolds number is $Re_D = U_\infty D/\nu = 3000$. The discretization consists of 168 spectral elements of order 9. The flow in the cylinder wake is severely underresolved, as witnessed by symmetry-plane contours of x-velocity shown in Fig. 3c (upper). Application of a 12th-order filter ($m = 6$) with filter width $\Delta = 0.01\,D$ and Krylov subspace dimension $j = 50$ effectively dampens the underresolved oscillatory components in the wake (Fig. 3c, lower) while allowing the well-resolved horseshoe vortex structures (a) to pass through the filter with little change (b).

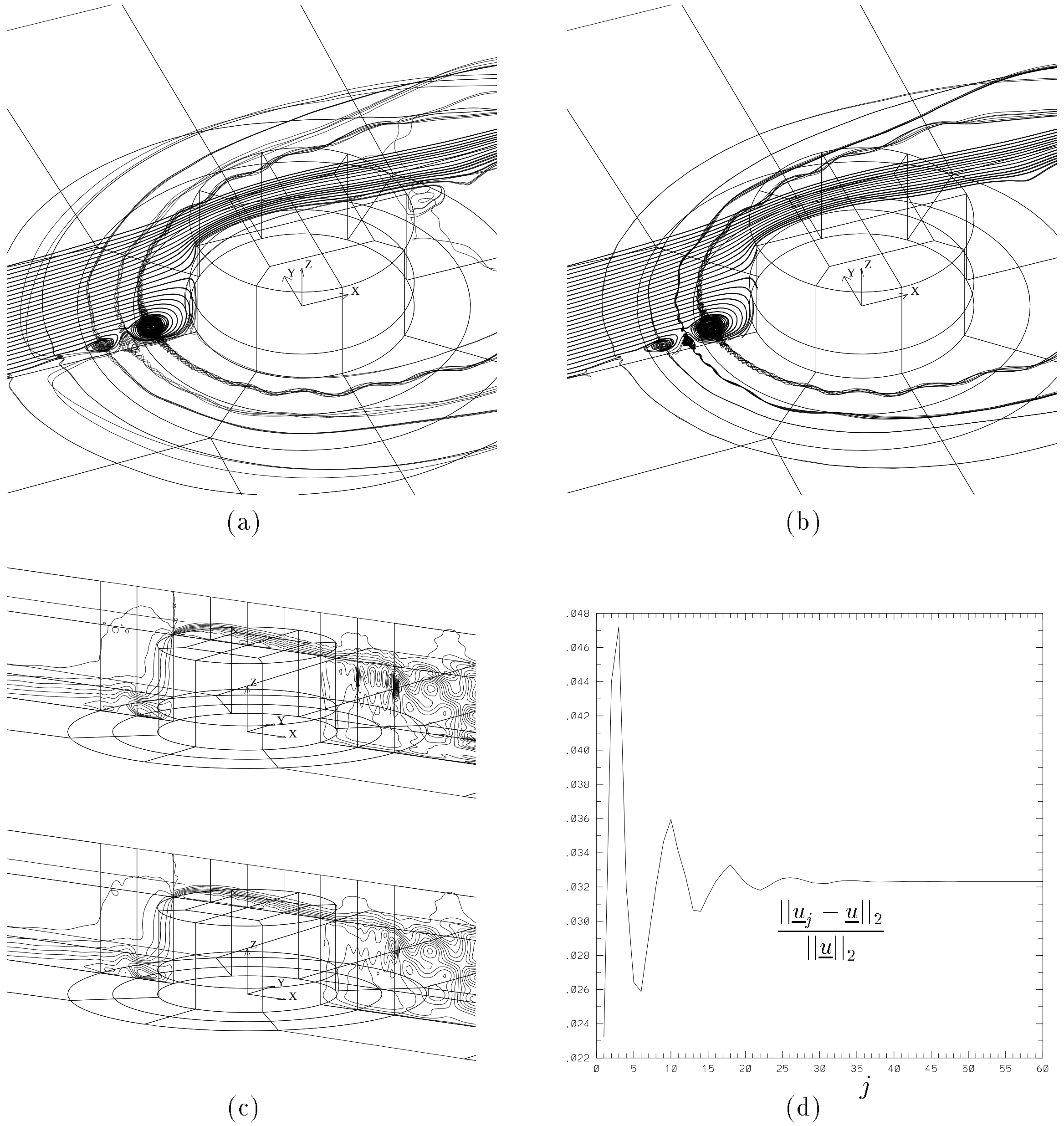
As presented, the Krylov-based filtering scheme requires three input parameters: the order, m, the filter-width, Δ, and the Krylov subspace dimension, j. The former are true parameters of the filter and not dependent upon the method used to implement it. The Krylov subspace dimension required to obtain an accurate approximation is a function of m, Δ, and $\underline{u}$. Unfortunately, it is difficult to monitor the residual, $||\underline{u} - P_m \bar{\underline{u}}_j||_2$ during the CG iteration because of the ill-conditioning associated with P_m. However, it is possible to monitor the convergence of $||\underline{u} - \bar{\underline{u}}_j||_2$ as shown, for example, in Fig. 3d. One can simply estimate the decay of the oscillatory envelope and stop the CG iteration when the deviation is within a desired tolerance of the estimated mean. From Fig. 3d we estimated that a Krylov subspace dimension of $j = 50$ would be sufficient to filter the field shown in Fig. 3a. Comparisons reveal that increasing j to 150 results in only a half of a percent further change in the filtered output; that is, the output is indeed reasonably well converged at $j = 50$.

It is apparent from the streamlines in (b) that the filtered field is no longer divergence free. This is to be expected because a divergence-free constraint was never enforced. However, if desired, the PDE-based approach employed here could readily be extended to accommodate such a constraint, yielding a Stokes-like system in which all three velocity components are computed simultaneously.

We conclude by noting that several techniques might be used to improve the efficiency of this general filtering approach. One important idea is that a given Krylov space, R, might be used to filter similar right-hand sides by simply changing the right-most term ($\underline{u}$) in either (10) or (12) (see, e.g., [2, 3]). In addition, we note that multi-level solvers might be used to speed the filtering process, following ideas outlined in [7].

Acknowledgments

This work was supported by the NSF under Grant ASC-9405403 and by the AFOSR under Grant F49620-95-1-0074. The work of J. Mullen was supported by a NASA student fellowship, Grant NGT-51029.



(a)

(b)

(c)

(d)

Figure 3: Example of Krylov-based filter applied in a complex geometry: horse-shoe vortices in unfiltered (a) and filtered (b) flow field; (c) centerplane contours of unfiltered (upper) and filtered (lower) streamwise velocity; (d) convergence of Krylov-based filter vs. subspace dimension j.

References

[1] A. A. Aldama, *Filtering Techniques for Turbulent Flow Simulation*, Springer-Verlag, 1990.

[2] T. F. Chan and W. L. Wan, "Analysis of projection methods for solving linear systems with multiple right-hand sides," Tech. Rep. CAM-94-26, UCLA, Dept. of Math., Los Angeles (1994).

[3] P. F. Fischer, "Projection techniques for iterative solution of $A\underline{x} = \underline{b}$ with successive right-hand sides," ICASE Tech. Rep. 93-90 (1993).

[4] P. F. Fischer, "An overlapping Schwarz method for spectral element solution of the incompressible Navier-Stokes equations," *J. Comp. Phys.* **133**, 84–101 (1997).

[5] M. Germano, U. Piomelli, P. Moin, and W. H. Cabot, "A dynamic subgrid-scale eddy viscosity model," *Phys. Fluids A* **3**, 1760 (1991).

[6] G. H. Golub and C. F. Van Loan, *Matrix Computations*, 2nd ed., Johns Hopkins University Press, Baltimore, 1989.

[7] C. C. Kuo, T. F. Chan, and C. Tong, "Multilevel filtering elliptic preconditioners," *SIAM Matrix Anal. Appl.* **11** 3, 403–429 (1990).

[8] D. K. Lilly, "A proposed modification of the Germano subgrid-scale closure method," *Phys. Fluids A* **4**, 633 (1992).

[9] Y. Maday, and A. T. Patera, "Spectral element methods for the Navier-Stokes equations", in *State of the Art Surveys in Computational Mechanics*, edited by A. K. Noor, ASME, New York, 71–143 (1989).

[10] W. H. Raymond, "Diffusion and numerical filters," *Monthly Weather Review* **122**, 757–761 (1994).

[11] H. A. Van der Vorst, "An iterative solution method for solving $f(A)x = b$ using Krylov subspace information obtained for the symmetric positive definite matrix A," *J. Comput. and Appl. Math.* **18**, 247–263 (1987).

[12] Y. Zhou, M. Hossain, and G. Vahala, "A critical look at the use of filters in large eddy simulation," *Phys. Letters A* **139**, 7, 330–332 (1989).