

A dynamic Smagorinsky model with dynamic determination of the filter width ratio ^a

Andrés E. Tejada-Martínez ^b and Kenneth E. Jansen
Department of Mechanical, Aerospace and Nuclear Engineering,
and Scientific Computation Research Center,
Rensselaer Polytechnic Institute, Troy, NY 12180

February 25, 2004

Abstract

Various low-pass, spatial test filters specific to dynamic model large-eddy simulation (LES) on finite element topologies are proposed and analyzed. A number of simulations of decaying isotropic turbulence are performed using the stabilized finite element method of Whiting and Jansen [1] with the purpose of understanding the dependence of the dynamic model on the test filter of choice and the elusive filter width ratio. From these numerical experiments, a key assumption is extracted leading to the derivation of a new dynamic model in which the sole model parameter, the filter width ratio, is computed dynamically. Traditionally, the dynamic model parameter has taken the filter width ratio as a constant. The new dynamic model is tested on LES of decaying isotropic turbulence on hexahedral, tetrahedral, and wedge topologies.

^aThis material is based upon work supported by the National Science Foundation under Grant No. 9985340. ^b Present address: Center for Coastal Physical Oceanography, Old Dominion University, Norfolk, VA 23529

1 Introduction

Large-eddy simulation (LES) is a technique for computation of turbulent flows where the large-scale component of the flow, carrying most of the energy, is resolved by the LES equations, and the small-scale unresolved residual motions are modeled. The LES equations are obtained from the Navier-Stokes equations by applying a primary spatial filter, which is assumed to damp scales on the order of the filter width (or the characteristic length scale of the filter). The constant coefficient Smagorinsky model, developed by Smagorinsky [2], is sometimes used to represent unresolved residual motions, which manifest in the LES equations in the form of an unknown residual stress tensor. Smagorinsky related the residual stress tensor to the strain rate tensor of the resolved motions through an eddy viscosity. In turn, the eddy viscosity is given by the norm of the resolved strain rate tensor multiplied by the square of a length scale characteristic of the unresolved residual motions. The characteristic length scale is set proportional to the primary filter width, wherein the coefficient of proportionality is denoted as the Smagorinsky constant.

A procedure developed by Germano *et al.* [3] and Lilly [4] is often used in which the constant coefficient model becomes a dynamic coefficient model where the Smagorinsky constant is no longer taken as constant but allowed to vary in space and time. The formulation of the dynamic coefficient model (or simply dynamic model) requires the sequential application of two well-characterized filters on the Navier-Stokes equations, namely the primary filter, mentioned earlier, and a test filter. We refer to well-characterized filters as filters with homogeneous shapes leading to computable widths through the use of known formulas such as the ones discussed by Lund [5]. Homogeneity of a filter implies that its shape is symmetric and constant in space, qualities that allow for commutation between the filter itself and differentiation. Knowledge of the primary and test filters is important when assigning the only adjustable parameter in the dynamic model. This parameter, denoted as the filter width ratio, is precisely the width of the filter resulting from sequential application of the primary and test filters divided by the width of the primary filter.

When numerically solving the LES equations, the discretization often assumes the role of the primary filter, herein referred to as the implicit numerical filter or the grid filter, while the test filter is applied explicitly. As a result of the former statement, the name “subgrid-scale motions” becomes appropriate when referring to “residual motions”. In general, the implicit numerical filter is not well-characterized [6], giving rise to an ambiguity when

assigning the filter width ratio for the dynamic model. In other words, not knowing the shape and thereby the width of the grid filter, renders the filter width ratio unknown. To bypass this ambiguity, if the discretization involves finite differences, the grid filter is often assumed to be twice the characteristic size of the grid, such as

$$\bar{\Delta} = 2(h_1 h_2 h_3)^{1/3} \quad \text{or} \quad \bar{\Delta} = 2(h_1^2 + h_2^2 + h_3^2)^{1/2}, \quad (1)$$

where h_i is the grid spacing in the i -th direction. There exists an equivalence between the exact derivative of a filtered variable and the central difference derivative of the unfiltered variable [7, 5, 8], leading to the conclusion that the implicit numerical filter behaves like the well-known box filter of width twice the grid size. In brief, after choosing the test filter and assuming a specific form for the implicit numerical filter, the filter width ratio can be computed [9]. However, in general an assumption regarding the form of the numerical filter can not be made, leaving one to estimate its width.

Explicit filtering of nonlinear terms in the filtered equations could be used to remove the indeterminacy of the numerical discretization. However, indeterminacy of the numerical discretizations has not been the main thrust behind proponents of explicit filtering of nonlinear terms. The main thrust has been to address concern about the fact that information from the smallest resolvable scales is used to model the unresolvable residual stress, and if such resolvable scales are affected by numerical error, so will the model. Explicit filtering of nonlinear terms can remove the smallest scales polluted by numerical error, thus helping the performance of the residual stress model. However, there is some benefit from including additional smaller scales in the simulations even if they are contaminated by numerical error [10]. It is suggested by Lund *et al.* [10] that this benefit is likely due to the fact that the error is pushed out to small scales which have relatively weak impact on low-order statistics. Instead of exploring explicit filtering of nonlinear terms in the filtered equations, we focus on how to avoid the indeterminacy of the numerical filter by formulating a new dynamic model, without directly impacting the equations. It should be reiterated that the procedure proposed herein to dynamically compute the filter width ratio is only necessary when explicit filtering is not carried out. It is only in this case that the grid filter width, the test filtered grid filter width and therefore the filter width ratio are not precisely defined.

Recent works [11, 12] have attempted to address the fact that the dynamic model formulation presupposes that the test filter is similar in shape to the primary filter. However, all hopes of knowing in advance the shape of

the test filter based on the shape of the primary filter are eliminated because the unknown grid filter acts as the primary filter. To that extent, the previous two references have investigated several test filters in an effort to find the best filter shape corresponding to best simulation results. Unfortunately, in these studies favorable results have been incorrectly attributed to certain test filter shapes due to an inaccurate determination of test filter widths. As is shown by Lund [5] and the present authors [13], dynamic model results of decaying isotropic turbulence can be independent of the test filter shape as long as the width of the test filter is accurately and consistently computed. Furthermore, although results may be independent of the test filter shape, they may not be optimal [13]. Additionally, results can be sensitive to the filter width ratio as will be shown later and as is shown by Chester *et al.* [14], thus motivating the need for its accurate determination. This sensitivity issue deserves special attention specifically for cases when LES grids are coarsened, leading to stronger subgrid-scale model dissipations.

Certainly, the expressions in (1) could be thought of as a measure of the resolution threshold of a discretization. In wall-bounded flows, the resolution level of a discretization is increased near the walls by refining the grid so as to capture strong gradients present there. A change in the resolution level will lead to a change in the dynamic model's filter width ratio, assuming that the resolution is proportional to the width of the grid filter. While it would be desirable to have the filter width ratio solely dependent on the numerical scheme, the grid employed and the test filter used in the dynamic model, this is not always the case. Many numerical schemes, like the one used herein, alter their character based on flow, hence the filter width ratio should evolve with the flow as well. For example, in turbulent channel flow, our experiences have shown us that stabilized finite element methods increase numerical energy dissipation in regions where the flow is more unstable due to strong advection. In the case of decaying isotropic turbulence, the initial conditions are not in complete satisfaction of the discretized equations (initial conditions are obtained from matching experimental spectrum and thus are solutions to spectral discretization), thus a transient is expected throughout the simulation. In summary, the numerical method and the flow are inevitably intertwined, thus one can expect for the numerical scheme and its dependents, such as the filter width ratio, to evolve with the flow and vice versa. The previous notion is realized in the present work, in which a time- and space- dependent (dynamic) expression for the filter width ratio is derived and shown to lead to good dynamic model LES results for decay of isotropic turbulence. The dynamic formulation of the filter width ratio (herein referred to as DFWR) is based on results recently summarized by

the authors [13], thus we will first review these results.

2 The dynamic subgrid-scale model

Ideally, the filtered Navier-Stokes equations are obtained by applying a primary homogeneous spatial filter (denoted with an over-bar) generating a closure problem in the form of an unknown residual stress tensor,

$$\tau_{ij} = \overline{u_i u_j} - \bar{u}_i \bar{u}_j, \quad (2)$$

reflecting the effect of the filtered-out residual scales on the resolved scales. The trace of the residual stress is absorbed into the filtered pressure and the deviatoric portion is modeled.

Recall that in a large-eddy simulation, the discretization acts as the filter. Common references to the filter as the “grid filter” and to the unknown residual stress tensor as the “subgrid-scale” stress tensor lie within the previous interpretation of the primary filter. It should be emphasized that the subgrid-scale stress stems from a closure problem and not from the discretization’s inability to represent the smallest scales in the flow. Recent works [15, 16, 17, 18] have focused in modeling the subgrid-scale stress from the latter point of view.

The deviatoric or traceless component of the subgrid-scale stress,

$$\tau_{ij}^d = \tau_{ij} - \frac{1}{3} \tau_{kk} \delta_{ij} = (\overline{u_i u_j} - \bar{u}_i \bar{u}_j) - \frac{1}{3} (\overline{u_k u_k} - \bar{u}_k \bar{u}_k) \delta_{ij}, \quad (3)$$

appearing on the right hand side of the filtered equations, is expressed through the constant coefficient Smagorinsky model [2] as

$$\tau_{ij}^d = -2 \underbrace{(C_s \bar{\Delta})^2 |\bar{S}|}_{\text{eddy viscosity}} \bar{S}_{ij}, \quad (4)$$

where $\bar{\Delta}$ is the width of the grid filter, C_s is the Smagorinsky constant, $\bar{S}_{ij} = (\partial \bar{u}_i / \partial x_j + \partial \bar{u}_j / \partial x_i) / 2$ is the filtered strain-rate tensor, and $|\bar{S}| = \sqrt{(2\bar{S}_{ij}\bar{S}_{ij})}$ is the norm of the filtered strain-rate tensor. Note that $\bar{\Delta}$, the width of the grid filter, is an unknown. Often, the grid filter width is taken proportional to the smallest resolvable length scale of the discretization, which in some instances can be given approximately by some definition of the grid size such as the ones in (1).

To obtain the dynamic subgrid-scale model, a second filter, referred to as the test filter (denoted with an over-hat), is applied to the once-filtered

Navier-Stokes equations. The test filter generates another unknown residual stress tensor denoted as the subtest-scale stress and defined as

$$T_{ij} = \widehat{\bar{u}_i \bar{u}_j} - \hat{\bar{u}}_i \hat{\bar{u}}_j \quad (5)$$

Next, the Germano identity between the grid and test filtered fields [3],

$$L_{ij} = T_{ij} - \hat{\tau}_{ij}, \quad (6)$$

is used to dynamically determine $(C_s \Delta)^2$ in the Smagorinsky model. The usefulness of the tensor in (6) lies in that it can be expressed in terms of the filtered or resolved velocity \bar{u}_i as well as in terms of the Smagorinsky model. In terms of the resolved velocity, L_{ij} becomes

$$L_{ij} = \widehat{\bar{u}_i \bar{u}_j} - \hat{\bar{u}}_i \hat{\bar{u}}_j. \quad (7)$$

In terms of the Smagorinsky model, the deviatoric portion of L_{ij} can be expressed by test filtering the expression in (4),

$$\widehat{\tau}_{ij}^d = -2(C_s \bar{\Delta})^2 |\widehat{\bar{S}}| \widehat{\bar{S}}_{ij}, \quad (8)$$

and by modeling the deviatoric portion of the subtest-scale stress as

$$T_{ij}^d = -2(C_s \hat{\Delta})^2 |\hat{\bar{S}}| \hat{\bar{S}}_{ij} \quad (9)$$

under the assumption of scale-invariance [19]. In (9) the strain-rate tensor, $\hat{\bar{S}}_{ij}$, and its norm, $|\hat{\bar{S}}|$, are based on the double-filtered velocity $\hat{\bar{u}}_i$. The bar-hat notation denotes application of the filter (of width $\hat{\Delta}$) resulting from sequential applications of the grid filter and the test filter. Although the test filter appears explicitly in the computations and its characteristics can be well-defined, the grid filter does not appear explicitly and can not be well-characterized. Hence, the width of the test filter, $\hat{\Delta}$ can be obtained using known definitions, however, the width $\hat{\Delta}$ is not readily obtainable.

The model coefficient $(C_s \bar{\Delta})^2$ is dynamically computed by minimizing the square of the difference $(D_{ij} D_{ij})$ between the modeled L_{ij}^d and the resolved L_{ij}^d with respect to C_s [4], where the difference is given as

$$D_{ij} = L_{ij}^d - 2(C_s \bar{\Delta})^2 M_{ij}, \quad (10)$$

and

$$M_{ij} = |\widehat{\bar{S}}| \widehat{\bar{S}}_{ij} - \alpha |\hat{\bar{S}}| \hat{\bar{S}}_{ij}. \quad (11)$$

We refer to $\alpha = (\hat{\Delta}/\bar{\Delta})^2$ as the square of the filter width ratio. The minimization procedure, leads to

$$(C_s \bar{\Delta})^2 = \frac{1}{2} \frac{\langle L_{ij}^d M_{ij} \rangle}{\langle M_{kl} M_{kl} \rangle} = \frac{1}{2} \frac{\langle L_{ij} M_{ij} \rangle}{\langle M_{kl} M_{kl} \rangle}, \quad (12)$$

where the second equality holds because M_{ij} is trace-free. To prevent the previous expression from becoming negative giving rise to a negative eddy viscosity, the numerator and denominator are averaged over homogeneous directions. Averaging is denoted by brackets.

The rest of this article focuses on the accurate calculation of the filter width ratio squared (α), since it has been observed that, for relatively coarse simulations, results tend to depend on this ratio.

3 Test filters on finite elements and their effect on the dynamic model

In the present work we use a stabilized finite element formulation, namely the streamline upwind Petrov-Galerkin (SUPG) formulation [1, 20] with piecewise continuous tri-linear Lagrange basis functions. Within the finite element framework, a generalized box filtered function at a vertex 0 (located at \mathbf{x}_0) on any grid topology is taken as

$$\hat{f}(\mathbf{x}_0) = \int_{\mathcal{R}^d} G(\mathbf{x}_0, \mathbf{y}) f(\mathbf{y}) d\mathbf{y}, \quad (13)$$

where integration is over the entire real space [21, 13]. However, due to the small compact support of $G(\mathbf{x}, \mathbf{y})$, the generalized box filter kernel, or for that matter any other kernel commonly used in LES, the region of integration is greatly reduced. In the case of (13), the region of integration is reduced to the union of elements which share vertex 0, denoted as Ω_0 . The generalized box filter kernel at vertex 0, $G(\mathbf{x}_0, \mathbf{y})$, can be written as

$$G(\mathbf{x}_0, \mathbf{y}) = \begin{cases} \frac{1}{\text{volume}(\Omega_0)} & \text{if } \mathbf{y} \text{ is in } \Omega_0 \\ 0 & \text{otherwise.} \end{cases} \quad (14)$$

A Gaussian quadrature rule approximation of this generalized box test filter on unstructured (or irregularly connected) topologies has been shown to work well for the dynamic model [21]. However, an effort to understand the interplay between the test filter, the dynamic model and the numerical

method was not made until recently [13]. In the previous reference, the authors chose to restrict quadrature rule approximations of this test filter to regularly connected topologies for the purpose of defining filter widths, thereby enabling a clear understanding of the relationship between the test filter, the model and the numerical method. Such an understanding would be of great importance in truly judging the performance of dynamic model LES in the more general settings of a complex geometry. Although in the present work we still focus on test filtering on regularly connected geometry, our results could certainly be extended to general geometries via several new filtering procedures [22, 23]. In the following sub-sections we give a detailed recompilation of and expound on the test filters resulting from Gauss quadrature approximations of (14). In later sections we use our understanding of the interplay between test filters, the model and the method to dynamically determine the model’s filter width ratio.

3.1 Discrete test filters on finite elements

Discrete test filter widths can be computed via the second moment of the filter kernel or via the transfer function (the Fourier transform multiplied by $\sqrt{2\pi}$) of the filter kernel. For the majority of our simulations, we will use the latter approach, since it lends well to the three grid topologies considered: regularly connected hexahedral (“hex”), tetrahedral (“tet”), and wedge topologies. The former approach is advantageous for filters arising from products of one-dimensional filters such as is the case for our filters on hexes, but not on tets nor wedges. First, we describe the filters and their transfer functions before using the latter to compute filter widths.

In what follows, we will be interested in computing the filter width of filters derived from Gaussian quadrature. A few points of clarification are necessary given that this approach is similar to but not the same as traditional filtering used in LES. First, as our quadrature points will be associated with Gaussian quadrature, they will not in general be equispaced. This precludes the use of Fast Fourier Transforms but as the calculation is only part of the determination of the filter width and not part of our actual flow computation, this is of no consequence. We simply use the more general notion of a Fourier transform to obtain the transfer function from non-equispaced points. A second point that readers may be concerned with pertains to the number of quadrature points (function evaluations) used to obtain the filtered value (and therefore the transfer function). For a given filter shape G if one integrates (13) exactly there will be a single transfer function and thus a single filter width. We are intentionally considering dif-

ferent quadrature rules (and thus different levels of approximation of (13)) to create different transfer functions and thus different filter widths. This is similar to the choice of using Simpson’s rule or trapezoidal rule in the finite difference literature. As the quadrature rule is increased, equation (13) will eventually be exactly integrated (for rational polynomial f and simple shapes G). Further increase in the quadrature will result in the same exact filter, the same transfer function and thus the same filter width.

3.1.1 One-dimensional test filters

We begin with a one-dimensional description of the different filters arising from quadrature approximations of (13). We point the reader to the sketch in Figure 1 as an example. Using one-point quadrature (one function evaluation in the middle of the “cell” which in this case is a line) to evaluate the filtering operation in (13), we obtain that a filtered function at vertex 0 (located at $x = x_0$) can be written as

$$\frac{1}{2h} \int_{x_{-1}}^{x_1} f(y) dy \approx \hat{f}_0 = \frac{1}{2}(f_{-1/2} + f_{1/2}), \quad (\text{S1}) \quad (15)$$

where $f_0 = f(x_0)$, $f_{-1/2} = f(x_{-1/2})$, $f_{1/2} = f(x_{1/2})$, and S1 stands for standard filter with one-point quadrature. The transfer function induced by this filtering operation is

$$G(kh) = \cos(kx_{1/2}), \quad (16)$$

where k is the wavenumber and h is the constant spacing between vertices. Note that the previous expression is the Fourier transform of the discrete kernel induced by the operation in (15) scaled by 2π . The Fourier transform may possess a real and an imaginary component. However, the symmetry between quadrature point locations $x = x_{-1/2}$ and $x = x_{1/2}$ (i.e. $x_{-1/2} = -x_{1/2}$) yields a purely real Fourier transform (or transfer function), from which a filter width can be obtained. All discrete filters studied here will possess this important symmetry characteristic. Filter kernels with Fourier transforms having a non-zero imaginary component do not act as true filters and ideally should be avoided. The reason for this is that in addition to damping the amplitude components of solution fields (which is what a true filter should do), kernels with imaginary Fourier transform components act on the phase components as well.

The above procedure, produced \hat{f} at all of the vertices or nodes of our discretization. It is straightforward to use these nodal values to construct a

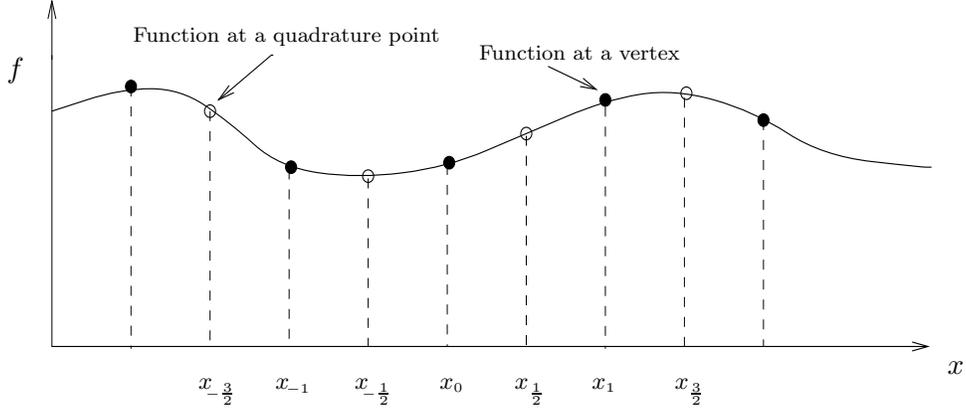


Figure 1: Sketch of function evaluations for filter S1 and W1. Each quadrature point is located at the middle of an element (the interval between two vertices). The spacing between vertices is h .

piecewise continuous linear function that can be filtered again with the box filter of (13) and (14) using either a one-point or a two-point quadrature, which for this one-dimensional case are both equivalent to the trapezoidal integration (due to the linearity of the function). We will refer to this second filter as the trapezoidal filter. Hence, we obtain the following filtered function at vertex 0:

$$\begin{aligned} \tilde{f}_0 &= \frac{1}{4}\hat{f}_{-1} + \frac{1}{2}\hat{f}_0 + \frac{1}{4}\hat{f}_1 \\ &= \frac{1}{8}f_{-3/2} + \frac{3}{8}f_{-1/2} + \frac{3}{8}f_{1/2} + \frac{1}{8}f_{3/2}, \end{aligned} \quad (\text{W1}) \quad (17)$$

where W1 is short notation for wide filter with one-point quadrature. Note that filter W1 uses information over a wider region than filter S1, hence the name wide filter is appropriate. The transfer function induced by this filtering operation is

$$G(kh) = \frac{3}{4} \cos(kx_{1/2}) + \frac{1}{4} \cos(kx_{3/2}). \quad (18)$$

Looking at Figure 2, we can think of performing the previous two filtering operations with two-point quadrature instead of one-point quadrature. If so, we obtain the following filtering operations:

$$\hat{f}_0 = \frac{1}{4}(f_{-2/3} + f_{-1/3} + f_{1/3} + f_{2/3}), \quad (\text{S2}) \quad \text{and} \quad (19)$$

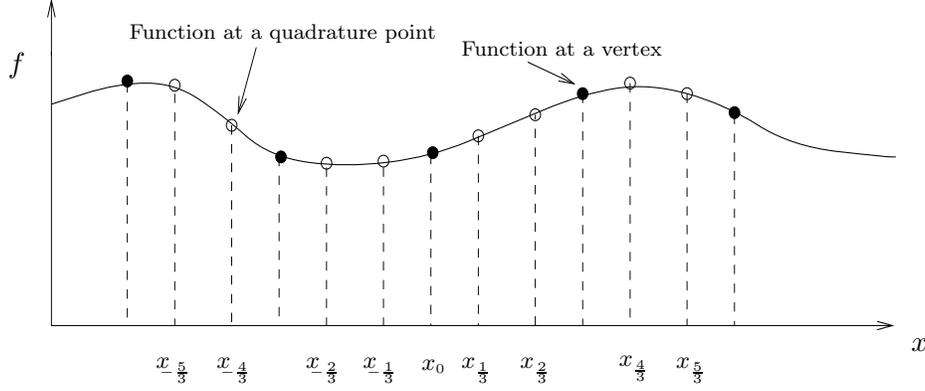


Figure 2: Sketch of function evaluations for filter S2 and W2. The spacing between vertices is h . Each quadrature point is located at a distance of αh away from its nearest vertex, where $\alpha = 0.211324865$.

$$\tilde{f}_0 = \frac{1}{16}(f_{-5/3} + f_{-4/3}) + \frac{3}{16}(f_{-2/3} + f_{-1/3}) + \frac{3}{16}(f_{1/3} + f_{2/3}) + \frac{1}{16}(f_{4/3} + f_{5/3}). \quad (\text{W2}) \quad (20)$$

The transfer functions corresponding to filters S2 and W2 are

$$G(kh) = \frac{1}{2}[\cos(kx_{1/3}) + \cos(kx_{2/3})] \quad \text{and} \quad (21)$$

$$G(kh) = \frac{1}{8}[\cos(kx_{5/3}) + \cos(kx_{4/3})] + \frac{3}{8}[\cos(kx_{2/3}) + \cos(kx_{1/3})], \quad (22)$$

respectively.

The transfer functions for the one-dimensional filters S1, S2, W1, and W2 are plotted in Figure 3 along with the transfer functions for the box filter approximated using the trapezoidal and Simpson's integration. The previous two filter approximations, which are often employed in finite difference discretizations, lead to

$$\hat{f}_0 = \frac{1}{4}f_{-1} + \frac{1}{2}f_0 + \frac{1}{4}f_1 \quad (\text{FD1}) \quad \text{and} \quad (23)$$

$$\hat{f}_0 = \frac{1}{6}f_{-1} + \frac{2}{3}f_0 + \frac{1}{6}f_1, \quad (\text{FD2}) \quad (24)$$

respectively. The transfer function for filter FD1 is

$$G(kh) = \frac{1}{2}[1 + \cos(kh)] \quad (25)$$

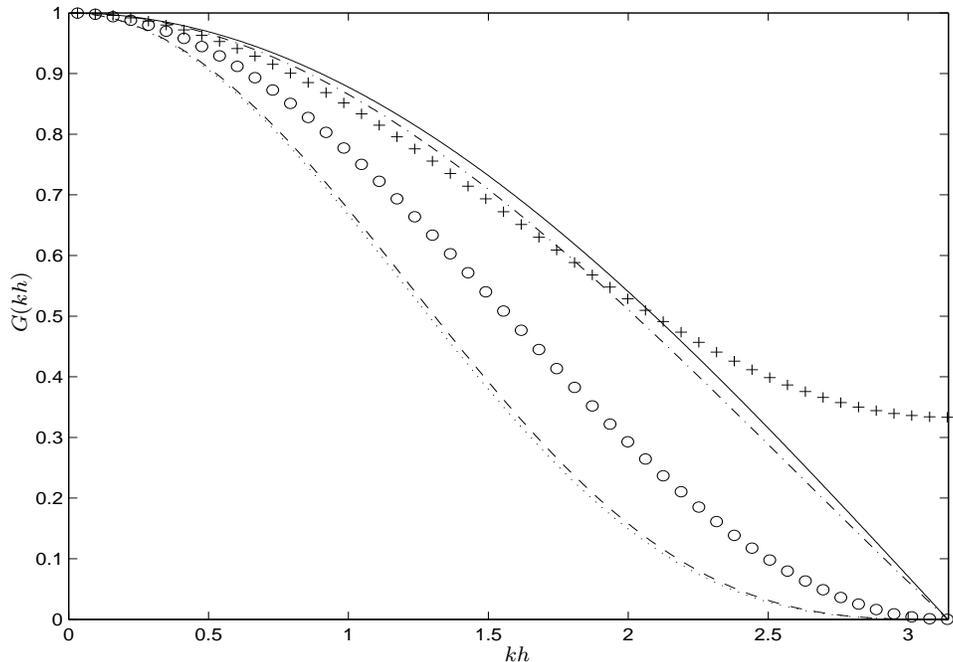


Figure 3: Transfer functions for one-dimensional filters. — : filter S1; ---- : filter W1; - · - : filter S2; ····· : filter W2; ○ : filter FD1; + : filter FD2.

and for filter FD2 is

$$G(kh) = \frac{2}{3} + \frac{1}{3} \cos(kh). \quad (26)$$

The main difference between the “finite difference-based” filters (FD1 and FD2) and the “finite element-based filters” (S1, S2, W1, and W2) is that the latter group takes into account the function space underlying the finite element method (FEM). The importance of this comes into play when computing the dynamic model. For example, if the FEM uses continuous piecewise linear basis functions, the finite element-based filters act to filter continuous piecewise quadratic functions such as the products between \bar{u}_i and \bar{u}_j . To see this, the reader is directed to the term $\widehat{\bar{u}_i \bar{u}_j}$ appearing in the dynamic model given by equations (6), (11), and (12). While there is some appeal to expressing (15), (17), (19), and (20) in terms of nodal values, this is only practical for the filtering of linear functions which is

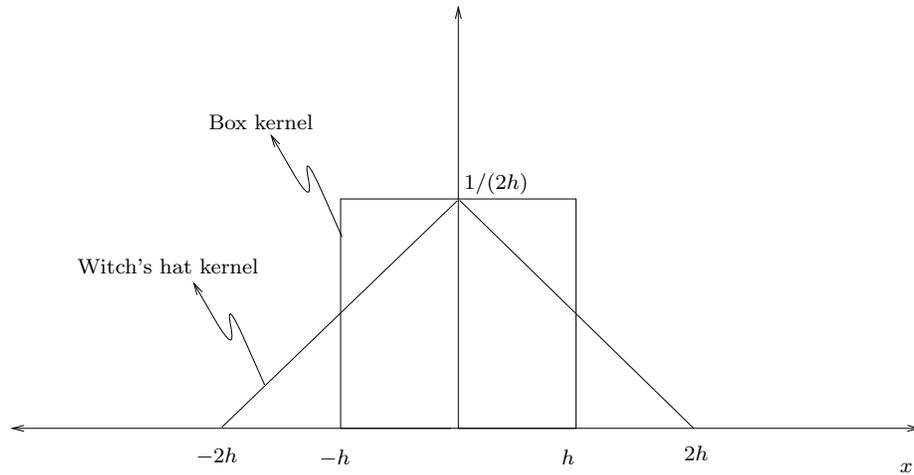


Figure 4: Box and witch’s hat filter kernels. Length h corresponds to the constant spacing between vertices in Figures 1 and 2.

insufficient for the filtering needs of the dynamic model as mentioned above. For this reason, we have computed the filter transfer functions using the quadrature sampled points directly, making them applicable to both linear and quadratic (products of linears) choices of f .

In real space, filters S1, S2, FD1, and FD2 are discrete approximations of the one-dimensional box filter kernel shown in Figure 4. Filters W1 and W2 can be shown to be discrete approximations of the one-dimensional “witch’s hat” filter with kernel shown also in Figure 4.

By construction, the wide filters are approximations of the filter resulting from two successive applications of the box filter. It is well known [24] that the Fourier transform of a filtered function is simply the product between the filter transfer function and the Fourier transform of the original function. Hence, twice application of the box filter in Fourier space is given by the product between the square of the box filter transfer function and the Fourier transform of the original function. It follows that the transfer function of the filter resulting from sequential applications of the box filter is the square of the box filter transfer function. Application of the inverse Fourier transform to the square of the box filter transfer function yields the witch’s hat kernel of Figure 4.

In conclusion, each wide filter (W1 or W2), denoted with a over-hat/over-tilde ($\hat{\tilde{\cdot}}$), can be obtained from a standard filter (S1 or S2) by sequential

applications of the standard filter, denoted with an over-hat ($\hat{\cdot}$), and the trapezoidal filter, denoted with an over-tilde ($\tilde{\cdot}$).

3.1.2 Multi-dimensional test filters

As mentioned earlier, the finite element-based filters on regularly connected quadrilateral (two-dimensional) or hexahedral (three-dimensional) elements reduce to products of their one-dimensional counterparts in the x -, y -, and/or z -directions. However, this is not the case for triangular (two-dimensional) or tetrahedral (three-dimensional) topologies. In general, multi-dimensional finite element-based filters S1, S2, W1, and W2 can be constructed in the same fashion as their one-dimensional counterparts described in the previous sub-section. Furthermore, any multi-dimensional finite element-based test filtered function (with test filter S1, S2, W1, or W2) at vertex location $\mathbf{x} = \mathbf{x}_0$ on two- or three-dimensional regularly connected topologies takes the form

$$\hat{f}(\mathbf{x}_0) = \sum_{i=-N/2}^{N/2} W_i f(\mathbf{x}_i), \quad (27)$$

where $f(\mathbf{x}_i)$ is $f(\mathbf{x})$ evaluated at the i^{th} quadrature point and the filter weights, W_i , sum to one. The symmetry between quadrature points ($\mathbf{x}_i = \mathbf{x}_{-i}$) due to the regularity of the topologies leads to the following purely real transfer functions:

$$G(\mathbf{k}^*) = 2 \sum_{i=1}^{N/2} W_i \cos(\mathbf{k} \cdot \mathbf{x}_i), \quad (28)$$

where the non-dimensional wavenumber vector \mathbf{k}^* is given as $(k_x h_x, k_y h_y, k_z h_z)$ with $k_z h_z = 0$ for two-dimensional topologies. The grid spacings (h_x, h_y, h_z) are implicit in the quadrature point locations \mathbf{x}_i . Length h_i is the constant spacing between grid points in the i^{th} direction and k_i is the wavenumber in that direction. For purposes of computing filter widths, the absolute value of \mathbf{k}^* will be referred to as the radial wavenumber, k_r . As an example of finite element-based test filters in two dimensions, consider test filtering a function on the regularly connected triangular and quadrilateral grids shown in Figures 5a and 5b using one-point quadrature. Such approximations lead to filter S1 on triangles and filter S1 on quadrilaterals. Filter S1 on triangles and filter S1 on quadrilaterals and their transfer function on these two-dimensional grids reduce to (27) and (28) with $(N = 6, W_i = 1/6)$ and $(N = 4, W_i = 1/4)$, respectively, for all i . The transfer functions for these two cases with $h_x = h_y$ are shown in Figure 6. Notice the direction bias

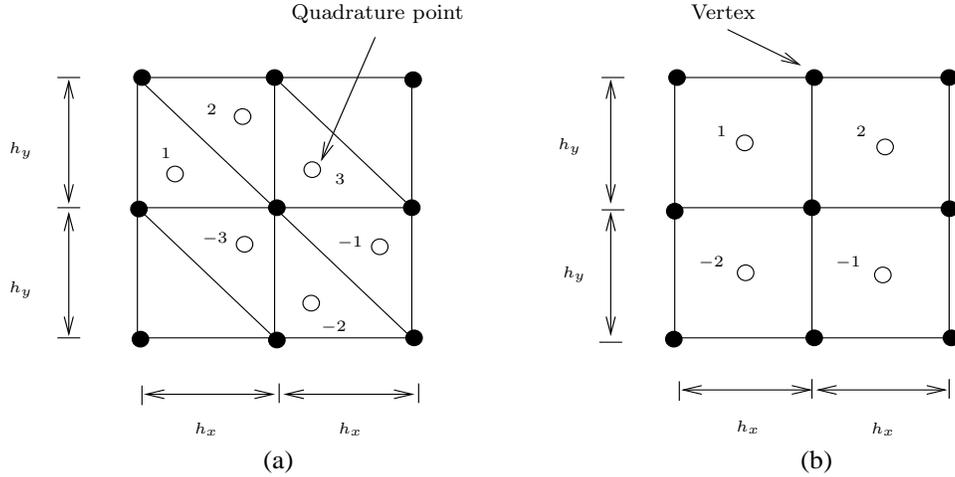


Figure 5: Sketch of regularly connected triangular and rectangular grids. Each quadrature point is located at the centroid of its element. Quadrature point locations are symmetric ($\mathbf{x}_i = -\mathbf{x}_{-i}$).

introduced by the transfer function on triangles, as it filters more in one direction than in others.

Transfer functions of two-dimensional filters can be plotted as closed curve contours such as the ones in Figure 6. Transfer functions of three-dimensional filters can be plotted as closed isosurfaces. In the next three figures we show examples of such contours for filters S1, S2, W1, and W2 on regularly connected grids composed of hexahedral, tetrahedral and wedge elements with equal spacing in the x , y , and z -directions. Similar to filters on triangles, filters on tets and wedges introduce a directional bias by filtering in one direction more than in others, as can be seen from Figures 8 and 9. Notice the different orientation bias between the filters on tets and wedges in Figure 8. Only looking at the transfer function contours in Figures 7, 8 and 9, one might be led to think that there is not much difference between wide and standard filters. To appreciate the vast difference, the reader is directed to Tables I, II, and III highlighting the disparity existing between wide and standard filters in their widths and in their number of quadrature point (or function) evaluations.

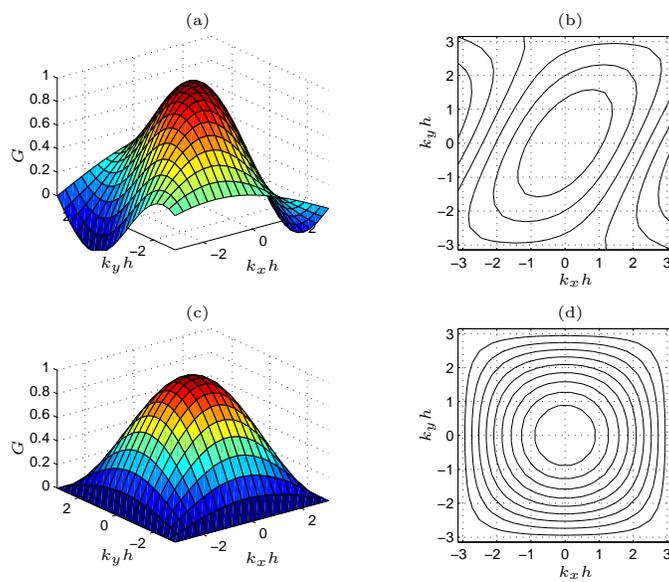


Figure 6: Transfer functions of finite element-based filters on triangles and quadrilaterals with one-point quadrature approximation (a) Transfer function of filter S1 on triangles. (b) Contours of (a). (c) Transfer function of filter S1 on quadrilaterals. (d) Contours of (c).

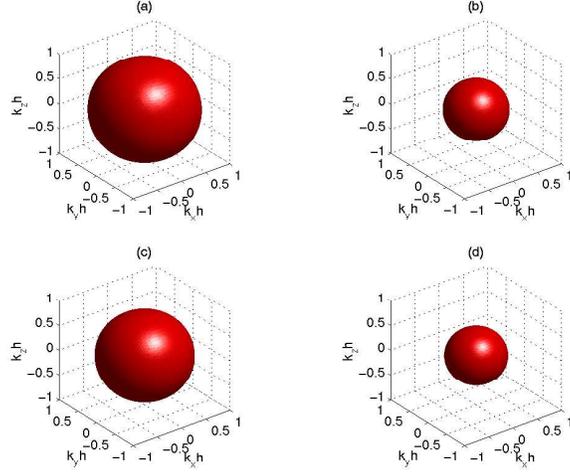


Figure 7: Finite element-based filter transfer function isosurfaces defined by $G(\mathbf{k}^*) = 0.75$ on hexahedral elements. (a) Isosurface for standard filter S1. (b) Isosurface for wide filter W1. (c) Isosurface for standard filter S2. (d) Isosurface for wide filter W2.

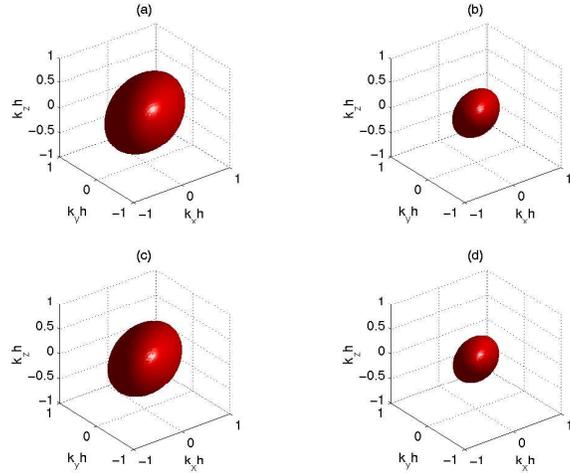


Figure 8: Finite element-based filter transfer function isosurfaces defined by $G(\mathbf{k}^*) = 0.85$ on tetrahedral elements. (a) Isosurface for standard filter S1. (b) Isosurface for wide filter W1. (c) Isosurface for standard filter S2. (d) Isosurface for wide filter W2.

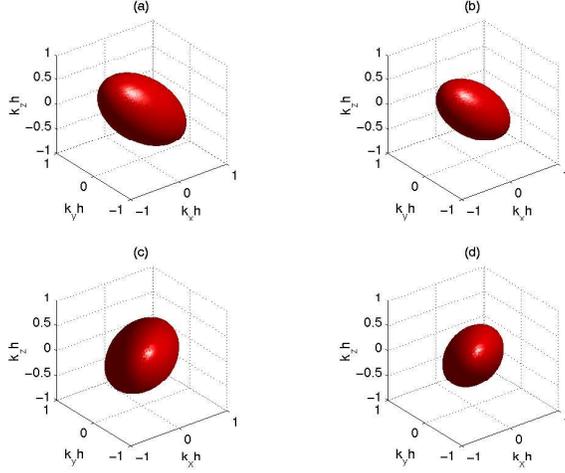


Figure 9: Comparison of finite element-based filter transfer function contours defined by $G(\mathbf{k}^*) = 0.85$ on wedge and tet elements. (a) Isosurface for standard filter S2 on wedges. (b) Isosurface for wide filter W2 on wedges. (c) Isosurface for standard filter S2 on tets. (d) Isosurface for wide filter W2 on tets.

3.2 Test filter widths

Widths of one-dimensional discrete test filters can be obtained through their transfer function [5]. For example, the width of any of the filters with transfer function shown in Figure 3 can be computed from the non-dimensional wavenumber kh , for which its transfer function equals 0.5 ($G(kh) = 0.5$). Here, we generalize this approach to evaluate the widths of the three-dimensional finite element-based filters. We define the filter width for such filters as the size of scales corresponding to the average radial wavenumber k_r^* , for which the filter transfer function assumes a specified value between 0 and 1. We proceed by setting the filter width as $\hat{\Delta} = \pi/k_r^*$. Clearly, the definition of the test filter width is not unique. However, for purposes of dynamic estimation of the filter width ratio in the model, the specified value of $G(\mathbf{k}^*)$ will be shown later to be unimportant. Instead, the ratio between widths of the wide and standard test filters will be shown to be important. In Tables I, II, and III we tabulate widths of finite element-based filters based on different values of the average radial wavenumber. Here we present the square of test filter widths, since this is a more appropriate representation in light of the dynamic model parameter $\alpha = (\hat{\Delta}/\bar{\Delta})^2$.

Filter	$G(\mathbf{k}^*) = 0.5$	$G(\mathbf{k}^*) = 0.65$	$G(\mathbf{k}^*) = 0.75$	$G(\mathbf{k}^*) = 0.85$
S1	3.28	5.04	7.36	12.7
S2	4.20	6.50	9.57	16.6
W1	9.03	14.4	21.1	37.0
W2	9.97	15.8	23.4	41.5

Table 1: Ratio of the filter width to mesh size squared, $((\hat{\Delta}/h)^2)$, for various choices of the radial wave number and filter type on regularly connected hexes with constant grid size h in the x -, y -, and z -directions.

Filter	$G(\mathbf{k}^*) = 0.5$	$G(\mathbf{k}^*) = 0.65$	$G(\mathbf{k}^*) = 0.75$	$G(\mathbf{k}^*) = 0.85$
S1	1.88	2.86	4.15	7.17
S2	2.15	3.33	4.90	8.51
W1	5.09	8.06	12.0	21.0
W2	5.40	8.57	12.7	22.4

Table 2: Same as Table I but for tetrahedra instead of hexahedra.

Filter	$(\hat{\Delta}/h)^2$
S2	10.8
W2	27.7

Table 3: Same as Table I but for wedges instead of hexahedra.

3.3 Test filters and the dynamic model: Isotropic turbulence

In this section we will use the three-dimensional finite element-based test filters previously described as test filters for the dynamic model in large-eddy simulations of decaying isotropic turbulence. The initial condition for these simulations is chosen such that its energy spectrum matches filtered experimental energy spectrum [25], using a procedure well described by the authors [18]. The filtered energy spectrum at three non-dimensional time stations, denoted as t_{42} , t_{98} and t_{171} is available from the experiments. In summary, the initial condition of the simulations matches experimental data at t_{42} . After the simulation is performed results are compared to the experimental energy spectrum at t_{98} and t_{171} . This problem is statistically homogeneous in all directions and has zero mean velocity. Thus a cube with sides of length 2π , periodic in all three directions can be taken as the domain. The domain is represented by 33 evenly spaced points in each of the three directions, resulting in a constant grid size $h = 2\pi/32$. Three topologies for this discretization will be explored: hexahedral, tetrahedral, and wedge elements. More specifically, the tetrahedral grid is the hexahedral grid with each hexahedral element split into six tetrahedral elements. The wedge grid is the hexahedral grid with each hexahedral element split into 2 wedge elements. In the three cases we have regularly connected grids on which test filter widths can be computed following the procedure outlined in the previous sub-section.

Of particular interest in this sub-section is to understand the role of the square of the filter width ratio, α , and the test filter appearing in the dynamic model in (12). An understanding of the role that these two play is essential to large-eddy simulations, especially those performed on coarse grids on which the dynamic model strongly affects the results. To demonstrate this, Figure 10 shows isotropic turbulence dynamic model results for different values of α at t_{98} and t_{171} , along with the energy spectrum of the initial condition of all the simulations matching the experimental spectrum at t_{42} . Notice the strong effect of this parameter on the energy spectrum in the inertial region, thus its accurate prediction is of primary importance. Furthermore, the simulations with $\alpha = 2$ and $\alpha = 3$ have been extended to t_{171} (the others were stopped at t_{98}). The $\alpha = 3$ simulation uses the customary estimate $\alpha \approx (\hat{\Delta}/h)^2 = 3$, which is based on the assumptions $\bar{\Delta} \approx h$ and $\hat{\Delta} \approx \hat{\Delta}$ and for which the test filter width $\hat{\Delta}$ has been computed based on the second moment of the test filter kernel used, filter S1. Clearly, this simulation shows that for our numerical method such an estimate of α is not appropriate as it leads to an over-prediction of energy at t_{98} and t_{171} .

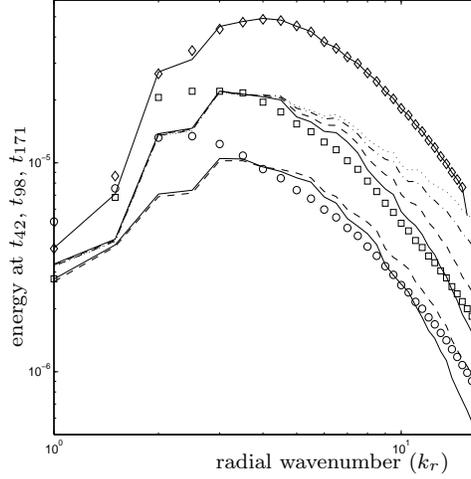


Figure 10: Effect of the filter width ratio squared (α) on dynamic model results of isotropic turbulence. Simulations were performed with filter S1 on hexes. \diamond : data at t_{42} ; \square : data at t_{98} ; \circ : data at t_{171} ; —: initial condition matching the data; —: simulation with $\alpha = 2$; ----: simulation with $\alpha = 3$; - · -: simulation with $\alpha = 4$; ·····: simulation with $\alpha = 5$.

A second customary estimate is to take $\bar{\Delta} \approx h$ and $\hat{\Delta}^2 \approx \hat{\Delta}^2 + h^2$. In this case, $\alpha \approx (\hat{\Delta}^2 + h^2)/h^2 = 4$. However, similar to $\alpha \approx 3$, the estimate $\alpha \approx 4$ leads to an over-prediction of energy at t_{98} . At first sight, the simulation with $\alpha = 2$ might seem optimal as it is a best fit of the t_{98} experimental spectrum extending through the highest resolved wavenumbers. However, the reader is reminded that in an LES we aim at good agreement in the inertial range prior to the highest resolved wavenumbers, as the latter scales are strongly affected by numerical error. It is commonly understood that low order methods, such as ours, are not able to represent spectra up to the highest resolved wavenumbers, unlike spectral methods. Looking closely at Figure 10, we see that at t_{98} the $\alpha = 2$ case does not give optimal results because the energy in the inertial range prior to the highest resolved wavenumbers is slightly over-predicted. In this case, the model coefficient $(C_s \bar{\Delta})^2$ is slightly low, and fortuitously, the energy spectra at the high resolved wavenumbers matches the data at t_{98} . However, this no longer occurs at t_{171} .

3.3.1 The filter width ratio and the test filters

In earlier sub-sections we showed how to calculate $\hat{\Delta}$ on regularly connected grids. Here we will make use of this information by re-defining the square of filter width ratio appearing in the model as

$$\alpha = \left(\frac{\hat{\Delta}}{\Delta}\right)^2 = \kappa \left(\frac{\hat{\Delta}}{h}\right)^2, \quad (29)$$

where h is the approximate size of scales resolved by the discretization or the grid size, and κ is a positive parameter that may potentially depend on the the implicit filter due to numerical discretization and the test filter.

In re-defining the filter width ratio by introducing (29), we are attempting to clarify its dependence on the test and implicit numerical filters. As results will show, the only question in setting α will be that due to the unknown implicit numerical filter, reflected through κ . The dependence of α on the test filter will be fully understood.

3.3.2 Numerical results on hexes

Using the hexahedral grid described earlier, we performed two sets of simulations. The two sets, both composed of simulations performed with the S1, W1, S2, and W2 filters, are distinguished by different values of parameter κ in (29). In both sets, the test filter widths were computed based on the average radial wavenumber ($k_r^* = |\mathbf{k}^*|$) of the closed isosurface defined by $G(\mathbf{k}^*) = 0.75$, where $G(\mathbf{k})$ is the transfer function for each of the test filters. Recall that based on this definition, test filter widths are determined as $\hat{\Delta} = \pi/k_r^*$. A list of the test filter widths, scaled by the grid spacing h appears in Table I. We use these widths to compute the filter width ratio squared based on the assumption in (29) as $\alpha = \kappa(\hat{\Delta}/h)^2$. For example, in the first set of the simulations we fix κ at 0.680, which for the dynamic model with test filter S1 yields $\alpha = 5$. In Table IV, we give a summary of the two sets of simulations on hexes. In Figure 11 we plot simulation results corresponding to the filter width ratios noted in Table IV. Here we see that results are independent of the test filter as long as its filter width is consistently computed. The dynamic model is robust enough that it gives nearly identical results for all test filters regardless of the vast difference between the filters. This vast difference can be seen in the transfer functions and filter kernels of standard and wide filters (depicted in previous sub-sections) as well as in the widths of such filters noted in Table IV.

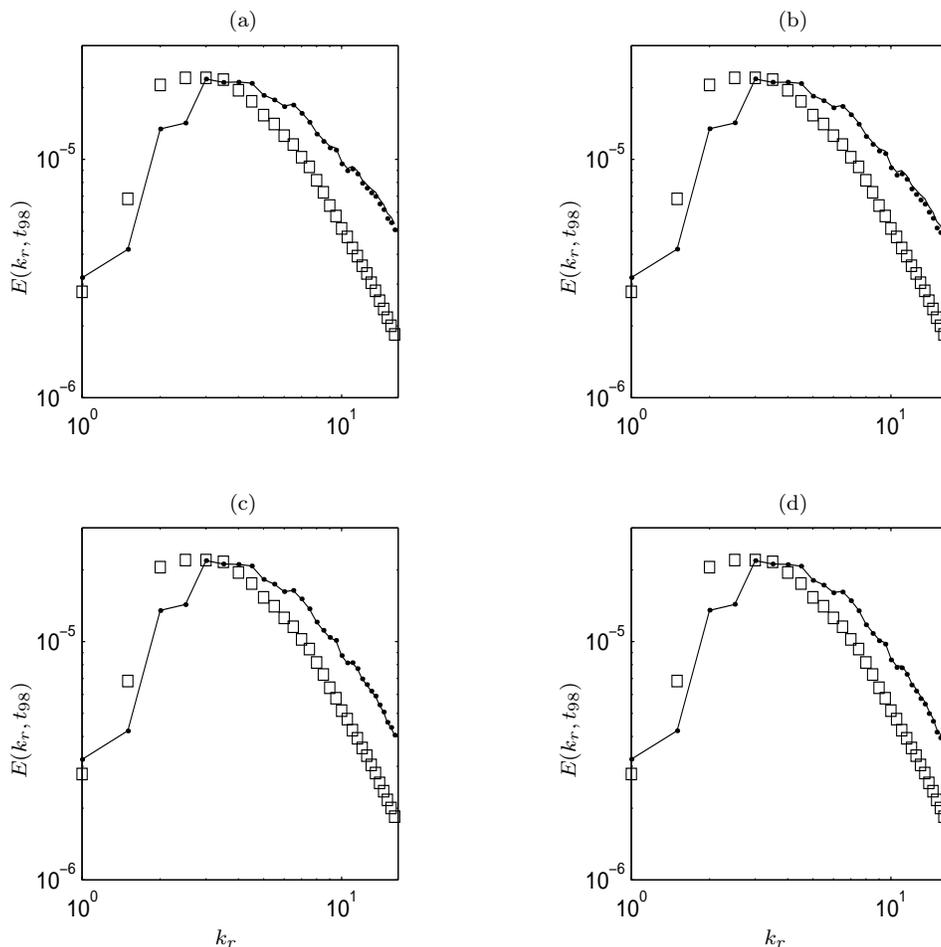


Figure 11: Sets of simulations on hexes comparing dynamic model results with different test filters. (a) Two simulations, one with filter S1, the other with filter W1, and both with $\kappa = 0.680$. (b) Two simulations, one with filter S2, the other with filter W2, and both with $\kappa = 0.680$. (c) Two simulations, one with filter S1, the other with filter W1, and both with $\kappa = 0.544$. (d) Two simulations, one with filter S2, the other with filter W2, and both with $\kappa = 0.544$. For all plots, — : standard filter S1 or S2; • : wide filter W1 or W2.

Filter	SET 1 ($\kappa = 0.680$)	SET 2 ($\kappa = 0.544$)
S1	5.00	4.00
W1	14.3	11.5
S2	6.51	5.21
W2	15.9	12.7

Table 4: Summary of filter width ratio squares, α , used for simulations on hexes. All test filters widths are based on $G(\mathbf{k}^*) = 0.75$. $\alpha \equiv (\hat{\Delta}/\bar{\Delta})^2 = \kappa(\hat{\Delta}/h)^2$.

The quality of a given simulation could be linked to the choice of κ (two choices of κ were given in Table IV and Figure 11), resulting in a tuning parameter. Rather than view κ in this way, we choose to view it as a property of the discretization (correcting the choice of h for $\bar{\Delta}$). The fact that each particular choice of κ leads to near identical results across a wide variety of properly characterized filters confirms that κ is independent of filter type and thus a property of the numerical method and possibly the flow. As can be seen in Table IV, the first choice of $\kappa = 0.680$ results in $\alpha = 5$, while the second choice ($\kappa = 0.544$) results in $\alpha = 4$. Clearly, neither of these choices is optimal (as is shown in Figure 10) but again, that is not our goal, rather as we will see later, our new method of dynamically determining the filter width ratio requires only that results of two properly characterized filters with the same value of κ return similar results (i.e. that κ be independent of filter choice).

3.3.3 Numerical results on tets and wedges

Similar to what is done in the previous sub-section, here we present sets of simulations with tetrahedral and wedge elements. In all sets, the test filter widths were computed based on the average radial wavenumber of the closed isosurface defined by $G(\mathbf{k}^*) = 0.85$. Note that in the previous sub-section we used $G(\mathbf{k}^*) = 0.75$, instead of $G(\mathbf{k}^*) = 0.85$, to obtain widths of test filters on hexes. The reason we did not use $G(\mathbf{k}^*) = 0.85$ for hexes is that it leads to high filter widths compared to tets and wedges. For example, looking at Tables I and II we see that $G(\mathbf{k}^*) = 0.85$ leads to filter widths which for hexes are approximately two times greater than for tets. For consistency regardless of topology, we choose a value of $G(\mathbf{k}^*)$ which leads to filter widths which are of the same order for hexes, tets, and wedges. Looking at Tables I, II, and III, we see that $G(\mathbf{k}^*) = 0.75$ for hexes leads to filter widths which are

Filter	SET 1 ($\kappa = 1.0$)	SET 2 ($\kappa = 0.558$)
S1	7.17	4.00
W1	21.0	11.7
S2	8.51	4.75
W2	22.4	12.5

Table 5: Summary of filter width ratio squares, α , used for simulations on tets. All test filters widths are based on $G(\mathbf{k}^*) = 0.85$. $\alpha \equiv (\hat{\Delta}/\bar{\Delta})^2 = \kappa(\hat{\Delta}/h)^2$.

Filter	SET 1 ($\kappa = 1.0$)	SET 2 ($\kappa = 0.558$)
S1	10.8	4.00
W1	27.7	11.7

Table 6: Summary of filter width ratio squares, α , used for simulations on wedges. All test filters widths are based on $G(\mathbf{k}^*) = 0.85$. $\alpha \equiv (\hat{\Delta}/\bar{\Delta})^2 = \kappa(\hat{\Delta}/h)^2$.

of the same order as those for tets and wedges with $G(\mathbf{k}^*) = 0.85$. It must be said that although we have chosen specific values of the filter transfer functions, $G(\mathbf{k})$, for computing filter widths, and this might seem biased, ultimately these values will not have an impact on the results with the new model to be derived. Tables V and VI contain information regarding the filter width ratios used in the simulations. In Figure 12 we plot simulation results on tets corresponding to the filter width ratios noted in Table V. Once again, we see the robustness of the dynamic model as it gives nearly identical results regardless of the test filter. In Figure 13 we plot simulation results on wedges corresponding to the filter width ratios noted in Table VI. Once again, results are nearly identical results regardless of the test filter.

Our simulations on hexes, tets and wedges, have shown that results are invariant as long as the filter widths are consistently computed with definitions given. This is an important conclusion because as was originally discussed by Lund [5], often in the literature [11] differences in simulation results have been incorrectly attributed to using test filters of different shapes as a result of not computing test filter widths with a proper definition. Furthermore, it is important to note that results are invariant to test filters on a set or fixed topology. Thus, for a fixed topology we can say that the optimal value of parameter κ (κ_{opt}) such that results match experimental

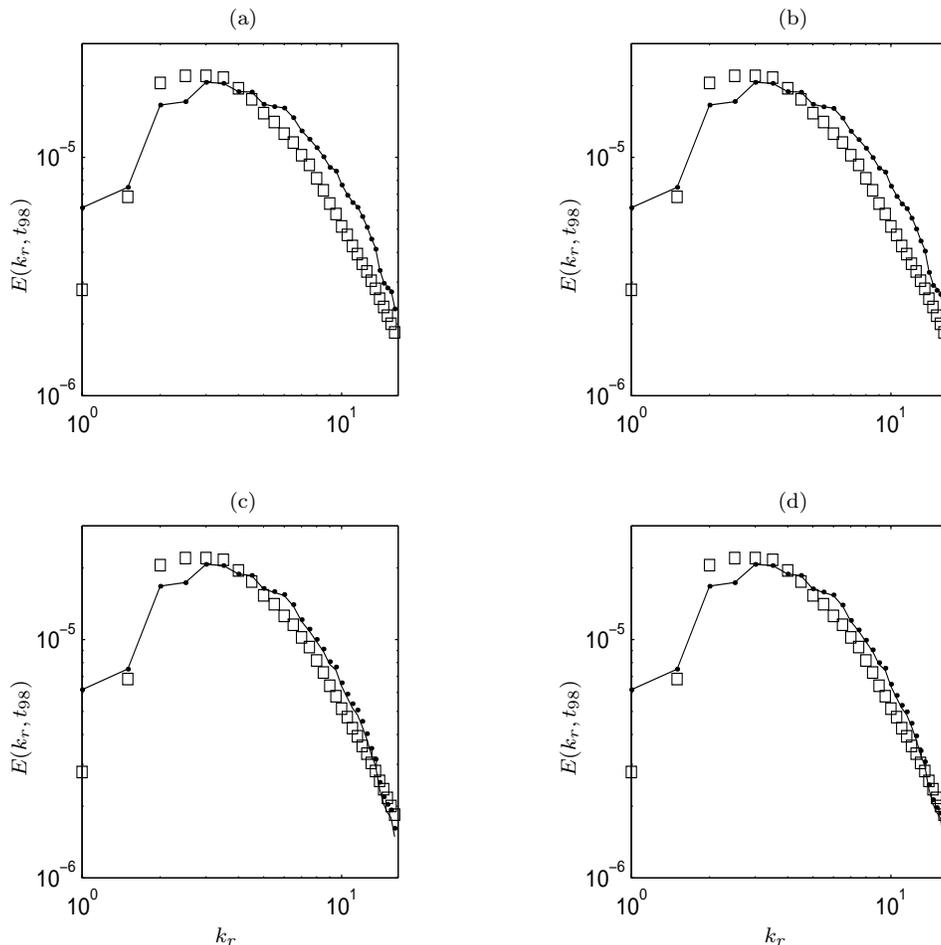


Figure 12: Sets of simulations on tets comparing dynamic model results with different test filters. (a) Two simulations, one with filter S1, the other with filter W1, and both with $\kappa = 1.0$. (b) Two simulations, one with filter S2, the other with filter W2, and both with $\kappa = 1.0$. (c) Two simulations, one with filter S1, the other with filter W1, and both with $\kappa = 0.588$. (d) Two simulations, one with filter S2, the other with filter W2, and both with $\kappa = 0.588$. For all plots, — : standard filter S1 or S2; • : wide filter W1 or W2.

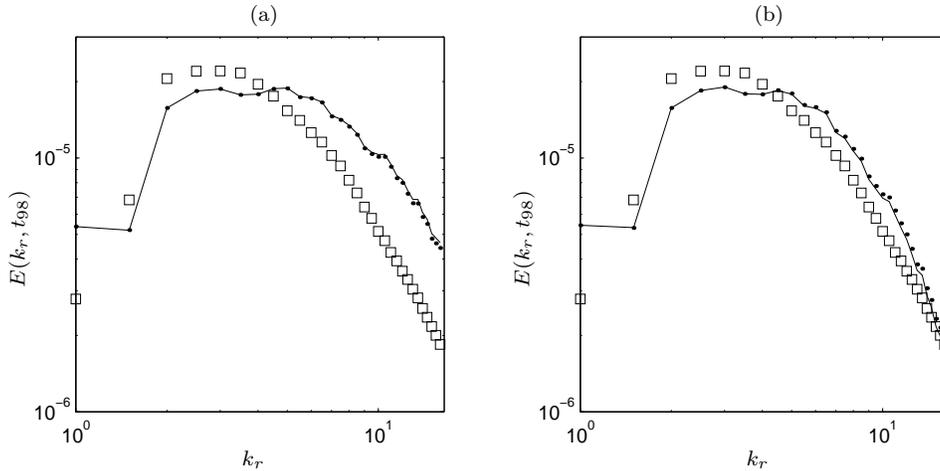


Figure 13: Sets of simulations on wedges comparing dynamic model results with different test filters. (a) Simulations with $\kappa = 1.0$. (b) Simulations with $\kappa = 0.370$. For both plots, — : standard filter S2; • : wide filter W2.

data well, is independent of the test filter used and solely dependent on the discretization. The difficulty now shifts to finding the value of κ_{opt} , which in turn provides the optimal value of the filter width ratio squared (α). Finally, our results give us confidence that the scale-invariance assumption in the dynamic model is not being violated. If it were being violated, then resulting spectra with standard (S) test filters would be much different than resulting spectra with wide (W) test filters. This is important as we will rely on the scale-invariance assumption for our up-coming derivations and tests.

4 Dynamic formulation of the filter width ratio

In this section we make use of results of the previous section to derive a formulation for dynamic computation of the model parameter α . Recall the subgrid-scale stress resulting from filtering the Navier-Stokes equations with what we referred to as a primary filter:

$$\tau_{ij} = \overline{u_i u_j} - \overline{u_i} \overline{u_j}. \quad (30)$$

and recall the subtest-scale stress resulting from successive filtering of the Navier-Stokes equations with the primary filter and a test filter:

$$T_{ij} = \widehat{u_i u_j} - \widehat{u_i} \widehat{u_j}. \quad (31)$$

Analogously, successively filtering the Navier-Stokes equations with the primary, test, and a second test filter yields the following secondary subtest-scale stress:

$$\mathcal{T}_{ij} = \widetilde{\widetilde{u_i u_j}} - \widetilde{\widetilde{u_i}} \widetilde{\widetilde{u_j}}. \quad (32)$$

In what remains of this work, the stress in (30) will be referred to as the stress at the $G_{\bar{\Delta}}$ -level where $G_{\bar{\Delta}}$ denotes the primary filter kernel of width $\bar{\Delta}$. Similarly, the stress in (31) will be referred to as the stress at the $G_{\hat{\Delta}}$ -level, where $G_{\hat{\Delta}}$ denotes the filter kernel of width $\hat{\Delta}$ resulting from successive applications of the primary and a test filter. Finally, the stress in (32) will be referred to as the stress at the $G_{\tilde{\Delta}}$ -level, where $G_{\tilde{\Delta}}$ denotes the filter kernel of width $\tilde{\Delta}$ resulting from successive applications of the primary filter, the test filter, and a second test filter. In summary, we will be working with three filter kernels: $G_{\bar{\Delta}}$, $G_{\hat{\Delta}}$, and $G_{\tilde{\Delta}}$.

Under the assumption of scale-invariance, the deviatoric portions of the stresses at the $G_{\bar{\Delta}}$ -level and at the $G_{\hat{\Delta}}$ -level are represented through the Smagorinsky model in (3) and (9), respectively. Analogously, the deviatoric portion of the stress at the $G_{\tilde{\Delta}}$ -level can be modeled as

$$\mathcal{T}_{ij}^d = -2(C_s \tilde{\Delta})^2 |\widetilde{\widetilde{S}}| \widetilde{\widetilde{S}}_{ij}, \quad (33)$$

where the strain rate tensor, $\widetilde{\widetilde{S}}_{ij}$, and its norm, $|\widetilde{\widetilde{S}}|$ are based on the triple-filtered velocity $\widetilde{\widetilde{u_i}}$.

An analogous identity to that of Germano, this time between the stresses at the $G_{\hat{\Delta}}$ -level and the $G_{\tilde{\Delta}}$ -level, leads to

$$Q_{ij} = \mathcal{T}_{ij} - \widetilde{\widetilde{T}}_{ij}, \quad (34)$$

which can be modeled through the Smagorinsky-type expressions in (9) and (33) and can be expressed through resolved quantities as

$$Q_{ij} = \widehat{\widetilde{\widetilde{u_i u_j}}} - \widetilde{\widetilde{u_i}} \widetilde{\widetilde{u_j}} \quad (35)$$

by using (31) and (32). The square of the difference between the modeled and resolved expressions for the deviatoric portion of Q_{ij} (Q_{ij}^d) is given

as $\mathcal{D}_{ij}\mathcal{D}_{ij}$, where

$$\mathcal{D}_{ij} = Q_{ij}^d - 2(C_s\widehat{\Delta})^2 N_{ij}. \quad (36)$$

Here,

$$N_{ij} = |\widetilde{S}|_{\widehat{S}_{ij}} - \gamma|\widetilde{S}|_{\widetilde{S}_{ij}}, \quad (37)$$

where $\gamma = (\widetilde{\Delta}/\widehat{\Delta})^2$. Minimization of (36) with respect to C_s leads to

$$(C_s\widehat{\Delta})^2 = \frac{1}{2} \frac{\langle Q_{ij}N_{ij} \rangle}{\langle N_{kl}N_{kl} \rangle}, \quad (38)$$

where Q_{ij} in (35) is used. The brackets denote averaging over spatially homogeneous directions, such as is done in expression (12). Notice that tensor N_{ij} requires knowledge of a ratio between filter widths, γ , analogous to tensor M_{ij} in (11). The ratio γ can be expressed as

$$\gamma = \left(\frac{\widetilde{\Delta}}{\widehat{\Delta}} \right)^2 = \frac{(\widetilde{\Delta}/\bar{\Delta})^2}{(\widehat{\Delta}/\bar{\Delta})^2}. \quad (39)$$

The denominator in the previous expression is precisely the square of the filter width ratio, α , in (11) when standard filters S1 or S2 are used as the test filter in the dynamic model. The numerator is α when wide filters W1 or W2 are used as the test filter in (11). Re-defining α in the way of (29), γ becomes

$$\gamma = \frac{(\widetilde{\Delta}/\bar{\Delta})^2}{(\widehat{\Delta}/\bar{\Delta})^2} = \frac{\kappa_1(\widetilde{\Delta}/h)^2}{\kappa_2(\widehat{\Delta}/h)^2}, \quad (40)$$

where $\widetilde{\Delta}$ is the width of the wide test filter and $\widehat{\Delta}$ is the width of the standard test filter. It was concluded in the previous section that dynamic model results are independent of the test filter (S1, S2, W1, or W2) as long as the test filter width is computed accurately and consistently. Dynamic model results with the different filters developed here are nearly indistinguishable thus, $\kappa_1 \approx \kappa_2$ can be taken, leading to

$$\gamma = \frac{\kappa_1(\widetilde{\Delta}/h)^2}{\kappa_2(\widehat{\Delta}/h)^2} = \left(\frac{\widetilde{\Delta}}{\widehat{\Delta}} \right)^2, \quad (41)$$

where the last ratio involves the widths of the wide test filter and the standard test filter. This ratio between test filters can be computed with the

definition shown and used earlier, giving rise to at least an approximate expression for γ and thus $(C_s \hat{\Delta})^2$ in (38).

Next, we divide the expression in (38) by that in (12) leading to

$$\alpha = \left(\frac{\hat{\Delta}}{\tilde{\Delta}} \right)^2 = \frac{\langle Q_{mn} N_{mn} \rangle \langle M_{kl} M_{kl} \rangle}{\langle N_{pq} N_{pq} \rangle \langle L_{ij} M_{ij} \rangle}. \quad (42)$$

Recall that $(C_s \hat{\Delta})^2 = \langle Q_{mn} N_{mn} \rangle / \langle N_{pq} N_{pq} \rangle$ can be approximately computed as was previously described, and M_{ij} in (11) is a function of α . Hence, expansion of (42) leads to the following quadratic algebraic equation for α :

$$c_0 - c_1 \alpha + c_2 \alpha^2 = 0, \quad (43)$$

where

$$\begin{aligned} c_2 &= \frac{\langle Q_{mn} N_{mn} \rangle}{\langle N_{pq} N_{pq} \rangle} \langle |\hat{S}|^2 \hat{S}_{ij} \hat{S}_{ij} \rangle + \langle L_{ij} |\hat{S}| \hat{S}_{ij} \rangle \\ c_1 &= 2 \frac{\langle Q_{mn} N_{mn} \rangle}{\langle N_{pq} N_{pq} \rangle} \langle |\hat{S}| \hat{S}_{ij} | \hat{S} | \hat{S}_{ij} \rangle + \langle L_{ij} |\hat{S}| \hat{S}_{ij} \rangle \\ c_0 &= \frac{\langle Q_{mn} N_{mn} \rangle}{\langle N_{pq} N_{pq} \rangle} \langle |\hat{S}| \hat{S}_{ij} | \hat{S} | \hat{S}_{ij} \rangle. \end{aligned} \quad (44)$$

Solution of the quadratic equation in (43) yields a dynamic filter width ratio, which can be used to calculate the dynamic model coefficient in (12). The only input parameter is the ratio between the widths of the test filters in (41), $(\hat{\Delta}/\tilde{\Delta})^2$, which can be directly computed once the filters have been defined. Note that the width of the grid filter is no longer required.

4.1 Numerical results with DFWR

The finite element-based filters presented in Section 3 lend perfectly for implementation of the dynamic filter width ratio. In expressions (38), (43), and (44), filter S1 or S2 can act as the first test filter, denoted with an over-hat. Furthermore, the trapezoidal filter can act as the second test filter denoted with an over-tilde. Recall that application of filter S1 or S2 followed by application of the trapezoidal filter yields wide filters W1 or W2, respectively. Thus, we refer to using filter S1 and the trapezoidal filter to compute the dynamic model parameter α through (43) as DFWR with S1 and W1. Furthermore, we refer to using filter S2 and the trapezoidal filter to compute the dynamic filter width ratio α through (43) as DFWR with S2 and W2.

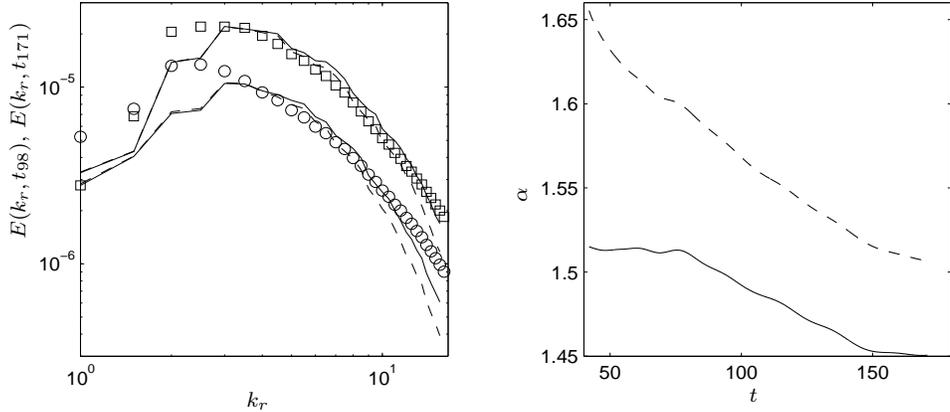


Figure 14: Dynamic model simulations on hexes with DFWR. Energy spectra at t_{98} are shown on the left and temporal evolution of the filter width ratio squared is shown on the right. — : DFWR with $(\hat{\Delta}/\hat{\Delta})^2 = 2.9$ and filters S1 and W1; ---- : DFWR with $(\hat{\Delta}/\hat{\Delta})^2 = 2.4$ and filters S2 and W2.

In Figures 14, 15, and 16, we present isotropic turbulence simulation results using DFWR on hex, tet, and wedge grids. Here we see that the resulting energy spectra at t_{98} match the experimental spectrum quite well, demonstrating the success of the DFWR formulation. The DFWR has essentially obviated the importance of the width of the undefined grid filter by spectrally sampling the resolved field and inherently extracting this information. In Figures 14, 15, and 16 we also see the temporal evolution of predicted filter width ratios squared (α). Though one might be concerned about encountering complex solutions to (43), every step of each simulation performed with this model yielded two positive, real roots. The second root, which we do not use, yields values for α of $O(10)$, leading to low values of the model coefficient.

With the difficulty of determining the dynamic model parameter α in (12) out of the way, we can truly say that the dynamic model on grids of hexahedral, tetrahedral, and wedge elements performs equally well at least for decaying isotropic turbulence. Despite the vast difference between standard and wide test filters and despite the directional bias introduced by the tet and wedge grids and the test filters on these topologies, the model can adequately represent the cascading transfer of energy that occurs in decay of isotropic turbulence.

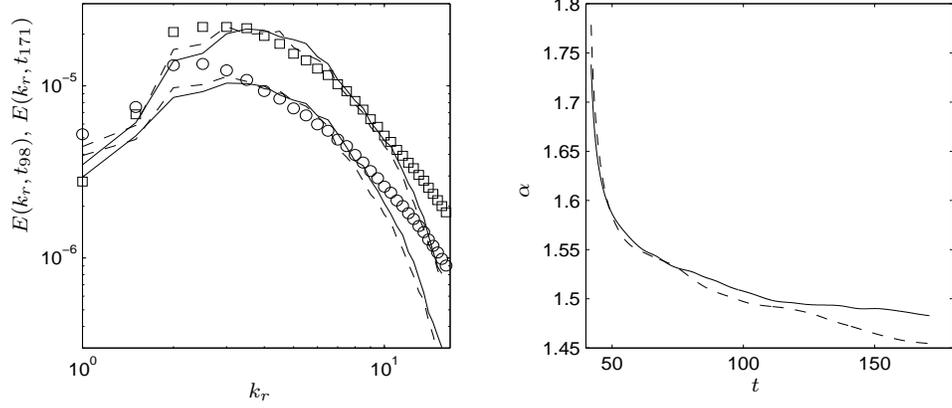


Figure 15: Dynamic model simulations on tets with DFWR. Energy spectra at t_{98} are shown on the left and temporal evolution of the filter width ratio squared is shown on the right. — : DFWR with $(\tilde{\Delta}/\hat{\Delta})^2 = 2.9$ and filters S1 and W1; ---- : DFWR with $(\tilde{\Delta}/\hat{\Delta})^2 = 2.6$ and filters S2 and W2.

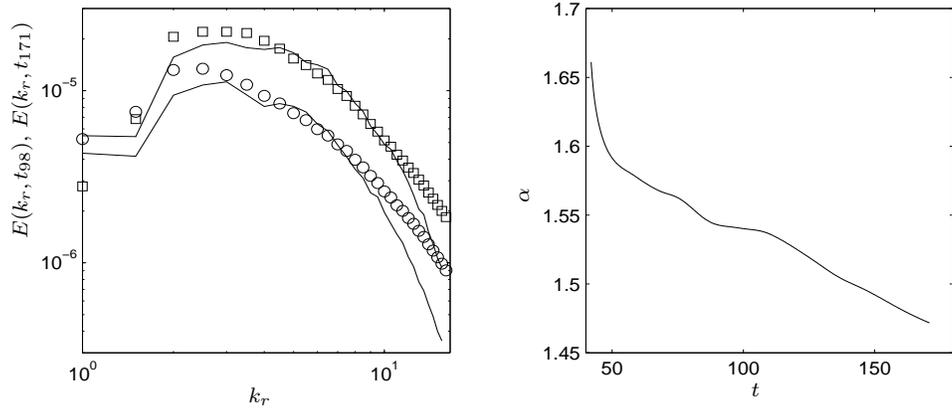


Figure 16: Dynamic model simulations on wedges with DFWR and $(\tilde{\Delta}/\hat{\Delta})^2 = 2.6$ Energy spectra at t_{98} are shown on the left and temporal evolution of the filter width ratio squared is shown on the right. DFWR uses filters S2 and W2.

$G(\mathbf{k}^*)$	(width of W1) ² /(width of S1) ²	(width of W2) ² /(width of S2) ²
0.50	2.75	2.37 \approx 2.4
0.65	2.86 \approx 2.9	2.43 \approx 2.4
0.75	2.87 \approx 2.9	2.44 \approx 2.4
0.85	2.91 \approx 2.9	2.5

Table 7: Limiting values of test filter width ratios $\tilde{\Delta}/\hat{\Delta}$ for filters on hexes.

$G(\mathbf{k}^*)$	(width of W1) ² /(width of S1) ²	(width of W2) ² /(width of S2) ²
0.50	2.71	2.51
0.65	2.82 \approx 2.8	2.57 \approx 2.6
0.75	2.89 \approx 2.9	2.59 \approx 2.6
0.85	2.92 \approx 2.9	2.63 \approx 2.6

Table 8: Limiting values of test filter width ratios $\tilde{\Delta}/\hat{\Delta}$ for filters on tets.

4.2 Test filter widths in the DFWR

The only information required in the formulation of DFWR is the width of the standard and wide test filters used. More specifically, we need the ratio between the width of the wide filter and the width of the standard filter, $\tilde{\Delta}/\hat{\Delta}$, referred to as the test filter width ratio. Recall that the width of these filters was defined as π/k_r^* , where k_r^* is the average radial wavenumber for which the filter transfer function is a specified value. Here, we discuss how we choose this value. In Tables I and II we presented test filter widths for various specified values of the transfer functions of standard and wide filters on hexes and tets. From these tables we see that as the specified value of the transfer functions varies between 0.5 and 1.0, the ratio between the filter widths of W1 (viz. W2) and S1 (viz. S2) is roughly constant. We choose this constant as the test filter width ratio in DFWR. The nearly constant behavior of the test filter widths is made clear by Tables VII and VIII. Although not shown in the tables, a similar behavior was found between standard and wide test filters on wedges.

5 Conclusion

The filter width ratio parameter can play a major role in dynamic model results, especially for simulations performed on relatively coarse grids. It was shown that dynamic model results of isotropic turbulence can be independent of the test filter used as long as its width is consistently computed. This lends credibility to the dynamic procedure as it should yield a Smagorinsky coefficient depending only on the properties of the grid filter (numerical method). In this article we have used this property of the dynamic procedure to obtain accurate estimation of the filter width ratio (referred to as DFWR) leading to accurate simulation results for decaying isotropic turbulence. It is important to note that DFWR as well as the dynamic model and other variants (i.e. dynamic mixed models) all rely on the scale-invariance assumption. Thus, in non-equilibrium flows, low Reynolds number turbulent flows, or very coarse grid simulations, for which the scale-invariance assumption is often violated, DFWR could potentially worsen this violation by injecting errors directly into the dynamic model via bad estimates of the filter width ratio. With this in mind, future research will focus on generalizing DFWR for flows or simulations in which scale-dependency of the dynamic model is required. Furthermore, in the near future, we hope to apply the dynamic estimation of the filter width ratio in channel flows.

References

- [1] C. H. Whiting and K. E. Jansen, “A stabilized finite element method for the incompressible Navier-Stokes equations using a hierarchical basis,” *International Journal of Numerical Methods in Fluids* **35**, 93-116 (2001).
- [2] J. Smagorinsky, “General circulation experiments with the primitive equations, I. The basic experiment,” *Monthly Weather Review* **91**, 99-152 (1963).
- [3] M. Germano, U. Piomelli, P. Moin and W. H. Cabot, “A dynamic subgrid-scale eddy viscosity model,” *Physics of Fluids* **3**, 1760 (1991).
- [4] D. K. Lilly, “A proposed modification of the Germano subgrid-scale closure,” *Physics of Fluids* **3**, 2746-2757 (1992)
- [5] T. S. Lund, “On the use of discrete filters for large-eddy simulation,” *Annual Research Briefs, Center for Turbulence Research, Stanford University*, 83-95 (1997).

- [6] M. Germano, “Turbulence: the filtering approach,” *Turbulence: the filtering approach*, *Journal of Fluid Mechanics* **238**, 325-336 (1992).
- [7] R. S. Rogallo and P. Moin, “Numerical simulation of turbulent flows,” *Annual Review of Fluid Mechanics* **16**, 99-137 (1984).
- [8] O. V. Vasilyev, “Computational Constraints on Large Eddy Simulation of Inhomogeneous Turbulent Complex Geometry Flow,” *Proceedings of the Third AFOSR International Conference on DNS/LES*, edited by C. Liu and L. Sakell and T. Beutner (2001).
- [9] G. S. Winckelmans, A. A. Wray and O. V. Vasilyev, “Testing of a new mixed model for LES: the Leonard model supplemented by a dynamic Smagorinsky term,” in *Proc. Summer Program VII (Center for Turbulence Research, Stanford University)*, 367-387 (1998).
- [10] T. S. Lund and H. J. Kaltenbach, “Explicit Filtering for LES,” *Annual Research Briefs, Center for Turbulence Research, Stanford University*, 83-95 (1995).
- [11] F. M. Najjar and D. K. Tafti, “Study of discrete test filters and finite difference approximations for the dynamic subgrid-stress model,” *Physics of Fluids* **8**, 1076-1088 (1996).
- [12] E. T. Spyropoulos and G. A. Blaisdell, “Evaluation of inhomogeneous formulations of the dynamic subgrid-scale model,” in *Transition, Turbulence, and Combustion*, edited by M. Y. Hussaini and T. B. Gatski and T. L. Jackson, **II**, 51-60 (1993).
- [13] A. E. Tejada-Martínez and K. E. Jansen, “Spatial test filters for dynamic model large-eddy simulation on finite elements,” *Communications in Numerical Methods in Engineering* **19**, 205-213 (2003).
- [14] S. Chester, F. Charlette and C. Meneveau, “Dynamic model for LES without Test Filtering: Quantifying the Accuracy of Taylor Series Approximations,” *Theoretical and Computational Fluid Dynamics* **15**, 165-181 (2001).
- [15] T. J. R. Hughes, L. Mazzei and K. E. Jansen, “Large-eddy simulation and the variational multiscale method,” *Computing and Visualization in Science* **3**, 47-59 (2000).

- [16] T. J. R. Hughes and L. Mazzei and A. A. Oberai and A.A. Wray, “The multiscale formulation of large eddy simulation: Decay of homogeneous isotropic turbulence,” *Physics of Fluids* **13**, 505-512 (2001).
- [17] T. J. R. Hughes and A. A. Oberai and L. Mazzei, “Large-eddy simulation of turbulent channel flows by the variational multiscale method,” *Physics of Fluids* **13**, 1784-1799 (2001).
- [18] K. E. Jansen and A. E. Tejada-Martínez, “An evaluation of the variational multiscale model for large-eddy simulation while using a hierarchical basis,” presented at the 40th AIAA Annual Meeting and Exhibit, Paper No. 2002-0283 (2002).
- [19] C. Meneveau and J. Katz, “Scale-invariance and turbulence models for large-eddy simulation,” *Annual Review of Fluid Mechanics* **415**, 1-32 (2000).
- [20] C. A. Taylor, T. J. R. Hughes and C. K. Zarins, “Finite element modeling of blood flow in arteries,” *Computer Methods in Applied Mechanics and Engineering* **158**, 155-196 (1998).
- [21] K. E. Jansen, “A stabilized finite element method for computing turbulence,” *Computer Methods in Applied Mechanics and Engineering* **174**, 299-317 (1999).
- [22] A. L. Marsden, O. V. Vasilyev and P. Moin, “Construction of Commutative Filters for LES on Unstructured Meshes,” *Journal of Computational Physics* **175**, 584-603 (2002).
- [23] J. S. Mullen and P. F. Fisher, “Filtering techniques for complex geometry fluid flows,” *Communications in Numerical Methods in Engineering* **15**, 9-18 (1999).
- [24] S. B. Pope, “*Turbulent Flows*”, Cambridge Press (2000).
- [25] G. Comte-Bellot and S. Corrsin, “Simple Eulerian time-correlation of full and narrow-band velocity signals in grid-generated ‘isotropic’ turbulence,” *Journal of Fluid Mechanics* **48**, 273-337 (1971).