Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 227 (2008) 5717–5735
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

Efficient form of the LANS-α turbulence model in a primitive-equation ocean model

Mark R. Petersen [a,b,*], Matthew W. Hecht [a], Beth A. Wingate [a,b]

[a] Computer, Computational and Statistical Sciences Division, Los Alamos National Laboratory, Los Alamos, NM, United States
[b] Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, NM, United States

Received 21 March 2007; received in revised form 21 January 2008; accepted 5 February 2008
Available online 4 March 2008

Abstract

The Lagrangian-averaged Navier–Stokes alpha (LANS-α) model is a turbulence parameterization that has been shown to capture some of the most important features of high resolution ocean modeling at lower resolution. The LANS-α model improves turbulence statistics with an additional nonlinear term and a smoothed advecting velocity. In this work we investigate two smoothing techniques—Helmholtz inversions and filters—and their effect on the LANS-α model's results and efficiency. We show that convolution filters are an effective smoothing method; that filters and Helmholtz inversions produce similar trends—statistics like higher-resolution non-LANS-α simulations—as the smoothing parameter is increased; and that the filter is computationally more efficient.

Filters must be constructed such that a pressure–velocity numerical instability is not excited. We show analytically that certain ranges of filter weights are unstable, and confirm this with numerical experiments. Our stability criterion also guarantees that the kinetic energy is well defined, and that the filtered velocity is smoother that the original velocity.

Simulations of LANS-α using the largest filter (width nine) in the POP primitive-equation ocean model resemble doubled-resolution simulations of standard POP in statistics such as kinetic energy, eddy kinetic energy, and potential temperature fields. The computational cost of adding LANS-α with this filter is only 27%, as compared to a factor of 8–10 for a doubling of resolution.
Published by Elsevier Inc.

Keywords: Lagrangian-averaged Navier–Stokes alpha model; Primitive equation ocean model; Geophysical turbulence

1. Introduction

The LANS-α model is a turbulence model that can reproduce some of the statistics and structures that are seen in standard simulations (without LANS-α) at higher resolution. The LANS-α model accomplishes this in part by using a smoothed velocity field for the advecting velocity in the momentum and tracer equations. The question addressed in this paper is, "What is the best way to smooth the velocity field in LANS-α?" This is not

* Corresponding author. Address: Computer, Computational and Statistical Sciences Division, Los Alamos National Laboratory, Los Alamos, NM, United States. Tel.: +1 505 667 7399; fax: +1 505 665 2659.
E-mail address: mpetersen@lanl.gov (M.R. Petersen).

0021-9991/$ - see front matter Published by Elsevier Inc.
doi:10.1016/j.jcp.2008.02.017

simply a matter of choosing the smoothing method that produces the best result. As a turbulence model, LANS-α must be extremely efficient as well; if the addition of LANS-α were expensive, it would be more sensible to drop the turbulence model altogether and simply run at higher resolution. Thus the choices we make in designing the LANS-α algorithm and smoothing filters are governed by the trade-offs of higher-resolution effects versus the cost of adding LANS-α to the model. One would like to obtain the effects of, say, a doubling of resolution while only increasing the computation time by a small fraction of the time required to double the resolution with the standard model.

We have designed a LANS-α algorithm and smoothing filter for primitive-equation ocean models that meets this criteria of a successful turbulence model. The algorithm is presented by Hecht, Holm, Petersen, and Wingate in a closely related work [1]. The focus of the present paper is the method used to smooth the advecting velocity. In the typical LANS-α model derivation, the smooth velocity, **u**, is computed from the rough velocity, **v**, with an inverse Helmholtz operator,

$$\mathbf{u} = (1 - \alpha^2\nabla^2)^{-1}\mathbf{v} \tag{1}$$

where the namesake α parameter is a length scale that determines the amount of smoothing [2]. The Helmholtz inversion is the basis of comparison for other smoothing methods in this work, as this is the relationship between **u** and **v** in the original LANS-α equations. With it we are able to run successful simulations of LANS-α; however, it requires an iterative solver and is therefore relatively expensive. Fortunately, local averaging methods work well in LANS-α and are much cheaper. In these methods, the smooth velocity is computed as a weighted average of nearby neighbors, and can be written as a convolution with the filter function g as

$$\mathbf{u} = g * \mathbf{v} = \int g(\mathbf{x} - \mathbf{y})\mathbf{v}d^3y. \tag{2}$$

Filters should not be considered an inferior substitute for the Helmholtz inversion in LANS-α. Rather, each is a valid method of smoothing within the LANS-α framework. The LANS-α equation using a filter (sometimes called the Kelvin-filtered Navier–Stokes equation) retains Kelvin's circulation theorem [3,4], which is a property of fundamental importance to LANS-α. If the filter is properly designed, other characteristics of LANS-α with the Helmholtz inversion are guaranteed: **u** is smoother than **v**, the energy is well defined, and LANS-α conserves energy in the absence of dissipation. In this paper we develop design criteria for filters that satisfy these requirements.

There is a precedent of using simple filters in LANS-α in large eddy simulation (LES) models. Geurts and Holm [5] used a simple three-point top-hat filter as a smoothing operator in LANS-α and Leray LES modeling of three-dimensional isotropic turbulent mixing. They found that the LANS-α and Leray models are considerably more accurate than dynamic eddy-viscosity models, and at a lower computational cost. The LANS-α and Leray models performed particularly well at capturing the flow features characteristic of the smaller resolved scales.

This paper is organized as follows: Section 2 briefly reviews the LANS-α equations for primitive-equations and the POP-α algorithm. The pressure–velocity instability, which develops with the incorrect choice of filter weights, is discussed in Section 3, and the treatment of the boundary using the filter is covered in Section 4. Using an eigenvalue analysis of the discretized filter in Section 5, we develop design criteria for the filter weights to ensure the proper smoothing and kinetic energy properties. The Green's function of the discrete Helmholz inversion is computed in Section 6 to make a quantitative comparison with the filter. Section 7 shows that POP-α can produce results qualitatively similar to those of higher-resolution simulations of standard POP using either the Helmholtz inversion or filters to smooth the velocity. In Section 8, we compare the merits of each. The family of filters presented here produce stable simulations and are 20–50% faster than Helmholtz inversions. Filters produce better temperature statistics in the baroclinic instability model problem, while the Helmholtz inversion produces higher kinetic and eddy kinetic energy.

2. The LANS-α model

This section gives a brief overview of the primitive-equation version of LANS-α. These equations were originally derived by Holm et al. [6], and are described at length, along with the POP-α algorithm, by Hecht et al. [1]. The equations are

$$\frac{\partial \mathbf{v}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{v} + u_3 \partial_z \mathbf{v} + v_j \nabla u_j + \mathbf{f} \times \mathbf{u} = -\frac{1}{\rho_0} \nabla \pi + \mathcal{F}(\mathbf{v}), \tag{3}$$

$$\frac{\partial \pi}{\partial z} + \rho_0 v_j \partial_z u_j = -\rho g, \tag{4}$$

$$\frac{\partial \varphi}{\partial t} + \mathbf{u} \cdot \nabla \varphi = \mathcal{D}(\varphi), \tag{5}$$

$$\nabla \cdot \mathbf{u} + \partial_z u_3 = 0, \tag{6}$$

$$\mathbf{u} = (1 - \alpha^2 \nabla^2)^{-1} \mathbf{v}, \tag{7}$$

$$\pi = p - \rho_0 \frac{1}{2} |\mathbf{u}|^2 - \rho_0 \frac{\alpha^2}{2} |\nabla \mathbf{u}|^2 \tag{8}$$

where $(v_1, v_2, v_3) = (\mathbf{v}, v_3)$ and $(u_1, u_2, u_3) = (\mathbf{u}, u_3)$ are the rough and smooth velocities, ∇ is the horizontal gradient, φ is a tracer (temperature, salinity, or a passive tracer), ρ is the full density, ρ_0 is the background density, $\mathcal{F}$ and $\mathcal{D}$ are momentum and tracer diffusion terms [7], $\mathbf{f}$ is the Coriolis parameter, g is gravitational acceleration, and α is the alpha model's smoothing length scale. These equations are incompressible (6) and Boussinesq. In the formulation used for this study the second term in (4) is omitted, and the hydrostatic pressure is used in the equation of state.

The principle feature of the LANS-α model is that the momentum Eq. (3) is an advection-diffusion equation for a Lagrangian-averaged velocity $\mathbf{v}$, while the advecting velocity is an Eulerian-averaged velocity $\mathbf{u}$. These names arise in the derivation of the LANS-α model [2], where velocities are averaged along a particle track (Lagrangian) or at a particular location (Eulerian). The Helmholtz relation (7) indicates that $\mathbf{u}$ is smoother that $\mathbf{v}$. In this paper we prefer the names *rough velocity* for $\mathbf{v}$, which is computed prognostically from (3), and *smooth velocity* for $\mathbf{u}$, which is computed diagnostically from (7). The LANS-α model has an additional nonlinear term $\sum_{j=1}^{3} v_j \nabla u_j$ in the momentum Eq. (3), which does not appear in the Leray model, and which allows LANS-α to satisfy Kelvin's circulation theorem while conserving energy and a form of potential vorticity in the absence of dissipation [2].

In this paper we investigate how to most effectively and efficiently smooth the rough velocity $\mathbf{v}$ to obtain the smooth velocity $\mathbf{u}$. This could be with a Helmholtz inversion, as in (7), or with a convolution using various filter functions, as in (2). LANS-α using a filter rather than a Helmholtz inversion still satisfies Kelvin's circulation theorem [3]. If the proper filters are chosen, one also conserves energy and a form of potential vorticity in the absence of dissipation.

The most commonly used form of the LANS-α equations use a three-dimensional (3D) smoothing. In the primitive equations, which assume a thin layer approximation, this reduces to a 2D smoothing, and so our filters are horizontal-only as well. 3D smoothing may be tested at a future time, but we expect that the computational cost would be too high to justify the benefit. The Helmholtz inversion is horizontally isotropic. Thus one of the design criteria for the filter is that it is horizontally isotropic as well.

The POP primitive-equation ocean–climate model [8,9], developed and maintained at Los Alamos National Laboratory, was augmented to include the LANS-α formulation [1]. POP is a split implicit/explicit code; that is, it takes an implicit step for the barotropic (vertically integrated 2D) velocity field, and takes an explicit step for the baroclinic (remaining 3D) velocity field. This is done so that the timestep is not limited by the fast free-surface gravity waves in the barotropic velocity field.

In the LANS-α version of POP, named POP-α, a smooth velocity must be computed for both the baroclinic and barotropic velocity fields. In the *explicit* baroclinic component of the code, the smooth baroclinic velocity, $\mathbf{u}$, is computed by simply smoothing $\mathbf{v}$. The *implicit* barotropic component is more complicated, and in the end a reduced version of the full barotropic algorithm, as derived from the LANS-α equations, was used [1]. The reduced algorithm produced nearly identical results to the full algorithm, but was three to four times faster.

A few pieces of the POP-α barotropic algorithm are needed to understand the pressure–velocity instability presented in the following section. The 2D barotropic velocities, denoted by capital letters, are found by integrating the full velocity from the bottom to the free surface height, η:

$$\mathbf{U} \equiv \frac{1}{H+\eta}\int_{-H}^{\eta} \mathbf{u}\,\mathrm{d}z, \tag{9}$$

where $H(x,y)$ is the ocean depth when the surface is at rest. By integrating the continuity Eq. (6), which states that the *smooth* velocity is divergence-free, we obtain a prognostic equation for the free surface height,

$$\int_{-H}^{\eta} (\nabla\cdot\mathbf{u} + \partial_z u_3)\mathrm{d}z = 0, \tag{10}$$

$$\frac{\partial\eta}{\partial t} + \nabla\cdot(H+\eta)\mathbf{U} = 0. \tag{11}$$

The free surface height now replaces the pressure in the barotropic form of the momentum equation,

$$\frac{\partial\mathbf{U}}{\partial t} + \mathbf{f}\times\mathbf{U} = -g\nabla\eta + \mathbf{G}, \tag{12}$$

where $\mathbf{G}$ contains the vertically integrated forcing terms. The barotropic momentum equation is the same as the shallow water equation.

3. Pressure–velocity instability

In this section, we present an analytic stability analysis of a pressure–velocity instability that constrains the filter weights in the barotropic filter. The pressure–velocity instability is a phenomenon particular to models where a smoothed velocity field is used in the equation for free surface height η (11). Consider a velocity field in one-dimensional (1D) discretized shallow water equations, where the pressure and velocity gridpoints are offset. If the velocity field were at the Nyquist frequency (alternating between positive and negative at each grid cell), then using the standard shallow water equations, the free surface height would raise (lower) where the velocity field converges (diverges). In either case, a pressure gradient force would ensue that would work against the tendency of the surface to undergo further deformation, providing a stabilizing effect.

In the LANS-α model the equation for the free surface height involves the *smooth* velocity because this equation is derived from the continuity Eq. (6), which involves the smooth velocity. When the outer stencil weights are sufficiently large in LANS-α, a *pressure–velocity instability* ensues in the highest frequencies. This can be shown with the simplest case, a 1D velocity field and a smoothing stencil of width three,

$$u_i = \frac{b v_{i-1} + a v_i + b v_{i+1}}{a+2b}. \tag{13}$$

In most cases, u_i and v_i will have the same sign at each gridpoint, as one would expect for a smoothing operator. However, for high frequencies and sufficiently large outer stencil weight b, u can have the opposite sign of v, as shown in Fig. 1 for the Nyquist frequency and $b = 0.7$.

The tendency equation for v depends on η, but η in turn is not directly dependent on the convergence/divergence of v but on that of u. In the case of a pure harmonic mode, restricting the local filter to produce a smoothed velocity u of the same sign everywhere as the rough velocity v is sufficient to also ensure that the divergence of smooth and rough velocities share the same sign, preserving the essential physical feedback mechanism of the underlying equations. Cases in which this physical feedback is respected or not, depending on the filter width, are illustrated in Fig. 2.

This understanding of the pressure–velocity instability allows us to construct an analytical stability criterion as follows: *The filter weights must be chosen so that the filtered velocity u retains the same sign as v for all harmonic modes resolved on the grid.*

For example, for the stencil of width three (13) where a is normalized to 1, consider the Nyquist frequency

$$v_{i+j} = \cos(\pi j/k) \tag{14}$$

with $k = 1$. Since $v_i = 1$, the condition that u and v are like-signed is $u_i > 0$. Using (13), the constraint on the stencil is $b < 0.5$. For a slightly higher wavenumber, $k = 1.5$, the constraint is weaker, $b < 1$. For still higher wavenumbers, u and v are like-signed for any choice of b. Thus b must be chosen based on the tightest constraint, $b < 0.5$.

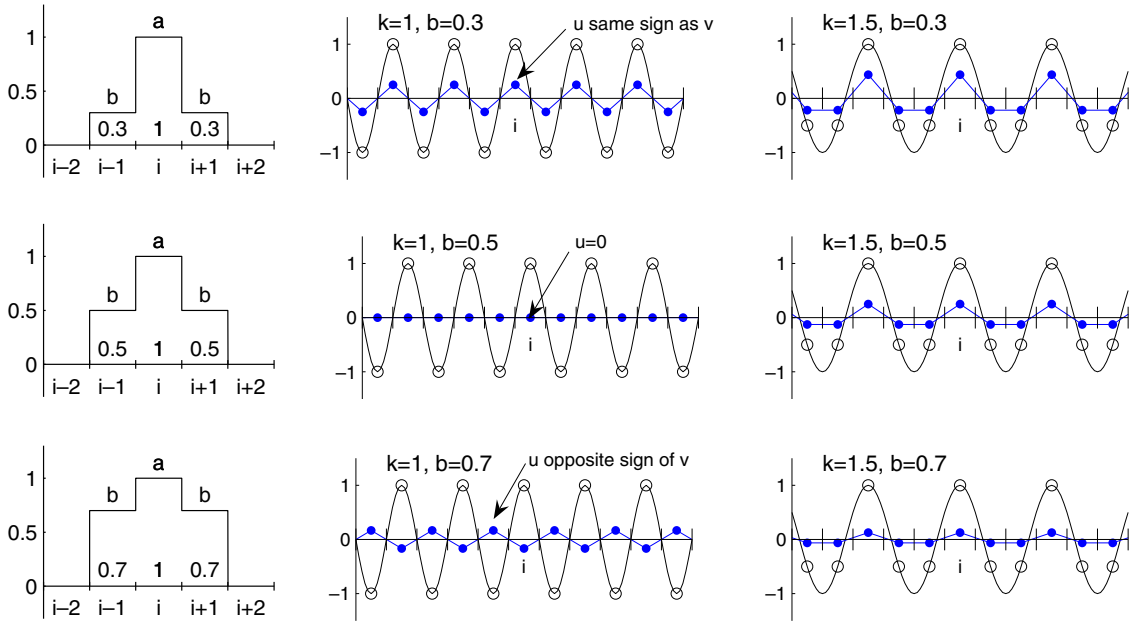

Fig. 1. The effects of filtering the rough velocity v (open circles) to compute the smooth velocity u (closed circles) for a 1D filter of width three, where the weight of the neighboring cell, b, is varied between 0.3 and 0.7, as shown in the left column. For the Nyquist frequency ($k = 1$, middle column) u is of opposite sign to v if $b > 0.5$; this results in an instability, since the equation for free surface height η involves u. For a slightly higher wavenumber ($k = 1.5$, right column) u has the same sign as v for $b < 1$.

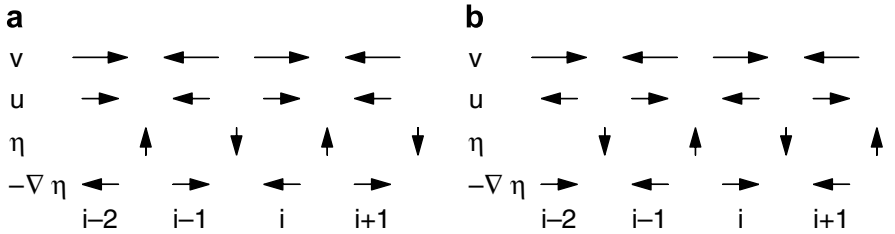

Fig. 2. Schematic of variables to illustrate the pressure–velocity feedback instability. In the stable case, $b < 0.5$ (a) the smooth and rough velocities, u and v, are of the same sign, and the pressure gradient force $-\nabla\eta$ counters the original velocity perturbation, as expected. In the unstable case, $b > 0.5$ (b), u and v have opposite signs; the free surface height, which is computed from u, increases (decreases) where u converges (diverges); now the pressure gradient force reinforces the rough velocity v, increasing this perturbation.

Numerical simulations of POP-α with a stencil of width three almost exactly correspond to the predictions of this stability analysis. The model runs stably when $b < 0.48$. At larger values of b an instability quickly grows in $\mathbf{v}$ and η at the Nyquist frequency, so that $\mathbf{v}$ is four times larger than $\mathbf{u}$ within 100 time steps (Fig. 3). By 150 timesteps the conjugate gradient routine in the barotropic solver fails to converge within 1000 iterations. Numerical simulations use a 2D stencil that is a squared version of the 1D stencil, as discussed below.

Because of the limitation imposed by the pressure–velocity instability, further smoothing must be accomplished by increasing the stencil size. The criterion that u and v must be like-signed for all frequencies applies, but larger stencils have more unknowns, one for each stencil weight. Fig. 4 shows these constraints in chart form for 1D filters of width 3, 5, 7, and 9 and for wavenumbers $k = 1 \ldots 4$. In general, these constraints require that the weights are each less than 0.5. Beyond that, each consecutive weight moving away from the center must be slightly less than the one before it. This latter requirement can be seen in the constraints for the stencil of width nine for wavenumber $k = 3$ and $k = 4$,

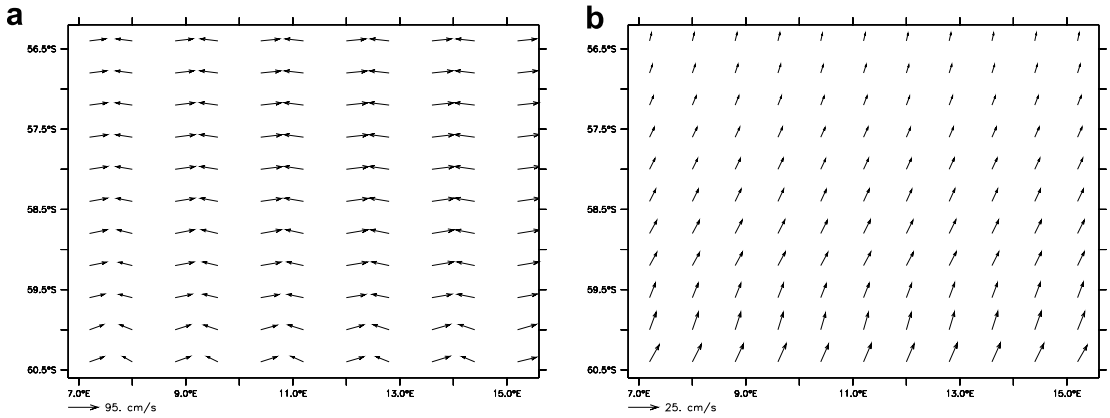

Fig. 3. An example of the pressure–velocity instability. Shown are the velocity fields of simulation 0.8F3 using $b = 0.52$ after 100 time steps. The Nyquist frequency grows exponentially in the rough velocity v (a), but is filtered out of the smooth velocity u (b) (note difference in vector length scales). The surface elevation η is computed using u, so pressure gradient forces don't counteract this unstable mode.

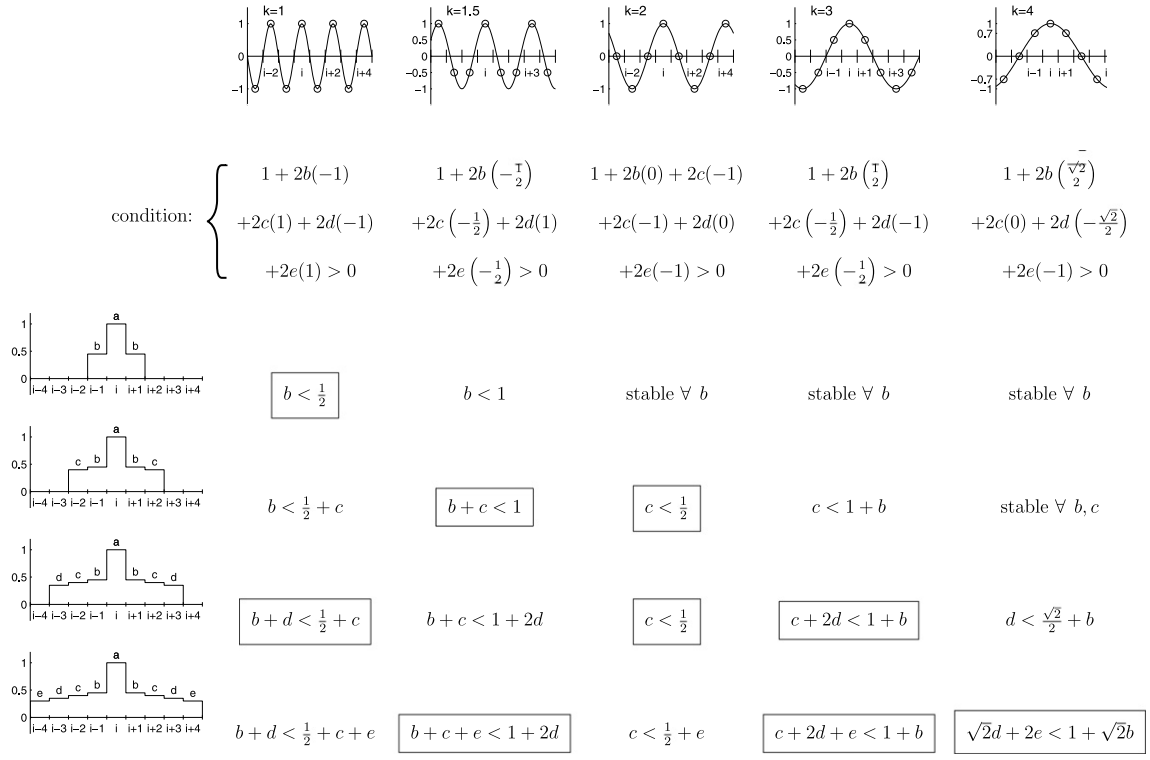

Fig. 4. Conditions required for stability for filters of width 3, 5, 7, and 9 (rows) for several wavenumbers of v (columns). Variables a–e are the weights of the 1D stencil, normalized such that $a = 1$. A stable scheme requires that u_i, the smooth velocity, is the same sign as v_i. Conditions that limit the weights b–e to be less that one-half are boxed.

$$c + 2d + e < 1 + b, \tag{15}$$

$$\sqrt{2}d + 2e < 1 + \sqrt{2}b. \tag{16}$$

Neither of these are satisfied if all the weights are 0.5, but are satisfied when the weights decrease as

$$b = 0.45, \quad c = 0.4, \quad d = 0.35, \quad e = 0.3. \tag{17}$$

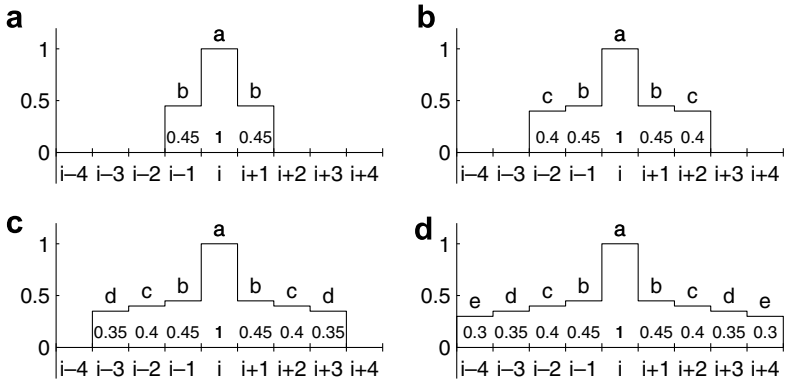

Fig. 5. Filters used in this study, with the value of each weight *b–e* under the corresponding variable. (a) 1D stencil, width 3, (b) width 5, (c) width 7 and (d) width 9.

a

j+4	e^2	ed	ec	eb	e	eb	ec	ed	e^2
j+3	de	d^2	dc	db	d	db	dc	d^2	de
j+2	ce	cd	c^2	cb	c	cb	c^2	cd	ce
j+1	be	bd	bc	b^2	b	b^2	bc	bd	be
j	e	d	c	b	a	b	c	d	e
j–1	be	bd	bc	b^2	b	b^2	bc	bd	be
j–2	ce	cd	c^2	cb	c	cb	c^2	cd	ce
j–3	de	d^2	dc	db	d	db	dc	d^2	de
j–4	e^2	ed	ec	eb	e	eb	ec	ed	e^2
	i–4	i–3	i–2	i–1	i	i+1	i+2	i+3	i+4

b

j+4	0.09	0.1	0.12	0.14	0.3	0.14	0.12	0.1	0.09
j+3	0.1	0.12	0.14	0.16	0.35	0.16	0.14	0.12	0.1
j+2	0.12	0.14	0.16	0.18	0.4	0.18	0.16	0.14	0.12
j+1	0.14	0.16	0.18	0.2	0.45	0.2	0.18	0.16	0.14
j	0.3	0.35	0.4	0.45	1	0.45	0.4	0.35	0.3
j–1	0.14	0.16	0.18	0.2	0.45	0.2	0.18	0.16	0.14
j–2	0.12	0.14	0.16	0.18	0.4	0.18	0.16	0.14	0.12
j–3	0.1	0.12	0.14	0.16	0.35	0.16	0.14	0.12	0.1
j–4	0.09	0.1	0.12	0.14	0.3	0.14	0.12	0.1	0.09
	i–4	i–3	i–2	i–1	i	i+1	i+2	i+3	i+4

Fig. 6. The 2D version of a filter stencil of width nine general form (a). This is simply the square of the 1D stencil. The 2D filters of width 3, 5, and 7 are subsets of this stencil. Our specific choice of weights is shown in (b).

Again, the results of this stability analysis were verified using the POP-α model. Simulations using the stencil weights in (17) were always stable (Fig. 5). If any of the weights are increased by 0.02, a fast-growing, high frequency appears in $\mathbf{v}$ and η (Fig. 3).

In the POP-α model the smoothing operator is 2D. The 2D stencil was chosen to be a squared version of the 1D stencil (Fig. 6). Such a stencil is isotropic, at least in terms of a 90° rotation, and thus follows the same stability analysis for a plane wave as the 1D case presented above. POP-α is very sensitive to these stability constraints; for example, if the corner weights of the stencil in Fig. 6b are increased by 10–20%, the simulation is unstable.

Other experiments were conducted where the 2D stencil was a diamond or circle instead of a square. The motivation here is that in the standard square 2D stencil (Fig. 6) the corner weights are small, and don't contribute much to the average. Thus computation time could be saved by removing the corners from the stencil, and the degree of smoothing should be nearly the same. However, in practice any stencil other than a full square stencil resulted in unstable simulations that did not converge in the barotropic solver. This was even true when only a single gridpoint was left off of the stencil on each corner, for example when the 0.09 weight is removed from a nine-wide stencil in Fig. 6b. Thus all of the simulations presented in this work use a complete square stencil.

4. Smoothing at the boundary

The boundary condition on the velocity in standard POP is no-slip. The choice of boundary conditions for the smooth and rough velocities in POP-α is closely related to the behavior of alpha near the boundary. Our

model Eqs. (3)–(8) are derived from a more general set (Eq. (18) in [2]) which includes the Lagrangian mean covariance, $\langle \xi\xi \rangle$, rather than α^2, where ξ is the displacement fluctuation. One may make a number of assumptions about the Lagrangian displacement that simplify the full equations: $\langle \xi\xi \rangle$ may be isotropic, constant in time, and constant in space. Each of these assumptions result in a reduced equation set and an associated boundary condition. If $\langle \xi\xi \rangle$ is isotropic and constant in space and time, the equations reduce to the model used in this paper (Eqs. (3)–(8)), where $\langle \xi\xi \rangle$ is replaced by α^2. Their associated boundary condition is that α is constant all the way to and including the boundary.

In order to follow the numerical formulation of POP, we chose no-slip boundaries for both the smooth and rough velocities ($\mathbf{u} = \mathbf{v} = 0$). A no-slip condition on $\mathbf{u}$ has been used in analytical studies as well [10,11]. The Helmholtz inversion smoothing method uses a constant α throughout the domain, which is consistent with the model equations.

It is not clear how one would construct a filter that is analogous to a constant α throughout the domain. In the present work, the stencil is reduced in size near the boundary so that all grid-cells used in the averaging calculation are water-cells, and none are land-cells. For example, at a grid-cell directly beside a land cell no averaging takes place (i.e. $u_{i,j} = v_{i,j}$). If the nearest land-cell is two grid-cells away, a filter of width three is used. This method of handling the boundaries proved to be stable and worked well for the smooth-walled channel domain presented in this paper.

Other filtering schemes near the boundary were considered: One could use a constant-size stencil and assign zero velocity to land-cells, but this strongly damps the smooth velocity near the boundary. Another possibility is to use large but nonsymmetric stencils near the boundary, but this proved to be unstable in our investigation.

Reducing the stencil size as one approaches the boundary is analogous to reducing alpha near the boundary. Using a variable α modifies the LANS-α equation set (Eqs. (3)–(8)). If one begins with Eq. (18) in [2] and assumes that the fluctuation covariance tensor is isotropic, constant in time, but not constant in space, then $\langle \xi\xi \rangle = \alpha^2 \mathbf{I}$, the Helmholtz operator is

$$\mathbf{u} = \left(1 - \nabla \cdot \alpha^2(\mathbf{x})\mathbf{I} \cdot \nabla\right)^{-1}\mathbf{v}, \tag{18}$$

$$= \left(1 - 2\alpha\nabla\alpha \cdot \nabla - \alpha^2\nabla^2\right)^{-1}\mathbf{v}, \tag{19}$$

and the momentum Eq. (3) includes the additional term $|\nabla\mathbf{u}|^2\alpha\nabla\alpha$ (see discussion in [11]). The effect of including these extra terms is beyond the scope of this work. A comprehensive study would compare a Helmholtz inversion smoothing, which could use (19) directly, with a filtering method, where a proxy value of α at each gridcell would be used to compute gradients of α. The additional terms required by variable α may play an important role near the boundary where velocity gradients are large, and will be the topic of a future study.

It makes physical sense for α to go to zero as one approaches the boundary; specifically, at any point the length scale α should be less than the distance to the boundary, because the Lagrangian displacement must be less than this distance. Support for this argument comes from direct numerical simulations of a turbulent channel flow, where all scales down to the Kolmogorov microscale were resolved, and no turbulence closure scheme was used. Measurements of α in these simulations indicate that α vanishes near the wall but approaches a constant away from the wall, where the turbulence is isotropic [12].

5. Eigenvalue analysis of the filter

Most theoretical results for the LANS-α model depend on the two velocities being related through the Helmholtz operator. We can replace the Helmholtz operator with filters if two critical properties are preserved. These are that (1) the filtered velocity $\mathbf{u}$ must be smoother than the unfiltered velocity $\mathbf{v}$, and (2) that the energy,

$$E = \frac{1}{2}\int \mathbf{u} \cdot \mathbf{v} d^3x \tag{20}$$

must be well defined (e.g., non-negative). When both of these criterion are satisfied, the filter may be used in place of the Helmholtz inversion operator in the LANS-α model. In this section we will review why the

Helmholtz inversion satisfies these criterion, and then proceed to show that the filter described in the previous section satisfies them as well.

The Helmholtz inversion is easily shown to be a smoothing operator using Fourier analysis,

$$\hat{\mathbf{u}}(k,l) = \frac{\hat{\mathbf{v}}(k,l)}{1+\alpha^2 k_h^2}, \tag{21}$$

where $\hat{\mathbf{u}}$ and $\hat{\mathbf{v}}$ are the Fourier coefficients, (k,l) are horizontal wavenumbers and $k_h^2 = k^2 + l^2$. For scales much larger than α the Fourier coefficients are unaffected, because $1/k_h \gg \alpha$ so $\alpha^2 k_h^2 \ll 1$ and $\hat{\mathbf{u}} \sim \hat{\mathbf{v}}$. For scales at or smaller than α the Fourier coefficients are strongly damped because $\alpha^2 k_h^2 > 1$ so that $\hat{\mathbf{u}} < \hat{\mathbf{v}}$. The energy for the Helmholtz inversion is also well defined, because (20) can be rewritten, using integration by parts, as

$$E = \frac{1}{2}\int |\mathbf{u}|^2 + \alpha^2 |\nabla \mathbf{u}|^2 d^3x, \tag{22}$$

which is nonnegative.

We next show that these properties also hold for our filters by using the eigenvalue problem applied to the discrete filter-operator matrix A. To illustrate this process, first consider the simple three-wide filter in 1D shown in (13). This filter can be written as a tridiagonal matrix,

$$A = \frac{1}{a+2b}\begin{bmatrix} a & b & 0 & \cdots & 0 \\ b & a & b & 0 & \cdots \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & b & a \end{bmatrix}. \tag{23}$$

so that a vector can be filtered using $\mathbf{u} = A\mathbf{v}$. The eigenvalues of A vary as a function of b. If $b = 0$, A is the identity matrix and the eigenvalues are all one. As b increases, the range of eigenvalues increases but always remains less than one (Fig. 7). The smallest eigenvalues are associated with the highest wavenumber modes (Fig. 8), indicating that the most oscillatory modes are damped the strongest, and that the matrix A is indeed a smoothing operator.

We now evaluate the eigenvalues of the full filter (Fig. 6) in 2D. On an $n \times n$ grid the filter operator is an n^2 matrix (Fig. 9). Here we consider Dirichlet boundary conditions of zero, but the results are similar for periodic boundaries. As in the 1D case, all eigenvalues are less than one. We have chosen weights so that most eigenvalues are near zero but positive (Fig. 10). The most oscillatory modes have very small eigenvalues, so that these are strongly damped (Fig. 11). This ensures that the filter is a smoothing operator.

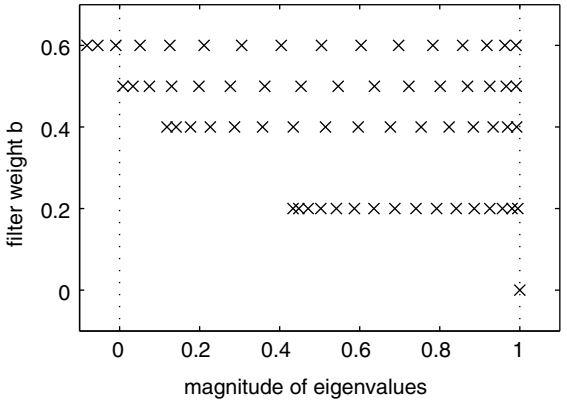

Fig. 7. Eigenvalues of the 1D, width-three filter operator using 16 gridpoints. When $b = 0$ the operator is simply the identity, so all eigenvalues are one. As b increases the eigenvalues spread out below one, indicating that some eigenvectors contract. Eigenvalues are never larger than one, so no eigenvectors increase. When $b > 0.5$ there are negative eigenvalues.

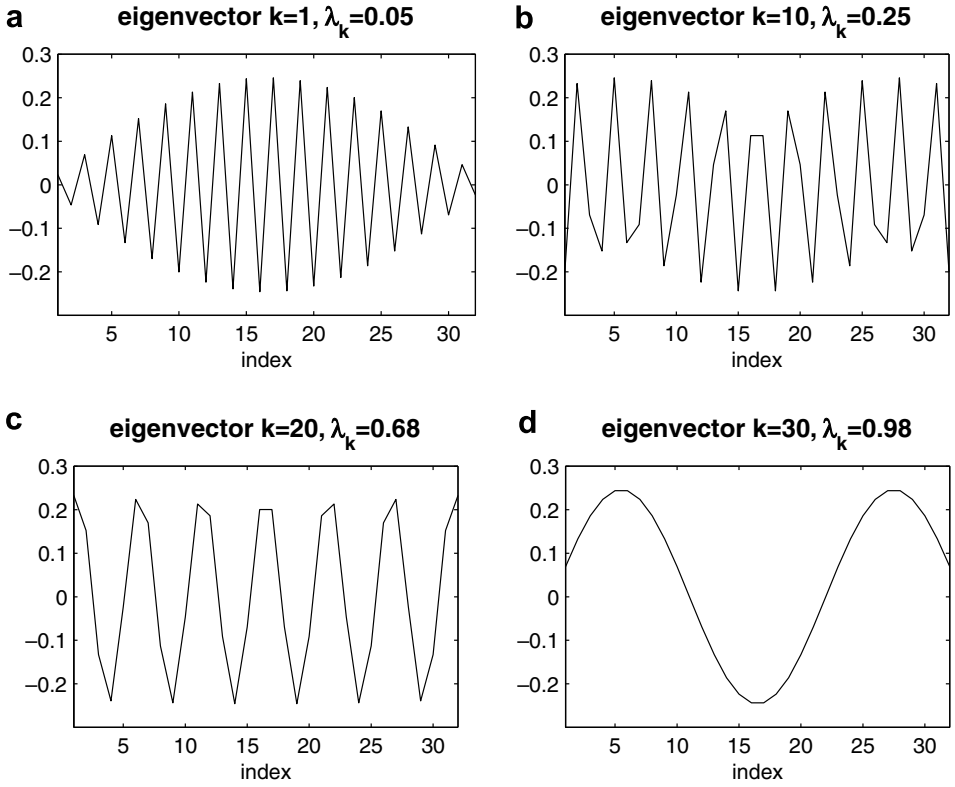

Fig. 8. Eigenvectors of the 1D, width-three filter operator using 32 gridpoints. High wavenumber modes (a and b) have eigenvalues near zero, while low wavenumber modes (c and d) have eigenvalues near one. This shows that the filter is a smoothing operator, because the highest wavenumbers are strongly damped.

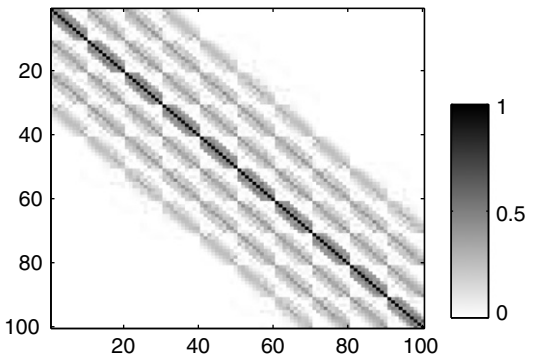

Fig. 9. Matrix representation of the 2D, 7×7 filter using 10×10 gridpoints. The full matrix is 100 elements square, while each block is 10 square. This matrix is not yet normalized by the factor $1/(a + 2b + 2c + 2d + 2e)^2$, so weights range between 0 and 1.

The last task is to ensure that the energy defined in (20) is well defined. The matrix A is positive definite if $\mathbf{v}^T A\mathbf{v} > 0$ for all nonzero vectors $\mathbf{v}$. Since the filtered velocity $\mathbf{u} = A\mathbf{v}$, this is equivalent to $\mathbf{v} \cdot \mathbf{u} > 0$, which means that the global energy (20) is nonnegative and therefore well defined. A matrix is positive definite if all of its eigenvalues are positive. Thus a criterion for our filter is that its minimum eigenvalue is positive. This leads to a process for choosing each filter weight: plot the minimum eigenvalue as a function of the filter

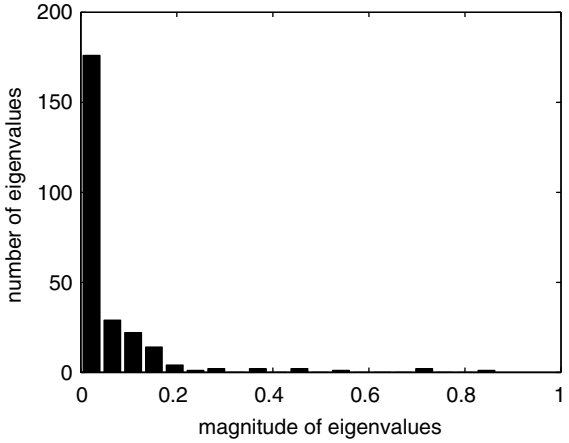

Fig. 10. Distribution of eigenvalues of the 2D, 9×9 filter operator using 16×16 gridpoints. Filter weights are as shown in Fig. 5. The majority of eigenvalues are less than 0.1, indicating that most eigenmodes are stongly damped.

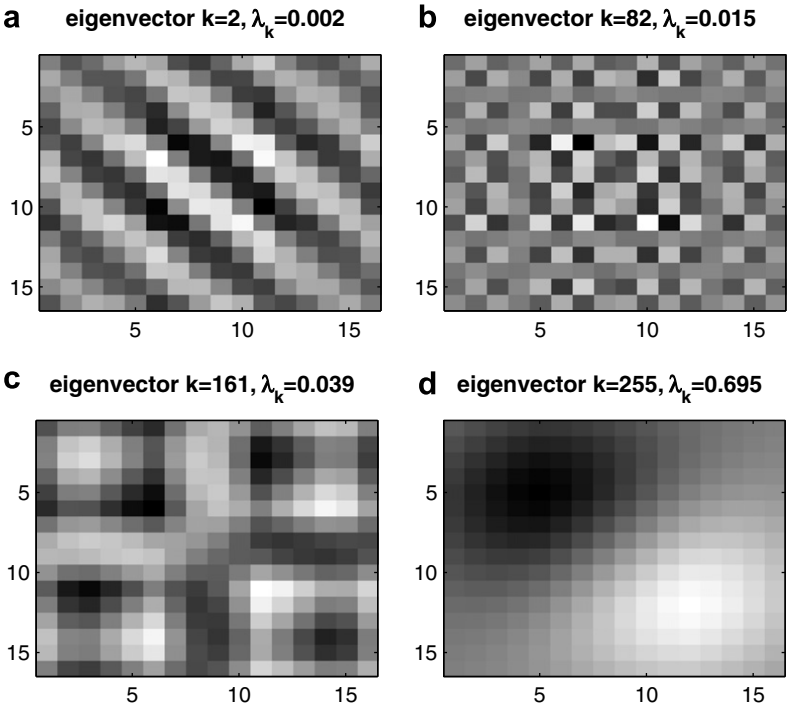

Fig. 11. Eigenmodes of the 2D, 9×9 filter operator using 16×16 gridpoints. As in the 1D case, highly oscillatory modes (a) have eigenvalues near zero and so are strongly damped, while less oscillatory modes (d) have eigenvalues near one and so are only weakly damped. This shows that the filter we have implemented is a smoothing operator.

weight, and choose the largest weight that guarantees stability. There is no reason to choose a smaller value, because then the filter smooths less. Fig. 12 shows that this process produces the same restrictions on filter weights as the pressure–velocity instability in the previous section. In fact, the two methods of analysis agree so well that the pressure–velocity instability appears to be a physical manifestation of the poorly defined energy. Numerical experiments where the filter weights were varied confirm that the simulations are unstable when the energy is poorly defined.

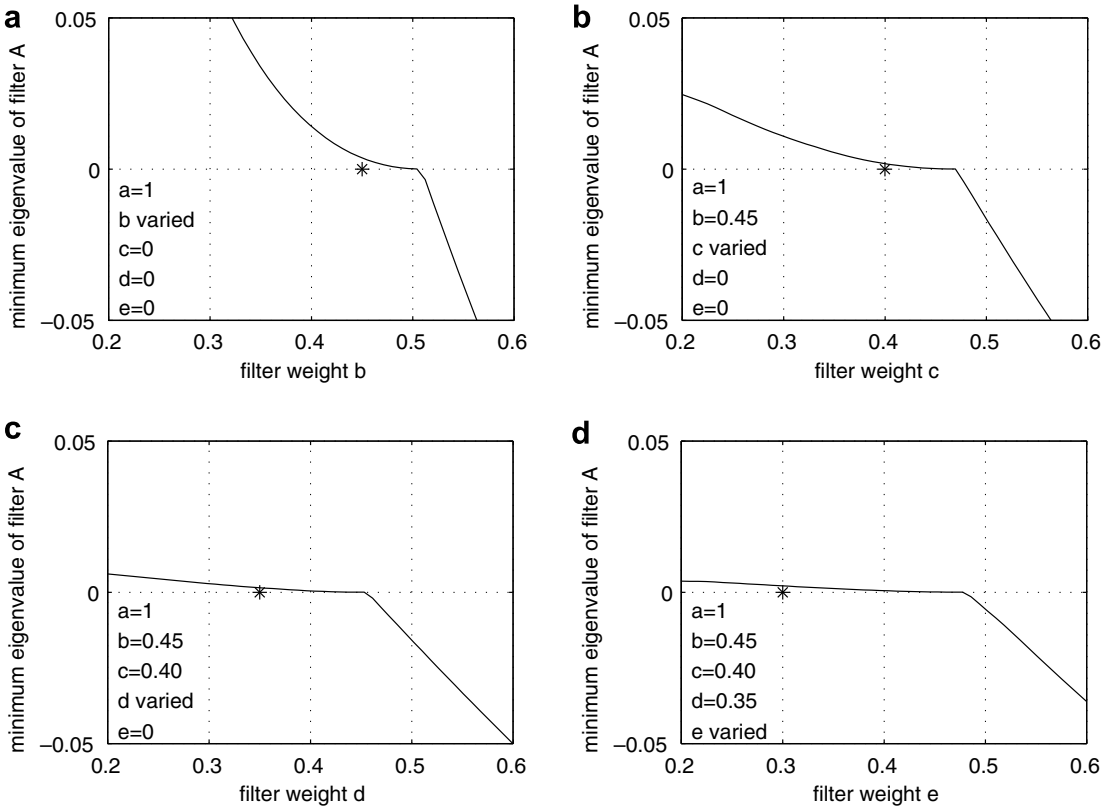

Fig. 12. Minimum eigenvalue of the 2D, 9×9 filter operator using 16×16 gridpoints, as a function of each filter weight. If all eigenvalues are strictly positive then the operator is positive definite, and the energy is well defined. The value of each filter weight is chosen (starred points) to satisfy this criterion.

6. Green's function of the Helmholtz inversion

In order to make a more quantitative comparison of the Helmholtz inversion and filters used in the smoothing operations, we have computed the discrete Green's function of the Helmholtz inversion. The Green's function can be considered as the response of a system when a unit point source is applied to the system. In this case the unit source is at a single grid cell, where the rough velocity $v_{i,j} = 1$, while $\mathbf{v} = 0$ at all other gridcells. This gridcell is located far from land to minimize boundary influences. The smooth velocity is then computed using the Helmholtz inversion (7) in POP-α in the standard way, which is with a preconditioned conjugate gradient solver. The resulting array, $\mathbf{u}$, is the Green's function, and can be compared directly to the filter stencil. In an actual simulation, the smooth velocity field is equivalent to a linear combination of these Green's functions applied to each gridcell of the rough velocity field.

The continuum form of the Helmholtz inversion represents a global operation, while the filter is fundamentally local. In our application, the discrete version of the Helmholtz inversion is also local, because it is limited by the number of iterations in the conjugate gradient solver, which communicates with only the nearest lateral neighbors on each iteration. The solver typically requires 8–10 iterations, and so the discrete Green's function is 17–21 gridcells wide, as compared to a maximum of nine for the filter. The fact that the filter stencil is smaller and requires less communication among processors is exactly why the filter is more efficient.

A direct comparison of the two shows that the Helmholtz inversion has a much larger domain of influence, although the coefficients drop off exponentially away from the central gridcell (Figs. 13 and 14). In contrast, the filter stencil is compact, and the filter weights were chosen to fall off as gradually as possible within the stability criterion of the pressure–velocity instability (Section 3). This results in filter weights that decrease linearly after $i + 1$, rather than logarithmically as with the Helmholtz inversion coefficients (Fig. 14). The goal of

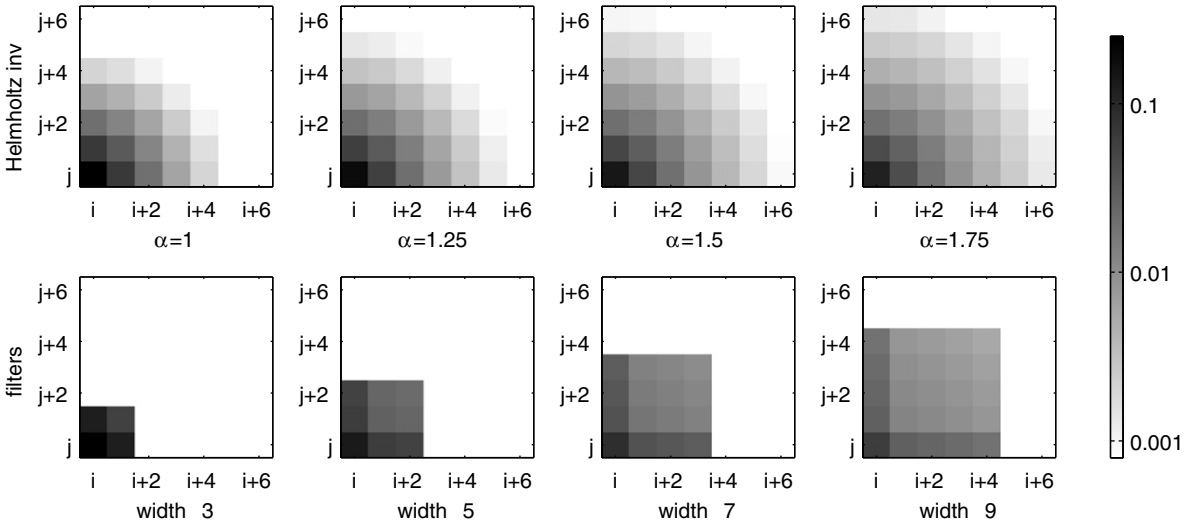

Fig. 13. Green's function of the Helmholtz inversion (top) and filter weights (bottom). The Helmholtz inversion has a larger domain of influence, while the filter is more compact, resulting in fewer computations. Only the upper-right quadrant of the stencil is shown, and values are not normalized in any way. Note the shading uses a log scale.

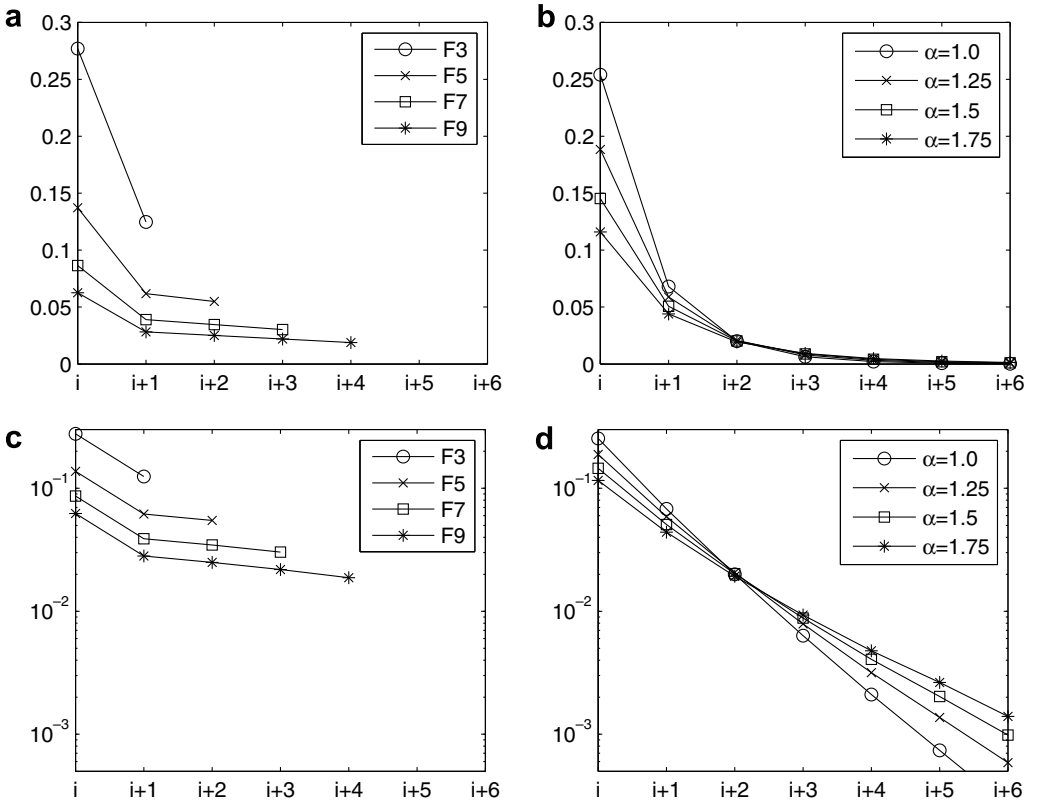

Fig. 14. Unnormalized filter weights (left) and Green's function of the Helmholtz inversion (right), as measured laterally in a 2D stencil. The Helmholtz inversion has non-zero coefficients far away from the central point i, but they drop off exponentially. The filter has a much smaller domain of influence, but the coefficients are higher at $i+3$ and $i+4$. (a) Filter weights (linear scale). (b) Helmholtz inv. coef. (linear scale). (c) Filter weights (log scale). (d) Helmholtz inv. coef. (log scale).

the filter design is to conduct the greatest averaging with the smallest number of computation and communication steps, which means the weights must each be as large as possible, while the stencil must be as small as possible.

With both the filter and the Helmholtz inversion, the central coefficient drops as the smoothing parameter increases. This is because the sum of all coefficients is one, and the data shown in Figs. 13 and 14 are not normalized by the central weight, as in previous sections. As the smoothing parameter increases, weights further from the central point must increase, and so the central weight decreases.

This exercise illustrates the point that there is not a one-to-one correspondence between a particular value of α and a filter width, because the associated operators produce distributions of weights of somewhat different shape. The effects of the two methods of smoothing are similar, but evidently not identical. Accordingly, the resulting turbulence statistics can be expected to be similar, but not identical.

7. Results

The model problem is a zonally periodic channel with a deep-sea ridge, eastward wind forcing, and surface thermal forcing that restores SST to a smooth profile from 12 °C in the north to 2 °C in the south, as described in [1]. This configuration is an idealization of the Antarctic Circulpolar Current, and was designed to induce baroclinic instability, where isopycnals are tilted from the horizontal meridionally by the surface thermal and wind forcing. If present, eddies in the flow advect heat meridionally, with the net effect that the isopycnals are less tilted; in this way, the eddies convert the potential energy of the tilted isopycnals to the kinetic energy of the eddies themselves. Simulations of standard POP at three resolutions show this effect. In the lowest resolution simulation (0.8, see Table 1) the Rossby Radius of deformation is not resolved, and no eddies are present. As the resolution doubles (0.4) and doubles again (0.2) more eddies exist, and the isopycnals are less tilted (Fig. 15).

The flatter isotherms at higher resolution create a sharper thermocline and colder temperatures below the thermocline (Fig. 16). (In this model configuration, salinity is constant, so the isotherms shown in the figures coincide with isopycnals.) Higher-resolution simulations of standard POP also show progressively higher kinetic energy (Fig. 17) and higher eddy kinetic energy (Fig. 18) as more eddies are resolved (see Table 2).

POP-α simulations are more energetic and have more eddy activity than standard POP simulations at the same resolution. The strength of the eddy activity is controlled by how much smoothing is done on the rough velocity $\mathbf{v}$: when a Helmholtz inversion is used, a larger α smooths more; when a filter is used, wider filter

Table 1
Model parameters for experiments discussed in this paper, fw is the filter width; grid is the number of gridpoints in (x, y, z); lon is the longitudinal grid-cell width; and lat is the latitudinal grid-cell width, in degrees

Name	Model	Smoothing	α	fw	Grid	lon	lat
0.8	POP	–	–	–	$40 \times 40 \times 34$	0.8	0.4
0.4	POP	–	–	–	$80 \times 80 \times 34$	0.4	0.2
0.2	POP	–	–	–	$160 \times 160 \times 34$	0.2	0.1
0.1	POP	–	–	–	$320 \times 320 \times 34$	0.1	0.05
0.8H	POP-α	Helmholtz	$\Delta x - 2\Delta x$	–	$40 \times 40 \times 34$	0.8	0.4
0.4H	POP-α	Helmholtz	$\Delta x - 1.75\Delta x$	–	$80 \times 80 \times 34$	0.4	0.2
0.8F3	POP-α	Filter	–	3	$40 \times 40 \times 34$	0.8	0.4
0.8F5	POP-α	Filter	–	5	$40 \times 40 \times 34$	0.8	0.4
0.8F7	POP-α	Filter	–	7	$40 \times 40 \times 34$	0.8	0.4
0.8F9	POP-α	Filter	–	9	$40 \times 40 \times 34$	0.8	0.4
0.4F3	POP-α	Filter	–	3	$80 \times 80 \times 34$	0.4	0.2
0.4F5	POP-α	Filter	–	5	$80 \times 80 \times 34$	0.4	0.2
0.4F7	POP-α	Filter	–	7	$80 \times 80 \times 34$	0.4	0.2
0.4F9	POP-α	Filter	–	9	$80 \times 80 \times 34$	0.4	0.2

The names are a concatenation of the meridional resolution, the type of smoothing, and the value of α or filter width. Values of α tested for 0.8H and 0.4H are at intervals of $0.25\Delta x$.

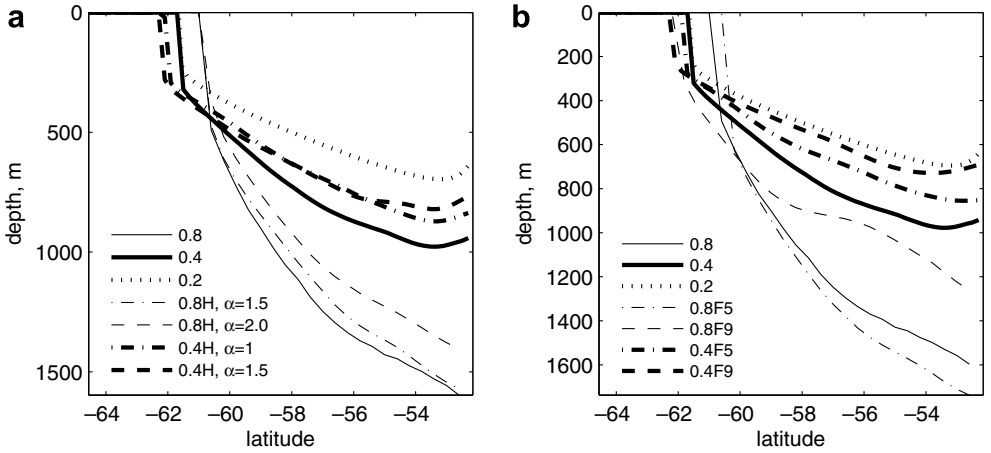

Fig. 15. Depth of 6 °C isotherm of potential temperature using the Helmholtz inversion (a) and the filter (b) to smooth the velocity. Higher resolution simulations of standard POP (dotted line) have isotherms which are less tilted than lower resolution simulations (solid lines). As α or filter size increases, the POP-α simulations show flatter isotherms, and approach a doubling of resolution using standard POP.

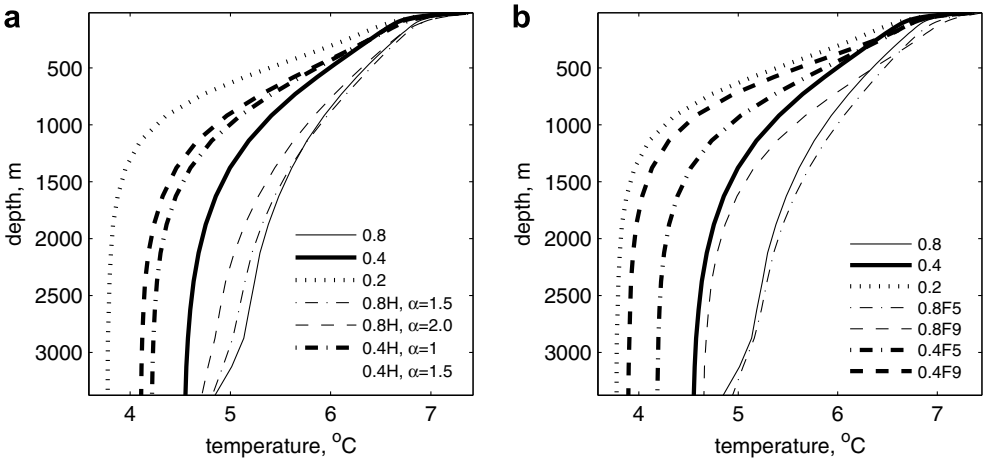

Fig. 16. Horizontally averaged potential temperature versus depth using the Helmholtz inversion (a) and the filter (b) to smooth the velocity. Higher-resolution simulations of standard POP (dotted line) are cooler and have a sharper thermocline than lower resolution simulations (solid lines). Again, POP-α simulations approach a doubling of resolution using standard POP.

stencils smooth more. In either case, Figs. 15–18 show that POP-α can produce kinetic and eddy kinetic energy that tend progressively to a doubling of resolution of standard POP or higher as the smoothing is increased.

It may seem surprising that the kinetic energy increases as the smoothing parameter is increased. This occurs because the smoothing is of the advective velocity **u**, allowing stronger and more numerous coherent structures to exist near the grid-scale in the rough velocity **v**. The result is that both smooth and rough velocities become more energetic as α (or filter width) increases [13]. Simulations using $\alpha = 2.0\Delta x$ and $\alpha = 1.75\Delta x$ for cases 0.8H and 0.4 H, respectively, are excessively energetic, as judged by higher-resolution standard POP statistics and the prominence of small-scale variability in the velocity fields (not shown). Clearly, α is a modeling parameter that must be chosen carefully, guided by the results of high resolution test cases without LANS-α. Further increases in α resulted in unstable simulations.

In the additional measure of temperature distributions, POP-α simulations tend towards higher-resolution standard POP distributions (Figs. 15 and 16). Here POP-α using a large filter (F9) more closely resembles the doubled-resolution POP temperature distribution than any simulations performed using the Helmholtz

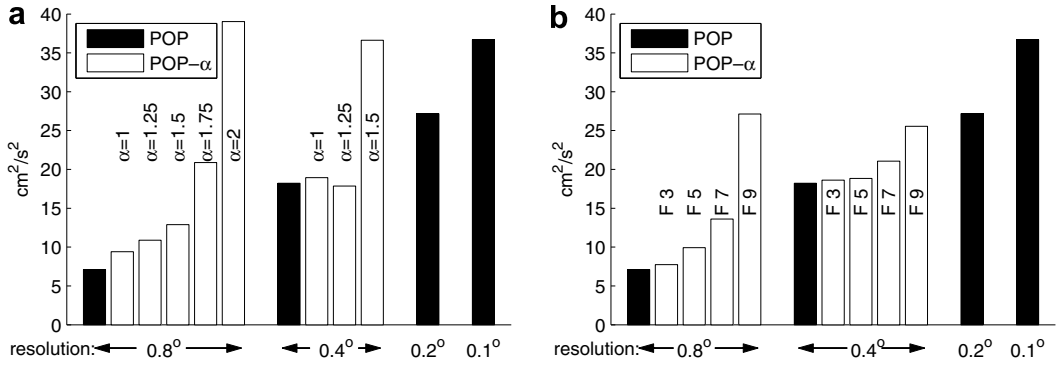

Fig. 17. Kinetic energy for all simulations. As the resolution increases with standard POP, the kinetic energy increases. Kinetic energy also increases using POP-α at fixed resolution, when α is increased using the Helmholtz inversion (a) or the stencil width of the filter is increased (b). The 0.4^o $\alpha = 1.75$ case has a value of 63 cm^2/s^2, but is not shown for scaling purposes. Each value was calculated by averaging over the entire domain, and in time after 500 years.

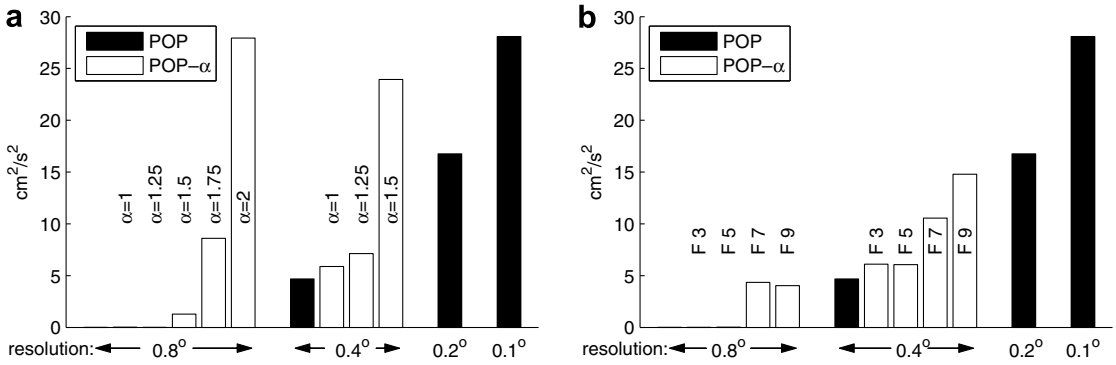

Fig. 18. Same as Fig. 17 but for eddy kinetic energy. Note that at the lowest resolution (0.8) the eddy kinetic energy of standard POP is indistinguishable from zero on this scale. The 0.4^o $\alpha = 1.75$ case has a value of 49 cm^2/s^2, but is not shown for scaling purposes. (a) Helmholtz inversion (b) filters.

Table 2
Minimum steps/day and the resulting clock time for various algorithms and resolutions

Algorithm	Cor	Steps/day			Clock time		
		Resolution			Resolution		
		0.8	0.4	0.2	0.8	0.4	0.2
POP	Implicit	12	22	40	4.16	21.3	195
POP	Explicit	20	32	52	5.82	28.4	212
POP-α F3	Explicit	18	24		5.09	22.7	
POP-α F5	Explicit	18	24		5.68	22.7	
POP-α F7	Explicit	18	24		6.18	25.6	
POP-α F9	Explicit	18	24		6.89	27.1	
POP-α Helm.	Explicit	12	18		10.21	33.0	

The second column states whether the barotropic Coriolis term is implicit or explicit. Clock time is in processor-hours per simulated decade.

inversion. This shows that, while both the Helmholtz inversion and the filter produce a velocity field **u** which is smoother than **v**, they do not produce identical results in all statistics.

Because POP-α has two velocities, some details are necessary to explain how these diagnostic quantities were computed. The kinetic energy is

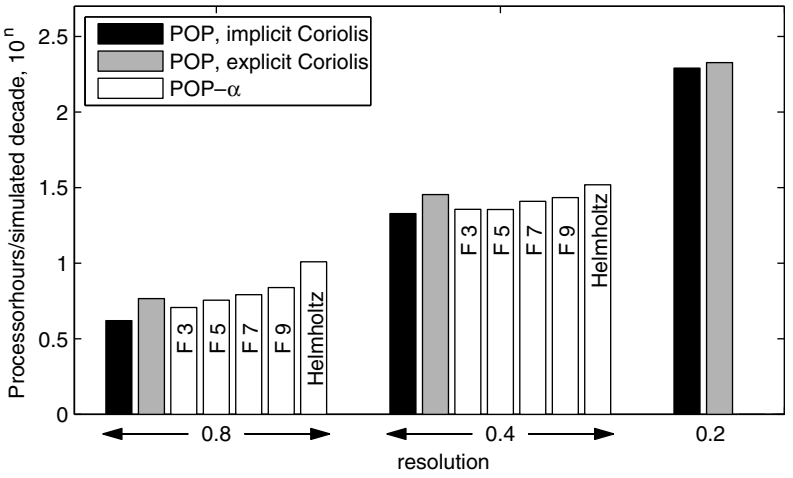

Fig. 19. Timing data from various implementations of POP and POP-α (note log scale). The POP-α filter algorithms (F3–F9) are much cheaper than a doubling of resolution using standard POP. Using a Helmholtz inversion to smooth is more expensive than the filters. All POP-α algorithms use explicit Coriolis discretization.

$$\mathrm{KE} = (u_1^2 + u_2^2)/2, \quad KE_\alpha = (u_1 v_1 + u_2 v_2)/2 \tag{24}$$

for POP and POP-α, where the subscripts indicate horizontal components. The eddy kinetic energy is

$$\mathrm{EKE} = \left(\overline{(u_1')^2} + \overline{(u_2')^2}\right)/2, \quad \mathrm{EKE}_\alpha = \left(\overline{u_1' v_1'} + \overline{u_2' v_2'}\right)/2, \tag{25}$$

where each variable is the sum of a mean and perturbation, e.g. $u_1 = \overline{u_1} + u_1'$, and the overbar indicates a five-year time-averaged mean.

There would seem to be three possible ways one could compute the kinetic energy for POP-α: using the products u^2, uv, or v^2. The product of the smooth and rough velocities, uv, is used because this the conserved kinetic energy that emerges from the derivation of the LANS-α equation [6,2]. For comparison, kinetic and eddy kinetic energies were also computed using u^2 and v^2. The product of the smooth velocities is smaller than uv, and the product of the rough velocities is larger (typically by 10% to 20%—see [13]). However, the general trends shown in Figs. 17 and 18 still hold if u^2 or v^2 are used instead of uv.

The computation time required for POP-α is slightly longer than standard POP due to the smoothing operations and additional nonlinear term, but is much less than a doubling of resolution in standard POP (Fig. 19). Computation increases as the filter width increases from three to nine; smoothing using the Helmholtz inversion is even slower than the nine-wide filter.

The standard POP algorithm may use an explicit or implicit discretization for the Coriolis term. Typically the implicit discretization is chosen because it allows longer timesteps to be taken, resulting in faster simulations. As described in [1], the POP-α algorithm must use the explicit implementation of the Coriolis term. Fortunately, POP-α can use longer timesteps than the standard POP algorithm with explicit Coriolis [1,14]. This explains why most of the POP-α simulations in Fig. 19 are faster than POP with an explicit Coriolis term, and just a bit slower than POP with an implicit Coriolis term.

8. Conclusions

An alpha-model is a class of turbulence model in which the velocity field has been smoothed in the Lagrangian of Hamilton's principle. This paper presents an assessment of two methods of smoothing this velocity field: the Helmholtz inversion and filters. The Helmholtz inversion requires an iterative conjugate gradient routine for each smoothing. The filter is simply a weighted average of nearby neighbors, and results in simulations that are 20 to 50% faster than those with the Helmholtz inversion. This disparity is expected to increase with larger domains and more processors, because more iterations are required for convergence in the conjugate gradient

method as the problem-size increases ([15], p. 530). The savings in computation time provides a strong motivation to choose a local filter over the global Helmholtz inversion.

Both the Helmholtz inversion and the filters produce results expected of the LANS-α turbulence model: statistics such as temperature profiles, kinetic and eddy kinetic energy tend towards statistics of higher-resolution non-alpha simulations. The parameter α is a length-scale that controls the amount of smoothing, and thus the strength of the turbulence model. Analogously, as the filter width varies from three to nine, the smoothing increases and the effects of LANS-α are stronger. For a particular value of α or filter width, the methods perform differently in our metrics: the filter has better temperature profiles (as judged by higher-resolution simulations), while the Helmholtz inversion has higher kinetic and eddy kinetic energy in most cases. A Green's function analysis of the discrete Helmholtz inversion operator revealed the details of the difference between the two smoothing methods. The Helmholtz inversion has a larger domain of influence, but the coefficients drop off more rapidly than the filters' do.

The filter stencil weights must be chosen carefully, otherwise POP-α will be unstable. If the outer weights are too large, the smooth velocity will be of the opposite sign as the rough velocity for the highest wave-number modes. When this occurs the pressure gradient, which is calculated from the smooth velocity, is of the wrong sign. The physically correct relationship between pressure gradient force and rough velocity is lost, and the mode grows. The conditions that bring about this pressure–velocity instability were identified exactly, and then verified by numerical simulations. We found that the first neighboring weight must be less than one-half the central weight, and each consecutive neighbor must be less than the one before. In 2D, this pattern is squared to make a stable stencil. The full square was required; leaving off the corners to make diamond-shaped stencils resulted in unstable simulations.

The filter design used in this study is more robust than the Helmholtz inversion. When α was too high (i.e the smoothing too strong) in the Helmholtz inversion method, the kinetic energy grew without bound and the simulation was unstable. This is typical of LANS-α models. Unstable simulations occurred when $\alpha > 2\Delta x$. For the filter, all stencils tested (up to 9×9) were stable. It is possible that larger filter widths would also produce highly energetic, unstable simulations.

It is important that the filters chosen are actually smoothing operators, and that the energy is well defined. Both of these properties were verified using an eigenvalue analysis of the filter, discretized as a matrix operator. In fact, the criterion on filter weights that guaranteed well-defined energy turned out to be the same as the weight limits to prevent the pressure–velocity instability. Thus it appears that the instability is a physical manifestation of poorly-defined energy in the equation set.

The goal of any turbulence model is to capture the effects of higher-resolution simulations without paying the computational price of running at that higher resolution. We have shown that our implementation of the LANS-α model in a primitive-equation ocean–climate model accomplishes this goal. Specifically, statistics in POP-α simulations such as kinetic energy, eddy kinetic energy, and temperature profiles are similar to standard POP simulations with double the resolution. But the LANS-α simulations only required 27% more computing time than standard POP, as compared to a factor of nine increase to double the resolution of standard POP (for the 0.4 case).

We are currently performing simulations to compare LANS-α with other turbulence models, including constant coefficient hyperviscosity, Gent-McWilliams isopycnal tracer mixing [16], the Leray model [2], and the Simplified Bardina model [17]. These results, to be published in the future, will consider the merits of LANS-α relative to competing models in the context of primitive-equation ocean models. We also intend to test LANS-α in ocean-basin and global simulations.

The comparison of the Helmholtz inversion and the filter in this study has shown that the filter is both cheaper and more robust than the Helmholtz inversion, while producing similar trends in the turbulence statistics. Thus our future work with the POP-α model will use local filters in place of the global Helmholtz for the majority of our simulations.

Acknowledgments

We thank T. Ringler for his suggestions on the Green's function analysis; E. Titi for valuable insight on the important properties of filters; B. Geurts for helpful conversations on his experience with filters in LES

models; and A. Fournier for advice on ways to compare the filters and Helmholtz inversion. This work was carried out under the auspices of the National Nuclear Security Administration of the U.S. Department of Energy at Los Alamos National Laboratory under Contract No. DE-AC52-06NA25396.

References

[1] M.W. Hecht, D.D. Holm, M.R. Petersen, B.A. Wingate, Implementation of the LANS-α turbulence model in a primitive equation ocean model, J. Comput. Phys. 227 (2008) 5691–5716.
[2] D.D. Holm, Fluctuation effects on 3D Lagrangian mean and Eulerian mean fluid motion, Physica D133 (1999) 215–269.
[3] C. Foias, D.D. Holm, E.S. Titi, The Navier–Stokes-alpha model of fluid turbulence, Phys. D Nonlinear Phenom. 152 (2001) 505–519.
[4] J. Domaradzki, D. Holm, Navier–Stokes alpha model: LES equations with non-linear dispersion, ERCOFTAC Bull. 48.
[5] B.J. Geurts, D.D. Holm, Leray and LANS-α modeling of turbulent mixing, J. Turbulence 7 (2006) 10.
[6] D. Holm, J. Marsden, T. Ratiu, The Euler–Poincare equations in geophysical fluid dynamics, in: Isaac Newton Institute Proceedings, 1998.
[7] R. Smith, P. Gent, Reference manual for the Parallel Ocean Program (POP), 2002. <http://climate.lanl.gov/source/projects/climate/Models/POP>.
[8] R.D. Smith, J.K. Dukowicz, R.C. Malone, Parallel ocean general circulation modeling, Physica D 60 (1992) 38–61.
[9] J.K. Dukowicz, R.D. Smith, Implicit free-surface method for the Bryan–Cox–Semtner ocean model, J. Geophys. Res. 99 (1994) 7991–8014.
[10] J.E. Marsden, S. Shkoller, Global well-posedness for the Lagrangian averaged Navier–Stokes (LANS-alpha) equations on bounded domains, Phil. Trans. Roy. Soc. A 359 (2001) 1449–1468.
[11] D.D. Holm, V. Putkaradze, P.D. Weidman, B.A. Wingate, Boundary effects on exact solutions of the Lagrangian-averaged Navier–Stokes-alpha equations, J. Stat. Phys. 113 (5–6) (2003) 841–854.
[12] H. Zhao, K. Mohseni, A dynamic model for the Lagrangian-averaged Navier–Stokes-α equations, Phys. Fluids 17 (2005) 5106.
[13] M.W. Hecht, D.D. Holm, M.R. Petersen, B.A. Wingate, LANS-α and Leray turbulence parameterizations in primitive equation ocean modeling, J. Phys. A, submitted for publication.
[14] B.A. Wingate, The maximum allowable time step for the shallow water alpha model and its relation to time-implicit differencing, Monthly Weather Rev. 132 (12) (2004) 2719–2731.
[15] G.H. Golub, C.F.V. Loan, Matrix computations, 3rd ed., Johns Hopkins University Press, 1996.
[16] P.R. Gent, J.C. McWilliams, Isopycnal mixing in ocean circulation models, J. Phys. Oceanogr. 20 (1) (1990) 150–155.
[17] Y. Cao, E.M. Lunasin, E.S. Titi, Global well-posedness of the three-dimensional viscous and inviscid simplified Bardina turbulence models, unpublished, 2006. URL <http://www.citebase.org/abstract?id=oai:arXiv.org:physics/0608096>.