Journal of Computational Physics **182,** 191–212 (2002)
doi:10.1006/jcph.2002.7161

A Super-Grid-Scale Model for Simulating Compressible Flow on Unbounded Domains

Tim Colonius and Hongyu Ran

Division of Engineering and Applied Science, California Institute of Technology, Pasadena, California 91125
E-mail: colonius@caltech.edu

Received January 8, 2002; revised June 20, 2002

A new buffer region (absorbing layer, sponge layer, fringe region) technique for computing compressible flows on unbounded domains is proposed. We exploit the connection between coordinate mapping from bounded to unbounded domains and filtering of the equations of motion in Fourier space in order to develop a model to damp flow disturbances (advective and acoustic) that propagate outside an arbitrarily defined near field. This effectively simulates a free-space boundary condition. Damping the solution in the far field is accomplished in a simple and effective way by applying a filter (similar to that used in large-eddy simulation) on a mesh in Fourier space, followed by a secondary filtering of the equations on the physical grid and implementation of a model for the unresolved scales. The final form of the buffer region is given in real space, independent of any discretization of the equations. Here we use a dealiased, Fourier spectral collocation method to demonstrate the efficacy of the buffer region for several model problems: acoustic wave propagation, convection of a finite-amplitude vortex, and a viscous starting jet in two dimensions. The results compare favorably to previous nonreflecting and absorbing boundary conditions.

1. INTRODUCTION

Many methods have been proposed for solving fluid flow equations on infinite or semi-infinite domains. The domain may be mapped to a finite computational domain, or it may be truncated at an artificial boundary. Mapping is problematic when disturbances become poorly resolved as they propagate into regions of coarser mesh. In central finite-difference (FD) schemes the disturbances can be shown to reflect from the mesh stretching, due to the dispersive nature of the FD approximation [33]. Spectral methods for mapped infinite (or highly stretched) domains are prone to similar artifacts due to aliasing of poorly resolved disturbances. In either case these problems are particularly acute for aeroacoustic computations where small-amplitude radiation from the flow may easily be swamped by even small artificial reflections.

0021-9991/02 $35.00

For these reasons, it is often preferable to impose an artificial boundary condition (ABC). ABCs for multidimensional flows have been studied extensively for many years, but the theory is limited to linear inviscid cases. We summarize some relevant conclusions here. For flow linearized about a uniform state, there are a wide variety of techniques available that are accurate and robust. Methods based on a Fourier/Laplace decomposition of the disturbances into normal modes, originally proposed by Engquist and Majda [7], have been extensively developed for the linearized Euler equations [10, 11] and were extended to an arbitrarily high order of accuracy, in discretized form, by Rowley and Colonius [26]. Boundary conditions based on asymptotic solutions of the linearized Euler equations for radiation from a fixed (point) source have also been developed (see [30], and references therein), and these offer similar accuracy to the low-order Fourier/Laplace conditions. A popular and more robust alternative is the (nonlinear) one-dimensional characteristic-based boundary conditions developed by Thompson [31, 32] and extended to viscous flows by Poinsot and Lele [23]. Though these can lead to large reflections of multidimensional disturbances, they can be improved when used in combination with the buffer region techniques described below.

Another approach for the linearized Euler equations is the perfectly matched layer (PML), in which the equations are altered in a buffer region near the artificial boundary in order to absorb outgoing acoustic and advected disturbances prior to their interaction with the artificial boundary. Early problems with the PML approach [1, 30] have been resolved and well-posed (stable) layers can now be constructed for inviscid disturbances to a uniform mean flow [2]. A caveat is that the approach depends on the (uniform) mean flow being aligned with one of the coordinate directions near the boundary.

When these techniques are used in linear equations with nonconstant coefficients (i.e., spatial variation of the base flow) then there are additional reflections that scale with the amplitude of the exiting disturbances and the gradients of the base flow, and for nonlinear equations there are reflections that scale with the square of the exiting disturbances. These reflections have been shown to dominate the solution under conditions that are typical of a turbulent outflow boundary [4]. In order to mitigate these effects, a variety of *ad hoc* buffer region techniques have been proposed for use in combination with ABCs. These include the addition of an explicit damping term in a region near the boundary [3, 9, 14, 15], artificial acceleration of disturbances to supersonic speed [29], and a combination of these [8]. On the whole, these conditions can be effective and have led to accurate solution of a variety of problems. A pitfall is that they rely on multiple constants (and functions) that have only been determined by trial and error.

Another approach, which is most similar to the one proposed in the present paper, is to introduce substantial mesh stretching in a region near the computation boundary, and to provide damping or filtering in that region to attenuate disturbances as they propagate through the layer [4, 24]. The filtering technique is particularly effective because as the disturbances propagate through the stretched mesh they become progressively more poorly resolved, whereby they may be removed by a low-order discrete filter. Again, this method relies on several tunable parameters, such as the extent and rate of mesh stretching, and the extent to which filtering and/or damping are applied.

Finally, in incompressible flow, the *fringe region* technique has been used to simulate spatially evolving (inflow/outflow) problems using periodic (Fourier spectral) methods. The technique, originally due to Spalart [27], involves adding a forcing term local to the fringe region, which attempts to compensate for the periodicity by damping the difference

between the periodic solution, and an approximate solution at the inflow boundary (which is imposed). A detailed theoretical justification of Spalart's technique was recently provided by Nordström *et al.* [22], who also discuss some other recent application of fringe (or buffer) regions. A somewhat different approach for compressible boundary layers was developed by Guo *et al.* [12]. In a buffer region near the periodic boundary, Guo *et al.* *windowed* the governing equations to prevent recirculation of the flow across the periodic boundaries. Specialized techniques were then developed in order to specify inflow disturbances in the upstream buffer. The present method also uses a windowing of the governing equations but differs in the approach to correcting the solution for the presence of windowing.

Given the increased use of buffer regions in a variety of problems, our motivation is to develop a more general technique with fewer tunable parameters. In this paper, we provide a model with two interpretations that help to establish a firm basis for its application. The first interpretation is similar to the grid stretching and filtering approach, discussed above. A second interpretation shows that the problem is analogous in some ways to the problem of turbulence modeling in LES. This interpretation allows us to construct particular correction terms (including damping) that are to be added to the equations in the buffer, by analogy with a subgrid stress model. In the next section, we describe an analogy between ABCs and multiscale turbulence modeling. In Section 3, the governing equations and the model terms are presented. The relation of such models to previous buffer regions involving grid stretching and filtering is also discussed. In Section 4, the method is demonstrated on several model problems and compared to previous ABCs. Concluding remarks and possible extensions of the method are placed in Section 5.

2. FOURIER ANALYSIS AND THE ANALOGY BETWEEN ABC AND LES

Consider LES on a domain consisting of a periodic box of length $2\pi L$. If N Fourier modes are retained in the solution, the grid spacing is $\Delta x = 2\pi L/N$ and disturbances with a wavelength less than $2\Delta x$ (corresponding to wavenumbers $k > k_{max} = N/(2L)$) cannot be resolved. The impact of the unresolved scales must be modeled by supplying an appropriate subgrid stress (SGS) model. In the absence of a model, energy continuously accumulates at the highest wavenumbers supported by the mesh, and the periodic image (in Fourier space) of the unresolved disturbances is spuriously returned into the calculation, eventually contaminating the solution at all scales.

An analogous situation exists in physical space. Again considering a periodic domain, the physical size of the periodic box, $2\pi L$, sets a constraint on the size of the largest length scale that can be computed in the box. This length scale also determines the *resolution in Fourier space*, $\Delta k = \frac{1}{L}$. Disturbances that have finer structure in Fourier space correspond to features larger than $2\pi L$. Similarly waves propagating to the edge of the box, if unaltered, reappear in physical domain in a process that is exactly analogous to aliasing but that takes place in the physical domain.

Wave space aliasing (the subgrid modeling problem) can be mitigated by filtering high wave numbers and providing a model for unresolved disturbances, the LES approach. This is a smoothing operation in real space or, equivalently, a windowing operation in wave space. Apparently, physical space aliasing (the ABC problem) could in principle be mitigated by a similar process where the roles of wave space and real space are merely interchanged. That is, the smoothing operation is performed in Fourier space (equivalent to a windowing

operation in physical space) and a model should be provide for the unresolved (wave space) scales. We propose such a model that is inspired by an existing LES model. It turns out that the resulting model is also similar conceptually to grid stretching and damping disturbances in a buffer region near the boundary.

The following terminology may help to further convey the analogy with LES: an artificial boundary condition is, in a sense, a "super-grid-scale" model for length scales that are too large to fit on the mesh (or that propagate to distances too large to be represented on the mesh). A simulation utilizing such a boundary condition could be thought of as a "near-eddy simulation," in the sense that only near-field disturbances are directly resolved.

3. THE MODEL

3.1. Windowing

We begin with a conservation law in Cartesian coordinates, $\mathbf{x} \in \Re^3$,

$$\frac{\partial Q}{\partial t} + \frac{\partial F_j(Q)}{\partial x_j} = 0, \tag{1}$$

where $F_j(Q)$ are the flux vectors. Following the analogy between LES and ABCs from the last section, we wish to filter the Fourier transform of Eq. (1). We define a window function $H(\mathbf{x})$ and its Fourier transform, $\hat{H}(\mathbf{k})$, as

$$\hat{H}(\mathbf{k}) = \int_{-\infty}^{\infty} H(\mathbf{x}) e^{-i2\pi \mathbf{x}\cdot\mathbf{k}}\, d\mathbf{x}, \quad H(\mathbf{x}) = \int_{-\infty}^{\infty} \hat{H}(\mathbf{k}) e^{i2\pi \mathbf{x}\cdot\mathbf{k}}\, d\mathbf{k}. \tag{2}$$

For H we have in mind a typical windowing function from signal processing that is the product of three separate windowing operations in the three coordinate directions,

$$H(\mathbf{x}) = h_1\left(\frac{x_1}{L_1}\right) h_2\left(\frac{x_2}{L_2}\right) h_3\left(\frac{x_3}{L_3}\right), \tag{3}$$

where L_1, L_2, and L_3 are the characteristic widths of the window in the three coordinate directions. A sketch of a typical window and its Fourier transform are depcited in Fig. 1.

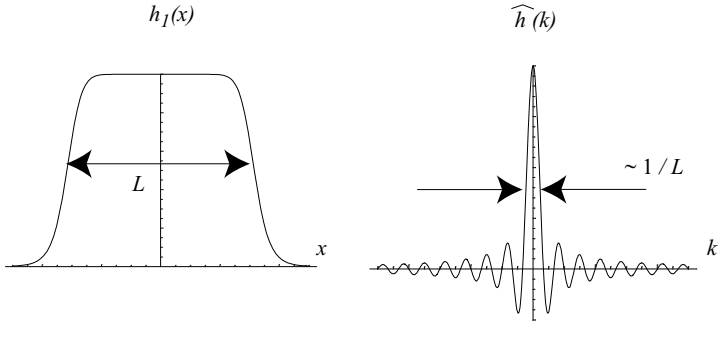

FIG. 1. Sketch of typical window function, h_j, and its Fourier transform, $\hat{h}_j$.

Multiplying Eq. (1) by the window we write

$$\frac{\partial \tilde{Q}}{\partial t} + H(\mathbf{x})\frac{\partial F_j(Q)}{\partial x_j} = 0, \tag{4}$$

where

$$\tilde{Q} = H(\mathbf{x})Q \quad \hat{\tilde{Q}} = \hat{H}(\mathbf{k}) * \hat{Q}(\mathbf{k}). \tag{5}$$

An asterisk, $*$, denotes convolution over $\Re^3$.

In order to close this equation, we must model the original dependent variable, Q, as a function of the windowed version, $\tilde{Q}$. Following a typical LES derivation (see, for example, Meneveau and Katz [19]) would lead to a supergrid-scale term on the right-hand side:[1]

$$\frac{\partial \tilde{Q}}{\partial t} + \frac{\partial F_j(\tilde{Q})}{\partial x_j} = \frac{\partial F_j(\tilde{Q})}{\partial x_j} - H\frac{\partial F_j(Q)}{\partial x_j}. \tag{7}$$

However, we follow here a different approach that is more akin to the idea of directly modeling the subgrid velocity field (e.g., [6, 28]). To this end, we return to Eq. (4) and pursue the singular process of forming

$$Q = H^{-1}\tilde{Q}, \tag{8}$$

or its equivalent in wave space,

$$\hat{Q} = \hat{H}^{-1} * \hat{\tilde{Q}}. \tag{9}$$

A direct regularization of the deconvolution in Eq. (9) has proven useful in the analogous (real space) version of this equation that arises in LES. Stolz and Adams [28], for example, have developed an LES model that utilizes an approximate (regularized) inverse filter,

$$G^{-1} \approx \sum_{\nu=0}^{N} (I - G)^{\nu}, \tag{10}$$

where $N = 5$ gave the best results in their numerical experiments.

For Eq. (9), the zeroth-order approximation is simply $\hat{H}^{-1} = 1$, which recovers the limit $L_j \to \infty$ (no windowing), for each j, and makes $Q = \tilde{Q}$. The *regularized* version of Eq. (4) is then, simply,

$$\frac{\partial \tilde{Q}}{\partial t} + H\frac{\partial F_j(\tilde{Q})}{\partial x_j} = 0. \tag{11}$$

[1] We note in passing that in the case of inviscid flow, for which $F_j(Q)$ is homogeneous of degree 1 (i.e., $F_j(\alpha Q) = \alpha F_j(Q)$), the right-hand side of Eq. (7) may be formally manipulated to obtain a closed-form equation for the windowed variables:

$$\frac{\partial \tilde{Q}}{\partial t} + \frac{\partial F_j(\tilde{Q})}{\partial x_j} = \frac{\partial H}{\partial x_j}F_j(Q) = \frac{1}{H}\frac{\partial H}{\partial x_j}F_j(\tilde{Q}). \tag{6}$$

Furthermore, Eq. (6) is not singular provided that the gradient of H decays to zero faster than H as $\mathbf{x} \to \infty$.

This is crude, amounting to no model at all for the supergrid. But together with further modeling, discussed in Section 3.3, it yields good results (Section 4). It seems that alternative models to those derived below could be based on attempts to model the right-hand side of Eq. (7) or to directly regularize Eq. (9) to higher order.

3.2. Comparison of Windowing and Grid Stretching

Before completing the model, it is instructive to compare Eq. (11) with the original conservation law (1), but under a suitable coordinate transformation that maps the infinite computational domain, $\mathbf{x}$, to a finite one, $\mathbf{y} \in ((-L_1, L_1), (-L_2, L_2), (-L_3, L_3))$. We recognize that the window functions, h_j, that have already been defined are suitable choices for such a mapping; i.e.,

$$\frac{dy_j}{dx_j} = h_j, \quad j = 1, 2, 3 \text{ (no summation implied)}, \tag{12}$$

which yields

$$\frac{\partial Q}{\partial t} + \sum_{j=1}^{3} h_j \frac{\partial F_j(Q)}{\partial y_j} = 0. \tag{13}$$

The function h_j is suitable because, in the form sketched in Fig. 1, it provides for a near-field region with uniform grid spacing, and a rapid expansion of the grid to infinity. Equation (13) is nearly identical to Eq. (11), differing only in small regions near the *corners and edges* of the three-dimensional box, where two or more of the individual functions h_j comprising H are simultaneously different from unity.[2]

The similarity of Eqs. (13) and (11) shows an important connection with the buffer region treatment in Ref. [4], where a region of exaggerated mesh stretching (though not infinite) was used in combination with filtering in a region near the artificial boundaries. That is, the effect of windowing should, at least to the crude level of approximation of Eq. (11), be similar to the effect of grid stretching.

3.3. Filtering

We return now to Eq. (11) and recall that we have filtered the equations in Fourier space (windowed in real space) but have yet to provide any model for the unresolvable (supergrid) disturbances. This model needs to eliminate poorly resolved disturbances near the boundary. To make further progress, we propose to filter Eq. (11) once again, but now in physical space. Consider a filter, $G(\mathbf{x})$, and its Fourier transform, $\hat{G}(\mathbf{k})$. Again we have in mind a filter that is the product of filter functions in each of the three coordinate directions,

$$G(\mathbf{x}) = g\left(\frac{x_1}{\sigma}\right) g\left(\frac{x_2}{\sigma}\right) g\left(\frac{x_3}{\sigma}\right), \tag{14}$$

[2] Another difference between Eqs. (13) and (11), irrelevant to the present discussion, is that for viscous flow the derivatives within $F_j(Q)$ in Eq. (13) would also need to be transformed according to Eq. (12).

where the σ controls the width of the filter. Filtering Eq. (11), we obtain

$$\frac{\partial \bar{\tilde{Q}}}{\partial t} + G * \left(H \frac{\partial F_j(\tilde{Q})}{\partial x_j} \right) = 0, \tag{15}$$

where

$$\bar{\tilde{Q}}(\mathbf{x}) = G * \tilde{Q}(\mathbf{x}). \tag{16}$$

Further development of Eq. (15) requires either regularizing the singular inversion of Eq. (16) or modeling its second term. This is identical to the problem of constructing a model for LES, except for the additional windowing factor, H. There are, of course, a large number of models and subsequent refinements that have been proposed for LES (see, for example, Meneveau and Katz [19]).

In the present work, we apply a line of reasoning that is motivated by the tensor diffusivity model for LES (e.g., [17, 18]). This model uses a Taylor series expansion of the filtered product of two arbitrary functions, f and h,

$$\overline{fh} = \bar{f}\,\bar{h} - c\frac{\sigma^2}{2}\frac{\partial \bar{f}}{\partial x_j}\frac{\partial \bar{h}}{\partial x_j} + \cdots, \tag{17}$$

where c is a numerical constant that depends on the particular choice for G ($c = 1$ for a Gaussian filter). Equation (17) holds under fairly general circumstances. Proceeding along these lines, we may expand (15), to $O(\sigma^4)$, as

$$\frac{\partial \bar{\tilde{Q}}}{\partial t} + \bar{H}\frac{\partial \overline{F_j(\tilde{Q})}}{\partial x_j} = c\frac{\sigma^2}{2}\frac{\partial \bar{H}}{\partial x_k}\frac{\partial^2 \overline{F_j(\tilde{Q})}}{\partial x_j \partial x_k}. \tag{18}$$

By similar arguments, it is obviously possible to expand any filtered quantity to $O(\sigma 4)$,

$$\bar{f} = f - c\frac{\sigma^2}{4}\frac{\partial^2 f}{\partial x_j \partial x_j} + \cdots = f - c\frac{\sigma^2}{4}\frac{\partial^2 \bar{f}}{\partial x_j \partial x_j} + \cdots, \tag{19}$$

and using this in Eq. (18) we obtain, again to $O(\sigma^4)$,

$$\frac{\partial \bar{\tilde{Q}}}{\partial t} + \left(H + c\frac{\sigma^2}{4}\frac{\partial^2 H}{\partial x_k \partial x_k} \right)\frac{\partial \overline{F_j(\tilde{Q})}}{\partial x_j} = c\frac{\sigma^2}{2}\frac{\partial H}{\partial x_k}\frac{\partial^2 \overline{F_j(\tilde{Q})}}{\partial x_j \partial x_k}. \tag{20}$$

Although Eq. (18) is equivalent to Eq. (20) to $O(\sigma^4)$, the solutions will differ when implemented numerically. We prefer Eq. (20) because the effect of the modeled terms are localized to the region near the boundary where H is less than unity and the gradient of H is nonzero. In the interior where H is constant Eq. (20) is equivalent to the filtered version of Eq. (1).

3.4. Stability

In applying the tensor diffusivity model to LES, it is not guaranteed that the model is locally dissipative [18, 20]. Nondissipative behavior can in turn lead to numerical instability, especially when finite-difference schemes are used for discretization.

It is easy to see that in the present context, the term on the right-hand side of Eq. (20) can represent diffusion with a viscosity of either sign and is therefore a concern. We show this by considering the special case of a one-dimensional flow that consists of small disturbances to an otherwise uniform flow state. In this case the fluxes can be linearized such that

$$\frac{\partial F_1(\tilde{Q})}{\partial x_1} = A_0 \frac{\partial \tilde{Q}}{\partial x_1}, \tag{21}$$

where A_0 is the linearized flux Jacobian tensor (with constant coefficients). Differentiating a second time to form the right-hand side of Eq. (20), we have

$$\frac{\partial \bar{\tilde{Q}}}{\partial t} + \left(H + c\frac{\sigma^2}{4}\frac{d^2 H}{dx_1^2} \right) A_0 \frac{\overline{\partial \tilde{Q}}}{\partial x_1} = c\frac{\sigma^2}{2}\frac{dH}{dx_1} A_0 \frac{\partial^2 \tilde{Q}}{\partial x_1^2}. \tag{22}$$

The sign of the dissipation therefore depends on the sign of $\frac{dH}{dx_1}$ and the eigenvalues of the flux Jacobian, A_0. For the type of window we have in mind, the derivative of H would be positive near $x_1 = -L_1$ and negative near $x_1 = L_1$. Outgoing waves correspond to negative eigenvalues near $x_1 = -L_1$ and to positive eigenvalues near $x_1 = L$ (with, of course, the opposite signs for incoming waves). Thus the tensor diffusivity term, in the linearized one-dimensional limit, corresponds to diffusion for incoming waves, but the outgoing waves would be amplified by a negative diffusivity, which would make the scheme unstable.

We remedy this situation by employing a flux-splitting technique on the right-hand side of Eq. (20), and subsequently reversing the sign of the fluxes that represent incoming waves (in the one-dimensional sense). To do this, we first neglect any viscous terms in $F_j(\tilde{Q})$ (only in the right-hand side of Eq. (18)), and next we split the flux term $(\partial F_j(\tilde{Q}))/\partial x_j$ into positive and negative eigenvalues,

$$\frac{\partial F_j(\tilde{Q})}{\partial x_j} = \tilde{A}_j \frac{\partial \tilde{Q}}{\partial x_j} = \left(\tilde{S}_j \tilde{\Lambda}_j^+ \tilde{S}_j^{-1} + \tilde{S}_j \tilde{\Lambda}_j^- \tilde{S}_j^{-1} \right) \frac{\partial \tilde{Q}}{\partial x_j}, \tag{23}$$

where the columns of S_j are the right eigenvectors of A_j and $\Lambda^{\pm}$ are diagonal matrices containing the positive and negative eigenvalues. In Eq. (20), this term is multiplied by the gradient of the window function which is negative near the right edge of the computational domain and positive near the left edge. We therefore wish to change the signs of the overall term (eignvalue times gradient of H) to positive on both sides. This can be accomplished, albeit crudely, by taking absolute values in both terms, to obtain

$$\frac{\partial \bar{\tilde{Q}}}{\partial t} + \left(H + c\frac{\sigma^2}{4}\frac{\partial^2 H}{\partial x_k \partial x_k} \right) \frac{\partial \overline{F_j(\tilde{Q})}}{\partial x_j} = c\frac{\sigma^2}{2} \left| \frac{\partial H}{\partial x_k} \right| \frac{\partial}{\partial x_k} \overline{\left(\left(\tilde{S}_j |\tilde{\Lambda}_j^+| \tilde{S}_j^{-1} + \tilde{S}_j |\tilde{\Lambda}_j^-| \tilde{S}_j^{-1} \right) \frac{\partial \tilde{Q}}{\partial x_j} \right)}. \tag{24}$$

It should be stressed that we have *not* modified the original fluxes, F_j, appearing on the left-hand side.

We found, through experimentation, that Eq. (24) is effective in actual computations, but that a simplified approach for the right-hand side performed just as well. This involves replacing the flux splitting on the right-hand side with the *maximum* eigenvalue of A_j over

the entire domain.[3] The eigenvalues of A_j are, of course, just u_j, $u_j + a$, $u_j - a$, where u_j is the velocity and a is the sound speed. These, in turn, can be closely estimated based on a typical Mach number of the flow of interest, to yield

$$\frac{\partial \bar{\tilde{Q}}}{\partial t} + \left(H + c\frac{\sigma^2}{4}\frac{\partial^2 H}{\partial x_k \partial x_k} \right) \frac{\partial \overline{F_j(\tilde{Q})}}{\partial x_j} = c\frac{\sigma^2}{2} a_{typ}(M_{typ} + 1) \left| \frac{\partial H}{\partial x_k} \right| \frac{\partial^2 \bar{\tilde{Q}}}{\partial x_j \partial x_k}, \tag{25}$$

where a_{typ} and M_{typ} are characteristic sound speed and Mach numbers for the particular flow of interest.

A further simplification can serve to significantly speed up the computation. We first replace H with $h_1 h_2 h_3$, and in each individual flux in Eq. (24) (i.e., the sum over j) we replace the window functions transverse to the particular direction with unity. This only modifies the model at the *corners* of a two-dimensional domain, and at the *corners and edges* of a three-dimensional domain, as was discussed in Section 3.2. We obtain

$$\frac{\partial \bar{\tilde{Q}}}{\partial t} + \sum_{j=1}^{3} \left(h_j + c\frac{\sigma^2}{4} h_j'' \right) \frac{\partial \overline{F_j(\tilde{Q})}}{\partial x_j} = c\frac{\sigma^2}{2} a_{typ}(M_{typ} + 1) \sum_{j=1}^{3} |h_j'| \frac{\partial^2 \bar{\tilde{Q}}}{\partial x_j \partial x_j}. \tag{26}$$

Finally, we note that for the filter expansions discussed in Section 3.3, we require that $\sigma^2 \sim \Delta x^2$, where Δx is the near-field grid spacing. For simplicity, we choose a Gaussian filter which gives $c = 1$ and we choose a grid cutoff of 2 points per wavelength which gives $\sigma = 2\Delta x$. Note that when the mesh spacing is unequal in different coordinate directions, Δx may be brought inside the summations in Eq. (29) and replaced with Δx_j. Thus, we finally obtain

$$\frac{\partial \bar{\tilde{Q}}}{\partial t} + \sum_{j=1}^{3} (h_j + (\Delta x_j)^2 h_j'') \frac{\partial \overline{F_j(\tilde{Q})}}{\partial x_j} = 2a_{typ}(M_{typ} + 1) \sum_{j=1}^{3} (\Delta x_j)^2 |h_j'| \frac{\partial^2 \bar{\tilde{Q}}}{\partial x_j \partial x_j}. \tag{27}$$

This is the central result of this paper. As written, it remains to be closed by relating the filtered and windowed variables, $\bar{Q}$, to the windowed variables, $\tilde{Q}$, through the choice of an appropriate filter, G, and an appropriate LES turbulence model (if any). These issues are discussed in the next section. It is shown in Section 4 that the model is effective and robust in practice.

While the modifications presented in the section are entirely *ad hoc*, it is shown in Section 4 that the overall model is surprisingly effective and robust in practice. It could certainly be argued that there is no reason, at this stage, to prefer the present model to buffer regions that have been based on other *ad hoc* arguments (which were discussed in Section 1). On the other hands, it seems that viewing the effect of the buffer region as a filtering of the equations in wave space, supplemented by a model for the unresolved scales, is a useful way to think about the problem, as it provides a framework into which other models (e.g., eddy viscosity, approximate deconvolution) or refinements to the present method could be inserted in future work.

[3] As pointed out by a referee, this is similar to using the Lax–Friedrich flux in standard flux splitting, which is known to be very dissipative.

3.5. Closure for LES and DNS

We are left in Eq. (27) with the filtered Navier–Stokes equations, including the supergrid model, but these equations are not yet closed because we have not written the filtered flux, $\overline{F_j(\tilde{Q})}$, in terms of the filtered dependent variables $\tilde{Q}$. This is the standard LES problem, for which there are many possible subgrid models to choose from. A discussion of the relative merits is beyond the scope of the present paper. A standard approach would be to rewrite Eq. (26) as

$$\frac{\partial \bar{\tilde{Q}}}{\partial t} + \sum_{j=1}^{3}(h_j + (\Delta x_j)^2 h_j'')\frac{\partial F_j(\bar{\tilde{Q}})}{\partial x_j}$$
$$= \sum_{j=1}^{3}(h_j + (\Delta x_j)^2 h_j'')\frac{\partial F_j(\bar{\tilde{Q}}) - \overline{F_j(\tilde{Q})}}{\partial x_j} + 2a_{typ}(M_{typ}+1)\sum_{j=1}^{3}(\Delta x_j)^2 |h_j'|\frac{\partial^2 \bar{\tilde{Q}}}{\partial x_j \partial x_j}, \quad (28)$$

where $F_j(\bar{\tilde{Q}}) - \overline{F_j(\tilde{Q})}$ is the subgrid stress tensor (SGS).

It is also desirable to have a DNS version of the supergrid model, whereby the equations reduce to the full Navier–Stokes equations everywhere except the buffer region. This can be accomplished by noting that in the DNS limit, we have $\bar{\tilde{Q}} = \tilde{Q}$, and we merely remove the overbars in Eq. (28) to obtain

$$\frac{\partial \tilde{Q}}{\partial t} + \sum_{j=1}^{3}(h_j + (\Delta x_j)^2 h_j'')\frac{\partial F_j(\tilde{Q})}{\partial x_j} = 2a_{typ}(M_{typ}+1)\sum_{j=1}^{3}(\Delta x_j)^2 |h_j'|\frac{\partial^2 \tilde{Q}}{\partial x_j \partial x_j}. \quad (29)$$

In the DNS limit, an examination of Eq. (29) reveals that in the interior where each $h_j = 1$, the equations reduce to the full Eq. (1).

3.6. The Window Function and Tunable Parameters

Before Eqs. (28) and (29) can be used for computation, the window functions, $h(x)$, must be specified in each coordinate direction. Note that for ease of notation we suppress the subscript j in this section. In what follows, we are referring to any (or each) of the three coordinate directions.

There will appear in the window function several parameters which fall into two categories. First are those parameters that are to be determined based on the physics of the flow to be computed, such as the near-field grid spacing, Δx, and on the extent in each coordinate direction of the physical portion of the domain (where the window function is unity), denoted here by L. Similarly, characteristic sound speed and Mach number (a_{typ}, M_{typ}) appearing in Eqs. (28) and (29) are also determined by the particular flow considered.

The second category includes "tunable" parameters that can be chosen to give the best results (i.e., nonreflectivity). It is shown here that these can be essentially reduced to two free parameters. To begin with, the window function $h(x)$ in Eqs. (28) and (29) can be concocted from different elementary functions that will give the desired behavior, shown in Fig. 1. We assume without loss of generality that the total length of the domain extends in each coordinate direction over $-1 < x < 1$ (which is then scaled appropriately to obtain the physical length L), and we define a buffer length, ϵ, such that the near field extends $-\epsilon < x < 1 - \epsilon$. We consider functions which are symmetric about $x = 0$ but note that this

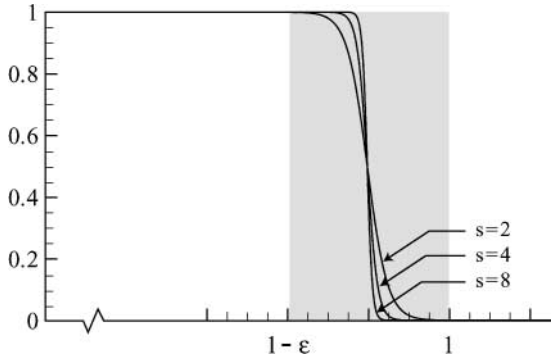

FIG. 2. Plot of window function in the buffer region, with different values of s.

can be readily generalized if it is desired to have buffers with different extents at either edge of each coordinate. A simple choice of function which gives the desired behavior is

$$h(x) = \frac{1}{2}\left(\tanh\left(2s\left[\frac{x+1-\epsilon/2}{\epsilon}\right]\right) - \tanh\left(2s\left[\frac{x-1+\epsilon/2}{\epsilon}\right]\right)\right), \tag{30}$$

where the dimensionless constant s controls the maximum slope of the window in the buffer as well as the curvature of the window near the edges of the buffer (see Fig. 2). Other choices, such as error functions or piecewise polynomials, are certainly possible, but they will give similar performance once they are scaled similarly. Moreover, functions that are continuously differentiable (such as tanh) are preferable when spectral discretization methods are to be used, as is done here. If spectral accuracy is desired, then the window function (30) should also be periodic to machine precision. Such a window can be constructed from the present one by adding a series of its periodic images. Because of the exponential nature of the window function, one image on either side of the domain is sufficient to give smooth behavior at the edge.

Since we have scaled the domain to vary over -1 to 1, we also have that $\epsilon = 2N_{bf}/N_{tot}$, where N_{bf} and N_{tot} are the number of grid points (in the given coordinate direction) in each buffer and over the total domain, respectively. Thus ϵ is also a measure of the computational expense of adding the buffer. As the buffer size increases indefinitely we obtain the exact infinite domain result (with perfect resolution and infinite expense). Thus the important question is how small can ϵ (or N_{bf}) be to obtain sufficiently accurate results? This is addressed through experimentation in the next section.

In the present choice of window function, the slope of the buffer function, s, must also be specified. In the context of the tanh functions used here, large values of s cause the variation of h from 0 to 1 to occur in a narrow bands centered at $\pm 1 \mp \epsilon/2$. It is undesirable to choose too large a value, as this simply adds additional points to the near field (where $h = 1$) and wastes points near the edge of the domain (where $h = 0$). On the other hand, too small a value will cause the buffer to spill over into the near field and not fall to zero at the edge. Based on Fig. 2 it can be seen that values between 2 and 4 fall into this intermediate range; through experimentation we have noted little difference in performance with s in this range. In computations presented in the next section we have used a conservative value of $s = 4$. It is also shown in the Appendix that larger values of s also have a deleterious impact on the maximum stable time step (when explicit time advancement is used).

4. EXAMPLES

In this section, we report the results of several tests of the numerical method on relatively simple flows. In all the examples we take a viscous compressible flow of a calorically perfect gas with $\gamma = 1.4$ and use zero bulk viscosity and other transport coefficients that vary according to Sutherland's law. In all the examples we use a Fourier spectral collocation method for evaluation of spatial derivatives together with explicit fourth-order Runge–Kutta time advancement in Eq. (29). Products involved in forming F_j were dealiased according to the 3/2 rule, as were the product of F_j with the window function and its gradients. Evaluation of F_j also requires division by the density, which is not dealiased. Specific numbers of grid points and domain sizes are given for each problem. We used a maximum CFL number of 0.5 in all computations. A discussion of time-step constraints associated with the buffer is given in the Appendix.

4.1. Propagation of a Nearly Linear Acoustic Pulse

We examine first the absorption of two-dimensional small-amplitude acoustic waves in an otherwise uniform flow, with a mean Mach number equal to 0.5. The configuration has become a standard figure of merit for ABCs: an initially Gaussian pressure pulse radiates out from the center of the computational domain. In what follows lengths are scaled to the width of the initial pulse. Hence the pressure is initialized as

$$p(\mathbf{x}, 0) = p_\infty + A \exp\left(-\left(\frac{x_1^2 + x_2^2}{L^2}\right)\right), \tag{31}$$

where A is set to 10^{-4} so that the acoustics are effectively linear. The velocities are initialized to zero. The Reynolds number, $a_\infty L/\nu_\infty = 10^5$, and viscous effects are unimportant over the small propagation distances considered.

The physical region of the domain extends to $\pm 8L$, with a uniform grid of 81×81 points (and thus a mesh spacing of $0.2L$ in each direction). We use three different buffers, with $N_{bf} = 10$, 20, and 30 points in each direction, with corresponding buffer lengths (on either side of the box) of 2, 3, and 4. For reference, we plot the solution at several early instants in time (for $N_{bf} = 20$) in Fig. 3. The 5% pulse width (i.e., the length scale at which the pulse is contained by contours of $\pm 5\%$ of its initial amplitude) can be seen to be on the order of $4L$ just prior to interaction with the buffer, and so the buffer is smaller than the pulse ($3L$ in the present case). Nevertheless, the pulse exits the physical domain with only small reflection.

The actual reflection error is now quantified in a series of comparisons with other ABCs. An exact solution, in the linear case, may be written down in terms of integrals, but it is more useful here to compute the error by simulating the solution on a domain that is sufficiently large that no reflections occur over the time interval of the test. This allows errors associated with the boundary condition to be isolated from the general discretization error. We compare the runs described above with a 400×400 node calculation with the same near-field grid spacing, and $N_{bf} = 10$ (the buffer in the large calculation being irrelevant). We have also computed, using a compact fourth-order finite-difference scheme (e.g., [16]) together with fourth-order Runge–Kutta time advancement, the solution obtained using several other boundary conditions. These include ABCs based on the Fourier/Laplace method [26], the one-dimensional characteristic boundary conditions (that are the linearization of those proposed by Thompson [31]), and, finally, the PML buffer layer approach.

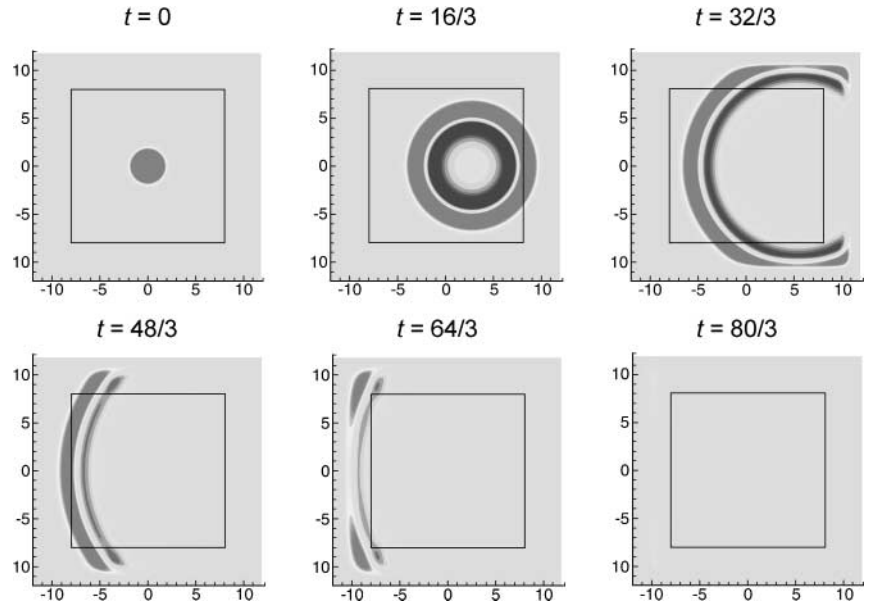

FIG. 3. Contours of the perturbation pressure at various instants in time (normalized by the ambient speed of sound and pulse width, L). Sixteen isocontours of pressure between -0.05 and 0.05, relative to the initial amplitude, are plotted. $N_{bf} = 20$. The square box to $\pm 8L$ indicates the boundary of the physical portion of the calculation.

For the Fourier/Laplace technique, we use a Padé rational function approximant to the square root operator that appears (in Fourier space) in the BC. Specifically we use the (0,0) and (2,2) Padé approximants to the square root [26]. The (0,0) approximation is equivalent to the earlier ABC of Giles [10]. In using the Fourier/Laplace ABC, problems can arise at the corner points, and in the present case we have simply written the ABC as a closure for the derivatives normal to the boundaries and applied *both* BCs at the corner points. This appears to be stable, for the present flow, but is unstable for higher Padé approximants.

The PML layer tests were conducted by Matt Barone (Stanford University) following the implementation by Abarbanel *et al.* [2]. All parameters chosen were identical to those that the authors used in the test cases they presented, except that the *layer strength* was changed to $C_x = C_y = 0.5$, which gave the best results with $N_{bf} = 20$. It was noted that the results were not very sensitive to this parameter provided that $0.25 \leq Cx \leq 1.0$. The PML layers were closed with the one-dimensional characteristic boundary conditions discussed above. One case was run with (0,0) Padé boundary conditions but the results were virtually identical.

The results of all of the comparisons are plotted in Fig. 4. We plot the sum of the rms error in the fluctuations in u, v, and p. The error is measured by comparison with simulation on a much larger domain, as discussed above. For the buffer conditions, the rms is taken over the *physical* portion of the domain only, $x, y \in \pm 8L$. For the nonbuffer ABC, we make the domain 101×101 nodes with $x, y \in \pm 10L$, but again compute the rms error only over the region $x, y \in \pm 8L$. This facilitates comparison since the run times will be similar for the nonbuffer ABCs and the buffer ABCs with $N_{bf} = 20$. It is clear that the present approach gives a smaller initial reflection (around $ta/L = 10$) than does PML with the same number of points in the buffer layer; with $N_{bf} = 30$ it outperforms even the most accurate local

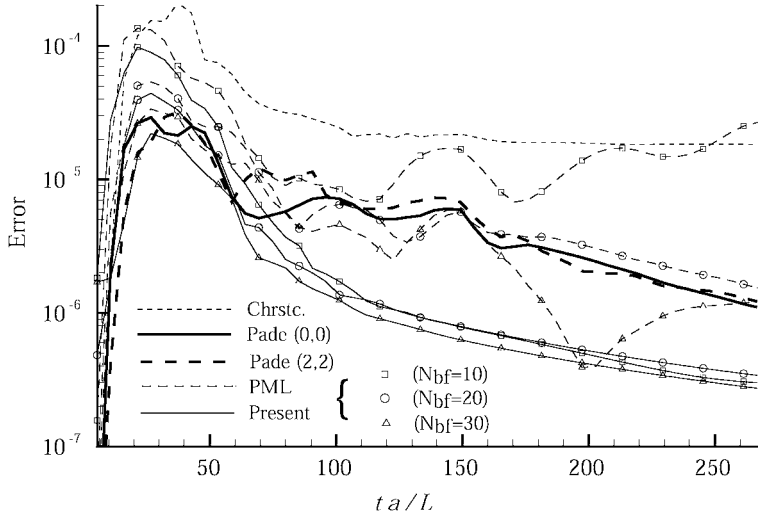

FIG. 4. The reflection error due to several different boundary conditions.

boundary conditions (the (2,2) Padé). This trend is also continued for long times, where the characteristic boundary conditions saturate at a high level of error while most of the other conditions tested have decaying error. Examining the solution at these times indicates that very little coherent signal is present in the domain; the solution is composed of short-wavelength modes that are continuously reflected by the boundaries and only damped slowly due to a slight dissipation associated with the time-marching scheme (see [26]) for a detailed discussion of these short-wavelength errors).

Finally, it may be noted that comparison of similar sets of boundary conditions (characteristic boundary condition, PML, and stretching/filtering) have been made in the context of electromagnetic wave scattering [13, 34]. There it was found that PML and stretching/filtering layers gave similar results, in both cases superior to characteristic-based conditions. It would appear that the present buffer layer formulation could also be readily applied to electromagnetic scattering problems, though we have not yet attempted to do so.

4.2. Advection of a Strong Vortex

We consider a vortical disturbance advecting in an otherwise uniform flow. We choose a zero-circulation vortex [5] since the velocity field associated with the vortex decays rapidly enough to localize the vorticity in a compact computational region. For inviscid flow, such a vortex is an exact solution of the governing equations for an arbitrary (axisymmetric) distribution of density in the vortex, provided that the pressure in the vortex is found by integrating:

$$\frac{\partial p(r)}{\partial r} = \frac{\rho(r) v_\theta(r)^2}{r}. \tag{32}$$

In what follows L now refers to the radius of the vortex (the location where the tangential velocity reaches a maximum), and we define the strength as the maximum tangential perturbation velocity, $V_{\max}$, divided by the free-stream velocity in which it advects, U. A similar test was described in [4], where Giles' [10] ABCs were applied to linearized (about

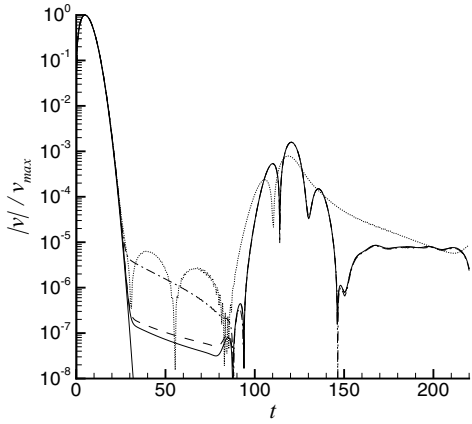

FIG. 5. Normal velocity at the center of the domain for vortex propagation with strength $V_{\max}/U$ equal to 0.005 (——), 0.05 (- - - -), 0.5, (– – –), 1.5, (· · · · · · · ·). The uniform background flow has $M_\infty = 0.2$.

the uniform base state) equations at the boundary. It was found that as the vortex advected out of the domain, the acoustic reflections, whose amplitude scaled with the *square* of the amplitude of the vortex, were produced. A similar test was also used to compare Thompson [31] BC and the buffer zone proposed by Freund [8].

In Fig. 5, the normal velocity at the center of the domain is plotted versus time for a variety of strengths $V_{\max}/U$ ranging from 0.005 to 1.5, all with $M_\infty = 0.2$, compared to the exact solution. For these cases, the physical portion domain extends to $\pm 20L$ of the initial position of the vortex and $\pm 10L$ in the spanwise direction, and $N_{bf} = 10$. The grid spacing is $0.1333L$ in both directions. Note that for the strongest case, there is a region of backflow at the downstream boundary as the vortex passes from the domain, since the maximum vortex velocity is greater than the advection velocity. Such a backflow represents a significant challenge for ABCs that depend on a labeling of incoming and outgoing disturbances (which would change as the velocity normal to the boundary changes sign at the boundary). Note that the normal velocity has been divided by $V_{\max}$ in the plots. The first bump in the curves occurs as the rear half of the vortex passes the center of the domain, shortly after the initialization. The vortex leaves the physical domain (entering the buffer region) around $tU/L \approx 20$ and produces a small acoustic reflection, which arrives back at the origin around $tU/L = 25$, i.e., the second bump in the curves. Thus the acoustic reflection is about two orders of magnitude smaller than the vortex velocity, which is good performance considering that the extent of the buffer region amounts to $10 \times 0.1333L = 1.333L$, just over half the total extent of the vortex.

In Fig. 6 the results are compared with the buffer treatment of Freund [8] and a simulation on a much larger domain that serves as an uncontaminated solution for comparison. As in [8], we take $M_\infty = 0.75$ and $V_{\max} = 0.6$, and here the grid spacing in the near field was $\Delta x = \Delta y = 0.0667L$. All parameters for Freund's buffer are taken from the example in [8]. In the figure, the integral of the kinetic energy (not including the base flow) over the physical portion of the computational domain (total domain less the buffer) is plotted relative to its initial value. For the inviscid vortex, this quantity would drop to zero exponentially fast as the vortex leaves the domain. Here we have taken $Re = UL/\nu = 7500$, and as the initial condition is not a solution of the viscous equations, there is a (small) acoustic transient

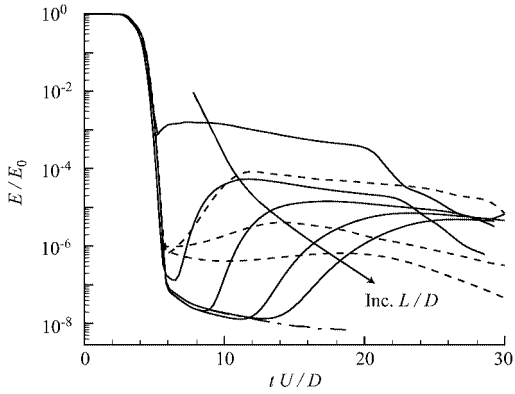

FIG. 6. Comparison of the buffer treatment of Freund [8] (- - - -, with $L/D = 2.0$, 4.2, and 7.2) with present results (——, with $L/D = 0.833$, 2.53, 4.17, 6.17, and 7.5) and the solution on a much larger domain (– - –). E is the integral of the kinetic energy (not including the base flow) over the physical portion of the computational domain.

wave that also propagates out of the domain. This gives rise to the slower decay of kinetic energy after the vortex leaves the domain around time $tU/L = 5$. With a buffer of size $L_{bf} = 0.833L$ ($N_{bf} = 12$), almost 99.9% of the kinetic energy is removed, and this is brought down by another order of magnitude for a buffer of extent $L_{bf} = 2.53L$, which is comparable to the roughly 99.99% energy removed for Freund's result with the same buffer size. We see some evidence of slow convergence, or saturation, in the accuracy of both buffer treatments as L_{bf} is increased indefinitely, but practically speaking, results with $L_{bf} = 2$ to $4L$ are adequate.

4.3. Two-Dimensional Starting Jet

As a final example, we consider the two-dimensional (planar), viscous, starting jet that is created by an applied body force in the x_1 direction that is impulsively switched on at $t = 0$. The body force, f_b, is nondimensionalized with L, the half width of the jet, and the ambient sound speed, a_∞, is a specified patch extending to $\pm L$ in the x_2 direction and smeared over a distance δ:

$$\frac{L\rho_\infty f_b}{a_\infty^2} = \frac{A}{2}\left(1 - \tanh\left(\frac{x_2/L - L/x_2}{2\delta/L}\right)\right)\exp -\frac{x_1^2}{\delta}. \tag{33}$$

We have run several cases with different parameters associated with the forcing and the computational domain (see Table I).

TABLE I
Physical and Computational Parameters for Starting Jets

Run	A	δ	N_1	N_2	N_{bf}	$x_{1\min}$	$x_{1\max}$	$x_{2\min}$	$x_{2\max}$	$Re = a_\infty L/\nu_\infty$
1	0.7	0.2	340	170	10	−6	15	−4	4	2,500
2	0.7	0.2	340	170	10	−1	9	−2.5	2.5	10,000
3	0.7	0.2	340	170	20	−1	9	−2.5	2.5	10,000
4	0.7	0.2	512	340	10	−3	13	−5	5	10,000

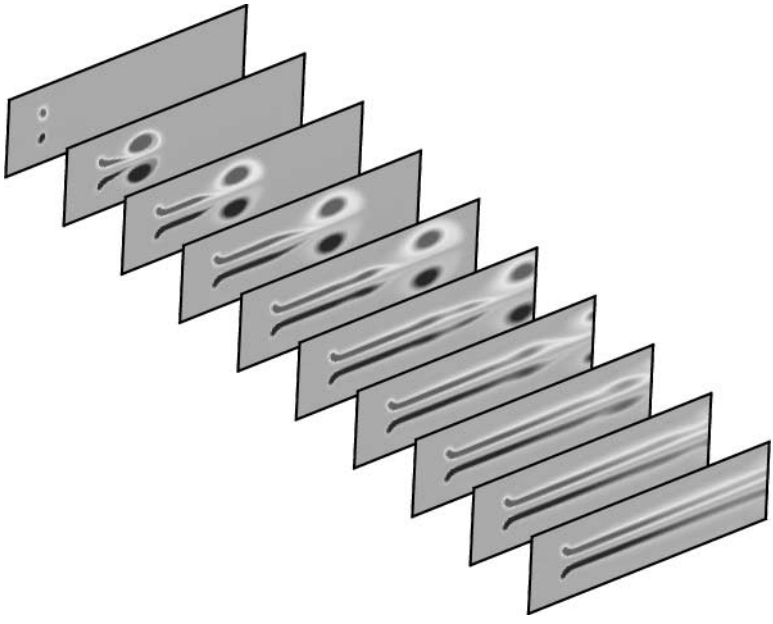

FIG. 7. A series of snapshots showing the vorticity during evolution of a two-dimensional impulsively started jet. Only the physical region of the computational domain is plotted.

An impulsive body force of this sort produces a leading vortex pair that pinches off from a trailing jet. The formation a the leading vortex ring (in the axisymmetric case) has been studied previously in experiments [25] and in computations that used a similar body force [21]. The toroidal radius of the ring is L and the leading ring is invariant to δ provided δ/L is small. Meanwhile δ directly controls the thickness of the mixing layers in the trailing jet. The development here, in the planar case, follows closely these previous studies. A series of snapshots for Run 1 in Fig. 7 show the development and pinch off of the leading vortex pair, and the subsequent development of a trailing jet. The flow is essentially steady at the last frame shown in the figure. The figure shows qualitatively the leading vortex pair and trailing jet being absorbed by the boundary at the right (an earlier acoustic transient at early

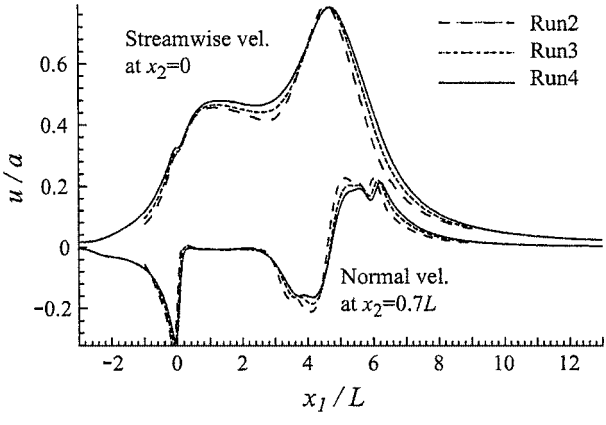

FIG. 8. Comparison of computations with different computational parameters.

times is also absorbed). It should be noted that the entire physical region is shown in the frames; the leading vortex pair is a significant fraction of the total width of the computational domain.

In Fig. 8, the streamwise velocity and normal velocity components are compared instantaneously for Runs 2, 3, and 4, at a time when the center of the vortex pair is situated near $x = 5L$. It can be seen that when N_{bf} is increased with the smaller domain, the results approach the result from the larger domain.

5. SUMMARY AND EXTENSIONS

A buffer region technique for computing compressible flows on unbounded domains was presented. The method was developed by exploiting an analogy with turbulence modeling: the artificial boundary condition is interpreted as a super-grid-scale model for eddies that propagate to distances larger than the mesh. Such a "near-eddy simulation" is analogous to a LES (where the model dissipates those eddies smaller than the mesh), provided that the roles of wave space and physical space are reversed.

The method is also closely connected with previous buffer techniques where mesh stretching and filtering and/or damping are employed. The chief differences are the modifications to the flux term, and the fact that the model yields a specific form for an added damping term in the buffer. An advantage of the present method is that it requires many fewer tunable parameters than other schemes. This is because once near-field extent and grid spacing are chosen (dictated by the relevant length scales of the problem under consideration), all that remains is to choose a suitable window function, $h(x)$, that falls from unity to zero over the buffer region. Any reasonable (i.e., monotonic) choice will require specification of the distance over which the window falls from unity to zero at the edge of the sponge, as well as a parameter (such as the maximum slope of the window) that controls the curvature of the window function near the edges. We suggest a particular window function, a combination of tanh functions, with two adjustable parameters: the number of grid points across the buffer, N_{bf}, to be user specified, and a parameter that controls the maximum slope of the window in the buffer, s, for which we suggest the range $2 < s < 4$. As was demonstrated in the model problems, N_{bf} is inversely proportional to the accuracy (i.e., magnitude of reflection) of the results, and directly proportional to the additional cost of the buffer. This presents the user with a simple trade-off of cost versus accuracy.

With $N_{bf} > 10$, the method was found to be quite accurate. For linear acoustic waves, the present treatment is more accurate than the PML technique for the same number of grid points in the buffer region. For passage of a strong (nonlinear) vortex propagating in an otherwise uniform flow, the method gave similar results to a previous buffer treatment [8]. It was shown that the present treatment leads to reflections that scaled linearly with the strength of the vortex, even when the maximum tangential velocity associated with the vortex was greater than the uniform background velocity (such that even an outflow boundary would feel a negative velocity during passage of the vortex).

For all of the model problems considered here, a dealiased Fourier pseudospectral scheme was used. But the equations for the buffer region were presented in real space and could be used with any discretization of the governing equations. Nevertheless, the use of the spectral method here shows that absorptivity of the buffer region was sufficient to prevent any periodic images from penetrating the domain. Spectral computations of compressible turbulent flows offer significant advantages over other schemes (e.g., high-order finite difference),

including the ability to dealias nonlinear products, which is advantageous in large-eddy simulations.

Finally, the method is suitable for any number of infinite coordinate directions and could in principle be applied in combination with other boundary conditions in other coordinate directions (e.g., infinite strips or slabs). In addition, we showed in an example how it could be used with a suitable body force to model a starting jet (without nozzle). Such body forces could allow further generalization of the method. For example, a general forcing term of the form

$$-\sigma(Q - Q_{target})$$

was used (in a finite-difference context) by Freund [9] to simulate the nozzle conditions for a turbulent jet. In this case the forcing is added to the right-hand side of the entire system of equations (Navier–Stokes, continuity, energy). We speculate that similar forcing could be used together with infinite coordinate directions and the present buffer region to simulate a variety of flows.

APPENDIX: TIME-STEP CONSTRAINTS DUE TO BUFFER

We present here a straightforward von Neumann stability analysis that may be applied to determine any impact of the buffer layers on the maximum permissible time step (when explicit time advancement is used). The notation follows closely that in Section 3.6. We consider Eq. (29) as a simple advection–diffusion equation in one-dimension by setting $F(Q) = a_{typ}(M_{typ} + 1)Q + \nu \frac{\partial Q}{\partial x}$ (with scalar Q) and replacing nonconstant coefficient terms (such as those involving $h(x)$) with worst-case values.

Consider first the time-step constraint associated with explicit time advancement of $F(Q)$. It is easy to see that the presence of the buffer will lower the maximum time permissible time step by an amount, p, where

$$p = \max(h(x) + (\Delta x)^2 h''(x)). \tag{34}$$

Using relations from the last section we find that the maximum is never greater than 1 provided that

$$\frac{\Delta x s}{\epsilon} < \sqrt{2}/2. \tag{35}$$

Since we have scaled the total domain to $-1 < x < 1$, this can be rearranged to give

$$N_{bf} > 2s/\sqrt{2}, \tag{36}$$

which is thus a requirement of N_{bf} so that there is no impact on the maximum permissible time step (for the flux term). This is not very restrictive, requiring only six points when $s = 4$ (its maximum recommended value). For fewer buffer points $p > 1$ is given by a complicated but irrelevant expression since the numerical experiments in Section 4 indicate that $N_{bf} \geq 10$ is required for reasonable accuracy.

Now consider the effect of adding the dissipative term on the right-hand side of Eqs. (29) and (28). For an explicit time advancement, the maximum time step for stability would be

given by

$$\Delta t < \frac{c_1(\Delta x)^2}{c_2 \nu}, \tag{37}$$

where c_1 and c_2 are $O(1)$ constants associated the choice of time advancement and spatial discretizations, respectively, and ν is the maximum value of the coefficient in front of $\frac{\partial^2 \bar{Q}}{\partial x^2}$. Inserting this coefficient from Eqs. (29) or (28), and noting that the maximum value of $|h'|$ is $\frac{s}{\epsilon}$, we obtain

$$\Delta t < \frac{c_1(\Delta x) N_{bf}}{2c_2 s\, a_{typ}(M_{typ}+1)}. \tag{38}$$

This term scales with Δx. If the actual time step used gives a CFL number, C, for the advective part of the flux term, then

$$(\Delta t)_{CFL} = \frac{C\Delta x}{a_{typ}(M_{typ}+1)}, \tag{39}$$

where C must be chosen low enough to ensure both stability and accuracy independent of any buffer consideration. In order that the buffer stability constraint not lower than the time step, we require

$$\Delta t < (\Delta t)_{CFL}. \tag{40}$$

Combining Eqs. (38) and (40) gives the constraint

$$N_{bf} > \frac{2Cc_2 s}{c_1}. \tag{41}$$

If we use spectral spatial discretization ($c_2 = \pi^2$) and fourth-order Runge–Kutta time advancement ($c_1 \approx 2.8$), then with $s = 4$ we obtain $N_{bf} > 28C$. For second-order finite-difference discretization, $c_2 = 4$ and $N_{bf} > 11C$. These numbers indicate that for relatively small buffers, C must be decreased in order to avoid instability in the buffer region or, conversely, that the buffer size must be increased if C is fixed. It is also interesting that the constraint depends on N_{bf} regardless of the total extent of the domain (or the grid spacing). We have found that in numerical experiments the scaling represented by Eq. (41) is obeyed but that the constants are somewhat conservative. For example with the spectral scheme used in the examples, $C_{\max} = 0.9$, and with $N_{bf} = 10$ and $s = 4$ we found instability when C was increased beyond 0.5 (whereas the formula would have predicted 0.35).

ACKNOWLEDGMENTS

The authors thank Mr. Matt Barone for generously sharing the data on PML presented in Section 4.1. TC wishes to acknowledge Prof. Jon Freund for many helpful discussions on boundary conditions.

REFERENCES

1. S. Abarbanel and D. D. Gottlieb, A mathematical analysis of the PML method, *J. Comput. Phys.* **134**, 357 (1997).

2. S. Abarbanel, D. Gottlieb, and J. S. Hesthaven, Well-posed perfectly matched layers for advective acoustics, *J. Comput. Phys.* **154**, 266 (1999).
3. S. C. Collis and S. K. Lele, *A Computational Investigation of Receptivity in High-Speed Flow Near a Swept Leading Edge*, Technical Report TF-71 (Department of Mechanical Engineering, Stanford University, 1997).
4. T. Colonius, S. K. Lele, and P. Moin, Boundary conditions for direct computation of aerodynamic sound generation, *AIAA J.* **31**(9), 1574 (1993).
5. T. Colonius, S. K. Lele, and P. Moin, The free compressible viscous vortex, *J. Fluid Mech.* **230**, 45 (1991).
6. J. A. Domaradzki and E. M. Saiki, A subgrid-scale model based on the estimation of unresolved scales of turbulence, *Phys. Fluids* **9**(7), 2148 (1997).
7. B. Engquist and A. Majda, Absorbing boundary conditions for the numerical simulation of waves, *Math. Comput.* **31**(139), 629 (1977).
8. J. B. Freund, Proposed inflow/outflow boundary condition for direct computation of aerodynamic sound, *AIAA J.* **35**(4), 740 (1997).
9. J. B. Freund, Noise sources in a low-reynolds-number turbulent jet at Mach 0.9, *J. Fluid Mech.* **438**, 277 (2001).
10. M. B. Giles, Nonreflecting boundary-conditions for Euler equation calculations, *AIAA J.* **28**(12), 2050 (1990).
11. J. W. Goodrich and T. Hagstrom, *A Comparison of Two Accurate Boundary Treatments for Computational Aeroacoustics*, AIAA Paper 97-1585 (1997).
12. Y. Guo, N. A. Adams, and L. Kleiser, Direct numerical simulation of transition in a spatially growing compressible boundary layer using a new fourier method, in *Direct and Large-Eddy Simulation*, edited by P. R. Voke, L. Kleiser, and J.-P. Chollet (Kluwer Academic, Dordrecht/Norwell, MA, 1994), Vol. 1, p. 249.
13. J. S. Hesthaven, P. G. Dinesen, and J. P. Lynov, Spectral collocation time-domain modeling of diffractive optical elements, *J. Comput. Phys.* **155**, 287 (1999).
14. M. Israeli and S. A. Orszag, Approximation of radiation boundary condition, *J. Compt. Phys.* **41**, 115 (1981).
15. M. Kloker, U. Konzelmann, and H. Fasel, Outflow boundary conditions for spatial Navier-Stokes simulations of transitional boundary layers, *AIAA J.* **31**, 620 (1993).
16. S. K. Lele, Compact finite-difference schemes with spectral-like resolution, *J. Comput. Phys.* **103**, 16 (1992).
17. A. Leonard, Energy cascade in large-eddy simulations of turbulent fluid flows, *Adv. Geophys.* **18**, 237 (1974).
18. A. Leonard and G. S. Winckelmans, A tensor-diffusivity subgrid model for large-eddy simulation, in *Direct and Large-Eddy Simulation III*, edited by P. R. Voke, N. D. Sandham, and L. Kleiser (Kluwer Academic, Dordrecht/Norwell, MA, 1999), p. 147.
19. C. Meneveau and J. Katz, Scale-invariance and turbulence models for large-eddy simulation, *Annu. Rev. Fluid Mech.* **32**, 1 (2000).
20. P. Moeleker and A. Leonard, Lagrangian methods for the tensor-diffusivity subgrid model, *J. Comput. Phys.* **167**, 1 (2001).
21. K. Mohseni, H. Ran, and T. Colonius, Numerical experiments on vortex formation, *J. Fluid Mech.* **430**, 267 (2001).
22. J. Nordström, N. Nordin, and D. Henningson, The fringe region technique and the Fourier method used in the direct numerical simulation of spatially evolving viscous flows, *SIAM J. Sci. Comput.* **20**(4), 1365 (1999).
23. T. J. Poinsot and S. K. Lele, Boundary conditions for direct simulations of compressible viscous flows, *J. Comput. Phys.* **101**, 104 (1992).
24. M. Rai and P. Moin, Direct simulations of turbulent flow using finite-difference schemes, *J. Comput. Phys.* **96**, 15 (1991).
25. M. Rosenfeld, E. Rambod, and M. Gharib, Circulation and formation number of laminar vortex rings, *J. Fluid Mech.* **376**, 297 (1998).
26. C. W. Rowley and T. Colonius, Discretely nonreflecting boundary conditions for linear hyperbolic systems, *J. Comput. Phys.* **157**, 500 (2000).
27. P. R. Spalart, Direct numerical study of leading edge contamination, in *Fluid Dynamics of Three-Dimensional Turbulent Shear Flows and Transition* (AGARD, Cesme, Turkey, 1988), Vol. AGARD-CP-438, p. 5.1.

28. S. Stolz and N. A. Adams, An approximate deconvolution procedure for large-eddy simulation, *Phys. Fluids* **11**(7), 1699 (1999).
29. S. Taasan and D. M. Nark, *An Absorbing Buffer Zone Technique of Acoustic Wave Propagation*, AIAA Paper 95-0164 (1995).
30. C. K. W. Tam, *Advances in Numerical Boundary Conditions for Computational Aeroacoustics*, AIAA Paper 97-1774 (1997).
31. K. W. Thompson, Time dependent boundary conditions for hyperbolic systems, *J. Comput. Phys.* **68**, 1 (1987).
32. K. W. Thompson, Time dependent boundary conditions for hyperbolic systems, II, *J. Comput. Phys.* **89**, 439 (1990).
33. L. N. Trefethen, Group velocity in finite difference schemes, *SIAM Rev.* **24**(2), 113 (1982).
34. B. Yang, D. Gottlieb, and J. S. Hesthaven, Spectral simulations of electromagnetic wave scattering, *J. Comput. Phys.* **134**, 216 (1997).