Journal of Computational Physics **167,** 439–474 (2001)
doi:10.1006/jcph.2000.6687, available online at http://www.idealibrary.com on IDEAL®

A Multilevel Algorithm for Large-Eddy Simulation of Turbulent Compressible Flows

Marc Terracol,* Pierre Sagaut,* and Claude Basdevant†

ONERA, 29 avenue de la Division Leclerc, 92322 Châtillon Cedex, France; and †Université Paris-Nord, LAGA UMR 7539, 99 avenue J.B. Clément, 93430 Villetaneuse, France
E-mail: sagaut@onera.fr

Received January 26, 2000; revised October 31, 2000

A multilevel method for large-eddy simulation of turbulent compressible flows is proposed. The method relies on the splitting of the turbulent flowfield into several frequency bands in space and time, each band being associated to a specific computational grid in physical space. This allows to take into account in a deterministic way the information contained on finer grids. A subgrid model adapted to such a decomposition—based on a generalization of the Germano's identity to multilevel decomposition—is also introduced. The approach is validated by several multilevel simulations in a subsonic plane channel flow configuration for a low and a high value of the Reynolds number, while reductions of the CPU times up to 80% are obtained.

Key Words: large-eddy simulation; multilevel algorithm; subgrid-scale modeling; turbulence.

---

1. INTRODUCTION

During the past few decades, the numerical simulation of turbulent unsteady flows has seen a considerable gain of interest. Many industrial configurations, for example in the aeronautical field, deal with such flows, and the numerical tool appears to be a good way to study and understand the involved physical phenomenons.

Nevertheless, unsteady numerical simulation of turbulent flows is still restricted to simple configurations, because turbulence is a complex three-dimensionnal phenomenon in which many different scales are present, ranging from the characteristic lengthscale of the problem itself, to the Kolmogorov dissipation scale. Thus, the complete resolution of a turbulent case involves very fine computation grids. That is why, even with the increasing capabilities of today's computers, direct numerical simulations (DNS) of turbulent flows are still restricted to low Reynolds numbers and simple geometries.

Large-eddy simulations (LES) allow the use of coarser meshes, by resolving directly only the largest scales of the flow. They are based on a turbulent flowfield's scale separation,

0021-9991/01 $35.00

obtained by a frequency filtering of the Navier–Stokes equations. Large scales of the flow are directly resolved, while small scales, referred to as subgrid scales (SGS), are represented through a statistical model. Such methods, because of the coarser meshes used, allow a significant decrease in computational costs. Nevertheless, the use of LES on computational grids that are too coarse generally provides poor results and depends heavily on the subgrid model used. This is particulary true in the case of flows whose dynamics are driven by coherent structures associated with a wide range of frequencies, because they cannot be described through a statistical model and need to be considered from a deterministic point of view. Thus, such flows still require the use of fine meshes, leading to high computational costs. That is why it appears necessary to develop new approaches that allow one to directly take into account small scales, while maintaining acceptable CPU times. Following this idea, we present here a multilevel strategy, which is based on a turbulent flowfield's decomposition into several frequency ranges that are treated separately.

Several authors have used such a strategy for incompressible flows simulations, and one can distinguish different approaches. Dubois *et al.* [9–11] and Debussche *et al.* [4] have used a spectral approach, in which a multiscale decomposition is obtained by the truncation of the spectral series expansion of the solution. They derived a dynamic algorithm, the Dynamic MultiLevel methodology (DML), in which the truncation level is adjusted dynamically in time through estimates of the small-scale time derivatives. This algorithm is based mainly on the quasi-static approximation, which stipulates that the time variation of the small scales can be neglected compared with the large scales one's during a short integration time. Simulations of incompressible isotropic three-dimensionnal homogeneous turbulence have been performed that confirmed this hypothesis. Voke [38] has used the same approximation, but his approach is different in the sense that he worked in physical space. The scale separation was then obtained by the use of several nested overlapping grids of different refinement degrees. To reduce the CPU times, a cycling strategy with V-cycles between the different grids was used, as in multigrid algorithms. In this approach, the integration times on each grid were taken of the order of the smallest eddy turnover time on each grid, thus providing quite long integration times, while the quasi-static approximation is only valuable for very short integration times. This is why this approach must be seen as a good way to accelerate transition to turbulence and convergence to a quasi-steady state, rather than a real unsteady method. Indeed, Voke defines the multimesh method as "*at least, a cheap way of turning pseudorandom initial velocities into something like real turbulence*," and he highlights the need of reducing the decorrelation between the large-scale and residual fields to reduce integration times on the fine level.

A different time integration strategy has been used by Sullivan *et al.* [37] and Boersma *et al.* [1]. In this case, the quasi-static approximation is not necessary, since no cycling strategy has been used. The reduction of the CPU times is obtained by the use of local grid refinements. The simulation is performed on a coarse grid, and on local finer grids which overlapp the coarse one, only in some critical regions of the flow. At each time step, the values of the coarse grid are updated in the overlapping regions by the use of fine-to-coarse interpolations. This strategy has been applied to the simulation of a three-dimensionnal incompressible planetary boundary-layer flow by Sullivan *et al.* [37] and to an incompressible two-dimensionnal mixing-layer flow by Boersma *et al.* [1] and in both cases provided a great improvement of the solution. Moreover, Sullivan *et al.* showed that the use of a two-way coupling between the grids, based on a fluxes correction on the coarse grid by using the fine grid information, improves the quality of the results. Another example of local grid refinement applied to the simulation of turbulent flows is the DNS of separated turbulent boundary layer performed by Manhart *et al.* [28].

A new trend in LES is the use of deconvolution methods, which appear in fact as a type of multilevel method. In that case, two filtering levels are considered, and the general idea is to perform the computation at a coarse level, while the fine one is used to generate some scales smaller than the resolved ones. These scales are then used to get an evaluation of the subgrid terms on the coarse level. One can cite the velocity estimation model of Domaradzki and Saiki [8] and its recent extension to physical space [7], in which the filtered velocity field is deconvolved on a two times finer computational grid to generate nonlinearily some smaller scales that are used to compute the subgrid stress tensor. Recent works, based on an approximate deconvolution procedure have also been carried out by Stolz and Adams [36].

A last multilevel strategy is based on the use of simplified evolution equations for the small (unresolved) scales of the flow, as in the multiscale variational method proposed by Hugues *et al.* [19], thus allowing the use of an exact equation for the coarse (resolved) scales.

Since spectral methods are still restricted to very simple geometrical cases, the method presented in this paper is based on a classical finite volume approach, combined with the use of different nested overlapping grids. Such a strategy allows us to take into account most of the scales in a deterministic way by the use of a fine grid, and thus to minimize parametrization errors, while a reduction of the CPU times is obtained through the use of coarser grids with less points and greater time steps.

We can distinguish two points of view to the application of such a method. From the physical point of view, the use of different refinement levels introduces different cutoff lengths and frequencies. This induces a continuous multilevel formalism generated by the different filtering levels. LES can be considered as a particular monolevel form of this formalism. From a numerical point of view, a discrete formalism is generated by the different discretization levels and implies the use of discrete operators to allow communication between the different grids.

In this paper, a parallel between these two points of view will be drawn. The two formalisms will be introduced and presented as a general multilevel context, which can be seen under the multiresolution approach introduced by Harten [15, 16].

An adapted multilevel subgrid closure is also proposed in Section 1.3, in which the interactions between the different frequency bands are taken into account in a deterministic way, while the statistical part of the subgrid model is modified dynamically, allowing interactions between distant frequency bands from the turbulent spectrum to be taken into account. All the developments are made in the case of compressible flows.

Deconvolution approaches as proposed by Stolz and Adams [36] or in the velocity estimation procedure of Domaradzki *et al.* [7, 8] use two different filtering levels and should be classified as a particular two-level case of the present method. However, in these methodologies, a fine level is only introduced to get an evaluation of the subgrid term at the computational level. The method proposed in this paper is a general multilevel approach, which can be classified in several ways. It can be seen as: (i) a deconvolution approach, if the computation is performed at the coarsest level, while finer ones are used to compute accurately the subgrid terms, and (ii) an acceleration technique to perform LES at lower cost, when the computation is performed at the finest level, while coarser levels are used to reduce the CPU times. This is the point of view that has been retained in the present study.

At last, it can be interpreted as a more general combined approach if the original LES grid corresponds to an intermediary level.

A multilevel algorithm is thus introduced in Section 2.2, in which a reduction of the CPU times in comparison with a monolevel LES is searched through a cycling strategy based on a quasi-static approximation of small-scale dynamics.

The approach is assessed in Section 3 by several multilevel simulations in a subsonic plane channel flow configuration, which appears as a good test case because of its near-wall behavior that presents small coherent structures in the viscous sublayer. Moreover, this configuration has been studied by many authors, so that a reference database is available.

2. CONTINUOUS FORMULATION

2.1. *Context and Governing Equations*

The proposed multilevel algorithm is based on N different filtering levels. Each of these levels $n \in [1, N]$ is characterized by a cutoff wavenumber k_n in spectral space, associated to a cutoff lengthscale $\overline{\Delta}^n$ in physical space, thus defining a turbulence spectrum partitionning (cf. Fig. 1). These parameters allow to define a low-pass frequency filter $\overline{(.)}^n$ for each level n, which, in classical LES, is formally defined as a convolution product of any function of space and time $\phi : (\Omega \times \mathbb{R}^+) \rightarrow \mathbb{R}$, with a filter function G_n:

$$\overline{\phi}^n(x,t) = (G_n \star \phi)(x,t) = \int_\Omega G_n(x-\xi)\phi(\xi,t)\,d\xi. \tag{1}$$

Hereafter, we will take $k_N < \cdots < k_1$, which implies $\overline{\Delta}^1 < \cdots < \overline{\Delta}^N$. That is to say, that level 1 is the finest one, and that level N is the coarsest one. The filtered variable $\overline{\phi}^n$ defined by (1) accounts only for the scales resolved by the filterwidth $\overline{\Delta}^n$, i.e., associated with wavenumbers $k < k_n$.

Considering the filtered variable $\overline{\phi}^1$ on the finest level, we obtain the filtered variables at the level n by the recursive applications of the filters $\overline{(.)}^1$ to $\overline{(.)}^n$ on the variable ϕ. The variables associated to the level n will so be noted $\overline{\overline{\phi}}^n$, where the notation $\overline{\overline{(.)}}^n$ stands for the recursive application of the filters $\overline{(.)}^1$ to $\overline{(.)}^n$, as mentionned above, and where $\overline{\overline{\phi}}^1 = \overline{\phi}^1$.

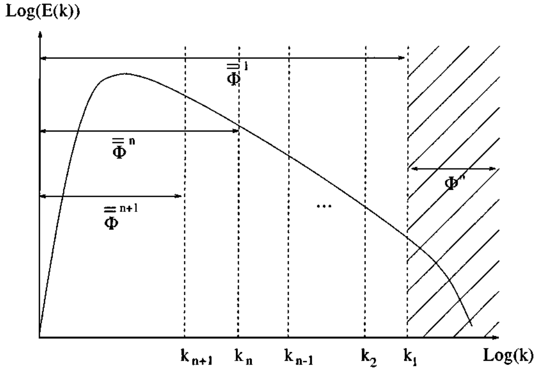


FIG. 1. Turbulence spectrum partitioning.

Thus, for any $n \in [1, N]$, $\overline{\overline{\phi}}^n$ is defined by

$$\begin{aligned}\overline{\overline{\phi}}^n &= G_n \star G_{n-1} \star \cdots \star G_1 \star \phi \\ &= G_1^n \star \phi \end{aligned} \tag{2}$$

with, for any $m \in [1, n] : G_m^n = G_n \star G_{n-1} \star \cdots \star G_m$.

In the following, the cutoff lengthscale associated to the filtering operator G_1^n will be referred to as $\overline{\overline{\Delta}}^n$. In the general case, $\overline{\overline{\Delta}}^n$ differs from $\overline{\Delta}^n$, since it is associated with a multiple filtering of the solution rather than a single filtering. In the particular case where each primary filter G_n is a Gaussian filter of width $\overline{\Delta}^n$, we have in spectral space

$$\widehat{G_n}(k) = \exp\left(-\frac{(k\overline{\Delta}^n)^2}{24}\right) \Rightarrow \widehat{G_1^n}(k) = \exp\left(-\frac{k^2 \sum_{l=1}^n (\overline{\Delta}^l)^2}{24}\right)$$

and the filter G_1^n defined by (2) is then a Gaussian filter of width $\overline{\overline{\Delta}}^n = (\sum_{l=1}^n (\overline{\Delta}^l)^2)^{\frac{1}{2}}$. As in classical LES, commutativity of the filters $\overline{(.)}^n$ with space and time derivatives will be supposed, i.e.,

$$\overline{\frac{\partial \phi}{\partial \xi}}^n = \frac{\partial \overline{\phi}^n}{\partial \xi}, \tag{3}$$

where $\xi = t$ or x_i, $i = 1, 2, 3$.

One shows easily that (3) implies that $\overline{\overline{(.)}}^n$ is also commuting with space and time derivatives. To take into account the compressible character of the flow, we introduce mass-weighted-filtered variables, as those introduced by Favre,

$$\tilde{\phi}^n = \frac{\overline{\rho\phi}^n}{\overline{\rho}^n} \quad \text{and} \quad \tilde{\tilde{\phi}}^n = \frac{\overline{\overline{\rho\phi}}^n}{\overline{\overline{\rho}}^n} = \widetilde{\tilde{\tilde{\phi}}^{(n-1)}}^n,$$

where ρ denotes density.

Remark. The use of spectral sharp cutoff filters for G_n, $n \in [1, N]$ is a very particular case. The filters G_n are then defined in spectral space by

$$\widehat{G_n}(k) = \begin{cases} 1 & \text{if } |k| < k_n \\ 0 & \text{else.} \end{cases}$$

Thus, we have for any $m \in [1, n] : \overline{\overline{\phi}^m}^n = \overline{\phi}^n$, which implies that $\overline{\overline{\phi}}^n = \overline{\phi}^n$, and also $\overline{\Delta}^n = \overline{\overline{\Delta}}^n$. In practice, sharp cutoff filters can only be used in spectral numerical algorithms. Here, we present a very general formalism in which no particular form of the filters is assumed. That is why $\overline{\phi}^n$ will be distinguished from $\overline{\overline{\phi}}^n$.

Applying successively the filters $\overline{(.)}^1$ to $\overline{(.)}^n$ on the dimensionless Navier–Stokes equations, we obtain the filtered equations for the level n,

$$\begin{cases} \frac{\partial}{\partial t}(\overline{\overline{\rho}}^n) + \frac{\partial}{\partial x_j}\left(\overline{\overline{\rho}}^n \tilde{\tilde{u}}^n_j\right) = 0 \\ \frac{\partial}{\partial t}\left(\overline{\overline{\rho}}^n \tilde{\tilde{u}}^n_i\right) + \frac{\partial}{\partial x_j}\left(\overline{\overline{\rho}}^n \tilde{\tilde{u}}^n_i \tilde{\tilde{u}}^n_j\right) = -\frac{\partial \overline{\overline{p}}^n}{\partial x_i} + \frac{\partial}{\partial x_j}\left(\overline{\overline{\sigma}}^n_{ij}\right) - \frac{\partial}{\partial x_j}\left(\tau^{(n)}_{ij}\right) - f_1\delta_{i1} \\ \frac{\partial}{\partial t}(\hat{\hat{E}}^n) + \frac{\partial}{\partial x_j}\left(\left(\hat{\hat{E}}^n + \overline{\overline{p}}^n\right)\tilde{\tilde{u}}^n_j\right) = \frac{\partial}{\partial x_j}\left(\overline{\overline{\sigma}}^n_{ij}\tilde{\tilde{u}}^n_j - \tau^{(n)}_{ij}\tilde{\tilde{u}}^n_j\right) - \frac{\partial}{\partial x_j}\left(\overline{\overline{Q}}^n_j + q^{(n)}_j\right) - f_1\tilde{\tilde{u}}^n_1, \end{cases} \tag{4}$$

where t is time, x_i $(i = 1, \ldots, 3)$ are the three spatial coordinates, δ_{ij} is Kronecker's delta, and where Einstein's summation convention is adopted. The three velocity components are noted u_i $(i = 1, \ldots, 3)$, p is the pressure, f_1 is a forcing term that will be described later in the paper, and $\hat{\hat{E}}^n$ is the total computable energy at the level n:

$$\hat{\hat{E}}^n = \frac{\overline{\overline{p}}^n}{\gamma - 1} + \frac{1}{2}\overline{\overline{\rho}}^n \tilde{\tilde{u}}^n_i \tilde{\tilde{u}}^n_i.$$

The filtered viscous stress tensor $\overline{\overline{\sigma}}^n$ is computed as

$$\overline{\overline{\sigma}}^n_{ij} = 2\frac{\mu(\tilde{\tilde{T}}^n)}{Re}\left(\tilde{\tilde{S}}^n_{ij} - \frac{1}{3}\tilde{\tilde{S}}^n_{kk}\delta_{ij}\right),$$

where

$$S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$$

and μ is the dynamic viscosity, given by Sutherland's law for air,

$$\mu(T) = T^{\frac{3}{2}}\frac{1 + C}{T + C},$$

where T is the temperature and $C = 110.4/T_0$, with T_0 being the reference temperature. At the level n, the filtered temperature $\tilde{\tilde{T}}^n$ is given by the ideal gas state law

$$\tilde{\tilde{T}}^n = \gamma M_0^2 \frac{\overline{\overline{p}}^n}{\overline{\overline{\rho}}^n},$$

where M_0 is the reference Mach number, and γ is the ratio of the specific heats (C_p/C_v). The filtered viscous heat flux vector $\overline{\overline{Q}}^n$ is given by

$$\overline{\overline{Q}}^n_j = -\frac{\mu(\tilde{\tilde{T}}^n)}{(\gamma - 1)Re\,Pr\,M_0^2}\widetilde{\widetilde{\frac{\partial T}{\partial x_j}}}^n,$$

where Re and Pr are, respectively, the reference Reynolds and Prandtl numbers.

Following Vreman *et al.* [39] and Domaradzki *et al.* [5], all subgrid quantities resulting from the nonlinearities of σ_{ij}, Q_j, and μ are neglected. The nonlinearities of the convective terms lead to two subgrid quantities: the subgrid stress tensor $\tau^{(n)}$ given by

$$\tau^{(n)}_{ij} = \overline{\overline{\rho}}^n\left(\widetilde{\widetilde{u_i u_j}}^n - \tilde{\tilde{u}}^n_i \tilde{\tilde{u}}^n_j\right) \tag{5}$$

and the subgrid heat flux vector $q^{(n)}$ given by

$$q^{(n)}_j = \overline{\overline{\rho}}^n C_p\left(\widetilde{\widetilde{u_j T}}^n - \tilde{\tilde{u}}^n_j \tilde{\tilde{T}}^n\right). \tag{6}$$

2.2. *Multilevel Representation of Variables*

With the notation introduced in Section 2.1, the components u_i of the velocity field can be decomposed at the level n as

$$u_i = \tilde{\tilde{u}}_i^n + \sum_{l=1}^{n-1} \delta u_i^l + u_i'', \tag{7}$$

where $\tilde{\tilde{u}}_i^n$ represents the resolved velocity field at the level n, i.e., associated with the wavenumbers $k < k_n$; $\delta u_i^l = \tilde{\tilde{u}}_i^l - \tilde{\tilde{u}}_i^{l+1}$ represents the velocity field frequency complement between the two levels $l + 1$ and l, i.e., corresponding to wavenumbers between k_{l+1} and k_l; and u_i'' are the subgrid scales unresolved even at the finest level of resolution. A similar decomposition is obtained for the other aerodynamic quantities.

We then get a multilevel representation of the variables at the level n for the velocity field components

$$ML(u_i) = \left(\tilde{\tilde{u}}_i^n, \delta u_i^{n-1}, \ldots, \delta u_i^1, u''\right). \tag{8}$$

In a numerical simulation, only the filtered variables are known. Thus, u'' remains unknown, and the matched quantities are those at the finest level of resolution. A particular form of the multilevel representation (8) for the filtered velocity field at the first level of resolution is written

$$ML\left(\tilde{u}_i^1\right) = ML\left(\tilde{\tilde{u}}_i^1\right) = \left(\tilde{\tilde{u}}_i^n, \delta u_i^{n-1}, \ldots, \delta u_i^1\right). \tag{9}$$

That means that the knowledge of $\tilde{\tilde{u}}_i^n$ and $\delta u_i^{n-1}, \ldots, \delta u_i^1$ for any $n \in [2, N]$ allows us to recover the value of the finest resolved velocity field $\tilde{u_i}^1$ by a simple summation of all the components of its multilevel representation.

2.3. *A Proposal for a Multilevel Subgrid Closure*

The two subgrid terms $\tau^{(n)}$ and $q^{(n)}$ resulting from the nonlinearity of the convective terms of the Navier–Stokes equations cannot be computed directly because the quantities $\widetilde{\widetilde{u_i\phi}}^n$, where ϕ is either u_j or T, remain unknown. Thus, these two terms need an appropriate modeling. Classical LES closures such as eddy-viscosity closures (Smagorinsky [35], dynamic Smagorinsky [29], mixed-scale model [31]) and hybrid scale-similarity closures [18, 33, 41], model the action of the small scales corresponding to wavenumbers greater than the level cutoff number; i.e., $k > k_n$ for level n (see Fig. 2a). In the multilevel case, the small scales associated with wavenumbers $k_n < k < k_1$ are resolved on the finer levels, and so, it is neither necessary nor suitable to model them. In contrast, it can be of great interest to take all these scales into account in a deterministic way rather than model them with a statistical closure (see Fig. 2b).

We propose here a two-part dynamic parametrization well suited for a multilevel algorithm that makes it possible to account for the information corresponding to wavenumbers $k_n < k < k_1$ in a deterministic way and to adapt dynamically the statistical part of the model. This closure is based on a generalization of Germano's identity [13] to the multilevel case, and on an extension of hybrid scale-similarity models [41].

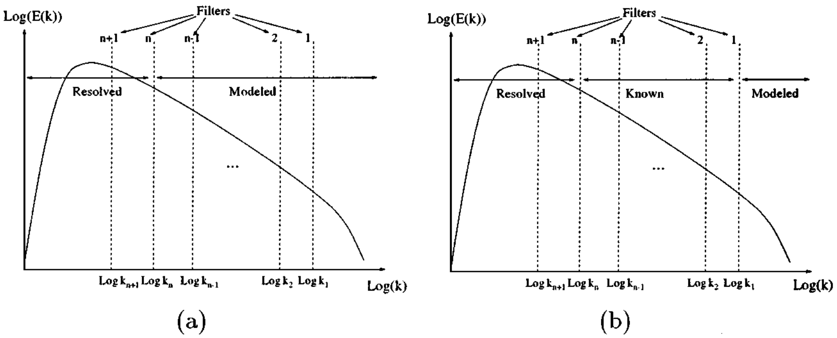


FIG. 2. “Classical” models (a) and proposed model (b).

2.3.1. Germano’s identity generalization. We introduce here a tensor $\mathcal{L}^{(n+1)}$, coming from a generalization of Germano’s procedure to the multilevel case:

$$\mathcal{L}_{ij}^{(n+1)} = \overline{\overline{\rho}}^{(n+1)} \left(\widetilde{\tilde{\tilde{u}}_i^n \tilde{\tilde{u}}_j^n}^{n+1} - \tilde{\tilde{u}}_i^{(n+1)} \tilde{\tilde{u}}_j^{(n+1)} \right). \tag{10}$$

This term has the advantage of being directly computable. Moreover, recalling the expression of the subgrid tensor $\tau^{(n)}$,

$$\tau_{ij}^{(n)} = \overline{\overline{\rho}}^n \left(\widetilde{\widetilde{u_i u_j}}^n - \tilde{\tilde{u}}_i^n \tilde{\tilde{u}}_j^n \right),$$

and using level $(n+1)$ as a test level, we get the exact relation, equivalent to Germano’s identity in the monolevel case,

$$\mathcal{L}_{ij}^{(n+1)} = \tau_{ij}^{(n+1)} - \overline{\tau_{ij}^{(n)}}^{(n+1)}. \tag{11}$$

By recurrence, we also get

$$\tau_{ij}^{(n)} = G_2^n \star \tau_{ij}^{(1)} + \mathcal{L}_{ij}^{(n)} + \sum_{k=2}^{n-1} G_{k+1}^n \star \mathcal{L}_{ij}^{(k)}. \tag{12}$$

Relation (11) will be used hereafter in the parametrization of $\tau^{(n)}$.

In a same manner, we introduce a vector $\mathcal{Q}^{(n+1)}$ for the subgrid heat flux:

$$\mathcal{Q}_i^{(n+1)} = \overline{\overline{\rho}}^{(n+1)} C_p \left(\widetilde{\tilde{\tilde{u}}_i^n \tilde{\tilde{T}}^n}^{n+1} - \tilde{\tilde{u}}_i^{(n+1)} \tilde{\tilde{T}}^{(n+1)} \right). \tag{13}$$

Recalling the expression of the subgrid heat flux vector $q_i^{(n)} = \overline{\overline{\rho}}^n C_p (\widetilde{\widetilde{u_i T}}^n - \tilde{\tilde{u}}_i^n \tilde{\tilde{T}}^n)$, we get the exact relation

$$\mathcal{Q}_i^{(n+1)} = q_i^{(n+1)} - \overline{q_i^{(n)}}^{(n+1)} \tag{14}$$

that will be used in the parametrization of $q^{(n)}$. By recurrence, we also get

$$q_i^{(n)} = G_2^n \star q_i^{(1)} + \mathcal{Q}_i^{(n)} + \sum_{k=2}^{n-1} G_{k+1}^n \star \mathcal{Q}_i^{(k)}. \tag{15}$$

2.3.2. Subgrid stress-tensor modeling. For simplicity, the following developments use the generalized central moments notation introduced by Germano [14]:

$$\mathcal{T}_n[a,b] = \widetilde{\widetilde{ab}}^n - \tilde{\tilde{a}}^n \tilde{\tilde{b}}^n.$$

Introducing the decomposition (7) of the velocity field in the expression of the subgrid stress tensor $\tau_{ij}^{(n)}$, we get an extension of Germano's consistent decomposition [12] to the multilevel case,

$$\tau_{ij}^{(n)} = L_{ij}^{(n)} + C_{ij}^{(n)} + R_{ij}^{(n)}, \tag{16}$$

where

$$\begin{aligned} L_{ij}^{(n)} &= \overline{\overline{\rho}}^n \mathcal{T}_n\left[\left(\tilde{\tilde{u}}_i^n + \sum_{l=1}^{n-1} \delta u_i^l\right), \left(\tilde{\tilde{u}}_j^n + \sum_{l=1}^{n-1} \delta u_j^l\right)\right] \\ C_{ij}^{(n)} &= \overline{\overline{\rho}}^n \left(\mathcal{T}_n\left[\left(\tilde{\tilde{u}}_i^n + \sum_{l=1}^{n-1} \delta u_i^l\right), u_j''\right] + \mathcal{T}_n\left[u_i'', \left(\tilde{\tilde{u}}_j^n + \sum_{l=1}^{n-1} \delta u_j^l\right)\right]\right) \\ R_{ij}^{(n)} &= \overline{\overline{\rho}}^n \mathcal{T}_n[u_i'', u_j'']. \end{aligned} \tag{17}$$

- $L_{ij}^{(n)}$ is the resolvable part of $\tau_{ij}^{(n)}$ and appears as an extension of Bardina's scale-similarity model to the multilevel case. It represents the large-scales interaction and, in addition, the interactions between large scales and the small scales resolved on the finer levels and interactions between these "resolved subgrid scales" themselves. This part of the subgrid stress tensor contains all the deterministic information that can be directly computed.
- $C_{ij}^{(n)}$ is the cross-term tensor, which represents interactions between the large scales and the unresolved subgrid scales and those between resolved and unresolved subgrid scales.
- $R_{ij}^{(n)}$ is the classical Reynolds stress tensor, which represents the interactions between the unresolved subgrid scales.

One can easily verify that each term of the decomposition (16) preserves the Galilean invariance property of the subgrid stress tensor.

We propose here to compute directly the $L_{ij}^{(n)}$ term and thus to model only the cross- and Reynolds terms of $\tau_{ij}^{(n)}$. Following Zang *et al.* [41], we propose to use a two-part dynamic parametrization for the anisotropic part of $\tau_{ij}^{(n)}$,

$$\left(\tau_{ij}^{(n)}\right)^* = \left(L_{ij}^{(n)}\right)^* - 2C_d^{(n)} \overline{\overline{\rho}}^n \left(\overline{\overline{\Delta}}^n\right)^2 \left|\tilde{\tilde{S}}^n\right| \left(\tilde{\tilde{S}}_{ij}^n\right)^*, \tag{18}$$

where $(\phi_{ij})^* = \phi_{ij} - \frac{1}{3}\phi_{kk}\delta_{ij}$. The deterministic information of the subgrid stress tensor contained in ${L_{ij}}^{(n)}$ will be computed directly, while the statistical Smagorinsky part will be adapted through a dynamic evaluation of the coefficient $C_d^{(n)}$. It will be evaluated using an extension of Zang *et al.*'s procedure in which our generalization of Germano's identity will be used instead of the classical one.

Introducing the decomposition (7) in the expression of the subgrid stress tensor at the level $(n+1)$ that will be used as a test level, we obtain a decomposition of $\tau_{ij}^{(n+1)}$, similar to the one obtained for $\tau_{ij}^{(n)}$:

$$\tau_{ij}^{(n+1)} = L_{ij}'^{(n)} + C_{ij}'^{(n)} + R_{ij}'^{(n)}. \tag{19}$$

Note that $L_{ij}^{\prime(n)}$ (resp. $C_{ij}^{\prime(n)}$, $R_{ij}^{\prime(n)}$) differs from $L_{ij}^{(n+1)}$ (resp. $C_{ij}^{(n+1)}$ $R_{ij}^{(n+1)}$) since it is obtained by using the decomposition of the field at level n (instead of $n+1$) into the expression of the subgrid stress tensor at level $n+1$. In particular, the expression for $L_{ij}^{\prime(n)}$ is

$$L_{ij}^{\prime(n)} = \overline{\overline{\rho}}^{(n+1)} \mathcal{T}_{n+1}\left[\left(\tilde{\tilde{u}}_i^n + \sum_{l=1}^{n-1} \delta u_i^l\right), \left(\tilde{\tilde{u}}_j^n + \sum_{l=1}^{n-1} \delta u_j^l\right)\right]. \tag{20}$$

From their definition, the two filters G_1^n and G_1^{n+1} are self-similar [2], i.e., they have the same shape, and only differ by their characteristic lengthscale. This allows us to consistently use the same model for the two levels n and $n+1$, with the same coefficient $C_d^{(n)}$.

We thus model $(\tau_{ij}^{(n+1)})^*$ in the same way as $(\tau_{ij}^{(n)})^*$ by

$$\left(\tau_{ij}^{(n+1)}\right)^* = \left(L_{ij}^{\prime(n)}\right)^* - 2C_d^{(n)} \overline{\overline{\rho}}^{(n+1)} \left(\overline{\overline{\Delta}}^{(n+1)}\right)^2 \left|\tilde{\tilde{S}}^{(n+1)}\right| \left(\tilde{\tilde{S}}_{ij}^{(n+1)}\right)^*. \tag{21}$$

From the two expressions (18) and (21), and assuming that $C_d^{(n)}$ is the same for all the band $[k_{n+1}, k_n]$, the generalized Germano's identity (11) gives

$$\left(\mathcal{L}_{ij}^{(n+1)}\right)^* = \left(L_{ij}^{\prime(n)} - \overline{L_{ij}^{(n)}}^{(n+1)}\right)^* - 2C_d^{(n)} M_{ij}^{(n)}, \tag{22}$$

where

$$M_{ij}^{(n)} = \overline{\overline{\rho}}^{(n+1)} \left[\left(\overline{\overline{\Delta}}^{(n+1)}\right)^2 \left|\tilde{\tilde{S}}^{(n+1)}\right| \left(\tilde{\tilde{S}}_{ij}^{(n+1)}\right)^* - \left(\overline{\overline{\Delta}}^{n}\right)^2 \widetilde{\left(|\tilde{\tilde{S}}^n|\left(\tilde{\tilde{S}}_{ij}^n\right)^*\right)}^{n+1}\right].$$

The optimized value for the coefficient $C_d^{(n)}$ is obtained through a least-squares minimization of the residual of relation (22), as proposed by Lilly [26]. We thus obtain

$$C_d^{(n)} = -\frac{\left(\mathcal{L}_{ij}^{(n+1)} - \left(L_{ij}^{\prime(n)} - \overline{L_{ij}^{(n)}}^{(n+1)}\right)\right)^* M_{ij}^{(n)}}{2M_{ij}^{(n)} M_{ij}^{(n)}}. \tag{23}$$

Remark. Considering a classical eddy-viscosity parametrization of $\tau^{(n)}$; i.e., imposing $L^{(n)} = L^{\prime(n)} = 0$, we get the classical value of $C_d^{(n)}$ of the dynamic version of Smagorinsky's model. Moreover, if we consider only the classical Germano's decomposition of the velocity field, i.e., $u_i = \tilde{\tilde{u}}_i^n + u_i'$, without taking into account the multilevel aspect, we get the classical dynamic mixed Bardina–Smagorinsky model as in Zang *et al.* [41].

2.3.3. Subgrid heat flux vector modeling. The same dynamic mixed methodology, as the one for $\tau^{(n)}$, is used to parametrize the subgrid heat flux vector $q^{(n)}$. First, we have to introduce the equivalent of the decomposition (7) for the temperature field. At the level n, it will be written as

$$T = \tilde{\tilde{T}}^n + \sum_{l=1}^{n-1} \delta T^l + T'', \tag{24}$$

where $\delta T^l = \tilde{\tilde{T}}^l - \tilde{\tilde{T}}^{l+1}$. Introducing the decomposition (7) of the velocity field and the decomposition (24) of the temperature field in the expression of the subgrid heat flux vector $q_i^{(n)}$, we get the decomposition

$$q_i^{(n)} = q_{L_i}^{(n)} + q_{C_i}^{(n)} + q_{R_i}^{(n)}, \tag{25}$$

where

$$
\begin{aligned}
q_{L_i}^{(n)} &= \overline{\overline{\rho}}^n C_p \mathcal{T}_n\left[\left(\tilde{\tilde{u}}_i^n + \sum_{l=1}^{n-1} \delta u_i^l\right), \left(\tilde{\tilde{T}}^n + \sum_{l=1}^{n-1} \delta T^l\right)\right] \\
q_{C_i}^{(n)} &= \overline{\overline{\rho}}^n C_p \left(\mathcal{T}_n\left[\left(\tilde{\tilde{u}}_i^n + \sum_{l=1}^{n-1} \delta u_i^l\right), T''\right] + \mathcal{T}_n\left[u_i'', \left(\tilde{\tilde{T}}^n + \sum_{l=1}^{n-1} \delta T^l\right)\right]\right) \\
q_{R_i}^{(n)} &= \overline{\overline{\rho}}^n C_p \mathcal{T}_n[u_i'', T''].
\end{aligned}
\tag{26}
$$

These three terms are, respectively, equivalent to $L_{ij}^{(n)}$, $C_{ij}^{(n)}$, and $R_{ij}^{(n)}$ obtained in the decomposition of $\tau_{ij}^{(n)}$. The term $q_{L_i}^{(n)}$, like $L_{ij}^{(n)}$, is directly computable and does not need any parametrization.

We thus propose using a two-part dynamic parametrization of $q_i^{(n)}$ under the form

$$
q_i^{(n)} = q_{L_i}^{(n)} - \overline{\overline{\rho}}^n C_p K_d^{(n)} (\overline{\overline{\Delta}}^n)^2 |\tilde{\tilde{S}}^n| \widetilde{\widetilde{\frac{\partial T}{\partial x_i}}}^n . \tag{27}
$$

As in the parametrization of $\tau_{ij}^{(n)}$, the level $(n+1)$ is used as a test level. The subgrid heat flux vector $q_i^{(n+1)}$ at this level is decomposed using the two decompositions (7) of u_i and (24) of T as

$$
q_i^{(n+1)} = q_{L_i}^{\prime(n)} + q_{C_i}^{\prime(n)} + q_{R_i}^{\prime(n)}. \tag{28}
$$

We then choose to model $q_i^{(n+1)}$ as

$$
q_i^{(n+1)} = q_{L_i}^{\prime(n)} - \overline{\overline{\rho}}^{(n+1)} C_p K_d^{(n)} \left(\overline{\overline{\Delta}}^{(n+1)}\right)^2 \left|\tilde{\tilde{S}}^{(n+1)}\right| \widetilde{\widetilde{\frac{\partial T}{\partial x_i}}}^{(n+1)} . \tag{29}
$$

From the two expressions (27) and (29), and assuming that $K_d^{(n)}$ is the same for the entire band $[k_{n+1}; k_n]$, the identity (14) gives

$$
\mathcal{Q}_i^{(n+1)} = \left(q_{L_i}^{\prime(n)} - \overline{q_{L_i}^{(n)}}^{(n+1)}\right) - K_d^{(n)} m_i^{(n)}, \tag{30}
$$

where

$$
m_i^{(n)} = \overline{\overline{\rho}}^{(n+1)} \left[\left(\overline{\overline{\Delta}}^{(n+1)}\right)^2 \left|\tilde{\tilde{S}}^{(n+1)}\right| \widetilde{\widetilde{\frac{\partial T}{\partial x_i}}}^{(n+1)} - (\overline{\overline{\Delta}}^n)^2 \widetilde{\left(|\tilde{\tilde{S}}^n| \widetilde{\widetilde{\frac{\partial T}{\partial x_i}}}^n \right)}^{n+1} \right].
$$

The optimized value for the coefficient $K_d^{(n)}$ is obtained through a least-squares minimization of the residual of relation (30). We thus obtain

$$
K_d^{(n)} = -\frac{\left(\mathcal{Q}_i^{(n+1)} - \left(q_{L_i}^{\prime(n)}\right) - \overline{q_{L_i}^{(n)}}^{(n+1)}\right) m_i^{(n)}}{m_i^{(n)} m_i^{(n)}}. \tag{31}
$$

3. DISCRETE FORMULATION

All the formalism introduced in Section 1 can only find its place in a numerical simulation if an equivalent discrete formalism is introduced. This formalism, which can be seen as a particular case of the multiresolution representation of data introduced by Harten [15, 16], will be presented first. Then, it will be shown how such a formalism can be used in a multilevel algorithm.

3.1. *Discrete Formalism*

From a discrete point of view, the different filtering levels are obtained by the use of N nested overlapping discretization grids of the domain Ω, referred to as $\Omega_1, \ldots, \Omega_N$ hereafter, where Ω_1 corresponds to the finest grid, and Ω_N to the coarsest one.

For any continuous function ϕ of space and time, the filtered variable ϕ^1 on the finest computational grid Ω_1 is defined by the application of a linear discretization operator $\mathcal{D}_1$ on the variable ϕ. Here, Ω_1 is the fine discretization of the domain Ω; i.e., $\mathcal{D}_1 : \mathcal{F}(\Omega, \mathbb{R}) \rightarrow \mathcal{F}(\Omega_1, \mathbb{R})$. The fine variable ϕ^1 is thus defined as

$$\phi^1 = \mathcal{D}_1 \phi. \tag{32}$$

The variable ϕ^1 appears as the discrete equivalent of the continuous quantity $\overline{\overline{\phi}}^{1}$. In other words, as in classical LES, the continuous filter G_1 remains unknown and the only effective filtering is done by the space discretization.

Any variable on the coarser grids is then defined from ϕ^1, by successive applications of fine-to-coarse interpolation operators referred to as *restriction* operators as in standard multigrid terminology. The variable ϕ^n on the nth level, which is the discrete equivalent of $\overline{\overline{\phi}}^{n}$, is thus defined as

$$\begin{aligned} \phi^n &= R^n_{n-1} \phi^{n-1} \\ &= \left(R^n_{n-1} \circ R^{n-1}_{n-2} \circ \cdots \circ R^2_1\right) \phi^1 \\ &= R^n_1 \phi^1, \end{aligned} \tag{33}$$

with $R^n_1 = R^n_{n-1} \circ R^{n-1}_{n-2} \circ \cdots \circ R^2_1 : \mathcal{F}(\Omega_1, \mathbb{R}) \rightarrow \mathcal{F}(\Omega_n, \mathbb{R})$.

Similarly to R^{n+1}_n, we define coarse-to-fine prediction operators P^n_{n+1} from $\mathcal{F}(\Omega_{n+1}, \mathbb{R})$ onto $\mathcal{F}(\Omega_n, \mathbb{R})$, which allow us to recover an approximation of ϕ^n on the grid Ω_n from the knowledge of ϕ^{n+1} on the coarser grid Ω_{n+1}. These operators will be referred to as *prolongation* operators. Applying successively the operators R^{n+1}_n and P^n_{n+1} on ϕ^n, we can define the interpolation error on the grid Ω_n by

$$\delta\phi^n = \phi^n - P^n_{n+1} \circ R^{n+1}_n \phi^n = \phi^n - P^n_{n+1} \phi^{n+1}. \tag{34}$$

In the LES context, we will consider that the application of the discrete restriction operator R^n_{n-1} on any variable is equivalent to the application of the continuous low-pass filter $\overline{(.)}^n$ on this variable. This is equivalent to saying that the only filtering on each grid Ω_n is done by the space discretization. The interpolation error $\delta\phi^n$ can be then considered as the frequency complement between the two levels n and $n + 1$.

For any $n \in [2, N]$, the sequence of restriction and prolongation operators R^{n+1}_n and P^n_{n+1} allows us to get a multilevel representation of ϕ^1:

$$ML(\phi^1) = (\phi^n, \delta\phi^{n-1}, \ldots, \delta\phi^1). \tag{35}$$

TABLE I
Equivalence Table between Continuous and Discrete Formalisms

Continuous formalism	Discrete formalism
$\overline{(.)}^1 \equiv \overline{\overline{(.)}}^1$	$\mathcal{D}_1 : \Omega \to \Omega_1$
$\overline{(.)}^n, n \in [2, N]$	$R^n_{n-1} : \Omega_{n-1} \to \Omega_n, n \in [2, N]$
$\overline{\overline{(.)}}^n, n \in [2, N]$	$R^n_1 = R^n_{n-1} \circ R^{n-1}_{n-2} \circ \dots \circ R^2_1 : \Omega_1 \to \Omega_n, n \in [2, N]$

More generally, we have for any $m \in [1, n-1]$

$$ML(\phi^m) = (\phi^n, \delta\phi^{n-1}, \dots, \delta\phi^m). \tag{36}$$

Thus, for any $n \in [2, N]$ and $m \in [1, n-1]$, we can recover the value of ϕ^m from the knowledge of its multilevel representation.

An equivalence between the continuous formalism and its discrete counterpart is summerized in Table I and Fig. 3.

Remark. For the velocity field components and for the temperature field, Favre-filtering is prescribed. Thus, we will use

$$\phi^n = \frac{R^n_1((\rho\phi)^1)}{R^n_1(\rho^1)}. \tag{37}$$

This definition is strictly equivalent to the continuous one: $\tilde{\tilde{\phi}}^n = \overline{\overline{\rho\phi}}^n / \overline{\overline{\rho}}^n$.

3.2. *Multilevel Algorithm*

We propose in this section a numerical algorithm, which is based on the multilevel representation of the flow variables ρ, u_i $(i = 1, \dots, 3)$, and T, and on the quasi-static

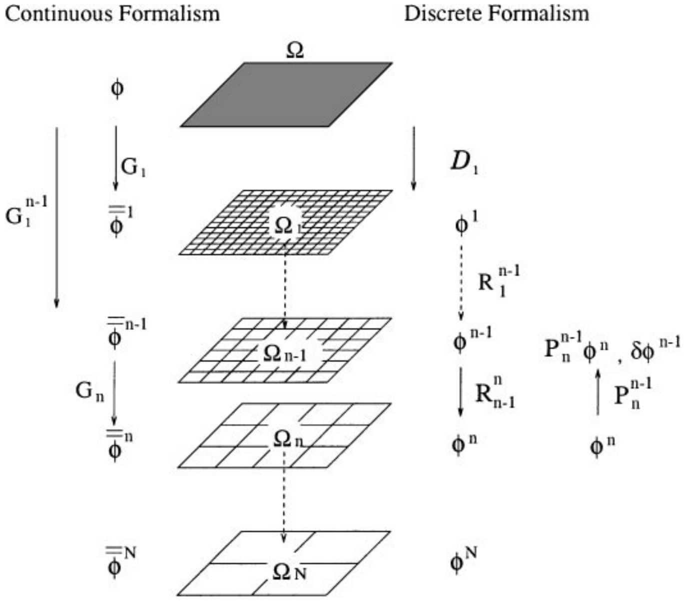


FIG. 3. Correspondance between continuous and discrete formalisms.

approximation [11]. The algorithm uses N nested overlapping grids, as mentionned previously. Starting from an initial flowfield on the finest grid Ω_1, we generate fields on the coarser grids by successive applications of fine-to-coarse grid restriction operators R_n^{n+1} $(n = 1, \ldots, N-1)$. This is equivalent, as previously said, to successive applications of low-pass frequency filters on the initial fine grid field.

On each grid Ω^n, $n \in [2, N]$, the computational variables are then

$$\rho^n = R_1^n(\rho^1)$$

$$u_i^n = \frac{R_1^n\left(\rho^1 u_i^1\right)}{R_1^n(\rho^1)}$$

$$T^n = \frac{R_1^n(\rho^1 T^1)}{R_1^n(\rho^1)}.$$

At each fine-to-coarse restriction step $(\Omega_n \xrightarrow[R_n^{n+1}]{} \Omega_{n+1})$, and for each flow variable ϕ, the frequency complement $\delta\phi^n$ is stored on the fine grid Ω_n to keep the small scales of the flow in memory.

The computation takes place on the coarse grids, and the resulting flowfield is then interpolated to the fine grids with the coarse-to-fine grid prolongation operators P_n^{n-1} $(n \in [2, N])$. At each prolongation step, the resulting fine flowfield is enriched with the frequency complement stored during the restriction step, kept frozen during the entire time of integration on the coarse grids.

The computation takes place as a succession of V-cycles, as in standard multigrid algorithms. Figure 4 presents schematically the multilevel algorithm in frequential and physical spaces, and one temporal integration cycle, in the case of a bidimensionnal three-level simulation. The integration times on each grid, referred to as Δt_n on Fig. 4, will be described later in the paper. The freezing of the frequency complements is equivalent, from a physical point of view, to the assumption that the small-scale variation on the grid Ω_n, $n \in [1, N-1]$, can be neglected during time integration on the coarser grids Ω_m, $m \in [n+1, N]$. This

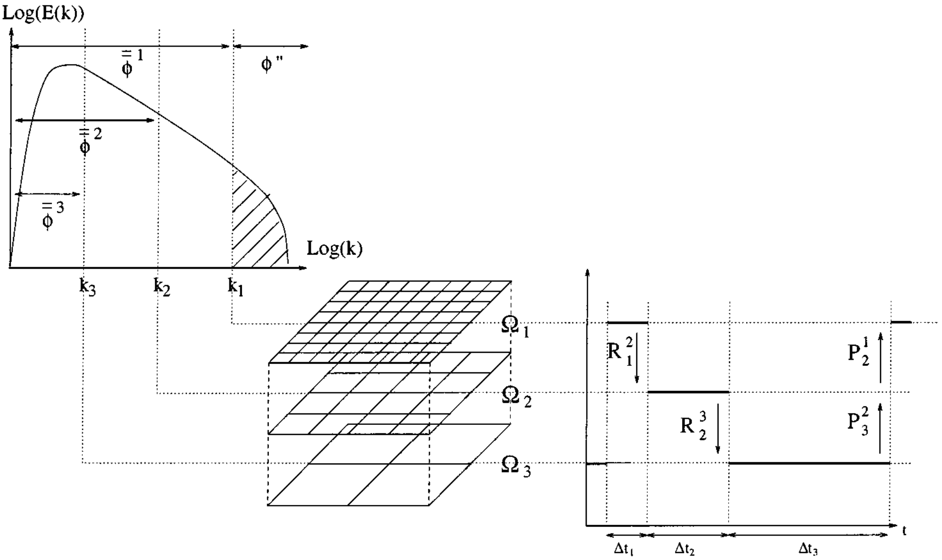


FIG. 4. Schematic representation of the multilevel algorithm.

is the quasi-static approximation. In [9–11], the authors have used the fact that since the small scales of the flow have a very different time behavior than the large ones, they can be computed with less accuracy. In particular, estimates of $\frac{\partial \delta\phi^n}{\partial t}$ have been obtained, with the result that

$$\left|\frac{\partial \delta\phi^n}{\partial t}\right|_0 \ll \left|\frac{\partial \phi^n}{\partial t}\right|_0,$$

where $|\_|_0$ is the norm associated with kinetic energy. Hence, it appears reasonable to expect that the small-scale variation during a short integration time can be neglected compared with the large-scale ones.

On each prolongation step, the value of ϕ^n at the time $t + \Delta t$ is then evaluated as

$$\phi^n(t + \Delta t) = P^n_{n+1}\phi^{n+1}(t + \Delta t) + \delta\phi^n(t). \tag{38}$$

One V-cycle is then defined by the following sequence.

1. $n = 1$
2. Apply the numerical scheme to ϕ^n during an integration time Δt_n.
3. Restriction step: If $n < N$ compute the field at level $n + 1$ by restriction: $\phi^{n+1} = R^{n+1}_n(\phi^n)$ and compute the frequency complement between levels n and $n + 1$ at level n: $\delta\phi^n = \phi^n - P^n_{n+1} \circ R^{n+1}_n(\phi^n)$.
4. If $n < N$, $n \leftarrow n + 1$ and go to 2.
5. Prolongation step: $\phi^{n-1} = P^{n-1}_n(\phi^n) + \delta\phi^{n-1}$.
6. If $n > 2$, $n \leftarrow n - 1$ and go to 5.
7. Go to 1.

4. APPLICATION: THE SUBSONIC CHANNEL FLOW

4.1. *Problem Configuration*

The test case that has been retained is the now well-known subsonic channel flow configuration, which consists in a flow between two infinite isothermal walls. It has been studied by many authors, both in the incompressible and compressible contexts. One can cite the incompressible DNS from Kim *et al.* [21] and Moser *et al.* [30] and the compressible LES by Lenormand *et al.* [24, 25].

This case is a typical example of a multiscale problem: Large structures dominate in the center of the channel, while the near-wall dynamics is driven by small streaky structures in the viscous sublayer. These streaks can only be captured in a deterministic way, and thus very fine grids are required in the near-wall region.

The nominal Mach number value is $M_0 = 0.5$. Two Reynolds number values, based on the mean bulk velocity u_b, the mean bulk density ρ_b, the wall viscosity $\mu(T_w)$, and the channel half-width $L_z/2$, have been considered: $Re = 2800$ and $Re = 11{,}000$, corresponding to two values of the skin-friction Reynolds number Re_τ, based on the wall shear velocity u_τ, of 180 and 590 respectively. The computations associated with these two cases will be referred to as LoNG-xxx and HiNG-xxx respectively, where N is the number of level considered and xxx is the subgrid model used.

The computational domain used for the low-Reynolds simulation is a box of dimensions $2\pi \times \frac{4\pi}{3} \times 2$ in the respective x, y, and z directions, where x is the streamwise direction, y is the spanwise one, and z is the wall-normal one. Uniform grid spacings are used in the streamwise and spanwise directions, while a hyperbolic tangent law is used in the wall-normal one. For the high-Reynolds case, the domain dimensions are taken equal to $2\pi \times \pi \times 2$ in the respective x, y, and z directions.

The computation is performed on N nested overlapping grids $\Omega_1, \ldots, \Omega_N$, where for any n, each coarse grid Ω_{n+1} is defined from the finer one Ω_n by keeping half the number of points in each direction. The number of points used for the finest mesh ($n = 1$) in the respective x, y, and z directions are $32 \times 64 \times 128$ for the low-Reynolds simulation and $52 \times 120 \times 128$ for the high-Reynolds one. A finer mesh of dimensions $40 \times 94 \times 128$ has also been used for the low-Reynolds case and is associated with the Lo3G-ML2 and Lo4G-ML2 cases studied in Section 4.5.3, and one of dimension $96 \times 184 \times 128$ has been used for the Hi4G-ML2 case.

Periodic boundary conditions are used in the streamwise and spanwise directions, and a classical isothermal no-slip condition is used for wall boundaries.

4.2. *Numerical Method*

4.2.1. Numerical scheme. A classical cell-centered finite-volume solver is used to solve the filtered Navier–Stokes system (4) on each grid. This system is rewritten under the compact form

$$\frac{\partial V}{\partial t} + \nabla \cdot (F_c(V)) + \nabla \cdot (F_v(V, \nabla V)) = 0, \tag{39}$$

where $V = (\overline{\overline{\rho}}^n, \overline{\overline{\rho u_1}}^n, \overline{\overline{\rho u_2}}^n, \overline{\overline{\rho u_3}}^n, \hat{\hat{E}}^n)^T$, and where F_c and F_v are respectively the convective and viscous fluxes. To ensure that the numerical scheme's dissipation doesn't cancel the effects of the subgrid model, no artificial dissipation has been used.

The divergence of the fluxes and the gradients of the numerical quantities are simply evaluated using Green's formulas on each cell. Nevertheless, to ensure the stability of the scheme, and in particular to reduce aliasing errors, the evaluation of the divergence of the numerical fluxes is done in a special way.

- A skew-symmetric formulation [23, 40] is used for the convective fluxes. This is equivalent, with the one-dimensionnal notation introduced in Fig. 5, to computing the convective flux $F_{c_{i+\frac{1}{2}}}$ at the cell interface $i + \frac{1}{2}$ as $F_c(V_{i+\frac{1}{2}})$, instead of using the classical formulation $\frac{1}{2}(F_c(V_i) + F_c(V_{i+1}))$.

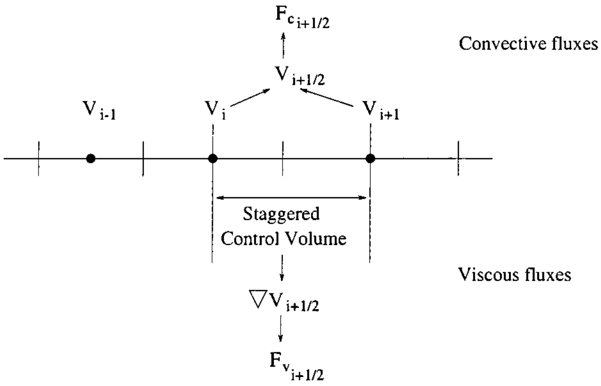


FIG. 5. Numerical fluxes evaluation.

- The viscous fluxes are evaluated by the use of staggered cells. Gradients of V are evaluated directly at cell interfaces by the application of Green's formulas on staggered control volumes. Then, the viscous fluxes $F_{v_{i+\frac{1}{2}}}$ at the cell interface are computed directly as $F_v(V_{i+\frac{1}{2}}, \nabla V_{i+\frac{1}{2}})$ instead of via the classical formulation $\frac{1}{2}(F_v(V_i, \nabla V_i) + F_v(V_{i+1}, \nabla V_{i+1}))$.

Finally, a classical third-order compact Runge–Kutta scheme is used for time integration [40].

Globally, the numerical scheme used here is second-order accurate in space and third-order accurate in time.

3.2.2. Grid transfer operators. Several classical grid transfer operators have been tested, such as, for example, the bilinear operators used in standard AMR and multigrid algorithms. Such operators, because of their quite high dissipation, lead to poor results. The operators used in our simulations are simple differential operators, the coefficients of which are evaluated to get

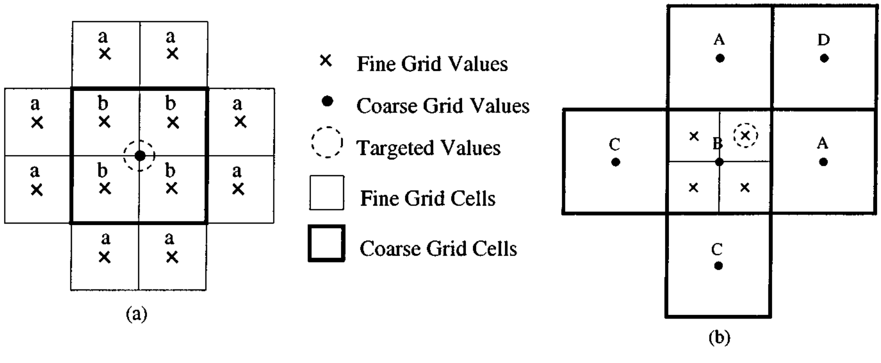
$$\begin{cases} R_n^{n+1} = \mathcal{I}d^n + \varepsilon_1\left(\Delta x_n^2 \dfrac{\partial^2}{\partial x^2} + \Delta y_n^2 \dfrac{\partial^2}{\partial y^2} + \Delta z_n^2 \dfrac{\partial^2}{\partial z^2}\right) + \mathcal{O}\left(\Delta x_n^3, \Delta y_n^3, \Delta z_n^3\right) \\ P_{n+1}^n = \mathcal{I}d^n + \varepsilon_2\left(\Delta x_n^2 \dfrac{\partial^2}{\partial x^2} + \Delta y_n^2 \dfrac{\partial^2}{\partial y^2} + \Delta z_n^2 \dfrac{\partial^2}{\partial z^2}\right) + \mathcal{O}\left(\Delta x_n^3, \Delta y_n^3, \Delta z_n^3\right), \end{cases}$$

where $\mathcal{I}d^n$ is the identity operator at the level n, and Δx_n, Δy_n, and Δz_n are the space steps at this level in the respective x, y, and z directions.

These operators are obtained by a three-dimensionnal extension of the stencils presented on Fig. 6, which shows the coefficients associated with each point from the fine grid field in the restriction step (a) and from the coarse grid field in the prolongation step (b). A classical volume ponderation is also used for nonuniform meshes.

The coefficients obtained in the three-dimensionnal case are

$$\begin{cases} a = -\frac{1}{64} + \frac{1}{8}\varepsilon_1 \\ b = \frac{11}{64} - \frac{3}{8}\varepsilon_1 \end{cases} \quad \text{and} \quad \begin{cases} A = \frac{1}{32} + \frac{1}{4}\varepsilon_2 \\ B = 1 - \frac{3}{2}\varepsilon_2 \\ C = -\frac{3}{32} + \frac{1}{4}\varepsilon_2 \\ D = \frac{1}{16}. \end{cases}$$



(a) (b)

FIG. 6. Restriction (a) and prolongation (b) stencils.

To get the maximal accuracy on the global operator $P^n_{n+1\,o}R^{n+1}_n$, we chose to take $\varepsilon_2 = -\varepsilon_1$. Our best results were obtained with $\varepsilon_2 = -\varepsilon_1 = 0$. In this case, both R^{n+1}_n, P^n_{n+1} and $P^n_{n+1\,o}R^{n+1}_n$ are third-order accurate. Simulations with other values of ε_1 and ε_2 have been performed, such as, for example, with $\varepsilon_1 = -\varepsilon_2 = \frac{1}{8}$, which leads to a well-known simple operator R^{n+1}_n (with $b = \frac{1}{8}$ and $a = 0$) but provides quite poor results because of the global scheme's behavior is too dissipative and thus cancels the small-scale contribution. In that case, R^{n+1}_n is equivalent to

$$\mathcal{I}d^n + \frac{1}{8}\left(\Delta x_n^2 \frac{\partial^2}{\partial x^2} + \Delta y_n^2 \frac{\partial^2}{\partial y^2} + \Delta z_n^2 \frac{\partial^2}{\partial z^2}\right) + \mathcal{O}\left(\Delta x_n^3, \Delta y_n^3, \Delta z_n^3\right),$$

which acts as a second-order dissipative operator. However, the choice $\varepsilon_1 = 0$ leads to a nondissipative operator in which the truncation error is reduced to a third-order dispersive term.

A consistent approach for relating fine and coarse variables in large-eddy simulations with AMR has been proposed recently by Cook [3], but this has not been tested here.

4.2.3. Subgrid terms computation. Because of the complexity of the expression of the scale-similarity terms in the subgrid quantities parametrization, some approximations are needed to compute them. We find in their continuous expression some terms involving operators like $G^n_1 \star G^n_1$. These terms will be evaluated in the discrete case by using the operator $F_o R^n_1$ instead, where F is a classical discrete filter with coefficients $\frac{1}{24}[1; 22; 1]$, resulting from the tensorial product in the homogeneous directions of the flow of the 1-D filters F_x and F_y:

$$\begin{aligned}
F_x(\phi) &= \frac{1}{24}(\phi_{i-1,j,k} + 22\phi_{i,j,k} + \phi_{i+1,j,k})\\
F_y(\phi) &= \frac{1}{24}(\phi_{i,j-1,k} + 22\phi_{i,j,k} + \phi_{i,j+1,k})\\
\Rightarrow \quad F(\phi) &= F_y(F_x(\phi)).
\end{aligned}$$

The discrete filter F thus defined is the discrete equivalent of a Gaussian or box filter of length $\overline{\Delta}^n$ [32]. For ϕ and ψ equal to $\tilde{\tilde{u}}^n$, $\sum_{l=1}^{n-1} \delta u^l$, $\tilde{\tilde{T}}^n$, or $\sum_{l=1}^{n-1} \delta T^l$, we will then use the approximations

$$\begin{cases}
\widetilde{\widetilde{\phi\psi}}^n \approx \frac{F(\overline{\overline{\rho}}^n \phi\psi)}{F(\overline{\rho}^n)}\\
\tilde{\tilde{\phi}}^n \approx \frac{F(\overline{\overline{\rho}}^n \phi)}{F(\overline{\rho}^n)}.
\end{cases}$$

4.3. *Forcing Term*

Because of the periodic boundary condition used in the streamwise direction, the computation needs the introduction of a forcing term in the filtered Navier–Stokes equations, to ensure mass flow rate conservation and convergence to a quasi-steady state. This is the term noted f_1 in Section 2.1. This term is computed following the numerical procedure proposed by Lenormand *et al.* [24, 25].

The driving term is then updated at each time step p by

$$f_1^{p+1} = f_1^p + \frac{\Delta t}{L_y L_z}[\alpha(Q^{p+1} - Q_0) + \beta(Q^p - Q_0)], \tag{40}$$

where Q_0 is the targeted mass flow rate, Q^p is the mass flow rate at time step p, and Q^{p+1} is its first-order prediction at time step $p+1$, given by

$$Q^{p+1} = Q^p - \Delta t \left[L_y L_z f_1^p + \frac{2L_y}{Re} \mu \frac{\partial \left\langle u_1^p \right\rangle_{xy}}{\partial z} \bigg|_{z=0} \right], \tag{41}$$

where $\langle . \rangle_{xy}$ denotes averaging in the xy plane. The two parameters α and β are taken equal to be $2/\Delta t$ and $-0.2/\Delta t$ respectively to ensure stability [24, 25].

4.4. *Initial Conditions*

The simulation is initiated from laminar parabolic profiles perturbed with random noise. The initial dimensionless profiles are given by

$$\begin{aligned} \rho(t=0) &= 1 + s\varepsilon \\ u(t=0,z) &= U_c[1-(z-1)^2](1+s\varepsilon) \\ v(t=0,z) &= s\varepsilon \\ w(t=0,z) &= s\varepsilon \\ T(t=0,z) &= 1 + \frac{\gamma-1}{3} Pr\, M_0^2 U_c^2 [1-(z-1)^4], \end{aligned} \tag{42}$$

where ε is a random number between -1 and 1, $U_c = 1.5$ is the dimensionless centerline streamwise velocity, and s is taken equal to 0.1.

4.5. *Numerical Tests*

For both the two Reynolds cases considered here, several computations have been performed. First, classical monolevel LES using the dynamic Smagorinsky closure have been carried out on the finest grid (runs Lo1G-dyn and Hi1G-dyn). The results of these simulations are used as reference results in the following to evaluate the multilevel procedure. The importance of an adapted subgrid closure on the coarse levels is studied by two-level simulations using different types of closures in Section 4.5.1. The influence of the cycling strategy is then studied by two-level simulations with variable coarse-grid integration time in Section 4.5.2. Then, the ability of the multilevel method to deal with more levels is evaluated in Section 4.5.3 by three- and four-level simulations. Finally, a more detailled analysis of the multilevel subgrid closure is performed by *a priori* tests in Section 4.5.4.

All the simulations use integration time steps evaluated with a CFL condition of 0.95 on each grid. Since compressibility effects remain very small (maximum variation of density of 4% close to the walls), the results are also compared to those obtained by Moser *et al.* [30] in their incompressible DNS. In all the computational cases, numerical instabilities induced by intense negative values of the coefficient $C_d^{(n)}$ are classicaly prevented by plane-averaging in the homogeneous directions.

Tables II and III present, for the low- and high-Reynolds cases respectively, the different simulation parameters, including the number of grids used, grid spacings in wall units, the subgrid model used, and the integrated friction Reynolds number and velocity. The CPU times, nondimensionnalized by the time required for the monolevel simulation, are also presented.

TABLE II

Simulation Parameters and Integrated Values for the Low-Reynolds Case

Case	Grid	Δx^+	Δy^+	$\Delta z^+_{\min}$	SGS model	Re_τ	u_τ	t_{CPU}
Lo1G-dyn	1	35	12	1	Dyn. Smag.	185	6.36×10^{-2}	1
Lo2G-dyn	1	35	12	1	Dyn. Smag.	177	6.05×10^{-2}	0.41
	2	70	24	2				
Lo2G-ML	1	35	12	1	Multilevel			
	2	70	24	2				
(a) $\Delta t_2 = 2\Delta t_1$						179.5	6.2×10^{-2}	0.45
(b) $\Delta t_2 = \Delta t_1$						182.2	6.23×10^{-2}	0.675
(c) $\Delta t_2 = 4\Delta t_1$						180.5	6.17×10^{-2}	0.3
Lo2G-MLnoS	1	35	12	1	Multilevel	179.3	6.19×10^{-2}	0.43
	2	70	24	2	$C_d^{(2)} = 0$			
Lo3G-ML2	1	30	8	0.75	Multilevel	180.5	6.21×10^{-2}	0.48
	2	60	16	1.5				
	3	120	32	3.2				
Lo4G-ML2	1	30	8	0.75	Multilevel	181.7	6.24×10^{-2}	0.23
	2	60	16	1.5				
	3	120	32	3.2				
	4	240	64	6.6				
DNS	1	17.7	5.9	0.054	—	178.5	6.33×10^{-2}	—

TABLE III

Simulation Parameters and Integrated Values for the High-Reynolds Case

Case	Grid	Δx^+	Δy^+	$\Delta z^+_{\min}$	SGS model	Re_τ	u_τ	t_{CPU}
Hi1G-dyn	1	71	15.4	1	Dyn. Smag.	594	5.2×10^{-2}	1
Hi2G-dyn	1	71	15.4	1	Dyn. Smag.	581	5.05×10^{-2}	0.41
	2	142	30.8	2				
Hi2G-ML	1	71	15.4	1	Multilevel			
	2	142	30.8	2				
(a) $\Delta t_2 = 2\Delta t_1$						589	5.17×10^{-2}	0.45
(b) $\Delta t_2 = \Delta t_1$						599	5.24×10^{-2}	0.675
(c) $\Delta t_2 = 4\Delta t_1$						588	5.16×10^{-2}	0.3
Hi2G-MLnoS	1	71	15.4	1	Multilevel	594	5.2×10^{-2}	0.43
	2	142	30.8	2	$C_d^{(2)} = 0$			
Hi3G-ML	1	71	15.4	1	Multilevel	593	5.2×10^{-2}	0.21
	2	142	30.8	2				
	3	284	61.6	4.4				
Hi3G-ML2	1	40	10	0.75	Multilevel	603	5.28×10^{-2}	1.25
	2	80	20	1.5				
	3	160	40	3.4				
Hi4G-ML2	1	40	10	0.75	Multilevel	604	5.29×10^{-2}	0.6
	2	80	20	1.5				
	3	160	40	3.4				
	4	320	80	8.2				
DNS	1	9.7	4.8	0.044	—	587.2	5.36×10^{-2}	—

4.5.1. Influence of the subgrid closure. In this section, the influence of the subgrid closure used on the coarse levels is studied by two-level simulations ($N = 2$), performed with one time step on each grid per V-cycle (i.e., $\Delta t_2 = 2\Delta t_1$). The results of two-level simulations using respectively the classical dynamic Smagorinsky closure on both the two levels (runs Lo2G-dyn and Hi2G-dyn) and the multilevel closure (runs Lo2G-ML(a) and Hi2G-ML(a)) are compared to those from the fine reference LES (runs Lo1G-dyn and Hi1G-dyn) and to the DNS results from Moser *et al.* [30]. Since the dynamic Smagorinsky term present in the multilevel closure on the coarse levels accounts only for nonlocal interactions between distant wavenumbers of the spectrum, it appears interesting to study the effect of removing this term on the coarse levels. Indeed, following the results of the works of Kerr *et al.* [20], Domaradzki *et al.* [6], and Kraichnan [22], the main part of the energy transfer is due to local interactions between neighboring wavenumbers. This is the basis of all deconvolution approaches in which an approximate reconstruction of scales half the size of the resolved ones is performed to compute the model. The runs Lo2G-MLnoS and Hi2G-MLnoS have thus been performed to see whether the dynamic Smagorinsky term is necessary on the coarse levels, by setting $C_d^{(n)}$ to zero on the coarse grid.

First, it is seen from Tables II and III that all the LES performed in this section provide skin-friction parameters in good agreement with the reference DNS results. The skin-friction velocity is slightly underestimated, due to the second-order-accurate scheme used and compressibility effects, but the error with the DNS values remains lower than 6% in all cases. The computed skin-friction Reynolds numbers, however, are very close to the targeted DNS value. While the computed skin-friction parameters vary little from one two-level simulation to another, slightly better results are obtained in the case of two-level simulations using the multilevel closure (runs Lo2G-ML(a), Lo2G-MLnoS, Hi2G-ML(a), and Hi2G-MLnoS).

One can see how the use of two nested grids reduces the CPU times, with the ratio presented in the last column. As excepted, the multilevel closure is a little more expensive than the classical one, with a 10% increase in CPU time. This is due to the larger amount of operations required to compute the scale-similarity terms in the multilevel closure, which are not present in the classical dynamic eddy-viscosity closure. However, the CPU gain factor remains very interesting and greater than two in all the two-level cases considered here.

Figure 7 compares, in wall units, the mean plane-averaged streamwise velocity profiles obtained in each case for the low-Reynolds simulation with the DNS results and the theoretical logarithmic law. One can see that the use of the multilevel algorithm with a classical SGS model provides poor results (run Lo2G-dyn), which are quite equivalent to those obtained in a coarse simulation, both in the linear and logarithmic zones. However, the use of the multilevel SGS closure, both with and without the dynamic Smagorinsky part (runs Lo2G-ML(a) and Lo2G-MLnoS) provides better results, presenting only small discrepancies from those of the fine monolevel LES (run Lo1G-dyn) or the DNS. Both the linear and logarithmic zones are well captured.

The mean plane-averaged streamwise velocity profiles obtained in the high-Reynolds case are plotted on Fig. 8. We see here the effect of the quite coarse mesh used for the second-order-accurate numerical scheme, resulting in a difference between the computed velocity field by LES and the DNS results. These computations appear unable to get the correct slope in the logarithmic zone. This is a classical effect of any second-order-accurate scheme, which is also observed by other authors (see the numerical studies of Kravchenko and Moin [23] and of Shah and Ferziger [34] for instance). Nevertheless, we can see again that the use of the multilevel SGS closure (runs Hi2G-ML(a) and Hi2G-MLnoS) leads to results in very good agreement with the fine monolevel ones (run Hi1G-dyn), while the use

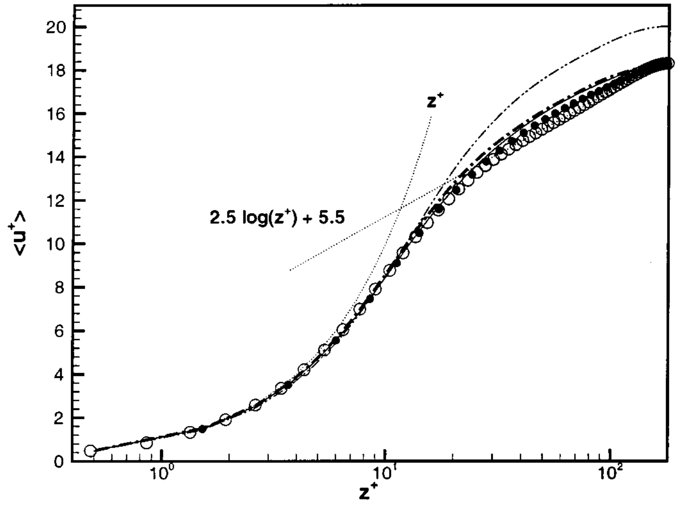


FIG. 7. Mean streamwise velocity profiles—low Reynolds case. ○: DNS Moser *et al.*, ●: Lo1G-dyn, –··–··–: Lo2G-dyn, —: Lo2G-ML(a), –·–·–: Lo2G-MLnoS, ·········: Wall laws.

of a classical SGS model on both the two levels (run Hi2G-dyn) leads to some differences in the logarithmic zone.

Figure 9 presents, in wall units, the resolved plane-averaged turbulent kinetic energy profiles (k) obtained in each case for the low-Reynolds simulations. The same remarks as for the mean profiles can be made. The two-level simulation with the standard dynamic Smagorinsky model (run Lo2G-dyn) exhibits a general coarse simulation behavior. The peak value is obtained at $z^+ = 18$, and its amplitude is very high compared with the monolevel LES (run Lo1G-dyn) and DNS results. The use of the multilevel SGS closure (runs Lo2G-ML(a) and Lo2G-MLnoS) provides a great improvement of the results: The peak value is obtained at $z^+ = 14$, as in the fine monolevel LES and in the DNS. The amplitude of the peak value is a little overestimated in the case where no dynamic Smagorinsky term is

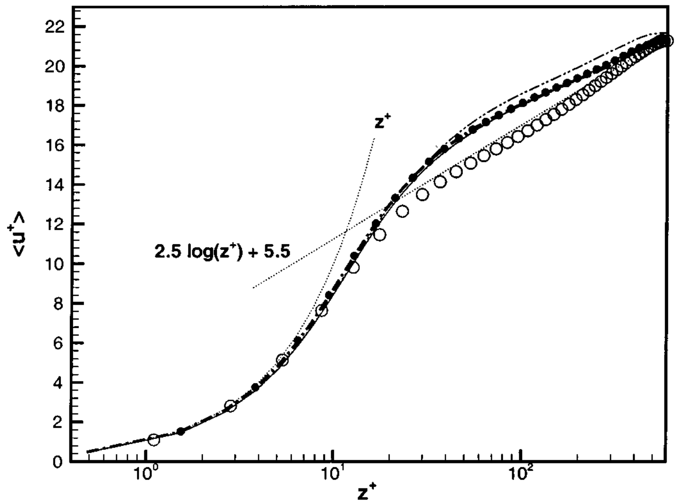


FIG. 8. Mean streamwise velocity profiles—high Reynolds case. ○: DNS Moser *et al.*, ●: Hi1G-dyn, –··–··–: Hi2G-dyn, —: Hi2G-ML(a), –·–·–: Hi2G-MLnoS, ·········: Wall laws.

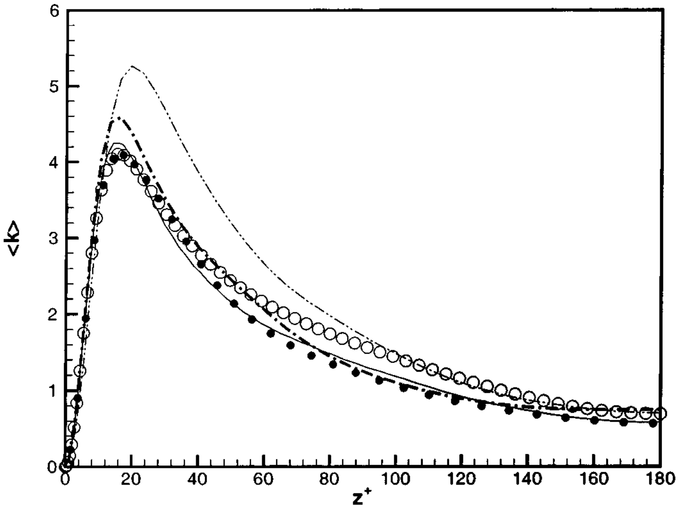


FIG. 9. Resolved turbulent kinetic energy profiles—low Reynolds case. ○: DNS Moser *et al.*, ●: Lo1G-dyn, –··–··–: Lo2G-dyn, —: Lo2G-ML(a), –·–·–: Lo2G-MLnoS.

present on the coarse level, while a little less turbulent energy is present in the core region of the channel. This can be interpreted as an underdissipative behavior of the model on the coarse grid in the buffer zone close to the wall. However, the differences between the results of runs Lo2G-ML(a) and Lo2G-MLnoS remain small.

The plane-averaged turbulent kinetic energy profiles obtained in the high-Reynolds simulations are presented on Fig. 10. Again, the effects of the second-order scheme are visible, by an overestimation of the peak turbulent kinetic energy value. Nevertheless, while all the two-level LES performed tend to overestimate the peak value, one can see that the results obtained with the multilevel SGS closure are better than those obtained with the dynamic Smagorinsky model on the two grids (run Hi2G-dyn), and again they are in very good agreement with the fine monolevel LES (run Hi1G-dyn). This can be seen for both the amplitude

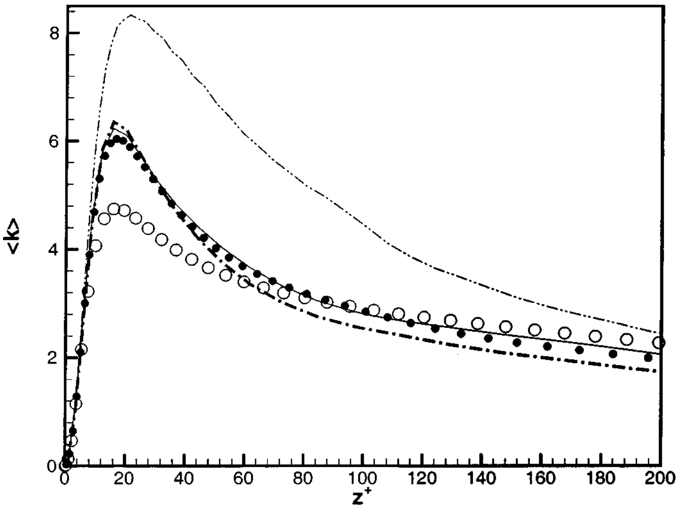


FIG. 10. Resolved turbulent kinetic energy profiles—high Reynolds case. ○: DNS Moser *et al.*, ●: Hi1G-dyn, –··–··–: Hi2G-dyn, —: Hi2G-ML(a), –·–·–: Hi2G-MLnoS.

of the peak value and its position ($z^+ = 17$ for the DNS, $z^+ = 18$ for the Hi1G-dyn, Hi2G-ML(a), and Hi2G-MLnoS simulations, and $z^+ = 21$ for Hi2G-dyn). Again some small differences are seen between the two multilevel simulations with or without the dynamic Smagorinsky part on the coarse level, the Hi2G-MLnoS giving again a little less turbulent energy in the core region and a slightly higher peak value.

All the results presented in this section tend to demonstrate that the use of a well-suited SGS closure is required in the multilevel case. This is because most of the integration time is performed on the coarsest grid. Hence, the global results are strongly influenced by the coarse-grid simulation. The model proposed in Section 2.3 allows us to take into account the fine-grid information even on the coarse grid and thus minimize the effects of the coarse-grid time integration in the multilevel algorithm. The results obtained with this model in the multilevel simulations are greatly improved and tend to show the efficiency of such a closure, which allows us to recover the results of a fine monolevel LES at lower cost.

It is globally observed that the use of the dynamic Smagorinsky term on the coarse level has only a small influence on the final results. However, this term provides some additional dissipation, as will be shown in Section 4.5.4.

4.5.2. Influence of coarse grid integration times. The multilevel cycling strategy used in this study relies on the quasi-static approximation of the smallest resolved scales. Thus, it is interesting to study the effect of increasing or reducing the time during which this approximation is imposed, that is, the time interval during which coarse-grid time integration is performed. In this section, this point is studied by performing two-level simulations with different integration times on the coarse level. Runs Lo2G-ML(b) and Hi2G-ML(b) have been performed, by reducing the integration time on the coarse level by a factor of two in comparison with the runs Lo2G-ML(a) and Hi2G-ML(a), which use one time step on each grid per V-cycle. In contrast, runs Lo2G-ML(c) and Hi2G-ML(c) have been performed, by increasing this time by a factor of two on the coarse level in comparison with runs Lo2G-ML(a) and Hi2G-ML(a).

From Figs. 11 and 13, presenting respectively the mean streamwise velocity and resolved turbulent kinetic energy profiles obtained in the low-Reynolds case, it is seen that the

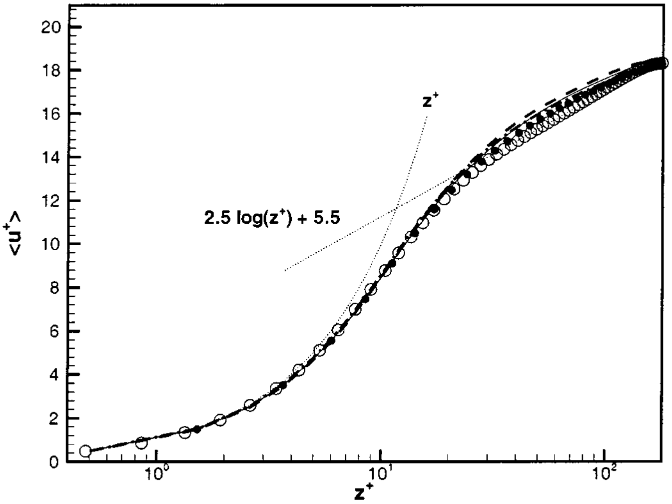


FIG. 11. Mean streamwise velocity profiles—low Reynolds case. ○: DNS Moser *et al.*, ●: Lo1G-dyn, —: Lo2G-ML(a), –·–·–: Lo2G-ML(b), – – –: Lo2G-ML(c), ·········: Wall laws.

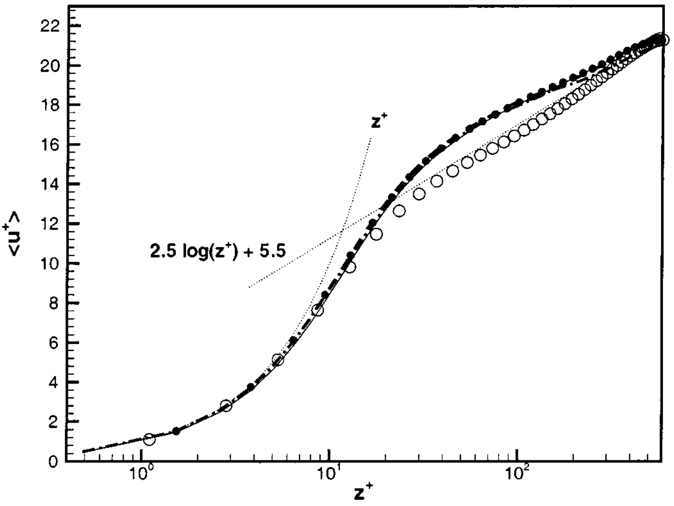


FIG. 12. Mean streamwise velocity profiles—high Reynolds case. ○: DNS Moser *et al.*, ●: Hi1G-dyn, —: Hi2G-ML(a), –·–·-: Hi2G-ML(b), – – –: Hi2G-ML(c), ·········: Wall laws.

influence of Δt_2 on the final results remains small. Only small differences in the mean profiles are seen for the run Lo2G-ML(c) performed with a greater value of Δt_2, in which a slightly smaller value of the skin-friction velocity used for normalization is obtained (see Table II). However, reducing Δt_2 does not improve the results since they are already in good agreement with the reference simulations when $\Delta t_2 = \Delta t_1$.

For the high-Reynolds case, it is first seen from Figs. 12 and 14, showing the mean streamwise velocity and resolved turbulent kinetic energy profiles, that the results obtained in the case Hi2G-ML(c) are very similar to those obtained in the case Hi2G-ML(a). That means that increasing Δt_2 by a factor of two does not have any influence on the results in this case. Only small differences in the mean streamwise velocity profiles are seen in the

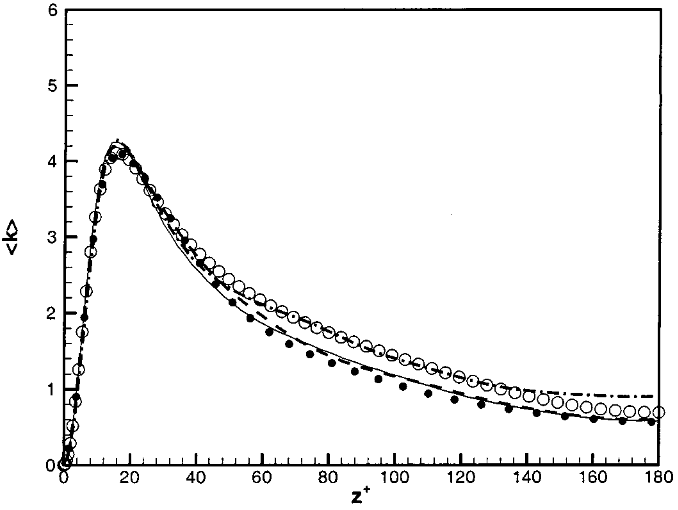


FIG. 13. Resolved turbulent kinetic energy profiles—low Reynolds case. ○: DNS Moser *et al.*, ●: Lo1G-dyn, —: Lo2G-ML(a), –·–·–: Lo2G-ML(b), – – –: Lo2G-ML(c).

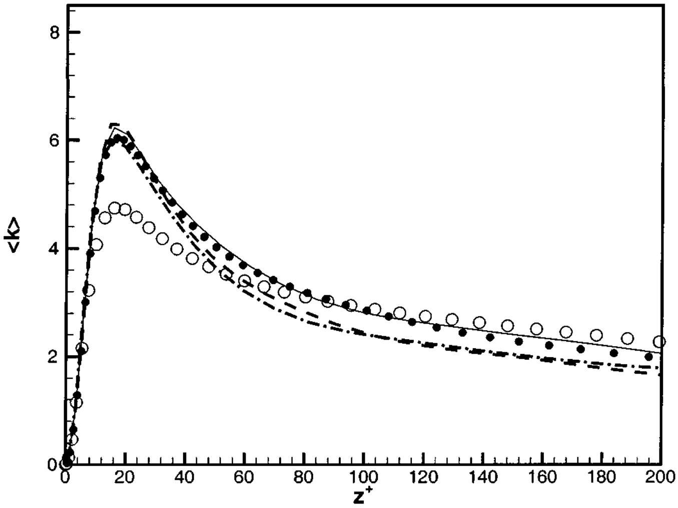


FIG. 14. Resolved turbulent kinetic energy profiles—high Reynolds case. ○: DNS Moser *et al.*, ●: Hi1G-dyn, —: Hi2G-ML(a), –·–·–: Hi2G-ML(b), – – –: Hi2G-ML(c).

core region of the channel for the case Hi2G-ML(b), owing to a slightly higher value of the skin-friction velocity obtained in this case. However, the results of the three Hi2G-ML(x) runs are very similar.

4.5.3. Influence of the number of levels. The multilevel formalism introduced in the theoretical part of this work is very general and is written for an unlimited number of levels. However, it is clear that the algorithm is limited in practice to a reasonable number of levels, to keep the number of computational points high enough so that an LES can be performed at each level of resolution.

In this section, the ability of the multilevel algorithm to deal with more than two levels is shown by three- and four-level computations. Again, one time step per V-cycle is performed on each grid, leading to $\Delta t_2 = 2\Delta t_1$, $\Delta t_3 = 4\Delta t_1$, and $\Delta t_4 = 8\Delta t_1$.

In the low-Reynolds-number case, a finer grid is introduced to carry out these computations (see Section 4.1 for description). Three-level (run Lo3G-ML2) and four-level (run Lo4G-ML2) computations using the multilevel model have been performed with this new mesh resolution.

In the high-Reynolds case, a three-level computation has been performed using the original mesh (run Hi3G-ML). To see if the use of a finer grid can improve the results, three- and four-level simulations have also been carried out with the use of a very fine grid including more than two million points, which is also described in Section 4.1 (runs Hi3G-ML2 and Hi4G-ML2).

All the computation parameters are summarized in Tables II and III. First, it is observed that all these simulations give skin-friction parameters in very good agreement with the monolevel LES and the DNS, and this for both the two Reynolds numbers considered here. For the low-Reynolds case, it is observed that even with a finer mesh used, the multilevel computations considered here took at most half the time needed for the run Lo1G-dyn. For the high-Reynolds case, it is striking that the CPU time reduction reaches a factor of nearly five for the Hi3G-ML run. The computations performed with the finer mesh (runs Hi3G-ML2 and Hi4G-ML2) give a skin-friction velocity value in very good agreement with the DNS ones, while the skin-friction Reynolds number is slightly overestimated, because of the

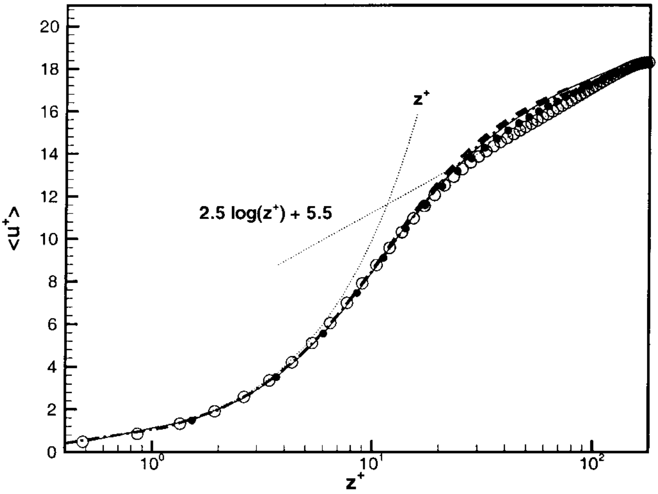


FIG. 15. Mean streamwise velocity profiles—low Reynolds case. ○: DNS Moser *et al.*, ●: Lo1G-dyn, —: Lo2G-ML(a), – – –: Lo3G-ML2, –·–·–: Lo4G-ML2, ·········: Wall laws.

small variation of density at the wall, which is not taken into account in the incompressible DNS.

Figure 15 shows the mean streamwise velocity profiles obtained for the low-Reynolds case, in wall units. All the multilevel simulations considered here agree well with the fine Lo1G-dyn reference result. From Fig. 17, which presents the mean resolved turbulent kinetic energy profiles, it is seen that the three-level run Lo3G-ML2 is in very good good agreement with the two-level Lo2G-ML(a) and also the fine Lo1G-dyn reference simulation. The four-level run Lo4G-ML2 gives slightly different results: The peak value is smaller than in the other LES cases, and more turbulent energy is present in the core region. However, the agreement with the reference results remains quite good.

Figure 16 shows the mean streamwise velocity profiles obtained for the high-Reynolds case. Here, the three-level run Hi3G-ML, even while performed with the original mesh,

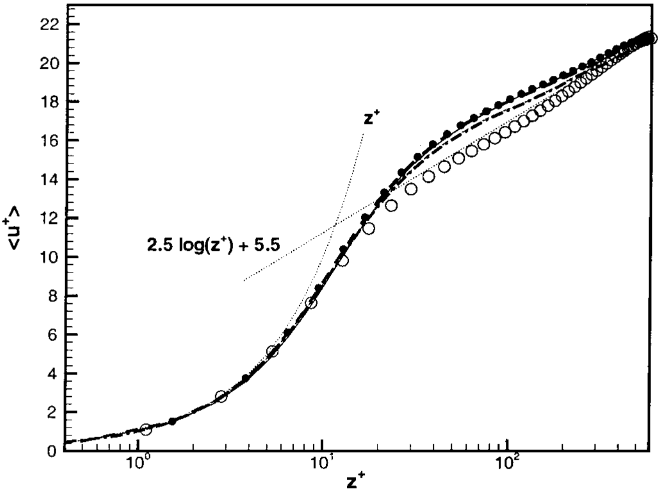


FIG. 16. Mean streamwise velocity profiles—high Reynolds case. ○: DNS Moser *et al.*, ●: Hi1G-dyn, —: Hi2G-ML(a), – – –: Hi3G-ML, -·-·-·-: Hi3G-ML2, –·–·–: Hi4G-ML2, ·········: Wall laws.

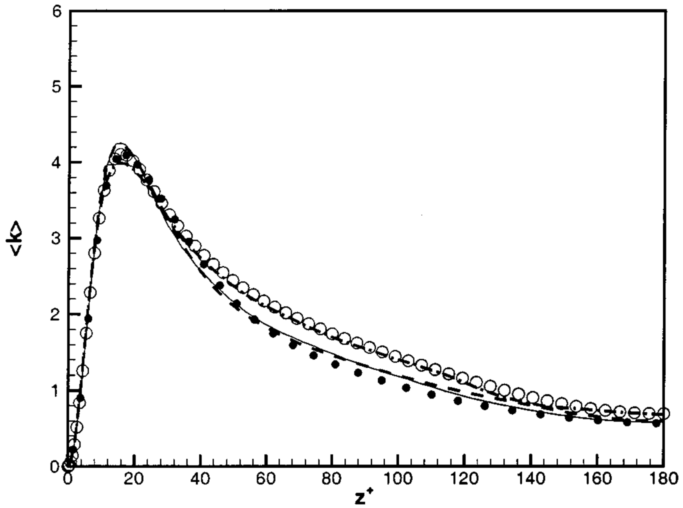


FIG. 17. Resolved turbulent kinetic energy profiles—low Reynolds case. ○: DNS Moser *et al.*, ●: Lo1G-dyn, —: Lo2G-ML(a), – – –: Lo3G-ML2, –·–·–: Lo4G-ML2.

gives the same results as the Hi1G-dyn and Hi2G-ML runs. The same comment can be made for the mean resolved turbulent kinetic energy profiles plotted on Fig. 18, where just a little less energy in the core region of the flow is seen.

For runs Hi3G-ML2 and Hi4G-ML2, which are performed with finer grids, an improvement of the results is obtained in comparison with the results obtained with the original grids. In these cases, the logarithmic zone of the flow is better described, and the resolved turbulent kinetic energy peak value is decreased in comparison with the other LES performed here. Some discrepancies with the DNS are still present, due to the second-order-accurate scheme used, but it is shown that the results are improved in comparison with the Hi1G-dyn run, with acceptable CPU time ratios (1.25 and 0.6 for runs Hi3G-ML2 and Hi4G-ML2, respectively).

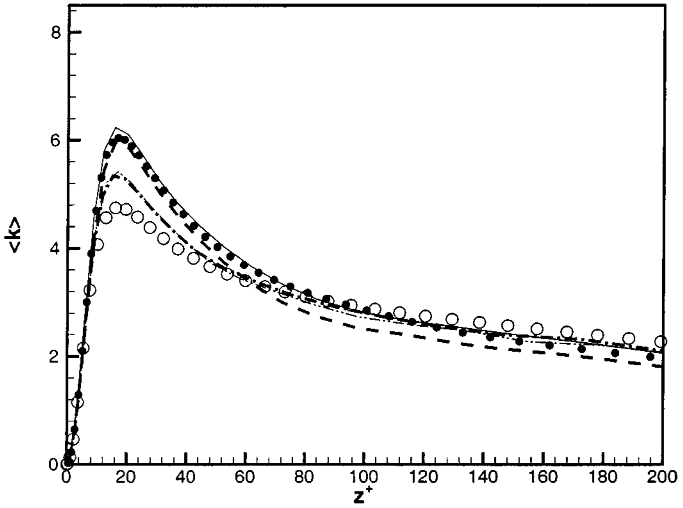


FIG. 18. Resolved turbulent kinetic energy profiles—high Reynolds case. ○: DNS Moser *et al.*, ●: Hi1G-dyn, —: Hi2G-ML(a), – – –: Hi3G-ML, ·········: Hi3G-ML2, –·–·–: Hi4G-ML2.

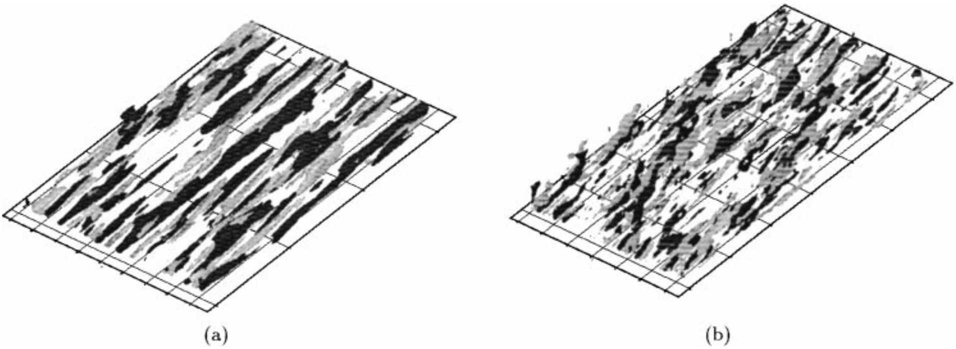


FIG. 19. Instantaneous streaks visualizations in the (a) Lo4G-ML2 and (b) Hi4G-ML2 runs. Dark and pale surfaces indicate positive and negative streamwise vorticity contours, respectively.

For all the three- and four-level simulations presented here, it should be noted that the streaks are very poorly or even not resolved at all on the coarsest grid, since the resolution is too coarse (see Tables II and III). However, three-dimensional visualizations of some iso-streamwise-vorticity surfaces of the flow highlight the fact that these structures are well represented by the multilevel computations (see Fig. 19 for the four-level simulation results). This indicates that these structures are contained in the frequency complements between the different levels and demonstrates the ability of capturing some very small local phenomena with a multilevel approach.

4.5.4. Multilevel closure analysis. In this section, an *a priori* analysis of the multilevel subgrid model is performed for the low-Reynolds number simulation. The results, extracted from the two-level LES performed with the multilevel closure (run Lo2G-ML(a)) are compared to previous subgrid term analysis performed by Domaradzki *et al.* [8] and Horiuti *et al.* [17] using filtered DNS results. Notice that no time-averaging has been performed here.

At each level, the expression of the subgrid stress tensor can be split into two parts: a scale-similarity part $L^{(n)}$ and a Smagorinsky part $S_m^{(n)}$,

$$\left(\tau_{ij}^{(n)}\right)^* = \left(L_{ij}^{(n)}\right)^* \underbrace{-2\nu_{SGS}^{(n)}\overline{\rho}^n\left(\tilde{\tilde{S}}_{ij}^n\right)^*}_{S_{m_{ij}}^{(n)}}, \tag{43}$$

where $\nu_{SGS}^{(n)} = (C_d^{(n)}(\overline{\Delta}^n)^2|\tilde{\tilde{S}}^n|)$ is generally referred to as subgrid viscosity.

For any n, the scale-similarity term $L^{(n)}$ can be split into two terms $L_1^{(n)}$ and $L_2^{(n)}$. The tensor $L_1^{(n)}$ is identical to the classical Bardina scale-similarity term, involving products between the components of the field resolved at the level n only ($\tilde{\tilde{u}}_i^n$), while $L_2^{(n)}$ contains products involving the components of the field resolved on the finer levels ($\sum_{l=1}^{n-1}\delta u_i^l$) and accounts for interactions between the two frequency bands $[0, k_n]$ and $[k_n, k_1]$ from the spectrum. From its definition, it is evident that $L_2^{(n)} = 0$ on the finest grid $n = 1$.

The contributions of each of the terms $S_m^{(n)}$, $L_1^{(n)}$, and $L_2^{(n)}$ to the global SGS stress tensor are plotted in wall units in Figs. 20–23, which present their plane-averaged trace-free (xx) and (xz) components for the fine and the coarse grids. One can see that the main part of the SGS tensor is due to the scale-similarity term L_1, while the other terms are quite negligible. This conclusion has been drawn by several authors in *a priori* testings of DNS data which have highlighted the high correlation coefficient existing between the

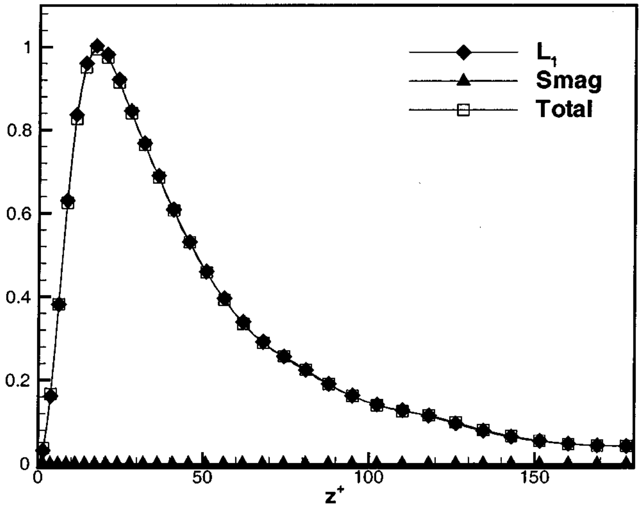


FIG. 20. SGS tensor trace-free τ_{11} component profiles—fine grid.

scale-similarity term L_1 and the real SGS tensor τ. Nevertheless, *a posteriori* testings have shown that the Smagorinsky term is important in providing sufficient SGS dissipation in a numerical simulation [27]. Notice also that, as expected, the amplitude of the components of the SGS tensor is greater on the coarse grid, showing that more subgrid scales are present and need to be modeled.

It is noticed that L_2 remains negative, but no conclusion about this can be drawn directly, since only its effect in terms of dissipation can be interpreted.

The dissipation profiles for the fine (resp. coarse) grid are plotted in wall units in Figs. 24 (resp. 25), 26 (resp. 27), and 28 (resp. 29), showing respectively the total (ε), forward (ε^-), and backward (ε^+) plane-averaged SGS dissipations provided by each of the terms S_m, L_1, and L_2.

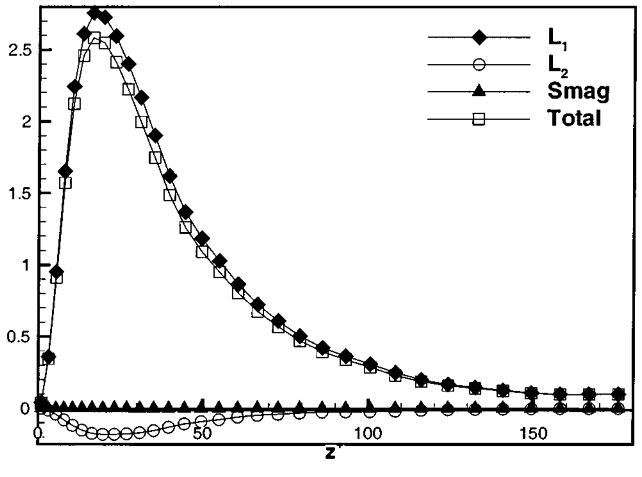


FIG. 21. SGS tensor trace-free τ_{11} component profiles—coarse grid.

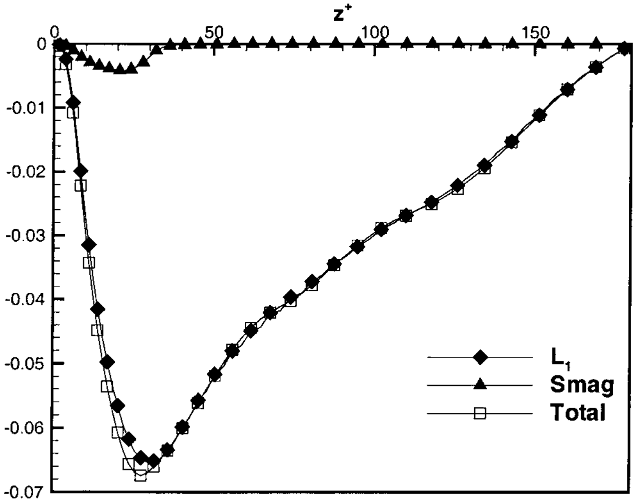


FIG. 22. SGS tensor τ_{13} component profiles—fine grid.

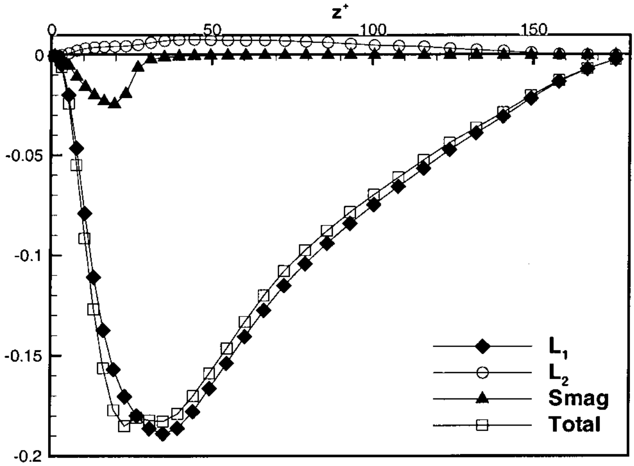


FIG. 23. SGS tensor τ_{13} component profiles—coarse grid.

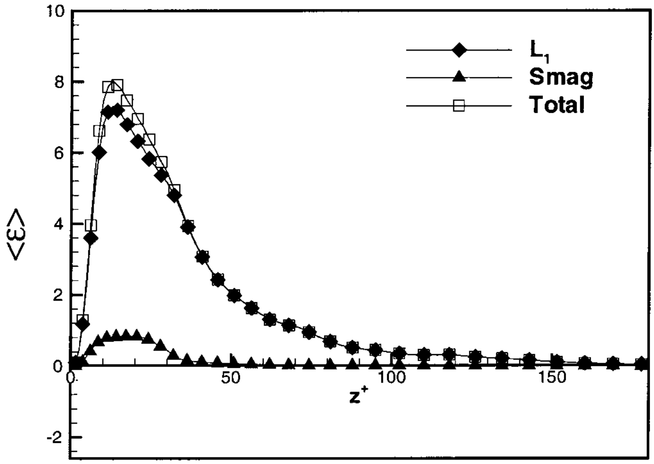


FIG. 24. SGS dissipation profiles—fine grid.

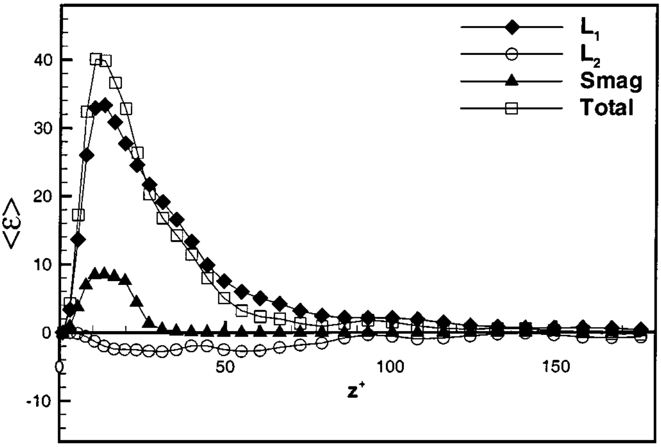


FIG. 25. SGS dissipation profiles—coarse grid.

These three terms are computed as

$$\begin{aligned}\varepsilon &= -\tau_{ij}^{(n)}\widetilde{\widetilde{S_{ij}}}^{n}\\ \varepsilon^{-} &= \frac{1}{2}(\varepsilon + |\varepsilon|)\\ \varepsilon^{+} &= \frac{1}{2}(\varepsilon - |\varepsilon|).\end{aligned}$$

Figures 24 and 25 reveal that the main SGS dissipation is due to the Bardina term, while the Smagorinsky term provides some nonnegligible additional dissipation. One can see from Fig. 25 that the additional term L_2 provides a small general backscatter effect (negative dissipation), but this seems to be quite negligible compared with the other term contributions. The global dissipation peak value is obtained at $z^+ \simeq 12$, which is in good agreement with filtered DNS results [8].

It is to be noted here that the forward dissipation provided by the dynamic Smagorinsky term on the coarse grid represents between approximately 10 and 20% of the total SGS dissipation. This confirms the studies of Kerr *et al.* [20] and Domaradzki *et al.* [6], which have highlighted the fact that the nonlocal energy transfers are one order of magnitude lower than the total one. Moreover, this term does not provide any backscatter effect, as can be seen on Figs. 28 and 29.

These figures show that local backscatter effects are taken into account by the two scale-similarity terms L_1 and L_2. These two terms are able to take into account the interactions of scales close to the cutoff lengthscale, and thus they exhibit some local backscatter phenomena. This is a well-known particularity of scale-similarity models, in which the SGS stress-tensor axis are not aligned with those of the strain-rate tensor.

Moreover, Figs. 26–29 show that the total forward (resp. backward) subgrid dissipation peak value is obtained at $z^+ \simeq 14$ (resp. $z^+ \simeq 27$), which is again in quite good agreement with filtered DNS results [17] in which it is predicted at $z^+ \simeq 12$ (resp. $z^+ \simeq 25$).

While the contribution of the term L_2 to the global subgrid dissipation seems to be negligible, Fig. 29 shows that it is not the case for its contribution to local backward dissipation, which appears to be at least as important as that of the classical Bardina term L_1.

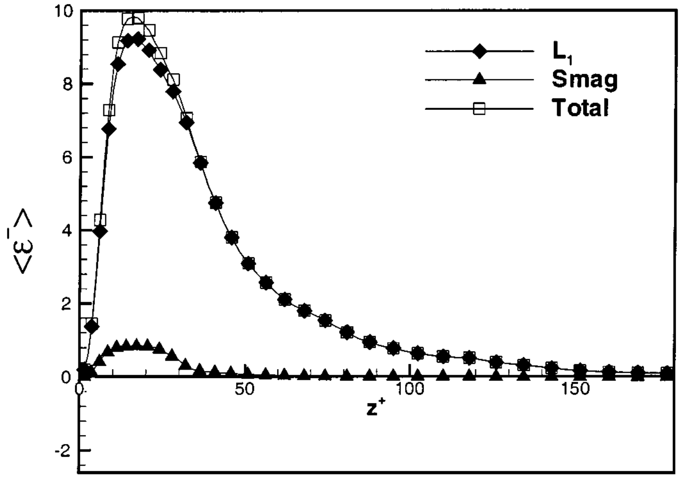


FIG. 26. Forward SGS dissipation profiles—fine grid.

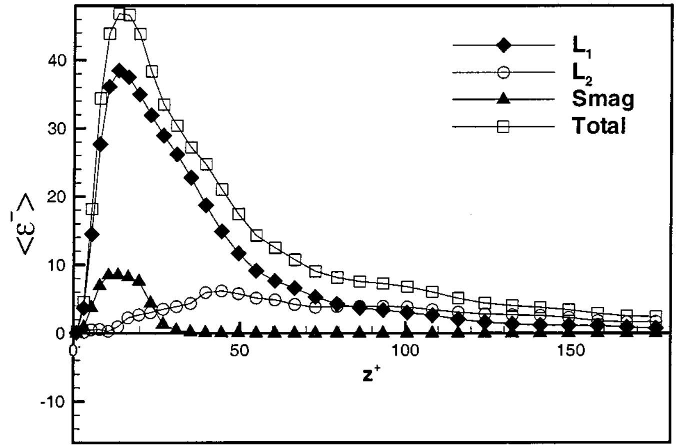


FIG. 27. Forward SGS dissipation profiles—coarse grid.

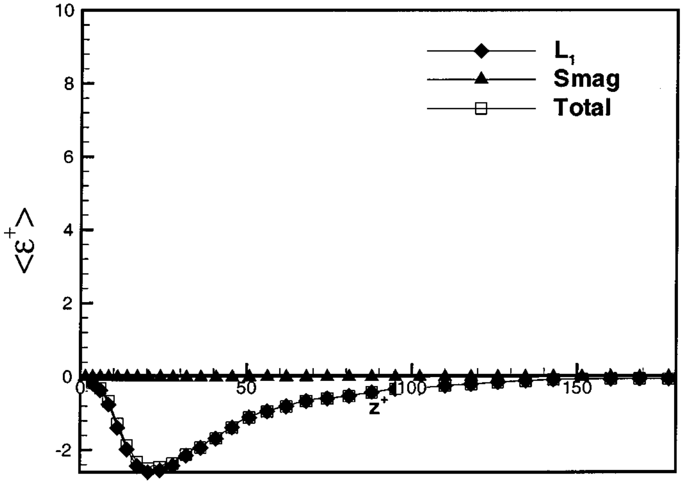


FIG. 28. Backward SGS dissipation profiles—fine grid.

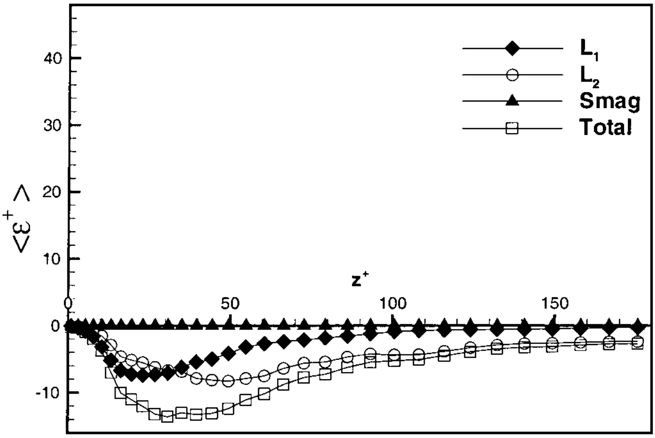


FIG. 29. Backward SGS dissipation profiles—coarse grid.

5. CONCLUSIONS

A multilevel algorithm, close to the multigrid methodology, and its application to the large-eddy simulation of turbulent compressible flows has been proposed. The use of several nested computational grids has been presented from a frequential point of view, with the result that, under the quasi-static approximation, the high-frequency part of the flow resolved on the finest grid can be used directly for the subgrid term computation on the coarse grids. This is achieved by way of a two-part mixed model in which the scale-similarity Bardina part is modified on the coarse grids by the use of a multilevel decomposition of the flow variables, while a dynamic Smagorinsky part is added to take into account the interactions of the resolved frequencies with the unresolved ones.

It has been shown that the use of a suited multilevel closure is required, by simulations performed in a plane channel flow configuration, for two values of the Reynolds number. Numerical simulations conducted in both cases show that the coarse-grid time integration has a strong influence on the results and that the use of a classical statistical subgrid model leads to poor results. However, the proposed multilevel closure allows us to take into account on the coarse grids the deterministic information computed on the fine ones and thus to minimize the statistically modeled part at each level of resolution. A great improvement is obtained on the quality of the results with the use of this model, and its behavior shows good agreement with the subgrid terms extracted from filtered Direct Simulations.

Globally, the algorithm is shown to significantly reduce the CPU times, with a savings of up to 80% for the test case considered in the paper, and without any significant loss of accuracy on the results when the multilevel closure is used. The method has been assessed with up to four levels, and it has been demonstrated that it can correctly account for very fine phenomena, without having to solve them with much accuracy.

ACKNOWLEDGMENTS

The authors thank Dr. V. Couaillier and Dr. V. Gleize for useful discussions.

REFERENCES

1. B. J. Boersma, M. N. Kooper, F. T. M. Nieuwstadt, and P. Wesseling, Local grid refinement in large-eddy simulations, *J. Eng. Math.* **32**, 161 (1997).

2. D. Carati and E. V. Eijnden, On the self-similarity assumption in dynamic models for large eddy simulations, *Phys. Fluids* **9**(7), 2165 (1997).
3. A. W. Cook, A consistent approach to large-eddy simulation using adaptative mesh refinement, *J. Comput. Phys.* **154**, 117 (1999).
4. A. Debussche, T. Dubois, and R. Temam, *The Nonlinear Galerkin Method: A Multi-Scale Method Applied to the Simulation of Homogeneous Turbulent Flows*, ICASE Report No 93-93 (1993).
5. J. A. Domaradzki, T. Dubois, and A. Honein, A subgrid-scale estimation model applied to large eddy simulations of compressible turbulence, in *Annual Research Briefs* (Center for Turbulence Research, 1998), p. 351.
6. J. A. Domaradzki and W. Liu, Energy transfer in numerically simulated wall-bounded turbulent flows, *Phys. Fluids* **6**(4), 1583 (1994).
7. J. A. Domaradzki and K. C. Loh, The subgrid-scale estimation model in the physical space representation, *Phys. Fluids* **11**(8), 2330 (1999).
8. 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).
9. T. Dubois, F. Jauberteau, and R. Temam, Dynamic multilevel methods in turbulence simulation, *Comput. Fluids Dyn. Rev.* 679 (1995).
10. T. Dubois, F. Jauberteau, and R. Temam, A comparative study of multilevel schemes in homogeneous turbulence, in *Lecture Notes in Physics* (Springer–Verlag, New York, 1996), Vol. 490, p. 388.
11. T. Dubois, F. Jauberteau, and R. Temam, Incremental unknowns, multilevel methods and the numerical simulation of turbulence, *Comput. Methods Appl. Mech. Eng.* **159**, 123 (1998).
12. M. Germano, A proposal for a redefinition of the turbulent stresses in the filtered Navier–Stokes equations, *Phys. Fluids* **29**(7), 2323 (1986).
13. M. Germano, Turbulence: The filtering approach, *J. Fluid Mech.* **238**, 325 (1992).
14. M. Germano, A statistical formulation of the dynamic model, *Phys. Fluids* **8**(2), 565 (1996).
15. A. Harten, Multiresolution representation of data: A general framework, *SIAM J. Numer. Anal.* **33**(3), 1205 (1996).
16. A. Harten, *Multiresolution Representation and Numerical Algorithms: A Brief Review*, ICASE Report No. 94-59 (October 1994).
17. K. Horiuti, Assessment of the subgrid-scale models at low and high Reynolds numbers, in *Annual Research Briefs* (Center for Turbulence Research, 1996), p. 211.
18. K. Horiuti, A new dynamic two-parameter mixed model for large-eddy simulation, *Phys. Fluids* **9**(11), 3443 (1997).
19. T. J. R. Hugues, G. R. Feijóo, L. Mazzei, and J. B. Quincy, The variational multiscale method—A paradigm for computational mechanics, *Comput. Meth. Appl. Mech. Eng.* **166**, 3 (1998).
20. M. R. Kerr, J. A. Domaradzki, and G. Barbier, Small-scale properties of nonlinear interactions and subgrid-scale energy transfer in isotropic turbulence, *Phys. Fluids* **8**, 197 (1996).
21. J. Kim, P. Moin, and R. Moser, Turbulence statistics in fully developped channel flow at low Reynolds number, *J. Fluid Mech.* **177**, 133 (1987).
22. R. H. Kraichnan, Inertial-range transfer in two- and three-dimensional turbulence, *J. Fluid Mech.* **47**(3), 525 (1971).
23. A. G. Kravchenko and P. Moin, On the effect of numerical errors in large eddy simulation of turbulent flows, *J. Comput. Phys.* **131**, 310 (1997).
24. E. Lenormand, P. Sagaut, and L. Ta Phuoc, Large-eddy simulation of compressible channel flow at moderate Reynolds number, *Int. J. Numer. Meth. Fluids* **32**, 369 (2000).
25. E. Lenormand, P. Sagaut, L. Ta Phuoc, and P. Comte, Subgrid-scale models for large-eddy simulation of compressible wall bounded flows, *AIAA J.* **38**(8), 1340 (2000).
26. D. K. Lilly, A proposed modification of the Germano subgrid-scale closure method, *Phys. Fluids A* **4**(3), 633 (1992).
27. S. Liu, C. Meneveau, and J. Katz, On the properties of similarity subgrid-scale models as deduced from measurements in a turbulent jet, *J. Fluid Mech.* **275**, 83 (1994).

28. M. Manhart and R. Friedrich, Towards DNS of turbulent boundary layers, in *Direct and Large-Eddy Simulation III* (Kluwer Academic Dordrecht/Norwell, MA, 1999), p. 429.
29. P. Moin, K. Squires, W. Cabot, and S. Lee, A dynamic subgrid-scale model for compressible turbulence and scalar transport, *Phys. Fluids*, **3**(11) 2746 (1991).
30. R. Moser, J. Kim, and N. N. Mansour, Direct numerical simulation of turbulent channel flow up to $\mathrm{Re}_\tau = 590$, *Phys. Fluids* **11**(4), 943 (1999).
31. P. Sagaut, *Introduction à la simulation des grandes échelles pour les écoulements de fluide incompressible, Mathématiques et applications* (Springer-Verlag, Berlin/New York, 1998), Vol. 30.
32. P. Sagaut and R. Grohens, Discrete filters for large eddy simulation, *Int. J. Numer. Meth. Fluids* **31**, 1195 (1999).
33. M. V. Salvetti and S. Banerjee, A priori tests of a new dynamic subgrid-scale model for finite-difference large-eddy simulations, *Phys. Fluids*, **7**(11), 2831 (1995).
34. K. B. Shah and J. H. Ferziger, A new non-eddy viscosity subgrid-scale model and its application to channel flow, in *Annual Research Briefs* (Center for Turbulence Research, 1995), pp. 73–90.
35. J. Smagorinsky, General circulation experiments with the primitive equations, *Mon. Weather Rev.* **3**, 99 (1963).
36. S. Stolz and N. A. Adams, An approximate deconvolution procedure for large-eddy simulation, *Phys. Fluids* **11**(7), 1699 (1999).
37. P. P. Sullivan, J. C. McWilliams, and C. Moeng, A grid nesting method for large-eddy simulation of planetary boundary-layer flows, *Boundary-Layer Meteorol.* **80**,167 (1996).
38. P. R. Voke, *Multiple Mesh Simulation of Turbulent Flow*, Report QMW EP-1082 (University of London, 1990).
39. A. W. Vreman, B. J. Geurts, J. G. M. Kuerten, and P. J. Zandbergen, A finite volume approach to large eddy simulation of compressible, homogeneous, isotropic, decaying turbulence, *Int. J. Numer. Meth. Fluids* **15**, 799 (1992).
40. C. Weber, F. Ducros, and A. Corjon, *Large-Eddy Simulation of Complex Turbulent Flows*, Technical Paper 98-2651 (AIAA Press, Washington, DC, 1998).
41. Y. Zang, R. L. Street, and J. R. Koseff, A dynamic mixed subgrid-scale model and its applications to turbulent recirculating flows, *Phys. Fluids* **5**(12), 3186 (1993).