AIAA JOURNAL
Vol. 46, No. 7, July 2008

Growth of Rounding Errors and Repetitivity of Large-Eddy Simulations

Jean-Mathieu Senoner,* Marta García,* Simon Mendez,* Gabriel Staffelbach,† and Olivier Vermorel†
Centre Européen de Recherche et de Formation Avancée en Calcul Scientifique,
31057 Toulouse Cedex 01, France
and
Thierry Poinsot‡
Institut de Mécanique des Fluides,
31400 Toulouse, France

DOI: 10.2514/1.34862

This paper studies the propagation of rounding errors in large-eddy simulation and shows that instantaneous flowfields produced by large-eddy simulation are partially controlled by these rounding errors and depend on multiple parameters: number of processors used for parallel simulation (even in an explicit code), changes in initial conditions (even of the order of machine accuracy), machine precision (simple, double, or quadruple), etc. Using a laminar Poiseuille pipe flow, a fully developed turbulent channel flow, and a complex burner geometry as test cases, results show that only turbulent flows exhibit a high sensitivity to these parameters. These results confirm that large-eddy simulation reflects the true nature of turbulence insofar as it may exponentially amplify infinitely small perturbations on initial conditions in time. However, they highlight an often overlooked limitation of large-eddy simulation in terms of validation and prediction of unsteady phenomena.

Introduction

LARGE-EDDY simulation (LES) has become the most efficient tool to predict nonreacting [1,2], as well as reacting, turbulent flows [3–8]. The main strength of LES compared with classical Reynolds-averaged methods is that, like direct numerical simulation (DNS) [9–11], LES explicitly captures large-scale unsteady motions due to turbulence, instead of modeling them. This feature implies that like DNS, LES is also subject to the separation of trajectories [12,13]: the flow solution exhibited by LES is very sensitive to any small perturbations of a given reference state. This limits the predictability time of LES/DNS because the initial conditions of a simulation aiming to reproduce a natural phenomenon are always affected by measurement uncertainties, and the determination of predictability times of numerical simulations has been an important field of investigation [14,15]. An often ignored aspect is that the numerical perturbations occurring in LES lead to another predictability issue: simulations started with strictly identical initial conditions may yield different results after a certain time. These numerical perturbations have different sources:

1) Rounding errors are the first source of random noise in any finite precision computation; they constitute an unavoidable forcing for the Navier–Stokes equations and may lead to LES variability. The study of error growth in finite precision computations is an important topic in applied numerical mathematics [16–19] but has found few applications in multidimensional fluid mechanics because of the complexity of the codes used in computational fluid dynamics (CFD).

2) Because of its large computational resource requirements, modern LES heavily relies on parallel computing. Therefore, in codes using domain partitioning, i.e., most of them, the treatment of interfaces is an additional "noise" source in the Navier–Stokes equations. Even in explicit codes, where the algorithm is independent of the number of processors, the different summation orders with which a nodal value is reconstructed at partition interfaces may induce nonassociativity errors. For example, in explicit codes on unstructured meshes using cell vertex methods [20], the residual at one node is obtained by adding the weighted residuals of the surrounding cells. Of course, the exact result of this addition is independent of the addition ordering but this is not true for rounding errors. Therefore, additions of more than two summands may yield distinct results for floating-point accumulation. For example, the rounding errors in $(a + b) + c$ and in $a + (b + c)$ may be different, in particular if there are large differences in orders of magnitude between the summands [21] and, after a few tens of thousands time steps, the LES result may be affected. Because the propagation of these rounding errors is induced by nondeterministic message arrival at partition interfaces, such behavior may occur for any unstructured parallel CFD code, regardless of the numerical schemes used. As a consequence, the simulation output might change when run on a different number of processors. The case of implicit codes in time [2,3,22] or in space, such as compact schemes [23–25], is not considered here; for such schemes, the methods [26,27] used to solve the linear system appearing at each time step depend on the number of processors. Therefore, the propagation of rounding errors is not the only reason why solutions obtained with different numbers of processors vary.

3) Even on a single processor computation, internal parameters of the partitioning algorithm may couple with rounding errors to force the LES solution. For example, a different reordering of nodes using the Cuthill–McKee (CM) or the reverse Cuthill–McKee (RCM) algorithm [28,29] may produce the same effect as a simple perturbation and can be the source of solution divergence.

4) Finally, compilation options, in particular those affecting code optimization, and obviously those affecting truncation operations, are a fourth source of LES variability. Different optimization options of the compiler are not tested in the following. Such tests would certainly be of interest, although care must be taken because too

Received 28 September 2007; revision received 21 February 2008; accepted for publication 25 February 2008.

*Ph.D. Student, Computational Fluid Dynamics Team.

†Senior Research Fellow, Computational Fluid Dynamics Team.

‡Research Director, Institut National Polytechnique de Toulouse and Centre National de la Recherche Scientifique. Associate Fellow AIAA.

aggressive optimization options can affect scheduling of operations in a physically wrong manner and lead to erroneous results.

The solution of a given LES/DNS at a certain instant unavoidably changes when the rounding errors are not exactly identical, and LES/DNS solutions are known to have a meaning only in a statistical manner [30]. It is, however, a real difficulty in the practical use of LES/DNS because it means that running the same simulation on two different machines or one machine with a different number of processors is equivalent to a perturbation of initial conditions and can lead to different instantaneous results after a certain time. For steady flows in the mean, statistics should not depend on these changes and mean profiles should be identical. However, when the objective of the LES is the study of unsteady phenomena, such as ignition or quenching in a combustor [31,32], knowing that results depend on these parameters is certainly a sobering thought and a drawback in terms of industrial exploitation. This paper tries to address these issues and answer a simple question which is of interest for all practitioners of LES: How does the instantaneous solution produced by LES depend on the number of processors used to run the simulation? On the initial condition? On internal details of the algorithm?

The first section gives an example of the effects of the number of processors in a simple case: a rectangular turbulent channel flow computed with a fully explicit LES code [33]. This example shows that even in an explicit code, running a simulation twice on a different number of processors can lead to totally different instantaneous solutions. The second section then gives a systematic description of the effects of rounding errors in three flows: a turbulent channel flow, a laminar Poiseuille flow, and a complex burner flow. For all cases, the difference between two instantaneous solutions obtained by changing either the number of processors, the initial condition, or the node ordering is quantified in terms of norms between the two solutions. The effects of time step and machine precision (simple, double, and quadruple) are also investigated in this section. All simulations have been performed on an IBM JS21 supercomputer.

The numerical solver uses a cell vertex formulation, i.e., the discrete values of the conserved variables are stored at the cell vertices, whereas fluxes are obtained by averaging along the cell edges [20]. A compact conservative formulation of the compressible Navier–Stokes equations is considered:

$$\frac{\partial \mathbf{w}}{\partial t} + \nabla \cdot \mathcal{F} = \mathbf{0} \tag{1}$$

where $\mathbf{w}$ represents the vector of conservative variables, and $\mathcal{F}$ is the tensor of both viscous and inviscid fluxes. The Green–Gauss theorem is used to compute the numerical residual in each computational cell K_e:

$$\mathbf{r}_e := \frac{1}{V_e} \oint_{\partial K_e} \mathcal{F}_h \cdot \mathbf{n}\, \mathrm{d}S \tag{2}$$

where $\mathcal{F}_h$ and V_e, respectively, denote a suitable numerical approximation of the flux $\mathcal{F}$ and the volume of the computational cell K_e.

In the semidiscrete form, the scheme then writes:

$$\frac{\partial \mathbf{w}_j}{\partial t} = \frac{1}{V_j} \sum_{e \in D_j} \mathbf{D}_{j,e} V_e \mathbf{r}_e \tag{3}$$

where $\mathbf{D}_{j,e}$ is the distribution matrix that weighs the residual of cell K_e to the node j, and thus depends on the numerical scheme. V_j is the volume of the dual cell associated with the node j. The spatial discretization described here can be combined to explicit time-stepping approaches, such as Runge–Kutta, to obtain a fully discretized scheme. The scheme used here for all simulations is the Lax–Wendroff scheme [33,34]. The fully discretized scheme writes

$$\frac{\mathbf{w}_j^{n+1} - \mathbf{w}_j^n}{\Delta t} = \frac{1}{V_j} \sum_{e \in D_j} \left(\frac{1}{n_v^e} \mathbf{I} + \frac{\Delta t}{2 V_e} \mathcal{A}_e^n \cdot \mathbf{n}_{j,e} \right) V_e \mathbf{r}_e \tag{4}$$

where n_v^e, $\mathcal{A}_e^n$, and $\mathbf{n}_{j,e}$, respectively, denote the number of vertices of the cell K_e, the Jacobian matrices of the cell K_e, and the normal vector associated with the dual cell of the node j on the cell K_e. Additional tests were performed using a third-order scheme in space and time [35], resulting in the same conclusions.

Effects of Number of Processors on Fully Developed Turbulent Channel Flow LES

The first example is the LES of a rectangular fully developed turbulent channel flow with channel dimensions 75 × 25 × 50 mm (Fig. 1). A pressure gradient is applied to a periodic channel flow and random disturbances are added to pass transition to turbulence. There are no boundary conditions except for the walls. The Reynolds number is $Re_\tau = \delta u_\tau / \nu = 1500$, where δ is half the channel height and u_τ the friction velocity at the wall, $u_\tau = (\tau_{\text{wall}}/\rho)^{1/2}$ with τ_{wall} being the wall stress. The mesh contains 30^3 hexahedral elements, and it is not refined at walls where a law-of-the-wall [8] is used. The first grid point is at a reduced distance $y^+ = y u_\tau / \nu \approx 100$ from the wall. The subgrid model is the Smagorinsky model, the value of the constant is $C_S = 0.18$. The Courant–Friedrichs–Lewy (CFL) number λ controlling the time step Δt is $\lambda = \max[(u + c)\Delta t / \Delta]$, where u is the local convective velocity, c the speed of sound, and Δ the mesh size. The chosen value of the CFL number is $\lambda = 0.7$. For all simulations discussed next, the initial condition corresponds to a snapshot of the flow at a given instant, long after turbulence was initialized, so that it is fully established. The domain partitioning method is perfectly equivalent on any number of processors. The recursive inertial bisection [36,37] algorithm has been used to partition the grid, and the Cuthill-McKee algorithm is considered as the default node reordering strategy on subdomains.

Figures 2–4 show fields of axial velocity in the central plane of the channel at three instants after the run initialization. Two simulations performed on, respectively, four (TC1) and eight processors (TC2) with identical initial conditions are compared. The characteristics of all presented simulations are displayed in Tables 1 and 2. The specific times correspond to (in wall units) $t^+ = 7.68$, $t^+ = 18.43$, and $t^+ = 26.11$, respectively, where $t^+ = u_\tau t / \delta$. Obviously, the two flowfields observed at $t^+ = 7.68$ are identical. However, at $t^+ = 18.43$, differences start to become visible. Finally, at $t^+ = 26.11$, the instantaneous flowfields obtained in TC1 and TC2 are totally different. Even though the instantaneous flowfields are different, statistics remain the same: mean and root mean square axial velocity profiles averaged over $t^+ \approx 60$ are identical for both simulations, as can be seen in Figs. 5 and 6.

This very simple example illustrates the main question of the present work: Is the result of Figs. 2–4 reasonable? If it is not a simple programming error (the next section will show that it is not), can other parameters produce similar effects?

Sensitivity of Laminar and Turbulent Flows to Small Perturbations

To understand how LES can produce diverging instantaneous results such as those shown in the previous section, simple tests were performed to investigate the effects of various aspects of the methodology: 1) laminar/turbulent baseline flow, 2) number of processors, 3) initial condition, 4) node ordering, 5) time step,

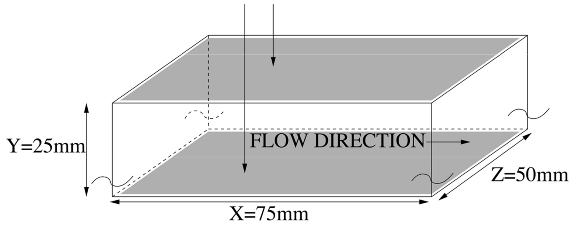
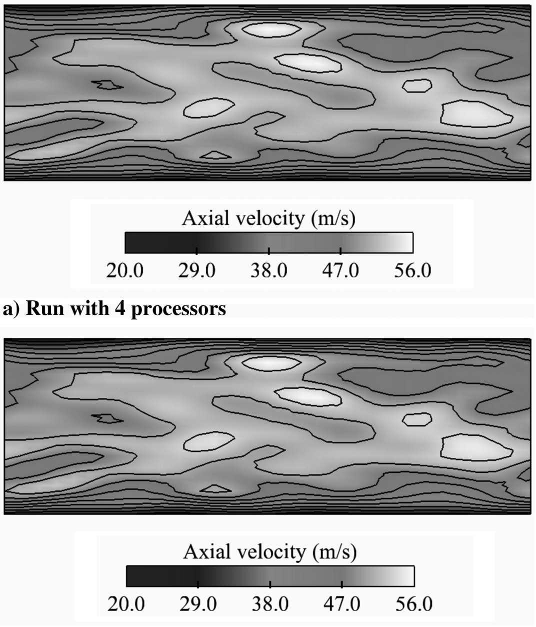


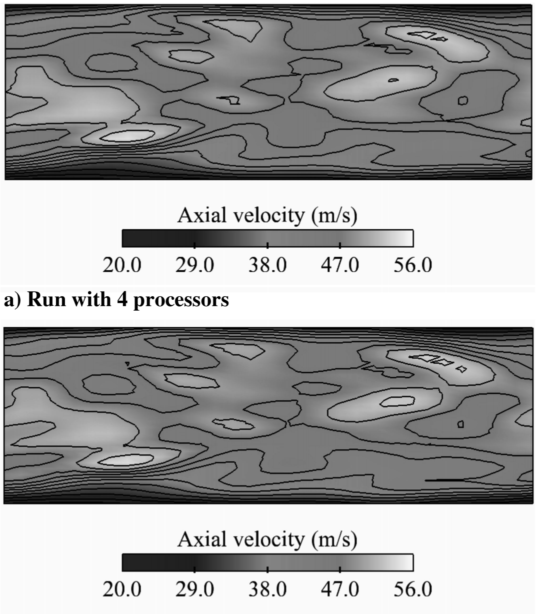
Fig. 1 Schematic of periodic channel. Upper and lower boundaries consist of walls, all other boundaries are pairwise periodic.



a) Run with 4 processors

b) Run with 8 processors

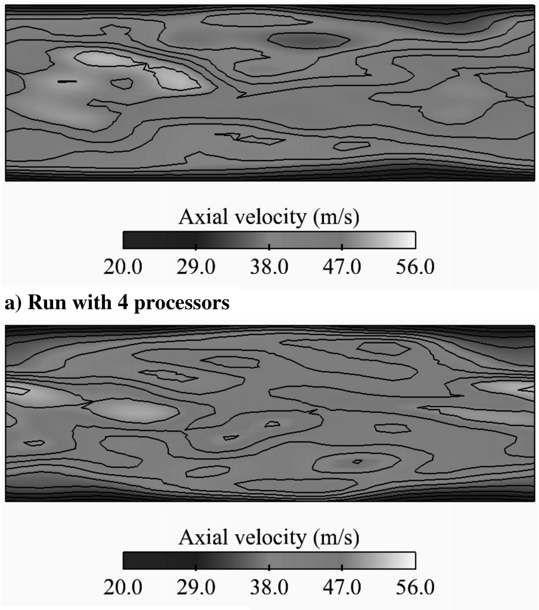
Fig. 2 Instantaneous field of axial velocity in the central plane of the channel at $t^+ = 7.68$: a) run TC1, b) run TC2.



a) Run with 4 processors

b) Run with 8 processors

Fig. 3 Instantaneous field of axial velocity in the central plane of the channel at $t^+ = 18.43$: a) run TC1, b) run TC2.



a) Run with 4 processors

b) Run with 8 processors

Fig. 4 Instantaneous field of axial velocity in the central plane of the channel at $t^+ = 26.11$: a) run TC1, b) run TC2.

6) floating-point representation according to the Institute of Electrical and Electronics Engineers (IEEE) standard, and 7) investigation of a more realistic configuration with nonperiodic boundary conditions.

For these tests, the objective is to quantify the differences between two LES solutions produced by a couple of simulations in Tables 1 and 2. Let u_1 and u_2 be the axial velocity components of two given instantaneous solutions at the same instant after initialization. A proper method to compare the latter is to use the following norms:

$$N_{\max} = \max(|u_1(\mathbf{x}) - u_2(\mathbf{x})|) \quad \text{for } \mathbf{x} \in \Omega \tag{5}$$

$$N_{\text{mean}} = \left(\frac{1}{V_\Omega} \int_\Omega [u_1(\mathbf{x}) - u_2(\mathbf{x})]^2 \, \mathrm{d}\Omega \right)^{\frac{1}{2}} \quad \text{for } \mathbf{x} \in \Omega \tag{6}$$

where Ω and V_Ω, respectively, denote the computational domain and its volume. Both norms are expressed in meters per second. $N_{\max}$ provides the maximum local axial velocity difference in the field between two solutions, whereas N_{mean} yields a volumetrically averaged axial velocity difference between the two solutions. The growth of $N_{\max}$ and N_{mean} vs the number of time steps will be used as a direct indicator for the divergence of the solutions.

Fully Deterministic LES?

First, it is useful to indicate that performing any of the LES of Table 1 twice on the same machine with the same number of

Table 1 Summary of LES runs (fully developed turbulent channel flow)

Run ID	No. processors	Initial conditions	Precision	Graph ordering	CFL λ
TC1	4	Fixed	Double	CM	0.7
TC2	8	Fixed	Double	CM	0.7
TC3	1	Fixed	Double	CM	0.7
TC4	1	Modif.	Double	CM	0.7
TC5	1	Fixed	Double	RCM	0.7
TC6	4	Fixed	Double	CM	0.35
TC7	8	Fixed	Double	CM	0.35
TC8	4	Fixed	Simple	CM	0.7
TC9	8	Fixed	Simple	CM	0.7
TC10	28	Fixed	Quadr.	CM	0.7
TC11	32	Fixed	Quadr.	CM	0.7

Table 2 Summary of laminar runs (Poiseuille flow)

Run ID	No. processors	Initial conditions	Precision	Graph ordering	CFL λ
LP1	4	Fixed	Double	CM	0.7
LP2	8	Fixed	Double	CM	0.7

processors, the same initial conditions, and the same partition algorithm, leads to exactly the same solution: $N_{\max}$ and N_{mean} being zero to machine accuracy. In that sense, the LES remains fully deterministic. However, this is true only if the order of operations at interfaces is not determined by the order of message arrival so that summations are always carried out in the same order. Otherwise, the randomness induced by the nondeterministic order of message arrival is enough to induce diverging solutions. Note that nondeterministic message arrival is usually implemented in parallel codes to improve performance, and that blocking messages order can severely affect the overall simulation cost.

Influence of Turbulence

The first test is to compare a turbulent channel flow studied in the previous section and a laminar flow. A three-dimensional Poiseuille flow was used as a test case. The Poiseuille computation is performed on a pipe geometry with 361 × 26 points. The flow is laminar and the Reynolds number based on the bulk velocity and diameter is approximately 500. The boundary conditions are set periodic at the inlet/outlet and no slip at the duct wall; a constant axial pressure gradient is imposed in the entire domain. Run parameters of the laminar Poiseuille flow are displayed in Table 2.

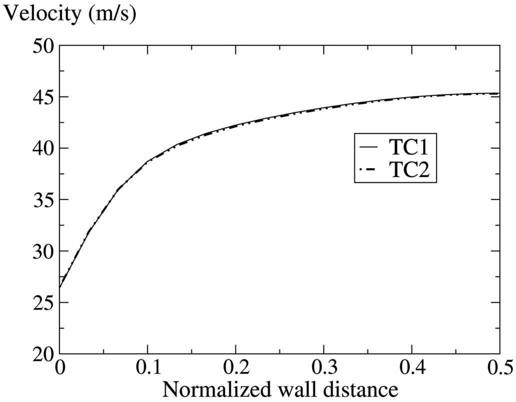


Fig. 5 Comparison of the mean velocity profiles for TC1 (4 processors) and TC2 (8 processors) over half-channel height.

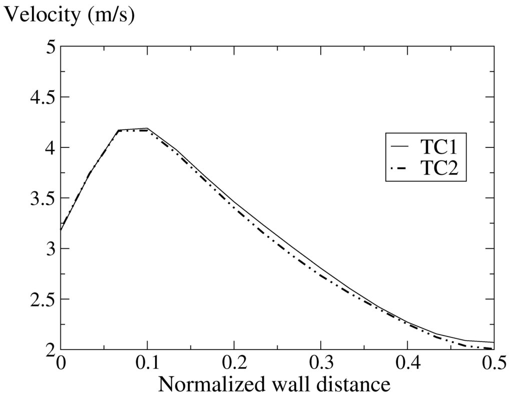


Fig. 6 Comparison of the root mean square velocity profiles for TC1 (4 processors) and TC2 (8 processors) over half channel height.

Figure 7 shows the evolutions of $N_{\max}$ and N_{mean} vs time step for runs TC1/TC2 and LP1/LP2. Note that the first point of the graph is the evaluation of the difference after one time step. The only parameter tested here is a change of the number of processors. As expected from the snapshots of Figs. 2–4, the turbulent channel flow simulations are very sensitive to a change in the number of processors, and the solutions of TC1 and TC2 diverge rapidly, leading to a maximum difference of 20 m/s and a mean difference of 3–4 m/s after 90,000 time steps. On the other hand, the difference between LP1 and LP2 hardly increases and levels off when reaching values on the order of 10^{-12} m/s, despite the periodic boundary conditions. This is expected because there is only one stable solution for the Poiseuille flow for infinite times and, accordingly, laminar flows do not induce exponential divergence of trajectories. However, this simple test case confirms that the turbulent character of the flow is the source of the divergence of solutions. This phenomenon must not be confused with the growth of a hydrodynamic mode, which is induced by the bifurcation in phase space of an equilibrium state of a given physical system. Obviously, such an equilibrium state does not exist for a fully developed turbulent channel flow. Moreover, the stagnation of absolute and mean differences between TC1/TC2 simply implies that after 90,000 time steps solutions have become fully uncorrelated and should not be misinterpreted as the saturation of an exponentially growing mode.

The basic mechanism leading to Figs. 2–4 is that the turbulent flow acts as an amplifier for rounding errors generated by the fact that the mesh is decomposed differently in TC1 and TC2. The source of this difference is the new node reordering obtained for both decompositions. This implies a different ordering when adding the contributions to a cell residual for nodes inside the subdomains, but mainly at partition interfaces. This random noise roughly starts at machine accuracy (Fig. 7) at a few points in the flow and grows continuously if the flow is turbulent.

The growth rate of the differences between solutions in simulations TC1 and TC2 cannot be estimated in a simple manner. A description for the determination of growth rates of trajectory separation in two-dimensional vortical flows is given by Leith [14], and is briefly summarized in the following. A description of vortices as points with associated circulations and negligible viscosity is assumed. Under these hypotheses, a set of linearized ordinary differential equations can be derived to evaluate the time evolution of the distance between two neighboring flowfield trajectories differing by an arbitrary infinitesimal perturbation $\delta(t)$. This system admits

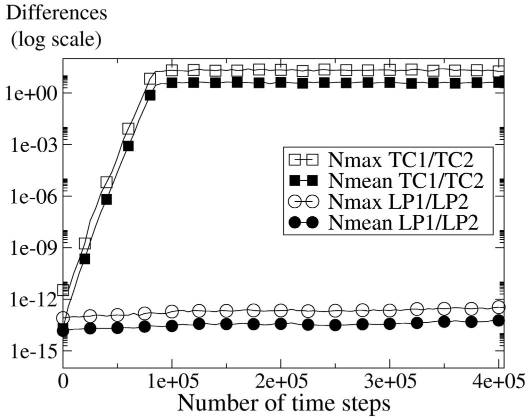


Fig. 7 Differences between solutions vs time step. Squares: turbulent channel flow. Circles: laminar Poiseuille flow.

exponential solutions, the growth rates of which are determined by the real part of the eigenvalues. The evolution of inviscid/conservative systems conserves volume in phase space. As the real part of the eigenvalues describes the separation of trajectories in time, it represents a measure of the evolution of the volume in phase space. Thus, the sum of the real parts vanishes and at least one of them has to be positive. At this stage, the number of degrees of freedom of the system imposes topological constraints on the trajectories and can prevent their separation, but a few degrees of freedom suffice for such systems to exhibit chaotic behavior, as demonstrated by the famous Lorenz attractor [38]. When considering dissipative two- or three-dimensional turbulent flows, a well-defined phase space does not exist. Therefore, predictability analysis is based on the evolution of the energy spectrum of two realizations of a given velocity field differing by a perturbation $\delta(\mathbf{u})$ but having the same statistics. It is possible to derive equations for the evolution of the error energy spectrum and define predictability times using simulations of decaying turbulence [15].

Therefore, a simple estimate of the growth rate from flow parameters a priori does not seem possible. However, one may suppose that, independently of the spatial distribution and amplitude of perturbations applied to a given turbulent flowfield, the separation of trajectories for various simulations yields similar exponential growth rates, which is confirmed in the following. Moreover, it is a purely physical phenomenon and, though induced by rounding errors, the growth rate should not depend on numerical parameters such as machine precision or time step. These aspects are addressed in forthcoming sections.

Influence of Initial Conditions

The previous section has shown that turbulence combined with a different domain partitioning (i.e., a different number of processors) is sufficient to lead to totally different instantaneous flow realizations. It is expected that a perturbation in initial conditions will have a similar effect as domain partitioning. This is verified in runs TC3 and TC4 which are run on a single processor, thereby eliminating issues linked to parallel implementation. The only difference between TC3 and TC4 is that, in TC4, the initial solution is identical to TC3, except at one random point where a single 10^{-16} m/s perturbation is applied to the streamwise velocity component. Simulations with different locations of the perturbation were run to ensure that the position did not affect results.

Figure 8 shows that the growth rate of the difference between TC3 and TC4 is exactly the same as the one observed between TC1 and TC2 (also displayed in Fig. 8): two solutions starting from a very slightly perturbed initial condition diverge as fast as two solutions starting from the same solution but running on different numbers of processors. Note that the difference between runs TC1 and TC2 comes from the accumulation of rounding errors along the interface between subdomains at each time step, whereas TC3 and TC4 differ only through the initial condition: the sequence of floating-point operations is exactly the same in TC3 and TC4. Still, the differences between TC3 and TC4 increase as fast as those between TC1 and TC2; this confirms that a turbulent flow amplifies any difference in the same manner, whether it is due to different sequences of rounding errors or to a perturbation of the initial conditions.

Effects of Node Ordering in Mesh

It has already been indicated that performing the same simulation twice (with the same number of processors and same initial conditions) leads to exactly the same result. However, this is only true as long as exactly the same code is used. It is not verified anymore as soon as a modification affecting rounding errors is done in the code. At this point, so many factors affecting rounding errors can be cited that a general discussion is pointless. This paper focuses on fully explicit codes and on one example only: the order used to add residuals at nodes in a cell vertex scheme. This order is controlled by the developer. For simulation TC5, the ordering of this addition was changed (reverse Cuthill-McKee algorithm); the residual at a given mesh node was assembled by adding the contributions to a cell residual in a different order. This change does not affect the flow data. In TC5, the node residual in a regular tetrahedral mesh is obtained by $1/4\{R_1 + [R_2 + (R_3 + R_4)]\}$ where R_i are the residuals of the cells surrounding the node and by $1/4\{R_4 + [R_3 + (R_2 + R_1)]\}$ in TC3. It has an effect, however, on rounding errors, and the cumulated effects of this nonassociativity error are what this test tries to demonstrate. TC5 and TC3 are performed with the same initial condition and run on one processor only. The only difference is the node reordering strategy.

As shown by Fig. 9, the differences between TC5 and TC3 are again similar to those observed between TC1 and TC2 (obtained by changing the number of processors). This confirms that rounding errors (and not the parallel character of the code) are the source of the solution divergence. It also shows that any modification of the code could lead to such a divergence, suggesting that repeating an LES simulation with a modified code will probably never yield the same instantaneous flowfields, potentially leading to discussions on the validity of the modifications.

Effects of Time Step

It is interesting to verify that numerical aspects do not influence the growth rate of the solutions difference and that the growth rate is only determined by the physical and geometrical parameters of the configuration. On that account, simulations TC6 and TC7 are performed with a time step reduced by a factor of two compared with simulations TC1 and TC2. TC6 and TC7 are carried out on, respectively, four and eight processors. The norms between TC6 and TC7 are displayed in Fig. 10 and compared with the norms between TC1 and TC2. From the preceding explanations, similar growth rates are expected when comparing the growth rates over physical time.

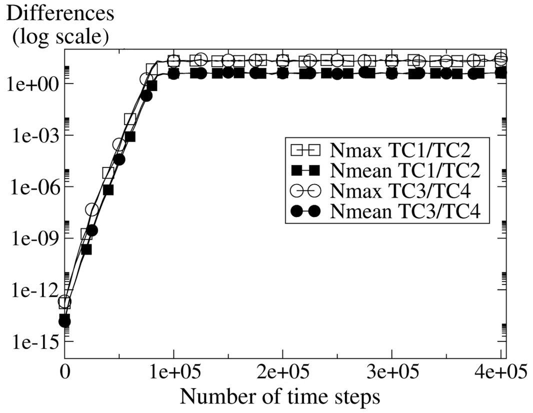


Fig. 8 Differences between solutions vs time step. Squares: different number of processors. Circles: different initial conditions.

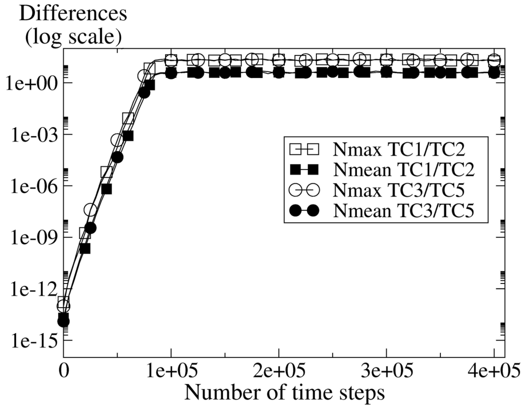


Fig. 9 Differences between solutions vs time step. Squares: different number of processors. Circles: different addition order.

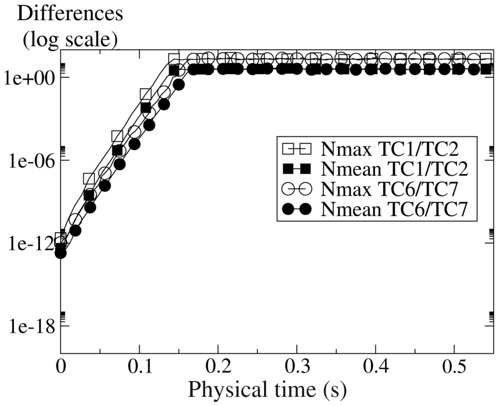


Fig. 10 Differences between solutions vs physical time. Squares: time step Δt. Circles: time step $\Delta t/2$.

The growth rates observed in Fig. 10 are indeed very similar. The slight difference is probably due to the variation of the numerical dispersion and dissipation properties of the scheme with the CFL number [39].

Effects of Machine Precision

A last test to verify that the divergence between solutions is not due to a programming error, but depends primarily on rounding errors, is to perform the same computation with simple/quadruple precision instead of double precision. Simulations TC1 and TC2 were repeated using simple precision in runs TC8 and TC9 (Table 1) and quadruple precision in TC10 and TC11. To compensate for the increase in computational time for quadruple precision simulations, roughly a factor of 10 compared with double precision, TC10 and TC11 are carried out on, respectively, 28 and 32 processors to yield a reasonable computing time. Results are displayed in Fig. 11 and compared with the difference between TC1 and TC2.

Figure 11 shows that the solutions differences for TC8/TC9 and TC10/TC11 roughly start from the respective machine accuracies (differences of 10^{-6} for simple precision after one time step, differences of 10^{-30} for quadruple precision after one time step) and increase exponentially with the same growth rate, before reaching identical difference levels for all three cases. This shows that higher precision computations cannot prevent the exponential divergence of trajectories but only delay it.

Propagation of Rounding Errors on More Realistic Configuration

The previous example corresponds to a periodic turbulent channel flow where perturbations cannot leave the computational domain. They can therefore be amplified indefinitely and this might explain the divergence of solutions observed in Figs. 2–11. To verify whether the previously discussed divergence phenomena are independent of the configuration and can also occur in nonperiodic flows, a more realistic configuration is tested.

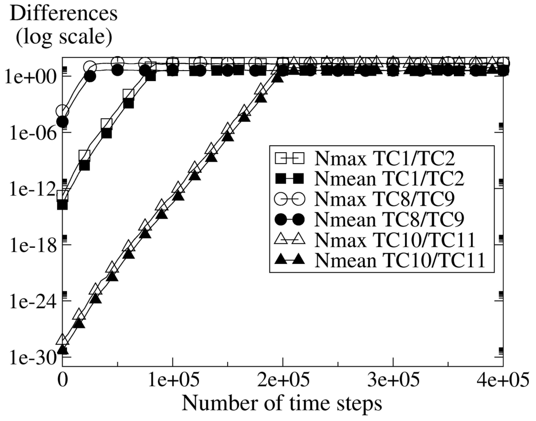


Fig. 11 Differences between solutions vs time step. Squares: double precision. Circles: simple precision. Triangles: quadruple precision.

The chosen configuration is the nonreacting flow in a complex swirled combustor including a plenum, a swirler, and a combustion chamber. The Reynolds number at the inlet of the combustion chamber (based on an equivalent radius of the swirler) is approximately 5200. To avoid the specification of the boundary condition at the outlet of the combustion chamber for further acoustic analysis, the atmosphere around the outlet has also been meshed (Fig. 12). Boundary conditions are summarized in Fig. 13. All solid boundaries are modeled using adiabatic wall laws. The inlet, coflow, and outlet boundary conditions rely on characteristic decomposition according to Moureau et al. [33] and Poinsot and Lele [40]. The inlet injects an air mixture (N_2, O_2) with 15 g/s at $T = 298$ K. The coflow imposed on the left sidewall aims to mimic air entrainment due to the outgoing flow of the combustion chamber. The coflow velocity is set to $u_z = 0.1$ m/s and $T = 298$ K.

The only parameter changing in the two simulations is the number of processors, and the parameters of the run are specified in Table 3.

A divergence of solutions similar to the periodic turbulent channel flow can be observed in Fig. 14.

Instantaneous fields of axial velocity for both runs in the central plane at $t = 32.4$ ms (80,000 time steps) after initialization show that instantaneous flowfields are uncorrelated (Fig. 15). The differences remain mostly confined to the highly turbulent zone at the inlet of the combustion chamber. This region is volumetrically small compared with the entire computed geometry, in particular the atmospheric region, which is why the mean difference curve remains low compared with the turbulent channel flow. The maximal local difference reaches values of 45 m/s.

Figure 16 displays the evolution of axial velocity for the two simulations for a point located at 20 cm of the chamber inlet plane, on the chamber axis. As expected, the signals diverge suddenly due to the exponential amplification of rounding errors, leading to uncorrelated signals after approximately 30 ms ($t = 0$ s corresponds to initialization).

These results clearly demonstrate that rounding errors may propagate until full uncorrelation, even for configurations with inlet/

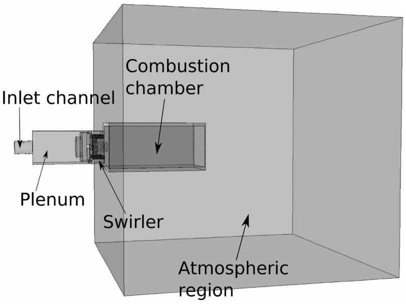


Fig. 12 Schematic of the complex burner geometry.

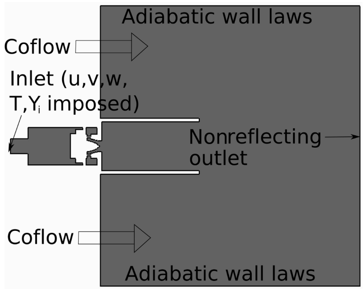


Fig. 13 Midplane cut with boundary-condition specification.

Table 3 Summary of complex burner geometry runs

Run ID	No. processors	Initial conditions	Precision	Graph ordering	CFL λ
CB1	28	Fixed	Double	CM	0.7
CB2	32	Fixed	Double	CM	0.7

outlet boundary conditions, and that it is not an artifact due to the periodicity of the turbulent channel flow (Fig. 1). The presence of recirculation zones in the complex burner geometry, where part of the fluid constantly remains in the computational domain, may be an explanation for this behavior, but this aspect requires further investigation.

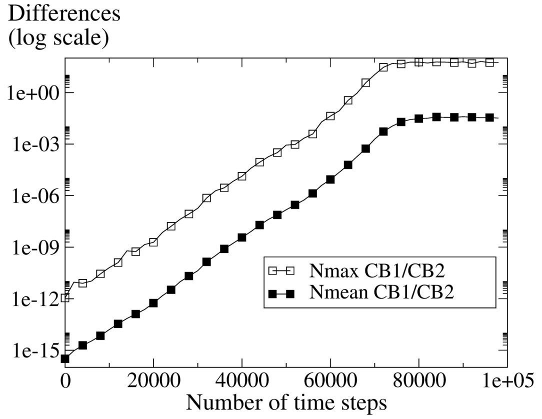
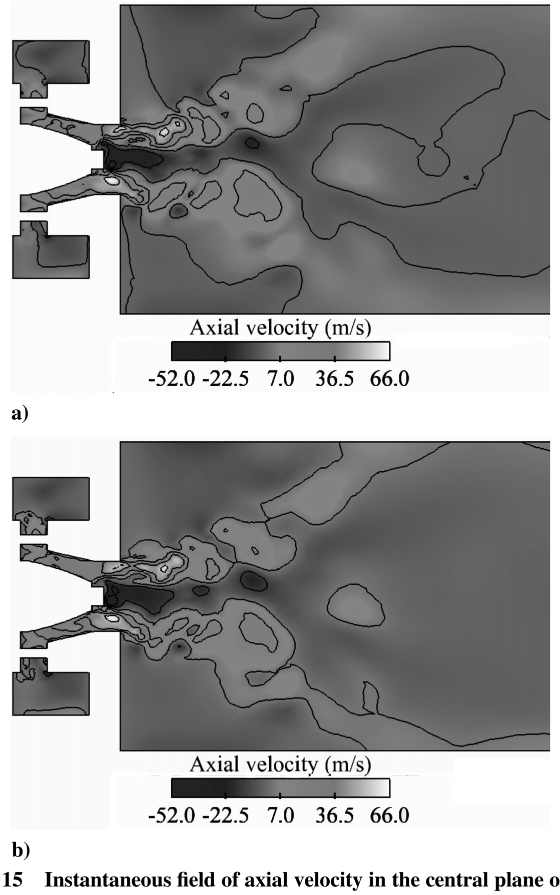


Fig. 14 Differences between CB1 and CB2 (different number of processors) vs time step.



a)

b)

Fig. 15 Instantaneous field of axial velocity in the central plane of the burner at $t = 32.4$ ms: a) run CB1, b) run CB2.

Conclusions

This work focused on the sensitivity of instantaneous large-eddy simulation fields to different rounding error propagation due to situations where various parameters of the run, such as number of processors, initial condition, time step, and changes in addition ordering of cell residuals for cell vertex methods, are modified. The baseline simulation used for the tests was a fully developed periodic turbulent channel flow, but a complex burner geometry displayed a similar behavior. The conclusions are as follows:

1) Any sufficiently turbulent flow computed in LES exhibits significant sensitivity to these parameters, leading to instantaneous solutions which can be totally different. As expected, laminar flows are almost insensitive to these parameters even for periodic simulations.

2) The divergence of solutions is due to two combined facts: 1) the exponential separation of trajectories in turbulent flows, and 2) the different propagation of rounding errors induced by domain partitioning and scheduling of operations. More generally, any change in the code lines affecting the propagation of rounding errors will have a similar effect. This implies that the validation of an LES code after modifications may only be based on statistical fields. This makes error detection in LES codes much more difficult than in classical codes; comparing instantaneous solutions is not a proper validation method for LES.

3) Small changes in initial conditions (of the order of machine accuracy at one point of the flow only) produce similar divergence of solutions.

4) Working with higher precision machines does not suppress the divergence of solutions, but delays it.

Converged statistics reflect the fact that most possible realizations of a turbulent flow have been taken. It is therefore clear that the propagation of rounding errors does not affect statistics of large-eddy and direct numerical simulations. However, instantaneous values may a priori only be used for times during which the differences between two runs with identical initial solution remain negligible with respect to a suitable error norm. The increase in floating-point representation delays the divergence of solutions, but the increase in computational costs appears too severe for practical applications. Another option consists of the use of software which inhibits the

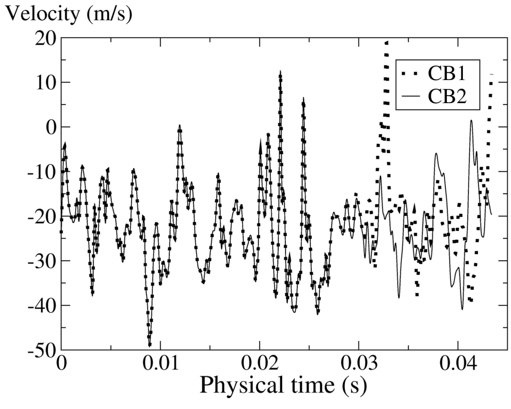


Fig. 16 Evolution of axial velocity at a point located on the chamber axis for CB1 and CB2 over time.

propagation of rounding errors. This would allow the increase of the computational predictability time of a given simulation. The ability to include such software in complex computational fluid dynamics codes must be investigated and the increase in computational expense is again a crucial aspect. More generally, these results demonstrate that the concept of numerical quality in LES will require much more detailed studies and tools than what has been used up to now for Reynolds-averaged simulations.

Acknowledgments

We thank F. Chaitin-Chatelin for helpful discussions. The first author gratefully acknowledges the funding by the European Community in the framework of the Marie Curie Early Stage Research Training Fellowship (contract number MEST-CT-2005-020426).

References

[1] Sagaut, P., *Large Eddy Simulation for Incompressible Flows*, 1st ed., Springer, Berlin, 2002.

[2] Mahesh, K., Constantinescu, G., and Moin, P., "Numerical Method for Large-Eddy Simulation In Complex Geometries," *Journal of Computational Physics*, Vol. 197, No. 1, 2004, pp. 215–240.
doi:10.1016/j.jcp.2003.11.031

[3] Di Mare, F., Jones, W. P., and Menzies, K., "Large Eddy Simulation of a Model Gas Turbine Combustor," *Combustion and Flame*, Vol. 137, No. 3, 2004, pp. 278–295.
doi:10.1016/j.combustflame.2004.01.008

[4] Poinsot, T., and Veynante, D., *Theoretical and Numerical Combustion*, 2nd ed., R. T. Edwards, Philadelphia, 2005.

[5] Pitsch, H., "Large Eddy Simulation of Turbulent Combustion," *Annual Review of Fluid Mechanics*, Vol. 38, Jan. 2006, pp. 453–482.
doi:10.1146/annurev.fluid.38.050304.092133

[6] El-Asrag, H., and Menon, S., "Large Eddy Simulation of Bluff-Body Stabilized Swirling Non-Premixed Flames," *Proceedings of the Combustion Institute*, Vol. 31, No. 2, 2007, pp. 1747–1754.
doi:10.1016/j.proci.2006.07.251

[7] Duwig, C., Fuchs, L., Griebel, P., Siewert, P., and Boschek, E., "Study of a Confined Turbulent Jet: Influence of Combustion and Pressure," *AIAA Journal*, Vol. 45, No. 3, 2007, pp. 624–661.
doi:10.2514/1.26352

[8] Schmitt, P., Poinsot, T. J., Schuermans, B., and Geigle, K., "Large-Eddy Simulation and Experimental Study of Heat Transfer, Nitric Oxide Emissions and Combustion Instability in a Swirled Turbulent High Pressure Burner," *Journal of Fluid Mechanics*, Vol. 570, 2007, pp. 17–46.
doi:10.1017/S0022112006003156

[9] Poinsot, T., Candel, S., and Trouvé, A., "Application of Direct Numerical Simulation to Premixed Turbulent Combustion," *Progress in Energy and Combustion Science*, Vol. 21, No. 6, 1995, pp. 531–576.
doi:10.1016/0360-1285(95)00011-9

[10] Moin, P., and Mahesh, K., "DNS: A Tool in Turbulence Research," *Annual Review of Fluid Mechanics*, Vol. 30, Jan. 1998, pp. 539–578.
doi:10.1146/annurev.fluid.30.1.539

[11] Vervisch, L., and Poinsot, T., "Direct Numerical Simulation of Non Premixed Turbulent Flames," *Annual Review of Fluid Mechanics*, Vol. 30, Jan. 1998, pp. 655–692.
doi:10.1146/annurev.fluid.30.1.655

[12] Tennekes, H., and Lumley, J. L., *First Course in Turbulence*, 1st ed., MIT Press, Cambridge, MA, 1972.

[13] Lesieur, M., *Turbulence in Fluids*, 1st ed., Martinus–Nijhoff, Dordrecht, The Netherlands, 1987, pp. 230–231.

[14] Leith, C., "Atmospheric Predictability and Two-Dimensional Turbulence," *Journal of the Atmospheric Sciences*, Vol. 28, No. 2, 1971, pp. 145–161.
doi:10.1175/1520-0469(1971)028<0145:APATDT>2.0.CO;2

[15] Métais, O., and Lesieur, M., "Statistical Predictability of Decaying Turbulence," *Journal of the Atmospheric Sciences*, Vol. 43, No. 9, 1986, pp. 857–870.
doi:10.1175/1520-0469(1986)043<0857:SPODT>2.0.CO;2

[16] Stoer, J. S., and Bulirsch, R., *Introduction to Numerical Analysis*, 1st ed., Springer, Berlin, 1980.

[17] Chaitin-Chatelin, F., and Frayssé, V., *Lectures on Finite Precision Computations*, 1st ed., Society for Industrial and Applied Mathematics, Philadelphia, 1996.

[18] Overton, M. L., *Numerical Computing with IEEE Floating Point Arithmetic*, 1st ed., Society for Industrial and Applied Mathematics, Philadelphia, 2001.

[19] Vignes, J., "Discrete Stochastic Arithmetic for Validating Results of Numerical Software," *Numerical Algorithms*, Vol. 37, Nos. 1–4, Dec. 2004, pp. 377–390.
doi:10.1023/B:NUMA.0000049483.75679.ce

[20] Schönfeld, T., and Rudgyard, M., "Steady and Unsteady Flow Simulations Using the Hybrid Flow Solver AVBP," *AIAA Journal*, Vol. 37, No. 11, 1999, pp. 1378–1385.

[21] Hanrot, G., Lefévre, V., Stehlé, D., and Zimmermann, P., "Worst Cases for a Periodic Function with Large Arguments," *Proceedings of the 18th IEEE Symposium on Computer Arithmetic*, edited by P. Kornerup, and J.-M. Muller, IEEE Computer Society Press, Los Alamitos, CA, 2007, pp. 133–140.

[22] Freitag, M., and Janicka, J., "Investigation of a Strongly Swirled Premixed Flame Using LES," *Proceedings of the Combustion Institute*, Vol. 31, No. 1, 2007, pp. 1477–1485.
doi:10.1016/j.proci.2006.07.225

[23] Lele, S., "Compact Finite Difference Schemes with Spectral-Like Resolution," *Journal of Computational Physics*, Vol. 103, No. 1, 1992, pp. 16–42.
doi:10.1016/0021-9991(92)90324-R

[24] Abarbanel, S. S., and Chertock, A. E., "Strict Stability of High-Order Compact Implicit Finite-Difference Schemes: The Role of Boundary Conditions for Hyperbolic PDEs, 1," *Journal of Computational Physics*, Vol. 160, No. 1, 2000, pp. 42–66.
doi:10.1006/jcph.2000.6420

[25] Sengupta, T. K., Ganerwal, G., and Dipankar, A., "High Accuracy Compact Schemes and Gibbs' Phenomenon," *Journal of Scientific Computing*, Vol. 21, No. 3, 2004, pp. 253–268.
doi:10.1007/s10915-004-1317-2

[26] Saad, Y., "Flexible Inner-Outer Preconditioned GMRES Algorithm," *Journal of Scientific Computing*, Vol. 14, No. 2, 1993, pp. 461–469.
doi:10.1137/0914028

[27] Fraysse, V., Giraud, L., and Gratton, S., "Set of Flexible-GMRES Routines for Real and Complex Arithmetics," Centre Européen de Recherche et de Formation Avancée en Calcul Scientifique TR/PA/98/20, 1998.

[28] Cuthill, E., and McKee, J., "Reducing the Bandwidth of Sparse Symmetric Matrices," *Proceedings of the 24th National Conference of the ACM*, Vol. 1, Assoc. for Computing Machinery, New York, 1969, pp. 157–172.

[29] Liu, W.-H., and Sherman, A. H., "Comparative Analysis of the Cuthill-Mckee and the Reverse Cuthill-Mckee Ordering Algorithms for Sparse Matrices," *SIAM Journal on Numerical Analysis*, Vol. 13, No. 2, 1976, pp. 198–213.
doi:10.1137/0713020

[30] Pope, S. B., "Ten Questions Concerning the Large-Eddy Simulation of Turbulent Flows," *New Journal of Physics*, Vol. 6, No. 1, 2004, p. 35.
doi:10.1088/1367-2630/6/1/035

[31] Sommerer, Y., Galley, D., Poinsot, T., Ducruix, S., Lacas, F., and Veynante, D., "Large Eddy Simulation and Experimental Study of Flashback and Blow-Off in a Lean Partially Premixed Swirled Burner," *Journal of Turbulence*, Vol. 5, No. 1, Article 37, 2004.
doi:10.1088/1468-5248/5/1/037

[32] Boileau, M., Staffelbach, G., Cuenot, B., Poinsot, T., and Bérat, C., "LES of an Ignition Sequence in a Gas Turbine Engine," *Combustion and Flame*, 2008 (to be published).

[33] Moureau, V., Lartigue, G., Sommerer, Y., Angelberger, C., Colin, O., and Poinsot, T., "High-Order Methods for DNS and LES of Compressible Multi-Component Reacting Flows on Fixed and Moving Grids," *Journal of Computational Physics*, Vol. 202, No. 2, 2005, pp. 710–736.
doi:10.1016/j.jcp.2004.08.003

[34] Hirsch, C., *Numerical Computation of Internal and External Flows: Computational Methods for Inviscid and Viscous Flows*, 1st ed., Vol. 2, Wiley, New York, 1988, pp. 224–306.

[35] Colin, O., and Rudgyard, M., "Development of High-Order Taylor-Galerkin Schemes for Unsteady Calculations," *Journal of Computational Physics*, Vol. 162, No. 2, 2000, pp. 338–371.
doi:10.1006/jcph.2000.6538

[36] Williams, R. D., "Performance of Dynamic Load Balancing Algorithms for Unstructured Mesh Calculations," *Concurrency, Practice and Experience*, Vol. 3, No. 5, 1991, pp. 457–481.
doi:10.1002/cpe.4330030502

[37] Taylor, V. E., and Nour-Omid, B., "Study of the Factorization Fill-In for a Parallel Implementation of the Finite Element Method," *International Journal for Numerical Methods in Engineering*, Vol. 37, No. 22, 1994,

pp. 3809–3823.
doi:10.1002/nme.1620372205

[38] Lorenz, E. N., “Deterministic Nonperiodic Flow,” *Journal of the Atmospheric Sciences*, Vol. 20, No. 2, 1963, pp. 130–141.
doi:10.1175/1520-0469(1963)020<0130:DNF>2.0.CO;2

[39] Hirsch, C., *Numerical Computation of Internal and External Flows: Fundamentals of Numerical Discretization*, 1st ed., Vol. 1, Wiley, New York, 1988, pp. 307–319.

[40] Poinsot, T., and Lele, S., “Boundary Conditions for Direct Simulations of Compressible Viscous Flows,” *Journal of Computational Physics*, Vol. 101, No. 1, 1992, pp. 104–129.
doi:10.1016/0021-9991(92)90046-2

P. Givi
Associate Editor