LARGE-EDDY SIMULATION USING UNSTRUCTURED GRIDS
Kenneth E. Jansen

Rensselaer Polytechnic Institute, Department of ME, AE & M, Troy, NY 12180 USA

Abstract

Large-eddy simulation (LES) has matured to the point where application to complex flows is desirable. The extension to higher Reynolds numbers leads to an impractical number of grid points with existing structured-grid methods. Furthermore, most real world flows are rather difficult to represent geometrically with structured grids. Unstructured-grid methods offer a release from both of these constraints. However, just as it took many years for structured-grid methods to be well understood and reliable tools for LES, unstructured-grid methods must be carefully studied before we can expect them to attain their full potential.

1 Introduction

In the past three years, important building blocks have been put into place making possible a careful study of LES on unstructured grids. The first building block was an efficient mesh generator which allowed the placement of points according to smooth variation of physical length scales. This length scale variation is in all three directions independently, which allows a large reduction in points when compared to structured-grid methods, which can only vary length scales in one direction at a time. The second building block was the development of a dynamic model appropriate for unstructured grids. The principle obstacle was the development of an unstructured-grid filtering operator. New filtering operators were developed in Jansen [1]. In the past year, some of these filters have been implemented into a highly parallelized finite element code based on the Galerkin/least-squares finite element method (see Jansen et al. [12] and Johan et al. [16]).

Within this paper we will describe the finite element formulation and the salient features leading to efficient parallel implementation. We also give a description of the extension of the dynamic sub-grid model to unstructured-grid simulations. After a cursory description of mesh generation techniques, results of past and current simulations will be provided to illustrate the efficiency and the accuracy of the approach. Results will be presented both from simple validation problems and from the very challenging simulation of flow over an airfoil at maximum lift.

2 The finite element formulation

There are many finite element and finite volume formulations available for application with unstructured grids. However, few have had their accuracy and stability as carefully scrutinized to the degree of what have become known as stabilized finite element methods. Two important members of this family, Galerkin/Least-Squares (GLS) and Streamline Upwind Petrov Galerkin (SUPG), have been proven stable and higher order accurate (converging at the optimal rate for a given function space) for the full range of flows, inviscid to viscous dominated, for multi-dimensional linear advective-diffusive systems (see Hughes et al. [11] and Hughes et al. [10]) and the frozen coefficient Navier-Stokes equations (see Franca and Hughes [6]). The interested reader is encouraged to look to these references for detail as the space of this paper allows only a summary of the formulation below.

Consider the compressible Navier-Stokes equations (complete with continuity and total energy equation) written in filtered form after the application of a sub-grid-scale model to the unresolved scale stress term.

\[
\mathcal{L}U = U_{i,t} + F_{i,adv} - F_{i,diff} = 0
\]  

(1)
where

\[
U = \begin{bmatrix}
U_1 \\
U_2 \\
U_3 \\
U_4 \\
U_5
\end{bmatrix} = \mathbf{T} \begin{bmatrix}
\frac{1}{\delta_{1i}} \\
\frac{1}{\delta_{2i}} \\
\frac{1}{\delta_{3i}} \\
\frac{1}{\delta_{4i}} \\
\frac{1}{\delta_{5i}}
\end{bmatrix}, \quad F_{i}^{adv} = \tilde{u}_i U + \mathbf{T} \begin{bmatrix}
0 \\
\delta_{1i} \\
\delta_{2i} \\
\delta_{3i} \\
\delta_{4i}
\end{bmatrix}, \quad F_{i}^{diff} = \begin{bmatrix}
0 \\
\tau_{1i} \\
\tau_{2i} \\
\tau_{3i} \\
\tau_{4i}
\end{bmatrix}
\]

(2)

and

\[
\tau_{ij} = (\mu + \mu_T)S_{ij}(\tilde{u}), \quad q_i = -(\kappa + \kappa_T)p_{tot} - \frac{2}{3} \delta_{ij} \tilde{u}_{k,k}, \quad \kappa_T = c_T \frac{\mu_T}{\rho_T}
\]

(3)

Here we use the overbar to denote an unweighted filter and a tilde to denote a density weighted filter.

To discretize these equations with finite elements the entire system (1) is dotted from the left by a vector of weight functions, \(W\), and integrated over the spatial domain to yield the so-called weak form or variational formulation. Integration by parts is then performed to move some of the derivatives onto the weight functions (reducing the continuity requirements). This process leads to the weak form (variational equation):

\[
0 = \int_{\Omega} \left( W \cdot U_{,t} - W \cdot F_{i}^{adv} + W \cdot F_{i}^{diff} \right) d\Omega - \int_{\Gamma} W \cdot (F_{i}^{adv} + F_{i}^{diff}) n_i d\Gamma \\
+ \sum_{s=1}^{n_{el}} \int_{\Omega^s} \mathcal{L}W \cdot \mathcal{T} \mathcal{L}U d\Omega
\]

(5)

where \(\Omega\) is the spatial domain of the problem with boundary \(\Gamma\), discretized into \(n_{el}\) elements. The first line of (5) contains the Galerkin approximation (interior and boundary) and the second line contains the least-squares stabilization. The stabilization matrix \(\mathbf{T}\) is an important ingredient in these methods and is well documented in Shakib [20].

To develop a numerical method, the weight functions, the solution variable \(U\), and its time derivative \(U_{,t}\) are expanded in terms of basis functions (typically piecewise polynomials, all calculations described herein were performed with linear basis). The integrals (5) are then evaluated using Gauss quadrature resulting in a system of non-linear ordinary differential equations which can be written as

\[
M \ddot{U} = R(\dot{U})
\]

(6)

where the check is added to make clear that \(\ddot{U}\) is the vector of solution values at discrete points (interpolated through the space by the basis functions) and \(U_{,t}\) are the time derivative values at the same points. At this point a time integrator is introduced to relate the time derivative to the solution variable yielding a non-linear matrix problem to be solved at each time step. An implicit, second order accurate family of time integrators has been developed and applied to this problem. The family is characterized by one free parameter, the amplification factor as the time step tends to infinity (\(\rho_{\infty}\), which varies from 0 (Gears Method) to 1 (Midpoint Rule). The non-linear matrix problem is solved in a predictor-corrector format yielding successive linear problems. Each linear problem is solved using the Matrix-Free Generalized Minimal RESidual (MF-GMRES) solution technique. Convergence of the non-linear problem is confirmed before moving to the next time step.

It should be pointed out that \(U\) is not necessarily the best choice for the variable to be interpolated linearly. For low Mach number flows we find the variables \((p, u, v, w, T)\) to perform best and therefore they are used in all the problems which follow. Presentation of the equations in this form would unnecessarily complicate our presentation here though, so we refer the interested reader to Hauke and Hughes [9] who presented a framework wherein any independent variable set could be used as the interpolation variable.

Finite element methods are relatively new to computational fluid dynamics. Most applications thus far have been to steady, laminar and Reynolds averaged flows. Until recently their application to unsteady turbulent flows was deemed too expensive and not sufficiently accurate. This assessment is puzzling given...
that the methods can be proven higher order accurate and that the unstructured grid can lead to point reductions which more than offset the additional work associated with the method. In fact the inclusion of a consistent mass matrix in (6) leads to a superconvergent phase accuracy in regions where the mesh is uniform or slowly varying. This superconvergence is the result of the rational approximation (Padé like) and can be illustrated by studying the growth of a small disturbance with the shape of the most unstable eigenmode of parallel channel flow. Here we repeat the flow conditions studied by Malik et al. [18] \( \left( \text{Re} = 7500, \alpha = 1 \right) \). The growth of the disturbance energy within the Navier-Stokes code (no subgrid-scale model) can be compared to linear stability theory to test the accuracy of the numerical method.

In Figure 1 we compare the growth rates of our formulation to that obtained by central differences on a staggered grid (the workhorse of the LES community). The central difference results come from a well tested code but are at a slightly higher Reynolds number \( (\text{Re} = 8000) \) and are thus compared to the exact growth rate at that Reynolds number. In both methods 32 cells were used in \( x \) and the \( y \) resolution was varied with a fixed stretching function and a variable number of points. In both methods there is an initial transient which has been removed by shifting the curves such that they are correct at one period \( (T_P) \) of the flow disturbance. Finally, the convergence rates can be seen in Figure 2. From these figures one can clearly see the superior accuracy of the finite element formulation both in terms of rate (slope) and in terms of the constant. It is worth pointing out here that the author has used the best central-difference staggered grid results available ([18] publish much worse results with decay on grids of \( N_y = 64 \)). These figures clearly challenge the notion that kinetic energy conservation (a property of central difference schemes not enjoyed by stabilized finite element methods) is paramount since, at least in this problem, accurate interaction of the disturbance with the mean flow is more critical than the very small additional dissipation generated by the stabilization operator.

### 3 Parallel implementations

The original implementation of this approach was on a Thinking Machines CM5 following the so-called data parallel approach. Finite element methods are amenable to this approach since the bulk of the computational effort lies in evaluating local integrals over each element domain. Parallelization of these operations is trivial after a gather of the data from the global nodes to an element based data structure. After the element integrations are performed in parallel, the results (residuals of (6)) are then scattered (assembled) back to the nodes where the equations are solved.

On the CM5 these gather/scatter operations are performed using CMSSL library procedures requiring no
preprocessing of the data. This made application to this platform quite easy. Unfortunately, other parallel platforms do not have libraries of this type. For this reason a second version of the code was developed, employing the Message Passing Interface (MPI) to do the gather and scatter operations. This version of the code requires that the problem be pre-processed to break the computational domain into $n$ pieces where $n$ is the number of processors, on which, the problem will be run. This preprocessing step is non-trivial for unstructured grids since load balance (equal distribution of the elements onto each processor) and minimal communication are necessary. Recently we have accomplished this pre-processing procedure and have begun testing our MPI version of the code on a variety of platforms (IBM-SP2, SGI-Origin, SUN-Sparc and Cray J90). The algorithm has shown perfect scalability on moderate sized problems (typically the bigger the better) due to the low ratio of communication to calculation that is inherent in finite element methods.

4 Dynamic model development

In section 2, the modeled form of the equations were given. To close these equations an expression for the subgrid-scale viscosity $\nu_T$ must be given. In the early nineties a dynamic procedure for determining this viscosity was developed by Germano et al. [7]. The model is based on a comparison of results of applying a subgrid-scale model (most applications use Smagorinsky) at two different grid resolutions. Usually, one grid resolution is chosen to be that of the calculation and the other is obtained by filtering the solutions obtained by the calculation. The filtering process is carried out through the convolution of the function with a filtering kernel $G$ viz.

$$f(x) = \frac{\int G(x, x') f(x') d\Omega}{\int G(x, x') d\Omega}$$  \hspace{1cm} (7)

Applications of this model flourished within the spectral methods community and the finite difference community where it gained wide acceptance. Use of this model on unstructured grids required an extension of the filtering operator. A variety of filtering operators were developed in [13]. To date the most successful one in applications has been the simplest one, the generalized tophat filter. In this approach the filter function for each nodal value is taken to be one on the domain formed by elements touching that node. A 2D visualization of the concept is included in Figure 3.

5 Model validation

To verify that both the numerical method and the subgrid-scale model are appropriate for large eddy simulation two canonical flows have been simulated; decay of isotropic turbulence and turbulent channel flow.
Figure 3: Domain of filter for structured and unstructured grids. In each case the filter $G$ takes on a value of 1 on the domain created by the elements which touch the hollow node and zero elsewhere. In 3D the structured grid filter domain becomes a hexahedron whereas the unstructured grid filter domain approximates an ellipsoid.

Figure 4: Decay of isotropic turbulence on a $32^3$ grid compared to the experimental data of [4]. On the left is the decay of resolved kinetic turbulent energy; on the right the filtered velocity spectra where the wavenumber ranges from 1 to 16 (normalization by $2\pi/L$, where $L$ is the size of the computational box).

--- Present dynamic model; ---- fixed coefficient Smagorinsky model ($C_s = 0.17$); ······· no model. ●, ●, ■ Experiment at $t = 42, 98$ and 171 respectively.

In each case we have not done a careful refinement study as would be appropriate for a physical study of these flows. Instead we have used very course grids to get an indication of the models performance with the resolution expected to be attainable in the more complex flows for which these methods are directed. It is worth pointing out that these flows do not have adequate length scale variation to give unstructured grids the point reduction advantage that we see in more complex flows.

5.1 Decay of isotropic turbulence

A common validation test for large-eddy simulation codes is the isotropic homogeneous turbulence decay. The simulation was run on a $32^3$ periodic grid and is designed to replicate the results of the experiment conducted by Comte-Bellot & Corrsin [4]. To accurately match the experiment, the initial velocity field of the simulation must match the filtered spectrum given at the first experimental station, corresponding to $t = 42$ with the conventions of the reference. Then, the actual decay is simulated and results are compared to the experimental data given at $t = 98$ and 171. During this computation, the time step is constant with 45 time steps per eddy turnover time at the first experimental station ($t = 42$).

Figure 4 shows the decay of the kinetic turbulent energy obtained with the dynamic model, with the Smagorinsky coefficient fixed at $C_s = 0.17$, or when no subgrid-scale model is used. The variations observed are important since they show that the numerical dissipation does not dominate the subgrid-scale model. Though the fixed-coefficient Smagorinsky model seems to provide even better results, the velocity spectra, presented in on the right in Figure 4, show that this would be a an erratic conclusion since the dynamic
model represents the shape of the spectrum much better while the fixed coefficient simply benefits from error cancellation. Finally the spectrum obtained with no model is flatter, which indicates that the numerical dissipation inherent to the scheme is not dominant.

5.2 Channel flow

The Reynolds number 180 channel flow was studied using direct numerical simulation (DNS) by Kim et al. [17]. We have performed a large-eddy simulation of the same flow conditions but with a course grid of $32 \times 64 \times 32$ elements. In Figure 5 we show the stress balance attained from averaging the flow over 10 time units (one time unit is the time it takes a particle to flow though the channel along the centerline). We have also included the rms of the velocity components in Figure 5. Here the results are compared to a 2nd order central difference scheme. Note the the improvement in all rms quantities, especially the $v_{rms}$ and $w_{rms}$ components.

6 Airfoil simulation

As mentioned earlier, the real promise of this method is its application to more complex flows where there is a large variation in length scales requiring resolution. Here, unstructured grids can refine or coarsen to match the physical length scale requirements of a particular region rather than letting the length scale of one region determine the grid requirements for the entire problem. While the same can be done, to some extent by overset or Chimera grids, for many flows there is a gradual and continuous change in length scales making this approach impractical.

We have chosen the NACA 4412 airfoil at maximum lift as the demonstration simulation for this approach for a variety of reasons. First, it is a problem of significant interest since it would be the first LES of an aircraft component. Second, this flow has been the subject of three experimental studies (Coles and Wadcock [3], Hastings and Williams [8], and Wadcock [21]). The first study found the maximum lift angle to be $13.87^\circ$. The later studies found the angle to be $12^\circ$. Wadcock reports in the later study that the early data agree very well with his new data at $12^\circ$, suggesting that the early experiment suffered from a non-parallel mean flow in the Caltech wind tunnel. It should be pointed out that the Reynolds-averaged simulations are usually run at $13.87^\circ$ and do not agree with the data when run at $12^\circ$. It is hoped that LES can clarify this controversy. The third reason for considering this flow is the variety of flow features which provide an important test of the dynamic model. Starting from the nose where the flow stagnates, thin laminar boundary layers are formed in a very favorable pressure gradient. This pressure gradient soon turns adverse, driving the flow toward a leading edge separation. Only the onset of turbulence can cause the flow to remain attached or to
reattach if it did separate. The persistent adverse pressure gradient eventually drives the turbulent flow to separate in the last 20 percent of chord. The separation bubble is closed near the trailing edge as the retarded upper surface boundary layer interacts with the very thin lower surface boundary layer. The large difference in boundary layers creates a challenging wake to simulate. Only the dynamic model can be expected to perform satisfactorily in this variety of situations: from the laminar regions where it must not modify the flow at all to the turbulent boundary layers and wake where it must represent a wide variety of subgrid-scale structures.

The flow configuration we have chosen is that of Wadcock [21] at Reynolds number based on chord $Re_c = u_{\infty} c / \nu = 1.64 \times 10^6$, Mach number $M = 0.2$, and $12^\circ$ angle of attack.

### 6.1 Mesh design

The design of the mesh followed the suggestion of Moin and Jiménez [19] and Chapman [2] with the spanwise spacing set at 50 wall units and the streamwise spacing set at 200 wall units in the near wall region. With the experimental values of $C_f$ available these spacings have been plotted as a function of streamwise position in Figure 6. Away from the wall the eddies no longer scale on wall units but instead scale on boundary layer thickness. A third curve has been included on Figure 6 to illustrate how this spacing varies with streamwise position based on Wadcock’s measured boundary layer thickness.

Several points can be made in this figure. First, all three curves change by over an order of magnitude from the tip to the tail region. This illustrates how an unstructured grid saves points by matching resolution to the local changes in the length scales in the streamwise direction. For example, a structured grid would be forced to carry the fine spanwise resolution required near the nose throughout the entire domain. Secondly, when comparing the near-wall spanwise resolution to the outer-layer resolution, it is clear that coarsening the spanwise resolution as the distance from the wall increases is justified. The final point, apparent from this figure, is that coarsening of the streamwise resolution in the outer layer is not justified. In fact, over much of the airfoil surface the outer-layer grid resolution is more restrictive than the inner-layer resolution. The choices of 200 wall units and 5 points per boundary layer thickness are somewhat arbitrary, but they are believed to be comparable in their degree of coarseness. It is interesting to observe that the crossover between these two curves corresponds to $Re_c = (u_{\tau} \delta_{99} / \nu) = 1000$. Therefore, when above 1000, the inner-layer resolution is the most restrictive. Otherwise, the outer-layer resolution is the most restrictive. Only at higher Reynolds numbers will coarsening in the streamwise direction be justified.

Early on it became apparent that existing mesh generators could not produce grids to match the above spacing requirements in a fashion smooth enough to retain accuracy. A significant amount of effort was
expended writing a custom mesh generator for this effort. Figure 7 shows the fruit of this labor. This, rather complex figure, shows a cut through the airfoil cross section connected to the surface mesh. The surface mesh appears nearly black because of the very fine near-wall resolution. The additional surfaces parallel to the surface mesh are offset so that the coarsening of the grid with increasing distance from the wall can be observed. The final grid is outside of the boundary layer and nearly all of the spanwise resolution has been shed, since it is not needed to compute a two-dimensional inviscid region. It is worth noting here that the equivalent structured grid, which would carry the finest z spacing and the given x spacing out to the boundary would require 27 times as many points.

6.2 Airfoil simulations

The first simulations of this problem assumed free transition and a far-field boundary condition to approximate real flight conditions. Despite grid refinement studies, described in Jansen [14], it became clear that the flow was not converging to the experimental results, especially the velocity profiles. Figure 8 offers some explanation of the lack of convergence. Here we compare to two experimental data sets from Hastings. Note the good agreement with the pressure data taken with the transition strip removed. However, all the remaining experimental data (velocity profiles, boundary layer parameters, and rms profiles) were taken with the transition strip in place, since they wished their study to be representative of higher Reynolds number flows. Though the shift in pressure data may appear small it seems to have a profound effect on the velocity profiles, since the flat region near the trailing edge signals a massive separation that is much milder in the untripped case.

This finding motivated a second study of this flow where the wind tunnel walls and the transition strip were accounted for in the calculation. Hasting’s transition strip was composed of Ballotini sprinkled randomly over the leading edge region of the flow. This was deemed more difficult to simulate than Wadcock’s trip. Wadcock used a piece of tape with serrations cut into the windward side. A mesh was constructed to represent the exact, position, height and shape of this tape which can be seen in Figure 9. The results of these simulations as well as the mesh used to model the wind tunnel walls can be found in Jansen [15]. The results were initially encouraging in the sense that the flow began to separate more after the inclusion of wind tunnel walls and the transition strip. With the larger separation came a thicker boundary layer causing growing concern about the width of the computational domain. As is common in large eddy simulation of statistically two-dimensional flows periodicity is assumed in the spanwise direction. For this to be valid requires that the spanwise extent be large enough that a two point correlation in the spanwise direction decays sufficiently to signal the flow decorrelated at the spanwise boundaries. Over most of the airfoil surface this is satisfied but as the boundary layer thickens near the trailing edge it is violated. In the course of our refinement studies we had narrowed the domain to \( z_{max} = 0.025c \). Under the new boundary conditions the boundary layer exceeded this dimension in the last 25% of the the airfoil surface. The effect of this narrow domain seemed to be a promotion of spanwise coherent structures which, though certainly making the separation larger, were feared to be unphysical.

Therefore, two additional calculations were undertaken. The first used the same number of points as the previous calculation but doubled the spanwise domain by doubling the size of each element’s spanwise dimension. The advantage to this calculation was that since it contained the same number of points, answers could be obtained in a comparable time frame. This simulation improved the narrow domain problem but at a cost of spanwise resolution. Therefore, the second additional calculation was initiated wherein the spanwise resolution remained the same while the domain was doubled. Since the number of degrees of freedom have nearly doubled, this calculation is quite large (approximately 8 million tetrahedra).

To date, only the first of these widened-domain simulations has run long enough to obtain converged statistics. In Figure 10 the velocity profiles from the two simulations which include the effect of the wind-tunnel walls (one with the narrow domain, one with doubled spanwise spacing and domain width) are compared to the earlier simulation which did not account for the wind tunnel walls or the transition strip. As mentioned earlier, accounting for the experiment’s boundary conditions has led to greater separation in the trailing edge region. However, the first widened-domain results suggests that, indeed, some of that improvement was due to the artificially narrow domain. This conjecture cannot be confirmed though until the fine-grid, wide-domain results reach a converged state, allowing comparison with these results.
Figure 7: Grid refinement in layers moving away from the wall.

Figure 8: Coefficient of pressure on the airfoil surface comparing calculation (—) to Hastig’s experiment with transition strip (○) and without transition strip (+).
Figure 9: Geometric model of the exact shape, height and position (shown in black on the airfoil surface) of Wadcock’s transition strip.

Figure 10: Profiles of tangential velocity component at various positions along the airfoil surface ($x/c = 0.59, 0.66, 0.78, 0.82, 0.95$, plots have been shifted by 1.5 at each station). Solutions correspond to: without wind tunnel walls or transition strip — , with wind tunnel walls and transition strip ($W/c = 0.025$) —— , with wind tunnel walls and transition strip ($W/c = 0.05$) ——— , Wadcock ○ , Hastings and Williams ++ .
7 Conclusions and future directions

While much progress has been made, the NACA 4412 airfoil simulation has still not yielded complete agreement between LES and experiment. Along the way it has served as a vehicle for the development and severe testing of both the accuracy and efficiency of our numerical method. To this end we have begun development of new, higher-order methods which employ hierarchical basis functions to add p adaptivity to the already present h adaptivity. These basis functions have already been tested on canonical problems (Helmholtz, Poisson, and linear elasticity; see Dey [5]). Their implementation into the Navier-Stokes code is currently being tested on simple flows. Their application to turbulent flows is expected shortly.

Other simulations are also being carried out with this code though the space remaining here allows only a brief mention of them. Bastin [1] is using the code to compute the noise from a supersonic jet ($M = 1.4$, $Re_D = 1.5 \times 10^6$). We have also recently started a simulation of turbulent flow over a cavity ($Re_H = 27000$, $Re_{\theta} = 5100$) with the goal of predicting/altering the acoustic resonance which sometimes occurs in this geometry.

8 Acknowledgments

The author wishes to acknowledge the collaboration of Francois Bastin on the decay of isotropic turbulence problem and on the development of the MPI version of the code, Farzin Shakib on the channel flow problem, S. Scott Collis, Juin Kim, and Christian Whiting on the small disturbance problem, and many helpful suggestions by Parviz Moin, Thomas Lund, Hans Kaltenbach, Rajat Mittal, and Massimiliano Fatica and others at the Center for Turbulence Research (CTR) on the airfoil simulation.

This work has been supported by the AFOSR and by CTR (a joint NASA-Stanford University program). It has also enjoyed computer resources from the Army High Performance Computing Research Center (AH-PCRC) and the Naval Research Lab (NRL).

References


