Towards Petascale Computing for Realistic Jet Noise Simulations

KURT M. AIKENS  
Purdue University  
Sch. of Aero. and Astro.  
701 W Stadium Ave  
W. Lafayette, IN, USA  
kaikens@purdue.edu

NITIN S. DHAMANKAR  
Purdue University  
Sch. of Aero. and Astro.  
701 W Stadium Ave  
W. Lafayette, IN, USA  
dhamank@purdue.edu

YINGCHONG SITU  
Purdue University  
Dept. of Comp. Sci.  
305 N University St  
W. Lafayette, IN, USA  
ysitu@cs.purdue.edu

YE WANG  
Purdue University  
Dept. of Comp. Sci.  
305 N University St  
W. Lafayette, IN, USA  
wang1351@purdue.edu

CHANDRA S. MARTHA  
Purdue University  
Sch. of Aero. and Astro.  
701 W Stadium Ave  
W. Lafayette, IN, USA  
cmart@purdue.edu

ZHIYUAN LI  
Purdue University  
Dept. of Comp. Sci.  
305 N University St  
W. Lafayette, IN, USA  
li@purdue.edu

GREGORY A. BLAISDELL  
Purdue University  
Sch. of Aero. and Astro.  
701 W Stadium Ave  
W. Lafayette, IN, USA  
blaisdel@purdue.edu

ANASTASIOS S. LYRINTZIS  
Embry–Riddle Astro. Univ.  
Dept. of Aero. Eng.  
600 S Clyde Morris Blvd  
Daytona Beach, FL, USA  
lyrintzi@erau.edu

Abstract: Jet noise is an important issue due to increased commercial air traffic, penalty fees for noisier aircraft, and stringent noise regulations as well as military operational requirements. Advanced computational techniques based on large eddy simulation (LES) have been used to study jet noise. We have previously developed an in-house LES application for simulating subsonic jets without the nozzle. This application is designed to feature supreme scalability, which has enabled large-scale experiments utilizing up to 91,125 processor cores. In this paper, we describe recent advancements of our methodology that allow simulations including nozzles and of jets at supersonic conditions with shock waves. We also discuss future directions in exploring hybrid and heterogeneous parallelism.

Keywords: computational fluid dynamics, aeroacoustics, aircraft noise

1 Introduction

Aviation has assumed a critical role in supporting global economic growth. Aircraft noise, a byproduct of aviation, however, is proving to have a tangible impact on the overall benefit of aviation, ranging from physical damage to the human body to financial penalties imposed on its originators and costs of noise mitigation measures. In recent years, designing quieter aircraft has become a new arena of competition for major aircraft manufacturers. Low-noise in-flight experience is gaining emphasis in advertising campaigns targeting ordinary customers. In military scenarios, injury-incurring sound levels near vertical and/or short take-off and landing (V/STOL) aircraft also corroborates the necessity of noise reduction in aircraft design.

An important design tool of modern reduced-noise aircraft is computational aeroacoustics (CAA) simulation of sound levels generated by aircraft airframes and engines. CAA, a relatively new discipline of computational fluid dynamics (CFD), is intended as a robust and accurate technique that complements traditional theoretical and empirical approaches in aircraft noise prediction. Different from practices in general CFD, CAA strongly relies on accurate prediction of small-amplitude acoustic fluctuations and their correct propagation to the far field. This demands that the underlying numerical methods provide high accuracy and good spectral resolution, while keeping diffusion and dispersion errors low.

The state-of-the-art CAA prediction of far-field noise is based on time-dependent CFD simulation of turbulent flows followed by integral postprocessing [18] that propagates the noise to the far field. This two-step approach first appeared in the 1990s and has continued to mature, partly owing to the emergence of large-scale computing platforms, which have turned computationally intensive techniques including direct numerical simulation (DNS) and LES into reality. Reference [1] is a recent review on this subject. We summarize recent high-fidelity simulations below.

For subsonic jets, references [35–37] utilize a multiblock solver with overset grid blocks to simulate nozzles with and without chevrons on grids up to 500 million points. They have also performed studies of inflow conditions upstream of the nozzle. In general, their results predict the acoustic high frequencies well.
but underpredict the lower ones. The specific cause of the underprediction is unknown. References [4,5] have studied the effect of important parameters on the noise. Using high-order methods on meshes with 252 million points, they have found that as the Reynolds number and boundary layer turbulence intensities increase, the jet potential core is lengthened, peak turbulence intensities are reduced, and far-field sound pressure levels decrease. This indicates that it is crucial to predict accurate turbulent quantities at the nozzle exit and perform simulations at realistic Reynolds numbers.

There are comparatively fewer high-fidelity supersonic jet simulations. While structured solvers for LES have been dominant, there has been a push towards unstructured solvers, which allow for more flexibility in meshing and modeling complex geometries but may also reduce accuracy. Utilizing this approach, [21] includes round converging-diverging nozzles to analyze heated and unheated jets at near-perfect expansion. Other applications include large simulations of hot overexpanded round jets issuing from realistic converging-diverging nozzles with chevrons [6] and underexpanded isothermal rectangular jets from nozzles with and without chevrons [23]. These simulations utilize hundreds of millions of grid points and scale to as many as 163,840 processors on the Intrepid cluster at the Argonne National Laboratory. Additional details on their numerical methods can be found in [10].

The above highlight some of the various LES techniques in use for jet noise studies. A survey of the literature finds that fine grids, high-order numerical methods, and accurate turbulent nozzle boundary layers are required for accurate quantitative sound level predictions. That said, there is much room for improvement and many challenges remain. Examples include simulation of flows at realistically high Reynolds numbers and inclusion of realistic nozzle geometries in high-order structured solvers.

We have recently developed a highly-parallel LES application [19,20] which aims to address these concerns and more accurately predict noise levels at the far field. Our LES application is designed to feature scalability across a wide range of processor core counts and has successfully carried out performance experiments utilizing between 2,744 and 91,125 cores. Previously, its capabilities were limited to simulations without nozzle geometries in subsonic conditions [20]. Recently, we have extended its functionalities for inclusion of nozzle geometries and simulation of supersonic jets with shock waves. We have also explored various opportunities in further improving its performance including hybrid and heterogeneous parallelism. In this paper, we describe the aforementioned efforts in more detail.

2 Methodology of LES

In LES, the effect of the large eddies is simulated directly, whereas that of the small eddies is modeled using a subgrid-scale (SGS) model. A variety of SGS models, such as the Smagorinsky [29] and dynamic Smagorinsky [9] models, have been employed in LES of jets. Alternatively, one can use a filter which is part of the numerical method [2,32] or rely on the dissipation inherent to an upwind biased numerical scheme [25,26] to provide an implicit SGS model. While unstructured solvers are frequently used for CFD because of their ability to easily handle complex geometries, the accuracy requirements for CAA often call for higher-order numerical schemes. This is due to the small amplitude of the acoustic fluctuations and the wide range of turbulent length scales that must be accurately resolved for many CAA problems. Structured solvers are ideal then, due to their available high-order methodologies and efficiency.

In our LES application, the filtered conservative-form Navier–Stokes equations are solved in generalized curvilinear coordinates on a uniform grid after spatial transformation. The classical fourth-order Runge–Kutta method is used for time integration, and a surface integral method based on the porous surface flowcs Williams–Hawkings equation [18] is used for evaluating acoustic signals. Spatial partial derivatives appearing in the governing equations are discretized using the following sixth-order compact finite difference scheme from [14]:

$$\frac{1}{3} f_{i+1} + \frac{1}{3} f_{i+1} - \frac{1}{3} f_{i-1} = \frac{7}{9h} (f_{i+1} - f_{i-1}) + \frac{1}{36h} (f_{i+2} - f_{i-2}) \quad (1)$$

where $f_i$ and $f_i'$ are the values of a function $f$ and its approximate first derivative at the $i$th point, and $h$ is the grid spacing. For points in the vicinity of the boundaries, lower-order and one-sided implicit difference formulae are used to ensure an overall tridiagonal formulation [14]. For nonlinear problems with non-periodic boundary conditions and nonuniform grids, the above compact difference scheme is unstable. As a remedy, we use the following sixth-order compact spatial filter from [38] to attenuate the high-wavenumber
modes:
\[
\alpha_f \bar{f}_{i-1} + \bar{f}_i + \alpha_f \bar{f}_{i+1} = \sum_{n=0}^{3} \frac{a_n}{2} (f_{i-n} + f_{i+n})
\]
where \( \bar{f}_i \) is the filtered value of \( f \) at the \( i \)th point, \( \alpha_f \) is a filter parameter satisfying \( |\alpha_f| < 0.5 \), and \( a_0, a_1, a_2, a_3 \) are constants depending on \( \alpha_f \). As with the compact difference scheme, lower-order and one-sided implicit formulæ are used for points in the vicinity of the boundaries.

A fast and accurate solver for the diagonally dominant tridiagonal linear systems induced by Equations 1 and 2 is crucial to our LES application. To this end, the transposition [31] and Schur complement methods [15] are known to have severe scalability limitations [20]. Better in this regard are some schemes that have overlapping blocks [33, 41] and constrain the linear systems to individual blocks. This is trivial to parallelize but inherently less accurate. For both high accuracy and scalability, our LES application uses the truncated SPIKE algorithm [24, 27] on non-overlapping blocks to solve these tridiagonal linear systems with theoretically optimal weak-scaling scalability [28]. Reference [20] reports a strong scaling test case with a 1,260 × 1,260 × 1,260 grid where our LES application achieves a speedup of 24.6 for an efficiency of 74% when the number of processor cores increases from 2,744 to 91,125.

For natural simulation of jets, advanced support for cylindrical grids has been included in our LES application. Various approaches [3, 22, 34] exist to handle the centerline singularity and avoid the time step penalty associated with the inherent concentrated grid point distribution. We have implemented three approaches: a point skipping method [3], the spectral method, and the windowed sinc filter [19]. The point skipping method produces good results [19], but our implementation uses the transposition method and thus inherits its scalability weaknesses. The sinc filter is more scalable, but some discrepancies have been noticed in the results. This is an area of future research.

3 Recent code implementations

3.1 Inlet and wall boundary conditions

To enable the inclusion of nozzle walls in jet simulations, several boundary conditions are implemented in our LES application. First, an adiabatic viscous wall formulation based on characteristic analysis is added [11, 12]. While it works well, its formulation incurs extra computational cost because modifications to individual terms in the inviscid flux derivatives require additional derivative computations. As a result, an extrapolation-based method [16] is added which reduces derivative computations by 33% compared to the characteristic boundary conditions. With this method, the continuity equation is advanced in time to compute the density on the wall, taking advantage of the no-slip velocity condition. The momentum and energy equations, however, are not time-advanced at the wall. Either an isothermal or adiabatic wall is assumed for calculating the pressure. In test cases of viscous flows over a cylinder, the two formulations are found to agree very well. The extrapolation-based boundary conditions are preferred for their efficiency.

In addition to wall boundary conditions, when including the nozzle geometry it is crucial to achieve a fully turbulent boundary layer on the nozzle wall to better match realistic operating conditions with high Reynolds numbers. A realistic turbulent boundary layer is required to accurately feed the turbulent shear layer downstream of the nozzle exit which directly influences the noise. Thus, a digital filter-based turbulent inflow boundary condition [13, 30, 39] is implemented. Furthermore, the method is extended to non-uniform curvilinear coordinates in a novel way which allows limited variation in the specification of integral length scales over the inlet plane [7].

This method has been tested on a spatially evolving zero pressure gradient flat plate turbulent boundary layer in a Reynolds number range of \( 1,410 \leq Re \leq 2,200 \) based on freestream velocity and momentum thickness. Using the current method, the turbulent stresses recover completely in a distance of about 11.5 boundary layer thicknesses. The velocity profile is also in reasonable agreement with the law of wall, indicating that the method is capable of producing sustained turbulent fluctuations. Figure 1 shows that the turbulence evolves into qualitatively realistic vortex structures in the nozzle boundary layer. The method has since been used successfully in jet simulations.

3.2 Characteristic filters for supersonic jets

To extend the implementation to include supersonic simulation capabilities, it is necessary to add shock capturing routines for simulation of off-design jets with shock waves. Based on the success of characteristic filters [8, 40] in our legacy code [17], they are ported to our current LES application. Characteristic filters
are based on the idea that discontinuous flow predictions with baseline methodologies can be corrected by adding the dissipative portion of a nonlinear scheme (e.g., total variation diminishing or weighted essentially non-oscillatory schemes) after a full time step. As a result, they are general and efficient since they can be coupled with any nondissipative base numerical scheme and do not need to be applied at each substage of the Runge-Kutta method. In our application, they are made even more efficient and less dissipative by applying them locally to shock wave regions only [17].

The shock capturing modules have been validated using simulations without nozzles. Turbulent statistics and acoustics results match well those from the legacy code. Additionally, a centerline treatment module has been developed to reduce CFL-type time step restrictions for cylindrical grids. Analogous to the point-skipping method from [3], the grid is artificially coarsened by adjusting the amount of applied dissipation along rings near the centerline. Supersonic jet acoustics simulations including the nozzle have not yet been completed, but cases with short downstream domains (e.g., 3 jet radii) have been simulated to ensure that the implementation is correct. Figure 2 shows instantaneous contours of velocity magnitude inside a military-style nozzle. Utilizing the turbulent inflow boundary conditions, preliminary results for converging-diverging nozzles are promising.

4 Future directions

As processor technologies continue to evolve, computation is outpacing communication in growth of efficiency by a widening margin on current large-scale computing platforms. Specific to our LES application, we previously measured on the Kraken cluster at the National Institute for Computational Sciences (NICS) that communication took approximately a third of the total running time. Our recent experiments on the Carter cluster at the Rosen Center for Advanced Computing (RCAC) of Purdue University show that that proportion has risen to about 50%. One possible solution is to adopt multiple forms of parallelism.

Currently, our LES application is parallelized using the Message Passing Interface (MPI) and thus exploits the single-program multiple-data (SPMD) parallelism. As we increase the number of MPI processes, the block size decreases, resulting in reduced communication efficiency as [28] shows. To alleviate such an issue, we are investigating running each MPI process on multiple processor cores using OpenMP as the means of intraprocess parallelization. This allows each process to handle larger blocks and enables more flexible load balancing when heterogeneous parallelism is incorporated. Our initial experiments have shown that an MPI–OpenMP hybrid implementation can be up to 3% faster than the MPI-only version. We plan to continue performance tuning for current symmetric multiprocessing (SMP) clusters and prepare for porting the code to Intel Xeon Phi coprocessors.

Using graphics processing units (GPUs) is another option for speeding up our LES application. We are currently in the process of experimentally porting the application to NVIDIA GPUs and have finished all truncated SPIKE-related code. Instead of simply offloading workload to the GPU, our strategy is GPU-centric, with the CPU used for control and communi-
cation only and otherwise left as additional computing power. We have achieved a speedup of 1.5 to 2.5 over the MPI–OpenMP version in computation but have seen longer times in communication. The main cause is that in our current implementation, GPU-to-GPU communication has to go through the main memory. We are investigating taking advantage of hardware support for direct GPU-to-GPU communication via the PCI Express bus and remote direct memory access (RDMA) over InfiniBand to reduce communication time and continue to port the entire application to GPUs.

We envision that in the long term, our LES application will combine both hybrid parallelism and accelerator-based heterogeneous parallelism.

Acknowledgments: This study is primarily supported by the National Science Foundation (NSF) via award CCF-0811587, with additional support from award OCI-0904675. Experiments of this study have used the Kraken cluster hosted by the NICS under Extreme Science and Engineering Discovery Environment (XSEDE) allocation TG-ASC040044N and the Carter cluster of the Rosen Center for Advanced Computing at Purdue University. XSEDE is supported by NSF award OCI-1053575.

References:


