Development of a Parallel 3-D LES Methodology for Jet Aeroacoustics via the Schur Complement

P. Lew,* A. Uzun† G. A. Blaisdell‡ and A. S. Lyrintzis§
School of Aeronautics and Astronautics
Purdue University
West Lafayette, IN 47907

Improvements in computing speed over the past decade have made Large Eddy Simulation (LES) an attractive tool to study jet noise. In this regard, high-order compact finite difference schemes along with high-order filters are used extensively in LES, especially for aeroacoustics problems, since these schemes not only have very high accuracy and spectral-like resolution, but also low-dispersion and diffusion errors. Due to the implicit nature of compact schemes, one technique of parallelization is based on the data transposition strategy. However, the computational resources that are spent on the communication between processors can be as high as 40%. This paper proposes an alternative parallelization methodology based on the Schur complement technique to address the substantial communication overhead of the transposition strategy. Computed results for 1-D and 2-D Linearized Euler Equation (LEE) test cases compare well to the corresponding exact solutions. The 3-D LES test case via the Schur complement shows on average a 3% and 1.5% difference in turbulent intensity and far-field acoustics, respectively, compared to results of a previously simulated single-block methodology using data transposition. However, the parallel 3-D Schur complement case is roughly 2 times slower compared to the single-block case. The relatively poor performance is mainly due to the one dimensional partitioning applied. A three dimensional decomposition is expected to remedy the poor performance and will be carried out in the future.

Nomenclature

\begin{align*}
a_j & \quad \text{Jet centerline speed of sound} \\
D_j & \quad \text{Jet diameter} \\
f & \quad \text{Frequency} \\
M_j & \quad \text{Mach number} = \frac{U_j}{a_j} \\
N_x & \quad \text{Number of grid points in the } x \text{ direction} \\
N_y & \quad \text{Number of grid points in the } y \text{ direction} \\
N_z & \quad \text{Number of grid points in the } z \text{ direction} \\
N_{bx} & \quad \text{Number of blocks in the } x \text{ direction} \\
N_{by} & \quad \text{Number of blocks in the } y \text{ direction} \\
N_{bz} & \quad \text{Number of blocks in the } z \text{ direction} \\
N_{procs} & \quad \text{Total number of processors} \\
r_o & \quad \text{Jet radius at inflow plane} \\
R & \quad \text{Radial arc length from centerline jet exit} \\
Re_D & \quad \text{Jet Reynolds number} = \frac{\rho_j U_j D_j}{\mu_j}
\end{align*}

*Graduate Research Assistant, Student Member AIAA.
†Post-Doctoral Research Associate, Florida State University, Tallahassee, FL, Senior Member AIAA.
‡Associate Professor, Senior Member AIAA.
§Professor, Associate Fellow AIAA.

Copyright © 2006 by P. Lew, A. Uzun, G. A. Blaisdell and A. S. Lyrintzis.
I. Introduction

There is a current need to better understand the noise generation mechanisms in turbulent subsonic jets more than ever. This is because during the past several years, airports locally and abroad have implemented strict regulations on aircraft with high jet noise emissions. These high jet noise levels are undesirable to the communities surrounding the airport. Hence, a goal was introduced by NASA in 1997 aimed at eliminating community noise problems near airports. The goal is to reduce the perceived noise levels of future aircraft by a factor of two (10 EPNdB) from subsonic aircraft by 2007, and by a factor of four (20 EPNdB) by 2022.1 This goal is not infeasible, but it is challenging nonetheless due to the fact that the underlying mechanisms/sources that cause jet noise are still not well understood and, therefore, cannot be fully controlled or optimized. Thus, the jet noise problem still remains one of the most elusive problems in aeroacoustics.

With the advent of fast parallel supercomputers, the application of state-of-the-art computational methods such as Direct Numerical Simulation (DNS)2–3 and Large Eddy Simulation (LES)4–12 to jet noise prediction is becoming more feasible. In the context of Computational Aeroacoustics (CAA), high-order numerical methods are needed to capture as much as possible the turbulent flow physics that inherently leads to sound generation for a jet. A popular numerical technique currently implemented in many DNS and LES solvers for CAA are implicit high-order compact finite-difference and filtering schemes.13–15 These schemes, while retaining the simplicity of low-order methods (i.e. small stencil), have high accuracy, good spectral-like resolution and low dispersion and diffusion errors.

As an example, consider the non-dissipative sixth-order compact scheme proposed by Lele13 to evaluate the spatial derivatives at the interior grid points

\[
\frac{1}{3}f_i^{+1} - f_i^{-1} + \frac{1}{3}f_i^{+1} = \frac{7}{9\Delta \xi} (f_{i+1} - f_{i-1}) + \frac{1}{36\Delta \xi} (f_{i+2} - f_{i-2}).
\]  

(1)

Here, \( f_i^{+1} \) is the approximation of the first derivative of \( f \) at point \( i \) in the \( \xi \) direction, and \( \Delta \xi \) is the grid spacing in the \( \xi \) direction which is uniform. In conjunction with the spatial derivative shown in Eq. (1), spatial filtering is used to eliminate numerical instabilities that can arise from the boundary conditions, unresolved scales and mesh non-uniformities.\(^{16,17}\) Eq. (2) shows the sixth-order tri-diagonal spatial filter proposed by Visbal and Gaitonde\(^{15}\) applied to the interior grid points

\[
\alpha_f f_{i-1} + f_i + \alpha_f f_{i+1} = \sum_{n=0}^{3} a_n (f_{i+n} + f_{i-n}),
\]

(2)

where \( \alpha_f \) is the filtering parameter governed by \(-0.5 < \alpha_f < 0.5\), and the \( a_n \) coefficients can be found in Visbal and Gaitonde’s report.\(^{15}\) Note that the tri-diagonal filtering scheme used here has a 7-point stencil.

The compact schemes shown above are implicit and require a solution of a linear system of equations along grid lines. Hence, care must be taken when crafting parallel methodologies along the interfaces when one decomposes the domain of the computational grid. One such method of parallelizing a solver that implements compact schemes is the transposition strategy.\(^{18}\) For example, assume that our computational domain has the shape shown in Figure 1. The computational grid is first partitioned into non-overlapping single blocks along the \( z \) direction, i.e. the top diagram of Figure 1. Then the spatial derivatives and filtering are computed.

---

American Institute of Aeronautics and Astronautics Paper 2006-0796
in the x and y directions independently from all other blocks. In order to apply the compact scheme and filter along the z direction, a data transposition is performed to re-align the blocks in the y direction (bottom diagram of Figure 1). Then the processors can apply the compact and filtering schemes along the z direction. When the computations along the z direction are complete, another transposition is applied back to the initial configuration to send the newly computed information back to the original configuration for further computations. Hence, the transposition strategy yields the best approximation when utilizing the implicit compact and filtering schemes since it uses every grid point along a given direction. However, the transposition scheme suffers from two setbacks. Firstly, it can only be used for simple geometries or single-block cases and applying the transposition technique to complex geometries is very difficult. Secondly, due to the transposition process, large amounts of data have to be sent between processors resulting in a large communication overhead. For a typical run using the transposition strategy, the computational resources that are spent on the communication between processors can be as high as 40% of the simulation process.\textsuperscript{19}

To address the transposition strategy’s shortcoming regarding complex geometries, a multiblock approach can be used.\textsuperscript{20,21} In a multiblock approach, a complex domain is divided into smaller, more manageable domains and the high-order compact differencing and filtering schemes (shown earlier) are applied in each block. One strategy is to use grid point overlaps, while maintaining the sixth order differencing across boundaries. In addition, grid point overlaps are needed to exchange information between neighboring blocks during the simulation.\textsuperscript{21} However, up to 30% of the total number of grid points of the computational domain could consist of grid point overlaps for a large (more than one hundred) number of blocks. Alternatively, single-sided compact differencing schemes can be applied near and at the boundaries of each block.\textsuperscript{22} This methodology undoubtedly is not as accurate compared to applying a centered compact differencing or filtering scheme as in the interior of the domain and thus might introduce small errors in the flow field. Hence, a more robust and efficient parallelization methodology is required to overcome the substantial communication overhead and grid augmentation due to overlaps, while maintaining the desirable characteristics of the compact differencing and filtering schemes throughout the solution domain.

Figure 1. Schematic of the transposition scheme.

With that in mind, our aim in this paper is to propose an alternate parallelization methodology for implicit compact differencing and filtering schemes, with particular emphasis on 3-D jet aeroacoustics based on the Schur complement.\textsuperscript{23,24} Hence, this paper is arranged as follows: Section II provides a brief background on the parallel Schur complement used in this study; in Section III, the Schur complement is applied to a 1-D and 2-D Linearized Euler Equation (LEE) test case and the results are compared to a corresponding exact solution. We then apply the parallel Schur complement to a full 3-D LES of a jet in Section IV. Finally, some concluding remarks and future work in this report are given in Section V.
II. Parallel Schur Complement

The Schur complement method which is also known as the sub-structuring technique has been widely used in structural mechanics to solve large-scale systems with limited memory computers for over 30 years.\textsuperscript{24} Despite the existence of high-memory systems coupled with low memory prices, the Schur complement still finds its applications in different areas of computational mechanics through parallel computing.\textsuperscript{25–27} To the best of our knowledge, the only application of a parallel Schur complement algorithm to solve a system of equations based on compact-like schemes was performed by Eliasson.\textsuperscript{28,29} In his report, Eliasson solves a 2-D Vlasov-Maxwell equation for a plasma with mobile magnetized electrons and ions. Although the problem is two-dimensional, he only applied a Schur complement type algorithm in one direction. We are proposing a different parallel Schur complement method than the one used by Eliasson as his methodology provides limited scalability. The proposed Schur complement method is meant to offer scalability for large-scale computational platforms and follows closely the methodology proposed by Kocak & Akay\textsuperscript{23} who used the Schur complement method in conjunction with a low order two-dimensional Finite Element Method (FEM). Hence, this section gives a brief description of the proposed parallel Schur complement method followed by an application to a compact scheme.

A. Brief Formulation

In the Schur complement method, a domain is first decomposed into non-overlapping sub-domains and a global solution is achieved via a coupled solution of these sub-domains. Neighboring sub-domains will share common grid points along the interface (See Figure 2). For simplicity, each sub-domain will be assigned a single processor. After applying a compact finite difference scheme (or filtering) to each sub-domain, a system of equations of a domain with $N$ sub-domains has the following form:

$$
\begin{bmatrix}
K_{11} & 0 & 0 & K_{1 \Gamma} \\
K_{22} & 0 & K_{2 \Gamma} & \\
0 & K_{N,N} & K_{N \Gamma} & \\
K_{1 \Gamma} & K_{2 \Gamma} & \cdots & K_{N \Gamma}
\end{bmatrix}
\begin{bmatrix}
f'_1 \\
f'_2 \\
f'_N \\
f'_\Gamma
\end{bmatrix}
= \begin{bmatrix}
f_1 \\
f_2 \\
f_N \\
f_\Gamma
\end{bmatrix},
$$

(3)

where $K_{ii}$ ($i = 1, \ldots, N$), $K_{i \Gamma}$, and $K_{i \Gamma} = K_{i \Gamma}^T$ denote the stiffness (coefficient) matrices of the sub-domains, the interfaces $\Gamma$, and coupling between sub-domains and the interfaces, respectively. The sub-domain vectors are $f'_i$ and $f_i$, and the interface vectors are $f'_\Gamma$ and $f_\Gamma$, respectively. The interface equations are solved before solving the solution of each sub-domain. Hence, the interface equation of the system can be written in a compact form as

$$
(K_{\Gamma \Gamma} - G_{\Gamma \Gamma}) f'_\Gamma = f_\Gamma - g_\Gamma,
$$

(4)

where

$$
G_{\Gamma \Gamma} = \sum_{i=1}^N K_{i \Gamma} A_{i \Gamma}, \quad g_\Gamma = \sum_{i=1}^N K_{i \Gamma} A_i,
$$

(5)

and

$$
A_{i \Gamma} = K_{i \Gamma}^{-1} K_{i \Gamma}, \quad A_i = K_{i i}^{-1} f'_i.
$$

(6)

The system of equations given in Eq. (4) is known as the Schur complement equation. The dimension of the Schur complement matrix depends on the number of unknowns on the interface, where the number of interface unknowns increases with the number of sub-domains.\textsuperscript{23} To assemble the Schur complement matrix in parallel, contributions to $G_{\Gamma \Gamma}$ and $g_\Gamma$ are first computed by each sub-domain processor without message passing. Then, the interface matrix coefficient, $(K_{\Gamma \Gamma} - G_{\Gamma \Gamma})$, and the source vector, $f_\Gamma - g_\Gamma$, are assembled via message passing.

B. Application of the Schur Complement to Compact Schemes

In Kocak & Akay’s paper,\textsuperscript{23} the parallel Schur complement algorithm was written with a finite element flavor in mind. Nevertheless, the algorithm is equally applicable to implicit compact differencing and filtering.
schemes. The Message-Passing Interface (MPI) libraries coupled with the Fortran 90 programming language are used throughout this work. Now, consider a one-dimensional grid partitioned into four sub-domains \((N = 4)\) shown in Figure 2. There are a total of 19 grid points and each sub-domain has 4 grid points. In addition, each sub-domain shares a common grid point at the interface, i.e. sub-domains 1 and 2 share grid point index 17, sub-domains 2 and 3 share grid point index 18 and finally sub-domains 3 and 4 share grid point index 19. Equation (1) will be used to estimate the spatial derivatives at the interior grid points. For the points next to the boundaries, i.e. \(i = 2\) and \(i = N_g - 1\), the following fourth-order central compact scheme is used

\[
\frac{1}{4} f_1' + f_2' + \frac{1}{4} f_3' = \frac{3}{4\Delta \xi} (f_3 - f_1),
\]

(7)

\[
\frac{1}{4} f_{N_g-2}' + f_{N_g-1}' + \frac{1}{4} f_{N_g}' = \frac{3}{4\Delta \xi} (f_{N_g} - f_{N_g-2}).
\]

(8)

Finally, for the points on the left and right boundary, i.e. \(i = 1\) and \(i = N_g\), the following one-sided third-order compact scheme is used

\[
f_1' + 2 f_2' = \frac{1}{2\Delta \xi} (-5 f_1 + 4 f_2 + f_3),
\]

(9)

\[
f_{N_g}' + 2 f_{N_g-1}' = \frac{1}{2\Delta \xi} (5 f_{N_g} - 4 f_{N_g-1} - f_{N_g-2}).
\]

(10)

Applying the compact scheme by Lele, i.e. Equations (1) and (7) through (10), to the domain in Figure 1, the resulting coefficient matrices in Eq. (3) are

\[
K_{11} = \begin{bmatrix}
1 & 2 & 0 & 0 \\
1/4 & 1 & 1/4 & 0 \\
0 & 1/3 & 1 & 1/3 \\
0 & 0 & 1/3 & 1
\end{bmatrix},
K_{22} = K_{33} = \begin{bmatrix}
1 & 1/3 & 0 & 0 \\
1/3 & 1 & 1/3 & 0 \\
0 & 1/3 & 1 & 1/3 \\
0 & 0 & 1/3 & 1
\end{bmatrix},
\]

\[
K_{44} = \begin{bmatrix}
1 & 1/3 & 0 & 0 \\
1/3 & 1 & 1/3 & 0 \\
0 & 1/4 & 1 & 1/4 \\
0 & 0 & 2 & 1
\end{bmatrix},
K_{\Gamma \Gamma} = \begin{bmatrix}
1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix},
\]

\[
K_{1\Gamma} = K_{\Gamma 1}^T = \begin{bmatrix}
1 & 0 & 0 & 0/3 \\
0 & 0 & 0 & 1/3 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix},
K_{2\Gamma} = K_{\Gamma 2}^T = \begin{bmatrix}
1/3 & 0 & 0 & 0 \\
0 & 0 & 0 & 1/3 \\
0 & 0 & 0 & 0
\end{bmatrix},
\]

\[
K_{3\Gamma} = K_{\Gamma 3}^T = \begin{bmatrix}
1/3 & 0 & 0 & 0 \\
0 & 0 & 0 & 1/3 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix},
K_{4\Gamma} = K_{\Gamma 4}^T = \begin{bmatrix}
1/3 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0
\end{bmatrix},
\]

(11)
and finally, the assembled Schur complement coefficient matrix in Eq. (3) is found to be

\[
K_{GT} - G_{GT} = \begin{bmatrix}
\begin{pmatrix}
\frac{548}{255} & \frac{-1}{165} & 0 \\
\frac{-1}{165} & \frac{14}{55} & \frac{-1}{165} \\
0 & \frac{-1}{165} & \frac{142}{525}
\end{pmatrix}
\end{bmatrix}
\] (12)

In the Schur complement matrix, i.e. Eq. (12), the off-diagonal terms or the interface coupling coefficients are equal and are, \(-1/165 = -0.00606\) for the case of four points in each sub-domain. Table 1 shows the effect of increasing the number of grid points in each sub-domain keeping the total number of sub-domains fixed at four. For the case of four points in each sub-domain, the Schur complement matrix must be solved directly via inversion or by the popular Thomas algorithm since the matrix is tri-diagonal. However, as the number of points in each sub-domain increases, we see that the coupling coefficients decreases. There is a dramatic decrease as the number of points increases. In other words, the Schur complement matrix becomes strongly diagonally dominant as the number of points in each sub-domain increases. In the case of 64 points in each sub-domain, the Schur complement matrix, in a machine precision sense, is a diagonal matrix. Hence, if the coupling coefficients are small enough compared to the diagonal terms, matrix inversion or some other algorithm may not be required and the interface variables can be approximated by dividing the right hand side of Eq. (3) with the diagonal terms of the Schur complement matrix. Eliasson also reported the same behavior of the coupling coefficients (as the number of sub-domain points increase) when he used the Schur complement method to solve the Vlasov-Maxwell equation. In his methodology, he uses Jacobi iteration to solve the Schur complement matrix. However, when the coupling coefficients were small enough, i.e. approaching machine zero, Jacobi iteration was not applied and the Schur complement matrix was solved directly. This behavior then of the coupling coefficient is particularly advantageous when extending the parallel Schur complement to 3-D calculations as this would save considerable amounts computation time. On the other hand, if we increase the number of sub-domains, keeping the total number of grid points fixed, the numerical value of the coupling coefficients will increase.

Table 1. Effect of number of points in each sub-domain on the coupling coefficient of the Schur complement matrix for the compact differencing scheme and spatial filtering scheme with four sub-domains total.

<table>
<thead>
<tr>
<th>Number of points in each sub-domain</th>
<th>Differencing Scheme</th>
<th>Filtering Scheme</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>(-6.06 \times 10^{-3})</td>
<td>(-5.94 \times 10^{-2})</td>
</tr>
<tr>
<td>8</td>
<td>(-1.29 \times 10^{-4})</td>
<td>(-1.40 \times 10^{-2})</td>
</tr>
<tr>
<td>16</td>
<td>(-5.85 \times 10^{-8})</td>
<td>(-8.11 \times 10^{-4})</td>
</tr>
<tr>
<td>32</td>
<td>(-1.20 \times 10^{-14})</td>
<td>(-2.75 \times 10^{-6})</td>
</tr>
<tr>
<td>64</td>
<td>(-5.06 \times 10^{-28})</td>
<td>(-3.16 \times 10^{-11})</td>
</tr>
</tbody>
</table>

Now for the spatial filtering. The filtering coefficient in Eq. (2) is set to \(\alpha_f = 0.47\). For the points next to the left-hand and right-hand side boundary, i.e. \(i = 2, 3\) and \(i = N - 2, N - 1\), a sixth-order, one-sided right-hand side stencil is used. The boundary points, however, are left unfiltered. See Reference [15] for more details regarding the formulation of this spatial filtering for the near boundary points. The Schur complement coupling coefficients for the tri-diagonal spatial filter are shown in the last column of Table 1. As we can see, the decrease in the coupling coefficient for the filtering scheme is not as dramatic compared to its compact differencing scheme counterpart. A significant decrease decrease occurs when the number of sub-domain points increases from 32 to 64. It is worth mentioning that in the general numerical methodology of LES, filtering is applied before the end of each time step advancement. The compact differencing scheme, however, is applied several times depending on the number of stages that are present in an explicit time advancement scheme. Therefore, even if direct inversion of the Schur complement matrix is required for the filtering scheme, the computational cost will not be as substantial compared to the compact scheme since it is only applied once and not several times for each time step.
III. 1-D and 2-D Linearized Euler Equation (LEE) Test Cases

This section details some work and results of a 1-D and 2-D LEE test case utilizing the Schur complement. The 1-D and 2-D LEE solutions are then compared to analytical solutions with excellent agreement.

A. 1-D LEE Test Case

The governing 1-D Linearized Euler Equation (LEE) is by

\[
\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{E}}{\partial x} = 0, \tag{13}
\]

where

\[
\mathbf{U} = \begin{pmatrix} \rho \\ u \\ p \end{pmatrix}, \quad \mathbf{E} = \begin{pmatrix} M\rho + u \\ Mu + p \\ Mp + u \end{pmatrix}. \tag{14}
\]

where \( \rho \) is the density, \( u \) is the velocity in the \( x \) direction, \( p \) is the pressure and \( M \) is the Mach number, respectively. Subsonic non-reflecting boundary conditions are prescribed at the inlet and outlet. The inlet boundary conditions are given by

\[
\begin{align*}
\frac{\partial \rho}{\partial t} &= \frac{\partial p}{\partial t} \tag{15} \\
\frac{\partial u}{\partial t} &= -\frac{\partial p}{\partial t} \tag{16} \\
\frac{\partial p}{\partial t} &= \frac{M - 1}{2} \left( \frac{\partial u}{\partial x} - \frac{\partial p}{\partial x} \right) \tag{17}
\end{align*}
\]

while the outlet boundary conditions are specified as

\[
\begin{align*}
\frac{\partial \rho}{\partial t} &= \frac{\partial p}{\partial t} - M \left( \frac{\partial \rho}{\partial x} - \frac{\partial p}{\partial x} \right) \tag{18} \\
\frac{\partial u}{\partial t} &= \frac{\partial p}{\partial t} \tag{19} \\
\frac{\partial p}{\partial t} &= -\frac{M + 1}{2} \left( \frac{\partial u}{\partial x} + \frac{\partial p}{\partial x} \right) \tag{20}
\end{align*}
\]

The 1-D Linearized Euler Equation above is solved on a mesh consisting of 131 equally spaced grid points with \( x \in [0, 20] \). The mesh is split into four non-overlapping domains similar to the one shown in Figure 2. Hence, each sub-domain has 32 grid points (\( N_{P_x} = 32 \)). The free stream Mach number is set at \( M = 0.5 \) and \( \Delta t = 0.005 \). The filtering coefficient is set to \( \alpha_f = 0.47 \). For the initial conditions, a modified aeroacoustic benchmark problem \(^{16,20,30} \) is specified by a Gaussian entropy disturbance given as

\[
\begin{align*}
\rho &= 1 + 0.1 \exp \left[ -\frac{(x - x_c)^2}{2} \right], \tag{21} \\
p &= 1, \tag{22} \\
u &= 1, \tag{23}
\end{align*}
\]

where \( x_c = 5 \). The compact scheme proposed by Lele\(^{13} \) is used for spatial discretization, i.e. Eqs. (1) and (7) through (10). For time advancement, the standard fourth-order Runge-Kutta method is used.

The results obtained with the parallel Schur complement method are compared to an analytical solution given by

\[
\begin{align*}
\rho &= 1 + 0.1 \exp \left[ -\frac{(x - Mt - x_c)^2}{2} \right]. \tag{24}
\end{align*}
\]

Figure 3(a) shows the initial density and pressure waveform along the \( x \)-axis. As time advances, the density waveform travels from left to right. Figure 3(b) shows the density waveform at \( t = 20 \). We notice that the solution of the parallel Schur method matches very well when compared to the analytical solution.
waveform eventually reaches the end of the domain and leaves. Table 2 shows the root mean square (R.M.S) error for the 1-D LEE test case for different number of sub-domain points. We also report the R.M.S. error obtained from a serial 1-D LEE code. Here, \( N_x \) is the total number of grid points, \( N_{Px} \) is the number of points per sub-domain and R.M.S. is simply the root mean square error. Keep in mind that the number of points per sub-domain, \( N_{Px} \), does not include the interface point. The number of domains is kept at four. The corresponding coupling coefficients for the three cases studied can be found in Table 1.

Table 2. Root mean square error of the density waveform from a serial code and the parallel Schur complement compared to an exact solution for 1-D LEE. Four sub-domains used throughout for the parallel Schur complement.

<table>
<thead>
<tr>
<th>( N_x )</th>
<th>( N_{Px} ) (Schur Only)</th>
<th>R.M.S. Error (Serial)</th>
<th>R.M.S. Error (Schur)</th>
</tr>
</thead>
<tbody>
<tr>
<td>67</td>
<td>16</td>
<td>9.96 \times 10^{-5}</td>
<td>1.94 \times 10^{-4}</td>
</tr>
<tr>
<td>131</td>
<td>32</td>
<td>1.19 \times 10^{-5}</td>
<td>1.85 \times 10^{-5}</td>
</tr>
<tr>
<td>259</td>
<td>64</td>
<td>1.11 \times 10^{-5}</td>
<td>1.43 \times 10^{-5}</td>
</tr>
</tbody>
</table>

Overall, the R.M.S error decreases as the total number of grid points and points per sub-domain increases. We also performed a test whereby only the diagonal terms of the Schur complement matrix were inverted. The R.M.S. error changed slightly for the first case of 67 points whereby the computed R.M.S error is 2.01 \times 10^{-4}. The remaining two cases for the R.M.S. error were different only after the 13th decimal. We also note the difference in the R.M.S. errors reported between the serial 1-D LEE version and the parallel Schur complement when compared to the exact solution. The largest difference is reported for the first test case of \( N_x = 67 \) points. It is important to note that using 67 grid points is rather coarse for a 1-D LEE problem such as the one here. However, there is only a small difference in the R.M.S. errors for the last two cases and is close to the serial results. The small differences in R.M.S. errors could be a direct result of the linear algebra performed to solve the Schur complement matrix. Also notice that the R.M.S errors do not decrease following a sixth-order accurate scale as the number of grid points is doubled. This is due to the fact that we did not keep the CFL number constant, i.e. we reduced \( \Delta x \) but kept \( \Delta t \) constant. Nonetheless, an overall error in the order of 10^{-4} and 10^{-5} is satisfactory. Hence, the Schur complement method has been applied to a simple 1-D LEE test case and the solution compares very well to its corresponding analytical solution.

Figure 3. Density waveform along the \( x \)-axis at two different instances in time for 1-D LEE. Case for 131 grid points.
B. 2-D LEE Test Case

In this section, the second test case chosen is a solution of a 2-D LEE. The chosen benchmark corresponds to Category (3) in Reference [30]. The LEE governing equation in 2-D can be written as

\[
\frac{\partial U}{\partial t} + \frac{\partial E}{\partial x} + \frac{\partial F}{\partial y} = 0,
\]

where

\[
U = \begin{pmatrix}
\rho \\
u \\
v \\
p
\end{pmatrix}, \quad E = \begin{pmatrix}
M_x \rho + u \\
M_x u + p \\
M_x v \\
M_x p + u
\end{pmatrix}, \quad F = \begin{pmatrix}
M_y \rho + v \\
M_y u + p \\
M_y v + p \\
M_y p + v
\end{pmatrix},
\]

and \(\rho, u, p\) are defined in the previous section, \(v\) is the velocity in the \(y\) direction, and \(M_x\) and \(M_y\) are the Mach numbers in the \(x\) and \(y\) directions, respectively. Tam and Webb’s non-reflecting boundary conditions are used on the top, bottom and right boundaries and are given by

\[
\frac{\partial \rho}{\partial t} = -M_x \frac{\partial \rho}{\partial x} - \frac{\partial p}{\partial t} + M_x \frac{\partial p}{\partial x},
\]

\[
\frac{\partial u}{\partial t} = -M_x \frac{\partial u}{\partial x} - \frac{\partial p}{\partial x},
\]

\[
\frac{\partial v}{\partial t} = -M_x \frac{\partial v}{\partial x} - \frac{\partial p}{\partial y},
\]

\[
\frac{\partial p}{\partial t} = -V(\theta) \left( \cos \theta \frac{\partial p}{\partial x} + \sin \theta \frac{\partial p}{\partial y} + \frac{p}{2r} \right),
\]

where

\[
V(\theta) = M \cos \theta + (1 - M^2 \sin^2 \theta)^{1/2},
\]

and \((r, \theta)\) are the polar coordinates for the boundary points centered at the location of the acoustic source. For this 2-D case, the origin \((0,0)\) is chosen as the acoustic source location whereby the domain size is set as \((x,y) \in [-100, 100] \times [-100, 100]\). Only the symmetry test is performed in this section whereby entropy and acoustic sources are specified as initial conditions,

\[
\rho = \exp \left[ -\frac{\ln(2)}{9} \right],
\]

\[
u = 0,
\]

\[
v = 0,
\]

\[
p = \exp \left[ -\frac{\ln(2)}{9} \frac{x^2 + y^2}{2} \right].
\]

The analytical solution for this problem can be found in Reference [30]. Figure 4(a) shows the computational grid used for the 2-D LEE problem along with its partitioning in the \(x\) direction. All points on the \(x\) and \(y\) axes are equally spaced. The Mach numbers are set to \(M_x = 0.5\) and \(M_y = 0\) with \(\Delta t = 0.5\). Figure 4(b) shows the pressure waveform solution computed by the Schur complement and compared to the analytical.
solution at $t = 33$. We note the very good agreement between the numerical and exact solution in Figure 4(b).

Table 3 shows the R.M.S error and speed-up tests for the 2-D LEE problem with two different number of grid points in each direction. $N_x$ and $N_y$ are the number of grid points in the $x$ and $y$ directions, respectively. In addition, $N_{bx}$ and $N_{by}$ are the number blocks in the $x$ and $y$ directions, respectively, and $N_{procs}$ is the total number of processors used. Keep in mind that the size of the Schur complement matrix is dependent on the number of interfaces in a particular direction. As an observation, test cases $B$ and $E$ have only one interface in the $x$ and $y$ directions. In addition, the coupling coefficient computed for $N_{Px} = 65$ is very close to the coupling coefficient for $N_{Px} = 64$ shown in Table 1. All runs were performed on a departmental Linux cluster whereby each node has two AMD 1.4 GHz processors and each node is connected via regular ethernet (see Acknowledgements).

Table 3. Root mean square (R.M.S) error of the pressure waveform along the $x$-axis of the parallel Schur complement compared to an exact solution for the 2-D LEE. The Speed-up study is compared between the parallel Schur complement codes only.

<table>
<thead>
<tr>
<th>Case</th>
<th>$N_x \times N_y$</th>
<th>$N_{Px} \times N_{Py}$</th>
<th>$N_{bx} \times N_{by}$</th>
<th>$N_{procs}$</th>
<th>R.M.S. Error</th>
<th>Speed-up</th>
</tr>
</thead>
<tbody>
<tr>
<td>Serial</td>
<td>131 × 131</td>
<td>131 × 131</td>
<td>1 × 1</td>
<td>1</td>
<td>8.17 × 10⁻⁴</td>
<td>-</td>
</tr>
<tr>
<td>$A$</td>
<td>131 × 131</td>
<td>32 × 131</td>
<td>4 × 1</td>
<td>4</td>
<td>9.54 × 10⁻⁴</td>
<td>1.00</td>
</tr>
<tr>
<td>$B$</td>
<td>131 × 131</td>
<td>65 × 65</td>
<td>2 × 2</td>
<td>4</td>
<td>9.54 × 10⁻⁴</td>
<td>1.36</td>
</tr>
<tr>
<td>$C$</td>
<td>131 × 131</td>
<td>32 × 65</td>
<td>4 × 2</td>
<td>8</td>
<td>9.73 × 10⁻⁴</td>
<td>2.01</td>
</tr>
<tr>
<td>Serial</td>
<td>259 × 259</td>
<td>259 × 259</td>
<td>1 × 1</td>
<td>1</td>
<td>1.66 × 10⁻⁴</td>
<td>-</td>
</tr>
<tr>
<td>$D$</td>
<td>259 × 259</td>
<td>64 × 259</td>
<td>4 × 1</td>
<td>4</td>
<td>2.12 × 10⁻⁴</td>
<td>1.00</td>
</tr>
<tr>
<td>$E$</td>
<td>259 × 259</td>
<td>129 × 129</td>
<td>2 × 2</td>
<td>4</td>
<td>2.11 × 10⁻⁴</td>
<td>1.23</td>
</tr>
<tr>
<td>$F$</td>
<td>259 × 259</td>
<td>64 × 129</td>
<td>4 × 2</td>
<td>8</td>
<td>2.12 × 10⁻⁴</td>
<td>1.89</td>
</tr>
</tbody>
</table>

Based on the R.M.S. errors in Table 3, we note that the agreement between the Schur complement and the exact solution is very good. We also show the results from a serial 2-D LEE code. For each case, as the number of points is roughly doubled in each direction, the R.M.S. error decreases by approximately four times for each corresponding block topology. As in the 1-D LEE test case, the CFL number is not
constant, which is why we are not seeing a sixth-order reduction in the R.M.S. errors. In addition, the block arrangements hardly had an effect on the R.M.S. errors. We observe that the R.M.S errors for the 2-D LEE parallel Schur complement are slightly higher than its serial counterpart. Again, we believe that these small errors are introduced when the Schur complement matrix is solved numerically. However, the speed-ups obtained between different block topologies are interesting. A speed-up per-time step of roughly 30% is gained for a block arrangement of 2 x 2 compared to a 4 x 1 using the same number of grid points. Gropp et al.\textsuperscript{32} reported a similar observation when they parallelized a 2-D Poisson problem using a second order finite difference scheme. This can be explained by examining the number of points along the shared interface between each block. For Case A, a buffer size of 131 points along with the four primitive variables (131 x 4) needs to be sent and received between each block. If a 2-D decomposition is done as in Case B, half that amount is needed and thus the increase in performance. Cases D and E also report similar observations in terms of performance though slightly lower. A speed-up of roughly 2 times is reported between cases A and C. For the speed-up analysis, Case A was taken as a reference for the coarse grid setup and Case D for the finer grid setup. As a note, timing is measured from the first time step to the last. The timing measured here does not include the start-up time which is found to be rather small (on the order of 2% of the simulation time).

We also used the same methodology whereby only the diagonal elements of the Schur complement matrix were inverted for the above cases. Here, we did not see any gain in speed-up using this methodology since this 2-D problem is rather computationally inexpensive, but we expect to see a performance gain for 3-D applications. Overall, the comparison between the computed 2-D LEE results via the Schur complement and the exact solution are very good. With the encouraging results obtained for the 1-D and 2-D LEE test cases, we now apply the parallel Schur complement to a full 3-D LES jet.

IV. 3-D Jet LES via the Schur Complement

This section details some recent work that was done for a 3-D LES jet test case using the parallel Schur complement methodology. Results obtained are compared to previously simulated results from our single-block code.\textsuperscript{33} The next section briefly describes the 3-D LES methodology aside from the parallelization used.

A. Brief Description of 3-D LES Methodology

The single-block 3-D LES code was developed by Uzun et al.\textsuperscript{34,35} and it includes both the classical\textsuperscript{36} and a localized dynamic\textsuperscript{37} Smagorinsky (DSM) subgrid-scale model. However, the modeling of the subgrid-scale stress tensor still raises some fundamental issues as discussed by Bogey & Bailly.\textsuperscript{38,39} Eddy-viscosity models such as the classical Smagorinsky subgrid-scale model\textsuperscript{36} and the localized dynamic subgrid-scale model (DSM)\textsuperscript{37,40} might dissipate the turbulent energy through a wide range of scales up to the larger ones, which should be dissipation free at sufficiently high Reynolds numbers.\textsuperscript{41} In addition, since the eddy-viscosity has the same functional form as the molecular viscosity the effective Reynolds number is reduced in the simulated flows.\textsuperscript{42} See References [38] and [43] for a thorough analysis and discussion on the shortcomings of eddy viscosity subgrid-scale model on jet flows. An alternative to the use of an explicit eddy-viscosity model is the use of spatial filtering for modeling the effects of the subgrid-scales. This approach minimizes the amount of dissipation on the smaller resolved scales. Using this alternative, the turbulent energy is only dissipated when it is transferred from the larger scales to the smaller scales discretized by the mesh grid.\textsuperscript{38} Hence, for the jet simulated here, we use spatial filtering as an implicit subgrid-scale model.

Since we have a near sonic jet, the unsteady, Favre-filtered, compressible, non-dimensional LES equations are solved. We transform from curvilinear coordinates to a uniform grid in computational space. The compact scheme used here is shown in Section II.B. The sixth-order tri-diagonal spatial filter proposed by Vishal and Gaitonde\textsuperscript{15} (partly shown in Section II.B) is employed with the filter parameter set to $\alpha_f = 0.47$. For time advancement, the explicit four-stage, fourth-order Runge-Kutta scheme is used. Tam and Dong’s 3-D radiation and outflow boundary conditions\textsuperscript{44} are implemented on the boundaries. In addition, a sponge zone\textsuperscript{45} is attached to the end of the computational domain to dissipate the vortices present in the flow field before they hit the outflow boundary. This is done so that unwanted reflections from the outflow boundary are suppressed. Figure 5 shows a schematic of the boundary conditions used in the 3-D LES code. A more in-depth discussion on the numerical methods used can be found in Uzun.\textsuperscript{18}
To excite the mean flow, randomized perturbations in the form of induced velocities from a vortex ring proposed by Bogey & Bailly\cite{46} are added to the velocity profile at a short distance (approximately one jet radius) downstream from the inflow boundary (see Figure 5). This is done to ensure the break up of the potential core within a reasonable distance. Studies regarding the effect of this inflow forcing on jet noise can be found in Lew et al.\cite{33} and Bogey & Bailly.\cite{11}

B. Setup

To test our Schur complement methodology for a 3-D round jet, we use a setup similar to that of a previous simulation with the single-block code\cite{33}. Further details can be found in Reference [33]. In brief, the physical part of the domain extends to approximately $25r_o$ in the streamwise direction and $-15r_o$ to $15r_o$ in the transverse $y$ and $z$ directions. The total number of grid points used here is $287 \times 128 \times 128$ in the $x$-$y$-$z$ directions respectively. This gives a total of approximately 4.7 million grid points. Figure 6(a) shows the $x-z$ cross sectional plane of the computational domain. Notice that there are more points packed near the shear layer in order to resolve the relatively high velocity gradients there. We consider a hyperbolic tangent velocity profile on the inflow boundary given by

$$u(r) = \frac{1}{2} U_o \left[ 1 - \tanh \left( b \left( \frac{r}{r_o} - \frac{r_o}{r} \right) \right) \right],$$

where $r = \sqrt{y^2 + z^2}$, $r_o$ is the initial jet radius and is set to unity, $U_o$ is the mean jet centerline velocity at the inflow boundary. The parameter that controls the thickness of the shear layer is $b$. In our code we have set this parameter to $b = 3.125$. A higher value of $b$ implies a thinner shear layer. For comparison, Freund\cite{3} used a value of 12.5 since his grid was fine enough to resolve thin shear layers. Hence, our $b$ parameter corresponds to that of a relatively thick shear layer. Based on the minimum grid resolution, the time step is set to $\Delta t = 0.25$. In addition, the following Crocco-Buseman relation for an isothermal jet is specified for the density profile on the inflow boundary

$$\rho(r) = \rho_o \left( 1 + \frac{\gamma - 1}{2} M_r^2 \frac{\bar{u}(r)}{U_o} \left( 1 - \frac{\bar{u}(r)}{U_o} \right) \right)^{-1},$$

where $M_r = 0.9$.

We study a subsonic jet with a Mach number of 0.9 and Reynolds number $Re_D = \rho_j U_j D_j/\mu_j = 100,000$ where $\rho_j$, $U_j$ ($U_j = U_o$) and $\mu_j$ are the jet centerline density, velocity and viscosity at the inflow. $D_j$ is simply the jet diameter. Since this is an isothermal jet, the centerline temperature is the same as the ambient
temperature. The vortex ring used here contains a total of 16 azimuthal jet modes of forcing. Bogey and Bailly\textsuperscript{11} performed a simulation with all modes present and later removed the first four modes and found that the jet was quieter with the latter case. Hence, for the Schur complement test case, we remove the first four modes of forcing. This also corresponds to Test case \textit{rf4} in Reference [33].

Figure 6(b) shows the block arrangement for the 3-D LES test case using the parallel Schur complement. A block arrangement of $N_{bx} \times N_{by} \times N_{bz} = 1 \times 1 \times 16$ is chosen mainly because we want to make a one-to-one comparison of the near-field and far-field solution between the Schur complement and single-block code. Recall that in Section I, the single-block code is partitioned in this manner as well. Hence, for both the single-block and Schur complement case a total of 16 processors are used. Also, Blocks 1 through 15 (See Figure 6(b)) have 7 points in each sub-domain along the $z$ direction not including the interface points. Block 16 however, has 8 grid points. We are aware that this will lead to a slight load imbalance, but we decided to keep the number of grid points the same between the two simulations.

It is also worth mentioning that a minimum of 7 grid points is required for the spatial tri-diagonal filter used here. In this case, the coupling coefficients for the compact differencing and spatial filtering scheme are found to be $-3.38 \times 10^{-4}$ and $-1.99 \times 10^{-2}$, respectively.

C. Computational Details of the 3-D LES Schur Complement Test Case

Before we delve into the results of the 3-D jet LES, we provide some computational and performance details for this test case. We ran the simulation over the first 10,000 time steps for the initial transients to exit the domain. We then collected flow statistics over 35,000 time steps. As a note, the procedure used here follows what was performed for the single-block case.\textsuperscript{33} Thus, for both simulations, a total of 45,000 time steps were needed to achieve reasonable converged statistics.

The parallel Schur complement and single-block simulation was performed on the Compaq Alphaserver Cluster \textit{Lemieux}, at the Pittsburgh Supercomputing Center. The single-block case took approximately 5 days to complete using 16 processors in parallel. On the other hand, a total of about 11.5 days of computing time was required for the 3-D LES Schur complement test case using 16 processors. Hence, the Schur complement case is roughly 2 times slower than its single-block counterpart. The rather dismal performance of the Schur complement case is not a cause for concern and for the most part not entirely surprising. In this test case, we have only decomposed the computational grid in one direction, i.e in the $z$ direction (See Figure 6(b)). And due to this 1-D decomposition, approximately 73\% of the simulation time was spent on the communication
process, which, is why the simulation takes almost twice as long to complete compared to the single-block code.

However, recall that a speed-up of approximately 30% was gained between a 1-D and 2-D decomposition for the 2-D LEE test case (See Section III.B). Hence, this strongly suggests that a decomposition in three dimensions would definitely increase the performance of our 3-D parallel Schur complement. Gropp et al.\textsuperscript{32} reported a dramatic speed-up of a factor of nearly 3 between a 3-D decomposition and a 1-D decomposition for a parallel 3-D Poisson problem using a second-order finite difference scheme. In essence, a 3-D decomposition will effectively reduce the ‘surface area’ shared between each block and thus reducing the amount of information being passed back and forth and communication time. As an example, a decomposition of $N_{bx} \times N_{by} \times N_{bz} = 4 \times 2 \times 2$ effectively reduces the surface area shared by the interfaces in the $z$ direction by a factor of 8. Furthermore, this arrangement will increase the number of points per domain to 64 or more in each direction allowing the direct solution using the diagonal elements. Hence, work is currently underway to decompose our computational grid in three dimensions and the results will be reported in a future study.

D. Near-field Turbulent Flow Results

Figure 7 shows the instantaneous dilatation contours of our jet along with the boundaries of each block using the parallel Schur complement. From Figure 7, we note that there is a smooth transition of the dilatation contours between each block, and a close-up view in Figure 8 shows this very clearly. Hence, as an initial assessment, the smoothness of the contours implies that the parallel Schur complement applied at the interfaces between each block is at least solving the governing equations sufficiently well.

![Figure 7. Instantaneous dilatation contours of our 3-D LES iso-thermal jet. The black square box is a close-up area and is shown in Figure 8.](image)

Now we will examine some one-point statistical results. The streamwise variation of the half-velocity radius normalized by the initial jet radius is an indicator of the jet spreading rate and is shown in Figure 9(a). In essence, it is a measure of how fast the jet grows. The half-velocity radius, $r_{1/2}$ at a particular downstream location is defined as the radial location where the mean streamwise velocity is one-half the jet mean centerline velocity. The slope of the linear fit gives the measure jet growth rate and is tabulated in...
Table 4 for both the single-block and Schur complement methodologies. As a note, the range of data for unheated incompressible jets from the experimental literature\textsuperscript{47} is reported to be $0.086 \leq A \leq 0.096$. Hence, our values are lower since our physical domain length is relatively short, i.e. $x = 25r_o$. A physical domain length of at least $60r_o$ or longer is needed for the growth rate to be within range of the experimental values (See References [10] and [48]). However, our main goal here is to make a direct comparison between the Schur complement and the single-block methodology results and not a direct comparison with laboratory experiments. From Table 4, we see that there is roughly a 3\% difference in the growth rate between the parallel Schur complement and single-block result.

Table 4. Several turbulent flow results for our isothermal jet LES for the single-block and parallel Schur complement methodology.

<table>
<thead>
<tr>
<th>Methodology</th>
<th>$A$, ($\frac{d(r_o/r)}{dx/r}$)</th>
<th>$C$, ($\frac{d}{dx/r} \frac{U_o}{U_c}$)</th>
<th>$x_c/r_o$, (0.95$U_o$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single-block</td>
<td>0.071</td>
<td>0.148</td>
<td>13.07</td>
</tr>
<tr>
<td>Schur Complement</td>
<td>0.069</td>
<td>0.142</td>
<td>13.34</td>
</tr>
<tr>
<td>% Difference</td>
<td>2.8%</td>
<td>4.0%</td>
<td>2.1%</td>
</tr>
</tbody>
</table>

The third column of Table 4 indicates the slope of the mean streamwise velocity decay, $C = (\frac{d}{dx/r_o} \frac{U_o}{U_c})$. In other words, it is a measure of how fast the mean axial centerline velocity decays and the slope of the linear curve fit in Figure 9(b). Here the percent difference is 4\% between the single-block code and Schur complement. Zaman\textsuperscript{49} reports an experimental value of $C = 0.16$ for an isothermal jet. In addition, we define the potential core length as the location where the jet centerline velocity reduces to 95\% of the inflow jet velocity, $U_c(x_c) = 0.95U_o$. This is marked in Figure 9(b) as well. The last column in Table 4 shows the potential core lengths for both cases and the percent error between them. Likewise, there is a small percent difference between the single-block and Schur complement result for the potential core lengths.

Figure 10 shows the variation of normalized Reynolds stresses at the end of the physical domain, $x = 25r_o$. 

Figure 8. Close-up view of the square box shown in the Figure 7.
Figure 9. (a) Streamwise variation of the half-velocity radius normalized by the jet radius for the single-block and Schur complement (b) Mean axial velocity centerline decay.

The normalized Reynolds stresses are defined in cylindrical coordinates as follows:

\[ \sigma_{xx} = \frac{v_x' v_x'}{\bar{U}_c^2(x)}, \quad \sigma_{rr} = \frac{v_r' v_r'}{\bar{U}_c^2(x)}, \quad \sigma_{\theta\theta} = \frac{v_\theta' v_\theta'}{\bar{U}_c^2(x)}, \quad \sigma_{rx} = \frac{v_r' v_x'}{\bar{U}_c^2(x)} \]  

where \( v_x', v_r', v_\theta' \) are the axial, radial and azimuthal components of the fluctuating velocity, respectively, \( \bar{U}_c(x) \) is the mean jet centerline velocity at a particular axial location, and the overbar denotes time-averaging.

Table 5 in turn gives the values for the peak Reynolds stresses for both test cases. Again we note that the percent difference between the Schur complement simulation and single-block methodology is on the order of 3% but nonetheless is very good. Figure 11 shows the axial root mean square (rms) velocity fluctuations along the jet shear layer \((r = r_o)\) and jet centerline, respectively. The velocity fluctuations are normalized by \( \bar{U}_o \), i.e, the exit jet centerline velocity rather than \( \bar{U}_c(x) \). Again we observe the slight difference in peak turbulence intensities between the single-block code and parallel Schur complement. In a similar fashion, we show the peak values for the turbulence intensities in Table 6. The axial peak locations differed very little and the error is not shown here. But for completeness, the peak locations are \( x_p = 9.77r_o \) and \( x_p = 19.02r_o \) for the shear layer and centerline jet axis, respectively. Although not shown here, the results obtained for the radial and azimuthal turbulence intensities also compare rather well with the single-block results.

<table>
<thead>
<tr>
<th>Test Case</th>
<th>((\sigma_{xx})_p)</th>
<th>((\sigma_{rr})_p)</th>
<th>((\sigma_{\theta\theta})_p)</th>
<th>((\sigma_{rx})_p)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single-block</td>
<td>0.0575</td>
<td>0.0353</td>
<td>0.0374</td>
<td>0.0179</td>
</tr>
<tr>
<td>Schur Complement</td>
<td>0.0561</td>
<td>0.0341</td>
<td>0.0365</td>
<td>0.0174</td>
</tr>
</tbody>
</table>

| % Difference  | 2.4% | 3.4% | 2.4% | 2.8% |

The differences between the single-block and Schur complement results are not entirely surprising and for the most part are to be expected. We list two possible reasons for their differences:

1. The random number generator used for the vortex ring inflow forcing. The two codes start with a different random seed. Also, the 3-D LES code has what is called restarting capability, so that a simulation can be run in many stages. During each restart stage, a different seed is fed into random
Figure 10. Reynolds stresses at $x = 25r_o$ for both the single-block code and parallel Schur complement.
generator and thus a new set of random numbers are generated. Hence, even if both simulations, i.e. single-block and Schur complement are started with the same seed, depending on how many restarts were done for the Schur complement run, the sequence of the random numbers for the Schur run is most probably not the same as that for the single-block run. This will cause some differences in the results. Since the Schur complement run took longer to complete compared to the single-block version, the simulation was performed in many more stages (due to wall time constraints on Lemieux).

2. The small differences due to the numerics of the two methods may lead to statistical variations, because of the chaotic nature of turbulent flows and the limited statistical sample size of the simulation data.

![Figure 11. Mean axial turbulence intensities along the shear layer and jet centerline axis.](image)

(a) Along $r = r_o$  
(b) Along the jet centerline

<table>
<thead>
<tr>
<th>Test Case</th>
<th>$\frac{\langle (v_x')^2 \rangle_{r=r_o}}{U_o^2}$</th>
<th>$\frac{\langle (v_x')^2 \rangle_{r=Centerline}}{U_o^2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single-block</td>
<td>0.190</td>
<td>0.147</td>
</tr>
<tr>
<td>Schur Complement</td>
<td>0.186</td>
<td>0.142</td>
</tr>
<tr>
<td>% Difference</td>
<td>2.1%</td>
<td>3.4%</td>
</tr>
</tbody>
</table>

### E. Preliminary Far-field Aeroacoustic Results

In this section, we look at some preliminary results for far-field aeroacoustics of the 3-D LES Schur complement test case. Our initial goal was to use the porous Ffowcs-Williams Hawkings (FWH) surface integral technique to predict the far-field sound generated from our jet for the 3-D LES Schur complement. Due to time constraints, we will perform the aforementioned analysis in a future study. Nonetheless, we did perform a study whereby two points were chosen on an imaginary FWH control surface and projected the spectral solution to the far-field. We did this for both the single-block and Schur complement codes.

Flow field data is gathered on the control points at every 5 time steps over a period of 25,000 time steps. The total acoustic sampling period corresponds to a time scale in which the ambient sound wave travels about 10 times the domain length in the streamwise direction. Based on the grid resolution around the chosen control points and assuming that 6 points per wavelength are needed to accurately resolve an acoustic wave, the maximum resolved frequency corresponds to a Strouhal number of $St = 1.1$. The
Strouhal number here is defined as \( St = fD_j/U_o \) where \( f \) is the frequency, \( D_j \) and \( U_o \) are the jet nozzle diameter and the jet centerline velocity on the inflow boundary, respectively. In addition, based on the data sampling rate, there are about 14.5 temporal points per period in the highest resolved frequency. The overall sound pressure level (OASPL) is computed along an arc of radius of \( R = 60r_o \) from the jet nozzle. The angle \( \Theta \) is measured relative to the centerline jet axis (see Figure 12).

![Figure 12](image12.png)

**Figure 12.** Schematic showing the center of the arc and how the angle \( \Theta \) is measured from the jet axis.

![Figure 13](image13.png)

**Figure 13.** Far-field acoustic pressure spectra at \( R = 60r_o \) for (a) \( \Theta = 60^\circ \) and (b) \( \Theta = 75^\circ \).

Figure 13 shows the acoustic pressure spectra at \( R = 60r_o \) for two different observer angles, i.e. \( \Theta = 60^\circ \).
and $\Theta = 75^\circ$. We note that overall the Schur complement spectra are lower by approximately 1.5 dB compared to the single-block results. The smallest difference is in the low frequency range, i.e. $0.05 \leq St \leq 0.2$ for both observer angles. Using the peak spectral level as a reference, Table 7 shows the difference between the single-block code and Schur complement methodology. Hence, as an initial observation, a 3% difference in the results of the Reynolds stresses and turbulence intensities translates to a 1.5% difference in the far-field noise levels (dB log scale) for the Schur complement. We must stress that this is a rather premature conclusion to make since we did not use the FWH method to compute the far-field sound. Again, we will report the results from the FWH method in a future study. In closing, the far-field acoustic results of the Schur complement is satisfactory.

Table 7. Tabulated results for the peak acoustic pressure spectra and sound pressure level for both cases.

<table>
<thead>
<tr>
<th>Test Case</th>
<th>Peak SPL/St 60°</th>
<th>Peak SPL/St 75°</th>
<th>SPL St 60°</th>
<th>SPL St 75°</th>
</tr>
</thead>
<tbody>
<tr>
<td>Angle, $\Theta$</td>
<td>60°</td>
<td>75°</td>
<td>60°</td>
<td>75°</td>
</tr>
<tr>
<td>Single-block</td>
<td>112.7</td>
<td>110.8</td>
<td>111.4</td>
<td>108.7</td>
</tr>
<tr>
<td>Schur Complement</td>
<td>111.3</td>
<td>109.5</td>
<td>109.8</td>
<td>107.0</td>
</tr>
<tr>
<td>% Difference</td>
<td>1.2%</td>
<td>1.8%</td>
<td>1.4%</td>
<td>1.6%</td>
</tr>
</tbody>
</table>

V. Concluding Remarks and Future Work

In all, we have developed a preliminary parallel 3-D LES methodology for jet aeroacoustics via the Schur complement in order to improve computational efficiency. To test this approach, we first developed a one-dimensional and two-dimensional LEE code utilizing the parallel Schur complement and compared the solution to its corresponding analytical result with very good agreement. Turbulent flow results from the 3-D LES Schur complement test case differed by approximately 3%, whereas the far-field acoustic results differed by about 1.5% compared to its single-block counterpart. These differences are believed to come from the different numerical errors arising in the Schur complement and the single-block codes and the limited sample size available in computing statistical quantities. Nonetheless, the differences are small and the 3-D jet LES Schur complement results are satisfactory. However, the 3-D LES Schur complement is roughly 2 times slower than its single-block counterpart mainly due to the fact that block partitioning was only done in one direction. Hence, work is currently well underway to extend our 3-D LES Schur complement code from a 1-D to a 3-D decomposition. Once the 3-D decomposition is completed, we will perform a scalability study of the parallel Schur complement and we are optimistic that we will see a performance gain over the single-block methodology. In addition, we will also use the porous Ffowcs-Williams Hawkings (FWH) surface integral acoustic technique to predict the far-field sound from the near-field simulation results of the Schur complement.

Acknowledgments

The first author would like to acknowledge partial support from the Purdue Research Foundation’s (PRF) Special Incentive Research Grant (SIRG). This work was partially supported by National Computational Science Alliance under grant number ASC040044N and utilized the SGI Altix computing system at the University of Illinois, Urbana-Champaign. In addition, some of the the simulations were carried out on Purdue University’s 320 processor IBM-SP3 supercomputer, the 120 processor Sun F6800 servers and on a 104-node School of Aeronautics and Astronautics Linux cluster acquired by a Defense University Research Instrumentation Program (DURIP) grant sponsored by ARO.

References

3 Freund, J. B., “Noise Sources in a Low-Reynolds-number Turbulent Jet at Mach 0.9,” Journal of Fluid Mechanics,


