2023 INCITE Proposal Submission
Proposal

Title: High-Fidelity Turbulence Simulation of Three-Dimensional Complex Flow Separation

Principal Investigator: Ali Uzun

Organization: National Institute of Aerospace

Date/Time Generated: 6/17/2022 2:16:32 PM

Section 1: PI and Co-PI Information

Question #1

Principal Investigator: *The PI is responsible for the project and managing any resources awarded to the project. If your project has multiple investigators, list the PI in this section and add any Co-PIs in the following section.*

Principal Investigator

First Name

Ali

Last Name

Uzun

Organization

National Institute of Aerospace

Email

ali.uzun@nianet.org

Work Phone

757-864-8798

Address Line 1

100 Exploration Way

Address Line 2

(No answer given.)

City

Hampton

State

VA

Zip Code

23666

Question #2

Co-PI (s)

First Name

Mujeeb

Last Name

Malik

Organization

NASA Langley Research Center

Email

m.r.malik@nasa.gov

Question #3

Institutional Contact: *For the PI's institution on the proposal, identify the agent who has the authority to review, negotiate, and sign the user agreement on behalf of that institution. The person who can commit an organization may be someone in the contracts or procurement department, legal, or if a*

university, the department head or Sponsored Research Office or Grants Department.

Institutional Contact

Institutional Contact Name

Susan Sorlie

Institutional Contact Phone

757-325-6956

Institutional Contact Email

susan.sorlie@nianet.org

Section 2: Project Information

Question #1

Select the category that best describes your project.

Research Category

Engineering: Fluids and Turbulence

Question #2

Please provide a project summary in two sentences that can be used to describe the impact of your project to the public (50 words maximum)

Project Summary

The proposed simulation will provide high-quality turbulence data to assess computational fluid dynamics tools for prediction of aircraft maximum lift. These validated computational tools will enable certification by analysis resulting in hundreds of millions of dollars savings in aircraft development programs and substantially reducing time to market.

Section 3: Early Career Track

Question #1

Early Career

Starting in the INCITE 2022 year, INCITE is committing 10% of allocatable time to an Early Career Track in INCITE. The goal of the early career track is to encourage the next generation of high-performance computing researchers. Researchers within 10 years from earning their PhD (after December 31st 2012) may choose to apply. Projects will go through the regular INCITE Computational Readiness and Peer Review process, but the INCITE Management Committee will consider meritorious projects in the Early Career Track separately.

***Who Can Apply**: Researchers less than 10 years out from their PhD that need LCF-level capabilities to advance their overall research plan and who have not been a previous INCITE PI.*

How to Apply:

In the regular application process, there will be a check-box to self-identify as early career.

- *The required CV should make eligibility clear.*
- *If awarded, how will this allocation fit into your overall research plan for the next 5 years?*

Projects will go through the regular INCITE review process. The INCITE Program is targeting at least 10% of allocatable time. When selecting the INCITE Career Track, PIs are not restricted to just competing in that track.

- *What is the Early Career Track?*
  - *The INCITE Program created the Early Career Track to encourage researchers establishing their research careers. INCITE will award at least 10% of allocatable time to meritorious projects.*
- *Will this increase my chances of receiving an award?*
  - *Potentially, this could increase chances of an award. Projects must still be deemed scientifically meritorious through the review process INCITE uses each year.*
- *What do I need to do to be considered on the Early Career Track?*
  - *In the application process, select 'Yes' at 'If you are within 10 years of your PhD, would you like to be considered in the Early Career Track?' You will need to write a paragraph about how the INCITE proposal fits into your 5-year research and career goals.*
- *What review criteria will be used for the Early Career Track?*
  - *The same criteria for computational readiness and scientific merit will be applied to projects in the Early Career Track as will be applied to projects in the traditional track. The different will be manifest in awards decisions by the INCITE management committee.*

Early Career Track

If you are within 10 years of your PhD, would you like to be considered in the Early Career Track? Choosing this does not reduce your chances of receiving an award.

No

If ‘yes’, what year was your PhD? If 'no' enter N/A

N/A

If ‘yes’, how will this allocation fit into your overall research plan for the next 5 years? If 'no' enter N/A.

N/A

Section 4: INCITE Allocation Request & Other Project Funding/Computing Resources

Question #1

OLCF Summit (IBM / AC922) Resource Request - 2023

Node Hours

565000

Storage (TB)

500

Off-Line Storage (TB)

1000

Question #2

OLCF Frontier (Cray Shasta) Resource Request – 2023

Question #3

OLCF Frontier (Cray Shasta) Resource Request – 2024

Question #4

OLCF Frontier (Cray Shasta) Resource Request – 2025

Question #5

ALCF Theta (Cray XC40) Resource Request - 2023

Question #6

ALCF Polaris Resource Request - 2023

Question #7

ALCF Polaris Resource Request - 2024

Question #8

ALCF Polaris Resource Request - 2025

Question #9

ALCF Aurora (Intel X^e) Resource Request – 2023

Question #10

ALCF Aurora (Intel X^e) Resource Request – 2024

Question #11

ALCF Aurora (Intel X^e) Resource Request – 2025

Question #12

List any funding this project receives from other funding agencies.

Funding Sources

Question #13

List any other high-performance computing allocations being received in support of this project.

Other High Performance Computing Resource Allocations

Section 5: Project Narrative and Supplemental Materials

Question #1

Using the templates provided here, please follow the INCITE Proposal Preparation Instructions to prepare your proposal. Elements needed include (1) Project Executive Summary, (2) Project Narrative, (3) Personnel Justification and Management Plan, (4) Milestone Table, (5) Publications Resulting from prior INCITE Awards (if appropriate), and (6) Biographical Sketches for the PI and all co-PI's. Concatenate all materials into a single PDF file. Prior to submission, it is strongly recommended that proposers review their proposals to ensure they comply with the proposal preparation instructions.

Concatenate all materials below into a single PDF file.

1. **Project Executive Summary (One Page Max)**
2. **Project Narrative (15 Pages Max)**
3. **Personnel Justification and Management Plan (1 Page Max)**
4. **Milestone Table**
5. **Publications resulting from prior INCITE Awards (if appropriate)**
6. **Biographical Sketches for the PI and all co-PI's.**

INCITE_2023.pdf
The attachment is on the following page.

PROJECT EXECUTIVE SUMMARY

Title: High-Fidelity Turbulence Simulation of Three-Dimensional Complex Flow Separation
PI and Co-PIs: Dr. Ali Uzun and Dr. Mujeeb Malik
Applying Institution/Organization: National Institute of Aerospace
Resource Name and Number of Node Hours Requested: Summit, 0.565 million node-hours
Amount of Storage Requested: 1000 Terabytes
Executive Summary:

Turbulent boundary layers subjected to pressure gradients are commonly found in many applications. A particularly interesting application is found over the upper surface of subsonic and transonic airfoils, wherein the favorable pressure gradient over the leading-edge region is followed by an adverse pressure gradient further downstream, generated due to a change in body contour and/or the presence of a shock. The favorable pressure gradient generally leads to acceleration, which can be strong in certain cases, while the interaction with the adverse pressure gradient often leads to separation. Similar phenomena also exist in turbomachinery and wind turbine applications. Flows subjected to such conditions have proven to be particularly challenging to predict using lower-fidelity simulation tools based on various turbulence modeling approaches. In aeronautical applications, reliable lift and drag predictions are strongly dependent upon accurate computation of these complex flows. Accurate prediction of maximum aircraft lift could enable significant reduction in certification flight testing, saving hundreds of million dollars in aircraft development programs.

To further investigate turbulent flows subjected to favorable/adverse pressure gradients and generate additional computational as well as experimental data, a new benchmark test case has been recently proposed. The test case involves the turbulent boundary layer development over a wall-mounted bump geometry that generates a favorable pressure gradient region in the front portion, followed by an adverse pressure gradient in the aft section. Our contribution to this activity will be in the form of a hybrid direct numerical simulation (DNS) – large-eddy simulation (LES) for the entire three-dimensional model. To our knowledge, the proposed calculation is the very first highest-fidelity simulation of the problem, and will provide valuable data for the improvement of turbulence models for better prediction of three-dimensional flow separation.

To exploit the full potential of GPUs, our flow solver employs high-order explicit finite difference and explicit time integration schemes made up of many independent multiply-add type operations, at which the GPU excels. Parallelization is based on a hybrid combination of MPI+OpenMP+CUDA Fortran. This GPU flow solver requires only about one nanosecond per grid point per Summit node per time step.

The proposed work has a strong potential for generating a detailed dataset that can guide the development of new and improved models for use in lower-fidelity simulations tools such as wall-modeled large-eddy simulations (WMLES) and Reynolds-averaged Navier-Stokes (RANS) calculations. Improved turbulence models with better accuracy will enable simulations of practical high Reynolds number problems with significantly reduced computational cost. Correct predictions of flows subjected to pressure gradients in turn can be used to improve aerodynamic designs and reduce drag. These improvements can ultimately pave the way towards the development of aerodynamic designs with reduced overall drag and hence aircraft fuel burn reduction. In other words, these scientific advances can provide the critical knowledge and design tools needed in the development of fuel-efficient air transportation technologies in which aerodynamic drag reduction offers significant performance, energy-savings and carbon sequestration benefits. Simulation techniques of varying levels of fidelity such as RANS, hybrid RANS-LES, WMLES and wall-resolved LES, whose embedded turbulence models can be further improved using the data generated by high-resolution simulations, have direct relevance to the DOE wind energy program as well. The proposed research is hence in the energy-efficiency area that is closely related to the DOE mission.

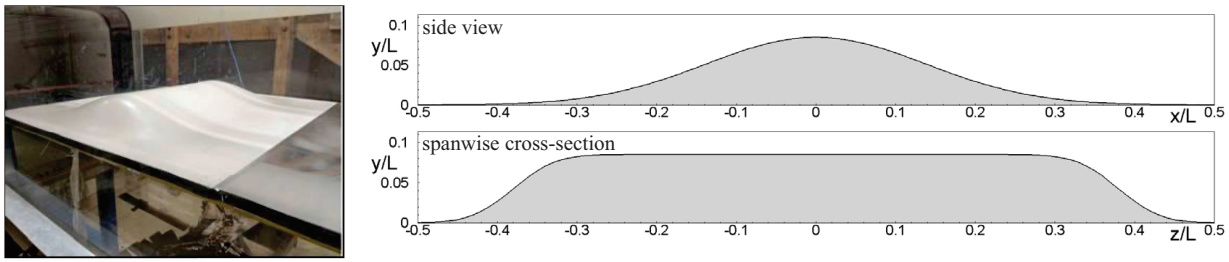
PROJECT NARRATIVE

1 SIGNIFICANCE OF RESEARCH

1.1 Research Goals and Impact

Turbulent boundary layers subjected to pressure gradients are commonly found in many practical applications. A particularly interesting application is found over the upper surface of subsonic and transonic airfoils, wherein the favorable pressure gradient over the leading-edge region is followed by an adverse pressure gradient further downstream, generated due to a change in body contour and/or the presence of a shock. The favorable pressure gradient generally leads to flow acceleration, which can be strong in certain cases, while the interaction with the adverse pressure gradient often leads to flow separation. Similar flow phenomena also exist in turbomachinery and wind turbine applications. Flows subjected to such conditions have proven to be particularly challenging to predict using lower-fidelity simulation tools based on various turbulence modeling approaches. A recent example demonstrating this deficiency of turbulence models can be found in the work of Spalart et al. [1], who employed wall-modeled large-eddy simulation (WMLES) in the form of an improved delayed detached eddy simulation (IDDES) to simulate the strong flow acceleration over a transonic bump and the subsequent shock-induced flow separation. Despite the large number of grid points used in the WMLES (nearly 1.7 billion points), comparisons of the results with the direct numerical simulation data revealed that the WMLES failed to predict the flow development in the accelerating flow region prior to flow separation. This deficiency led to an incorrect prediction of shock location, thereby yielding an incorrect flow separation location.

In aeronautical applications, reliable lift and drag predictions are strongly dependent upon accurate computation of these complex flows. Accurate prediction of maximum aircraft lift could enable significant reduction in certification flight testing, saving hundreds of million dollars in aircraft development programs. To further investigate turbulent flows subjected to favorable/adverse pressure gradients and generate additional computational as well as experimental data, a new benchmark test case has been recently proposed [2]. This new test case, which will be referred to as the "speed bump" problem from this point onward, contains a wall-mounted geometry represented in the form of a Gaussian distribution profile, as depicted in Figure 1. The problem involves a low-speed turbulent boundary layer passing over the speed bump profile that generates a favorable pressure gradient region over the front portion, followed by an adverse pressure gradient in the aft section. The speed bump model has a uniform height along much of the span, and its height is smoothly tapered down to zero by an error function near the side edges, as can be seen in Figure 1.



(a) Experimental model (b) Side view and spanwise cross-section (side walls at $z/L = \pm 0.5$, ceiling at $y/L = 0.5$)

Figure 1. 3-D speed bump geometry.

An experimental investigation campaign is underway for the speed bump flow [3, 4, 5, 6]. Meanwhile, several groups are in the process of studying this problem using various turbulence modeling approaches, including direct numerical simulation (DNS). Comparisons of the results from a number of simulations and

varying modeling approaches are of interest to the turbulence research community. This proposal requests computing resources to make a significant contribution to the ongoing community-wide exercise concerning the speed bump flow. The proposed high-fidelity simulation considers the entire three-dimensional experimental configuration rather than a slice of the geometry under the spanwise periodicity assumption. To our knowledge, the proposed simulation, which will be in the form of a hybrid DNS – wall-resolved large-eddy simulation (WRLES), is the very first highest-fidelity simulation of the three-dimensional model at the Reynolds number of $Re_L = 2$ million (based on the upstream reference velocity, U_∞, and model width, L), and requires about 160 billion grid points. This Reynolds number is high enough to preclude relaminarization due to the strong favorable pressure gradient, and also generate strong separation in the adverse pressure gradient region. Because of computational resource limitations, earlier highest-fidelity simulations of the problem were performed as spanwise-periodic DNS with a relatively narrow span of $0.04L$ for $Re_L = 1$ million [7, 8, 9], and as hybrid DNS-WRLES with a span of $0.08L$ for $Re_L = 2$ million [10]. As will be discussed, thanks to an efficient in-house GPU flow solver, a very high-fidelity simulation of the entire three-dimensional configuration is now feasible with a reasonable INCITE allocation on the Summit system.

To provide some background prior to discussing the proposed work details, our first spanwise-periodic DNS of the problem, performed at $Re_L = 1$ million, revealed evidence of flow relaminarization in the vicinity of the bump apex [7], which was also observed in the simulations of other groups [8, 9] at the same Re_L. This relaminarization is due to the relatively low Reynolds number of the incoming flow, which does not allow the turbulence to fully survive the acceleration caused by the favorable pressure gradient, as well as the additional stabilizing effect of convex surface curvature. In practical applications, the Reynolds number is generally high enough to preclude relaminarization due to flow acceleration. At present, such very high Reynolds number problems can only be computed in a reasonable time frame using lower-fidelity tools such as RANS and WMLES. Turbulence and wall models embedded within those tools generally cannot detect relaminarization. Hence, in order to truly test and improve the turbulence models used in lower-fidelity simulations tools for high Reynolds number applications, reliable data from a test case that does not contain any relaminarization, are needed. The $Re_L = 2$ million value chosen for the proposed simulation is high enough to eliminate the relaminarization phenomenon in this problem. A spanwise-periodic hybrid DNS-WRLES of the problem at this higher Re_L value was recently performed for a domain span of $0.08L$ [10]. The relatively narrow span of that simulation was found to constrain the downstream evolution of the large structures generated in the separated flow region, which then limits the usefulness of the data gathered in that region. Furthermore, the three-dimensional experimental model naturally leads to a three-dimensional flow separation with strong associated effects on the flow developing over the leeward side of the bump, as observed in the accompanying experiments. Such effects are obviously missing in a spanwise-periodic simulation. To address these issues and generate a high-fidelity numerical dataset under realistic conditions for the problem at hand, we propose a simulation of the entire three-dimensional experimental model, the further details of which are discussed in the next section.

In addition to recent related work performed for the problem of interest [7, 10], the proposed work builds upon our prior experience and knowledge gained from large-scale turbulence simulations made possible by two previous awards under the 2016 and 2017 ALCC programs. Thanks to those awards, turbulence simulations in the form of WRLES on grids containing as many as 24 billion points were performed for the transonic-shock induced flow separation problem over an axisymmetric bump [11], which bears certain similarities to the present problem of interest.

1.2 Relevance to the Department of Energy Mission

The proposed work has a strong potential for generating a detailed dataset that can guide the development of new and improved models for use in lower-fidelity simulations tools such as WMLES and RANS calculations. Improved turbulence models with better accuracy will enable simulations of practical high Reynolds number problems with significantly reduced computational cost. Correct predictions of flows subjected to pressure gradients in turn can be used to improve aerodynamic designs and reduce drag. Drag reduction on aerodynamic surfaces means reduced fuel consumption. These improvements can ultimately pave the way towards the development of aerodynamic designs with reduced overall drag and hence aircraft fuel burn reduction. In other words, these scientific advances can provide the critical knowledge and design tools needed in the development of fuel-efficient air transportation technologies in which aerodynamic drag reduction offers significant performance, energy-savings and carbon sequestration benefits. Simulation techniques of varying levels of fidelity such as RANS, hybrid RANS-LES, WMLES and WRLES, whose embedded turbulence models can be further improved using the data generated by high-resolution simulations, have direct relevance to the wind energy program of the Department of Energy (DOE) as well. The proposed research is therefore in the energy-efficiency area that is closely related to the DOE mission.

1.3 Anticipated Results and Relation to Other Projects

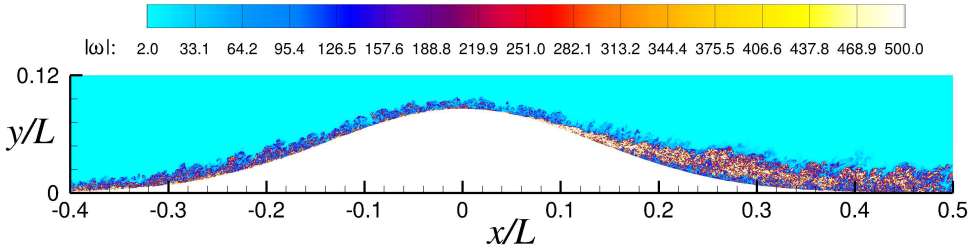
The computational database generated by the proposed work will be useful for gaining additional physical insight into the development of turbulent flows under favorable/adverse pressure gradient conditions. Additionally, the database will be made available to interested researchers and will constitute a unique asset for the turbulence research community. This dataset will help develop improved wall models for complex flows and assess the performance of lower-fidelity (such as RANS, WMLES or hybrid RANS-LES) computational tools. These lower fidelity, but improved, tools are needed for aircraft high-lift prediction required to reduce flight certification test points thus enabling certification by analysis and saving hundreds of million dollars from aircraft development programs. The emerging data-driven techniques for constructing lower fidelity models [12] would benefit from our data as well.

2 RESEARCH OBJECTIVES AND MILESTONES

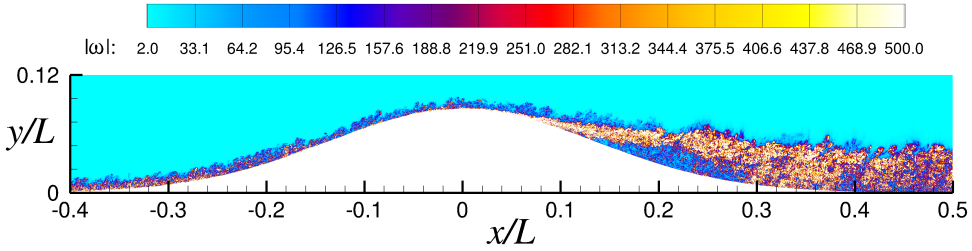
2.1 Goals

As noted earlier, the problem of interest for the proposed effort involves the turbulent flow passing over the speed bump geometry shown in Figure 1. Figure 2 depicts instantaneous snapshots of the turbulent flowfield on an $x-y$ plane, in terms of normalized vorticity magnitude contours (plotted over an exponential scale) for both $Re_L = 1$ and 2 million, extracted from our spanwise-periodic calculations. The flow visualizations show that the flow past the apex in the $Re_L = 1$ million case experiences very weak separation relative to that observed in the higher Re_L case. As discussed in Uzun and Malik [7], the time-averaged statistics for the lower Re_L case verify the incipient or very weak separation in the range where $0.195 \lesssim x/L \lesssim 0.268$. The separated flow in the $Re_L = 2$ million case is evident in the form of the shear layer represented by high levels of vorticity, and the reversed flow region beneath this shear layer, which contains relatively lower vorticity levels. Figure 2 clearly demonstrates that the chosen $Re_L = 2$ million case is indeed successful in generating stronger flow separation in the adverse pressure gradient region. Another important difference between the two cases is the tendency toward relaminarization or stabilization very near the wall upstream of the apex in the lower Re_L case, which is not clearly visible in the shown vorticity snapshots. The absence of relaminarization/stabilization in the higher Re_L case is verified by examining the flow structures very close to the wall in the accelerating region and through further analysis [10].

The subsequent reattachment of the separated flow further downstream in the $Re_L = 2$ million case generates a thicker recovery boundary layer compared to that in the lower Re_L case. The initial boundary layer thickness in the recovery region of the $Re_L = 2$ million case is comparable to the chosen periodic domain span of $0.08L$, and grows further as the recovery flow evolves. Although a wider span can help overcome the constraint imposed by the narrow span on the bump leeward flow evolution, such a spanwise-periodic calculation still cannot capture the three-dimensional nature of the flow separation in the experiment, as depicted by the surface oil flow from the work of Gray et al. [5] in Figure 3. These observations provide the justification for a high-fidelity simulation of the entire three-dimensional model. Experiments are necessary and useful, but they provide limited information. The proposed simulation will provide much more detailed data than the experiment, which will be useful for gaining additional insight into the flow physics and also enable a much more thorough evaluation of lower-fidelity simulations performed under similar conditions.



(a) $Re_L = 1$ million



(b) $Re_L = 2$ million

Figure 2. Instantaneous normalized total vorticity magnitude contours on an $x - y$ plane.

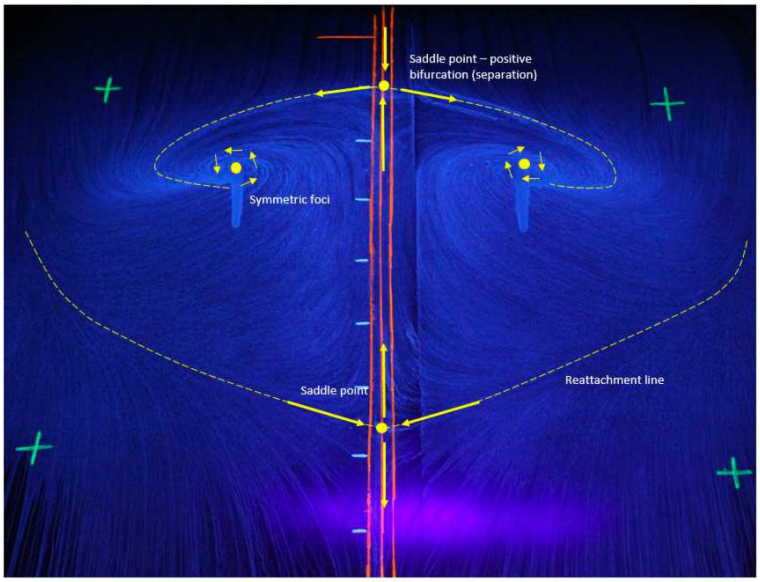


Figure 3. Surface oil flow depicting the separation topography downstream of the bump apex [5].

Our main goal here is therefore to perform a hybrid DNS-WRLES of the three-dimensional experimental speed bump configuration at $Re_L = 2$ million. The equation describing the full three-dimensional speed bump geometry [2] is given by

$$y(x,z) = \frac{h}{2}\left[1 + \mathrm{erf}\left(\left(\frac{L}{2} - 2z_0 - |z|\right)/z_0\right)\right]\exp\left(-(x/x_0)^2\right) \tag{1}$$

where x, y, z respectively denote the axial, vertical and spanwise directions, L is the cross-section width, $h = 0.085L$ (bump height), $x_0 = 0.195L$ and $z_0 = 0.06L$. The planned simulation is estimated to require about 160 billion grid points and is feasible with a reasonable INCITE allocation. The simulation will run as a DNS over the entire attached region and early part of the separated region, and then will transition to WRLES afterwards. The WRLES will be in the form of an implicit LES (ILES), which treats the numerical dissipation of the discretization scheme as an implicit SGS model. In other words, there is no explicit SGS model for the LES; hence, the computational costs of the DNS and the ILES are identical, and it is the local grid resolution that determines whether the code runs in DNS or ILES mode. Further justification for these choices is provided in the next subsection. For the fourth-order accurate GPU flow solver to be used in this study, the grid resolutions needed to ensure DNS are guided by the grid resolutions commonly used with spectral methods for DNS; we aim for similar resolutions with our solver. For attached flows, the local wall skin friction, normally available from a RANS calculation, is used to determine and adjust the grid spacings in wall units. For separated flows, criteria related to the number of points across the shear layer and the number of points covering the separation bubble along the three directions provide the guidance for the grid spacing requirements. In the ILES region, we estimate that the grid resolutions could be about 2–3 times coarser than those corresponding to DNS, except along the wall-normal direction near a viscous wall. In that region, the wall-normal resolution of ILES would be similar to that of DNS.

As discussed in the next subsection, our previous findings reveal a thin internal layer that develops beneath the incoming boundary layer subjected to strong acceleration upstream of the bump apex, and there is a strong connection between this internal layer and the detached shear layer found in the adverse pressure gradient region. It is therefore crucial to capture the correct development of the internal layer and have the proper initial conditions for the separated flow. Hence, DNS is chosen upstream of flow separation to ensure the correct flow development until the early stages of separation. Limited resources then necessitate a switch to WRLES or ILES for the remaining portion of the flowfield, which includes the rest of the separated flow and the recovery region downstream of separation. A similar hybrid DNS-WRLES strategy was previously employed in the spanwise-periodic calculation at $Re_L = 2$ million, and provided acceptable results under the limitations imposed by that particular computational setup, as will be seen.

Figure 4 depicts a schematic of the computational domain for the previous spanwise-periodic simulation performed for $Re_L = 2$ million with a span of $0.08L$ [10]. For that case, the outer boundary in the vertical direction was placed at $y/L = 1$, on which a nonreflecting characteristic boundary condition was applied. The computational domain schematic for the proposed fully three-dimensional simulation is similar to that shown in Figure 4, except for the modification of the top boundary to represent the tunnel ceiling at $y/L = 0.5$, and the inclusion of the tunnel side walls at $z/L = \pm 0.5$, as depicted in Figure 1(b). To reduce the computational cost, the ceiling and the side walls of the tunnel will be modeled as inviscid walls with added displacement thickness distributions obtained from a much cheaper RANS calculation of the same configuration. The displacement thickness distribution on the side walls will depend on x and y, while that on the ceiling will depend on x and z. The inflow boundary of the domain is at $x/L = -0.8$ while the outflow boundary is at $x/L = 2$. The physical domain ends at $x/L = 1$. The region from $x/L = 1$ to 2 forms the sponge zone, in which rapid grid stretching is applied along the streamwise direction. This zone contains only a few hundred points along the streamwise direction because of the significant grid stretching applied.

The sponge zone dampens the turbulence in the flowfield before it reaches the outflow boundary, where standard characteristic outflow boundary conditions are applied. Viscous isothermal boundary conditions are imposed on the lower boundary, which contains the speed bump profile. The uniform wall temperature is set the same as the reference freestream value.

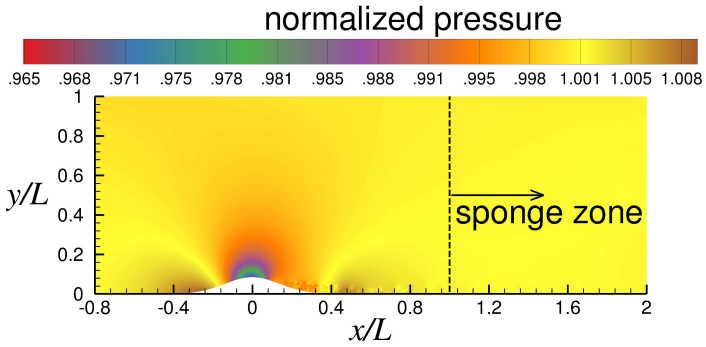


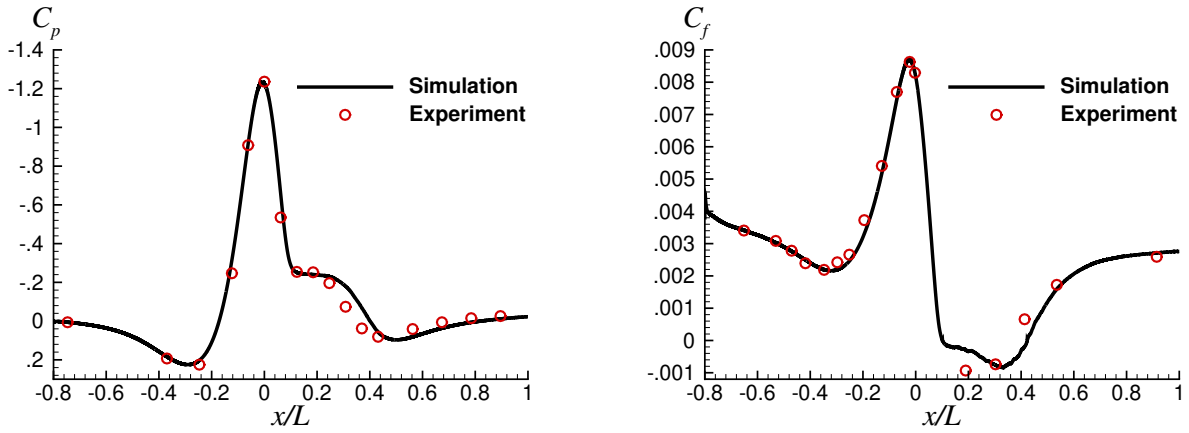
Figure 4. Domain schematic. Contours denote instantaneous pressure normalized by reference value.

For the spanwise-periodic case, the momentum-thickness Reynolds number of the incoming boundary layer is $Re_\theta \approx 1035$. For the three-dimensional case, we expect a similar value on the mid-span plane; however, the mean inflow Re_θ may vary along the span. The span-dependent mean flow to be imposed on the inflow boundary will be taken from the RANS calculation at the same Re_L. The rescaling-recycling technique, discussed in Uzun and Malik [13, 11, 10], will be modified for the proposed simulation. The inflow generation technique used in the spanwise-periodic case assumes a homogeneous mean flow along the span and hence performs spanwise averaging in combination with a moving time average to determine the mean boundary layer thickness and the mean wall friction velocity at the recycle station. The method only recycles the turbulent fluctuations while keeping the mean inflow profile fixed. It will be revised to allow the mean boundary layer thickness and the mean wall friction velocity at the recycle plane to vary along the span; these will be based on moving local time averages. The proposed simulation requires 160 billion points and will be performed using a GPU flow solver based on high-order explicit finite-difference and explicit time integration schemes. This flow solver was validated for a turbulent channel flow during the course of its development, as discussed further in section 3.4. The same code was also used to study the current problem of interest at $Re_L = 1$ million [7]. Further details of the flow solver are provided in section 3.2.

2.1.1 Previous Results from the Spanwise-Periodic $Re_L = 2$ Million Simulation

We now present a brief summary of our previous findings, which should be useful for providing justification for the choices made for the proposed simulation. Our first simulation of this test case was performed at $Re_L = 1$ million for a span of $0.04L$, using our GPU flow solver [7]. For the $Re_L = 2$ million simulation, we did not have access to a sufficient number of GPUs for the required grid. Therefore, the $Re_L = 2$ million simulation was performed using our CPU flow solver [10], which is also fourth-order accurate. Because of limited resources, the $Re_L = 2$ simulation is a hybrid DNS-WRLES with a span of $0.08L$, in which the WRLES is performed in the form of ILES. The domain is discretized using 15360 points along the streamwise direction, 384 points in the vertical direction and 1728 uniform points along the spanwise direction. Statistical data are gathered over $11L/U_\infty$, which covers 6.1 physical domain flow-through times. For the turbulent inflow generation, we employ the rescaling-recycling technique discussed in Uzun and Malik [13, 11, 10]. The mean flow imposed at the inflow boundary is taken from a RANS calculation performed with the low-Reynolds-number correction version of the Spalart-Allmaras model [14]. The mean inflow boundary-layer thickness at $x/L = -0.8$ is $\delta_{in} \approx 0.0055L$, giving $\delta_{in}/h \approx 0.065$. The corresponding inflow momentum-thickness Reynolds number is $Re_\theta \approx 1035$.

Figure 5 plots the variation of the surface pressure and skin-friction coefficients, which are given by $C_p = (p-p_\infty)/(0.5\rho_\infty U_\infty^2)$ and $C_f = \tau_w/(0.5\rho_\infty U_\infty^2)$, where ρ_∞, p_∞, U_∞, respectively, are the reference freestream density, pressure and velocity, p is the mean surface pressure and τ_w is the mean wall shear stress. The incoming boundary layer encounters an initially mild adverse pressure gradient. The adverse pressure gradient becomes progressively stronger as the flow approaches the bump foot, and causes the associated decrease in C_f in the upstream region. The pressure gradient becomes strongly favorable starting at $x/L \approx -0.29$ until very near the bump apex. The flow acceleration increases the C_f because of the steepening near-wall velocity gradient. The C_f peak is reached at $x/L \approx -0.024$. The pressure gradient becomes adverse immediately after the apex, and slows down the flow. The deceleration leads to strong separation at about $x/L = 0.1$, as discussed earlier during the examination of Figure 2. The flow separation is also verified by the negative C_f distribution, which indicates a fairly broad reversed flow region. The reattachment is at $x/L = 0.42$.



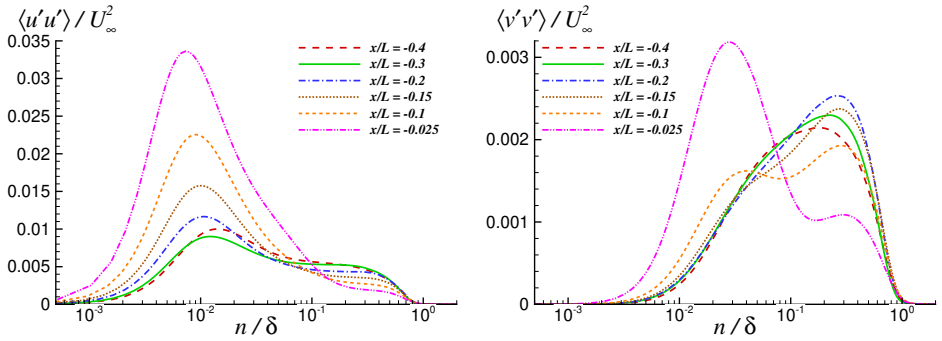
(a) C_p distributions (b) C_f distributions

Figure 5. Surface pressure coefficient, C_p, and skin-friction coefficient, C_f, distributions.

The C_p distribution is compared with the data taken on the centerline of the experiment by Williams et al. [4] at $Re_L \approx 1.98$ million. The C_f distribution is compared with the measurements available from a separate experiment conducted by Gray et al. [6] at $Re_L = 2$ million. The simulation result shows very good agreement with the experimental C_f measurement along the bump centerline over the entire attached region. Some C_f differences are found in the separation and recovery regions, which are not totally surprising. The three-dimensional effects of the experimental configuration, which are not duplicated in the present spanwise-periodic simulation, are expected to have more impact on the separation and recovery regions of the flow. The simulation C_p distribution also shows reasonable overall agreement with the experimental data from Williams et al. [4]. The plateau observed in the C_p distribution downstream of the apex is caused by the flow separation. The pressure rise observed after the plateau appears slightly delayed in the simulation relative to the experiment, which suggests that the reattachment location in the simulation is delayed relative to the experiment. This difference is not a surprise given the fact that no attempt was made to model the three-dimensional configuration of the experiment.

The evolution of the Reynolds stress components, scaled by U_∞^2, from $x/L = -0.4$ to just upstream of the apex for the $Re_L = 2$ million flow is shown in Figure 6. Here, $\langle u'u' \rangle$, $\langle v'v' \rangle$, $\langle w'w' \rangle$ and $\langle u'v' \rangle$, respectively, are the streamwise, wall-normal, spanwise and shear components of Reynolds stress in the local orthogonal system at a given station, and the $\langle\ \rangle$ operator denotes averaging in time and along the span. The wall-normal distance, n, is normalized by the local boundary layer thickness, δ. The first station at $x/L = -0.4$ is positioned slightly upstream of the acceleration region, while the last station at $x/L = -0.025$ is close to

where the C_f peak is found. An internal layer is triggered by the switch from the mild adverse to strong favorable pressure gradient at the foot of the bump, at $x/L \approx -0.29$. The formation of this internal layer is signaled by the formation of knee points in the streamwise and spanwise Reynolds stress profiles. The original peaks of the streamwise and spanwise components of the Reynolds stress are already quite close to the wall; thus, they become engulfed within this internal layer, while those of the wall-normal and shear components are located further away from the wall and happen to lie outside this layer. Consequently, we observe that the original peaks in the streamwise and spanwise stress profiles strengthen considerably as the internal layer develops further within the accelerating flow. We also see the emergence of inner peaks in the other two Reynolds stress components within the internal layer in the later stages of the acceleration. The additional analysis in Uzun and Malik [10] suggests a strong connection between this internal layer, and the free shear layer that develops in the deceleration region and separates. As noted earlier, the use of DNS in the attached region will ensure that the development of this thin internal layer is properly captured in the proposed fully three-dimensional simulation.



(a) Streamwise component (b) Wall-normal component

(c) Spanwise component (d) Shear component

Figure 6. Evolution of the Reynolds stress profiles in spanwise-periodic $Re_L = 2$ million simulation.

2.2 Strategy

The justification for the hybrid DNS-WRLES strategy chosen for the simulation is given in the previous subsection. As noted earlier, for the spanwise-periodic case, a similar strategy required about 10.2 billion points for a span of $0.08L$. A simple extrapolation of this narrow slice to the full span of L gives 127.5 billion points. The grid point distribution will be optimized using the data from the RANS calculations of the same configuration. We plan to use the FUN3D (**https://fun3d.larc.nasa.gov/index.html**) solver for the RANS to be performed with models such as Spalart-Allmaras and Menter Shear Stress Transport. The RANS data will also be used to extract the boundary layer displacement thicknesses on the ceiling and side walls, which will be useful for the inviscid wall boundary conditions to be imposed on those walls, as noted

earlier. Our project collaborator, Dr. Prahladh Iyer, is in the process of studying the same configuration using WMLES with another flow solver. As the requested resources are sufficient to perform only one hybrid DNS-WRLES, we will make use of the RANS and WMLES data to optimize our grid as best as possible. We also plan to introduce a slight grid refinement along the streamwise and wall-normal directions within the separated region, to enable better resolution of relatively thick wake found in the leeward side of the bump, as verified by the PIV measurements from the experiments [4, 6]. This grid refinement is estimated to increase the total grid point count by about 25%, which amounts to 160 billion points.

The simulation grid will be generated in several stages. In the first stage, an initial grid containing 10 billion grid points will be constructed using the Pointwise software (https://www.pointwise.com/), for one half of the three-dimensional geometry. This grid will be optimized as best as possible with the guidance from the available RANS and WMLES data. In the second stage, the initial grid will be refined by a factor of two along each spatial direction using a parallel grid refinement utility. The speed bump is defined by Equation (1); the grid refinement utility will ensure that the new points added onto the surface are located precisely on the surface. This refinement will generate a second grid containing 80 billion points for one half of the geometry. In the final stage, the second grid will be mirrored across the mid-span plane to generate the full grid, which will be partitioned for parallel computing. Our in-house utilities will be used for partitioning the grid and generating the necessary input for the flow solver. The justification for the number of Summit node-hours needed for the simulation is provided in section 3.1. This is a one-year project, whose main milestone is the completion of the simulation and full analysis of the simulation data by the end of the year.

2.3 Dissemination of Results

The findings from this project will be disseminated through publication in archival journals and presentations at established national meetings. Meetings attended will include major conferences in fluid dynamics and aerospace sciences. The detailed dataset gathered from the simulation will be shared with interested researchers. The simulation results will be announced via the NASA turbulence modeling resource website located at: https://turbmodels.larc.nasa.gov

3 COMPUTATIONAL READINESS

The proposed simulation will be performed using our own GPU flow solver and requires 6720 GPUs, or 1120 Summit nodes, for about 160 billion grid points. This calculation is only feasible on the Summit system at the moment. Note that 1120 nodes make up about 24% of the entire system. Thus, the problem size needs significant computational resources that satisfy the requirements of an INCITE award.

3.1 Use of Resources Requested

Resources are requested to perform a simulation on a 160 billion point grid. For such a grid size, we plan to use 6720 GPUs or 1120 Summit nodes. As will be discussed in section 3.3, in order to make the best use of limited computational resources, the GPU flow solver must be run at maximum efficiency. For that purpose, a large chunk of workload must be assigned to each GPU. The maximum GPU workload can be achieved by choosing a grid block size per GPU that nearly or fully maxes out the available GPU memory. Each GPU on Summit has 16 Gigabytes of global memory. For the given grid size, 1120 Summit nodes would be needed in order to allow the flow solver to run in its maximum efficiency mode.

To emulate the low-speed experimental flow conditions, the upstream Mach number will be set to 0.2 for the compressible flow solver. The time step is determined by the smallest grid size in the domain, which is found adjacent to the wall at around the bump apex as the wall skin friction is highest there. With a maximum

Courant-Friedrichs-Lewy (CFL) number of about 0.8, 1.12 million time steps will be needed to compute one time unit, L/U_∞, using the three-stage explicit Runge-Kutta scheme [15]. Based on the code execution timing on Summit, about 46.7 hours are needed to compute one time unit. The simulation is planned to be performed for a total time interval of $10.8L/U_\infty$. The physical domain size is $1.8L$ in the streamwise direction. The first $3.6L/U_\infty$ interval of the simulation will be used to drive the initial numerical transients out of the computational domain. This corresponds to two domain flow-through times. The remaining interval will be used to gather statistical data. A statistical sample over $7.2L/U_\infty$ would correspond to 4 domain flow-through times. The total run time needed is : $46.7 \times 10.8 \approx 504.36$ hours on 1120 Summit nodes. This corresponds to : $1120 \times 504.36 \approx 0.565$ million Summit node-hours.

Restart files will be saved frequently during the simulation, typically once every hour. We plan to back up a number of restart file dumps at evenly spaced intervals. For the given problem size, the size of a single restart file dump is about 45 Terabytes. The restart files to be backed up will be moved from scratch disk to archival storage. Much of the computation of the statistical data, such as time-averaged mean flow, Reynolds stresses, and quantities needed for the Reynolds stress budget analyses will be performed on the fly while the simulation runs. We plan to save the time history of the unsteady flowfield data on a number of planes in the critical regions of the flowfield for further analysis. We estimate to use up to about 500 Terabytes of space on the scratch disk during the simulation and post-processing phases, and about 1000 Terabytes of archival storage. After the project ends, the data stored in archival storage will be transferred to the mass storage system at the NASA Advanced Supercomputing Division.

3.2 Computational Approach

A direct numerical simulation (DNS) code, named GTHORS (GPU version of Turbulence with High-Order Resolution Solver) has been specifically developed for Graphics Processing Units (GPUs). The code automatically switches to ILES mode when the local grid resolution cannot satisfy the DNS requirement. This code is based on explicit high-order finite-difference and explicit time integration schemes. We believe such schemes have the best potential to extract the full capability of a GPU since they are essentially made up of many independent multiply-and-add type operations, at which the GPU architecture excels. The code solves the compressible Navier-Stokes equations discretized on multiblock structured grids. The methodology employs the optimized fourth-order accurate central explicit finite difference scheme with a 13-point stencil, developed by Bogey and Bailly [16], to compute all spatial derivatives in the governing equations. For boundary and near-boundary points, matching one-sided and biased schemes, developed by Berland et al. [17], are used. To ensure numerical stability, the optimized 11-point sixth-order explicit selective filter, developed by Bogey and Bailly [18], is used. This optimized filter is derived from the standard explicit tenth-order filter. The optimized filter modifies the coefficients of the standard filter in order to improve the filter damping function in wavenumber space and constrain the numerical dissipation to waves discretized by fewer than four grid points. The frequency of filtering (in terms of time steps) and the amount of damping applied by the filter are fine tuned depending on the specific grid resolution used in a given problem. This is done in order to set the minimal amount of numerical dissipation needed for stability.

Time advancement can be performed using either a third-order, three-stage explicit Runge-Kutta scheme [15] or a second-order, single-stage explicit scheme developed by Verstappen and Veldman [19]. This second-order scheme is a modified Adams-Bashforth scheme with improved stability properties. The second-order scheme runs at half the time step of the third-order Runge-Kutta scheme but requires only one right-hand-side computation per time step as opposed to three right-hand-side evaluations needed with the Runge-Kutta scheme. It therefore provides a speedup factor of 3/2 over the Runge-Kutta scheme. The flow solver employs three levels of parallelism based on a hybrid combination of MPI+OpenMP+CUDA Fortran. It

also implements strategies that overlap communication with computation, which are crucial for achieving good performance. Except for the routines that handle data input and output to and from the disk, the entire code runs on the GPU. All data arrays are therefore stored in the global GPU memory. This eliminates extensive back-and-forth data movement between the GPU memory and the host CPU memory.

3.2.1 Processing of Input and Output Data

A pre-processing utility partitions the computational grid for parallel processing prior to beginning the simulation. Each MPI task reads in its own local grid from a separate file. Similarly, each MPI task writes its local flow solution or restart file into a separate output file. Our local workstations will be used for grid generation and generation of initial conditions for the simulation. The input files will be transferred over the internet using the secure file transfer protocol. As noted earlier, much of the statistical data computation will be performed on the fly while the simulation runs. Any data that needs further processing will be transferred back over the internet to our local computing cluster. A parallel post-processing utility can read in the data from the individual output files for further analysis. Grid and flow solution files are commonly saved in unformatted multiblock PLOT3D data format and are visualized using the Tecplot software.

3.2.2 Data Management Plan

As noted in section 3.1, a number of restart file dumps as well as the time history of the unsteady flowfield on a number of planes will be saved and backed up on the archival storage system. These datasets will be moved to the mass storage system at the NASA Advanced Supercomputing Division after the project ends. We will use the secure file transfer protocol over the internet for the data transfer. These datasets will be made available to interested researchers. A typical dataset is expected to include but not limited to: unsteady flowfield time history on several planes in critical regions; time-averaged flowfield statistics (mean flow and Reynolds stresses); Reynolds stress budgets and two-point correlations. The datasets will be distributed in common data formats, such as PLOT3D and TECPLOT (http://www.tecplot.com/).

3.3 Parallel Performance

The scaling performance of our GPU code has been tested on Summit, using an allocation made under the OLCF Director's Discretion Projects Program. Each Summit node contains 44 host CPU cores and 6 NVIDIA Tesla V100 GPUs. In our parallelization strategy, we assign 6 MPI tasks per Summit node. Each MPI task gets 1 GPU and 7 cores on the host CPU, and works on one grid block. To overlap GPU computation with MPI communication, each MPI task launches 7 OpenMP threads on the host CPU cores (one thread per core). The master OpenMP thread controls and launches the computational kernels (written in CUDA Fortran) on the GPU. The six remaining helper threads (one per face of a grid block) manage the MPI sends and receives. The master and helper threads synchronize at critical junctures of the algorithm.

The flow solver implements strategies that overlap communication with computation. This overlap is achieved as follows. The GPUs first compute the near-boundary information needed by their neighbors and copy the data to their host CPUs for MPI communication. The GPUs then compute the interior points while the host CPUs handle the MPI communication in parallel. The host CPUs run several OpenMP threads (one dedicated thread per face of a grid block) to perform the MPI data exchange. Assuming a large enough GPU workload, by the time the GPUs have finished computing interior points, the MPI communication among the host CPUs has been completed and the host CPUs have copied the exchanged MPI data back to their corresponding GPUs. The GPUs can then update the points near their block interface using the data received from their neighbors and can proceed further. The data copies between the GPU and the host CPU take place asynchronously, meaning that the GPU can do the computing (as long as there is sufficient work) and the data exchange with the host CPU simultaneously. Moreover, the GPU can run several independent

tasks in parallel, as long as the resources needed by the computational kernels are available. The GPU operations are synchronized at critical points.

During the course of the developmental work, it was realized that the chosen explicit algorithms run extremely fast on the GPU since they are essentially made up of many independent multiply-add operations, at which the GPU excels. However, the host-assisted MPI communication among the GPUs was found to be a major bottleneck, as already known from others' similar experience in GPU code development. We observed that, unless we assign a large workload to the GPU and overlap computation with communication, MPI communication over the network cannot keep up with GPU computation. When the workload per GPU is too small, an inefficiency arises because the GPU finishes its work way too quickly and then sits idle while waiting for the communication to catch up. This idle time is nothing but a wasted opportunity to have the GPU do useful work. Hence, in order to have the GPU run at full efficiency and make the best use of limited computational resources, the present strategy is to assign the maximum possible workload to the GPU and overlap computation with communication as much as possible. To achieve the maximum possible workload, we adjust the grid block size per GPU in order to nearly or fully max out the available GPU global memory, which is 16 Gigabytes per GPU on Summit. This corresponds to about 24-25 million points per GPU.

In our code, the host-assisted MPI communication approach is chosen over the so-called "GPU-aware MPI" option, which accomplishes direct communication among the GPUs without the complication of sending the data through the host CPUs. This choice was made because the performance of the GPU-aware MPI implementation available at the time of code development was found to be rather poor and unacceptable. Although the implementation of host-assisted MPI communication is more involved, assigning a large workload to the GPUs and having the host CPUs handle the MPI communication overlaps computation with communication effectively and hides the communication cost, as noted above. It is therefore well worth the additional complication since it improves the overall code performance.

Because of the slow communication issue relative to computation, we do not expect to see good strong scaling performance on the current systems. Communication simply would not be able to keep up with computation if the GPUs were assigned too little work. This issue can only be resolved by a much faster communication network, which presently does not exist. Hence, the only way to make the best use of limited resources at present is to run the code in its maximum efficiency mode. Although the maximum efficiency mode may not minimize the wall-clock run time, it will certainly minimize the total number of node-hours needed for a given problem size, which is the more relevant performance metric given the fact that the available resources are limited.

A weak scaling study has been performed on the Summit system, in which the total workload per GPU is kept fixed while increasing the total number of GPUs. Our test case for the scaling study involves the flow over a flat plate. This particular test case is chosen because it bears good similarity to the actual problem of interest and it also enables easy adjustment of the problem size in terms of the total number of grid points. All subroutines or kernels that would normally get called during the simulation of the actual problem also get called during this test case as the governing equations that are solved are still the same. Hence, the performance figures obtained from this test case directly apply to the actual problem.

For the weak scaling study runs, the total number of GPUs is varied from 150 (25 nodes) up to 6000 (1000 nodes). Note that 1000 nodes make up 21.7% of the entire system. For each case, we assign a grid block of $256 \times 320 \times 288$ points to each GPU. Test runs for each node-count case were performed to measure the total time taken to run 1000 time steps and compute the corresponding average time per step. The third-order, three-stage explicit Runge-Kutta scheme is used for time advancement. Figure 7 shows how the average

time taken per step, in milliseconds, varies as the number of Summit nodes are increased. Note that the horizontal axis is plotted in logarithmic scale. Depending upon how the nodes assigned to a particular job are distributed within the system and the network route among the nodes, as well as the network traffic due to other jobs running on the system, code execution speeds may vary from one run to another. We therefore see a slight variation in the average time per step as the number of nodes changes. The average time per step is nearly constant and remains in between 149 and 150 milliseconds. This is the payoff for utilizing a strategy that assigns a large workload to the GPU and overlaps GPU computation with MPI communication as much as possible. The "mean" value of the time taken per step (i.e., the value averaged over all node-count runs) is about 149.5 milliseconds for the given grid block size per GPU. This corresponds to roughly one nanosecond per grid point per Summit node per time step.

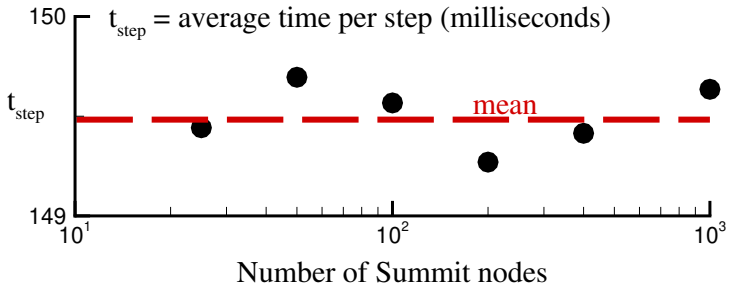


Figure 7. Weak scaling performance on the Summit system.

The overall performance of the GPU code was measured to be 10% of the peak theoretical double-precision performance of the V100 GPU, which is rated at 7.8 Tera-floating-point-operations (TFLOPs) per second. The most compute-intensive individual kernels (or subroutines) were found to achieve as high as 17% of the peak. To determine the performance limiting factor, the code was profiled using the available performance analysis tools. For a grid block of about 24 million points per GPU, it was found that, at each time step, the code moves about 71 Gigabytes worth of total data from the global memory to the registers of the processing units, and about another 39 Gigabytes worth of total computed data from the registers back to the global memory. Thus, there is a data movement of about 110 Gigabytes at each time step, between the global memory and the registers. Now, in order to illustrate what limits the code performance, suppose the code only moves this much data between the main memory and registers, but does not do any computing at all. How long would this data movement alone take? To answer this question, we first note that the V100 GPU architecture provides 900 Gigabytes/second of peak memory bandwidth. For simplicity, let us assume that the data movement takes place at the peak bandwidth. That would mean that at least $110/900 \approx 0.122$ seconds would be needed to move that amount of data. Our profiling measurements show that the most compute-intensive kernels generally achieve about 80 to 90% of the peak memory bandwidth, so the actual data movement takes a bit longer than this estimate. With an overall average rate of 85% of the peak bandwidth, the data movement would take about 0.144 seconds. We also know that for the given grid block size per GPU, the code takes about 0.15 seconds to perform all operations and advance the simulation for one time step. The profiling measurements show that the actual compute time is around 19% of the total elapsed time. Even though the actual computations overlap with the memory operations, the data movement between the memory and registers still constitutes a significant chunk of the time taken per computational time step. These observations lead us to the conclusion that our code performance is bound by the available memory bandwidth. In other words, the memory bandwidth is not sufficient to transfer data into and out of the registers at the rate demanded by the processing units. Thus, the well-known adage of the computer science world, "The FLOPs are free, you are paying for the memory bandwidth!" is very much valid here.

As noted above, with 80 to 90% of the peak memory bandwidth achieved by the most compute-intensive ker-

nels, our memory-bound GPU code is not far off from its maximum possible performance; hence, reaching a much greater percentage of the peak theoretical FLOPs per second performance of the V100 architecture is not feasible. We anticipate that potential memory bandwidth improvements in future-generation GPU architectures should enable our code to achieve higher FLOP counts per second on those systems.

3.3.1 Performance Comparison to CPU Flow Solver

We now provide the performance comparison between this new GPU code and our previous CPU code, which was most recently used in the simulation of flow separation problems [13, 11]. The speedup factors depend on how exactly the comparisons are made. For example, the "node-to-node" speedup factor, which is derived from the performance comparison based on one Summit node with 6 GPUs versus one dual-socket Intel Skylake CPU node with 40 cores, comes out to about 75×. This comparison is for the version of the CPU code based on the third-order, three-stage explicit Runge-Kutta scheme, same as that employed in the GPU code. Note that the CPU code uses high-order compact finite-difference [20] and filtering schemes [21, 22], whose implementations are well-optimized for the CPU. We estimate that the implementation of explicit finite-difference and filtering schemes, employed in the GPU code, into the CPU solver would result in a performance improvement of only about 25%. Even in such a case, the node-to-node speedup factor would still be about 60×. Basing the comparison on the number of GPUs versus the number of CPU cores would lead to an equivalent speedup factor of $60 \times 40/6 = 400\times$, meaning that one GPU is worth 400 CPU cores. We also note that one dual-socket Intel Skylake CPU, which contains 40 cores total, is comparable in price and power consumption to a V100 GPU. If we were to base the comparison on one V100 GPU versus one dual-socket Intel Skylake CPU, the corresponding speedup factor would be $400/40 = 10\times$.

The CPU code also has a second-order implicit time integration version. It is about 2.5 times more costly per time step than the explicit integration scheme. The explicit time integration scheme is normally run at a CFL number of around 0.8. Because of time accuracy concerns, the second-order implicit time integration scheme should not normally be run at a CFL number greater than 5 or so. This corresponds to a factor of about 6.25 increase in the time step with the implicit scheme. The increased computational cost per time step of the implicit scheme gives a speedup factor of about $6.25/2.5 = 2.5$ over the explicit scheme. Thus, comparing the GPU code to the implicit version of CPU code *as is*, we obtain a node-to-node speedup factor of $75/2.5 = 30\times$. The corresponding comparison based on one V100 GPU versus one dual-socket Intel Skylake CPU would yield a speedup factor of $30/6 = 5\times$. As noted earlier, it is possible to further accelerate the GPU code by switching to a second-order, single-stage explicit scheme developed by Verstappen and Veldman [19]. The second-order scheme runs at half the time step of the third-order Runge-Kutta scheme but requires only one right-hand-side computation per time step as opposed to three right-hand-side evaluations needed for the Runge-Kutta scheme. It therefore provides an acceleration factor of $3/2 = 1.5\times$ over the Runge-Kutta scheme. With the second-order explicit time advancement scheme implemented in the GPU code, the performance comparison to the implicit version of CPU code (also second-order accurate in time), based on one V100 GPU versus one dual-socket Intel Skylake CPU, would provide a speedup factor of 7.5×.

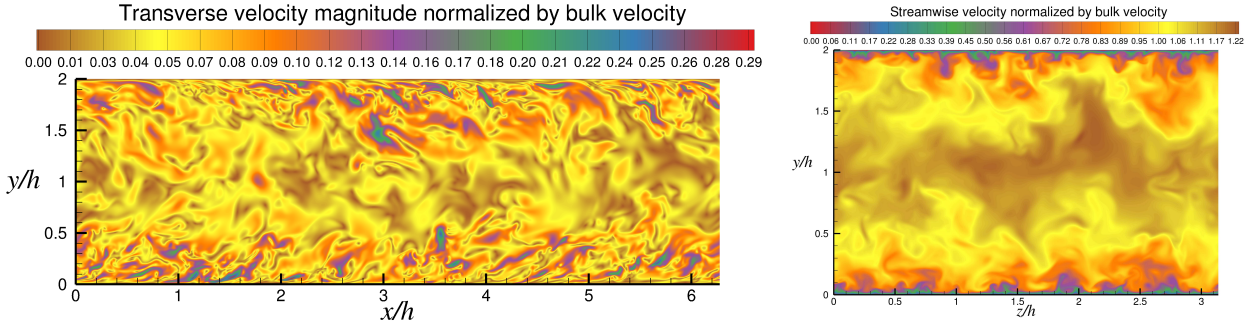
We should note here that a significant effort was also put into optimizing the implicit time integration version of the CPU code discussed above. The explicit time integration version of the CPU code is essentially the conversion from the optimized implicit version. Hence, the above performance comparisons are between the fastest-running versions of the GPU and CPU codes, and are as fair as currently possible.

3.4 Developmental Work and GPU Flow Solver Validation

Our GPU code development started in early 2019 on Summit and was completed by the end of 2019. The code is now ready for production runs. To validate this GPU flow solver, a turbulent channel flow

problem was considered. The Reynolds number of the fully-developed turbulent channel flow is $Re_\tau = \rho_{bulk} u_\tau h/\mu_{wall} = 590$, where ρ_{bulk} is the bulk density, u_τ is the wall friction velocity, h is the channel half-height and μ_{wall} is the viscosity on the wall. The domain size is $2\pi h$ in the streamwise direction, x, $2h$ in the wall-normal direction, y, and πh in the spanwise direction, z. The flow is periodic both in the streamwise and spanwise directions and is bounded by solid walls at $y = 0$ and $2h$. Because of the imposed streamwise periodicity, a source term is added to the streamwise momentum and energy equations to drive the flow at a constant mass flow rate. The Mach number based on bulk velocity and sound speed on the wall is set to 0.2.

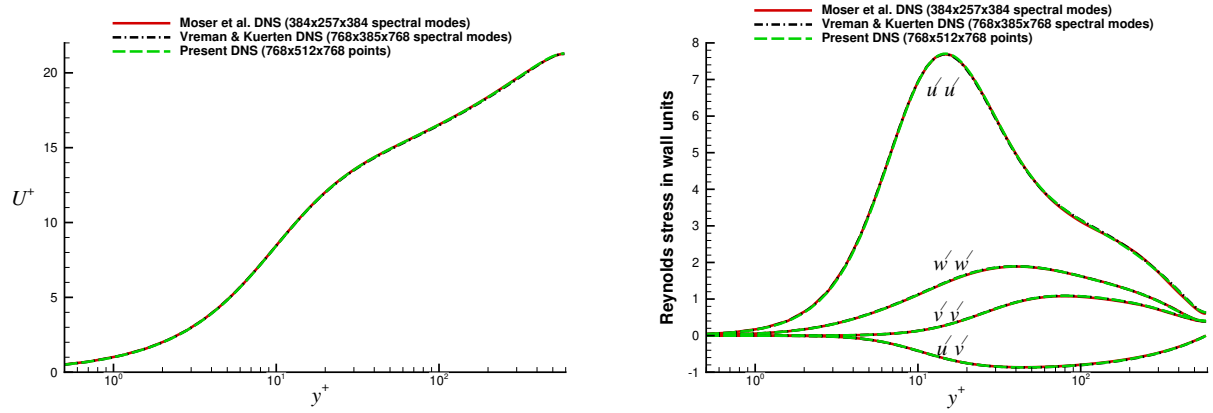
The grid used in the DNS contains $768 \times 512 \times 768$ points along x, y and z directions, respectively. The grid resolution in wall units is $\Delta x^+ \approx 4.8$ in the streamwise direction and $\Delta z^+ \approx 2.4$ in the spanwise direction. In the vertical direction, $\Delta y^+ \approx 0.23$ on the wall and $\Delta y^+ \approx 5.4$ at the channel centerline. For the computation, 18 GPUs are used on Summit. Each GPU solves a grid block of 256^3 points. The second-order time integration scheme [19] is used with a CFL number of 0.4. Explicit filtering [16, 18] is applied at every 5 time steps with a filtering parameter of $\sigma = 0.09$. To ensure full convergence of the time-averaged results, the flow statistics are averaged over $1009h/u_{bulk}$. Figure 8 depicts the instantaneous snapshots of the turbulent channel flow. As seen here, the flow structures appear very smooth and provide evidence that the minimal filtering applied in the present case is able to provide a solution free of any numerical wiggles.



(a) Transverse velocity magnitude contours (b) Streamwise velocity contours

Figure 8. Instantaneous snapshots of turbulent channel flow.

Figure 9 compares our mean streamwise velocity and Reynolds stress component profiles with those from Moser et al. [23] as well as Vreman and Kuerten [24]. These groups used spectral methods to perform an incompressible DNS at the same Re_τ. The comparisons show an excellent agreement with their results.



(a) Mean velocity (b) Reynolds stress components

Figure 9. Turbulent channel flow mean streamwise velocity and Reynolds stresses in wall units.

4 REFERENCES

[1] Spalart, P. R., Belyaev, K. V., Garbaruk, A. V., Shur, M. L., Strelets, M. K. and Travin, A. K., Large-Eddy and Direct Numerical Simulations of the Bachalo-Johnson Flow with Shock-Induced Separation, *Flow, Turbulence and Combustion*, 2017, 99(3–4), 865–885.

[2] Slotnick, J. P., Integrated CFD Validation Experiments for Prediction of Turbulent Separated Flows for Subsonic Transport Aircraft, NATO Science and Technology Organization, Meeting Proceedings RDP, STO-MP-AVT-307, 2019.

[3] Williams, O., Samuell, M., Sarwas, S., Robbins, M. and Ferrante, A., Experimental Study of a CFD Validation Test Case for Turbulent Separated Flows, AIAA Paper 2020-0092, AIAA SciTech 2020 Forum, Orlando, Florida, 2020.

[4] Williams, O., Samuell, M., Robbins, M. L., Annamalai, H. and Ferrante, A., Characterization of Separated Flowfield over Gaussian Speed-Bump CFD Validation Geometry, AIAA Paper 2021-1671, AIAA SciTech 2021 Forum, Virtual Event, 2021.

[5] Gray, P., Gluzman, I., Thomas, F., Corke, T., Lakebrink, M. and Mejia, K., A New Validation Experiment for Smooth-Body Separation, AIAA Paper 2021-2810, AIAA Aviation 2021 Forum, Virtual Event, 2021.

[6] Gray, P., Gluzman, I., Thomas, F., Corke, T. and Mejia, K., Experimental Characterization of Smooth Body Flow Separation Over Wall-Mounted Gaussian Bump, AIAA Paper 2022-1209, AIAA SciTech 2022 Forum, San Diego, CA & Virtual Event, 2022.

[7] Uzun, A. and Malik, M. R., Simulation of a Turbulent Flow Subjected to Favorable and Adverse Pressure Gradients, *Theoretical and Computational Fluid Dynamics*, 2021, 35(3), 293–329.

[8] Balin, R. and Jansen, K. E., Direct Numerical Simulation of a Turbulent Boundary Layer over a Bump with Strong Pressure Gradients, *Journal of Fluid Mechanics*, Vol. 918, A14, 2021.

[9] Shur, M. L., Spalart, P. R., Strelets, M. K. and Travin, A. K., Direct Numerical Simulation of the Two-Dimensional Speed Bump Flow at Increasing Reynolds Numbers, *International Journal of Heat and Fluid Flow*, 2021, 90, 108840.

[10] Uzun, A. and Malik, M. R., High-Fidelity Simulation of Turbulent Flow Past Gaussian Bump, *AIAA Journal*, 2022, 60(4), 2130–2149.

[11] Uzun, A. and Malik, M. R., Wall-Resolved Large-Eddy Simulations of Transonic Shock-Induced Flow Separation, *AIAA Journal*, 2019, 57(5), 1955–1972.

[12] Duraisamy, K. and Singh, A. P., Informing Turbulence Closures With Computational and Experimental Data, AIAA Paper 2016-1556, 54^{th} AIAA Aerospace Sciences Meeting, San Diego, CA, 2016.

[13] Uzun, A. and Malik, M. R., Large-Eddy Simulation of Flow over a Wall-Mounted Hump with Separation and Reattachment, *AIAA Journal*, 2018, 56(2), 715–730.

[14] Spalart, P. R. and Garbaruk, A. V., Correction to the Spalart-Allmaras Turbulence Model, Providing More Accurate Skin Friction, *AIAA Journal*, 2020, 58(5), 1903–1905.

[15] Gottlieb, S., Shu, C.-W. and Tadmor, E., Strong Stability-Preserving High-Order Time Discretization Methods, *SIAM Review*, 2001, 43(1), 89–112.

[16] Bogey, C. and Bailly, C., A Family of Low Dispersive and Low Dissipative Explicit Schemes for Flow and Noise Computations, *Journal of Computational Physics*, 2004, 194(1), 194–214.

[17] Berland, J., Bogey, C., Marsden, O. and Bailly, C., High-Order, Low Dispersive and Low Dissipative Explicit Schemes for Multiple-Scale and Boundary Problems, *Journal of Computational Physics*, 2007, 224(2), 637–662.

[18] Bogey, C. and Bailly, C., A Shock-Capturing Methodology Based on Adaptative Spatial Filtering for High-Order Non-Linear Computations, *Journal of Computational Physics*, 2009, 228(5), 1447–1465.

[19] Verstappen, R. W. C. P. and Veldman, A. E. P., Direct Numerical Simulation of Turbulence at Lower Costs, *Journal of Engineering Mathematics*, 1997, 32(2–3), 143–159.

[20] Ashcroft, G. and Zhang, X., Optimized Prefactored Compact Schemes, *Journal of Computational Physics*, 2003, 190(2), 459–477.

[21] Visbal, M. R. and Gaitonde, D. V., Very High-Order Spatially Implicit Schemes for Computational Acoustics on Curvilinear Meshes, *Journal of Computational Acoustics*, 2001, 9(4), 1259–1286.

[22] Gaitonde, D. V. and Visbal, M. R., Padé-Type Higher-Order Boundary Filters for the Navier-Stokes Equations, *AIAA Journal*, 2000, 38(11), 2103–2112.

[23] Moser, R. D., Kim, J. and Mansour, N. N., Direct Numerical Simulation of Turbulent Channel Flow up to $Re_\tau = 590$, *Physics of Fluids*, 1999, 11(4), 943–945.

[24] Vreman, A. W. and Kuerten, J. G. M., Statistics of Spatial Derivatives of Velocity and Pressure in Turbulent Channel Flow, *Physics of Fluids*, 2014, 26(8), 085103–1/29.

PERSONNEL JUSTIFICATION AND MANAGEMENT PLAN

PERSONNEL JUSTIFICATION

The PI, Dr. Ali Uzun is currently an Associate Principal Engineer at the National Institute of Aerospace. Previously, he was at the Florida State University where he conducted research in large-scale turbulence simulations, computational aeroacoustics, hydrodynamic stability analysis and parallel computing. He is the developer of the GPU flow solver referenced in the proposal. He has previously performed numerous large-scale simulations for a diverse range of fluid dynamics problems. In particular, his jet noise simulations represented some of the largest jet noise and computational aeroacoustics calculations ever performed at the time of their publication. His recent focus has been on wall-resolved simulations of separated flows and flows subjected to pressure gradients. He has conducted a number of large simulations on grids containing as many as 24 billion points using the 2016 and 2017 ALCC program awards.

The Co-PI, Dr. Mujeeb R. Malik is currently the Senior Aerodynamicist (ST) at NASA's Langley Research Center (LaRC). Previously, he has served as Head of the Computational AeroSciences Branch at NASA, where he led research in the development of advanced Computational Fluid Dynamics (CFD) methods over a wide speed regime, from subsonic to hypersonic, transition and turbulence modeling and airframe noise source modeling and control. Dr. Malik is an internationally recognized expert in the field of boundary layer stability, laminar-turbulent transition and flow control and has over 150 publications in these areas. He is the Technical Lead for Revolutionary Computational Aerosciences, which is aimed at advancing the state-of-the-art in computational methods, transitional and turbulent flow modeling, eddy resolving simulations and flow physics experiments. He sponsored NASA's CFD Vision 2030 study, which emphasized research in high performance computing and large-eddy simulations to solve the grand challenge problems in aerospace vehicle design. He is a Fellow of APS, AIAA and ASME.

The project collaborator, Dr. Prahladh Iyer is currently a Senior Research Scientist at the National Institute of Aerospace. His ongoing research focuses on the wall-modeled large-eddy simulations (WMLES) and RANS calculations of complex flows involving flow separation, as well as near-wall modeling for WMLES.

MANAGEMENT PLAN

Dr. Uzun will serve as the PI who will oversee the project progress. He will also be the main point of contact to provide updates on the status of the work including publications, awards, and highlights of accomplishments. Dr. Uzun will be in charge of grid generation, performing the proposed simulation and analyzing the results. He has extensive experience in large-scale simulations and he is also the developer of the GPU flow solver referenced in the proposal. He will have access to the entire allocation to perform the proposed simulation. Drs. Malik and Iyer will assist with the analysis and interpretation of the simulation results. Dr. Iyer will perform the RANS calculations that will generate the necessary information regarding the mean inflow state and tunnel wall displacement thickness distributions for the proposed simulation. He will also assist with the grid generation for the proposed simulation. Drs. Uzun, Malik and Iyer have offices within the Computational AeroSciences Branch of the NASA Langley Research Center. They will frequently interact during this project to discuss and evaluate the findings.

MILESTONE TABLE

Proposal Title: High-Fidelity Turbulence Simulation of Three-Dimensional Complex Flow Separation

Year 1		
Milestone	**Details**	**Dates**
Completion of the three-dimensional speed bump simulation on 160 billion grid points and analysis of the simulation results	**Resource:** Summit **Node hours:** 0.565 million **Filesystem storage (TB and dates):** 500TB (1/1/23 – 12/31/23) **Archival storage (TB and dates):** 1000TB (1/1/23 – 12/31/23) **Software Application:** GTHORS (in-house GPU flow solver) **Tasks:** a) Determine the mean inflow state and displacement thickness distributions on side walls and tunnel ceiling from the RANS calculation of the same configuration b) Generate the grid for the simulation c) Perform the simulation, analyze the results, compare the results against the experimental data **Dependencies:** None	1/1/23 – 12/31/23

PUBLICATIONS RESULTING FROM INCITE AWARDS

We do not have any publications resulting from INCITE awards because we have not received any previous INCITE awards.

BIOGRAPHICAL SKETCHES

Curriculum Vitae
PI, Dr. Ali Uzun
ali.uzun@nianet.org
ali.uzun@nasa.gov
757-864-8798

Professional Preparation

Ph.D. in Aeronautics & Astronautics, Purdue University, Indiana, 2003.
Master of Science in Mechanical Engineering, Purdue University, Indiana, 1999.
Bachelor of Science in Aeronautical Engineering, Middle East Technical University, Turkey, 1997.

Appointments

December 2019 – present, Associate Principal Engineer, National Institute of Aerospace, Hampton, VA.
July 2015 – December 2019, Senior Research Scientist, National Institute of Aerospace, Hampton, VA.
December 2010 – July 2015, Assistant Scholar Scientist, Florida Center for Advanced Aero-Propulsion, Florida State University, Tallahassee, FL.
December 2003 – December 2010, Research Associate, Florida State University, Tallahassee, FL.

Five Publications Most Relevant to This Proposal

1. A. Uzun and M. R. Malik, "High-Fidelity Simulation of Turbulent Flow Past a Gaussian Bump," *AIAA Journal*, 60(4): 2130-2149, 2022.
2. A. Uzun and M. R. Malik, "Simulation of a Turbulent Flow Subjected to Favorable and Adverse Pressure Gradients," *Theoretical and Computational Fluid Dynamics*, 35(3): 293-329, 2021.
3. A. Uzun and M. R. Malik, "Effect of Spatial Filtering in Implicit Large-Eddy Simulations of Separated Flows," *AIAA Journal*, 57(12): 5575-5580, 2019.
4. A. Uzun and M. R. Malik, "Wall-Resolved Large-Eddy Simulations of Transonic Shock-Induced Flow Separation," *AIAA Journal*, 57(5): 1955-1972, 2019.
5. A. Uzun and M. R. Malik, "Large-Eddy Simulation of Flow over a Wall-Mounted Hump with Separation and Reattachment," *AIAA Journal*, 56(2): 715-730, 2018.

Research Interests and Expertise

Dr. Ali Uzun is currently an Associate Principal Engineer at the National Institute of Aerospace. Previously, he was at the Florida State University where he conducted research in large-scale turbulence simulations, computational aeroacoustics, hydrodynamic stability analysis and parallel computing. He is the developer of the GPU flow solver referenced in the proposal. He has previously performed numerous large-scale simulations for a diverse range of fluid dynamics problems. In particular, his jet noise simulations represented some of the largest jet noise and computational aeroacoustics calculations ever performed at the time of their publication. Dr. Uzun collaborates with NASA Langley researchers on problems related to boundary layer transition and complex turbulent flows. His recent focus has been on

wall-resolved simulations of separated flows and flows subjected to favorable/adverse pressure gradients. He has conducted a number of large simulations using the 2016 and 2017 ALCC program awards.

Synergistic Activities

1. Dr. Uzun performed cutting-edge research in two NASA-funded projects and collaborated with NASA researchers. These projects were concerned with jet engine noise and airframe noise. He developed a unique turbulence simulation code based on high-order schemes and performed large-scale turbulence and aeroacoustics simulations in these projects.
2. He collaborated with an experimental research group at the Florida State University (FSU) led by Professor Farrukh Alvi, on supersonic impinging jets and on the simulation and optimization of micro-actuators, which generate unsteady pulsed microjets for high-speed flow and noise control applications. The simulations performed in this study complemented the experimental work.
3. He collaborated with the same experimental group at FSU on a computational-theoretical-experimental approach towards more effective flow control in problems related to shock wave – turbulent boundary layer interaction and high-speed jet noise reduction.
4. He served as a reviewer for *AIAA Journal, Journal of Aircraft, Theoretical and Computational Fluid Dynamics, Physics of Fluids, Journal of Fluid Mechanics, International Journal for Numerical Methods in Fluids, Progress in Computational Fluid Dynamics, Journal of Propulsion and Power, Journal of Sound and Vibration, Computers and Fluids* and *ASME Journal of Fluids Engineering*.
5. He served as a Science Advisory Board Member for the joint NSF-TeraGrid Science Advisory Board.

Collaborators

Professor Farrukh S. Alvi, Florida State University
Professor Tim Colonius, California Institute of Technology
Professor Lian Duan, Missouri University of Science and Technology
Professor M. Yousuff Hussaini, Florida State University
Professor Rajan Kumar, Florida State University
Dr. Philippe Spalart, Boeing (retired)

Curriculum Vitae
Co-PI, Dr. Mujeeb R. Malik
Mujeeb.r.malik@nasa.gov
757-864-6228

Professional Preparation

Ph.D., Mechanical Engineering, Iowa State University, Ames, Iowa, 1978.
M.Eng., University of Toronto, Ontario, Canada, 1975.
B.Sc., University of Engineering and Technology, Lahore, Pakistan, 1973.

Appointments

2009 – present, Senior Aerodynamicist, NASA Langley Research Center, Hampton, VA.
2005 – 2009, Head, Computational Aerosciences Branch, NASA Langley Research Center, Hampton, VA.
2003 – 2009, Senior Research Scientist, Computational Aerosciences Branch, NASA Langley Research Center, Hampton, VA.
1990 – 2003, President and Chief Scientist, High Technology Corporation, Hampton, VA.

Five Publications Most Relevant to This Proposal

1. A. Uzun and M. R. Malik, "High-Fidelity Simulation of Turbulent Flow Past a Gaussian Bump," *AIAA Journal*, 60(4): 2130-2149, 2022.
2. A. Uzun and M. R. Malik, "Simulation of a Turbulent Flow Subjected to Favorable and Adverse Pressure Gradients," *Theoretical and Computational Fluid Dynamics*, 35(3): 293-329, 2021.
3. P. Iyer and M. R. Malik, "Analysis of the Equilibrium Wall Model for High-Sped Turbulent Flows," *Physical Review Fluids*, Vol. 4, 074604, July 2019.
4. A. Uzun and M. R. Malik, "Wall-Resolved Large-Eddy Simulations of Transonic Shock-Induced Flow Separation," *AIAA Journal*, 57(5): 1955-1972, 2019.
5. A. Uzun and M. R. Malik, "Large-Eddy Simulation of Flow over a Wall-Mounted Hump with Separation and Reattachment," *AIAA Journal*, 56(2): 715-730, 2018.

Research Interests and Expertise

Dr. Mujeeb Malik is the Senior Aerodynamicist (ST) at NASA where he leads research in numerical methods, turbulence simulations and modeling; machine learning applied to transitional and turbulent flows; physical experiments to provide high quality data for CFD validation; all with the goal of making revolutionary advances in computational fluid dynamics capability. He has served as Head of Computational Aero-Sciences Branch at NASA Langley Research Center and President of High Technology Corporation, a Research & Development firm that he founded. Dr. Malik is an internationally recognized leader in the field of boundary layer stability, transition prediction and laminar flow control. His contributions spread over one hundred and fifty papers published in refereed journals or the proceedings of conferences and symposia. He has received many honors, including NASA's Exceptional Service Medal and NASA Silver Achievement Award. He is a Fellow of American Institute of Aeronautics and Astronautics, American Physical Society and American Society of Mechanical Engineers.

He pioneered the development of laminar-turbulent transition prediction software based on compressible boundary layer stability and parabolized stability equations. The software that he developed has long been used by aerospace companies (e.g., Boeing, Lockheed-Martin, Northrop-Grumman, etc.) for laminar flow control design. Flight experiments (e.g., NASA/Boeing F16-XL experiment, Pegasus Launch Vehicle) have been conducted at subsonic, supersonic and hypersonic speeds to validate the prediction technology developed by Dr. Malik. Aerospace engineers have further used this software to develop simplified design methodology and used it for various engineering applications.

Synergistic Activities

1. Dr. Malik chaired a special session on "NASA Juncture Flow" at AIAA SciTech 2020, where modeling and simulation results of the NASA juncture flow experiment were presented.
2. He organized and chaired a special session on "Progress Towards CFD Vision 2030" at AIAA Aviation 2019.
3. He delivered an invited plenary lecture on "NASA's Revolutionary Computational Aerosciences," at ICCFD10, July 9-13, 2018, Barcelona, Spain.
4. He organized and co-chaired Future CFD Technologies Workshop on January 6–7, 2018, in conjunction with AIAA SciTech 2018. The workshop was attended by about 90 scientists from government, industry and academia from US and abroad, and key emerging computational technologies and trends were highlighted.
5. He conceived, sponsored and led CFD Vision 2030 Study in 2013 that was conducted by experts from industry and academia. The findings and recommendations of the study are beginning to have a profound impact on research in numerical methods, physical modeling and high-performance computing.

Collaborators

Professor M. Yousuff Hussaini, Florida State University
Dr. Philippe Spalart, Boeing (retired)
Professor Dimitri Mavriplis, University of Wyoming
Dr. Michael Rogers, NASA Ames Research Center
Dr. Fei Li, NASA Langley Research Center

Section 6: Software Applications and Packages

Question #1

Please list any software packages used by the project, and indicate if they are on open source or export controlled.

Application Packages

Package Name

GTHORS

Indicate whether Open Source or Export Controlled.

Export Controlled

Section 7: Wrap-Up Questions

Question #1

National Security Decision Directive (NSDD) 189 defines Fundamental Research as "basic and applied research in science and engineering, the results of which ordinarily are published and shared broadly within the scientific community, as distinguished from proprietary research and from industrial development, design, production, and product utilization, the results of which ordinarily are restricted for proprietary or national security reasons." Publicly Available Information is defined as information obtainable free of charge (other than minor shipping or copying fees) and without restriction, which is available via the internet, journal publications, textbooks, articles, newspapers, magazines, etc.

The INCITE program distinguishes between the generation of proprietary information (deemed a proprietary project) and the use of proprietary information as input. In the latter, the project may be considered as Fundamental Research or nonproprietary under the terms of the nonproprietary user agreement. Proprietary information, including computer codes and data, brought into the LCF for use by the project - but not for generation of new intellectual property, etc., using the facility resources - may be protected under a nonproprietary user agreement.

Proprietary Information

Are the proposed project and its intended outcome considered Fundamental Research or Publicly Available Information?

Yes

Will the proposed project use proprietary information, intellectual property, or licensing?

No

Will the proposed project generate proprietary information, intellectual property, or licensing as the result of the work being proposed?

If the response is Yes, please contact the INCITE manager, INCITE@doeleadershipcomputing.org, prior to submittal to discuss the INCITE policy on proprietary work.

No

Question #2

The following questions are provided to determine whether research associated with an INCITE proposal may be export controlled. Responding to these questions can facilitate - but not substitute for - any export control review required for this proposal.

PIs are responsible for knowing whether their project uses or generates sensitive or restricted information. Department of Energy systems contain only data related to scientific research and do not contain personally identifiable information. Therefore, you should answer "Yes" if your project uses or generates data that fall under the Privacy Act of 1974 U.S.C. 552a. Use of high-performance computing resources to store, manipulate, or remotely access any national security information is prohibited. This includes, but is not limited to, classified information, unclassified controlled nuclear information (UCNI); naval nuclear propulsion information (NNPI); and the design or development of nuclear, biological, or chemical weapons or of any weapons of mass destruction. For more information contact the Office of Domestic and International Energy Policy, Department of Energy, Washington DC 20585, 202-586-9211.

Export Control

Does this project use or generate sensitive or restricted information?

No

Does the proposed project involve any of the following areas?

i. Military, space craft, satellites, missiles, and associated hardware, software or technical data

ii. Nuclear reactors and components, nuclear material enrichment equipment, components (Trigger List) and associated hardware, software or technical data

iii. Encryption above 128 bit software (source and object code)

iv. Weapons of mass destruction or their precursors (nuclear, chemical and biological)

No

Does the proposed project involve International Traffic in Arms Regulations (ITAR)?

No

Question #3

The following questions deal with health data. PIs are responsible for knowing if their project uses any health data and if that data is protected. Note that certain health data may fall both within these questions as well as be considered sensitive as per question #2. Questions regarding these answers to these questions should be directed to the centers or program manager prior to submission.

Health Data

Will this project use health data?

No

Will this project use human health data?

No

Will this project use Protected Health Information (PHI)?

No

Question #4

The PI and designated Project Manager agree to the following:

Monitor Agreement

I certify that the information provided herein contains no proprietary or export control material and is correct to the best of my knowledge.

Yes

I agree to provide periodic updates of research accomplishments and to

acknowledge INCITE and the LCF in publications resulting from an INCITE award.

Yes

I agree to monitor the usage associated with an INCITE award to ensure that usage is only for the project being described herein and that all U. S. Export Controls are complied with.

Yes

I understand that the INCITE program reserves the right to periodically redistribute allocations from underutilized projects.

Yes

Section 8: Outreach and Suggested Reviewers

Question #1

By what sources (colleagues, web sites, email notices, other) have you heard about the INCITE program? This information will help refine our outreach efforts.

Outreach

Question #2

Suggested Reviewers

Section 9: Testbed Resources

Question #1

The ALCF and OLCF have test bed resources for new technologies, details below. If you would like access to these resources to support the work in this proposal, please provide the information below. (1 Page Limit)

The OLCF Quantum Computing User Program is designed to enable research by providing a broad spectrum of user access to the best available quantum computing systems, evaluate technology by monitoring the breadth and performance of early quantum computing applications, and Engage the quantum computing community and support the growth of the quantum information science ecosystems. More information can be found here: https://www.olcf.ornl.gov/olcf-resources/compute-systems/quantum-computing-user-program/quantum-computing-user-support-documentation.

The ALCF AI Testbed provides access to next-generation of AI-accelerator machines to enable evaluation of both hardware and workflows. Current hardware available includes Cerebras C-2, Graphcore MK1, Groq, Habana Gaudi, and SambaNova Dataflow. New hardware is regularly acquired as it becomes available. Up to date information can be found here: https://www.alcf.anl.gov/alcf-ai-testbed.

Describe the experiments you would be interested in performing, resources required, and their relationship to the current proposal. Please note, these are smaller experimental resources and a large amount of resources are not available. Instead, these resources are to explore the possibilities for these technologies might innovate future work. This request does not contribute to the 15-page proposal limit.

TESTBED.pdf
The attachment is on the following page.

TESTBED RESOURCES

Not applicable to this proposal.