

GT2016-57468

**FAST HIGH ORDER
LARGE EDDY SIMULATIONS ON MANY CORE COMPUTING SYSTEMS FOR
TURBOMACHINERY**

Yi Lu

Cambridge Flow Solutions Ltd
Histon, Cambridge,
UK, CB24 9AD

Kai Liu

BoXeR Solutions KK, Chuo-ku,
Kobe,
Japan, 651-0087

W.N. Dawes

Whittle Laboratory
University of Cambridge
UK, CB3 0DY

0 ABSTRACT

The overall aim of our research is to enable overnight high fidelity LES for realistic industry problems on affordable computing resource. We have adopted a “3E” approach: high spatial discretization Efficiency on general unstructured meshes, high Efficiency accurate time integration and high computing Efficiency on modern low cost HPC hardware. Our approach is centered on high order Flux Reconstruction with local time stepping – the STEFR algorithm [1]. In this paper, an offload-mode version of this code is described targeted at a heterogeneous many-core computing system based on low cost commodity hardware - Intel PHI cards. Three key techniques are introduced to achieve high FLOP rates - and optimal usage of non-equilibrium memory of both CPU and the many core coprocessor - with three levels of parallelization, multi-level non-equilibrium mesh partition and an asynchronous computing structure. A series of high order LES runs for a high lift low pressure turbine blade and a transonic turbine blade, with different order of accuracy, both fully wall-resolved and wall-modelled, were performed, analyzed and presented. This work demonstrates that the high order STEFR method has the potential to support over-night LES for realistic industrial problems on affordable computing resource.

1 INTRODUCTION

One of the most important requirements from industry for the next generation of CFD software is the ability to provide affordable high fidelity results and analysis for large scale, real geometry problems. For a non-linear system such as the Navier-Stokes equations, any under-resolved high frequency parts of the flow commonly lead to aliasing errors in the lower frequency parts which are of more interest in terms of industrial performance assessment. The more complex the flowfield, the

more difficult it is for proper modelling of the high frequency part of the flow, and the easier it is to cause both inaccuracy and even instability in simulations. High fidelity simulations which attempt to resolve the higher frequency part (ie. the smaller turbulent scales) of the flowfield, are increasingly Large Eddy Simulations (LES) or even Direct Numerical Simulation (DNS). Both provide more general ability to handle turbulence without modelling and, for both, higher order methods are more efficient at correctly representing the wider energy spectrum resolved.

As part of our research we introduced the STEFR method [1] which supports high order accuracy with both space and time discretization on arbitrary, general unstructured meshes. In particular, time-accurate local time-stepping was enabled by using a very efficient predictor-corrector type time integration method. The approach was demonstrated [2] to be 10~100 times faster for realistic simulations as compared to conventional uniform time-stepping methods – and the wider the range of scales to the geometry, the more complex the physical problem, the higher this “speed up ratio” becomes.

Nevertheless, LES is still too time-consuming and still relies on expensive HPC devices and, even on very large modern clusters, the complexity and the range of geometry scales of industrial problems which can be handled routinely by LES are still very limited compared to commonly used RANS solvers [3]. This inhibits the ability of the higher fidelity resolution of flow mechanisms enabled high order LES to benefit industry designs. In order to reduce still further the cost of LES, this paper introduces the implementation of the STEFR method on a multi-node many-core computing system consisting of CPUs and Intel PHI coprocessors. This system is built from low cost, commodity hardware and has low running costs. In particular, we built an 8 node heterogeneous computing system, each node has 2 Intel Xeon CPUs, each has 8 physical cores each with 6 many-core Intel PHI cards each of which in turn has 57 physical cores.

The biggest challenge of this research, is the uniform time-marching process at the heart of STEFR during which each cell has an adaptive local time step. This brings difficulties for load balancing the irregular data-communications. A three level parallel data communication model was constructed, with a multi-constraint, multi-level non-equilibrium mesh partition, to make maximum use of both the higher computing ability of Intel PHI coprocessor and the larger on-site memory of CPUs.

This paper is structured as follows. First, the basic numerical formulation is reviewed, including high order space and time discretizations. Next, the construction of ‘‘offload’’ mode code is presented with three key novel techniques. A series of high order LES runs for a high lift low pressure turbine blade and a transonic turbine blade were performed to test and compare different choices: third order versus fourth order, wall resolved versus wall modelled, different mesh densities, zero turbulent density inlet versus synthetic fluctuated turbulent inlet. Statistics for computational cost and memory consumption for each simulation are listed; some analysis of the observed flow will be presented as well. The motivation in this paper is not only to provide some validation evidence for the functional performance of the present STEFR method but also to extract data on the computational efficiency on our novel hardware architecture.

2 NUMERICAL METHOD

2.1 FLUX RECONSTRUCTION DISCRETISATION

The Flux Reconstruction (FR) approach was originally proposed by Huynh [3] for 1D conservation laws. Wang and Gao [4] extended the idea to simplex meshes for Euler equations and further for Navier-Stokes equations [5]. These approaches are summarized by the Correction Procedure via Reconstruction (CPR) method of Haga, Gao and Wang [6]. An infinite range of high-order energy stable flux reconstruction schemes were developed by Vincent, Castonguay and Jameson [7]. The FR approach is simple, flexible and very efficient thanks to its differential form without any numerical integration (quadrature), and it has proved it is capable of higher efficiency than other high order schemes (for example [8] & [9]).

In FR discretization, all elements are transformed from the physical domain (x, y, z) to a local domain (ξ, η, ζ) . Following the coordinates transformation, we define (with J the Jacobian):

$$\hat{U} = |J|U \quad (1)$$

$$\begin{aligned} F^\xi &= |J|(\xi_x F_x + \xi_y F_y + \xi_z F_z) \\ F^\eta &= |J|(\eta_x F_x + \eta_y F_y + \eta_z F_z) \\ F^\zeta &= |J|(\zeta_x F_x + \zeta_y F_y + \zeta_z F_z) \end{aligned} \quad (2)$$

The governing Navier-Stokes equations re-cast in local domain coordinates become:

$$\frac{\partial \hat{U}}{\partial t} + \nabla^\xi \cdot \vec{F}^\xi = \frac{\partial \hat{U}}{\partial t} + \frac{\partial F^\xi}{\partial \xi} + \frac{\partial F^\eta}{\partial \eta} + \frac{\partial F^\zeta}{\partial \zeta} = 0 \quad (3)$$

When discretized on an arbitrary, unstructured mesh with non-overlapping elements, the j -th solution points of the i -th element lead to the uniform FR formulation for all different types of elements as:

$$\frac{\partial \hat{U}_{i,j}}{\partial t} + \left(\nabla^\xi \cdot \vec{F}^\xi(U_i) \right)_{i,j} + \sum_{s=1}^{N_s} \sum_{m=1}^{K_s} \alpha_{j,s,m} (\tilde{F}^\xi|_n - \bar{F}^\xi|_n)_{i,s,m} = 0 \quad (4)$$

where N_s is the number of faces for the element and K_s equals number of flux points on the face. \tilde{F}^ξ , \tilde{F}^η and \tilde{F}^ζ denote the common flux which takes the form of Riemann fluxes for the inviscid flux and central averaged values for viscous part. In particular, the difference between the common flux and the outer normal projection of the local flux $(\tilde{F}^\xi|_n - \bar{F}^\xi|_n)$, is called the ‘‘correction flux’’, the same as in the 1D FR formulation, and which is used to update the DOFs by exchanging information with adjacent elements. The FR coefficients α can be obtained through the ‘‘lift operation’’ [4] for the standard element types - the detail operations can be found in [8].

2.2 SPACE TIME EXTENSION

Inspired by the Continuous Extension Runge-Kutta (CERK) approach for high order discontinuous method, introduced by Gassner *et al* [10], a ‘‘predictor-corrector’’ type space-time extension for high order FR (which we call STEFR) was devised, which uses the ‘‘flux divergence’’ part to construct the local predictor by using the continuous Runge-Kutta method [11], and then the ‘‘correction flux’’ part of the FR discretization is used as the corrector. In the original FR formulation, the flux divergence represents the major part of the compute workload and is completely local to the elements, whereas the linear combination of correction flux parts, which takes adjacent elements into account, is used for updating the degree of freedoms (DOFs). The STEFR method combines these two parts smoothly by using staggered operations which are simple, efficient, accurate for both space discretization and time-marching and permit local time-stepping.

We rewrite Equation (4) as the following ‘‘unified’’ form:

$$\frac{\partial U_{i,j}}{\partial t} = \mathbf{R}_{i,j}^D(U_i) + \sum_{s=1}^{N_s} \mathbf{R}_{i,j,s}^C(U_i, U_{i,s}^{nb})$$

with \mathbf{R}^D and \mathbf{R}^C given as:

$$\begin{aligned} \mathbf{R}_{i,j}^D &= -\frac{1}{|J|_{i,j}} \left(\nabla^\xi \cdot \vec{F}^\xi(U_i) \right)_{i,j}, \\ \mathbf{R}_{i,j,s}^C &= -\frac{1}{|J|_{i,j}} \sum_{s=1}^{N_s} \sum_{m=1}^{K_s} \alpha_{j,s,m} (\tilde{F}^\xi|_n - \bar{F}^\xi|_n)_{i,s,m} \end{aligned} \quad (8)$$

Integrating Equation (8) over time step interval $t \in [t^n, t^{n+1}]$ one obtains:

$$U_{i,j}^{n+1} - U_{i,j}^n = \int_{t^n}^{t^{n+1}} \mathbf{R}_{i,j}^D(U_i) + \sum_{s=1}^{N_s} \mathbf{R}_{i,j,s}^C(U_i, U_{i,s}^{nb}) dt \quad (9)$$

Inspired by the space-time extension of DG(STEDG) [10], we now construct a local space-time approximation $v = v(\vec{x}_i, t)$ for $t \in [t^n, t^{n+1}]$ by solving the following time-dependent ODE:

$$\frac{dv_{i,j}}{dt} = \mathbf{R}_{i,j}^D(t, v(\vec{x}_i, t)), \quad v(\vec{x}_i, t = 0) = U_i^n(\vec{x}_i) \quad (10)$$

From Equations (7) and (8) it can be seen that \mathbf{R}^D is completely local, which indicates the time evolution of $v = v(\vec{x}_i, t)$ is local also. The continuous Runge-Kutta method [11] is adopted to solve Equation (10) using the following procedure as:

$$\begin{aligned} v_{i,j}(\tau) &= U_{i,j}^n + \Delta t \sum_{l=1}^{N_t} \dot{B}_l(\tau) \dot{H}_l \\ \dot{B}_l(\tau) &= \sum_{m=1}^{O_t} b_{l,m} \tau^m \\ \dot{H}_l &= \mathbf{R}_{i,j}^D(v_i^l) \\ v_{i,j}^l &= U_{i,j}^n + \Delta t \sum_{n=1}^l a_{l,n} \dot{H}_{l-1} \end{aligned} \quad (11)$$

where $\tau \in [0,1]$ is the non-dimensional time, $\Delta t = t^{n+1} - t^n$, O_t is the order of time integration and N_t is the related number of stages and the coefficients \mathbf{a} and \mathbf{b} are given by Owren and Zennaro [9]. Integrating Equation (10) for $t \in [t^n, t^{n+1}]$, one obtains

$$\int_{t^n}^{t^{n+1}} \mathbf{R}_{i,j}^D(v_i^l) dt = v_{i,j}(\tau = 1) - U_{i,j}^n \quad (12)$$

Taking the space-time polynomial $v = v(\vec{x}_i, t)$ as a local predictor, and the combination of correction flux \mathbf{R}^C as corrector, substituting Equation (12) into Equation (9) results in the final step of the space time extension of the Flux Reconstruction (STEFR) scheme as:

$$U_{i,j}^{n+1} = v_{i,j}(\tau_{max}) + \Delta t \int_0^1 \sum_{s=1}^{N_s} \mathbf{R}_{i,j,s}^C(v_i(\tau), v_{i,s}^{nb}(\tau^{nb})) d\tau \quad (13)$$

where τ^{nb} indicates the non-dimensional time for adjacent element with the same physical time.

2.3 EFFICIENT PARALLEL IMPLEMENTATION

Central to the efficiency of the STEFR method is that all elements are allowed to use their maximum allowed local time-step - which is itself adaptive during the simulations. This irregular time-stepping brings a lot of challenges for the time integration in maintaining physical time synchronicity - especially on parallel computing systems. This section focus on application of STEFR and its parallel implementation - targeted particularly at the emerging many core hardware platforms.

For the convenience of data exchange on element interfaces which is used for time integration of correction fluxes, the actual time-steps for all elements are set as powers of 2 with respect to the global smallest time-step. To illustrate this Figure 1 presents snapshots of a transient prediction during a global synchronous time step for a typical 3D simulation - which in this example has 176 inner iterations in total. Compared to the more commonly

used uniform step time marching methods, this approach needs to spend more effort in setting up getting executable queues and asynchronous parallel communications.

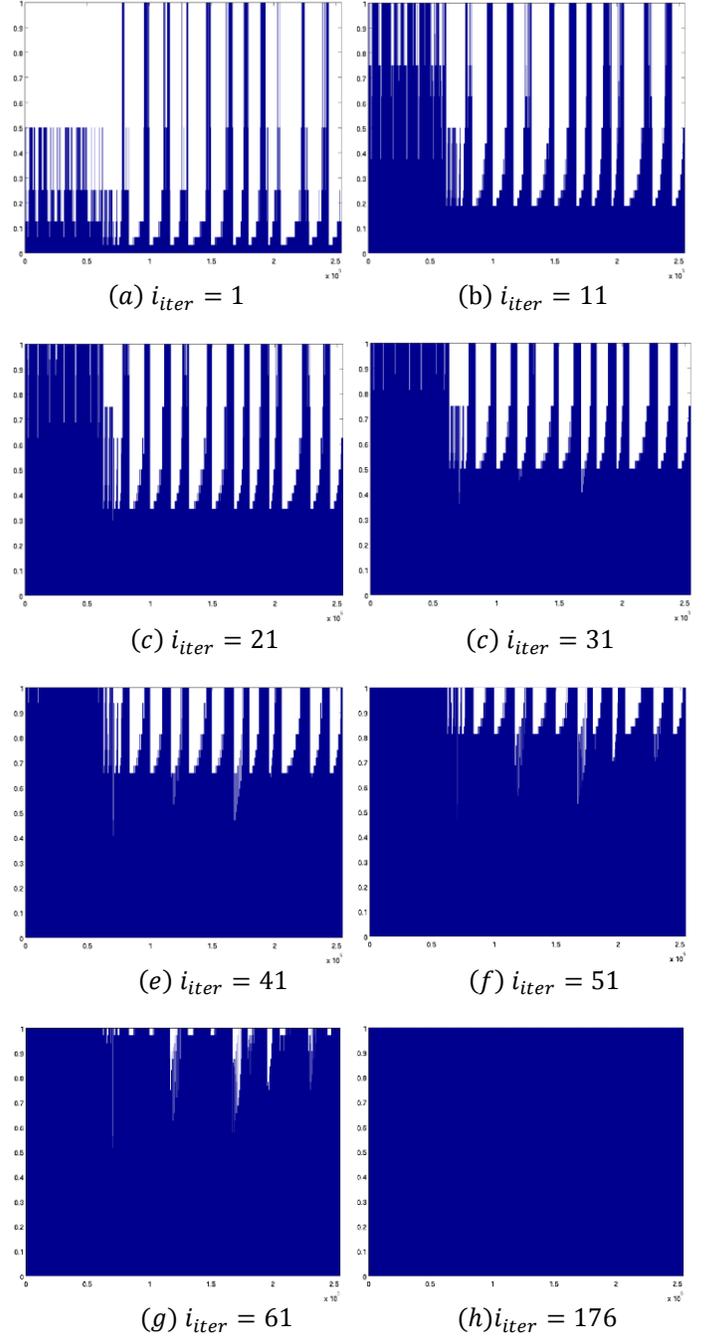


Figure 1. Snapshots time marching of one step: horizontal is element index and vertical is normalized prediction time

Therefore, the key to successful application of the STEFR method is minimizing the parallel communications. For the heterogeneous computing architecture adopted in this work (and discussed in the next Section) we use OpenMP for multi-threaded parallel looping inside of each shared memory computing unit, whereas the communications between different computing units are undertaken by MPI. For most simulations on different computing systems, the fraction of wall-clock time

for effective functional evolutions is over 70% of the computational resource for most cases, including the correction flux calculations, predictions and corrections. The efficient implementation of STEFR using local time-stepping is much more complex than other explicit methods using uniform time-stepping, such as explicit Runge-Kutta methods - however, the algorithm presented in this section has proved very efficient - even for large scale simulations, with more than 2 billion DOFs, and speed up ratios up to ~ 100 as compared to using global uniform time-stepping have been achieved [2].

2.4 OFFLOAD MODE IMPLEMENTATION

The theme of this paper is “3E”: high spatial discretization Efficiency on general unstructured meshes, high Efficiency accurate time integration and high computing Efficiency on modern low cost HPC hardware. The previous Sections have discussed the first of these two efficiencies – this Section looks at a rather novel hardware architecture.

Many-core computing systems are widely used and have progressed rapidly in recent years because of its high cost-effectiveness compared to pure, “traditional” multi-core CPU computing system in the HPC area. These computing systems are based on different many-core units including NVIDIA Tesla GPUs, AMD GPUs and Intel PHI co-processors. As is clear from the numerical review of the STEFR method earlier, its time marching method is not uniform and the data-communication is irregular. Also, for some computing loops of a single time marching step, the number of executive elements is maybe quite small especially in the final stage of inner iterative as shown in Figure 1(e)~ Figure 1(g). Therefore, several available many-core units have physical computing threads which are not suitable for STEFR method, such as NVIDIA Tesla GPUs and AMD GPUs. As reported in this paper, the Intel PHI co-processors have been chosen to build our heterogeneous computing system in order to trade fewer computing cores against each physical core having much stronger computing ability.

To support this work we built a heterogeneous many-core computing system consisting of 8 nodes, each node has 2 Intel Xeon CPUs each with 8 physical cores and 6 many-core Intel PHI cards each with in turn 57 physical cores. All components are commodity items, easily and cheaply available. The system architecture is illustrated in Figure 2. This type of system holds out great promise going forward for a step change reduction in hardware costs – and hence, if the system can be driven efficiently, a step change reduction in LES solution time scales and cost.

There are three different types of data-communication used in the computing system as shown in Figure 3: CPUs to CPUs, PHI co-processors and CPUs, internal data-communications between PHI co-processors/CPU. Due to the irregular time marching process, the principle behind the design of the data-communication model is to reduce the usage of distributed memory, and make use of more communication latency. As shown in Figure 3, asynchronous MPI is used for communication between CPUs though Infiniband (which has already

demonstrated its high efficiency [1][2]).The data-transfer performance for small packages of data between host CPUs and PHI cards using Intel MPI is very poor even using OFED, therefore the “offload mode” [12] code was written which has mirror memory on host CPUs of all data-structures allocated on PHI co-processors and which speeds up the data transfer process. For each of the many-cores on host CPUs of each node, and on each PHI coprocessor, all executive loops are performed on shared-memory by using OpenMP’s multi-threading method. In order to reduce small package data-communication, the “offload mode” data transfer is synchronous between CPUs and PHI coprocessors, there is no communication between different PHI coprocessors and between PHI co-processors and CPUs on other nodes by using smart partitioning.

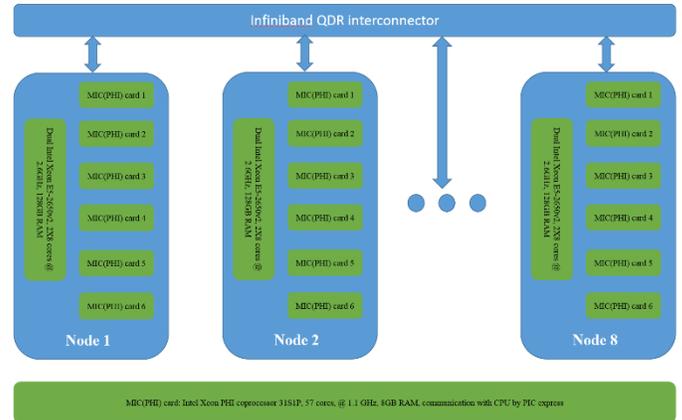


Figure 2. Intel PHI based system architecture

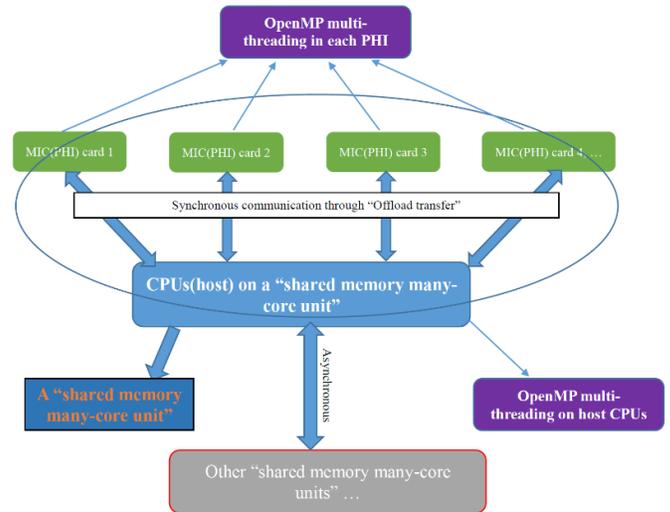


Figure 3. Three level data-communication model

Another challenge for modern many-core computing units is the limited memory (8GB per PHI coprocessor) compared to CPUs (128GB per node). From the numerical scheme review in Section 1.2, in the STEFR method, the computational cost for single cell depends on its smallest size (and associated time limit) - however, the memory consumption still scales with the element number. So, in this work, a special multi-level, multi-constraint smart partitioning algorithm was written, to automatically

allocate more small size elements to PHI co-processors (typically near wall boundary layer elements) and put more elements on the CPUs to maintain load balancing and reduce data-communication size.

3 VALIDATION ON TWO STANDARD TURBINE TEST CASES

This Section will detail a series of LES runs performed on two well-known, standard turbine test cases and report the levels of software & hardware efficiency we were able to achieve in pursuit of our 3E goal.

3.1 LOW PRESSURE TURBINE BLADE T106A

This classic test case concerns the transitional and separated flow on the so-called T106A high-lift subsonic turbine cascade. A typical domain and mesh is shown in Figure 4 - all the high order hybrid unstructured meshes used in this paper were generated by *BOXERMesh* [2].

The Reynolds number is set equal to 1.1×10^5 based on the inlet velocity and the axial blade chord C . The inlet Mach number is equal to 0.1. The spanwise length of the domain $L_z = 0.075C$, this is regarded as sufficient to capture the main flow structures.

A series of LES runs were performed, with different solution orders of accuracy, both fully wall-resolved & wall-modelled, and with both turbulent inlet and zero turbulence intensity inlet boundary conditions; the Appendix summarizes all the runs. The motivation in this paper is partly to provide some validation evidence for the functional performance of the present STEFR method and also to extract data on the computational efficiency on our novel hardware architecture. Hence, only representative results will be shown here and discussed. In the Appendix “ T_p ” denotes the characteristic flow-past time scale – to gather reliable statistics the LES must cover several of these scales.

Near-wall modelling is very important for this type of simulation and two approaches were used: “fully wall-resolved” with very fine near-wall mesh spacing and a simple van Driest damping factor for the sub-grid model; and “wall-modelled” following the approach of Moin *et al* [15] based on a local solution of wall-normal equations derived from the Navier-Stokes equations (in other words a rather sophisticated “law of the wall” sensitive to local pressure gradient).

Turning to the results, first, Figures 5a & b show time-averaged and transient Mach number for Case T106A-1. This case is 3rd order accurate, and wall-resolved with near wall $Y^+ \sim 3$. The “speed up ratio” derived from the SREFR local time stepping algorithm is a factor of ~ 7 compared to the classical uniform constant time step approach. Figure 4c shows iso-surfaces of Q-criterion for transient result.

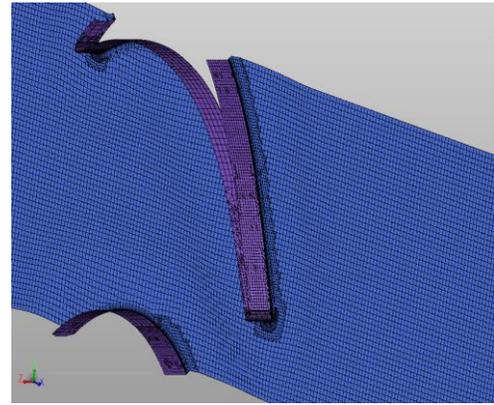


Figure 4. A typical mesh for the T106 test case.

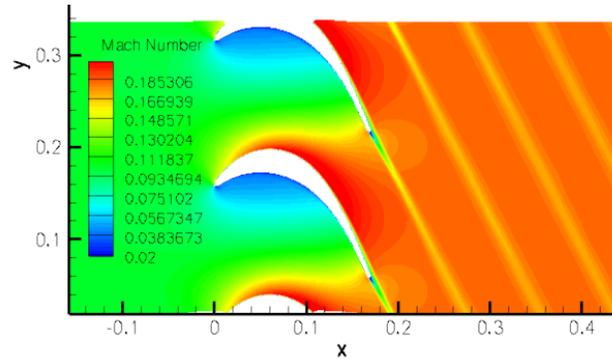


Figure 5a. Time average Mach number for Case T106A-1

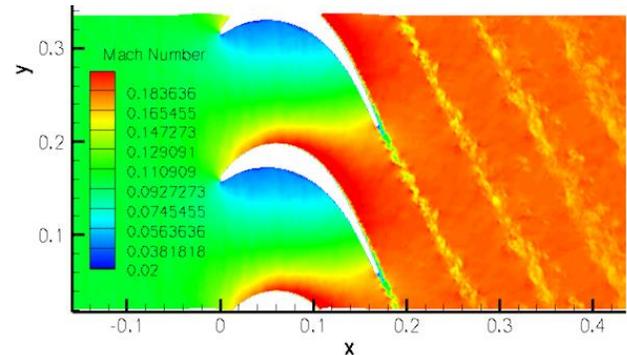


Figure 5b. Transient Mach number for Case T106A-1

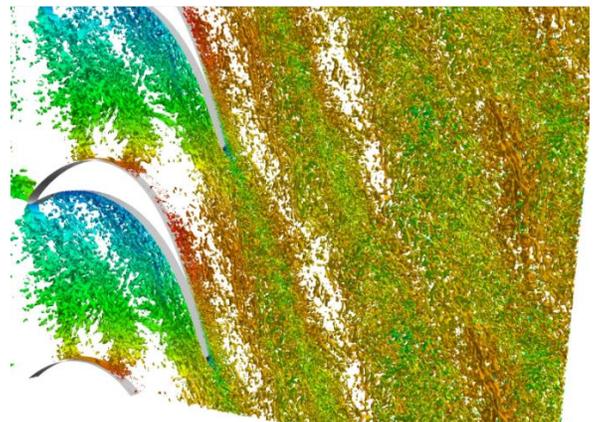


Figure 5c. ISO-Surface of Q-criterion(Q=50000) for transient result, colored by Mach number

The predicted turbulence spectrum within the blade wake is presented in Figure 6 showing that the present LES resolves the flow well into the inertial range and agrees well with the expected “-5/3 law”.

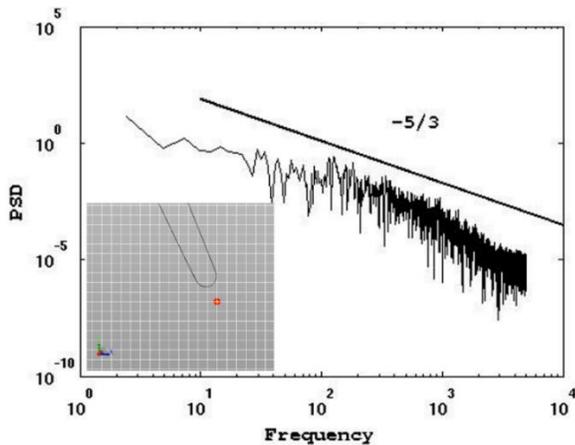


Figure 6. Power spectral density of total velocity within the blade wake ($x=0.172, y=0.05655$), case T106A-4

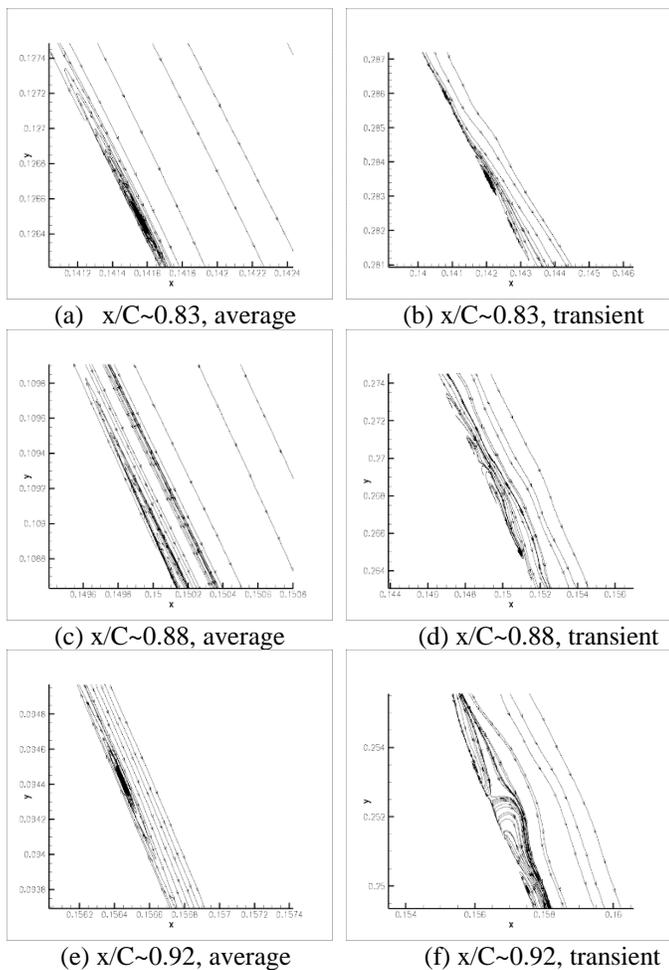


Figure 7. Comparisons of streamlines around the transitional/separation zone for both the time-averaged flow and a transient snapshot; Case T106A-4

The flow features laminar separation around $x/C \sim 0.88$ and a relatively slow natural transition. Figure 7 shows comparisons of streamlines around the transitional/separation zone for both the time-averaged flow and a transient snapshot. These results were extracted from Case T106A-4. This case is 4th order accurate, and wall-resolved with near wall $Y^+ \sim 5$. The “speed up ratio” derived from our SREFR local time stepping algorithm is a factor of ~ 8 compared to a classical uniform constant time step approach. Comparisons of velocity vectors around the transitional/separation zone for both the time-averaged flow and a transient snapshot are shown in Figure 8 – again for case T106A-4. A separation bubble can be seen to have formed by about $x/C \sim 0.92$ (a little later than in the experiments). What is interesting is that the spatial scale of the disturbance & reversed flow occupies a much larger wall-normal extent than the time-averaged wall-normal bubble scale.

Figure 9 compares measured and predicted blade suction side skin friction for Cases T106A-1 (3rd order accurate, wall-resolved with near wall $Y^+ \sim 3$) and T106A-4 (4th order accurate, and wall-resolved with near wall $Y^+ \sim 5$). The comparisons, especially for the 4th order accurate simulation are very encouraging.

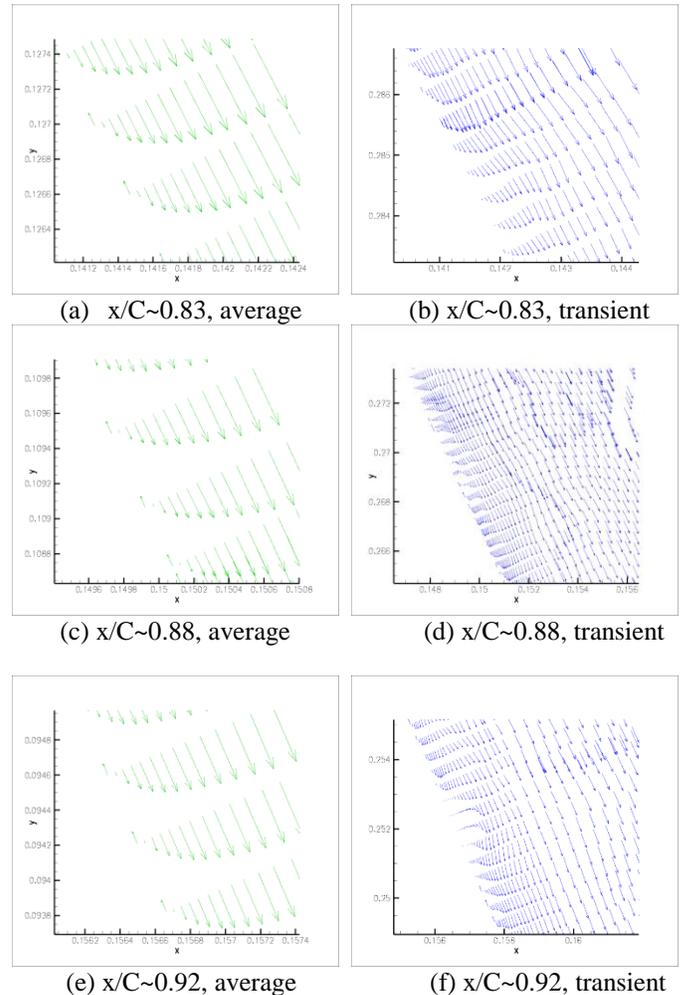


Figure 8. Comparisons of velocity vector around the transitional/separation zone for both the time-averaged flow and a transient snapshot; case T106A-4

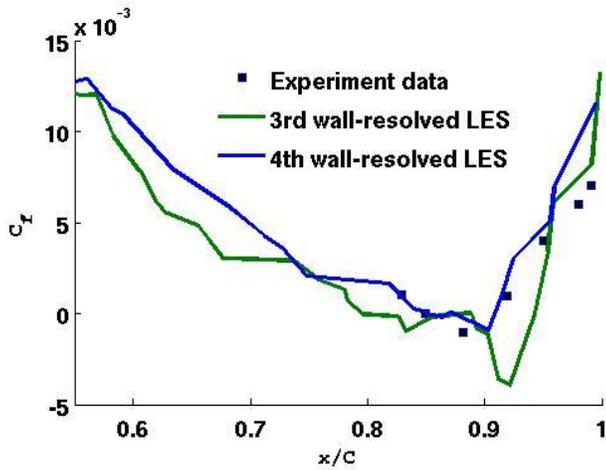


Figure 9. Measured and predicted blade suction side skin friction for Cases T106A-1 (3rd order accurate, wall-resolved with near wall $Y^+ \sim 3$) and T106A-4 (4th order accurate, and wall-resolved with near wall $Y^+ \sim 5$)

The next set of comparisons is shown in Figure 8 – measured and predicted wall-normal RMS velocity profiles using data from Case T106A-4. Again, the comparisons are very encouraging although the transition is evidently a little over predicted but is fully complete by $x/C \sim 0.92$.

As a test, one LES run, T106A-11, was performed with an inlet turbulence intensity of 3.8%, using an isotropic fluctuations inlet boundary condition [17] in which synthesized perturbations in the three velocity components at inlet are set up using a sum of discrete waves. The simulation was wall-resolved, 4th order accurate and had near-wall $Y^+ \sim 5$. However, comparing mean velocity profiles around the transitional region with Cases T106A-4 and T106A-5, which had exactly the same mesh but only different inlet turbulence, there was nearly no observed effect of the inlet turbulence within the near-wall region (the laminar sublayer) and only very small differences at the start of transition point and the re-attached region. In the simulations the rather small turbulence intensity has no obvious effect on transition for this case.

Finally, in terms of computer resource, most cases were run on just one of the eight nodes on our Intel PHI cluster, Cases 1 and 4 needed wall-clock times of ~ 32 and ~ 45 hours per characteristic flow-past time scale, T_p . To demonstrate our progress towards our ultimate goal of overnight turn around Appendix A also shows Case T106A-6 run on four nodes (half the capacity of our low cost machine) needing only ~ 22 wall-clock hours per time scale.

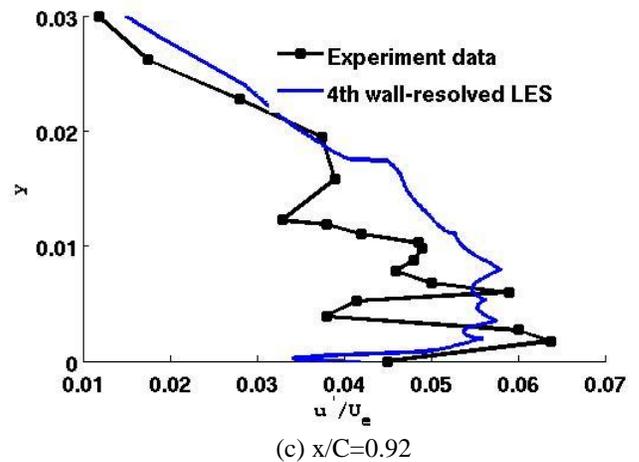
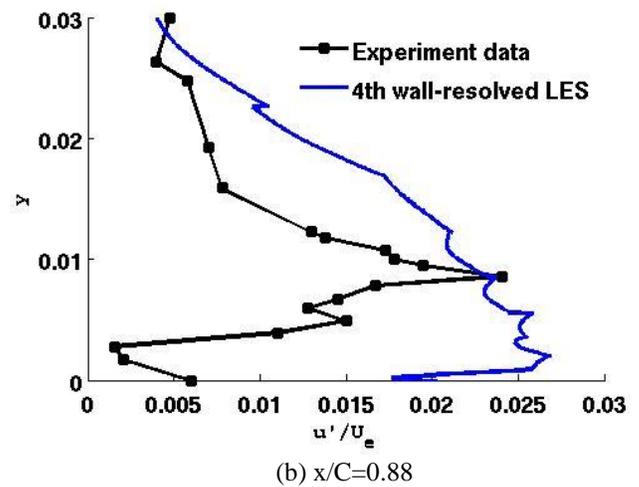
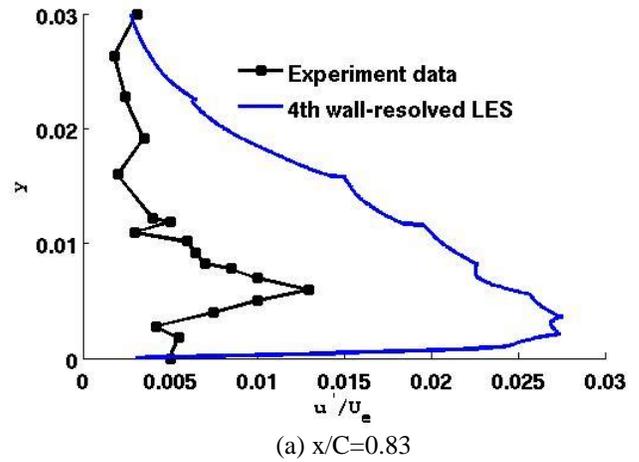


Figure 10. RMS velocity profiles around the transitional/separation zone; measurements and predictions using data from Case T106A-4 (4th order accurate, and wall-resolved with near wall $Y^+ \sim 5$)

3.2 TRANSONIC TURBINE BLADE

The computational domain & unstructured mesh for the next test case, the VKI-LS59 blade, is presented in Figure 11. The exit Reynolds number is 8.5×10^5 with exit Mach number 0.95 [18] meaning that shockwaves are present in the blade-blade flow (in contrast to the T106 case). In the LES, the extent of the spanwise domain is set to about 4.2% of the chord length. This is judged to be enough to capture the 3D structure scales. All results are based on statistics and analysis taken from the simulation with flow physical time $> 10T_p$. Three simulations were performed in this work, details of the runs are listed in Table 2 of Appendix A.

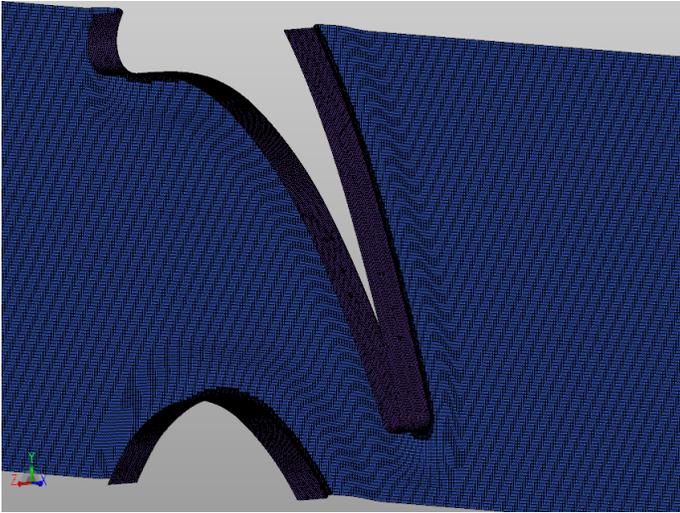


Figure 11. 3D hybrid unstructured mesh for VKI-LS59 blade

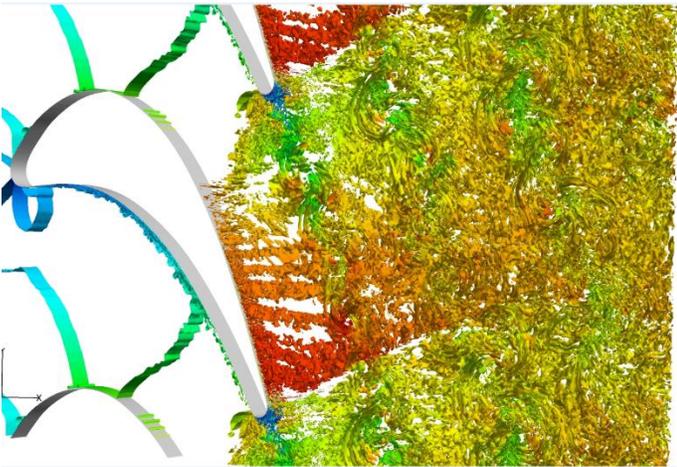


Figure 12. Iso-surface of Q-criterion(Q=50000) for a snapshot of a transient result for Case VKI-LS59-2, colored by Mach number.

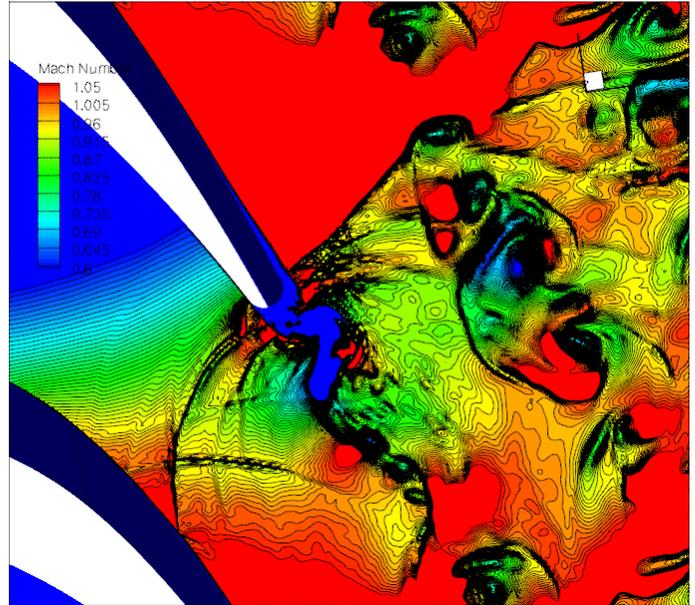


Figure 13. Instantaneous Mach number for case VKI-LS59-1 (wall-resolved LES)

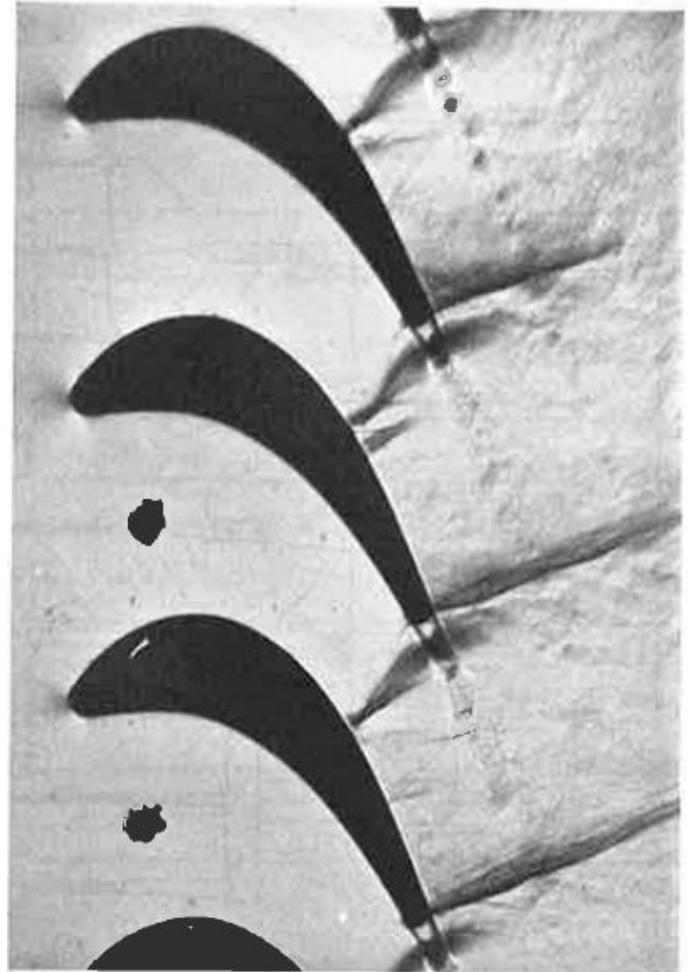


Figure 14. Schlieren picture for the VKI-LS59 [18]

To get an overview of the results, Figure 12 shows iso-surfaces of Q-criterion for a snapshot of a transient result for Case VKI-LS59-2 (wall-resolved). Compared to the previous shock-free T106 flow, Figure 5c, there is a much stronger vortical motion downstream of the trailing edge. This is even clearer in Figure 13 which shows a snapshot of instantaneous Mach number. This more than just a simple von Karman vortex street from the blade trailing edge – the suction & pressure side boundary layers separate at the trailing edge and form shear layers which couple into an unsteady shock motion producing a very strong vortical motion in the wake. This will elevate the blade loss coefficient above expected steady flow levels. The experimental Schlieren [18] in Figure 14 tends to corroborates the LES. Figure. 15 presents the Power Spectral Density of the total velocity within the wake. As before the present LES resolves the flow well into the inertial sub-range.

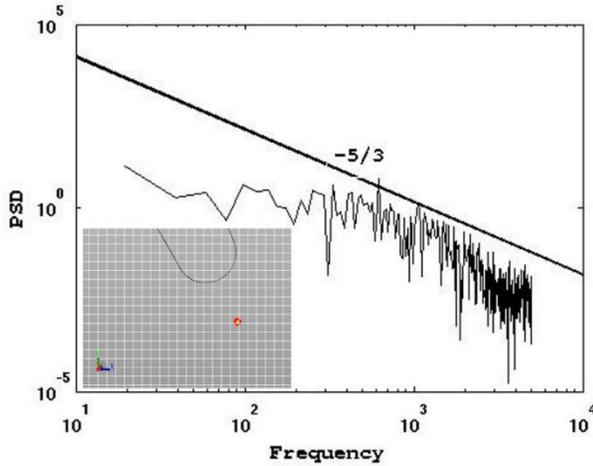


Figure 15. Power spectral density of total velocity within the wake ($x=0.0855, y=0.0139$), case VKI-LS59-2

The next set of figures, Figure 16, shows time-averaged blade-to-blade Mach number for each of Cases VKI-LS59-1, 2 & 3. Each case was run 3rd order accurate, Cases 1 & 2 were wall-resolved with near-wall Y^+ values of ~ 1.5 and ~ 3 respectively; Case 3 was wall-modelled with $Y^+ \sim 30$. Figure 17 shows comparison of blade surface isentropic Mach number between experiment and time-average LES results. The mesh densities in Case 2 (refined wall-resolved) and Case3 (wall-modelled) are very similar differing only in the Y^+ value for the first boundary layer mesh (~ 3 and ~ 30 respectively) – and both have very much smaller values of near-wall X^+ and Z^+ (both ~ 15) compared to the baseline wall-resolved Case 1 (X^+ and Z^+ both ~ 106).

It can be seen comparing Figures 16a, b & c that the much finer streamwise & spanwise mesh resolution predicts a time-averaged position of the strong shock downstream of the trailing edge which is distinctively closer in location and angle compared to the experimental Schlieren in Figure 14. This observation is true also for the passage shock – best observed via the time-averaged isentropic blade surface Mach numbers in Figure 17. The refined wall-resolved and wall-modelled runs, Cases 2 & 3, predict much better the pressure distribution through the passage shock around $X/C \sim 0.55$ on the suction side than in the baseline

Case 1. Clearly, strict control of X^+ and Z^+ as well as Y^+ is needed for accurate predictions – obviously, adequate resolution of the boundary layer and the interaction with the shock is critical for accurate prediction of flows of this type.

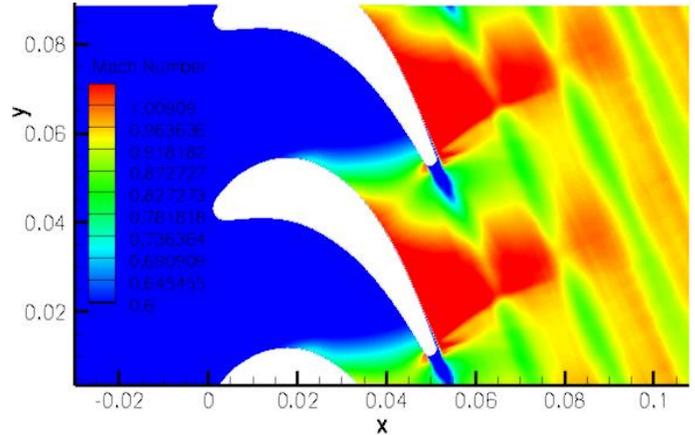


Figure 16a. Average Mach number for Case VKI-LS59-1

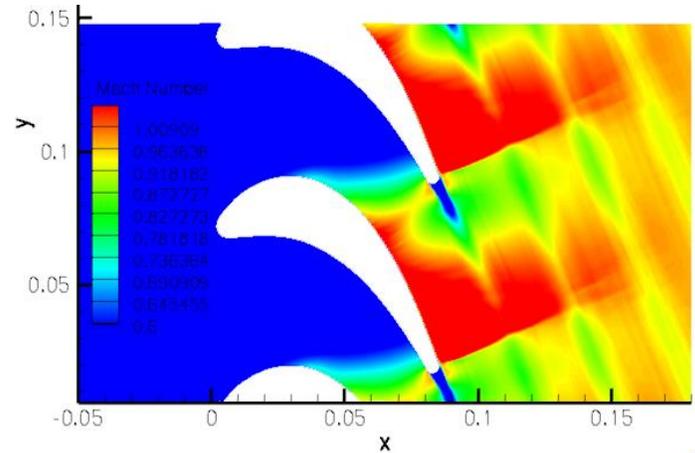


Figure 16b. Average Mach number for Case VKI-LS59-2

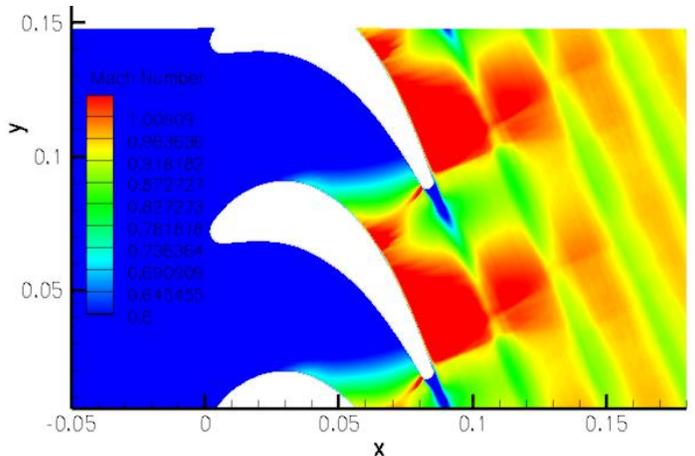


Figure 16c. Average Mach number for Case VKI-LS59-3

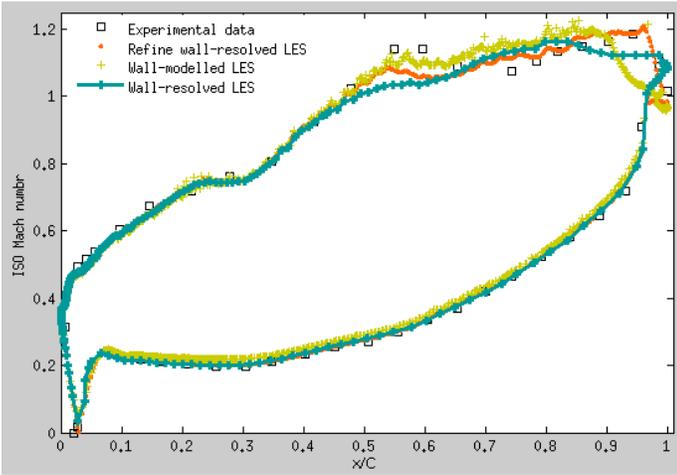


Figure 17. Comparison of isentropic blade surface Mach number between experiment and time-average LES results

Nevertheless the second weak passage shock observed in the measurements (see Figure 12) is not predicted in any of the three simulations. There could be two possible reasons: the first might be an artifact of differences between the time-averaging method used in the numerical simulations and the unknown averaging implicit in the experiment data. The second might be inferred from the comparison of Cases VKI-LS59-2 and VKI-LS59-3 - the wall-modelled LES predicted the main passage shock slightly better, which might indicate inadequate performance of the piecewise integrated method [2] used in STEFR to predict moving shocks inside the boundary layer. This shock capturing method is derived from 1D analysis directly applied as if to isentropic cells but likely needing adjustment for high aspect ratio anisotropic cells.

Finally, in terms of computer resource, all three cases were run on just one of the eight nodes on our Intel PHI cluster, Cases 1, 2 and 3 needed wall-clock times of ~32, ~28 and ~6 hours per characteristic flow-past time scale. This clearly demonstrate our progress towards our ultimate goal of overnight turn around. Case VKI-LS-3 was 3rd order accurate with 116M DOF, wall-modelled and had near-wall $Y^+ \sim 30$ and we could run 4.5 characteristic flow passing periods, T_p , per 24 hour wall clock time.

4 PERFORMANCE OF STEFR & THE INTEL PHI SYSTEM ON A LARGER, COMPLEX GEOMETRY PROBLEM

Finally, a very much larger test case, the NASA-Gulfstream airframe landing gear [20], is included to illustrate the performance of the present STEFR approach on our low cost Intel PHI based computing system. The geometry and details of the associated high order are shown on Figure 18. Statistics for the mesh and LES run configurations are listed on Table 3 of Appendix A

This case had 862M DOFs and was run on all eight nodes of our low-cost cluster and achieved a wall-clock time of ~14 hours

per characteristic flow past time based on the size of strut – within our 24 hour goal!

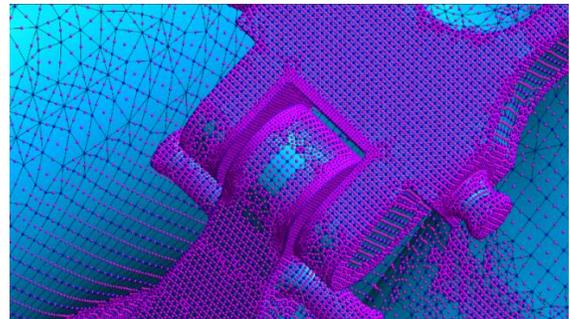
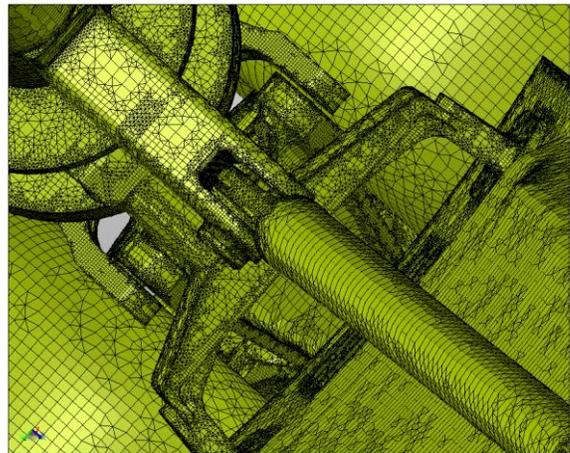
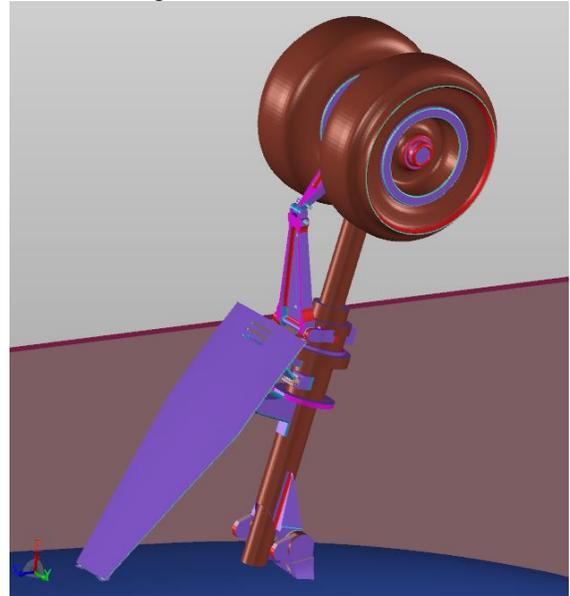


Figure 18. The geometry of the generic NASA/Gulfstream landing gear with a detail views of the mesh showing the high order collocation points

As a single snapshot visualization of the results, Figure 19 shows an iso-surface (visualized by wireframe) of Q-criterion ($Q=800000$) of a transient result for NASA/Gulfstream Landing Gear Case. Noise producing wake structures are clear. Full post-processing will be presented in a forthcoming paper.

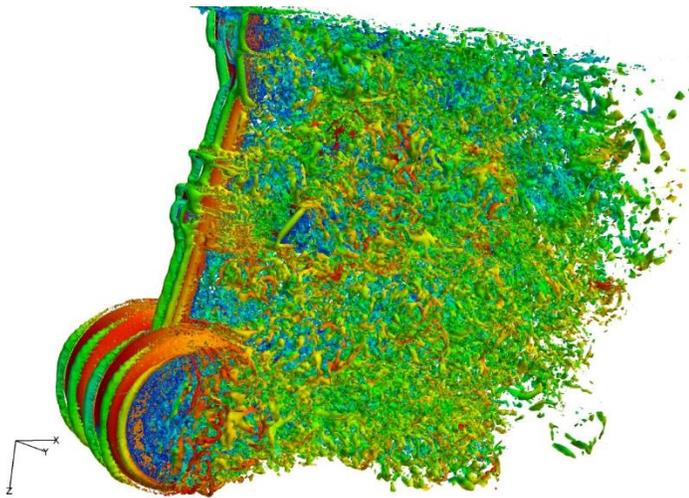


Figure 19. Iso-surface (visualized by wireframe) of Q-criterion ($Q=800000$) for a snapshot of a transient result for NASA/Gulfstream Landing Gear Case, colored by Mach number

5 CONCLUSIONS

The overall aim of our research is to enable overnight high fidelity LES for realistic industry problems on affordable computing resource. We have adopted a “3E” approach: high spatial discretization Efficiency on general unstructured meshes, high Efficiency accurate time integration and high computing Efficiency on modern low cost HPC hardware. The motivation in this paper is not only to provide some validation evidence for the functional performance of the present STEFR method but also to extract data on the computational efficiency on our novel hardware architecture.

The STEFR method, is able to use coarser meshes but with higher order the discretization is able to resolve into the higher frequency spectrum for LES. The simulations benefit from a significant speed up by using time accurate local time-stepping, compared to conventional uniform time stepping. Validation results with two standard turbine cases are very encouraging.

In terms of wall-clock time, our adoption of a novel hardware platform and associated modifications to our basic LES solver architecture clearly demonstrate our progress towards our ultimate goal of overnight turn-around.

6 ACKNOWLEDGMENTS

The authors are grateful to Cambridge Flow Solutions Ltd. For permission to publish this paper.

7 REFERENCES

[1] Yi Lu, Kai Liu, and W. N. Dawes. Large eddy simulations using high order flux reconstruction method on hybrid unstructured meshes. In *AIAA Science and Technology Forum and Exposition (SciTech2014)*, AIAA2014-0424, 2014.

[2] Yi Lu, Kai Liu, and W. N. Dawes. Flow simulation system based on high order space-time extension of flux reconstruction method. In

AIAA Science and Technology Forum and Exposition (SciTech2015), AIAA2015-0833, 2015.

[3] G. Dufour, N. Gourdain, F. Duchaine, O. Vermorel, L. Y. M Gicquel, J.F. Boussuge and T. Poinot. Numerical investigations in turbomachinery: the state of the art. *Nodes for the von karman Institute for Fluid Dynamics September*, 12-25 2009

[3] Huynh, H. T. “A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods”, *18th AIAA Computational Fluid Dynamics Conference, AIAA 2007-4079*, 2007

[4] Z. J. Wang and Haiyang Gao, “A unifying lifting collocation penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids,” *Journal of Computational Physics*, Vol. 228, No. 21, 2009, pp. 8161, 8186.

[5] Haiyang Gao and Z. J. Wang. “A high-order lifting collocation penalty formulation for the Navier-Stokes equations on 2d mixed grids”. In *19th AIAA Computational Fluid Dynamics, AIAA 2009-3784*, 2009.

[6] T. Haga, H. Gao and Z. J. Wang. “A high-order unifying discontinuous formulation for the navier-stokes equations on 3d mixed grids”. *Mathematical Modelling of Natural Phenomena*, Vol. 6, No. 21, 2011, pp. 28, 56.

[7] P.Vincent, P. Castonguay, and A.Jameson. “A new class of high-order energy stable flux reconstruction schemes”. *Journal of Scientific Computing*, Vol.47, No. 1, 2010, pp. 50,72.

[8] Y.Lu. “Local Reconstruction High Order Method and Experimental Research for Internal Flow of Turbomachinery”. *PhD thesis*, Tsinghua University, China.

[9] C. Liang, C. Cox, and M. Plesniak. “A comparison of computational efficiencies of spectral difference method and correction procedure via reconstruction”. *Journal of Computational Physics*, Vol.239, , 2013, pp. 244, 261.

[10] G. Gassner, M. Dumsner, F. Hindenlang, and C.D Munz. Explicit one-step time discretization for discontinuous galerkin and finite volume schemes based on local predictors. *Journal of Computational Physics*, Vol.230, 2011, pp. 4232, 4247.

[11] B. Owren and M Zennaro. Derivative of efficient continuous explicit runge-kutta methods. *Journal of Science Computing*, Vol 239, 2013, pp.138, 146.

[12] Jim Jeffers and James Reinders. Intel Xeon Phi Coprocessor High-Performance Programming, 2013

[13] Engber M, Fottner L. The effect of incoming wakes on boundary layer transition of a highly loaded turbine cascade. *AGARD REP 85th Symposium on Loss Mechanisms and Unsteady Flows in Turbomachines*. AGARD, 1996

[14] Hodson, H. “Turbulence Modelling for Unsteady Flows in Axial Turbine: TURMUNSLAT,” Brite-Euram Project, *Final TR CT96-1043*, von Karman Inst. April 2000, 85-99

[15] G. I. Park. and P. Moin., An improved dynamic non-equilibrium wall-model for large eddy simulation, *Phys. Fluids* 26, 015108 (2014)

[16] B. Raverdy, I. Mary, P. Sagaut, and N. Liams, High-Resolution Large-Eddy Simulation of Flow Around Low-Pressure Turbine Blade, *AIAA Journal*, Vol. 41, No. 3, 2003

[17] L. Davidson, Using Isotropic Synthetic Fluctuations as Inlet Boundary Conditions for Unsteady Simulations, *Advances and Applications in Fluid Mechanics*, Vol 1, No 1, pp. 1-35, 2007

[18] N.C. Baines C.H.Sieverding R. Kiock, F. Iehthaus. The transonic flow through a plane turbine cascade as measured in four european wind tunnels. *Journal of Engineering for Gas Turbine and Power*, 108:277–284, 1986.

[19] Yi Lu, W. N. Dawes. High Order Large Eddy Simulations for a transonic turbine blade using hybrid unstructured meshes. *ASME Paper GT2015-42283*, 2015

[20] M. R. Khorrami, D. P. Lockard, Jr. W. M. Humphreys, M. M. Choudhari and T. Van de Ven, Preliminary Analysis of Acoustic Measurements from the NASA-Gulfstream Airframe Noise Flight Test. *AIAA Paper 2008-2814*, 2008

APPENDIX A

Table 1. Configurations for low pressure turbine blade T106A simulations, flow passing time T_p is based on the size of chord length

Case ID	Inlet turbulence intensity	Near wall resolution	Order of accuracy	Speed Up Ratio	x+	y+	z+	Number of elements	Number of DOFs	Number of nodes on cluster	Wall-clock time for $1T_p$ (hours)
T106A-1	0	Wall-resolved	3 rd	7.32	27.2	3.07	27.2	460K	58.7M	1	27.33
T106A-6	0	Wall-resolved	3 rd	9.88	18.1	3.07	18.1	1.27M	166M	4	19.1
T106A-3	0	Wall-model	3 rd	11.53	27.2	16.1	27.2	409K	52.3M	1	11.4
T106A-4	0	Wall-resolved	4 th	8.27	45.4	5.15	45.4	110K	31M	1	38.8
T106A-11	3.8%	Wall-resolved	4 th	8.27	45.4	5.15	45.4	110K	31M	1	38.8

Table 2. Configurations for transonic turbine blade VKI-LS59 simulations , flow passing time T_p is based on the size of chord length

Case ID	Inlet turbulence intensity	Near wall resolution	Order of accuracy	Speed Up Ratio	x+	y+	z+	Number of elements	Number of DOFs	Number of nodes on cluster	Wall-clock time for $1T_p$ (hours)
VKI-LS59-1	0	Wall-resolved	3 rd	8.07	106.5	1.5	106.5	440K	54.8M	1	28.25
VKI-LS59-2	0	Refined wall-resolved	3 rd	12.97	15.7	3	15.7	1.07M	137.3M	4	23.37
VKI-LS59-3	0	Wall-model	3 th	5.33	15.7	30	15.7	902K	115.9M	1	5.26

Table 3. Configurations for landing gear acoustic simulation, flow passing time T_p is based on the size of struct length

Number of cells	Order of accuracy	Number of DOFs	Memory consuming (GB)	Maximum(~) local time step(s), dt_{max}	Minimum(~) local time step(s), dt_{min}	Coarsest cell size (mm)	Finest cell size (mm)
11175544	THIRD	862,615,440	301.1	3.34e - 05	6.12e - 10	114.3	0.0018

Case ID	Near wall resolution	Order of accuracy	Speed Up Ratio	Number of nodes on cluster	Wall-clock time for $1T_p$ (hours)
Landing-Gear-1	Partly Wall-resolved	3 rd	34.85	8	14.02