Simulation of a Single-Element Lean-Direct Injection Combustor Using a Polyhedral Mesh Derived from Hanging-Node Elements

Thomas Wey[1] and Nan-Suey Liu[2]

NASA Glenn Research Center

Cleveland, Ohio 44135 USA

Abstract

This paper summarizes the procedures of generating a polyhedral mesh derived from hanging-node elements as well as presents sample results from its application to the numerical solution of a single-element LDI combustor using an open-source version of NCC.

Introduction

The lean direct injection (LDI) concept has the potential for low emissions under operational (compression ratios up to 60:1 and peak temperature 3000 ° F) conditions. For a LDI combustor, most of the air directly enters the combustor through the swirler. In this concept, the liquid fuel is injected from fuel injectors directly into the incoming swirling airflow, and the swirling air stream is used to atomize the injected liquid as well as to promote fuel-air mixing. The flame structure can be very complex and locally range from non-premixed to premixed burning.

Recently, a single-element LDI combustor experiment has been used as a test bed for assessing, further developing and validating the capability of two-phase turbulent combustion modeling and simulation. A series of numerical calculations have been performed by using (1) the time filtered Navier-Stokes (TFNS) methodology and (2) the large eddy simulation (LES) methodology. The sub-grid models employed for turbulent mixing and combustion include the well-mixed model, the linear eddy mixing (LEM) model; the Eulerian filtered mass density function (EUFDF/EUPDF) model, and the flamelet-based model. Results from these methodologies invoking various sub-grid models are summarized in Reference [1], and a more detailed description of the TFNS approach can be found in Reference [2]. It should be pointed out that TFNS is not LES,

[1] Research Aerospace Engineer, Combustion Branch, Member AIAA.
[2] Research Aerospace Engineer, Combustion Branch, Associate Fellow AIAA.

nor hybrid RANS/LES, nor, in general, unsteady Reynolds-averaged Navier-Stokes (URANS). Like the LES, TFNS is capable of capturing the dynamically important, unsteady turbulent flow structures, even when RANS-grade meshes are used. Unlike the LES, the grid resolution and the turbulence model fidelity are not formally linked, therefore, in principle; a grid independent solution can be unambiguously attained in the TFNS approach.

In previous work [3], a mesh-based enhancement for the scalar mixing was proposed. It was speculated there that greater number of the flux-exchange between solution elements, i.e. using polyhedrons as control volumes, may enhance the mixing of the scalar for the modeling of the turbulence-chemistry interaction. In Ref. 3, a honey-comb like polyhedral mesh was generated from a regular unstructured mesh through the processes of refinement, reconnection and agglomeration. A generic swirler combustor refined from a set of all-tetrahedral mesh is shown in Figure 1.

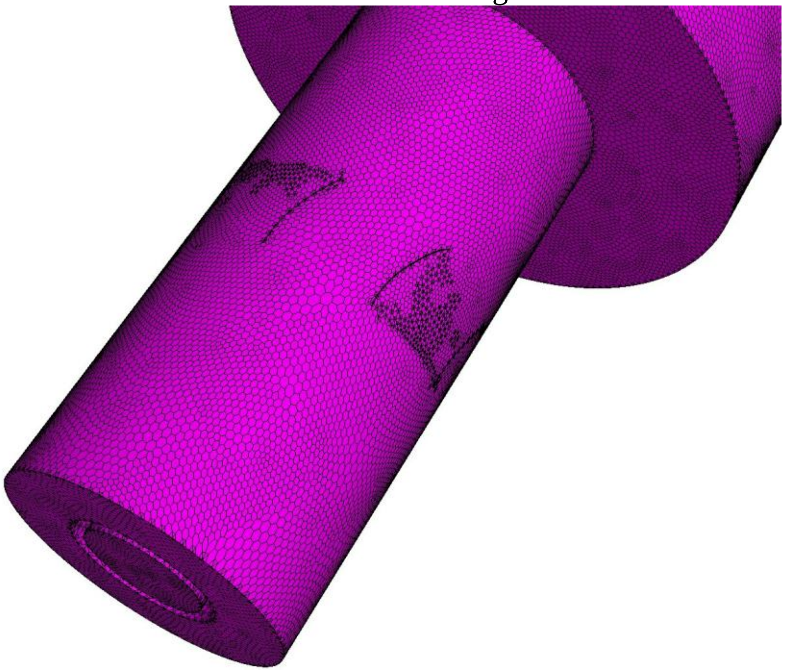

Figure 1 Honey-comb like boundary polygons of a polyhedron generated from a set of all-tetrahedron mesh for a generic swirler combustor.

Another polyhedron mesh for a single-element LDI combustor refined from a set of all-hexahedral mesh is shown in Figure 2. This combustor consists of an air passage with a sixty-degree six-bladed air swirler, a converging-diversion section and a chamber with a square cross-section. The fuel is injected through the center of the swirler and exit at the throat of the venture.

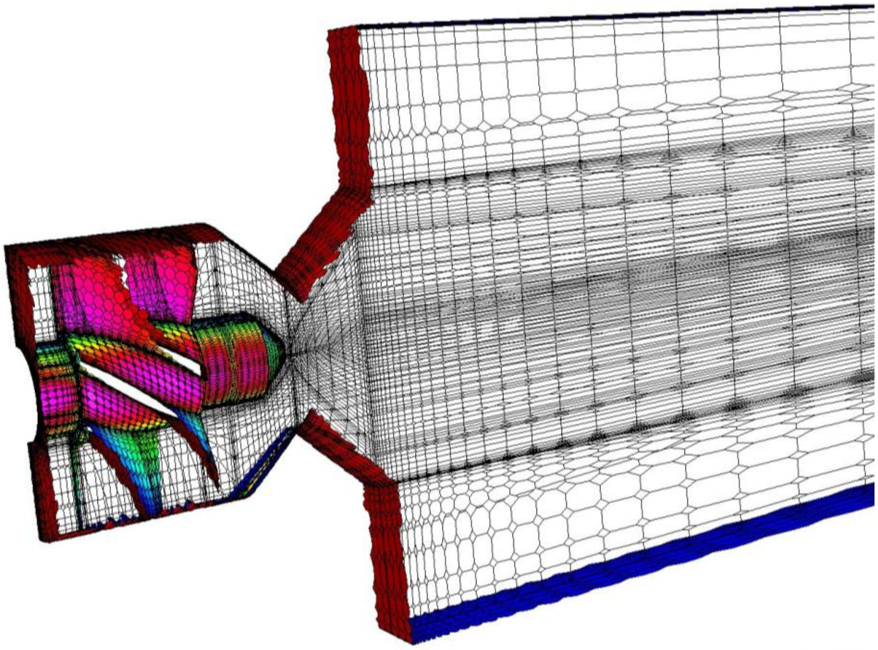

Figure 2 A plane cut of polyhedrons around mid-plane of a single-element LDI combustor.

The effort to pursue the mesh-based enhancement for the scalar mixing is continuing in the present work. However the polyhedral mesh is generated or more precisely post-processed from a set of hanging-node or conforming unstructured mesh.

Generation of Polyhedral Mesh via Refining Simple Regular Mesh

The process starts by applying a 3-h edge refinement to each edge of the mesh, i.e. dividing each edge into three equal-size segments. For a quadrilateral facet, the 3-h refinement is then accomplished by inserting four points into the facet, and subsequently dividing it into nine finer quadrilaterals. Each point is located between the centroid of facet and the corner node of the facet and 1/3 from the centroid. For a triangular facet, the 3-h process needs three additional points, six quadrilaterals and one triangle internally. (Concept of refinement is shown for a two-dimensional mixed triangular and quadrilateral mesh in Figure 3.)

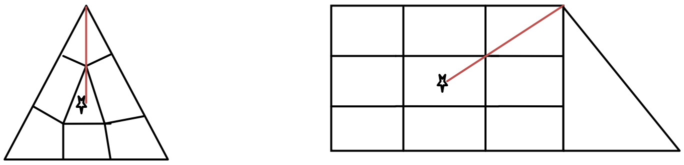

Figure 3 Applying the 3-h refinement to each edge and insert 4 or 3 points into specified facets for two-dimensional cases.

The process is readily extended to three dimensional elements. For a hexahedral element, the 3-h process will add twenty seven hexahedral elements. For a tetrahedral element, it will add four six-face polyhedrons based upon four corner nodes of the tetrahedron; one center tetrahedron; four prisms; six hexahedrons due to center sub-segments of each edge of the tetrahedron. In Figure 4, polyhedrons are derived from a set of tetrahedrons, while in Figure 5; polyhedrons are derived from a set of hexahedrons.

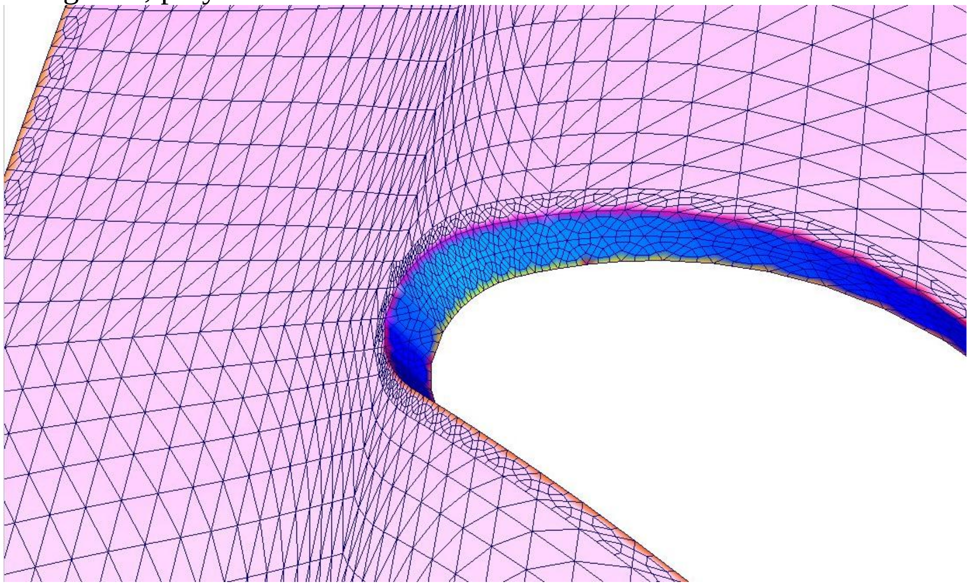

Figure 4 Hanging-node polyhedrons refined from a set of all-tetrahedron rotor mesh.

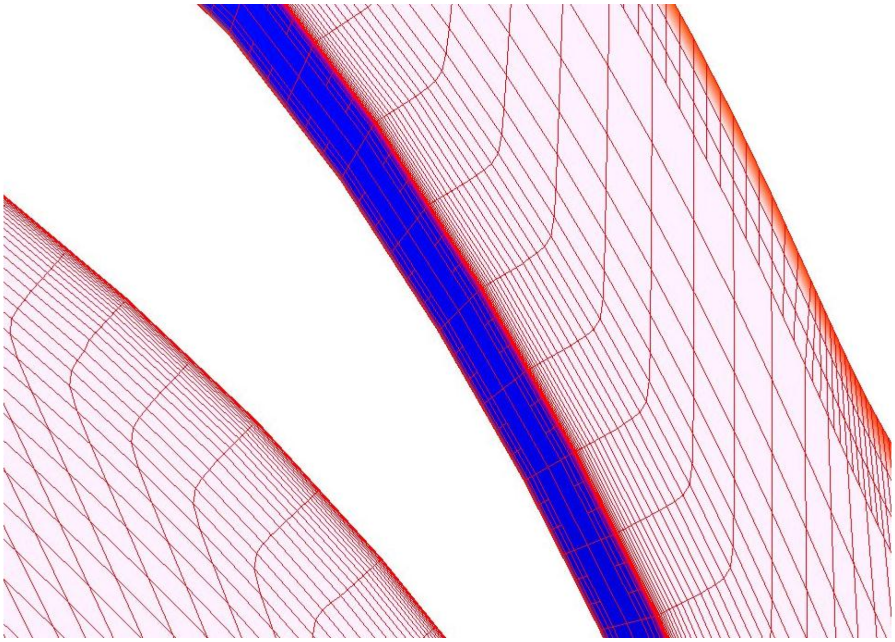

Figure 5 Hanging-node polyhedrons refined from a set of all-hexahedron rotor mesh.

For a prism element, it will add eighteen hexahedrons and three prisms. For a pyramid, it will add one eight-face polyhedron; one center pyramid; four prisms; thirteen hexahedrons.

The “3-h refinement” is chosen here because it can not only achieve the goal of refining a conforming hexahedral mesh to another finer conforming [4] but also hanging-node hexahedral mesh, while with 2-h refinement, it can only create hang-node hexahedrons. A simple example is shown in Figure 6. A mesh, generated by a grid-based meshing method, is composed of coarse hexahedrons, refined conforming hexahedrons and near-wall much finer viscous hang-node hexahedrons.

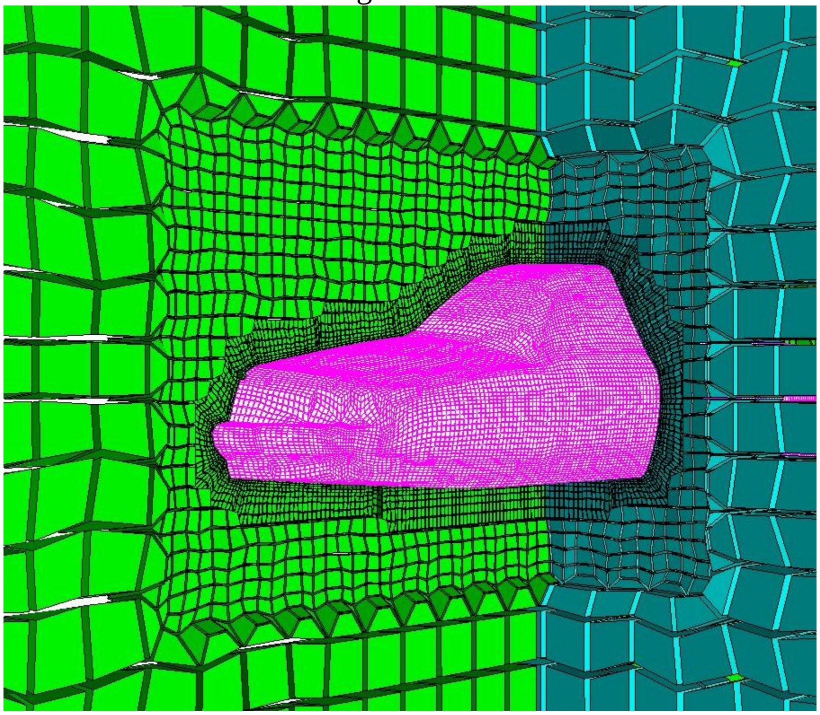

Figure 6 Mixed conforming and hanging-node hexahedron mesh refined from a set of coarse conforming hexahedrons.

All polyhedrons generated by the current method are convex. No special attention is needed for spray droplet search unlike the previous honey-comb like polyhedron mesh in which some cells near the corners and ridges of the boundary faces could be concave. For a concave polyhedron, the centroid of the visible region of the cell is used as the solution center. The visible region of the cell is defined as the union of the points that is visible from any point on the boundary of the cell and vice versa, i.e., it can see any point on the boundary of the cell.

Sample Applications of Hanging-Node Polyhedral Meshes

(1) Refining existing hexahedral mesh

An all-hexahedron mesh describing a single-element LDI combustor is selected for the purpose of demonstration. The geometry of the single-element LDI combustor consists of an air swirler, a convergent-divergent venturi, followed by a rectangular combustion chamber. The fuel injector tip is at the throat of the venturi. The original mesh contains 833,024 hexahedrons and 861,823 nodes. Two meshes are generated by (1) specifying the region to be refined and (2) selecting temperature of the field solutions as the criterion for refinement. The former grid consists of 1,771,832 polyhedrons, 1,854,007 nodes and a center plane cut is shown in Figure 7. The latter grid consists of 1,711,460 cells and 1,802,789 nodes. A center plane cut is shown in Figure 8. The number of solution elements is compatible between these two grids.

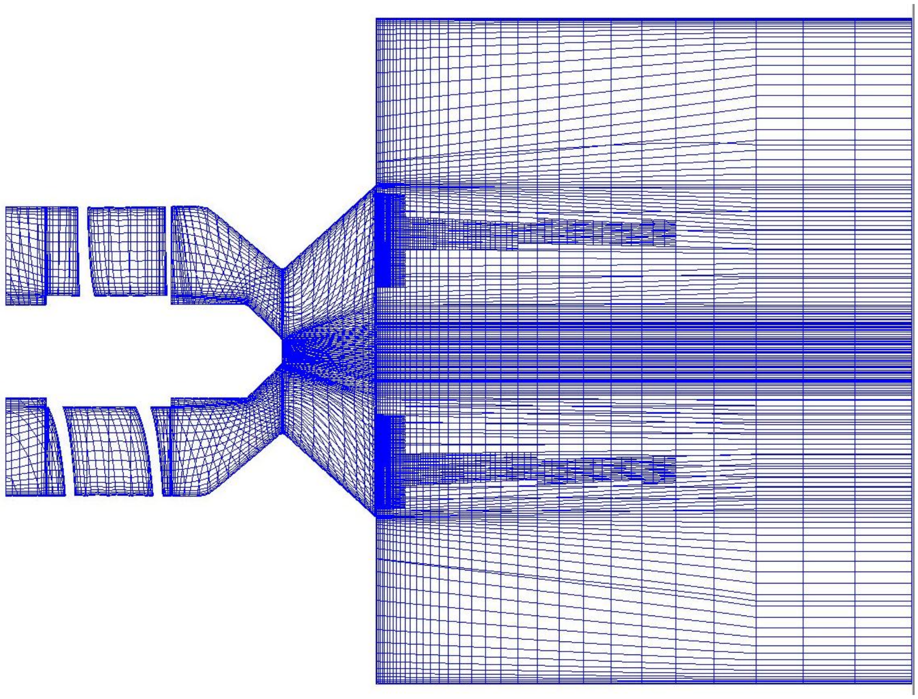

Figure 7 A plane cut of polyhedrons around the mid-plane of the combustor with specified region refined.

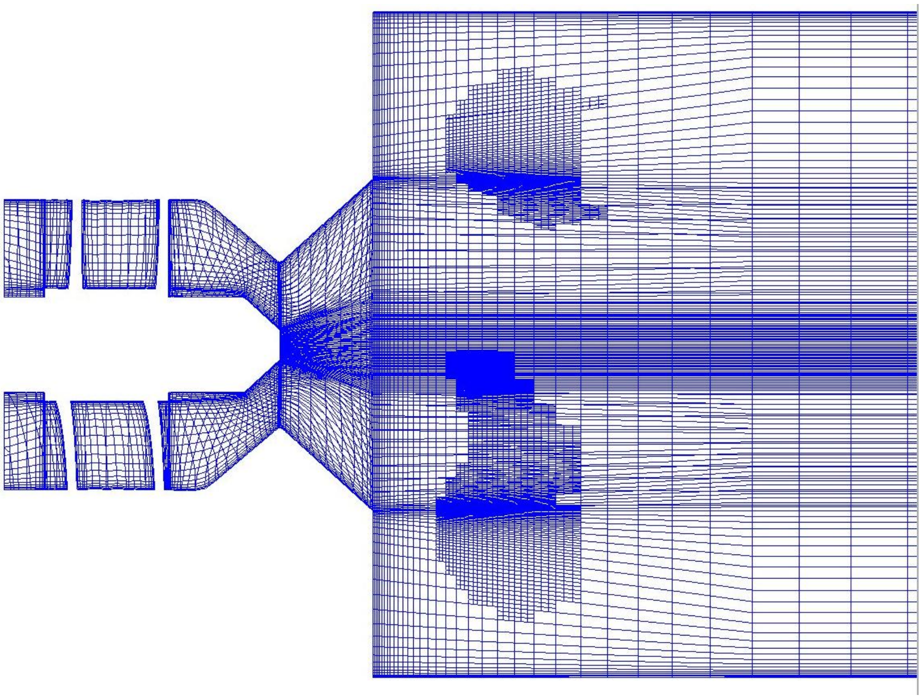

Figure 8 A plane cut of polyhedrons around the mid-plane of the combustor with the temperature as

the criterion scalar for refinement.

A cross-reference pointer system that describes the relations between the parent cells and the children cells has been setup for the further refinements, which are based upon the specifications of different regions and solution criterions. This pointer system also allows the existing solutions to be interpolated from the parent grid to the children grid and vice versa for the iteration of different refinements.

Domain decomposition is a very important subject for massively parallel computing. The cross-reference pointer array can also be used to partition the newly generated hanging-node mesh by assigning all children cells to be the same as their parent's domains. Undoubtedly, this will create unbalanced computation loads but it still keeps the number of faces across the different computational domains to be small, especially if the original parent mesh is partitioned by the highly efficient METIS 4.0.1, the freely available software from University of Minnesota [5].

(2) Multi-phase Reacting Cases

At the inlet of the swirler, the inflow velocity, the static temperature and the density of the gas phase are specified as 20.14 m/s, 294.28 K, 1.19 kg/m^3 respectively. At the outflow boundary, the static pressure is imposed at 101325 Pa. The generalized wall function is applied to solid wall boundaries. The thermal boundary condition for all the solid surfaces of the combustor is set to adiabatic.

In addition to the boundary conditions for the gaseous phase described above, the liquid fuel, C12H23, is injected into the computational domain from a pressure swirl injector. The diameter of the orifice is .0006 m. The fuel atomizer is set at a pressure 110kPa and delivers a flow rate of 0.025 kg/min. The spray has a 90 degree spray angle which is the same as the converging-diverging venture attached to the swirler. An initial droplet size distribution is prescribed to provide the liquid fuel injection condition,

$$\frac{dn}{n} = 4.21 \times 10^6 [\frac{d}{d_{32}}]^{3.5} e^{-16.98(\frac{d}{d_{32}})^{0.4}} \frac{dd}{d_{32}}$$

where n is the total number of the droplets and dn is the number of droplets in the size range between d and $d + dd$. This correlation also requires the specification of Sauter mean diameter, d_{32}, and the number of droplet classes. The equivalence ratio computed from the gaseous and liquid inlet boundary conditions is about 0.72. The adiabatic flame temperature is around 2100 K.

From Figure 9 to Figure 11, time filtered Navier-Stokes (TFNS) results, obtained with the well-mixed combustion model and from using hanging-node polyhedrons derived by specifying regions, are shown for axial velocity, pressure and temperature, respectively.

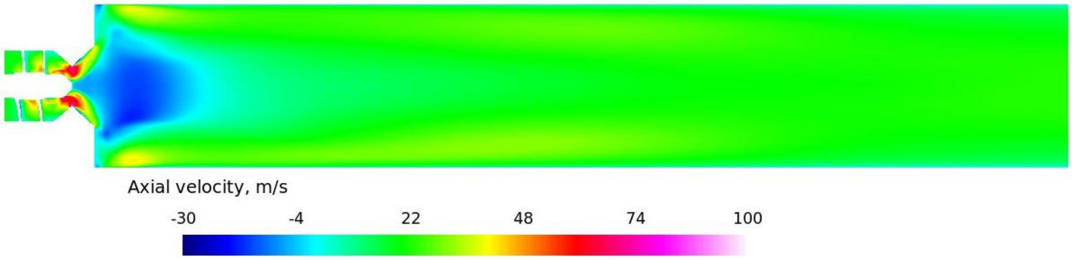

Figure 9 Axial velocity contours of reacting flow at z=0 plane. (1771832 polyhedrons)

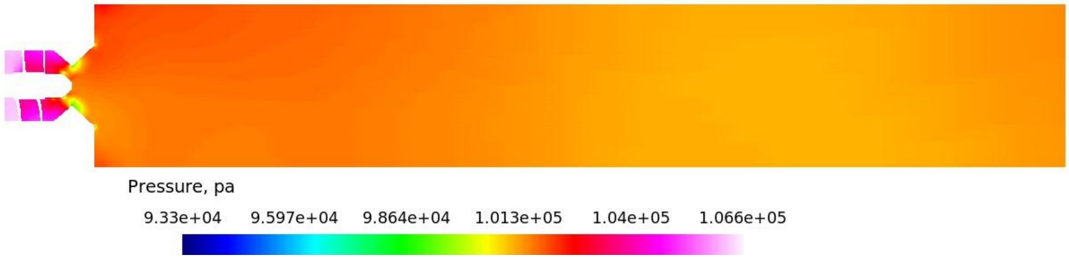

Figure 10 Axial pressure contours of reacting flow at z=0 plane. (1771832 polyhedrons)

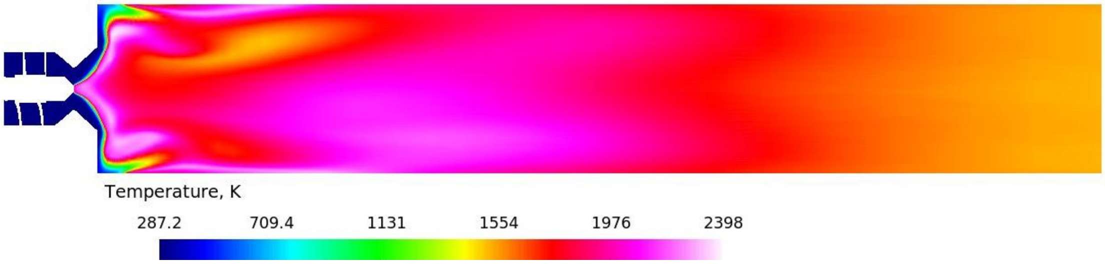

Figure 11 Axial temperature contours of reacting flow at z=0 plane. (1771832 polyhedrons)

From Figure 12 to Figure 13, TFNS results, obtained with the well-mixed combustion model and from using the polyhedral mesh derived by selecting temperature as the criterion, are shown for axial velocity, pressure and temperature in the center plane (i.e. z=0 plane), respectively.

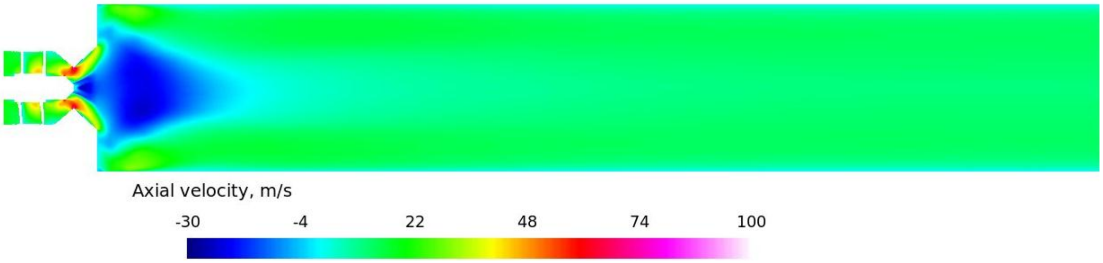

Figure 12 Axial velocity contours of reacting flow at z=0 plane. (1711460 polyhedrons)

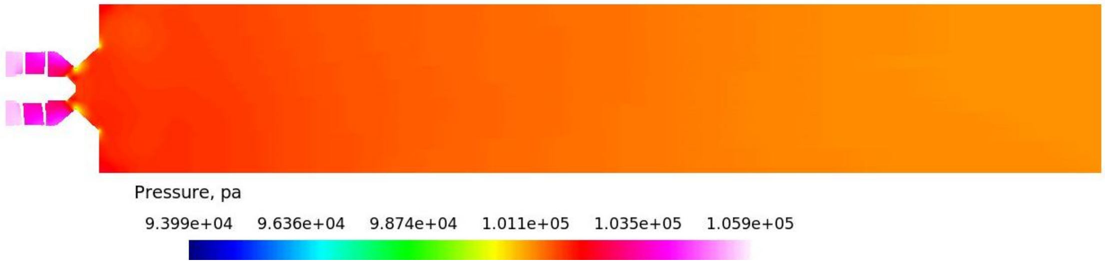

Figure 13 Pressure contours of reacting flow at z=0 plane. (1711460 polyhedrons)

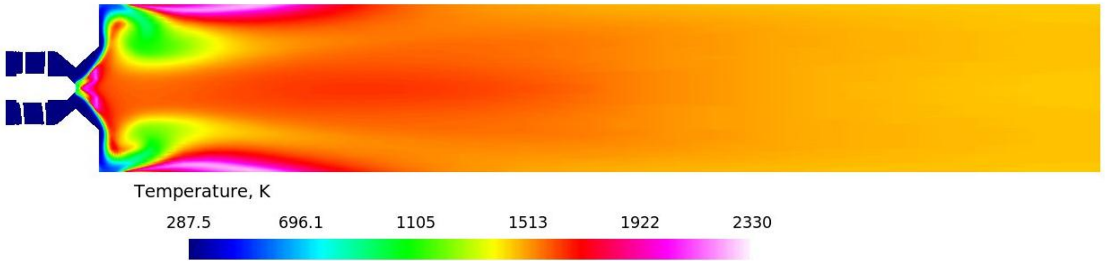

Figure 14 Temperature contours of reacting flow at z=0 plane. (1711460 polyhedrons)

The time-averaged centerline axial velocity and the centerline temperature are presented in Figures 15 and 16, along with the measured data. The discrepancy between the computed values and the experimental data may be due to the user inputs for the spray solve are not well-prepared.

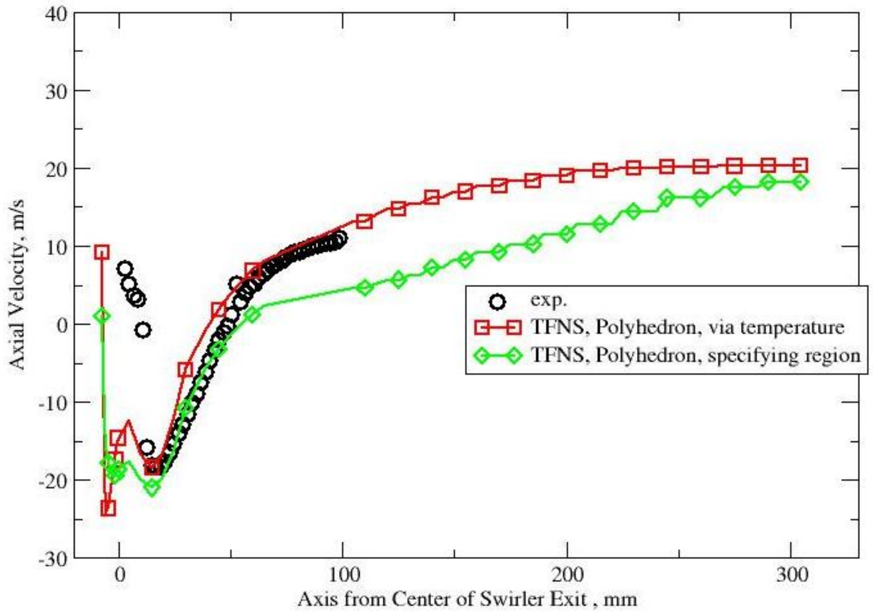

Figure 15 Comparison of the mean axial velocity along the center line.

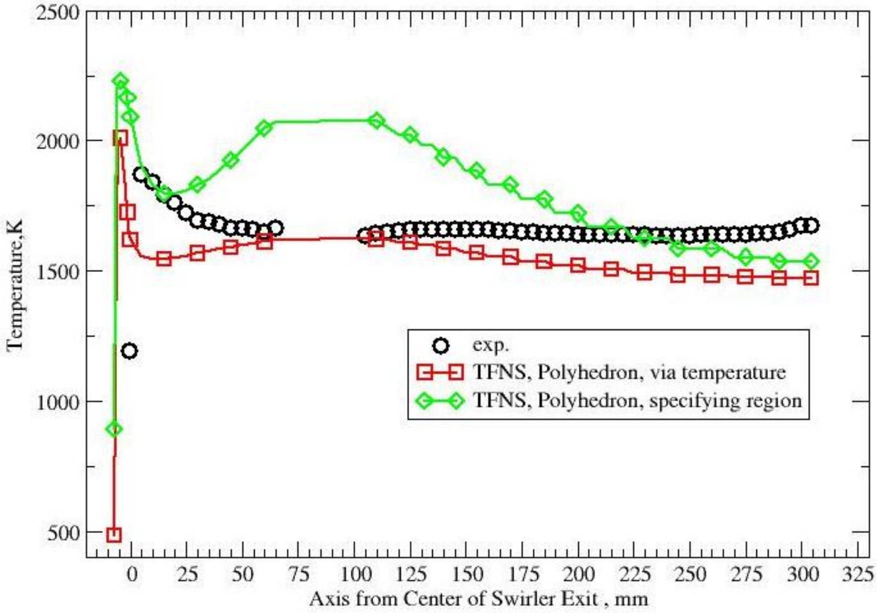

Figure 16 Comparison of the mean temperature along the center line.

Concluding Remarks

Capability of generating and using hanging-node polyhedral mesh for simulations of multi-phase reacting flows is now embodied in a preliminary version of the OpenNCC, which is intended as the self-contained, releasable edition of the National Combustion Code (NCC). A stand-alone single-element LDI combustor is selected for the demonstration. Two meshes are generated from an existing all-hexahedron conforming mesh. It is observed that the latter mesh, generated by selecting temperature distribution for refinement, resulted in better solutions than the former mesh which is refined by specifying the targeted regions. However the discrepancy between the computed results and experimental is noticeable.

Acknowledgements

This work is supported by the NASA Fundamental Aeronautics Program.

References

[1] Liu, N.-S., "Assessment and Improvement of Engineering Simulation for Multiphase Turbulent Combustion in a Lean Direct Injection Combustor," ISABE-2011-1108, 20th International Symposium on Air Breathing Engines, September 12-16, 2011, Gothenburg, Sweden.
[2] Liu, N.-S., Shih, T.-H., and Wey, C.T., "Numerical Simulations of Two-Phase Reacting Flow in a Single-Element Lean Direct Injection (LDI) Combustor Using NCC," NASA/TM-2011-217031, July 2011.
[3] Wey, T., and Liu, N.-S., "Simulation of a Single-Element Lean-Direct Injection Combustor Using Arbitrary Polyhedral Mesh," AIAA 2012-0204, 50th Aerospace Sciences Meeting and Exhibit January 9-12, 2012, Nashville, Tennessee.
[4] Wey, T., "Unstructured Hexahedral Mesh Generation Using the Analogy of the Particle Traces --- Dual Use of Overset Grid Generation Techniques," AIAA 2001-1097, 39th Aerospace Sciences Meeting and Exhibit January 8-11, 2001, Reno, NV
[5] Karypis, G. and Kumar, V. "A Fast and High Quality Multilevel Scheme for Partitioning Irregular Graphs," SIAM Journal on Scientific Computing, Vol. 20, No. 1, pp. 359—392, 1999.