A new approach to modelling the two way coupling for momentum transfer in a hollow-cone spray

Andreas Papoutsakis[1,*], Sergei S. Sazhin[1], Steven Begg[1], Ionut Danaila[2], Francky Luddens[2]

[1] Advanced Engineering Centre, School of Computing, Engineering and Mathematics, University of Brighton, Brighton, BN24GJ, UK

[2] Laboratoire de Mathématiques Raphaël Salem, Université de Rouen Normandie, F-76801, Saint-Étienne-du-Rouvray, France

*Corresponding author: a.papoutsakis@brighton.ac.uk

Abstract

A new approach to modelling the interaction between droplets and the carrier phase is suggested. The new model is applied to the analysis of a spray injected into a chamber of quiescent air, using an Eulerian-Lagrangian approach. The conservative formulation of the equations for mass, momentum and energy transport is used for the analysis of the carrier phase. The dispersed phase is modelled using the Lagrangian approach with droplets represented by individual parcels.

The implementation of the Discontinuous Galerkin method (ForestDG), based on a topological representation of the computational mesh by a hierarchical structure consisting of oct- quad- and binary trees, is used in our analysis. Adaptive mesh refinement (h-refinement) enables us to increase the spatial resolution for the computational mesh in the vicinity of the points of interest such as interfaces, geometrical features, or flow discontinuities. The local increase in the expansion order (p-refinement) at areas of high strain rates or vorticity magnitude results in an increase of the order of the accuracy of discretisation of shear layers and vortices.

The initial domain consists of a graph of unitarian-trees representing hexahedral, prismatic and tetrahedral elements. The ancestral elements of the mesh can be split into self-similar elements allowing each tree to grow branches to an arbitrary level of refinement. The connectivity of the elements, their genealogy and their partitioning are described by linked lists of pointers. These are attached to the tree data structure which facilitates the on-the-fly splitting, merging and repartitioning of the computational mesh by rearranging the links of each node of the tree. This enables us to refine the computational mesh in the vicinity of the droplet parcels aiming to accurately resolve the coupling between the two phases.

Keywords

Droplets, Sprays, Vortex Rings, Discontinuous Galerkin, Adaptive Mesh Refinement.

Introduction

The need to accurately model the interaction between droplets and carrier phase (coupling) in various engineering applications is well known [1]. Although various approaches to modelling this coupling have been suggested [1, 2, 3, 4, 5], this problem is far from being solved. The main focus of this paper is on the description of the new approach to this coupling and the application of this approach to modelling realistic sprays in gasoline-engine-like conditions. This new approach is based on the application of the adaptive mesh refinement in the vicinity of the droplet parcels. The Discontinuous Galerkin (DG) method [6, 7, 8, 9] is used for solving the equations for the carrier phase. The latter method combines high order accuracy with the ability to handle complex geometries described by hybrid unstructured meshes by incorporating a minimal computational stencil. However, the computational efficiency of this method (alongside the spectral volume [8, 10] and spectral difference [8, 11, 12] methods is generally believed to be inferior to more commonly used methods as the Finite Differences (FD) and the Finite Volume (FV)[13, 14] methods.

Solution adaptive refinement strategies of *h/p*–type can reduce the computational time for high resolution simulations of complex flows, using various methods, including the DG method, without compromising numerical accuracy [15, 16, 17, 18, 19, 20]. These refinement strategies result in irregular meshes with hanging nodes and polynomial approximations of different orders across the elements [6]. These issues can be effectively dealt with by the DG method. This makes this method ideal for handling irregular meshes with hanging nodes.

In this paper a new mesh adaptive implementation of the Discontinuous Galerkin methodology is suggested. This implementation allows on the fly local *h/p* refinement and de-refinement of the computational mesh [21]. The development of a new computational code (ForestDG) based on a hierarchical representation of a Forest of binary, quad- and oct- trees, is described. The accuracy and performance of the new code are assessed. The preliminary results of its application to modelling gasoline fuel sprays are described.

Governing equations

The Eulerian-Lagrangian approach is used for the description of the two phases. The droplets are suspended in the carrier gas phase and are modelled using the Lagrangian approach. The carrier phase is modelled as an Eulerian flow field, described by the state vector $\mathbf{U}(\mathbf{x}, t)$, which contains the values of density, momentum and energy at each position of the computational domain $\mathbf{x}$ at time t. The model is developed for a general case of compressible

flow and the effects of turbulence are taken into account using the standard LES approach. The Favre averaging operator $\widetilde{(\cdot)} = \overline{\rho(\cdot)}/\overline{\rho}$ is used for the separation of the small turbulent fluctuations from the large ones. The state vector for the Favre averaged velocity $\mathbf{u}$ and specific energy e is defined as $\widetilde{\mathbf{U}}(\mathbf{x},t) = (\overline{\rho}, \overline{\rho}\widetilde{u}_1, \overline{\rho}\widetilde{u}_2, \overline{\rho}\widetilde{u}_3, \overline{\rho}\widetilde{e})$. The conservation of mass, momentum and energy [22] provides the set of the governing equations for the turbulent compressible flow of the carrier phase, in the following strong conservative form for $\widehat{\mathbf{U}}$:

$$\frac{\partial \widetilde{\mathbf{U}}}{dt} + \nabla \cdot \mathbf{f}_{inv}\left(\widetilde{\mathbf{U}}\right) - \frac{1}{\mathrm{Re}}\nabla \cdot \mathbf{f}_{vis}\left(\widetilde{\mathbf{U}}, \widetilde{\mathbf{\Theta}}\right) = \mathbf{w}_d\left(\widetilde{\mathbf{U}}\right), \ \widetilde{\mathbf{\Theta}} = \nabla \mathbf{f}_{aux}\left(\widetilde{\mathbf{U}}\right) \ , \tag{1}$$

where $\mathbf{w}_d$ is the vector of the source terms stemming from the two way coupling for the momentum and energy transfer between the carrier and the discrete phase, $\mathbf{f}_{inv}$ is the 5×3 tensor of the inviscid fluxes and $\mathbf{f}_{vis}$ is the 5×3 tensor for the viscous fluxes, defined as:

$$\mathbf{f}_{inv} = \begin{bmatrix} \overline{\rho}\widetilde{u}_j \\ \overline{\rho}\widetilde{u}_i\widetilde{u}_j + \overline{p}\delta_{i,j} \\ (\overline{\rho}\widetilde{e} + \overline{p})\widetilde{u}_j \end{bmatrix}, \ \mathbf{f}_{vis} = \begin{bmatrix} 0 \\ 2(\mu + \mu_t)S_{i,j}^* \\ 2(\mu + \mu_t)\widetilde{u}_iS_{i,j}^* \end{bmatrix}, \ \mathbf{w}_d = -\begin{bmatrix} 0 \\ n_d f_{d_i} \\ n_d f_{d_j}\widetilde{u}_j \end{bmatrix}, \ \mathbf{f}_{aux} = \begin{bmatrix} 0 \\ \widetilde{u}_i \\ \widetilde{e} \end{bmatrix} \ . \tag{2}$$

$S_{ij}^* = \frac{1}{2}\left(\frac{\partial \widetilde{u}_j}{\partial x_i} + \frac{\partial \widetilde{u}_i}{\partial x_j}\right) - \frac{1}{3}\delta_{ij}\frac{\partial \widetilde{u}_k}{\partial x_k}$ is the traceless rate of strain tensor related to the viscous stress tensor $\tau_{ij} = 2\mu S_{ij}^*$, with μ being the local non-dimensional viscosity of the gas face. The viscous fluxes are evaluated from $\widetilde{\mathbf{\Theta}}$ which contains the spatial gradients of $\widetilde{\mathbf{U}}$ as $\widetilde{\Theta}_{ij} = \frac{\partial \widetilde{U}_i}{\partial x_j}$. $\widetilde{\mathbf{\Theta}}$ is the divergence of the state vector $\widetilde{\mathbf{\Theta}} = \nabla \widetilde{\mathbf{U}}$ and constitutes a set of auxiliary variables which is solved separately. Equations (1) are the coupled form of the governing equation for $\widetilde{\mathbf{X}} = \left[\widetilde{\mathbf{U}}, \widetilde{\mathbf{\Theta}}\right]$.

The contribution of the sub-grid scales, not accounted by the spatially filtered state vector, is taken into account by the turbulent viscosity term μ_t in the definition of the viscous fluxes in Equation (2). This term is approximated using the standard Smagorinsky model:

$$\mu_t = \widetilde{\rho}(c_s\Delta)^2|\widetilde{S_{ij}}| \ , \tag{3}$$

where c_s is the Smagorinsky constant assumed equal to 0.1, $\Delta = V_m^{1/3}$ is the filter width assumed equal to the characteristic length of the computational element, and $|\widetilde{S_ij}| = \sqrt{2\widetilde{S}_{ij}\widetilde{S}_{ij}}$ is the magnitude of the resolved rate of strain tensor. Equation (1) is non-dimensionalised over the characteristic length of the flow, gas dynamic viscosity μ_g at the ambient conditions. The Reynolds number of the flow is estimated as $\mathrm{Re} = \rho_g cL/\mu_g$, where c is the velocity of sound.

The discrete phase is modelled as parcels of droplets with diameters d_d and velocities $\mathbf{v_d}$. The effect of the dispersed phase on the energy and momentum of the carrier phase is modelled as the source term $\mathbf{w}_d$ in Equation (2). n_d is the droplet number density. The term f_i in Equation (2) is the force acting on each individual droplet in the parcel. Assuming that the flow is Stokesian, the expression for the drag force can be presented as:

$$n_d\mathbf{f}_d = n_d\frac{3\pi d_d\mu}{\mathrm{Re}}\left(\widetilde{\mathbf{u}} - \mathbf{v}_d\right) \ . \tag{4}$$

Note that in some cases considered later $\mathrm{Re}_d > 1$ when the Stokesian approximation is not strictly valid. In these cases the results of our analysis can be used for qualitative but not quantitative description of the process.

The trajectories of droplets are described by the following equations:

$$\frac{d\mathbf{x}_d}{dt} = \mathbf{v}_d, \quad d\mathbf{v}_d = \mathbf{f}_d dt + \left(\frac{k_t}{\tau_t}\right)^{1/2}d\mathbf{W}_t \ , \quad \mathbf{f}_d = \frac{18\mu}{\mathrm{Re}d_d^2\rho_l}\left(\widetilde{\mathbf{u}} - \mathbf{v}_d\right) \ . \tag{5}$$

The term $d\mathbf{W}_t$ represents the increment of the Wiener process, which introduces the stochastic component for the motion of the dispersed phase to model the effect of the unresolved turbulent scales to the droplet trajectories. In the case of equilibrium, the subgrid turbulent kinetic energy term k_t is evaluated from the 'production equals dissipation' assumption as $k_t = 2\Delta^2 c_s^{4/3}\widetilde{S}_{ij}\widetilde{S}_{ij}$. The term τ_t is the time scale of the interaction between the droplet and the turbulent motions modelled as $\tau_t = \tau_d^{1.6}\left(\frac{\sqrt{k_t}}{\Delta}\right)^{0.6}$ [23].

Discretisation of the equations

We consider the discretisation of the computational domain Ω into N elements E_m ($\Omega = \cup E_m$). A weak formulation of the governing equations is derived by multiplying the conservative forms of these equations with a test function $w(\mathbf{x})$ and integrating them over the element. In the Galerkin context, the test function is taken from the same set of polynomial basis functions as used for the interpolation of the state vector $\widetilde{\mathbf{X}}$. The interpolated distribution $\mathbf{X}_h^m$ for $\widetilde{\mathbf{X}}$ is defined for each element E_m as the weighted sum of N_p polynomial basis functions:

$$\mathbf{X}_h^m = \sum_{i=1}^{N_p}\mathbf{c}_i^m(t)b_i(\mathbf{x}) \ , \text{for } m = 1, N_p \ , \tag{6}$$

where p is the maximum degree of the basis functions. In this expansion, the solution coefficients $\mathbf{c}_i(t)$ are the degrees of freedom. $b_i(\mathbf{x})$ is the tensor product of the Legendre polynomial basis functions in the three spatial dimensions. The integral formulation of Equations (1) is expressed as:

$$\int_{E_m} b_i \frac{\partial \mathbf{U}_h^m}{\partial t} dE + \oint_{S_m} b_i \mathbf{f}\left(\mathbf{U}_h^m\right) \cdot \mathbf{n} dS - \int_{E_m} \nabla b_i \cdot \mathbf{f}\left(\mathbf{X}_h^m\right) dE = \int_{E_m} b_i \mathbf{w}_d^m dE \quad \text{for} \quad i = 1, N_p, \ m = 1, N , \quad (7)$$

$$\int_{E_m} b_i \mathbf{\Theta}_h^m dE = \oint_{S_m} b_i \mathbf{f}_{aux}\left(\mathbf{U}_h^m\right) \cdot \mathbf{n} dS - \int_{E_m} \nabla b_i \cdot \mathbf{f}_{aux}\left(\mathbf{X}_h^m\right) dE \quad \text{for} \quad i = 1, N_p, \ m = 1, N , \quad (8)$$

providing a set of $N_p \times N$ equations for $\mathbf{c}_i^m(t)$.

The surface integrals are defined on the surface of the element $S_m = \partial E_m$ with $\mathbf{n}$ defined as the outward normal unit vector. In the weak formulation presented in Equation (7), the flux $\mathbf{f}$ represents both the viscous and inviscid fluxes in Equation (1) as $\mathbf{f} = \mathbf{f}_{inv} - (1/\text{Re})\, \mathbf{f}_{vis}$; the flux $\mathbf{f}_{aux}$ was defined in Equation (2). In the DG context we do not require the continuity of the interpolated variables across the element faces. Thus the values of $\mathbf{U}_h^m$ and $\mathbf{\Theta}_h^m$ on the faces S_m are defined twice. The conservation of fluxes at the boundaries of the elements infers from the approach to the evaluation of the surface integrals used in our analysis. In the Local Lax-Friedrichs (LLF) scheme for the evaluation of viscous fluxes the signs of surface sides of the elements (refered to as minus $(-)$ or plus $(+)$) are introduced.

Calculating the fluxes from one of the two sides for the adjoin elements guarantees the conservation properties of the scheme. Specifically for the LLF scheme the surface integrals for $\mathbf{f}_{vis}$ and $\mathbf{f}_{aux}$ in Equation (1) are evaluated from the opposite sides, enhancing the stability of the scheme, and reducing the stiffness of the problem since:

$$\mathbf{f}_{vis} = \mathbf{f}_{vis}\left(\mathbf{U}^{-\,m}_{h}, \mathbf{\Theta}^{-\,m}_{h}\right) , \mathbf{f}_{aux} = \mathbf{f}_{aux}\left(\mathbf{U}^{+\,m}_{h}\right) . \quad (9)$$

The inviscid flux $\mathbf{f}_{inv}$ is evaluated from the mean value of the variables on the two face sides $\{\mathbf{U}_h^m\}$, where an artificial diffusion term, proportional to the jump of the fluxes on the bounding surface $[[\mathbf{U}_h^m]]$, is introduced via the equation:

$$\mathbf{f}_{inv} = \mathbf{f}_{inv}\left(\{\mathbf{U}_h^m\}\right) + \frac{c}{2}[[\mathbf{U}_h^m]] , \quad (10)$$

where c is defined as $c_i = \max\left(|f'_{inv}(U_i^-)|, |f'_{inv}(U_i^+)|\right)$ in the local Lax-Friedrichs scheme.

The volume and surface integrals in Equations (7) and (8) are defined in the physical space. The integrals are evaluated in the transformed domain for the canonical elements using the Legendre-Gauss quadrature rule and the Jacobian of the transformation [24].

Mesh representation

Unstructured, conforming grids are represented using a finite serial addressing of the cells with each cell determined by the address of it's vertices. The connectivity of the cells is then defined by appointing the numerical addresses of the neighbouring cells to each face. In our case, the serial array of the cells is substituted by a graph of elements forming a forest of nodes.

Each node refered to as tree node contains all the pointers needed to define a cell by its relative relations rather than its address in a global addressing system. This scheme provides the versatility of adding or removing nodes and manipulating the relations between the nodes by maintaining the relative relations of the node without interfering to the addressing scheme of the remaining nodes. A node is accessed by a dynamic linked list structure, naming an arbitrary node as the "first" node of the mesh. By assigning a "next" node for every cell we can go through all cells of the grid. This is achieved by advancing to the "next" node each time starting from the "first" as shown in Figure 1(Left). The tree in our implementation is a data structure that contains all necessary information needed for the
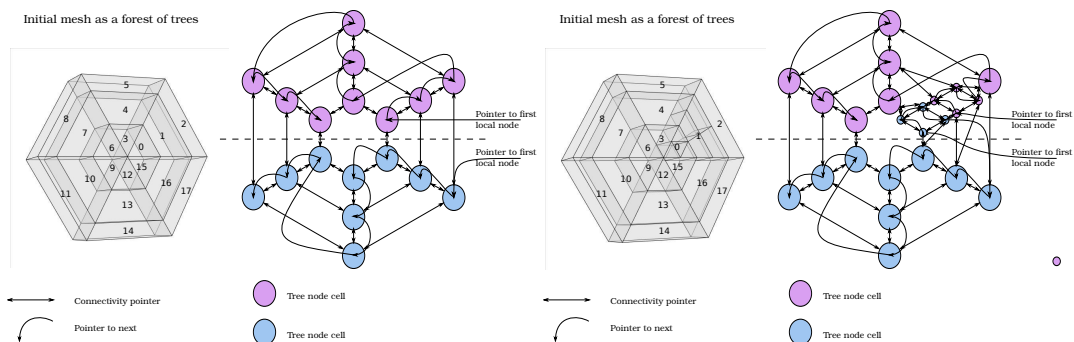

Figure 1. Left: Example of hybrid unstructured mesh and the corresponding graph. **Right:** Prismatic and hexahedral cells are split into four cells; the nodes of the adapted mesh are repartitioned by introducing the new nodes to the local element lists and the connectivity pointers are re-defined.

definition of the relative relations of the cell and also its geometrical characteristics, i.e. the nodes, the edges and

the faces. A node of the graph can be split furnishing a tree of nodes while a single node is perceived as a unitarian tree. An example of a developed forest is shown in the figure 2.

The actual solution vector $\mathbf{X}$ is stored at a special data structure of the tree node, named leaf. The leaf contains the characteristic coefficients of the basis functions used for the description of the conservative variables, the element Jacobian matrix, the mass matrix and the matrices used for the calculation of the derivative and basis values at the face and volume quadrature points. Eventually, the leaf data structure contains all the memory consuming information that describes the actual field. In the ForestDG implementation, the initial unrefined mesh is transformed
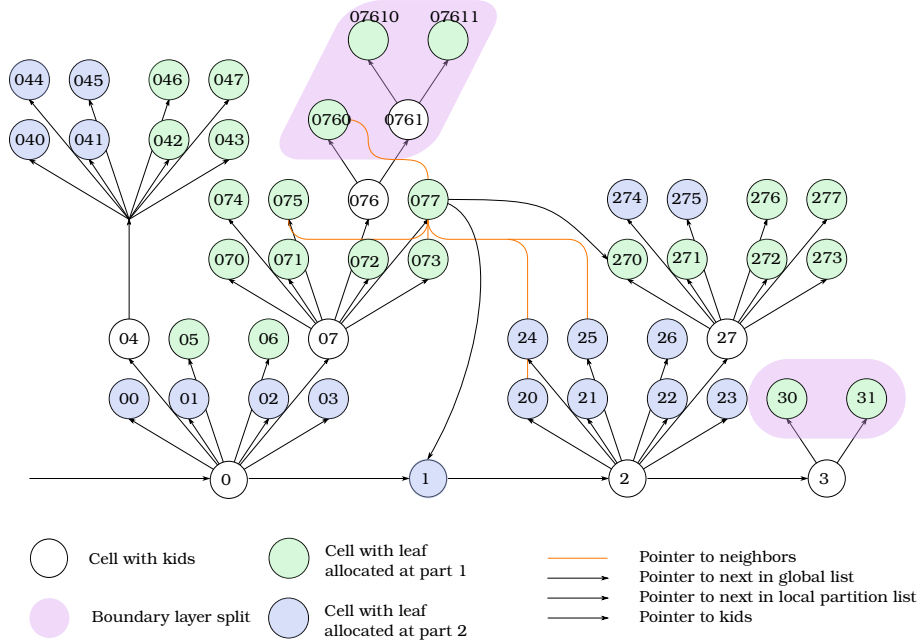

Figure 2. A forest of oct-trees representing a tree dimensional adapted topology.

into a graph representation and the nodes are partitioned using the METIS [25] graph domain decomposition library. In the event of splitting a tree, a number of new nodes, allocated by the `kids[ikid]` pointer array, are created
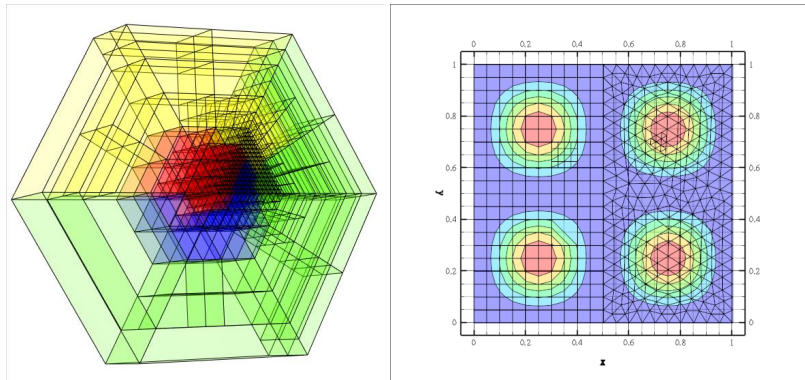

Figure 3. Left: Oct tree splitting of a hybrid unstructured mesh. Node 0 is split to level 5 resulting in a 4736 element grid starting from 18 initial nodes; blue, green, yellow and red represent the mapping of the four different partitions. **Right**: Examples of projecting the solution during merging and spliting cells for the manufactured solution used for testing the order of accuracy of the discretisation.

depending of the type of splitting maintaining the hierarchical structure of the forest of trees. A quad tree type of splitting results in four children, an oct-tree splitting results in eight children. The parent node is removed from the linked list that controls the accessing of the cells and replaced by the children nodes. The parent node is removed from the linked list and replaced by the kids.

When a cell is split into a number of kids or when a set of kids merge to a bigger cell, the solution has to be projected to the new cells. In order to achieve this, the position of the existing quadrature points must be identified in the new geometry as shown in the Figure 4 (Left). Since the basis of the interpolation is evaluated in the computational space which spans from -1 to 1, the coordinates for the quadrature points need to be transformed in the computational space. The transformation of the ith canonical coordinate for each kid η_i^{kid} to the canonical coordinates of the parent η_i has the general form:

$$\eta_i = a_{1,2,3}^{i,kid}\eta_1^{kid}\eta_2^{kid}\eta_3^{kid} + b_{1,2}^{i,kid}\eta_1^{kid}\eta_2^{kid} + b_{1,3}^{i,kid}\eta_1^{kid}\eta_3^{kid} + b_{2,3}^{i,kid}\eta_2^{kid}\eta_3^{kid} + c_1^{i,kid}\eta_1^{kid} + c_2^{i,kid}\eta_2^{kid} + c_3^{i,kid}\eta_3^{i,kid} + d^{i,kid} \; , \quad (11)$$

where the coefficients $a^{kid}, b^{kid}, c^{kid}$ and d^{kid} are calculated for each kid depending on its position and shape in the canonical space as shown in the left column of the Figure 4 (Left). For prismatic cels the above expression simplifies to:

$$\eta_1 = b_{1,2}^{1,kid} \eta_1^{kid} \eta_2^{kid} + c_1^{1,kid} \eta_1^{kid} + c_2^{1,kid} \eta_2^{kid} + d^{1,kid}, \quad \eta_2 = c_2^{2,kid} \eta_2^{kid} + d^{2,kid}, \quad \eta_3 = c_3^{3,kid} \eta_3^{kid} + d^{3,kid}, \tag{12}$$

and for hexahedral cells is expressed as:

$$\eta_1 = c_1^{1,kid} \eta_1^{kid} + d^{1,kid}, \quad \eta_2 = c_2^{2,kid} \eta_2^{kid} + d^{2,kid}, \quad \eta_3 = c_3^{3,kid} \eta_3^{kid} + d^{3,kid}. \tag{13}$$

The 3×3 system in Equation (12) is linear quadratic while Equations (13) are linear. Both systems can be reversed analyticaly and we can obtain the reverse transformation. The Newton-Raphson method is used for reversing the system in Equation (11) for the tetrahedral elements shown in Figure 4 (Left). The forward transformation is used for projecting the solution of a parent to the quadrature points of a kid and the reverse transformation is used for projecting the solution of a parent at the quadrature points of a kid. The above transformations are also used for the evaluation of the fluxes for non-conformal faces. Non-conformalities arise due to changes of the expansion order because of p-refinement or due to hanging nodes from h-refinement or form a combination of both as shown in Figure 4 (Right). A smoothing pass assures either 2:1 or 1:2 non-conformalities, (1:4 in three dimensions). No such restriction is applied for the change of the order p.
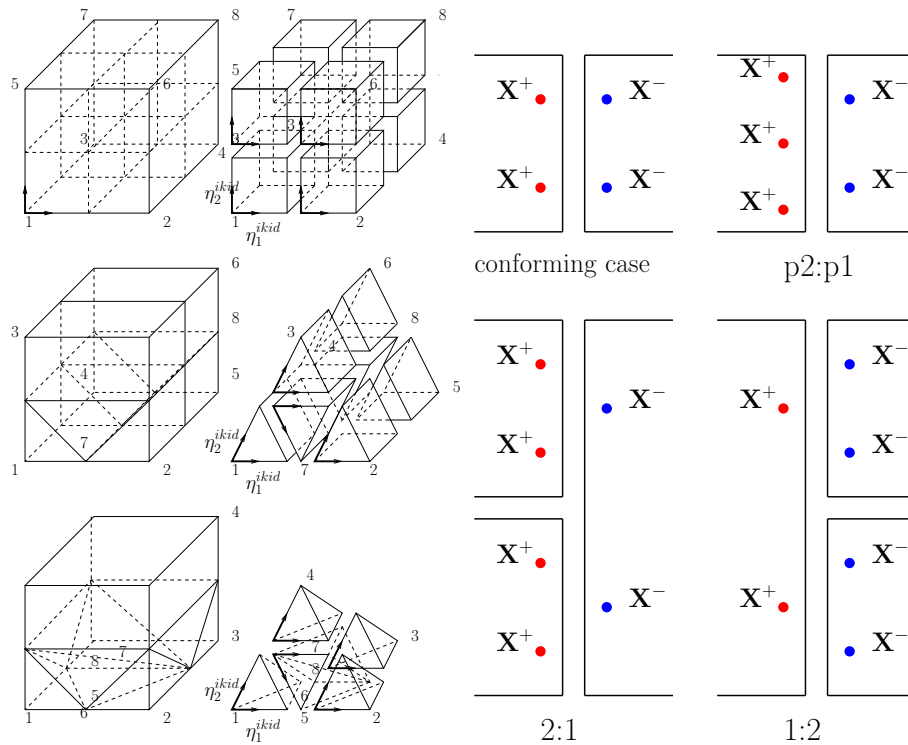

Figure 4. Left: Canonical space transformations for hexahedral, prismatic and tetrahedral elements. **Right:** Types of non-conformalities encountered in h/p adaptive cases.

Hexahedral elements are split into eight self similar children so that one node is positioned on the vertex of the higher level cell, three nodes are positioned at the midpoints of the adjacent edges, three nodes are positioned on the tree adjacent faces centroids and a final node is positioned at the higher level hexahedral centroid as shown in the Figure 5(Left). The numbering of the children follows the numbering of the higher level cell vertices so that the 1st kid is adjacent to the 1st vertex of the cell an so on. For prismatic elements, splitting leads to self similar elements and the cell numbering follows the vertex numbering of the higher level cell with the exception of the 7-th and 8-th children which are placed along the core of the prism with reversed orientations, as shown in the Figure 5 (Right). Having split the cells to the required resolution, the connectivity of the cells needs to be remapped. This is achieved by tracking the neighbouring cells, neighbouring faces and the orientation of these faces. An example of connectivity tracking for two high level adjacent cells A and B is shown in the Figure 5, in which cells A and B have the same orientation. The relative orientation of two cells is characterized by two parameters: the face numbers and the relative angle between the faces. For the i-th kid of the higher level cell A the neighbouring cell is opposite to the i-th cell B, which is described by the following pointer assignment:

$$A-> kids[i]-> neig[ifc_A] = B-> kids[Op[i, ifc_A]], \tag{14}$$

where ifc_A is the face of the element A that is shared with B. The array Op is a transformation operator that provides the kid opposite to kid i relatively to face ifc_A. In the general case, however, the neighbouring ancestral cells do
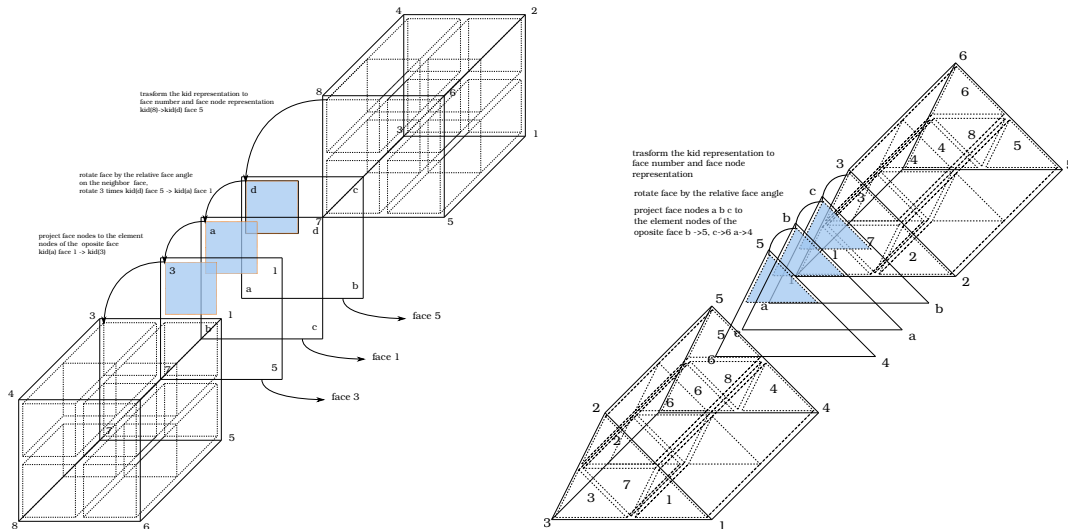
Figure 5. Left: Tracking of a neighbouring element and neighbouring face at a hexahedral to hexahedral connectivity instance. **Right:** Tracking of a neighbouring element and neighbouring face at a prism to prism connectivity instance.

not have the same orientation. As a result, the connectivity is calculated in steps of transformations shown by the following pointer assignment:

$$A- > kids[i]- > neig[ifc_A] = B- > kids[Op[Tr[i, ifc_B, ang_B], ifc_A]] ,$$ (15)

In this expression, operator Tr remaps the kid address i for element B into a coordinate system parallel to the coordinate system of element A. This transformation aims to align the orientation of the neighbouring cells.

Assessment of the accuracy of the discretisation

The method of manufactured solutions is used for the investigation of the order of accuracy of the DG discretisation. A two-dimensional manufactured solution shown in Figure 3 (Right) is used. The flow is assumed to be unidirectional in the z-axis and is described as:

$$\rho = 1.0; \quad p = 1.0; \quad T = 1.0; \quad u = 0, \ v = 0, \ w(\mathbf{x}, t) = w_0 \left(1 - \cos(4\pi x)\right)\left(1 - \cos(4\pi y)\right) .$$ (16)

Introducing this flow field into the governing equations we obtain analytical expressions for the residuals. These residuals are introduced as source terms in the governing equations to sustain the manufactured solution. The error of the numerical solution, compared with the exact solution in Equation (16), is expressed by the L_2 norm which is computed as [26]:

$$L_2 = \left(\frac{1}{V_\Omega} \sum_{m=1}^{N} \int_{E_m} \left(\mathbf{u} - \mathbf{u}^{\text{exact}}\right)^2 dV\right)^{1/2} .$$ (17)

The maximum order of accuracy is $p + 1$ for basis functions which are polynomials of degree p.

As can be seen in Figures 6, the L_2 norm of the error between the manufactured solution and analytical expression in Equation (16) reduces to the second order of the mesh resolution. Results similar to those shown in Figures 6, but for the third and fourth order of accuracy of discretisation when bases with p=2 and p=3 basis are used, are presented in Figure 7. As follows from this figure, the expected order of accuracy is achieved in this case.

Application to gasoline fuel spray

The implementation of this modelling approach has been applied to the case of the simulation of a high-pressure, hollow-cone, gasoline, fuel spray utilised in a modern spray-guided combustion system. An accurate prediction of the spray and gas phase characteristics, prior to ignition of the mixture, plays a key role in determining the optimal engine operating conditions for maximum combustion efficiency in highly lean, and stratified mixtures. Experimental observations of the fuel spray were conducted in a quiescent chamber of fixed volume. The ambient gas conditions were 20 degC and 1 bar. The piezoelectric fuel injector was mounted in a vertical position at the top of the chamber. Measurements of the spray shape (geometry and thickness of plume) and its droplet size and velocity distributions were carried out using high-speed photography and Phase Doppler Anemometry, respectively. The experimental set-up and measurement procedure have been described in [5].

The fuel used in our analysis is iso-octane injected at a pressure of 100 bar for a duration of 1 msec. Mass flow rate $\dot{m}$ (measured experimentally using a rate tube) increased linearly from zero up to a maximum value of $\dot{m} = 30$g/s in 0.1 msec. During the following 0.8 msec the mass flow rate remains constant and decreases to zero during the last 0.1 msec. Every 10 time steps of the simulation 10 droplet parcels are released at an angle of $42.5°$ relative to the axis of symmetry with a spread (divergence of wall thickness) of $6°$. Droplet velocities were inferred from the mass
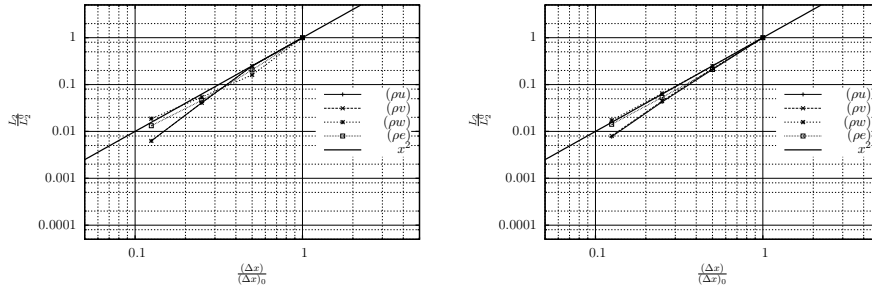
EDITORIAL UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Figure 6. Relative L_2 error for the momentum components and the energy, versus mesh resolution (h) for a second order accurate discretisation (p=1). **Left:** Hexahedral elements. **Right:** Prismatic elements.
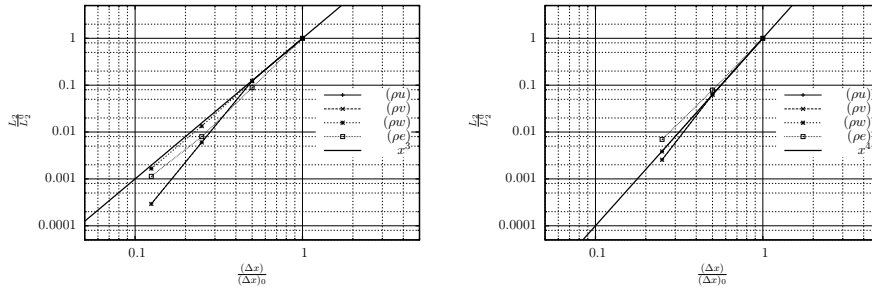

Figure 7. Relative L_2 error for the momentum components and the energy, versus mesh resolution for hexahedral elements. **Left:** Third order accurate discretisation (p=2). **Right:** Fourth order accurate discretisation (p=3).

flow rate of the injected fuel and prelimanary PDA measurements. The number of individual droplets in the parcel was calculated from the same mass flow rate assuming that the droplet diameters are equal to d_d. These diameters for different parcels were in the range $3\,\mu$m to $9\,\mu$m. The most probable droplet diameter, used in our simulations ($6\,\mu$m) corresponded to typical d_d observed in the experiments. The preliminary results of the application of the new code to the analysis of sprays in gasoline engine-like conditions are shown in Figure 8. The results presented in this figure agree with the results of experimental observations of these sprays [5]. Depending on the distance from the orifice, the computational mesh is refined 2 to 3 times at a sector around the spray, 3 to 4 times at areas of high strain rate and 4 to 5 times for cells that contain droplets. The order is increased to p2 (3rd order) in areas of high vorticity.
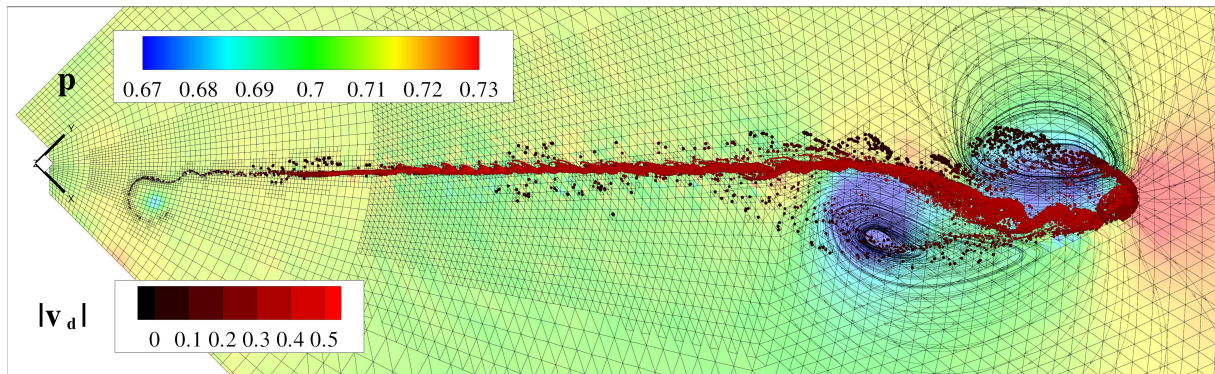

Figure 8. Droplet distribution at $t = 1.10$ msec. A uniform initial distribution of droplets with diameters from 0.003 to 0.009 mm was assumed. Size of the circles corresponds to droplet sizes and their colours correspond to the non-dimensional droplet velocities $|\mathbf{v_d}|$ normalised by the speed of sound). Contour colour shows the distribution of non-dimensional pressure (normalised by ρc^2).

Conclusions

Preliminary results of the development of a new model for the analysis of the interaction between droplets and the carrier phase are described. This model uses the implementation of the Discontinuous Galerkin method (ForestDG), based on a topological representation of the computational mesh by a hierarchical structure consisting of oct- quad- and binary trees. Adaptive mesh refinement (h-refinement) used in the analysis enables us to increase the spatial resolution for the computational mesh in the vicinity of the points of interest such as interfaces, geometrical features, or flow discontinuities. The local increase in the expansion order (p-refinement) at areas of high strain rates or vorticity magnitude results in an increase of the order of the accuracy of discretisation of shear layers and vortices. In our analysis, the initial domain consists of a graph of unitarian-trees representing hexahedral, prismatic and tetrahedral elements. The ancestral elements of the mesh are split into self-similar elements allowing each tree to grow branches to an arbitrary level of refinement. The connectivity of the elements, their genealogy and their

partitioning are described by linked lists of pointers. These are attached to the tree data structure which facilitates the on-the-fly splitting, merging, and repartitioning of the computational mesh by rearranging the links of each node of the tree. This enables us to refine the computational mesh in the vicinity of the droplet parcels aiming to accurately resolve the coupling between the two phases.

The accuracy of the new code is assessed and preliminary results of the implementation of the code to the analysis of a hollow-cone spray in gasoline engine-like conditions are presented.

Acknowledgements

Nomenclature

c	Speed of sound [m s^{-1}]	$\mathbf{w}_d$	Two-way coupling source term [-]
c_s	Smagorinsky constant [-]	$\mathbf{X}$	State and auxiliary variables vector [-]
d_d	Droplet diameter [-]	Greek symbols	
Δ	Filter width [-]	$\boldsymbol{\Theta}$	Auxiliary variables vector [-]
e	Energy [-]	μ_g	Dynamic viscosity of carrier phase [kg m^{-2} s^{-1}]
$\mathbf{f}$	Flux vector [-]	μ_t	Turbulent viscosity [-]
$\mathbf{f}_d$	Stokes force [-]	ρ	Density [-]
k_t	Subgrid TKE [-]	ρ_g	Density of carrier phase [kg m^{-3}]
L	Reference length [m]	Subscripts	
n_d	Number density [-]	aux	Auxilary
Re	Reynolds number [-]	d	Droplet
S_{ij}	Rate of strain tensor [-]	g	Gas phase
S_{ij}^*	Traceless rate of strain tesnor [-]	inv	Inviscid
u	Carrier phase velocity [-]	t	Turbulent
$\mathbf{U}$	State vector [-]	vis	Viscus
v	Droplet parcel velocity [-]		

References

[1] Sazhin, S. S. *Droplets and Sprays*, p. 345. Springer, London, 2014.

[2] Sazhina, E. M., Sazhin, S. S., Heikal, M.R., Babushok, V. I., and Johns, R. J. R., 2000, *Combustion Science and Technology*, 160(1):317–344.

[3] Rybdylova, O., Osiptsov, A. N., Sazhin, S. S., Begg, S., and Heikal, M., 2015, *International Communications in Heat and Mass Transfer*, 15(1):685–686.

[4] Rybdylova, O., Qubeissi, M. A., Braun, M., Crua, C., Manin, J., Pickett, L. M., de Sercey, G., Sazhina, E. M., Sazhin, S. S., and Heikal, M., 2016, *International Communications in Heat and Mass Transfer*, 76:265–270.

[5] Zaripov, T. S., Gilfanov, A. K., Begg, S. M., Rybdylova, O., Sazhin, S. S., and Heikal, M. R., 2017, *In Press, Atomization and Sprays*.

[6] Cockburn, B., Karniadakis, G. E., and Shu, C. W., 2000, *Lecture Notes in Computational Science and Engineering*, (11):3–50.

[7] Ekaterinaris, J.A., 2005, *Progress in Aerospace Sciences*, 41(3-4):192–300.

[8] Wang, Z., 2007, *Progress in Aerospace Sciences*, 43(3-4):1–41.

[9] Cockburn, B. and Shu, C.-W., 1998, *J. Comput. Phys.*, 141(2):199–224.

[10] Liu, Y., Vinokur, M., and Wang, Z., 2006, *J. Comput. Phys.*, 212(2):454–472.

[11] Liu, Y., Vinokur, M., and Wang, Z., 2006, *J. Comput. Phys*, 216(2):780–801.

[12] Yu, M., Wang, Z.J., and Liu, Y., 2014, *J. Comput. Phys.*, 259(15):70–95.

[13] Cheng, J. and Shu, C. W., 2007, *J. Comput. Phys.*, 227(2):567–1596.

[14] Jiang, G. S. and Shu, C. W., 1996, *J. Comput. Phys.*, 126:202–228.

[15] Berger, M. J. and Colella, P., 1989, *J. Comput. Phys.*, 82(2):64–84.

[16] Djaffar, A. Y., Wagdi, G., and Habashi, W. G., 1997, *AIAA Journal*, 35(5):1294–1302.

[17] Flaherty, J.E., Krivodonova, L., Remachle, J. F., and Shepard, M. S., 2002, *Finite Elements in Analysis and Design*, 38(10):889–908.

[18] Yelash, L., Müller, A., Lukáčová-Medviďová, M. Giraldo, F.X., and Wirth, V., 2014, *J. Comput. Phys.*, 268(1).

[19] van der Vegt, J. J. W. and van der Ven, H., 1998, *Journal of Computational Physics*, 141:46–77.

[20] van der Vegt, J. J. W. and van der Ven, H., 2002, *Journal of Computational Physics*, 182:546–58.

[21] Papoutsakis, A., Panourgias, K. T., and Ekaterinaris, J. A., 2014, *AIAA Paper 2014-0068, Aerospace Sciences Meeting, AIAA SciTech Forum*.

[22] Anderson, J. *Computational Fluid Dynamics*. Computational Fluid Dynamics: The Basics with Applications. McGraw-Hill Education, 1995.

[23] Jones, W.P., Marquis, A.J., and Vogiatzaki, K., 2014, *Combustion and Flame*, 161(1):222 – 239.

[24] Karniadakis, G.E. and Sherwin, S. *Spectral/hp Element Methods for CFD*. Oxford University Press, 2nd edition, 2003.

[25] Karypis, G. and Kumar, V., 2009 MeTis: Unstructured Graph Partitioning and Sparse Matrix Ordering System, Version 4.0. `http://www.cs.umn.edu/~metis`.

[26] Wang, Z.J., Fidkowski, K., Abgrall, R., Bassi, F., Caraeni, D., Cary, A., Deconinck, H., Hartmann, R., Hillewaert, K., Huynh, H. T., Kroll, N., May, G., Persson, P., Leer, B.van, and Visbal, M., 2013, *International Journal for Numerical Methods in Fluids*, 72(8):811–845.