

Fire Dynamics Simulator: Advances in simulation capability for complex geometry

Marcos Vanella^{1,2}, Randall McDermott¹, Glenn Forney¹, and Kevin McGrattan¹

¹Fire Research Division, National Institute of Standards and Technology

²Department of Mechanical and Aerospace Engineering, The George Washington University

Abstract

We describe our effort on the enhancement of fire simulation capability around complex geometry within the Fire dynamics Simulator (FDS). An efficient algorithm based on spatial intersection and polygon/polyhedral reconstruction is developed to define a boundary conforming cut-cell mesh around geometrically complex objects in three dimensions. These target geometries do not need to lay on the FDS grid planes. A conservative finite volume unstructured discretization of the energy and scalar transport equations (chemical species) on this cut-cell grid is implemented. We explore the implicit solution for transport of chemical species, addressing stability issues arising in small cells close to the object, as well as explicit time integration by means of linking small cells to larger neighbors. Velocity reconstruction on faces near the solid surfaces is done using a direct forcing immersed boundary method (IBM). Finally, tight conservation defined by the pressure Cartesian unstructured discretization and solution is also evaluated, as a counterpart to the global pressure solution method present in FDS. A description of the work done on the above areas is provided, together with a simulation example.

1 Introduction

The Fire dynamics simulator (FDS) is employed in evaluation and design of fire protection systems, forensic work, and study of wild land fires, among others. It is a multiphysics, time-accurate fire model based on a buoyant flow low-Mach approximation, that evolves explicitly the scalar transport, momentum and energy equations, as well as providing combustion, solid degradation and radiation solutions [10, 6]. Eddy resolving techniques are implemented within the gas phase, in particular eddy viscosity based Large Eddy Simulation (LES). Further, the domains employed in three dimensions are decomposed in structured Cartesian staggered mesh blocks. FDS allows for the definition of obstacles within these meshes using a Cartesian-mesh level of approximation (*Lego-block* geometries). That is, objects surfaces either need to, or are forced to match Cartesian faces present on the fluid grid. This is a natural choice for regular compartments, where walls are generally aligned with grid planes. On the other hand, there has been recent interest in providing more accurate geometry descriptions within the tool in order to satisfy demand for simulation on architecturally challenging structures (i.e. curved walls and roofs), train and airplane cabins and cargo compartments, and wildfires on rough terrains. In this manuscript we describe the work being done to enhance FDS, targeting higher geometrical fidelity of computations on these scenarios. Given the requirement that the underlying FDS grid system should be maintained, and complex geometry capability should be viewed as an *add-on*, a natural path forward is to employ schemes where the spatial discretization is modified on cells being intersected by the geometry. These correspond to what are known as cut-cell or embedded boundary methods, where the cut-cells are in general polyhedra resulting on either gas or solid phases of Cartesian cells that have been split by the boundary surface. See Figs. 1a, 1b for a schematic.

The numerical solution of the governing equations on the cut-cell region essentially leads to an unstructured discretization, with cells of different sizes and shapes. Besides enhancing geometrical fidelity, the cut-cell scheme adopted should enforce strict conservation on scalars along the simulation. The finite volume discretization of scalar transport equations in the cut-cell region is adopted towards this end. As we shall see, cut-cells have their own numerical issues, one of which is the *small cell* problem for explicit integration. This is in essence, the timestep limitation they impose to advance the numerical integration under stability constraints (i.e. CFL and Von Neuman conditions).

In this work we explore implicit time integration [4, 8] on the cut-cell region as a manner to alleviate these constraints, as well as the classical approach of linking small cells to larger neighboring counterparts, solving the problem effectively on cells resulting from said unions. In FDS the thermodynamic divergence constraint factored from the sensible enthalpy evolution equation is used as a proxy for energy [9]. Conservation for energy in the cut-cell region requires the unstructured cut-cell spatial discretization of the different terms in the divergence expression. On the other hand, it is convenient to perform the fractional step projection of the momentum equations [3, 11] in the underlying Cartesian grid, while using immersed boundary (IBM) reconstruction of velocities [5, 2]. The link between the cut-cells thermodynamic divergence and the underlying Cartesian cell divergence is provided by a divergence integral equivalence argument. This is a fundamental step in coupling the scalar and energy evolution with the momentum equations. Finally, the use of an unstructured Cartesian Poisson solver as the means to impose exactly the no-penetration condition on immersed boundaries (a classical problem of IBM) is explored. Other areas where the presence of cut-cells leads to modification of the current FDS implementation are combustion and radiation. In combustion, the batch reactor model volumes are the ones computed on cut-cells on near wall regions, and for radiation the radiative transport equation finite volume solver needs to be modified on said region. In the following sections, a succinct description of the work done on the above areas is provided, together with an example showcasing our vision for integrating this simulation capability within FDS.

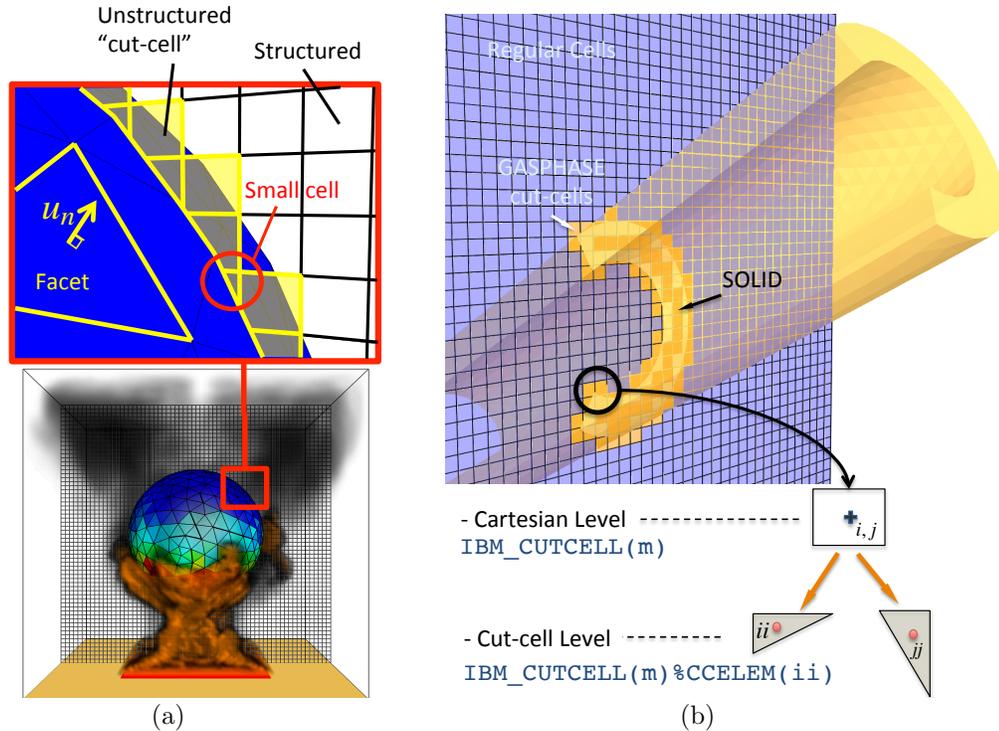


Figure 1: Regular Cartesian and cut-cell grids: (a) Cut-cell region defined around triangulated spherical surface. (b) Cut-cells defined from C-shaped beam, and 2 refinement level interpretation.

2 Cut-cell definition engine for complex geometry

The computational geometry engine performs the task of defining all geometric entities (namely cut-faces and cut-cell volumes) which together with their geometric properties are used in discretizing the fire model equations. As requirements for this code unit, we seek for it to be robust, able to handle general bodies and grid sizes, and also parallelizable. Ideally we want an efficient scheme that would provide small overhead in cases of moving objects, where regridding and interpolation should take place at every time level. Towards this end, a method based on body-Cartesian plane intersections definition has been implemented. The immersed bodies surface meshes, or Lagrangian meshes, are defined as triangulations typical for tessellations for finite element discretizations on solid media. A two-level grid refinement hierarchy is employed for geometric entities of interest in the gas phase, namely edges, faces and cells. The coarse level is defined by the Cartesian entities, whereas the fine level is defined by the cut-cell or unstructured counterparts. See Fig. 1b.

Then, for each Eulerian mesh in FDS and for every immersed object the procedure for populating the data structures containing cut-faces and cut-cells is summarized as follows:

1. Body-plane intersection elements (segments, triangles) are defined for all Cartesian grid planes. Intersections along surface triangles also defined. See Figs. 2a,b.
2. Cut-faces on Cartesian planes are defined by joining segments. Same for cut-faces along surface triangles. Area properties are computed for all cut-faces. See Fig. 2c.
3. Working by Cartesian cell, cut face sets are found for each cut-cell volume (Fig. 2d). Volume properties are computed efficiently for the resulting polyhedra from their corresponding faces, using the divergence

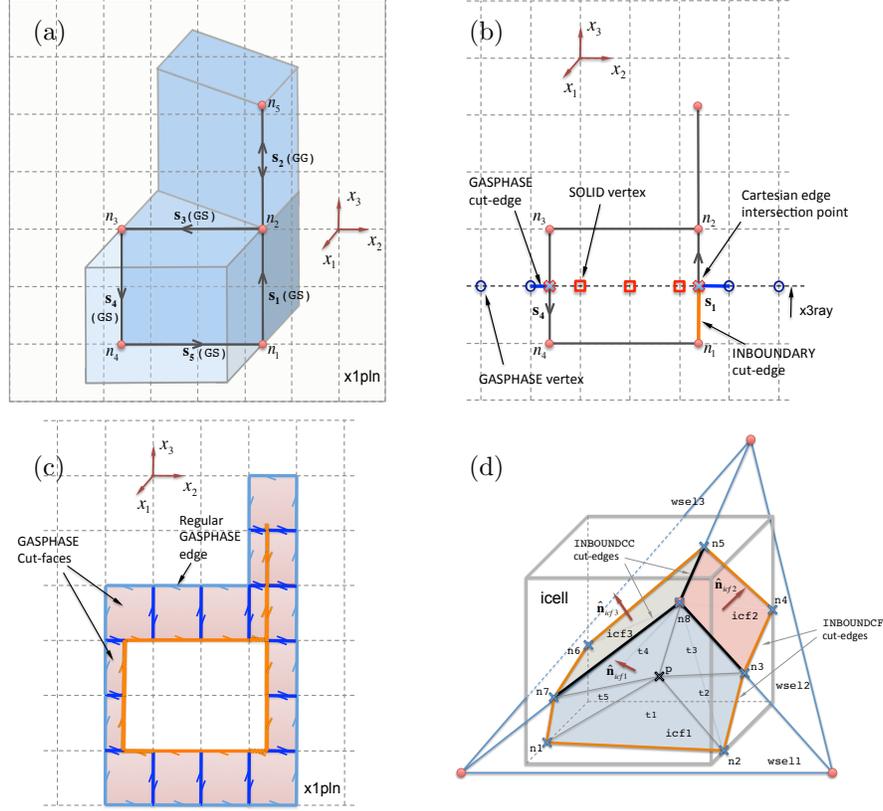


Figure 2: Computational geometry scheme: (a) Intersection segments are found for Cartesian plane $x1pln$ and immersed object. (b) Cartesian cell vertices on the plane are tagged regarding if they belong to gas or solid regions. Intersection points, boundary and gas phase cut-edges are defined on the plane. (c)-(d) Cut-edges are used to define cut-face on $x1pln$, and laying in the boundary. (d) Cut faces are used to defined cut-cells on gas phase.

theorem. For example, the volume of a cut-cell individualized by the single index ii can be defined by:

$$V_{ii} = \int_{\Omega_{ii}} \nabla \cdot \mathbf{F}_v d\Omega \quad (1)$$

where we take $\mathbf{F}_v = x\hat{\mathbf{i}}$, such that $\nabla \cdot \hat{\mathbf{F}}_v = 1$ as needed. Using the divergence theorem:

$$V_{ii} = \int_{\partial\Omega_{ii}} \mathbf{F}_v \cdot \hat{\mathbf{n}} dA = \sum_{k=1}^{nf_c} (\hat{\mathbf{i}} \cdot \hat{\mathbf{n}}_{ii,k}) \int_{\Omega_k} x dA = \sum_{k=1}^{nf_c} Ix_k \quad (2)$$

where nf_c is the number of faces (cut-face or regular) which are boundary of cut-cell ii . Then having area integrals Ix_k already computed on cut-faces on the previous step 2, the volume computation for cut-cell ii is trivial.

4. Finally, interpolation stencils are computed for face and cell centroids to be used in immersed boundary velocity reconstruction as well as in thermal boundary conditions imposition.

3 Scalar transport

Consider a set of gaseous, reacting chemical species $\alpha = 1, \dots, N$ flowing on a given spatial domain $\Omega \in \mathbb{R}^n$, $n = 2, 3$, with boundary $\partial\Omega$, parameterized by an Eulerian reference frame. These species are transported on a given point \mathbf{x} in space with a mass weighted average velocity $\mathbf{u}(\mathbf{x}, t)$. The scalar mass transport equations are:

$$\frac{\partial \rho Y_\alpha}{\partial t} + \nabla \cdot (\rho Y_\alpha \mathbf{u}) = -\nabla \cdot (\mathbf{J}_\alpha) + \dot{m}_\alpha''', \quad \alpha = 1, \dots, N \quad (3)$$

Assuming Fick's Law for binary diffusion with respect to a background most abundant species, we define the diffusive mass flux

$$\mathbf{J}_\alpha = -\rho D_\alpha \nabla Y_\alpha = -\left(D_\alpha \nabla (\rho Y_\alpha) - \frac{D_\alpha}{\rho} \nabla \rho (\rho Y_\alpha) \right) \quad (4)$$

with this last expression in the previous we arrive at the format of equations we will use in the cut-cell region to advance scalars:

$$\frac{\partial \rho Y_\alpha}{\partial t} + \nabla \cdot (\rho Y_\alpha \mathbf{u}') = \nabla \cdot (D_\alpha \nabla (\rho Y_\alpha)) + \dot{m}_\alpha''', \quad \alpha = 1, \dots, N \quad (5)$$

where $\mathbf{u}' = \mathbf{u} + \frac{D_\alpha}{\rho} \nabla \rho$. The spatial discretization of these equations on the cut-cell region follows the finite volume method (FVM) (i.e. [7]). Referring to Fig. 3a, and considering averaged quantities for cells and faces, the volume integral of equations (5) on a given cut-cell ii , leads to advective and diffusive terms of the form:

$$\int_{\Omega_{ii}} \nabla \cdot (\rho Y_\alpha \mathbf{u}') d\Omega = \sum_{k=1}^{n_{fc}} (\rho Y_\alpha \mathbf{u}')_k \cdot \hat{\mathbf{n}}_{ii,k} A_k \quad (6)$$

$$\int_{\Omega_{ii}} \nabla \cdot (D_\alpha \nabla (\rho Y_\alpha)) d\Omega = \sum_{k=1}^{n_{fc}} (D_\alpha \nabla (\rho Y_\alpha))_k \cdot \hat{\mathbf{n}}_{ii,k} A_k \quad (7)$$

where the average fluxes on faces are approximated by the values estimated a face centroids. Also,

$$(\rho Y_\alpha \mathbf{u}')_k = \left[\overline{(\rho Y_\alpha)_k}^{fl} \mathbf{u}_k + \overline{(\rho Y_\alpha)_k}^{lin} \left(\frac{D_\alpha}{\rho} \nabla \rho \right)_k \right] \quad (8)$$

where k refer to the face centroid value of the quantity, and $\overline{(\rho Y_\alpha)_k}^{fl}$ is the flux limited interpolation of ρY_α to face k (in our tests Godunov, or MINMOD flux limiters). Also, $\overline{(\rho Y_\alpha)_k}^{lin}$ is the linear interpolation of said quantity to face k .

As commented in the introduction, it is well known that cut-cell methods pose a significant time constraint when used with explicit time integration methods. This is so because, inevitably for general problems, cut-cells arise whose small-size severely penalizes the time step. For explicit time integration, each cell on the gas phase, including cut-cells, needs to meet CFL and Von Neuman stability constraints. Several different ways have been proposed in the literature to deal with this problem, i.e. cell merging, mixing or linking methods. In general, these lead to ad hoc selection procedures for surrounding cells, having to deal with many special cases, and in some cases solution deterioration close to the boundary.

We explore the use of implicit integration of scalars to handle the small cell-problem in a robust manner. Nevertheless, to reduce the cost of building discretization matrices and solving linear systems of the size of the number of unknowns for the whole domain, we only solve implicitly our problem in a band of cells surrounding the cut-cells. See Fig. 3b. In general, the explicit region is advanced first. For the integration to be conservative, the normal flux (advective+diffusive) of (ρY_α) used in the explicit-implicit (*EXIM*) boundary has to be the same for implicit and explicit region time advancement. This constraint provides the non-homogeneous flux boundary condition to be used in integrating the implicit region.

So, assuming an Explicit Euler method is used for the *EX* region of Fig. 3b, and dropping the reactive

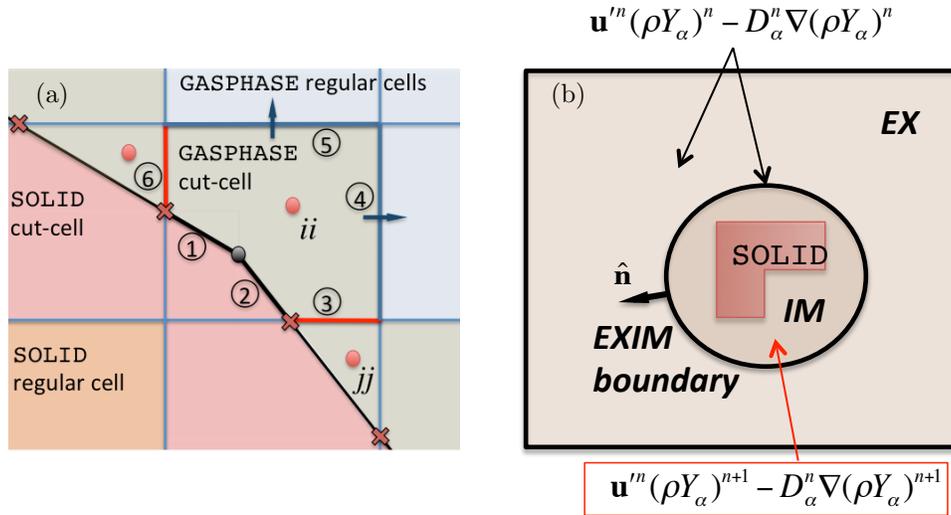


Figure 3: Regular Cartesian and cut-cell discretization: (a) Regular and cut faces on the gas phase region as well as solid boundary define cut-cell ii in 2D. (b) Schematic of Explicit and Implicit time integration regions. The $EXIM$ boundary is located two Cartesian cells away from the immersed object, and used to define boundary fluxes for the implicit time integration.

term \dot{m}_α''' , equation (5) is time discretized as:

$$\frac{(\rho Y_\alpha)^{n+1} - (\rho Y_\alpha)^n}{\Delta t} = -\nabla \cdot (\mathbf{u}^n(\rho Y_\alpha)^n - D_\alpha^n \nabla(\rho Y_\alpha)^n) \quad , \quad in \Omega_{EX} \quad (9)$$

Using the flux $\mathbf{u}^n(\rho Y_\alpha)^n - D_\alpha^n \nabla(\rho Y_\alpha)^n$ as boundary condition in the $EXIM$ boundary, the implicit region IM around the immersed solid can be integrated adopting the Implicit Euler method, for example:

$$\frac{(\rho Y_\alpha)^{n+1} - (\rho Y_\alpha)^n}{\Delta t} = -\nabla \cdot (\mathbf{u}^m(\rho Y_\alpha)^{n+1} - D_\alpha^m \nabla(\rho Y_\alpha)^{n+1}) \quad , \quad in \Omega_{IM} \quad (10)$$

where \mathbf{u}^n, D_α^n correspond to the last known values of velocity, density and diffusivity (the coefficients have been linearized in time), consistent with what is available on the predictor step of the FDS SSPRK2 Runge-Kutta integration scheme. The corrector step of such a method can be combined with again an implicit Euler step using the last known values of velocity, density and diffusivity (predicted values), or using the trapezoidal rule, which was seen to maintain the time accuracy of SSPRK2 in the implicit region for tests on the heat equation. Implicit integration requires building the FVM advection and diffusion sparse matrices, and solve a system of equations for all the IM region unknowns for each species and each time integration substep. Very small cut-cells can make this system ill-conditioned. Therefore, cells that have a volume very small ($< 10^{-4}$) compared to the Cartesian cell volume are linked to neighbors, as described below. Currently we use the Pardiso direct solver from the Intel MKL library [1] to perform the linear system solution task.

The capability to integrate explicitly the cut-cell region IM has also been implemented using cell linking; that is, using the same number of unknown for cut-cells and their linked neighboring cells. Then, building the condensed matrix vector systems to the new independent unknowns in the same way as done for implicit integration, the cut-cell region can be integrated using the FDS SSPRK2 scheme. As a thumb rule, all cut cells whose volume is less than 95% the Cartesian cell volume should be linked to Cartesian neighboring cells in order to preserve stability.

4 Energy and momentum coupling

Starting from the sensible enthalpy evolution equation, the divergence of the velocity field can be factored as [9]:

$$\begin{aligned}
(\nabla \cdot \mathbf{u})^{th} &= \left[\frac{1}{\rho c_p T} - \frac{1}{\bar{p}} \right] \frac{\partial \bar{p}}{\partial t} + \frac{w \rho_0 g_z}{\rho c_p T} \\
&+ \frac{1}{\rho c_p T} [\dot{q}''' - \nabla \cdot \dot{\mathbf{q}}'' - \mathbf{u} \cdot \nabla(\rho h_s)] \\
&+ \frac{1}{\rho} \sum_{\alpha} \left(\frac{\bar{W}}{W_{\alpha}} - \frac{h_{s,\alpha}}{c_p T} \right) [\dot{m}_{\alpha}''' - \nabla \cdot \mathbf{J}_{\alpha} - \mathbf{u} \cdot \nabla(\rho Y_{\alpha})]
\end{aligned} \tag{11}$$

where h_s is the local sensible enthalpy, \dot{q}''' is the heat release rate due to combustion, $\dot{\mathbf{q}}''$ is the heat flux, sum of conduction, convection and radiation, and \bar{p} and T are the background pressure and local temperature on the Low Mach number approximation adopted. The mixture specific heat at constant pressure and molecular weight are $c_p = \sum_{\alpha=1}^N c_{p,\alpha} Y_{\alpha}$ and $\bar{W} = \left(\sum_{\alpha=1}^N Y_{\alpha} / W_{\alpha} \right)^{-1}$ respectively, built from the individual species counterparts as defined. Finally, w, ρ_0, g_z on the stratification term are respectively, the local vertical velocity, density for standard conditions (depends on vertical coordinate), and gravity acceleration. We call this divergence expression the thermodynamic divergence $(\nabla \cdot \mathbf{u})^{th}$. The projection scheme for velocities enforces on each cell of the spatial discretization this final divergence on the discrete velocity field. We use the finite volume version of the previous equation to obtain the volume integrated target thermodynamic divergence on each cut cell volume, and to recompute the thermodynamic divergence on regular gas phase cells adjacent to cut-cells. For cut-cell ii the corresponding discrete volume integrated expression is:

$$\begin{aligned}
(\nabla \cdot \mathbf{u})_{ii}^{th} V_{ii} &= \left[\frac{1}{(\rho c_p T)_{ii}} - \frac{1}{\bar{p}_{ii}} \right] \frac{\partial \bar{p}_{ii}}{\partial t} V_{ii} + \frac{w_{ii} \rho_0 g_z}{(\rho c_p T)_{ii}} \\
&+ \frac{1}{(\rho c_p T)_{ii}} \left[\dot{q}''' V_{ii} - \sum_{k=1}^{n_{fc}} \dot{\mathbf{q}}''_{ii,k} \cdot \hat{\mathbf{n}}_{ii,k} A_k - \overline{\mathbf{u} \cdot \nabla(\rho h_s)} V_{ii} \right] \\
&+ \frac{1}{\rho_{ii}} \sum_{\alpha} \left(\frac{\bar{W}}{W_{\alpha}} - \frac{h_{s,\alpha}}{c_p T} \right)_{ii} \left[\dot{m}_{\alpha}''' V_{ii} - \sum_{k=1}^{n_{fc}} \mathbf{J}_{\alpha,ii,k} \cdot \hat{\mathbf{n}}_{ii,k} A_k - \overline{\mathbf{u} \cdot \nabla(\rho Y_{\alpha})} V_{ii} \right]
\end{aligned} \tag{12}$$

where the over line terms refer to flux limited interpolation of corresponding scalars, terms defined with subscript ii refer to cell defined quantities, and the vertical velocity w_{ii} has been interpolated to the cut-cell centroid. All terms with the exception of radiative heat flux have been implemented and tested in the cut-cell region. The use of this divergence expression within the projection scheme for time advancement of the momentum equations, including the IBM velocity reconstruction on faces near the boundary is described next. See Fig. 4a. Having advanced scalars on the domain (i.e. both explicit and implicit regions) consider an explicit Euler projection step from time level t_n to $t_n + \Delta t$ for simplicity:

1. Immersed boundary interpolation to get end of step target velocities in cut-face centroids, from approximate (non projected) end of step velocities on external fluid locations and object boundary conditions. For each i directed velocity component, $u_i^{ibm} = c_0 u_i^B + c_1 u_i^{int}$.
2. Flux average target velocities to underlying Cartesian faces: $\bar{u}_i^{ibm} = \frac{1}{A_{cart}} \sum_k (u_i^{ibm} A_{cf})_k$. Here, A_{cart} and $A_{cf,k}$ are the area of the Cartesian face, and k cut-face laying on such face.
3. Compute direct forcing at Cartesian level, as usual: $\bar{F}_i^n = - \left(\frac{\bar{u}_i^{ibm} - \bar{u}_i^n}{\Delta t} + \frac{\delta H^n}{\delta x_i} \right)$.
4. Compute thermodynamic divergence $(\nabla \cdot \mathbf{u})_{ii}^{th}$ from expression (12) on cut-cells.

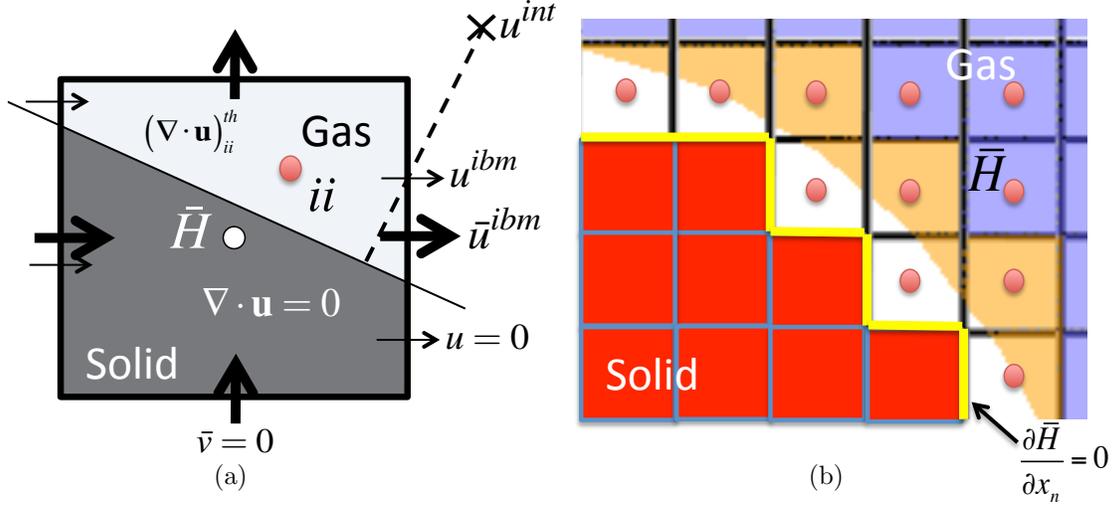


Figure 4: (a) Collocation of face and cell-centered variables for cut-cell and underlying Cartesian cell. Divergence integral equivalence. (b) Unstructured Cartesian solution for Poisson equation and Bernoulli integral \bar{H} around immersed solid object.

5. Use the divergence integral equivalence among Cartesian cell and cut-cells within it

$$\int_{\Omega_{cart}} (\nabla \cdot \bar{\mathbf{u}})^{th} d\Omega = \sum_{ii} \int_{\Omega_{ii}} (\nabla \cdot \mathbf{u})_{ii}^{th} d\Omega \quad (13)$$

to compute the Cartesian level target divergence $(\nabla \cdot \bar{\mathbf{u}})^{th}$. In Fig. 4a we have assumed the solid is rigid (incompressible), although this is not a requirement.

6. Solve Cartesian level Poisson equation for the Bernoulli integral \bar{H}

$$\nabla^2 \bar{H} = - \left(\nabla \cdot \bar{\mathbf{F}}^n + \frac{(\nabla \cdot \bar{\mathbf{u}})^{th} - (\nabla \cdot \bar{\mathbf{u}})^n}{\Delta t} \right) \quad (14)$$

7. Finally project Cartesian velocities into the target divergence field: $\bar{u}_i^{n+1} = \bar{u}_i^n - \Delta t \left(F_i^n + \frac{\delta \bar{H}}{\delta x_i} \right)$, and reconstruct cut-face velocities.

Generally, in direct forcing immersed boundary methods the Poisson equation (14) is solved along the whole Cartesian domain, including solid cells that are entirely inside immersed objects. Therefore, the boundary condition for \bar{H} that arises from the projection scheme is not being enforced exactly, and normal velocities and species penetration into the solid is expected. This effect is common for IBM, although undesirable for conservation and combustion. We have coupled the projection step with a Poisson solution that is done only on gas phase and underlying Cartesian cells, using the required homogeneous Neuman condition on internal faces separating solid cells from cut faces. See Fig. 4b. This effectively results in zero mass penetration into the solid, albeit at the cost of solving an unstructured Poisson problem on all gas cells. The reader is directed to Susanne Kilians talk on FEMTC2016.

5 Example of Integration to FDS simulation workflow: Propane fire on realistic train car model.

In order to showcase the application of the complex geometry capability on a realistic test case, consider the train car model of Fig. 5a. We intend to simulate the fire development within the compartment for a propane burner located at the back of the car, where a specified heat release rate is 800 kW.

Initially, the model is defined on CAD software as in Fig. 5a, as a disjoint set of volumes with surrounding closed surfaces. If a different geometrical entity (i.e. &GEOM) within FDS is required, they should be defined separately or in different CAD layers. On the figure, Rhino3D[®] has been used to develop the 3D model from provided blueprints. On the present complex geometry implementation, at least one cartesian cell should be completely immersed across the geometry thickness, so care has to be taken in defining the wall thickness of our model. This is indeed of little importance for the actual fire physics inside the car, but required by the cut-cell definition algorithm and overall computational scheme. Once the 3D model has been defined, it can be exported on a file format to be read by the meshing software of choice (in our case the COMSOL[®] meshing package). See Fig. 5b. The volume mesh for the geometry is defined as a tessellation of tetrahedra, being the surface mesh defined with oriented triangles with a normal out of the body defined by the classical FEM right hand rule. The size of triangles is, as a rule of thumb, expected to be consistent with the local geometry curvatures and not smaller than the expected fluid grid sizes. The last guideline is defined in the interest of not incurring unnecessary expense that will not lead to better resolution on the gas phase physics. The resulting mesh (and eventually material type and immersed boundary conditions) is written in neutral text format containing nodes, surface triangles and tetrahedra. This mesh file will, in the future, be converted into FDS input format by scripts provided by the FDS development team, or within the Thunderhead software Pyrosim[®].

Once the &GEOM object has been introduced in the FDS input file, the rest of the simulation setup is defined as usual. See Fig. 6a. In this particular showcase the burner has been defined as an FDS &OBST of 0.25 m² area, shown in the back of the car on Figs. 6b,c. The simulation is carried out for 50 s, disabling stratification and radiation, and using the Cartesian unstructured Poisson solver option plus explicit time integration in the cut-cell region. As noted, work still remains to be done in adapting the radiation unit to the complex geometry scheme. In the named figures flow visualizations of smoke and HRRPUA, as well as temperature slices obtained in Smokeview are shown. Although this is just a demonstration with no validation data available, the maximum temperatures seen at the roof of the car are of the order of 1500 °C. Although these are high, they are in line with what is expected for calculations where the radiative heat losses have been suppressed.

6 Summary

A brief description of the work being done on enhancing complex geometry simulation capability within the Fire Dynamics Simulator has been provided. The conservative cut-cell scheme employed together with a finite volume discretization and explicit-implicit or fully explicit time integration for chemical species has been detailed. Also, the work done on solving the energy and momentum equations within the overall scheme have been discussed. Velocity reconstruction in the vicinity of obstacles is done using a typical direct forcing immersed boundary method. An example showcasing the workflow for integrating this simulation capability within FDS has been provided. Our work continues in verifying and validating the proposed scheme for different fire scenarios, and developing the complex geometry routines in different areas, in particular the radiation solver, Lagrangian particles, moving boundaries, and data transfer to couple FDS with commercial thermo mechanical solvers. The result of this work is planned to be an *add on* feature of FDS, providing the flexibility of combining complex geometry (&GEOM) objects with the block objects (&OBST) already present in FDS.

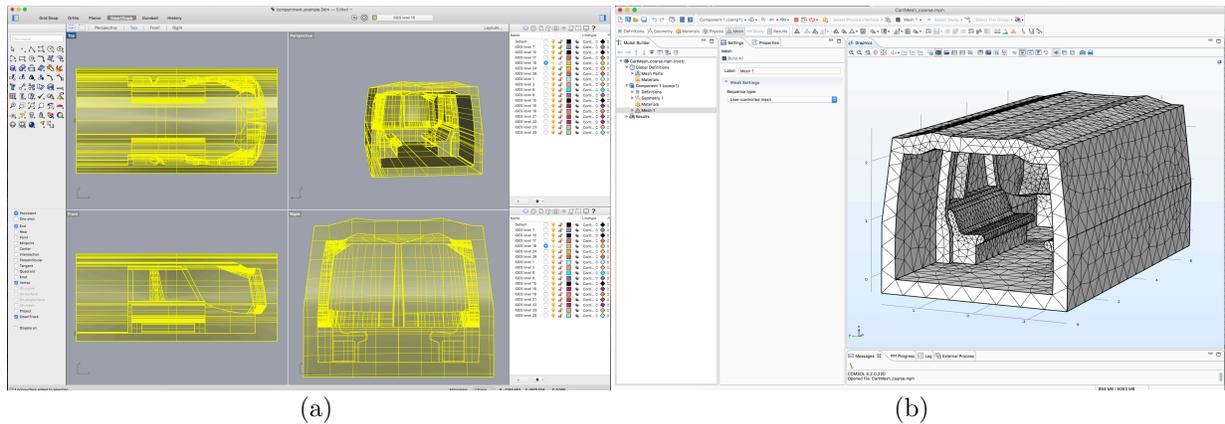


Figure 5: Realistic train car model: (a) 3D Cad model defined as a set of volumes in CAD software Rhino3D[®]. (b) Car volumes and surfaces meshes obtained with COMSOL[®] mesh engine. Tetrahedra are used as volume elements, triangles as surface elements.

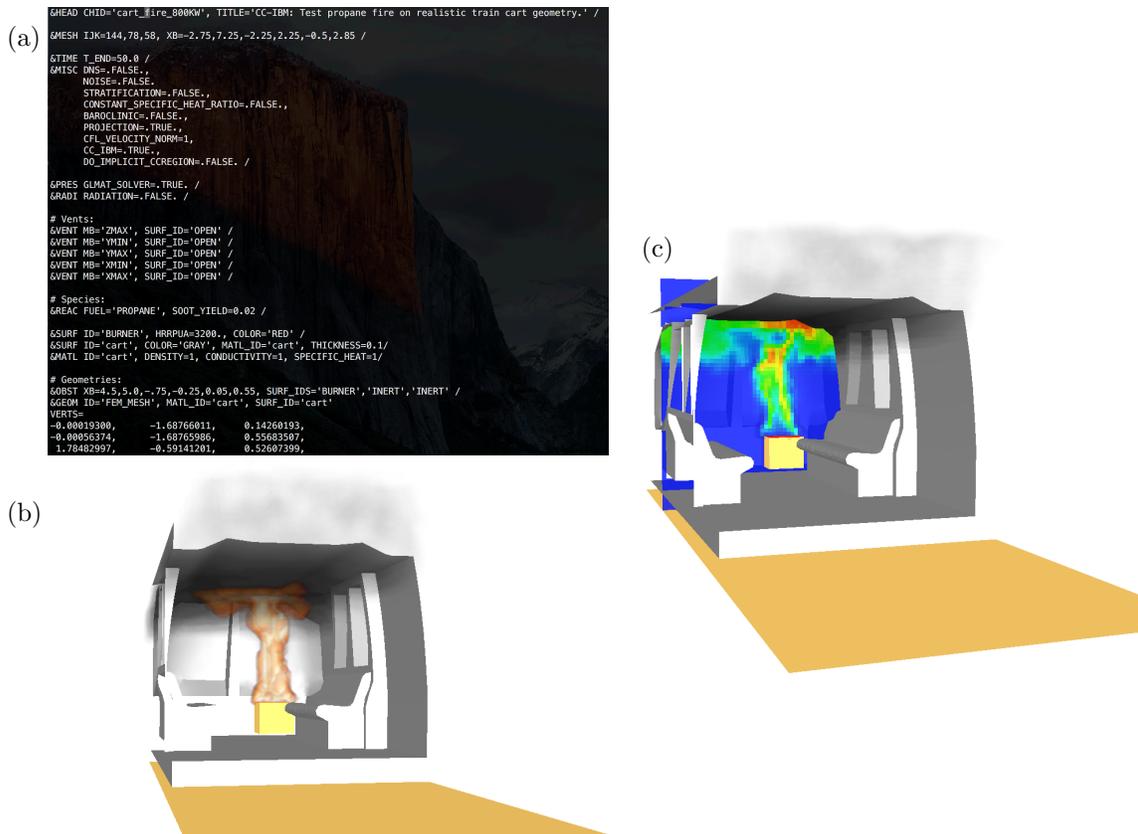


Figure 6: FDS simulation of 800 kW propane fire in realistic train car model: (a) FDS input file containing meshed &GEOM train car. (b) Smokeview visualization of smoke and HRRPUV of the problem at $t = 23.5$ s. (c) Smokeview visualization of smoke and temperature slice across the burner at $t = 45.5$ s. Temperature colormap from 0 °C to 1500 °C.

Acknowledgements

The authors would like to thank Fabian Braennstroem for providing the example train car CAD model, as well as helping with the model mesh generation.

References

- [1] *Intel math kernel library. developer reference for fortran*, mkl2017 ed., Intel Corporation, 2017, Santa Clara, USA.
- [2] E. Balaras, *Modeling complex boundaries using an external force field on fixed cartesian grids in large-eddy simulations*, *Comput. Fluids* **33** (2004), 375–404.
- [3] A. Chorin, *Numerical solution of the Navier-Stokes equations*, *Math. Comp.* **22** (1968), 745–762.
- [4] Clint N Dawson and Todd F Dupont, *Explicit/implicit, conservative domain decomposition procedures for parabolic problems based on block-centered finite differences*, *SIAM Journal on Numerical Analysis* **31** (1994), no. 4, 1045–1061.
- [5] E. A. Fadlun, Roberto Verzicco, Paolo Orlandi, and J. Mohd-Yusof, *Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations*, *J. Comput. Phys.* **161** (2000), 35–60.
- [6] G.P. Forney, *Smokeview, A Tool for Visualizing Fire Dynamics Simulation Data, Volume II: Technical Reference Guide*, Gaithersburg, Maryland, sixth ed., May 2013.
- [7] Charles Hirsch, *Chapter 5 - finite volume method and conservative discretization with an introduction to finite element method*, *Numerical Computation of Internal and External Flows (Second Edition)* (Charles Hirsch, ed.), Butterworth-Heinemann, Oxford, second edition ed., 2007, pp. 203 – 248.
- [8] Sandra May and Marsha Berger, *A mixed explicit implicit time stepping scheme for cartesian embedded boundary meshes*, *Finite Volumes for complex Applications VII - Methods and Theoretical Aspects* (Jürgen Fuhrmann, Mario Ohlberger, and Christian Rohde, eds.), Springer, June 2014, pp. 393–400.
- [9] Randall J. McDermott, *A velocity divergence constraint for large-eddy simulation of low-mach flows*, *J. Comput. Phys.* **274** (2014), 413–431.
- [10] K. McGrattan, S. Hostikka, R. McDermott, J. Floyd, C. Weinschenk, and K. Overholt, *Fire Dynamics Simulator, Technical Reference Guide*, National Institute of Standards and Technology, Gaithersburg, Maryland, USA, and VTT Technical Research Centre of Finland, Espoo, Finland, sixth ed., September 2013, Vol. 1: Mathematical Model; Vol. 2: Verification Guide; Vol. 3: Validation Guide; Vol. 4: Configuration Management Plan.
- [11] J. Van Kan, *A second order accurate pressure-correction scheme for viscous incompressible flow*, *SIAM J. Sci. Stat. Comput.* **7** (1986), 870–891.