



IFP-C3D: an Unstructured Parallel Solver for Reactive Compressible Gas Flow with Spray

J. Bohbot, N. Gillet, A. Benkenida

► To cite this version:

J. Bohbot, N. Gillet, A. Benkenida. IFP-C3D: an Unstructured Parallel Solver for Reactive Compressible Gas Flow with Spray. Oil & Gas Science and Technology - Revue d'IFP Energies nouvelles, 2009, 64 (3), pp.309-335. 10.2516/ogst/2009016 . hal-02001540

HAL Id: hal-02001540

<https://ifp.hal.science/hal-02001540>

Submitted on 31 Jan 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

IFP-C3D: an Unstructured Parallel Solver for Reactive Compressible Gas Flow with Spray

J. Bohbot, N. Gillet and A. Benkenida

Institut français du pétrole, IFP, 1-4 avenue de Bois-Préau, 92852 Reuil-Malmaison Cedex - France
e-mail: julien.bohbot@ifp.fr - nicolas.gillet@ifp.fr - adlene.benkenida@ifp.fr

Résumé — Un code parallèle non structuré pour la résolution des équations compressibles réactives avec spray — Le code IFP-C3D dédié à la simulation de chambre de combustion de moteur à combustion interne est présenté dans cet article. IFP-C3D est un code parallèle utilisant un formalisme non structuré et des grilles hexaédriques. Il résout les équations de Navier-Stokes compressibles formulées dans le formalisme ALE et en utilisant la méthode d'intégration spatiale des volumes finis sur grille décalée. L'architecture du code et les équations implantées sont résolument multi-physiques afin de prendre en compte l'ensemble des phénomènes physiques présents dans les moteurs. La présence de parties mobiles telles que les soupapes d'admission et d'échappement et le mouvement des pistons nécessitent des algorithmes originaux, tels que l'interpolation (remapping) entre différents maillages afin de changer régulièrement de maillage en cours de simulation, et des méthodes sophistiquées de mouvement de maillage basées sur une interpolation temporelle conditionnée par un maillage cible. L'injection de carburant liquide est modélisée par une approche stochastique Lagrangienne ainsi que la formation de film liquide. Les modèles physiques originaux développés par l'IFP pour modéliser la combustion essence et Diesel tels que les modèles ECFM et ECFM3Z, l'allumage par auto-inflammation (modèle TKI) et par bougies d'allumage (modèle AKTIM) et l'injection liquide permettent la simulation de toutes les configurations moteurs comme les moteurs Diesel à combustion non conventionnelle de type homogène (HCCI) mais aussi les moteurs à hydrogène à injection directe. La simulation de configurations 3D nécessite l'utilisation de supercalculateurs afin de rendre compatibles les temps de retour des calculs avec les exigences industrielles. Le code IFP-C3D a été parallélisé en utilisant la librairie MPI permettant de distribuer le calcul sur un grand nombre de processeurs, ce qui permet de profiter de la généralisation des supercalculateurs de type Clusters composés de plus de 1000 processeurs et de réduire considérablement les temps de retour des simulations 3D.

Abstract — IFP-C3D: an Unstructured Parallel Solver for Reactive Compressible Gas Flow with Spray — IFP-C3D, a hexahedral unstructured parallel solver dedicated to multiphysics calculation, is being developed at IFP to compute the compressible combustion in internal engines. IFP-C3D uses an unstructured formalism, the finite volume method on staggered grids, time splitting, SIMPLE loop, sub-cycled advection, turbulent and Lagrangian spray and a liquid film model. Original algorithms and models such as the conditional temporal interpolation methodology for moving grids, the remapping algorithm for transferring quantities on different meshes during the computation enable IFP-C3D to deal with complex moving geometries with large volume deformation induced by all moving geometrical parts (intake/exhaust valve, piston). The Van Leer and Superbee slope limiters are used for advective fluxes and the wall law for the heat transfer model. Physical models developed at IFP for combustion (ECFM gasoline combustion model and ECFM3Z for Diesel combustion model), for ignition (TKI for

auto-ignition and AKTIM for spark plug ignition) and for spray modelling enable the simulation of a large variety of innovative engine configurations from non-conventional Diesel engines using for instance HCCI combustion mode, to direct injection hydrogen internal combustion engines. Large super-scalar machines up to 1 000 processors are being widely used and IFP-C3D has been optimized for running on these Cluster machines. IFP-C3D is parallelized using the Message Passing Interface (MPI) library to distribute calculation over a large number of processors. Moreover, IFP-C3D uses an optimized linear algebraic library to solve linear matrix systems and the METIS partitionner library to distribute the computational load equally for all meshes used during the calculation and in particular during the remap stage when new meshes are loaded. Numerical results and timing are presented to demonstrate the computational efficiency of the code.

INTRODUCTION

Understanding and developing new engine concepts requires more and more help from 3D CFD and combustion modelling. Nevertheless, grid generation and computation time remain expensive and time consuming. For years, efforts have been directed toward making these key points easier and faster, and it seems that unstructured grids running on parallel computers are efficient (Heel *et al.*, 1998; O'Rourke *et al.*, 1999; Tatschl *et al.*, 2001). Since the introduction of the original KIVA in 1985, KIVA programs have become by far the most widely used CFD (Computational Fluid Dynamics) programs for multidimensional combustion modelling. At IFP, for many years, a modified version of KIVA-II (Amsden *et al.*, 1989), called KMB (Habchi and Torres, 1992), has been developed and used. Based on structured multi-blocks formalism, it can model multi-valve engines of the last generation. Even if KMB is adapted to engine calculations, there are strong constraints like the time taken by structured grid generation or the exclusive use of very expensive vector computers. Also, bad shaped cells and corner cells in complex geometry limit the numerical accuracy and highly increase the CPU time. Since 1992, KIVA-3 (Amsden *et al.*, 1992), removed the handicap by the use of a block-structured mesh that entirely eliminated the need to create regions of unused cells. In addition, the use of indirect addressing for neighbor connectivity allowed data storage arrays to be sorted, which minimized the length of vector loops and eliminated testing on cell and vertex flags. Further, tailored boundary condition data was carried in tables that allowed KIVA-3 to sweep in shorter vectors over only those vertices or cells involved. A distributed-memory implementation release of KIVA-3 has been developed in 1999 (Aytegin *et al.*, 1999). This distributed-memory implementation of the KIVA-3 code based on one-dimensional domain decomposition was successfully developed and tested. All of the essential features of KIVA-3 excluding chemical reactions, spray dynamics, and piston movement have been parallelized. KIVA-3V (Amsden *et al.*, 1997), can model any number of vertical or canted valves in the cylinder head of an Internal Combustion (IC) engine. The valves are treated as solid objects that move through the mesh using the familiar "snapper" technique used for piston motion in KIVA-3.

Other CFD codes are also available to simulate internal combustion engine as AVL's code Fire, CD-Adapco's code StarCD capable of simulating engineering problems that involve turbulence, combustion, heat transfer, reacting flows and multiphase physics. Since 2004, an open source Computational Fluid Dynamics (CFD) toolbox OpenFOAM® (Open Field Operation and Manipulation) is available to simulate anything from complex fluid flows involving chemical reactions, turbulence and heat transfer. OpenFOAM uses finite volume to solve systems of partial differential equations ascribed on any 3D unstructured mesh of polyhedral cells. The fluid flow solvers are developed within a robust, implicit, pressure-velocity, iterative solution framework, although alternative techniques are applied to other continuum mechanics solvers.

IFP have developed over the past few years a new parallel unstructured code devoted to internal CFD with spray and combustion modelling. This code named "IFP-C3D" is entirely dedicated to the simulation of compressible reactive flows with combustion and sprays. Moving-mesh strategies are integrated with all physical models needed to simulate internal combustion engines. Original moving-mesh functionality make it easier to use the code for complex moving geometries with intake/exhaust valves and also to add or remove geometrical parts during the calculation. Extensive use of the remeshing method reduces the CPU time and increases the accuracy of the simulations. Furthermore, IFP-C3D provides an unstructured internal automatic mesher that enables a three dimensional wedge mesh to be created with periodic boundary conditions to avoid, for basic geometries, the use of an external software mesher. IFP-C3D code is fully parallelized for shared and distributed memory computer and uses optimised parallel mathematical libraries for linear algebraic system calculations. This code also used CGNS standard format for Input/Output data to facilitate the exchange of data between sites and applications and help to stabilise the archiving of aerodynamic data. The data are stored in a compact, binary format and are accessible through a complete and extendable library of functions. Moreover, the HDF5 extension of the CGNS format provides a parallel I/O system that gives it high potential for the treatment of large I/O data.

1 GOVERNING EQUATIONS

IFP-C3D solves the conservation equations of mass, mass species, momentum energy and the k - ϵ turbulence equations for chemical reactive flows with sprays. Many fictive species are also used in the code to model tracers of real species or to add tracers for physical modelling. Fictive species satisfied the conservation equation of mass and momentum but are not taken into account for conservation of energy.

If we note, ρ the density, $\dot{\rho}^{Spray}$ the density spray (including the liquid film evaporation part) source term V the volume and $\frac{D}{Dt}$ the total derivative of time, the conservation of mass is expressed as:

$$\frac{D}{Dt} \int_V \rho dV = \int_V \dot{\rho}^{Spray} dV \quad (1)$$

If we note F^{Spray} the rate of momentum gain per unit of volume due to the spray, u the velocity vector, p the pressure and σ the viscous stress tensor and k the turbulent kinetic energy, the conservation of momentum can be written as:

$$\begin{aligned} \frac{D}{Dt} \int_V \rho u dV = & \int_V -\frac{1}{a^2} \nabla p dV - A_{turb} \int_V \frac{2}{3} \nabla \rho k dV \\ & + \int_V \nabla \cdot \sigma dV + \int_V F^{Spray} dV \end{aligned} \quad (2)$$

with:

$$\sigma = \mu [\nabla u + \nabla u^T] + \lambda \nabla \cdot u \mathbf{I} \quad (3)$$

(μ and λ are the first and second coefficients of viscosity. A_{turb} is equal to 1 for turbulent flows and 0 for laminar flows.

The internal energy equation is:

$$\begin{aligned} \frac{D}{Dt} \int_V \rho E dV = & \int_V -p \nabla \cdot u dV + (1 - A_{turb}) \int_V \sigma : \nabla u dV \\ & - \int_V \nabla \cdot Q dV + \int_V \dot{Q}^{Chem} dV \\ & + \int_V \dot{Q}^{Spray} dV + A_{turb} \int_V \rho \epsilon dV \end{aligned} \quad (4)$$

where E is the specific internal energy. The heat flux vector Q is the sum of contributions due to the enthalpy diffusion and to the heat conduction.

$$Q = -K \nabla T - \rho D \sum_{i=1,N} h_i \nabla \left(\frac{\rho_i}{\rho} \right)$$

with N equal to the number of chemical species, K the thermal conductivity and D the gas mass diffusion coefficient.

If the turbulence model is used (A_{turb} is equal to 1), the two additional transport equations for the turbulent kinetic energy k and the dissipation rate ϵ have to be solved:

$$\begin{aligned} \frac{D}{Dt} \int_V \rho k dV = & \int_V -\frac{2}{3} \rho k \nabla \cdot u dV + \int_V \sigma : \nabla u dV \\ & + \int_V \nabla \cdot \left(\frac{\mu}{Pr_k} k \right) dV + \int_V \dot{W}^{Spray} dV - \int_V \rho \epsilon dV \end{aligned} \quad (5)$$

$$\begin{aligned} \frac{D}{Dt} \int_V \rho \epsilon dV = & - \int_V \left(\frac{2}{3} c_1 - c_3 \right) \rho \epsilon \nabla \cdot u dV + \int_V \nabla \cdot \left(\frac{\mu}{Pr_\epsilon} \epsilon \right) dV \\ & + \int_V \frac{\epsilon}{k} \left[c_1 \sigma : \nabla u + c_s \dot{W}^{Spray} - c_2 \rho \epsilon \right] dV \end{aligned} \quad (6)$$

where the turbulent constant Pr_ϵ , Pr_k , c_1 , c_2 , c_3 , and c_s are constants whose values have been determined from experiments or results. In addition to the k - ϵ standard turbulence model, IFP-C3D also provides the RNG k - ϵ turbulent model.

2 NUMERICAL IMPLEMENTATION

2.1 Unstructured Grid and Connectivity Aspect

The unstructured formalism reduces constraints during the grid process generation and avoids the difficult numerical processing of cells located at the corner of two blocks present in the structured multiblock formalism. With unstructured grids, cells have a better shape and local refinement of zones where the gradients are suspected to be high is less expensive. With hexahedral grids, gradient calculations are accurate and moving algorithms may maintain a good grid quality when the geometry is moving.

Because of the unstructured approach, the connectivity (node, cell and face) must be computed and registered. Nodes can have any number of neighbours, while hexahedron have 6 neighbours respectively to its 6 faces. Sorting cells and nodes indexes for a given face give the unstructured grids and mesh descriptors and connectivity has to be defined. The staggered hexahedral description with cell-centred scalar and node-centered vectors is used. The topological element "face" is widely used in the solver. This element can be linked easily to hexahedrons and nodes. It is defined by one or two hexahedrons (fluid or boundary faces) and 4 nodes, nodes 1 to 4 in Figure 1. Sorting cells and nodes indexes for a given face can give the orientation that we need for solving the transport equations. The 8 sorted border nodes, node 5 to 12 in Figure 1, are also registered and used for rising numerical accuracy in convection terms.

Because of the unstructured approach, the connectivity (node, cell and face) must be all computed and registered. Nodes can have any number of neighbours, while a hexahedron can only have 6 respectively to its 6 faces.

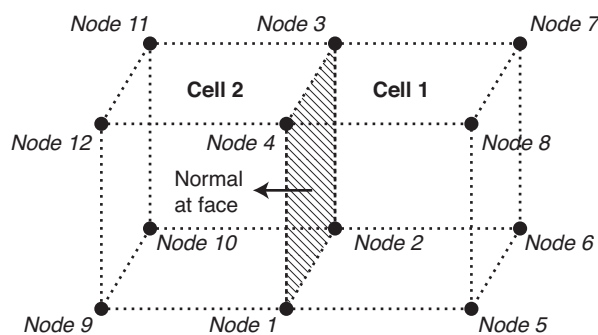


Figure 1

Unstructured grid and connectivity.

The performance of unstructured grid codes on the workstation and distributed memory parallel computers is substantially affected by the efficiency of the memory hierarchy. This efficiency depends on the order of computation and of the numbering of the grid given by the connectivity calculation. To improve the efficiency of the code IFP-C3D, renumbering methods are used to reduce the bandwidth and envelope size of the connectivity matrix. There are various algorithms that renumber unstructured grids. The main idea behind renumbering for improving cache performance is to place neighbouring members (cell centers, faces) of a set close together in memory. The bandwidth reduction family of renumbering methods that place non-zero elements of a sparse matrix close to the main diagonal can be used for renumbering unstructured grids when they are applied on the

connectivity matrix of cells and faces. Several renumbering algorithms have been tested, among them the “Reverse Cuthill McKee” (Cuthill E.H. *et al.*, 1969), Sloan (Sloan, 1986), Gibbs-King and Gibbs-Poole-Stockmeyer (Lewis, 1982). Table 1 shows results on an engine configuration grid which contains approximately 60 000 elements. The Gibbs-King algorithm gives the best reducing factor of the bandwidth and envelope size.

Figure 2 shows the original and renumbered connectivity matrices using the Gibbs King algorithm of an Port Fuel Injection engine test case. CPU time results are reported in Table 1. When the grid is ordered in a random way (automatic unstructured grid generator), the renumbering method makes the code faster by about 40%. Of course, the CPU reduction rate depends on the memory architecture of the computer.

2.2 Numerical Scheme and Code Architecture

The conservation equations previously presented have to be solved using the Arbitrary Lagrangian Eulerian formalism to take into account the effect of the moving geometric parts and of the large volume variation. IFP-C3D uses the time-splitting method to split the physical time-step into three stages. The time-splitting begins with the source terms (stage A), then follows a full implicit Lagrangian stage (stage B), and finally a sub-cycled explicit Eulerian phase (stage C). SI units are used in IFP-C3D.

In Stage A, source terms of the chemical reactions on gas (auto-ignition, combustion, post-oxidation, chemical equilibrium, etc.) of the Lagrangian fuel injection (spray, liquid film) and of the spark ignition (AKTIM model) are calculated and added to the conservation equations.

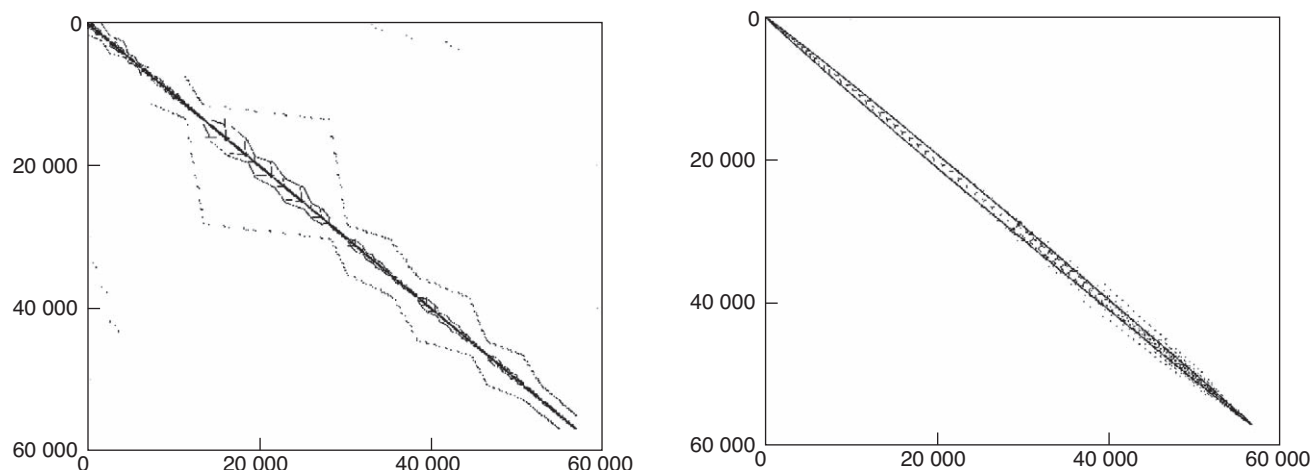


Figure 2

Original and renumbered connectivity matrix using the Gibbs King algorithm.

TABLE 1
Renumbering method comparison

	Original	RCM	SLOAN	Gibbs King	GPS
Bandwidth	39 582	5 083	2 946	2 368	2 556
Envelope size	120.4 E6	81.03 E6	46.05 E6	41.35 E6	41.76 E6

In the second stage, Stage B (Lagrangian stage), the original Semi-Implicit method introduced by Patankar (1980) is retained in its fully implicit version. The coupled implicit equations (momentum, temperature and pressure) are solved with the SIMPLE algorithm. The SIMPLE method consists of an iteration on the coupled equations until the maximum number of iterations is reached or if the convergence criterion on the initial pressure is reached. The pressure equation is solved using the BiCGSTAB iterative method with ILU preconditioning. The temperature and velocity equations are solved using the residual conjugate gradient with Jacobi preconditioning. When the SIMPLE algorithm has converged, the diffusion terms of the turbulent equation are solved. The use of the ILU preconditioning and of the BiCGSTAB method for the pressure solver gives a better rate of convergence of the SIMPLE algorithm to avoid small pressure oscillations. The use of the conjugate residual method to solve the pressure equation was not able to reach the severe convergence criterion for the pressure without excessive consumption of time. If we note r_k the residual and b the second member, the criterion of the convergence in IFP-C3D is set to $\|r_k\|^2 < \varepsilon_p \cdot \|b\|^2$ with ε_p a constant. As plotted in Figure 3, if the convergence criterion is set to ε_p equal to 1×10^{-20} , the CPU time needed to solve the pressure equation becomes excessively high with the Conjugate Residual method whereas the ILU/BiCGSTAB spent the same time for all the convergence criteria. These timings were obtained for a Port Fuel Injection engine calculation. Moreover, the ILU(0) preconditioning and the BiCGSTAB iterative method associated to the high convergence criterion increased the robustness of the Stage B algorithm for all diffusion solvers (temperature, velocity).

In Stage C, the vertices are moved from the Lagrangian trajectory calculated during Stage B to their final location calculated by the moving grid algorithm for the next time step. This translation needs the finite volume fluxes to be calculated. The fluxes of the scalar like mass, internal energy and turbulence quantities are computed between cells using a Quasi Second Order Upwind Scheme. Using connectivity of cells/face, slopes are calculated with the Superbee limiter. The faces of the control volume surrounding the vertex are also moving and the momentum fluxing calculates fluxes of momentum through each facet of the vertex. The slopes for the momentum fluxes are computed with the Van Leer limiter. The computed slopes allow the calculation of scalar

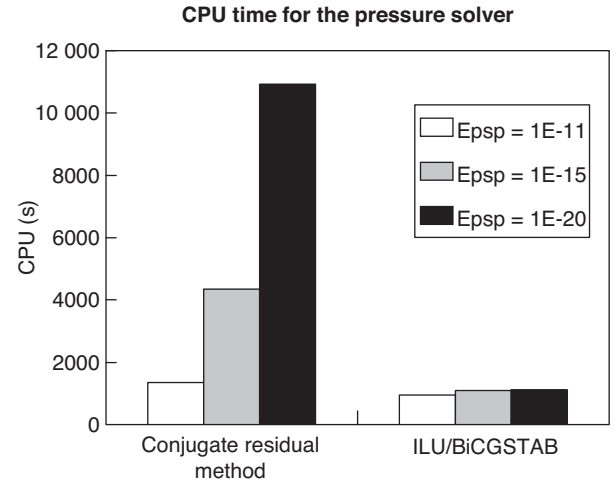


Figure 3

CPU time comparison between CGR vs. ILU/BiCGSTAB.

values at the center of faces and vector values at the middle of the edge. These slope calculations ensure a second-order upwind advection for every scalar and momentum.

To assess the validity of our development, the inviscid Gresho case (Gresho, 1990) can be solved with IFP-C3D. A stationary vortex is set in a 3D cubic box of length 0.02 m. The velocity profile along the radius is triangular as shown in Figure 4. During 3 ms of physical time, with a time step of 0.025 s, i.e. a CFL of 0.5, a decay of kinetic energy occurs. Thanks to the slope limiters, *Superbee* and *Van Leer*, 77% of this energy remains in the box with IFP-C3D, which is close to 74% for CHAD (O'Rourke and Sahota, 1998).

2.3 Grid Motion Aspect

New algorithms to move grids without any flow have been investigated. As found in the literature, the spring analogy method (Sinha *et al.*, 1998) connects all nodes with springs and relaxes the whole system after any deformation. Nevertheless it proves itself to be too slow for following the piston or the valves motion and also it leads to cells with a negative determinant. Also found in the literature, the corrected Laplacian relaxation (Torres and Henriot, 1994) performs the well-known Laplacian for the position of each node weighted by a curvature term. This method allows

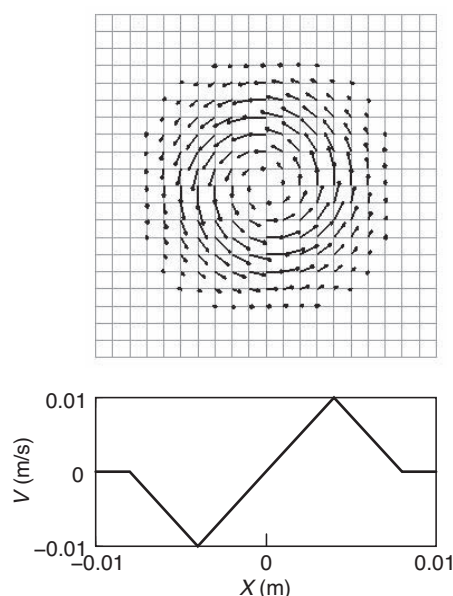


Figure 4

Initial velocity field for the Gresho case.

sharp corners in the grid. It is very attractive but also expensive in term of computing time because of the relaxation convergence. Another method, which was developed at IFP, uses an average of the displacement of every neighbour for a given node. By iterating, new positions can be computed. It gives good results for simple geometry, keeping the whole grid aspect ratio almost constant during the motion. The temporal interpolation methodology (Duclos *et al.*, 1998) is widely used when performing simulation with IFP-C3D. In this method, a target grid at a final angle is fully computed outside the solver by external grid generators (Ansys Icemcfd HEXA...). Then, during the run, the position of a node at a given time is computed by an interpolation between its former position and its target position, weighted by its proximity from the piston and the valves. Of course, actual piston and valves laws are followed exactly during the interpolation. Because of the unstructured aspect, to estimate the weight of the piston and the valves on a given node displacement, an iterative algorithm is needed and therefore the method is slightly more CPU time-consuming. Following our experience with temporal interpolation, target grids are needed at least for each geometrical maximum constraint, such as Top Dead Center (TDC), full valve opening time, etc. which means approximately every 60 crank angle degrees (CAD) depending on the complexity of the engine. The first initial grid will be given at intake TDC (which means 360 CAD before combustion TDC) or at valve opening time, then a second one 60 CAD (where the valves and the piston are very close), a third one at full valve opening time (which is commonly around 100-120 CAD after the opening), a fourth

one at intake Bottom Dead Center (BDC) and, finally, a fifth grid at valve closing time. For compression and combustion only one or two grids might be enough. Finally, for compression runs, it is possible to use a vertical-scaling algorithm. In this method, the new position of a node is computed from its former one keeping the Z coordinates ratio constant between a top fixed node and a bottom piston-like moving node. These two limiting nodes can be identified for each node of the grid at the beginning of the stroke. In that part of computation, full advantage of unstructured formalism can be taken, filling the roof with fixed hexahedrons while the cylinder is filled with pile-up cells.

2.4 Remapping Methods

One of the key issues in the 3D simulation of combustion chambers is the mesh movement strategy due to moving boundaries (*e.g.* intake/exhaust valves, piston). This strategy is usually not acceptable for grid-based numerical methods because of the risk of extremely distorted meshes. Consequently, remeshing techniques are needed to ensure good mesh quality over the entire engine cycle. Generally, a remapping algorithm is used for Arbitrary Lagrangian Eulerian (ALE) simulations to ensure the coherence between the motion of the computational grid and the motion of the fluid. These remapping algorithms need fixed grid connectivity and cannot be used for global remeshing. The remapping techniques developed in IFP-C3D ensure global remeshing including a varying grid connectivity.

The remapping method is commonly used during an engine simulation with IFP-C3D and is completely integrated in the moving grid strategies. Changing the mesh during the calculation allows to geometrical parts to be added or removed. For example when intake valve is closed, the intake pipe can be removed from the calculation. Indeed, the fluid domain in this pipe does not have any boundary conditions with the fluid domain in the combustion chamber. During combustion, the intake and exhaust valves are closed and the intake and exhaust pipe are not needed to compute the combustion. Remapping has to be used also to simulate the valve closure. The valve is considered to be closed when the distance between the valve and the seat valve is equal to $3 \cdot 10^{-3}$ m and a new mesh is needed to remove cells in the seat valve region.

Because of large geometrical deformations, users need to adapt the cell distribution of the mesh to the new geometry. During combustion, several meshes can be used to add or remove cells in the combustion chamber. For example, for a Diesel engine calculation it is common to use at least three different meshes, one for the compression stage, a second one for the injection and combustion stage and the last one for the expansion stage. All these meshes are automatically generated by IFP-C3D's internal mesher to the corresponding remapping crank angle.

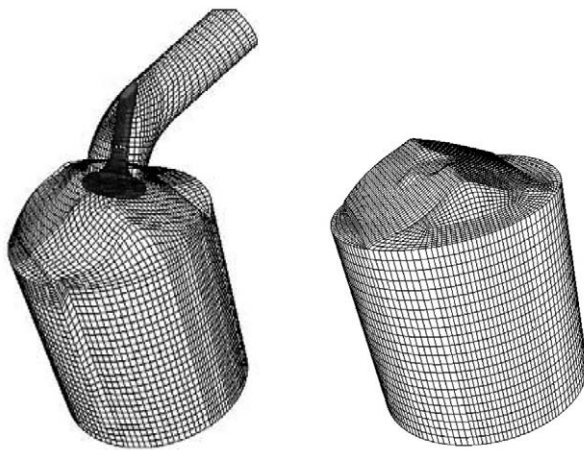


Figure 5

Mesh used for intake stage calculation and for the combustion stage calculation. The intake pipe and valve have been removed.

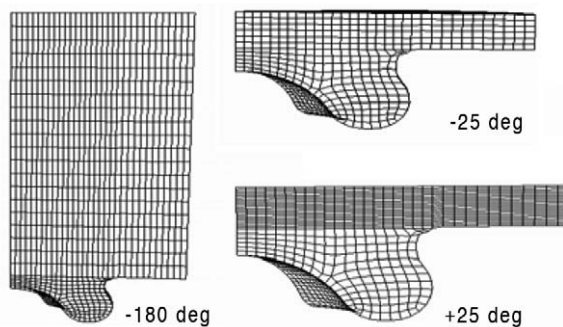


Figure 6

Different meshes used for a Diesel engine simulation. Meshes correspond respectively to a crank angle (-180 deg., -25 deg., +25 deg.).

The remapping method has to be conservative and may not affect the numerical results. Naturally, changing the node distribution during a calculation has a dissipative effect on the numerical results and it is necessary to use the remapping method carefully. To ensure numerical results accuracy, it is better to avoid the remapping during fuel injection and combustion ignition stages.

A non-conservative remapping method based on a gradient interpolation procedure is used in IFP-C3D. This method allows the interpolation of variables from one grid to another grid (donor to target grid) with a basic interpolation scheme. A post processing procedure called “repair paradigm” is conjointly used to maintain global conservation. The repair paradigm developed by Shashkov and Wendroff (2004) is applied to ensure conservation of the conservative quantities (internal energy, kinetic energy, species mass, nodal momentum) which is essential for pollutant predictions. As

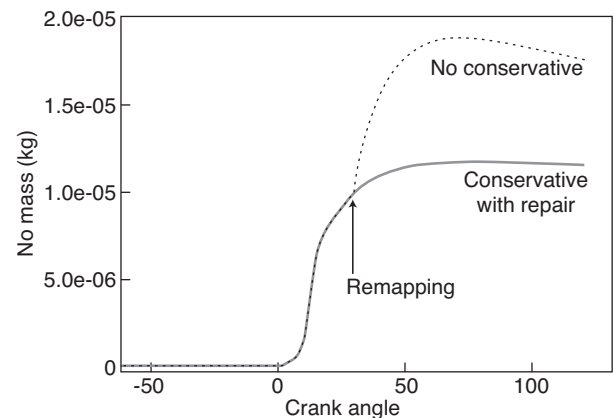


Figure 7

Impact of remapping on NOx mass prediction.

shown by Shashkov and Wendroff (2004), the repair procedure allows to ensure global mass conservation and significantly reduces the remapping errors. The impact of different mesh remapping techniques on staggered grids in the context of combustion simulations, in particular for direct injection Diesel engines, has been studied by Bohbot and Gillet (2006) and demonstrate the accuracy of this remapping method. Figure 7 shows the error that occurs in NOx prediction for a Diesel engine simulation when remapping is not conservative. The repair algorithm used in IFP-C3D maintains the accuracy of the pollutant model and avoids numerical discontinuity. The remapping can be used to add cells in the combustion chamber during the expansion phase. For instance, Figure 8 shows the iso-line of CO₂ in a Gasoline Direct Injected engine at 30 deg. after the TDC and demonstrates that the chemical mixture composition is not altered by the conservative remapping.

3 WALL AND BOUNDARY CONDITIONS

Because the grid resolution is generally low near the walls, a turbulent wall law and a heat transfer law should be used. Kays and Crawford law (Angelberger *et al.*, 1997) is available in IFP-C3D. Kinetic energy dissipation and heat transfer are directly integrated in the turbulence and internal energy equations. Velocities are corrected and projected on the wall under slip conditions. These projections used in phase B and C are made knowing each orthogonal vector to each face. Three kinds of nodes on the wall are defined. For free nodes no projection is made. For nodes near a geometrical acute corner, a projection should be made following the corner direction. For more general nodes, a projection should be made on a tangential plane at the wall. In order to solve intake and exhaust strokes for engines, open boundaries were implemented in IFP-C3D. Specific flags are set for the inlet

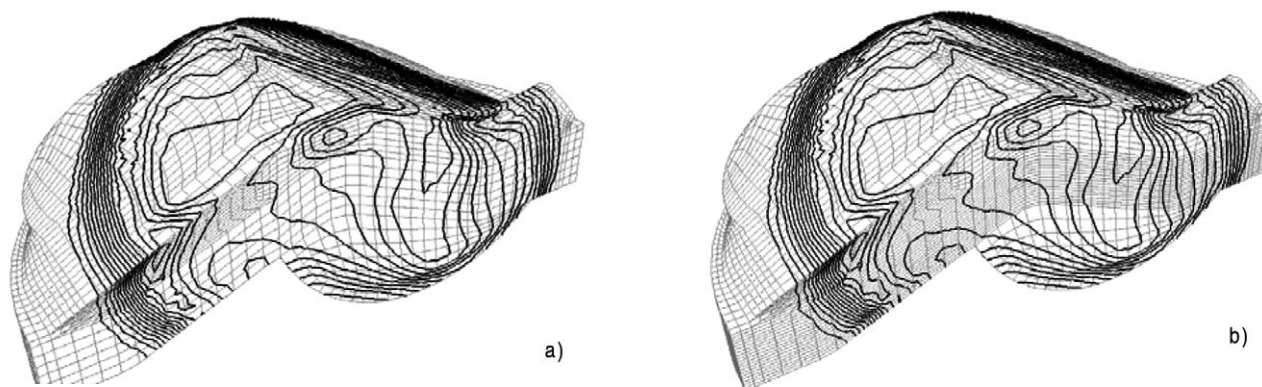


Figure 8

Conservative remapping on a GDI engine. Isoline of CO_2 . Cells are added on the z -direction in the combustion chamber.

and outlet nodes that allow building inlet and outlet face arrays to be built. At inlet, the time dependent mass flow rate, which is often known from 1D analysis or deduced from measured intake pressure, is set.

From the entrance area and the mass density downstream, a velocity is deduced. This velocity is set at the input nodes. Time dependant scalar quantities, such as species or energy (internal and turbulent) are fixed upstream of the input faces. Input fluxed quantities are then computed. At the outlet, a fixed pressure is set at a given capacity distance. Then all quantities, like velocity at output nodes and scalar fluxes through output faces, can be computed. Pressure waves can also go out when the capacity distance is non-zero. Finally, it is assumed that no diffusion occurs at inlet and outlet.

4 FUEL INJECTION MODELLING

4.1 Lagrangian Spray Description

A Lagrangian spray model has been developed in IFP-C3D. One of the main tasks is to locate efficiently all particles in the Eulerian grid. This is done by using the relative position of a particle to faces and by using the face connectivity. The spray source terms, such as injection and evaporation, are implemented in a fully multi-nozzle, multi-rate and multi-component way. Spray/wall interaction processes are described using a submodel developed by Naber and Reitz (1988). This model uses an analogy with the oblique impact on a wall of liquid jets. Following impact, the droplet trajectories are specified to be tangent to the wall surface. To take into account spray atomization and break-up for calculation of high pressure Diesel injection, the Wave model of Reitz (1987) has been implemented in conjunction with the FIPA break-up model. Figure 9 shows a comparison between IFP-C3D results and experimental data of an injection in a high pressure bomb with evaporation. As shown in Figure 9,

liquid and vapor penetration in the bomb are well reproduced. The asymptotic behaviour of the liquid penetration is well reproduced too.

4.2 Lagrangian Liquid Film Modelling (Impact, Separation)

The spray impact at the wall can lead to the formation of a liquid film. To model this phenomenon, a Lagrangian description of the film was implemented in IFP-C3D (cf. Habchi, 2006). Conjointly, an evaporation model was developed based on Direct Numerical Simulation (DNS) results of liquid film evaporation in a turbulent channel. New mass, momentum and energy wall laws were developed in order to take into account the strong gradients of gas density and viscosity near the liquid film, and the gas boundary-layer blowing due to the film evaporation. The liquid film model uses an integral method with a third order polynomial to compute the heat flux inside the liquid film. The liquid film thermal boundary layer during the transient heating stage is also taken into account. The shear stress and heat and mass fluxes at the liquid film interface are reproduced accurately using the new wall laws. A film clearing model at the valves and a separation model was implemented. All these models allow IFP-C3D to deal in particular with the impact of the fuel injection on the valves for Port Fuel Injection engines especially during cold-start operating conditions. In order to predict the fuel mixture preparation inside the cylinder of port fuel injection engines, a model for the aerodynamic stripping of the fuel film deposited on the manifold walls and a model for the fuel film separation and atomisation near the sharp edges is developed in IFP-C3D. As seen in Figure 10, a wind tunnel experiment has been simulated with IFP-C3D to predict the liquid film separation phenomena. The liquid film separation is well modelled and the separation criterion based on an analogy with Rayleigh-Taylor instabilities driven by the inertial forces of the liquid film enables the liquid film separation to be captured.

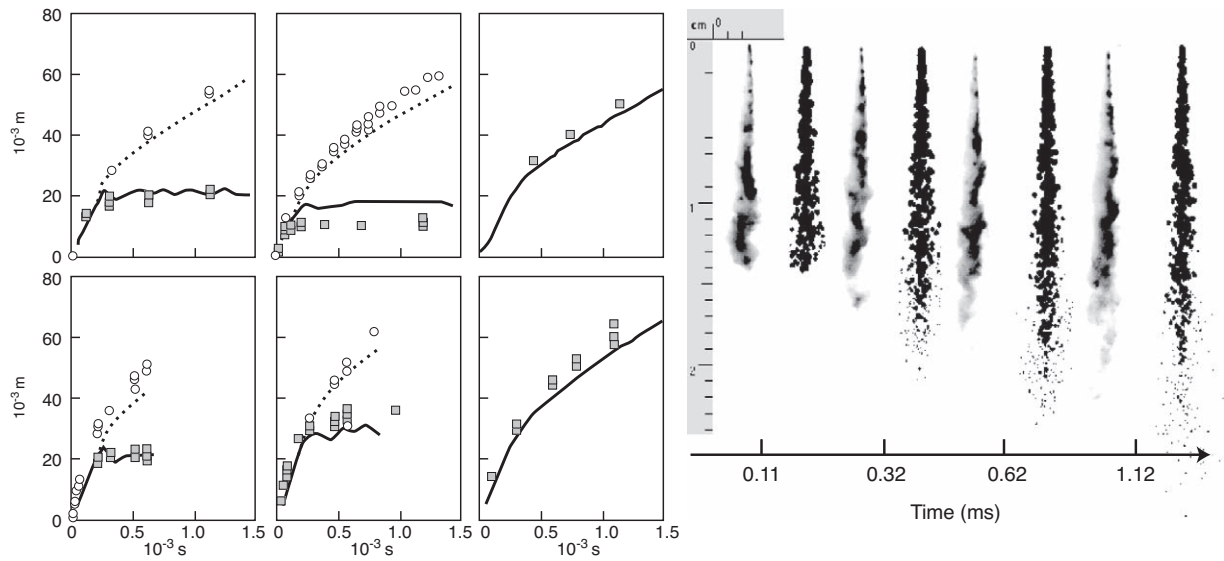


Figure 9

High pressure cell injection test case. Computation vs experiment.

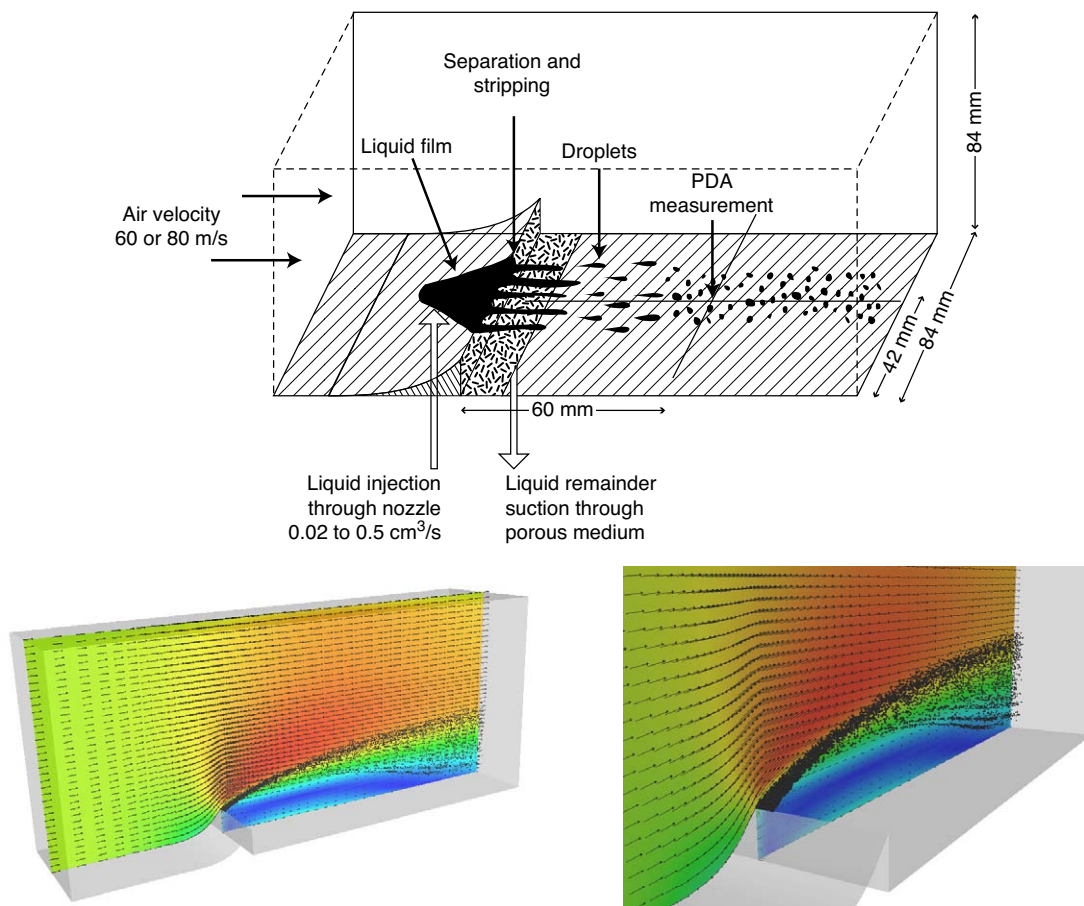


Figure 10

Computation example: air velocity 80 m/s, fuel rate 0.5 cm³/s .numerical tunnel configuration: 20 cm long, section 5 cm per 5 cm; computation of droplets distribution are made at 6 cm along the tunnel axis.

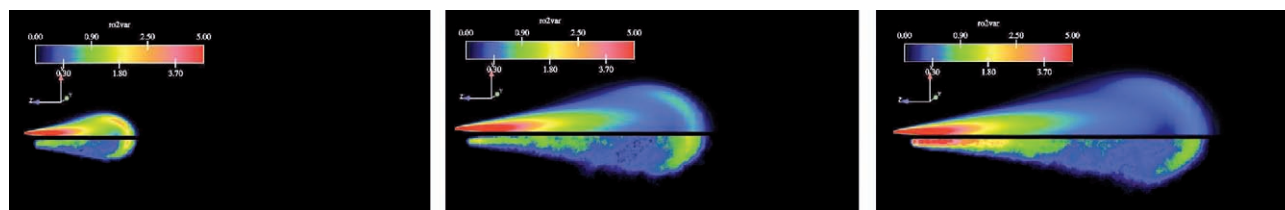


Figure 11

2D fields of fuel concentration fluctuation ($t = 0.5$ ms, 1 ms, 1.5 ms).

Liquid film prediction and spray impact is also necessary for direct injected Diesel engines, in particular to detect impact of spray on the wall of the combustion chamber, especially for no conventional spray guided engines, for example with the Narrow Angle Direct Injection concept (Walter and Gatellier, 2002) where the spray is directly injected at the center of the piston bowl. For all these reasons, injection timing and angle have to be defined to prevent liquid fuel reaching the piston walls.

4.3 Gaseous Injection

Several alternative fuels (*e.g.* hydrogen, natural gas, etc.) are currently being studied in order to decrease the dependence of the automotive industry on oil. To simulate the injection of fuel that is not liquid directly in the combustion chamber, IFP-C3D has to be able to predict the penetration and the mixing process of gas injection (Hydrogen, methane, etc.). Figure 11 shows the comparison between IFP-C3D numerical results and experimental results on the injection of methane on air. 3D field comparisons of fuel concentration show that the numerical scheme and inlet boundary conditions enables to IFP-C3D to predict the gas penetration on air (*Fig. 12*).

5 COMBUSTION MODELLING

Because of the wide variety of engine configurations, IFP-C3D has a lot of physical sub-models to compute combustion using different fuels such as gasoline, Diesel or hydrogen engines. For Gasoline engine, the ECFM model (Extended Coherent Flame Model) is used conjointly with the AKTIM (Arc and Kernel Tracking Ignition Model) model (Duclos and Colin, 2001) to compute the spark ignition. For hydrogen, an adaptation of the ECFM model has been developed (ECFM-H2 (Jay *et al.*, 2007)) it used a correlation for the laminar flame speed of hydrogen over a wide range of operating conditions and was validated against experimental data and detailed chemistry computations. The transport equation of the flame surface density is adapted to the combustion of hydrogen by introducing the laminar propagation term in the equation (usually neglected in engine applications because it is small compared to turbulent contribution).

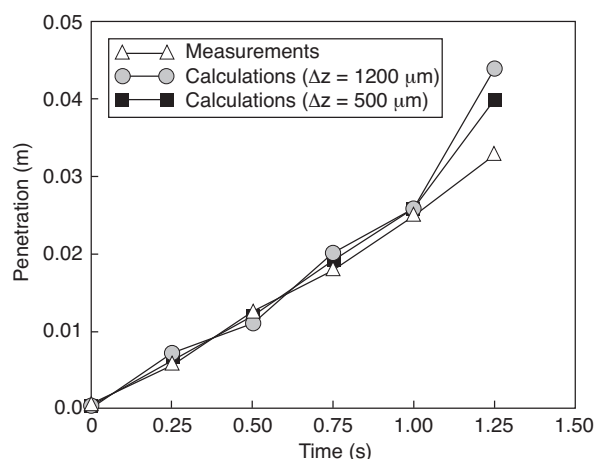


Figure 12

Penetration of methane injection on air.

For Diesel engine the combustion process is a blend of two elementary combustion modes: the chemically-controlled auto-ignition and subsequent rapid oxidation (premixed combustion) on the one side and the mixing-controlled diffusion flame on the other side. To accurately model the behaviour of any Diesel fuelled engine, the combustion modelling has to be able to simultaneously represent both these combustion modes and dynamically combines them depending on the local in-cell conditions. A unified model called ECFM3Z (Colin and Benkenida, 2004) (3-Zones Extended Coherent Flame Model) is available and can therefore be seen as a simplified CMC (Conditional Moment Closure) type model where the mixture fraction space would be discretized by only three points. The conditioning technique is extended to the three mixing zones and allows the reconstruction, like in the ECFM model, of the gas properties in the unburned and burned gases of the mixed zone. Application of the model to internal combustion engine calculations needs auto-ignition modelling that implies the necessity of auto-ignition modelling coupled for premixed and diffusion flames description. The TKI (Colin *et al.*, 2005) (Tabulated Kinetics of Ignition) model has been developed in IFP-C3D to represent the chemical kinetics related to the combustion process which relies on tabulated auto-ignition variables deduced from detailed chemistry calculations.

These chemistry calculations were performed with the Senkin code, which is part of the Chemkin package.

Indeed, in order to make the use of all species easier in combustion and chemistry, three kinds are defined: the fuel species (for multi-fuel combustion), the chemical species other than fuels (oxidants, neutral species, products of combustion, etc.) and the fictitious species. Pointers with an explicit name and tabulated libraries in International System Units are defined for all of them.

5.1 ECFM, the Extended Coherent Flame Model

The ECFM model is based on the resolution of a transport equation for Σ (flame density surface) with all the production and dissipation terms. Because of Reynolds Averaged Navier-Stokes formalism, only average state is known in a cell. ECFM is implemented in IFP-C3D with the computation of three different states in a given cell: the mean state, the fresh gas state and the burned gas state. This implementation gives a better precision for computing the flame properties, the equilibrium reactions involving CO (Amsden *et al.*, 1989), and the kinetic reactions involving NO following an extended Zeldovitch mechanism (Heywood, 1988). Building these three states at each time step for each cell is done by defining three fictitious species, one for the fuel, one for the oxygen and one for the fresh gas enthalpy. These fictitious species, along with the flame surface density, are diffused and advected.

The exact transport equation of Σ is (Trouvé and Poinso, 1994):

$$\underbrace{\frac{\partial \Sigma}{\partial t}}_I + \underbrace{\frac{\partial}{\partial x_i} (\langle u_i \rangle_s \Sigma)}_{II} = \underbrace{\left\langle (\delta_{ij} - n_i n_j) \frac{\partial u_i}{\partial x_j} \right\rangle_s}_{III} \Sigma - \underbrace{\frac{\partial}{\partial x_i} (\langle S_d n_i \rangle_s \Sigma)}_{IV} + \underbrace{\left\langle S_d \frac{\partial n_i}{\partial x_i} \right\rangle_s \Sigma}_V \quad (7)$$

where $\langle \rangle_s$ is the averaging operator along the flame front and n_i the component of the unit vector normal to this front and oriented towards fresh gases. The following transport equation (which represents a closure of the previous one) is the one used in the ECFM model to compute Σ (Colin *et al.*, 2003):

$$\begin{aligned} \underbrace{\frac{\partial \Sigma}{\partial t}}_I + \underbrace{\frac{\partial}{\partial x_i} (\tilde{u}_i \Sigma)}_{II} = & \underbrace{\frac{\partial}{\partial x_i} \left[\left(\frac{\mu}{S_c} + \frac{\mu_t}{S_{ct}} \right) \frac{\partial (\Sigma / \bar{\rho})}{\partial x_i} \right]}_{III} + \underbrace{\frac{2}{3} \frac{\partial \tilde{u}_i}{\partial x_i} \Sigma}_{IV} + \underbrace{\alpha K_t \Sigma}_V \\ & + \underbrace{\frac{2}{3} \frac{\partial \bar{\rho}''}{\partial \bar{\rho}^b} S_L \frac{1 - \bar{c}}{\bar{c}} \Sigma^2}_{VI} - \underbrace{\beta S_L \frac{\Sigma^2}{1 - \bar{c}}}_{VII} \end{aligned} \quad (8)$$

where S_L , $\bar{\rho}''$ and $\bar{\rho}^b$ represent the laminar flame speed and the mass densities of fresh and burned gases, respectively. \bar{c} is the volume fraction of burned gases. α and β are modelling constants equal to 1.6 and 1, respectively. The different terms in the transport equation of Σ represent (I) the non-stationary accumulation, (II) the convective transport by the mean velocity field, (III) the laminar and turbulent diffusions, (IV) the stretch due to the mean flow, (V) the turbulent stretch, (VI) the effect of thermal expansion and positive curvature, (VII) the destruction term due to negative curvature.

5.2 ECFM3Z, a Unified Combustion Model

The unified model ECFM3Z had been developed by Colin and Benkenida (2004) in IFP-C3D. In this model, the unburned/burned gas zones description of the ECFM model is retained, as well as the premixed flame description based on the flame surface density equation. In order to account for diffusion flame and mixing processes, each computational cell is split into three mixing zones: a pure fuel zone, a pure air plus possible residual gases (EGR) zone and a mixed zone, as schematically presented in Figure 13. This structure allows us to take into account the three main combustion modes (Auto-ignition, Premixed Flame and Diffusion Flame) encountered in most combustion devices. During injection, the evaporation of the spray droplets leads to a source of mass in the fuel zone. The fuel and air (+EGR) from the fuel zone and from the Air zone are turbulently mixed to form the mixed region. The characteristic mixing time τ_m depends on the characteristic time-scale of the turbulence. The mixture zone may then auto-ignite. Auto-ignition can be handled by several specific models which provide the auto-ignition delay that ECFM3Z integrates over time. When the auto-ignition delay is reached, combustion in the mixed region starts, thereby dividing it into two zones: a fresh (or unburned) gas zone and a burnt gas zone. As the mixed fresh gases in the mixed region are oxidized by auto-ignition, the hot products formed are added to the burnt-gas mixing region.

In real mixtures, a pure air (+EGR) and a pure fuel region cannot be found. Instead, the mixture can be described as a continuous distribution of intermediate states between pure component states. This type of structure can be modelled by a probability density function (pdf) of the mixture fraction variable. Calling this mixture fraction variable Z , the standard ECFM3Z model can be seen as a first stage approximation in which the pdf function is built only on three Dirac distributions:

$$P(Z) = a \delta(Z) + b \delta(Z - ZM) + c \delta(Z - 1)$$

The first delta function corresponds to the unmixed air region A, the second one to the mixed region M and the third one to the unmixed fuel region. In an attempt to improve the

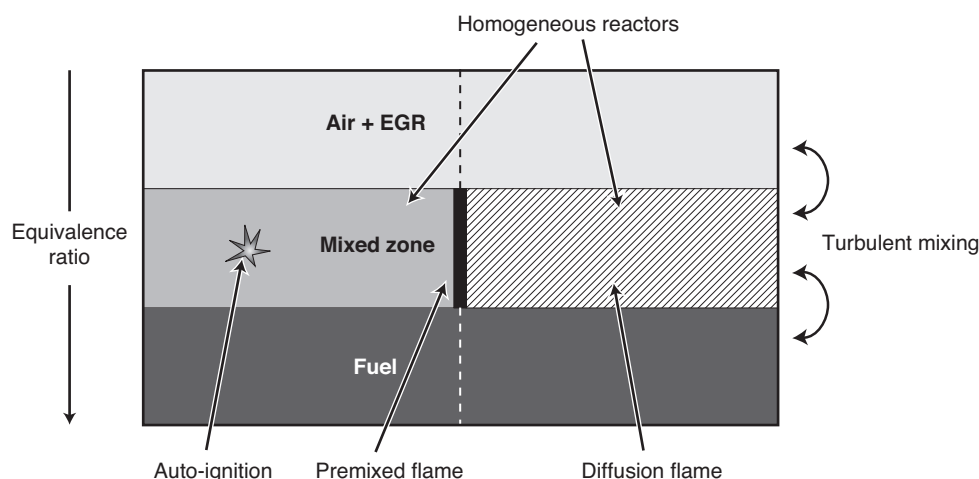


Figure 13

Sub-grid description of ECFM-3Z model.

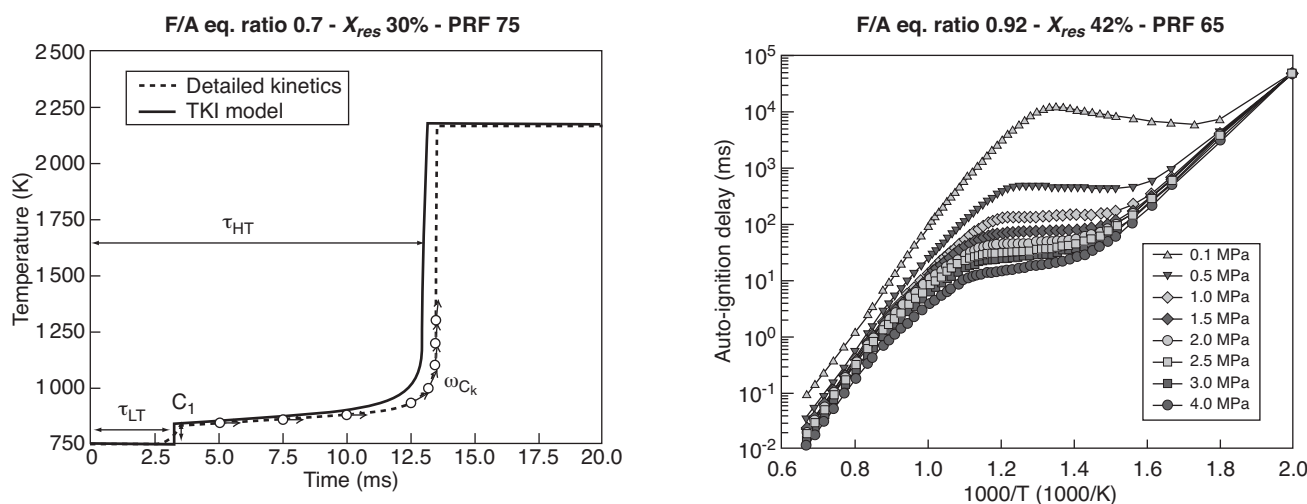


Figure 14

Comparison of temperature evolution from detailed kinetics and the TKI model for a constant volume reactor (left). Delay evolution as a function of temperature for various pressures (right).

mixture formation modelling, the ECFM3Z-pdf model (Colin, 2007) has been created as an extension of the standard ECFM3Z model relying on the use of a probability density function to describe the mixture in each sub-cell region. In the standard ECFM3Z model, the Dirac-based distribution induces the description of the mixture in the mixing region as a homogeneous reactor whose state is only controlled by the mass flows from the fuel and air regions. The addition of a probability density function introduces an actual mixture modelling which is able to account for the sub-cell inhomogeneities. This mainly leads to the existence of a continuous distribution of Fuel/Air equivalence ratio in the mixing region which, among others, may allow the combustion in the cell even if the mean Fuel/Air equivalence ratio is not in the flammability region or may produce soot particles.

5.3 Tabulated Auto-ignition Modelling

In practical applications, especially for new low emissions combustion concepts such as HCCI engines, cool flames can be encountered. They are characterized by a first rapid increase of temperature (Favre average of the progress variable \tilde{c} goes from 0 to a small fraction C_1) after a delay τ_{LT} has been reached, followed by a slow-down of reactions until the main delay τ_{HT} is reached (see Fig. 14).

In most engine combustion problems featuring a complex geometry, the direct coupling between 3D CFD and detailed chemistry leads to a huge number of transported scalars and to CPU time incompatible with industrial applications. In order to save CPU time while retaining most of the information provided by detailed chemistry, several similar methods

based on *a priori* detailed chemistry calculations and results stored in look-up tables were recently proposed. Among them, the Tabulated Kinetics of Ignition (TKI) auto-ignition model (Colin *et al.*, 2005) relies on tabulated auto-ignition parameters deduced from detailed chemistry calculations performed with the Senkin code, which is part of the Chemkin package. Detailed chemistry calculations are performed for a constant volume reactor. Since fuel in the 3D CFD code is burned in a single step, the temperature rise is proportional to the amount of fuel consumed. The assumption is then made here that a progress variable based on fuel in the 3D CFD code is equivalent to a progress variable based on temperature in the detailed chemistry code. Consequently, the target is to correctly reproduce the temperature evolution, like the one represented in Figure 14.

The auto-ignition delay curves plotted as a function of temperature feature typical S-shaped curves (*Fig. 14*) with an inflexion in the cool flame region, also called negative temperature region, because in this region the delays increase when the temperature increases. The lower and higher limits of this region vary with the thermodynamic conditions.

In Figure 14 (left), the thermodynamic conditions lead to a first part of the combustion controlled by the cool flame. To represent this phenomenon, two auto-ignition parameters are defined: the auto-ignition delay τ_{LT} and the portion C_1 of fuel consumed by the cool flame. Then, main ignition is modelled on the basis of a second delay τ_{HT} and of reaction rates tabulated as a function of the progress variable. When the thermodynamic conditions lead to the absence of the cool flame effect, $\tau_{LT} = \tau_{HT}$ and $C_1 = 0$.

As in an internal combustion engine local thermodynamic conditions evolve with the piston motion, all model parameters (and among them delays τ_{LT} and τ_{HT}) extracted from database vary. To take into account this evolution for the in-cell auto-ignition delay, an intermediate species is of which concentration Y_I has a growth rate proportional to the local fuel tracer concentration Y_{TFu} and to the relevant auto-ignition delay. The fuel tracer concentration Y_{TFu} represents the amount of fuel in region M that would exist in the computational cell without chemical reactions; this species is fictive as it is neither consumed nor created by chemical reactions. From the definition of the intermediate species, the auto-ignition delay in the computational cell is reached when the intermediate variable Y_I is equal to the amount of fuel tracer present in the fresh gases.

If the auto-ignition occurs in cool flame conditions, a portion C_1 of the fuel is consumed. The model assumes that the cool flame chemistry is fast and the characteristic time of fuel consumption τ_c is taken to be constant. After the cool flame combustion or if the auto-ignition occurs in main flame conditions, the fuel oxidation is described by the reaction rates ω_{kc} , the time derivative of the progress variable c based on the temperature profile. The reaction rate ω_{kc} is tabulated from detailed kinetics for distinct values of the progress

variable \tilde{c}_k . These values of are discrete and were chosen after several tests to correctly represent the main flame induction process after the cool flame delay has been reached. The 3D CFD code reaction progress variable is defined as the ratio of the fuel consumed $Y_{TFu} - Y_{Fu}$ over the total amount of available fuel vapour Y_{TFu} , leading to:

$$\tilde{c} = 1 - \frac{Y_{Fu}}{Y_{TFu}}$$

5.4 AKTIM: a Spark Plug Ignition Model

The Ark Kernel Tracking Ignition Model (AKTIM) (Duclos and Colin, 2001) is used for spark ignition modelling to describe the flame kernel expansion in SI engines. This model is divided into four complementary sub models. The first one describes the secondary electrical circuit of the inductive system of the spark plug with the available initial electric energy, the resistance and inductance. The geometry of the spark is modelled by a set of discrete particles that avoid the remeshing of the combustion chamber. The spark is spatially modelled by particles informally placed along the spark path. This spark is elongated in time by the mean flow field.

A simplified scheme of a classical inductive circuit systems is shown in Figure 15. The primary circuit includes the battery and the primary inductance L_p . When the switch is opened (at the prescribed time of ignition) the energy is stored in the primary inductance. Only (approximately) 60% of the primary energy is transferred to the spark plug, the energy remaining is dissipated by the secondary inductance L_s . In the AKTIM model, only the secondary electrical circuit is modelled. At the time of ignition (as an input parameter), the voltage between anode and cathode reaches the breakdown voltage and a spark is formed. The spark initiated between the two electrodes is represented by Lagrangian particles uniformly spaced between the two electrodes. These particles are convected by the mean flow field leading to the deformation of the spark. The model is thus capable of handling multiple breakdowns.

Flame kernels are represented by Lagrangian particles and are assumed to be quasi-spherical. These particles are initiated along the spark at the instant of breakdown with a burned gas mass inside the kernel equal to zero and an initial excess of energy. This energy is transferred into the fresh gases surrounding the spark. After ignition (when critical energy is reached), the evolution of each flame kernel is determined by the evolution of its excess of energy and its burned gas mass.

Mantel (1992) showed that air flow between electrodes is highly disturbed by the geometry of the spark plug. The flow velocity at the spark plug gap leads to the spark elongation and for the kernel flame convection and distortion which are

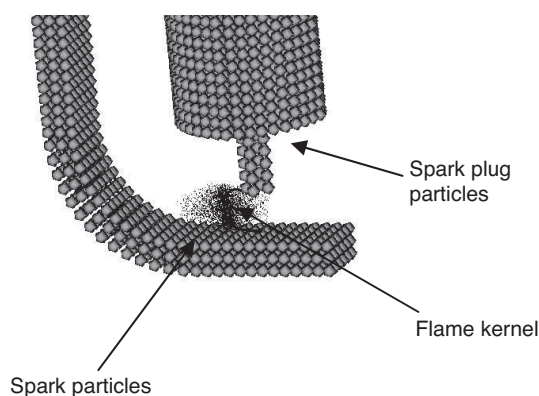
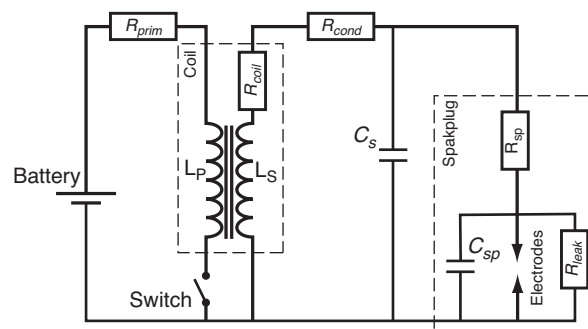


Figure 15

3D view of the spark plug, electrical spark and flame kernel particles a). Simplified electrical scheme b).



key parameter in the flame development. The spark plug is modelled with a set of fixed particles that are used to calculate each flame kernel its surface contact with the spark plug. These particles also induce drag on the mean flow field.

5.5 Pollutant Modelling

The pollutant modelling is done in the burnt gas part of the mixing region of each cell. Two set of reactions are considered: The reactions in the hot products formed on the unburned gases from the mixing region and the reactions linked with the turbulent mixing of fuel and air from the burnt gas side into the mixed burnt gases. The burnt gas part of the mixing region contains combustion products but also air and fuel which enable diffusion reactions leading to carbon monoxide and soot formation. In IFP-C3D, the pollutant modelling is obtained by coupling two formerly independent models: the first one called CORK (CO Reduced Kinetics) (Jay *et al.*, 2007) is dedicated to CO modelling and the second one called PSK (Phenomenological Soot Kinetics, Martinot *et al.*, 2001) is devoted to soot modelling.

The combustion in the burnt gas part of the mixing zone which leads to CO production and its oxidation into CO₂ is computed with a 6-step reduced kinetic model. This reduced 6-step kinetics is an improved version of the 4-step model of Hautman *et al.* (1981) obtained by the addition of 2 extra chemical reactions in order to allow the coupling of this scheme with the PSK soot model. The reduced kinetics approach is justified by the fact that in ECFM3Z, the burnt gas side of the mixed zone, where high temperature combustion occurs, is considered as a homogeneous reactor. The model is applied on each computational cell where gas composition and temperature are conditioned in the burnt gas part of the mixing region

To be coherent with the whole ECFM3Z model strategy, the soot modelling is performed in the burnt gas part of the mixing region which is considered as a homogeneous

reactor. In the original PSK model (Martinot *et al.*, 2001), the acetylene (C₂H₂) production was directly obtained through fuel decomposition. When coupled with the CORK model, the C₂H₂ is now indirectly produced based on fuel. Similarly, the original PSK model contains oxidation reactions for C₂H₂ and H₂ which are also part of the CORK model. As a consequence, the original 11-step PSK model is reduced to an 8-step phenomenological model when coupled with the CORK model. To ensure the simulation predictivity the CORK and PSK models are fully coupled to the ECFM3Z combustion model by taking into account the species in the mass and energy balances. Moreover, since the time-step of the 3D CFD engine code is often not coherent with the very small chemical characteristic time scales, a cell-by-cell sub cycling procedure based on the smallest chemical time of reactions has been developed. This ensures precision in the resolution of both sets of reactions while avoiding the need to consider extremely reduced time-steps for all cells and models

Most researches on the chemical reactions of nitrogen have been motivated by the impact of nitrogen compounds emitted from combustion sources on the environment. In the combustion of fuels, the oxidation of atmospheric nitrogen (N₂) by the thermal mechanism is the major source of NO_x emissions. Among the nitrogen oxides, nitric oxide (NO) is the main species emitted during a combustion process. Based on these observations, the most widely used model for the prediction of nitrogen oxide emissions is the extended Zeldovitch model (Zeldovitch *et al.*, 1947) which has been implanted in IFP-C3D.

6 PARALLELISM AND CPU PERFORMANCE

Parallelism is the key to performance on any system manufactured today. All domestic high performance systems in quantity production use commodity microprocessors at the core of their design. It has been repeatedly shown that such

CPUs routinely achieve only 20% of their peak performance on real scientific codes. Clearly, an application must scale to hundreds of CPUs on such systems, if it is to achieve the sustained performance to stay competitive in high-end computing and to give simulation return time compatible with industrial requirement. Because this need is clear, parallelism is being aggressively pursued on two fronts. The first is a continuation of the classic message-passing approach of clustered computing. The current leading technology for this approach is via the Message Passing Interface, commonly referred to as MPI (Dongarra *et al.*, 1993; Doss *et al.*, 1993). Nevertheless, the cost for parallelizing a numerical method using message passing paradigm is high, because parts of the code have to be completely restructured and the IO must be reorganized. Since this work cannot be done automatically, MPI parallel code development is expensive and time consuming. The second philosophy simply expands on compiler directive approach. This has been recently codified into an industry standard set of directives under the banner OpenMP (<http://openmp.org/wp/>). Each parallelization approach has its advantages and disadvantages and IFP-C3D had been parallelized with the two paradigms.

First IFP-C3D had been parallelized with the OPEN/MP compiler directives in all routines that are more CPU time-consuming. After a profiling of the sequential version, we have noted that most CPU time is spent in very few loops in solver and diffusion terms computation. Implementing OPEN-MP directives into them gives a parallelization of about 50% of the code. Parallelization was also developed for the combustion routines. After such encouraging results, more routines were implemented in order to achieve similar speed-up on the entire code. This first level of parallelism allows IFP-C3D to be used on shared memory machines.

In a second step, for large calculation and to use IFP-C3D on super-scalar machine, the MPI parallelism had been implemented. This parallelization of IFP-C3D with MPI had been a time consuming and difficult process because of the complexity of all physical sub-models included in the IFP-C3D code. The main difficulties came from the combination of Lagrangian discretisation that are used for spray and spark ignition modelling with the Eulerian formulation of the Navier-Stokes equations. Moreover, IFP-C3D numerical scheme is complex with the time-splitting technique in 3 different phases and with the Arbitrary Lagrangian Eulerian formulation used on staggered grid. Indeed, the parallelism process followed a development plan clearly defined in order to reduce problem with the software architecture. The MPI partitioning routine was first developed to manage the moving grid algorithm with remapping and the decomposition of 3D domain into sub-domains, each consisting of a number of cells. Each MPI process within the group computes one sub-domain. IFP-C3D uses the METIS generic graph partitioning library to decompose and distribute the 3D domains. This decomposition is done when IFP-C3D load a new mesh, at

the beginning of the calculation and also at each remapping stage. MPI exchange routines were developed to manage the data exchange at cells, vertex, faces centers and also wall faces centers. All these routines allow IFP-C3D to deal with staggered data and also with all Lagrangian models.

After a strict evaluation of this first MPI release of IFP-C3D, the Lagrangian phase (B) and the Eulerian Phase (C) were parallelized. For the pressure solver, the linear algebraic linear solver IFP-Solver used in IFP-C3D provides the MPI and OPEN_MP parallelism and only the development of the assembly part of the pressure matrix was necessary and was distributed in all the processors. All other solvers used during the SIMPLE loop were parallelized. To implement the parallelism in the Eulerian Phase C, the gradient and flux routine for scalar and vector values were modified to be distributed in different processors. The performance of these parallel parts was intensively tested and optimised before the development of the other model included in IFP-C3D. Numerical tests on a cube containing 5×10^5 nodes had been performed with an initial vortex located in the center of the box. To test effectively the Eulerian Phase (C), the lower wall moves and compresses the vortex. CPU time results are displayed in Table 3 (HEXA500k).

Subsequently this validation, turbulence modelling and combustion (Phase A without Lagrangian model) was implemented. The distribution of combustion models is direct because most of them use the chemical composition mixture of the cells to compute chemistry reactions and do not need any other information. The development of these models does not impact the performance of the MPI parallel version, IFP-C3D was able to compute combustion and intake/exhaust phase without liquid fuel injection. Performances of this release can be evaluated in Table 3 with a Port Fuel Injection (PFI no injection) test case. Results are very encouraging to begin the development of the Lagrangian model (Spray and Spark ignition).

The Lagrangian model (spray and spark ignition) were parallelised using a domain decomposition strategy. Each processor computes and knows the particles located in his part of the grid. This strategy was chosen because of the low CPU time cost of the Lagrangian part and also because Lagrangian particles used for liquid injection and for spark ignition vanish during the simulation and have a small life-time in comparison with the whole engine time calculation. The PFI test case was simulated with an injection located in the intake pipe. We computed the injection phase that could be critical in term of performance. Indeed all the particles are concentrated in a small part of the geometry and move with a high velocity. Moreover, particles can impact the wall and the valve and can create liquid film. The liquid film can also move from one domain to another and a routine to manage wall faces exchange was developed especially for this model. Table 5 shows the CPU time performance results and assess that the particles distribution does not deteriorate massively

the performance of the MPI release of IFP-C3D (cf. PFI-FINE-SPR vs PFI-FINE). Indeed, the speed-up and the parallel efficiency are equal with the same grid configuration with or without fuel injection.

Figure 16 shows the PFI-FINE-SPR test case and the fuel injection located in the intake pipe which impacts the valve and creates a liquid film. Figure 17 shows the domain decomposition used for 8 and 32 processors to perform these calculations and we can notice that the intake pipe is distributed in 3 processors and in 5 processors respectively.

We performed the most representative engine configurations such as Direct Injection Diesel engine, Gasoline Direct Injection engine and homogeneous combustion engine (IIE) and a Port fuel Injection 16-valves engine to test the perfor-

mance of the MPI parallel version of IFP-C3D. The description of all the test cases are listed in Table 3. The number of cells per processor used for each test case is listed in Table 4. To test the efficiency of the parallelism of all the physical model, each test case uses different physical modelling. For instance, the PFI-FINE a port fuel injection engine with only air admission. To test the efficiency of the Lagrangian spray and liquid film modelling, the PFI-FINE-SPR test case is equivalent to the PFI-FINE case with a fuel spray injected in the intake pipe. The PFI-4V test case is a Port fuel injections with four valves to test the effect of multiple air intake pipe on the parallelism efficiency. The IIE test case is the homogeneous combustion of a air/gasoline mixture and assess the effect of combustion modelling on parallelism. The GDI test

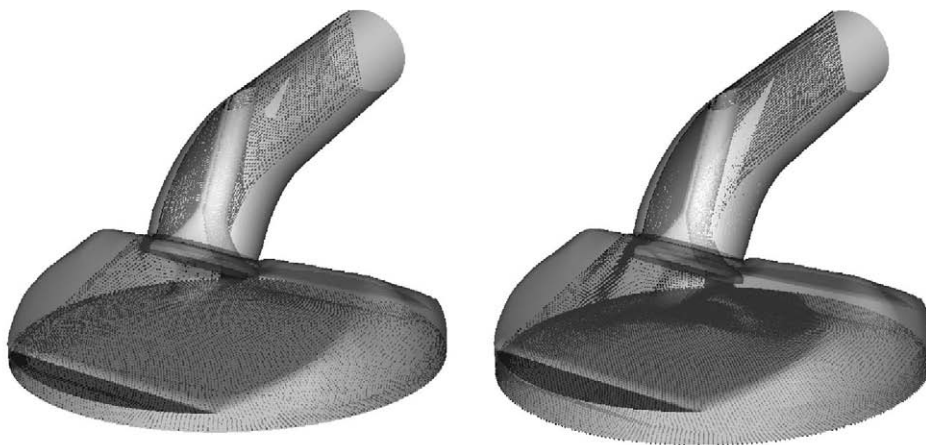


Figure 16

PFI-FINE test case with fuel injection. Spray particles (yellow particles) are located in the intake pipe and the fuel droplets propagate in the computational domain (numerical results obtained with 48 processors).

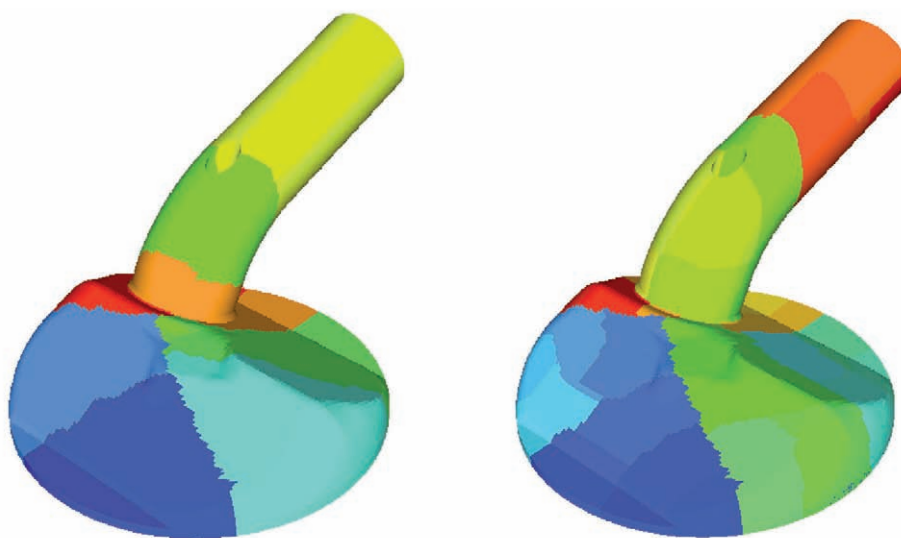


Figure 17

Domain decomposition with METIS partitioning library for 8 and 32 processors for the PFI-FINE and PFI-FINE-SPR test cases.

TABLE 2
IBM Cluster description

IBM Cluster general description	114 nodes Supermicro 1041 M
Number of processors per node	4 processors AMD Barcelona
Processors characteristics	Quadri-core 2.3 GHz, 1 Mo L2 memory
RAM per node	32 Go DDR2 memory
Local disk capacity	160 Go local disk
Interconnect network	Infiniband

case represents the simulation of a direct injection gasoline engine using the Lagrangian spark ignition model AKTIM and a Lagrangian gasoline spray. To test the parallel efficiency of the periodic boundary condition and the ECFM-3Z combustion modelling and the TKI auto-ignition modelling the DID test case is performed. However, the number of cells vary from 36 000 to 500 000 and we have test the most test case up to 64 processors for the large grids. Test cases containing more than on million nodes have been also computed to ensure the parallel implementation of IFP-C3D and to test IFP-C3D up to 128 processors. This IFP-C3D MPI release was tested on an IBM Cluster Machine described in Table 2. Table 5 shows the CPU performance results of the MPI release of IFP-C3D. The speed-up and the parallel efficiency are encouraging when the number of processors is lower than 16. The parallel efficiency is greater than 50% up to 64 processors. When distributing the calculation on 32 and 64 processors the parallel efficiency decreases but we can notice that the speed-up between 8-16 processors and 16-32 processors are constants (in the range [1.6, 1.8]). This first MPI release of IFP-C3D has not been fully optimized yet and we are confident that dynamic partitioning, memory optimisation and asynchronous communication will help us to achieve the 75% of parallel efficiency for 64 processors. Table 6 gives the memory efficiency of the MPI release of IFP-C3D and we can notice that the memory is multiplied by a factor respectively equal to 1.7, 2.2, 5, 18 when using 8, 16, 32 and 128 processors. Mesh partitioning has a dominant effect on parallel scalability for problems characterized by (almost) constant work per point. Poor load balance causes idleness at synchronization points, which are frequent in implicit methods (*e.g.*, at every conjugation step in a Krylov solver). Load balancing for a parallel system is one of the most important problems which has to be solved in order to enable the efficient use of parallel computer systems. Dynamic load balancing scheme is essential because of differences in network bandwidth, local cpu load, local memory load (may change dynamically, *e.g.* workstation owner may start a big edit or backup job leading to disk paging and loss of performance). The development of a dynamic load balancing scheme in IFP-C3D will help us to increase the

parallelism efficiency to take into account the CPU and memory load differences between domains that occurred with fuel injection and inhomogeneous combustion. For machines with physically distributed memory, MPI has been a natural and successful software model. For machines with distributed shared memory and non uniform memory access, both MPI and OpenMP have been used with respectable parallel scalability. For clusters with two or more SMPs on a single node, the mixed software model of threads within a node (OpenMP being a special case of threads because of the potential for highly efficient handling of the threads and memory by the compiler) and MPI between the nodes appears natural. Several researchers (*e.g.*, Bova *et al.*, 2000; Rabenseifner *et al.*, 2006) have used this mixed model with reasonable success. Thanks to the implementation of the shared and distributed memory parallelism in IFP-C3D, the OpenMP/MPI hybrid parallelism will be tested to increase the performance of the parallelism for distributed shared memory architecture machines.

7 1D/3D COUPLING CAPABILITIES

A 3D CFD (Computational Fluid Dynamics) code is commonly used for the optimisation of complex engine components (*e.g.* intake and exhaust systems of internal combustion engines, piston geometry and valve law). Generally, the main use of 3D calculations in contrast to 0D/1D simulations is the need for resolving three dimensional flow effects. However, the computational cost of 3D simulations is incompatible with the time requirement for virtual engine development. IFP-C3D had been fully integrated in the LMS Imagine Lab Platform where IFP have developed libraries dedicated to engines (IFP-ENGINE), to the combustion gas post processing (IFP-Exhaust) and for vehicle driving simulation (IFP-Drive). The platform integration of IFP-C3D code enables naturally a coupling procedure to be developed between the 0D/1D libraries and the 3D combustion code to allow 0D/1D/3D coupled simulations. The coupling process used a coupled combustion model in IFP-ENGINE that used 3D combustion numerical results. In fact, the coupled combustion model added to the IFP-ENGINE can modify the 3D input parameters, run the 3D code and take the 3D combustion numerical results for the simulation of the whole Engine system.

To complete the direct coupling method, we defined a temporal staggered algorithm to manage the time lag between the two codes. As shown in Figure 18, the temporal staggered algorithm can be split into three steps. First, the engine gas exchange system is calculated by IFP-ENGINE. When IFP-ENGINE reaches the intake valve closure (IVC) angle, the calculation is stopped and IFP-C3D starts. Then the 1D library sends the IVC angle, the trapped mass and the mixture composition to the 3D code. The 0-D/1-D informations are

TABLE 3
Description of the benchmark test cases used to evaluate the parallel CPU performances

Benchmark	Number of cells	Description
HEXA500k	500 000	Compression of a turbulent swirl in a hexahedral box.
PFI-FINE	420 000	Port fuel injection 2-valves gasoline engine intake phase calculation without fuel spray injection. Temporal moving grid algorithm.
PFI-FINE-SPR	420 000	Port fuel injection 2-valves gasoline engine intake phase calculation with fuel spray injection. Temporal moving grid algorithm.
PFI-4V	232 000	Port fuel injection 4-valves gasoline engine intake phase calculation without fuel spray injection. Temporal moving grid algorithm.
IIE	60 000	Gasoline homogeneous combustion calculation.
GDI	36 000	Direct injection gasoline engine with AKTIM spark ignition modelling (combustion calculation phase) and piezo electric injector.
DID	70 000	Direct injection diesel engine with spray.

TABLE 4
Benchmarks - Number of cells per processor used

Benchmarks / Cells per proc	1P	2P	4P	8P	16P	32P	64P
HEXA500k	500 000	250 000	125 000	62 500	31 250	15 625	7 532
PFI-FINE	420 000	210 000	105 000	52 500	26 250	13 125	6 562
PFI-FINE-SPR	420 000	210 000	105 000	52 500	26 250	13 125	6 562
PFI-4V	232 000	116 000	58 000	29 000	14 500	7 250	3 625
IIE	60 000	30 000	15 000	7 500	3 750	–	–
GDI	36 000	18 000	9 000	4 500	2 250	–	–
DID	70 000	35 000	17 500	8 750	4 375	2 175	–

TABLE 5
Speed-up and parallel efficiency of IFP-C3D using the MPI parallel library

Benchmark	1P	2P	4P	8P	16P	32P	64P
HEXA500k	1/100%	1.89/94.5%	3.65/91%	7.05/88%	11.74/71%	18.4/57.5%	23.1/36%
PFI-FINE	1/100%	1.47/73.5%	2.82/70%	5.01/62.5%	9.11/57%	15.7/49%	21.6/33.8%
PFI-FINE-SPR	1/100%	1.72/86%	3.31/82.8%	5.87/73%	10.7/67%	17.87/56%	22.0/34%
PFI-4V	1/100%	1.99/100%	3.45/86%	6.23/78%	10.75/67%	19.6/61%	24.1/38%
IIE	1/100%	1.70/85%	3.21/80%	6.51/81%	–	–	–
GDI	1/100%	1.49/75%	2.19/59%	4.91/61%	8.60/54%	–	–
DID	1/100%	1.55/77.5%	2.84/71%	4.54/56.8 %	6.12/38.2%	11.4/35.6%	–

TABLE 6
Parallel memory efficiency

Benchmark	1P	8P	16P	32P	128P
PFI-FINE-SPR	1	1.87	2.75	4.90	18.96
PFI-4V	1	1.71	2.45	4.30	16.46532602
DID	1	1.6	2.21	–	–

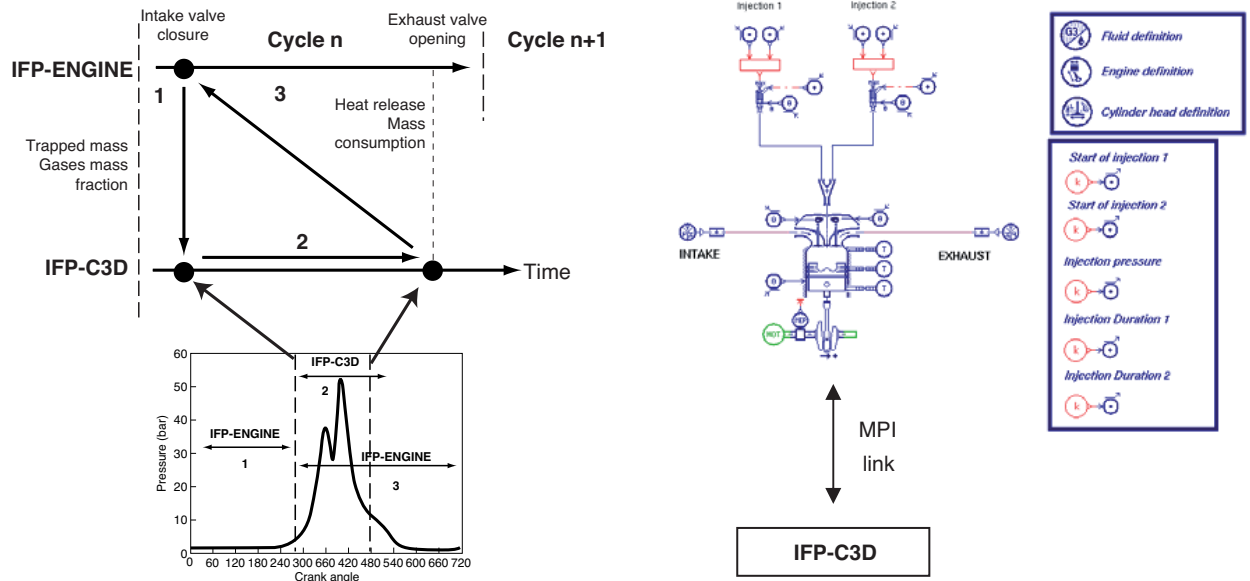


Figure 18

Temporal staggered algorithm uses for 0D/3D coupling.

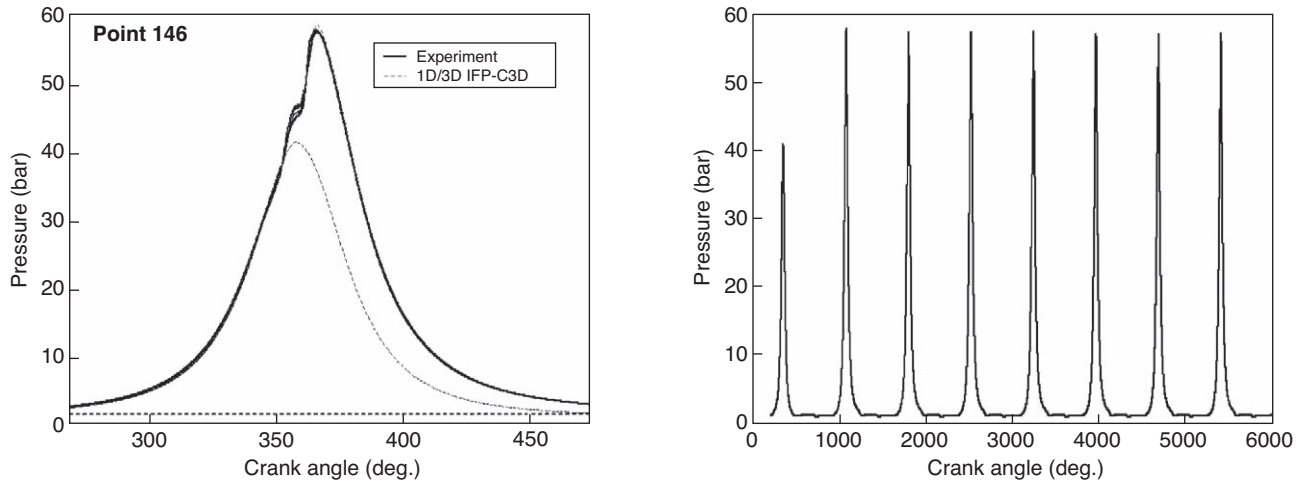


Figure 19

In-cylinder pressure comparison with experiment and 0D/1D/3D coupled calculation of a Direct Diesel Injection Engine (left). In-cylinder pressure histories (right).

used as boundary and initial conditions. The 3-D calculation is stopped at the exhaust valve opening (EVO) crank angle. Then, IFP-C3D sends the heat release histories (combustion, wall transfer) and the species mass consumption histories to the 1-D library. Finally, IFP-ENGINE restarts the calculation from the IVC to the next engine cycle. IFP-ENGINE uses during the combustion cycle the 3-D data to compute at each time-step the heat release and the mass consumption. To manage the coupled calculation, the MPI 2 (Message Passing Interface) paradigm is used to create the link between the two

codes. The MPI library is used with a master/slave mode. The master LMS-AMESim/IFP-ENGINE creates an intra-communicator with the IFP-C3D (slave) and can command and communicate with IFP-C3D at any time step. The master/slave mode allows the possibility to manage many slaves to extend the calculation for more than one cylinder but also to use 3-D modelling to other single components (such as air box, 3D pipe, etc.). Moreover, the CPU time need by 3D calculation and 1-D simulation are very different and generally 3-D calculations have to be performed using a parallel

architecture computer with a large memory. A homogeneous computer grid can be used with MPI paradigm and LMS-AMESim calculation can be performed on a laptop and 3-D calculations on a PC Cluster. This distributed calculation allows the simulation return time to be reduced and makes the coupled calculation time return compatible with industrial requirements.

This coupling procedure enables a large range of engine configurations to be simulated and makes easier to introduce an 0D/1D system simulator to perform 3D calculation. For instance, Bohbot *et al.* (2008) have used the coupled procedure to compute a direct injection Diesel engine with a 1D injector simulator which simulates the fuel injector behaviour for multi-injection events. Results plotted in Figure 19 show the in-cylinder pressure histories and the in-cylinder pressure comparison with test bench results that demonstrate the potential of the method. Knop *et al.* (2007) have used the coupling approach to study the benefits provided by a fully variable actuation device and an EGR circuit to optimize the combustion and to extend the operating range of Control Auto Ignition combustion concept.

8 RESULTS ON ENGINE CONFIGURATIONS

In the following paragraph, three engine configurations have been simulated that are used in most of the different models that we have presented in the previous sections. All these simulations were calculated on the multi-processor machine used previously (114 nodes containing 4 AMD quadri-core Barcelona processors). The number of processor used to perform each calculation depends of the size of the grid used.

8.1 HCCI Direct Injection Diesel Engine

Homogeneous Charge Compression Ignition, or HCCI, is a relatively new combustion technology. It is a hybrid of the traditional Spark Ignition (SI) and the compression ignition process (such as a Diesel engine). Unlike a traditional S.I. or Diesel engine, HCCI combustion takes place spontaneously and homogeneously without flame propagation. This eliminates heterogeneous air/fuel mixture regions. HCCI is a *lean* combustion process. These conditions translate to a lower local flame temperature which lower the amount of Nitric Oxide (NOx) produced in the process. NOx is a gas that is

believed to be responsible for the creation of ozone (O₃). IFP-C3D is able to simulate this new combustion process thanks to the TKI (auto-ignition) and ECFM3Z combustion and the pollutant formation models. A HCCI engine is simulated with IFP-C3D using 32 processors that allows to divide the calculation return time drastically by 11. The calculation return time for this test case is equal to 600 s. The remapping method is used to reduce the number of cells during the compression phase and to add cells during the expansion phase. We used 6 different wedge meshes generated with the internal mesher to perform the calculations listed in Table 7. Figure 20 shows three grids used to perform the calculation respectively in the crank angle range [−140°, −77°], [−51°, −35°], [+52°, +80°].

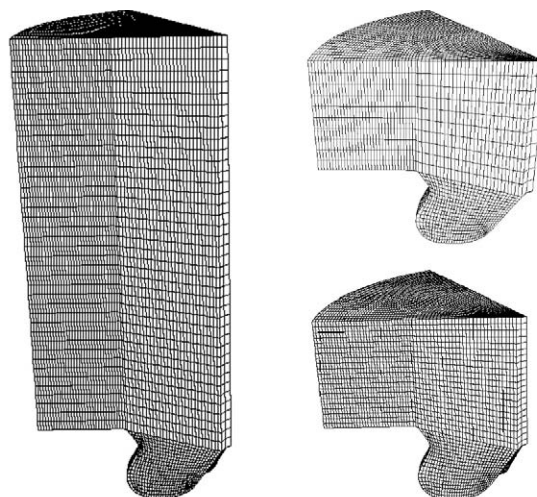


Figure 20

Grid used during the expansion phase (left −140°, and top right −51°) and during the expansion phase (bottom right +52°).

The main engine characteristics and the operating conditions are listed in Table 8. The injection is split into two injections at −7° and +11°. The EGR rate is equal to 53.5%. Raising the EGR level (up to 50%) has proven an effective way to control NOx emissions. It is proved that if combined with an appropriate injection strategy (the so-called “split injection strategy”) soot emissions can also be reduced.

TABLE 7

Grid used to perform the calculation

Crank angle (deg.)	−140°	−77°	−51°	−35°	+36°	+52°	+80°
Number of cells in Z-direction	40	20	10	05	10	21	42
Number of cells	78 000	45 500	29 250	21 125	29 250	47 125	81 250

TABLE 8
Operating conditions and engine parameters

Bore (mm)	81
Stroke (mm)	88
Connecting rod length (mm)	160
Compression ratio	16:1
Engine speed (rpm)	1500
IMEP (bar)	5.7
EGR rate (%)	53.5
Fuel air equivalence ratio	0.904
Number of hole	6
Injected mass (mg)	15.3
SOI 1st injection	7 BTDC
SOI 2nd injection	11 ATDC
Second injection mass	3.915

Simulations were performed using the up-to-date models on such running conditions. Figure 21 gives the comparison of computation results with experiment, including in-cylinder pressure. The simulation helped understand why a NADI™ combustion system did much better: the air entrainment, due to the vortex created by the first injection (cf. Fig. 22), brings fresh air into the bowl beneficial for the second injection, giving it ideal mixing and burning conditions. Such an air entrainment is entirely due to the narrow spray cone angle combined with the specific piston shape. This phenomenon does not occur in a conventional combustion system where the second injection takes place in already rich or even burnt mixtures.

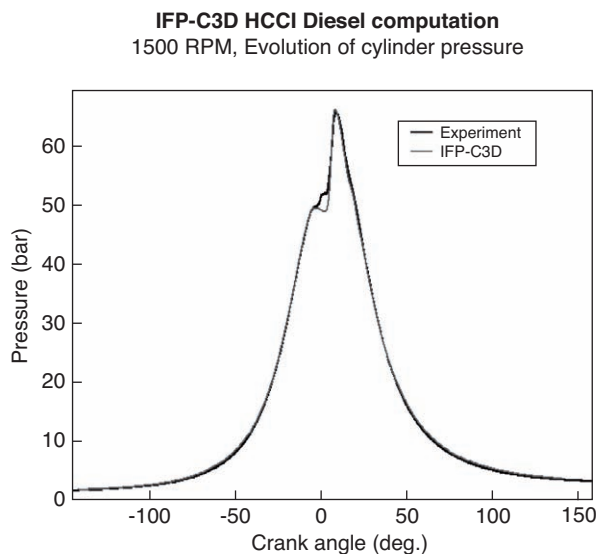


Figure 21
In-cylinder pressure comparison.

8.2 Turbocharged Spark Ignition Gasoline Engine

The aim of the present case is to test the developed models such as the one modelling the piezoelectric injector for an engine configuration. The combustion is computed using the ECFM model. The engine is a DISI single cylinder engine with 500 cm³ displacement volume for an operating point at 2000 RPM and IMEP = 3.8 bar with a high EGR rate (of around 30%) at intake.

Mesheres used to perform this calculation contain a number maximum of 430 000 cells during the intake calculation and 60 000 cells during the compression, combustion and expansion phase. The intake calculation is performed on 32 processors that allow to divide the simulation time by 20. The combustion phase is distributed on 12 processors to keep at least 5000 cells per processor to divide the calculation time by 8. The machine used is the same than previous calculation (nodes containing 4 AMD quadricore Barcelona processors). The calculation return time for the intake and combustion phase is equal to 4 hours.

As can be observed in Figure 23, fluid motion in the cylinder, trapped mass and in-cylinder composition are results of a previous complete intake stroke computation. At IVC the total residual mass fraction (EGR and burnt gases) is about 38%. It can be noticed that, in order to simplify the shape of the combustion chamber used for the computation of the intake stroke, the real piston (with bowl) was replaced by a flat piston. At the end of this intake computation, the physical quantities are transferred on the mesh representing the real shape of the combustion chamber (see Fig. 24) by the use of remapping algorithm. The mean in-cylinder fuel/air equivalence ratio is 0.5. The piezo-electric injector used was experimentally characterized in a high-pressure cell. The injection pressure is 20 MPa. Before performing engine computations, spray parameters (as droplet size and spray cone angle) were obtained by comparing experimental visualizations and measurements in a high-pressure cell with calculations under identical operating conditions. Here computations were carried out using only experimental injection and spark timing values. The computational mesh generated with the software Ansys HEXA is shown in Figure 24. It is composed of mesh cells with sides of about 1.5 mm (previous studies have shown that this cell size is sufficient to obtain a good level of accuracy). The number of cells in the vertical direction is increased or decreased (as a function of piston position) by using the remapping algorithm to obtain reasonable aspect ratios. It can be observed that no symmetry assumption is possible due to the piston shape. The spray injection and the combustion can be visualized in Figure 26 with the iso-surfaces of CO₂ mass.

The spray location and evolution in the chamber can be compared qualitatively to experimental endoscope visualizations. In Figure 25 it can be observed that the penetration and the evolution width of the spray is in good agreement with

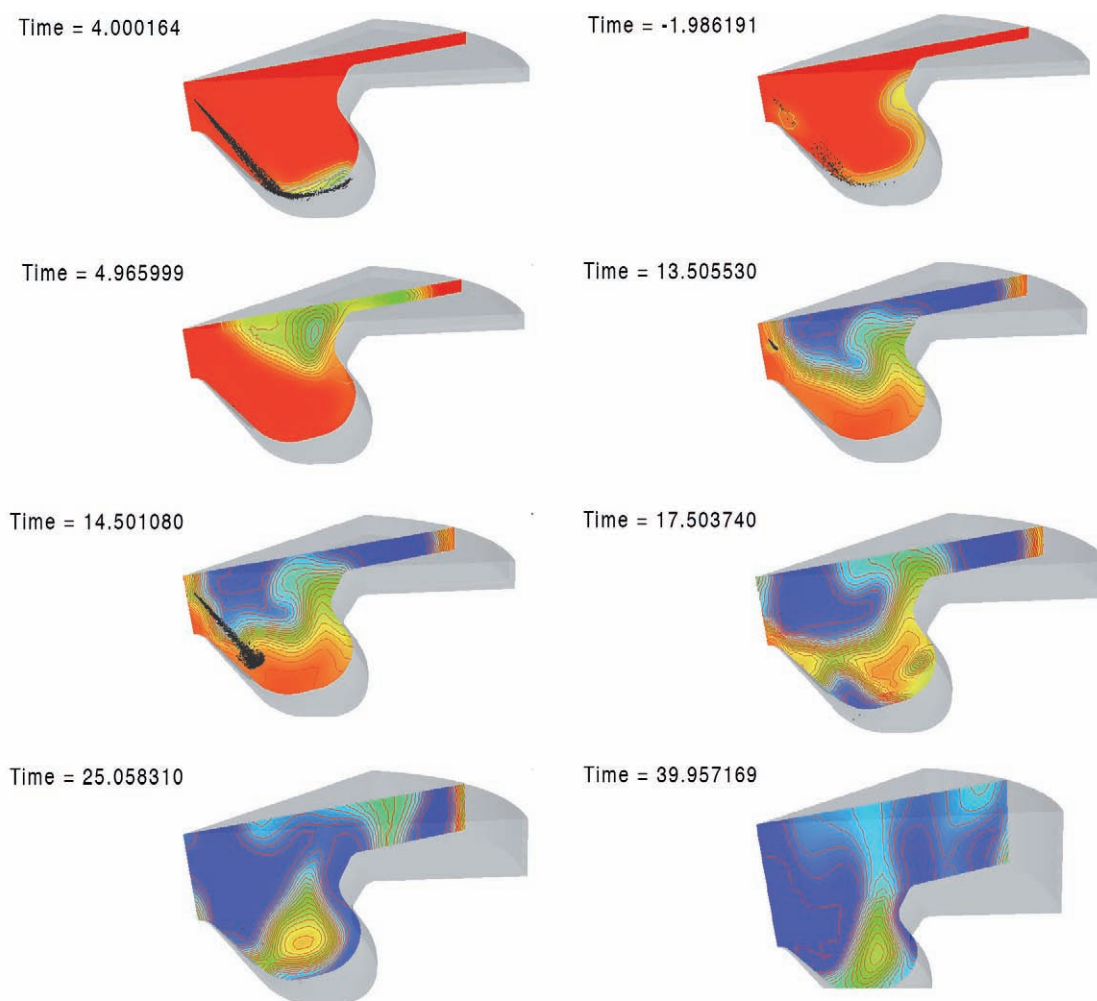


Figure 22

O₂ mass. The second injection timing is set to inject fuel on the unburnt gas region.

experimental observations (In this figure one can observe the spark plug representation using the AKTIM model can be observed). From this we can deduce that the fuel location and, as a consequence, the fuel/air equivalence ratio in the area of the spark plug is well represented. Hence, the mixing preparation which governs the start and the evolution of the combustion is well modelled.

The fuel is injected close to the spark plug. Hence, even if the mean fuel/air equivalence ratio is very low, the fuel/air equivalence ratio in the spark plug area is favourable to combustion. This is illustrated in Figure 8 where the distribution of fuel/air equivalence ratio is shown in a vertical plane passing through the spark plug at 7° CA after the start of injection (SOI). Added to the fact that the fuel/air equivalence ratio is good for the combustion in the area of the spark plug, the injection induces an increase of the Turbulent Kinetic Energy (TKE) which is important for the combustion evolution. The

evolution of the computed in-cylinder pressure compared to the experimental pressure is given in Figure 27. Computation results are in good agreement with experimental measurements.

CONCLUSION

IFP-C3D, an MPI parallel unstructured code devoted to the simulation of reactive flows with spray has been developed and validated on different engine configurations. The most up-to-date physical submodels have been developed in IFP-C3D like the ECFM-3Z model for the combustion, TKI and AKTIM model for respectively the auto-ignition and for the spark ignition, Lagrangian models for the fuel injection and for the liquid film. IFP-C3D has been fully parallelised with MPI library to be used on large Cluster machines. The simulation return time can be decreased by a factor of 20 on

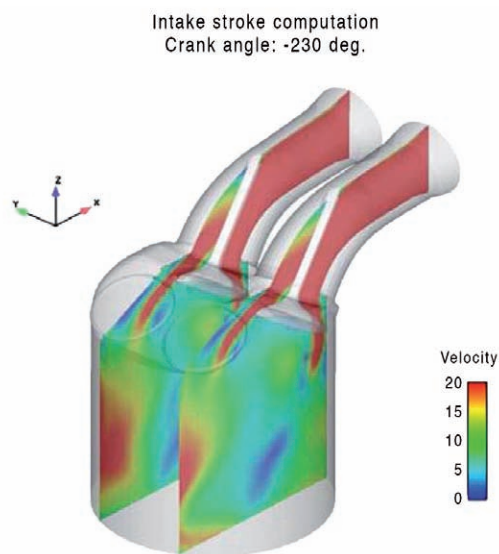


Figure 23
Intake stroke calculation.

32 processors which allow the simulation of complex engine configurations with refined grids. The next step will be the optimisation of the efficiency of the distributed parallelism. To reach the goal of 75% of efficiency up to 64 processors, a dynamic load balancing scheme have to be implemented to deal with non homogeneous combustion and fuel injection Lagrangian model. This load balancing scheme will be develop to take into account the calculation cost of combustion and chemistry reactions and the calculation cost of fuel spray particles evaporation. The run-time profiling done on the distributed parallel release shows that memory load and memory cache misses is mainly responsible of the loss of parallelism efficiency in distributed-shared memory machines. The memory load differences between processors can be solve by a load balancing scheme using this criterion but the memory cache miss problem is complex to solve because of the need to make large modification of the undirected loops and of the data structures. One way could be to develop an hybrid parallelism for distributed shared memory machines using OpenMP and MPI implementation in IFP-C3D. It seems natural to employ a hybrid programming model which uses OpenMP for parallelization inside the node and MPI for message passing between nodes. However,

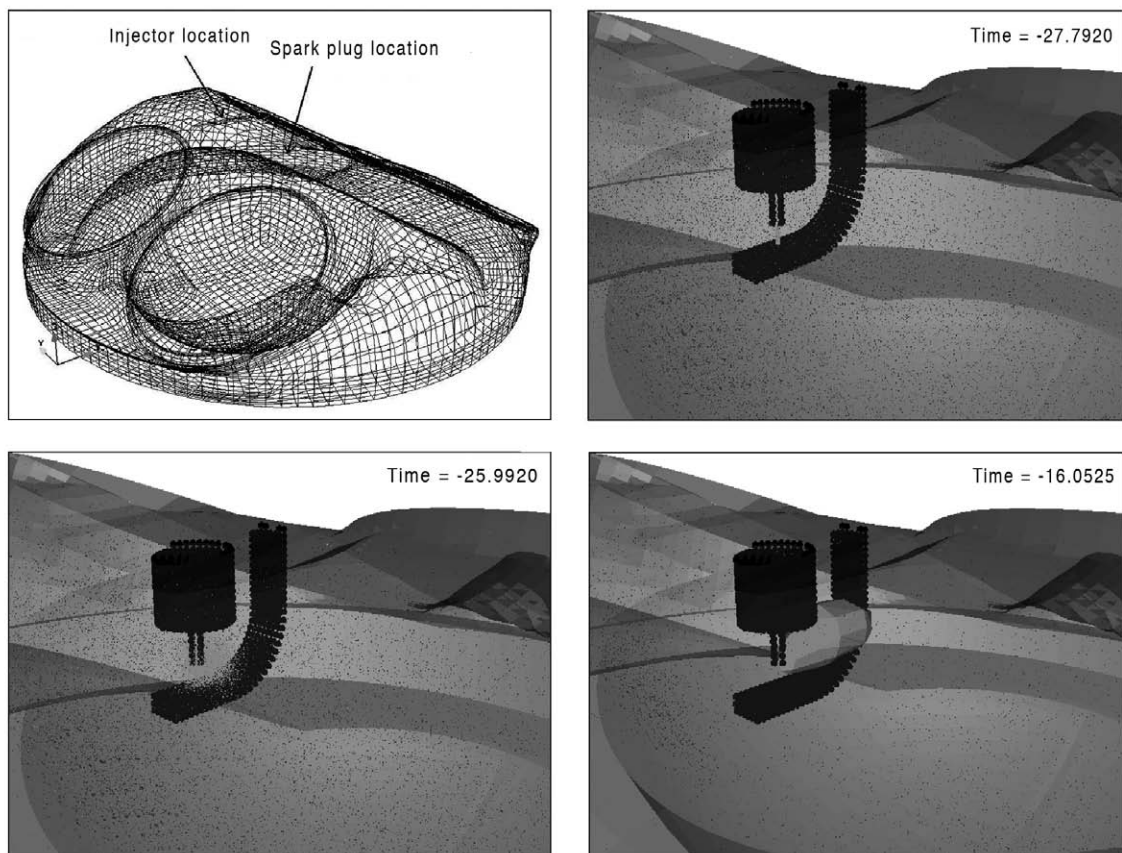


Figure 24
Mesh for the combustion calculation (left). View of the AKTIM SI modelling (spark particle, flame kernel) and of the fuel spray particles.

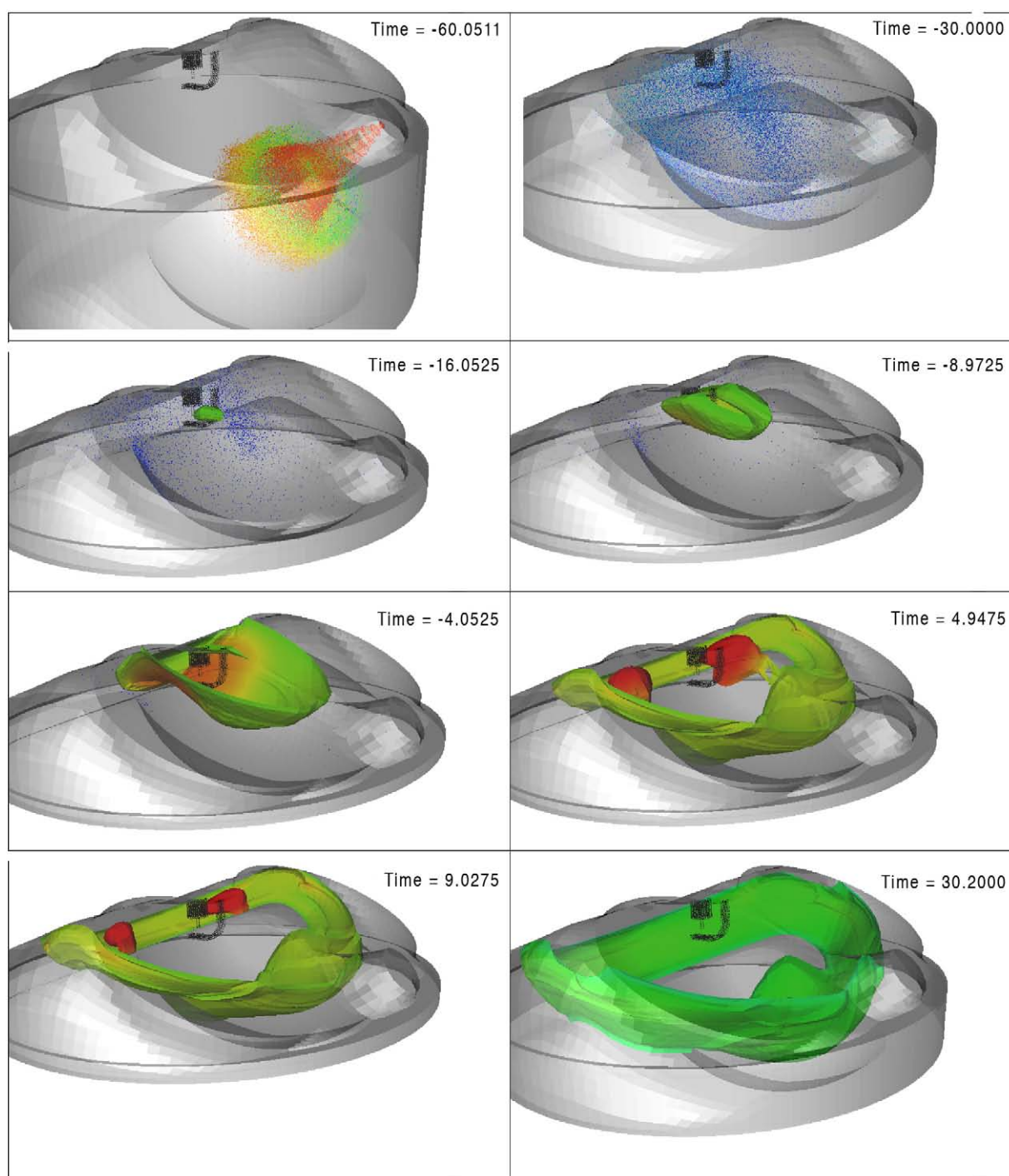


Figure 25

3D iso-surfaces of CO₂ mass with fuel spray injection.

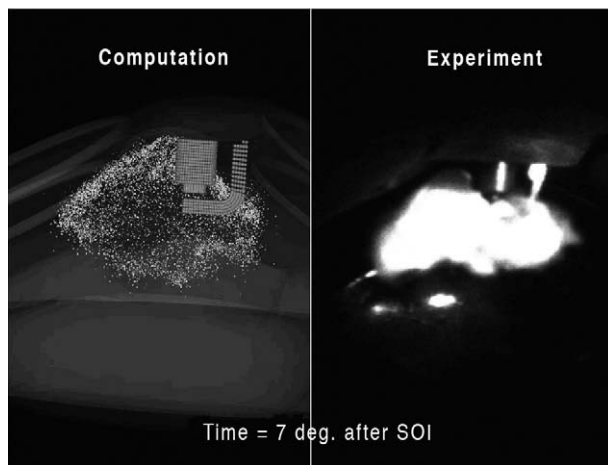


Figure 26

Visualization of the spray (left IFP-C3D, right experiment).

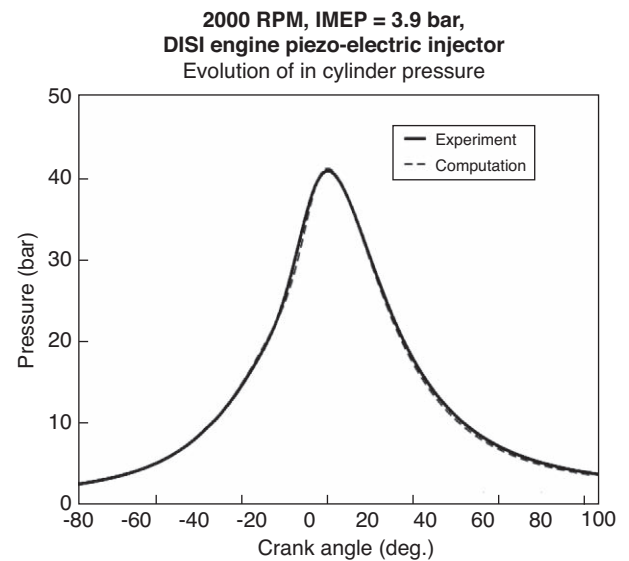


Figure 27

In-cylinder pressure.

there is always the option to use pure MPI and treat every CPU core as a separate entity with its own address space. And finally, looking at the multitude of machine architecture, the question arises whether it might be advantageous to employ a “mixed model” where more than one MPI process with multiple threads runs on a node so that there is at least some explicit intra-node communication.

New pollutant models based on chemical database interpolation (FPI) are studied to increase the predictivity of the pollutant emissions of internal engine combustion. An Eulerian multi-fluid modelling to simulate poly-disperse liquid spray is under development to increase the predictivity of the fuel injection. Simultaneously, numerical scheme are under development to simulate porous media flow to enable the simulation of the particle filter system. IFP-C3D uses RANS modelling for the turbulence and the development of the distributed parallelism which reduces drastically the simulation time allow the development of non statistical turbulence modelling like LES (Large Eddy Simulation) or RANS/LES (Detached Eddy Simulation). IFP-C3D is a code devoted to the simulation of internal combustion engine and contains the most up to date model for pollutant and combustion. The management of the different grids needed to make an engine simulation is original and gives for the engineer the possibility to manage the simulation of an entire engine cycle without the need to stop the calculation. IFP-C3D is well suited to the optimization of conventional configurations as well as the design of new engine combustion chambers (*e.g.* optimization of intake pipes or piston bowls). IFP-C3D is embedded in the LMS Imagine Lab AMESim environment as the 3D tool dedicated to CFD Engine calculations. As part of this system simulation environment, IFP-C3D can be

coupled with other models such as the cooling circuit (heat losses), the injection system (injection rate), the electronics (ignition parameters) or the valve- & crank-train system dynamics via 1D/3D coupling. These coupled calculations are the next step to simulate complex system that need the simulation of 3D components to enable full virtual system simulation and optimization.

REFERENCES

- Amsden A.A., O'Rourke P.J., Butler T.D. (1989) *KIVA-II, a Computer Program for Chemically Reactive Flows with Sprays*, Report LA-11560-MS, Los Alamos National Laboratory.
- Amsden A.A. (1992) *KIVA 3: A KIVA with Block Structured Mesh for Complex Geometry*, Los Alamos National Laboratory report LA 12503-MS.
- Amsden A.A. (1997) *KIVA 3-V: A Block Structured KIVA Program for Engine with Vertical or Canted Valves*, Los Alamos National Laboratory report LA 13313-MS.
- Angelberger C., Poinot T., Delhay B. (1997) Improving Near-Wall Combustion and Wall Heat Transfer Modeling in SI Engine Computations, *SAE paper 972881*.
- Aytekin Gel (1999) *A distributed-memory implementation of kiva-3 with refinements for large eddy simulation*, College of Engineering and Mineral Resources at West Virginia University.
- Baritaud T. (1989) Combustion and Fluid dynamic measurements in a spark ignition engine: effects of thermochemistry and velocity fields; turbulent flame speed, *SAE paper 892098*.
- Beard P., Duclos J.M., Habchi C., Bruneaux G., Mokeddem K., Baritaud T. (2000) Extension of Lagrangian-Eulerian Spray Modeling: Application to high Pressure Evaporating Diesel Sprays, *SAE paper 2000-01-1893*.
- Bohbot J., Gillet N. (2006) Impact of different mesh remapping techniques on 3D simulations in internal combustion engines, *Eccomass CFD Proceeding*, © TU Delft, Delft The Netherlands.

- Bohbot J., Chrysosakis Ch., Pacaud P., Benkenida A. (2008) Coupling of a 1-D Injection Model with a 3-D Combustion Code for Direct Injection Diesel Engine Simulations, *SAE paper* 2008-01-0358.
- Bova S.W., Breshears C.P., Cuicchi C.E., Demirbilek Z., Gabb H.A. (2000) Dual-level parallel analysis of harbor wave response using MPI and OpenMP, *Int. J. High Perf. C.* **14**, 49-64, 2000.
- Colin O. (2007) *Étude GSM D.C.I 2006 – Modélisation Diesel, Phase 1 : Amélioration du modèle de mél ECFM3Z*, Rapport IFP 59798 (In French).
- Colin O., Benkenida A. (2004) The 3-Zones Extended Coherent Flame Model (ECFM3Z) for Computing Premixed Diffusion Combustion, *Oil Gas Sci. Technol. – Rev. IFP* **59**, 6, 593-609.
- Colin O., Pires da Cruz A., Jay S. (2005) Detailed Chemistry Based Auto-ignition Model Including Low Temperature Phenomena Applied to 3-D Engine Calculations, *Proc. Combust. Institute* **30**, 2649-2656.
- Cuthill E.H., McKee J. (1969) Reducing the bandwidth of sparse symmetric matrices, ACM publication, *24th National Conference Assoc. Comp. Mach.*, 1969, pp. 157-172.
- Dongarra J.J., Hempel R., Hey A.J.G., Walker D.W. (1993) *A proposal for a user-level, message passing interface in a distributed memory environment*, Technical Report TM-12231, Oak Ridge National Laboratory, February 1993.
- Doss N., Gropp W., Lusk E., Skjellum A. (1993) *A model implementation of MPI*, Technical report, Argonne National Laboratory.
- Duclos J.-M., Colin O. (2001) Arc Kernel Tracking Ignition Model for 3D Spark-Ignition Engine Calculations, paper (2-25), *COMODIA*, 2001.
- Duclos J.M., Zolver M. (1998) 3D Modeling of Intake, Injection and Combustion in a DI-SI Engine under Homogeneous and Stratified Operating Conditions, *COMODIA*, 1998.
- Duclos J.M., Zolver M., Baritaud T. (1999) 3D Modeling of Combustion for DI-SI Engines, *Oil Gas Sci. Technol. – Rev. IFP* **54**, 2, 259-264.
- Gresho P.M. (1990) On the theory of semi-implicit projection methods for viscous incompressible flow and its implementation via a finite element method that also introduces a nearly consistent mass matrix, part 2: applications, *Int. J. Numer. Meth. Fl.* **11**, 621.
- Habchi C. (2006) Modélisation Tridimensionnelle de l'injection et de la préparation du mélange gaz/carburant dans les moteurs à combustion interne, *Mémoire de HDR*, Institut National Polytechnique de Toulouse.
- Habchi C., Torres A. (1992) A 3D Multi-Block Structured Version of the KIVA-II code, *First European CFD Conference proceedings*, pp. 502-512.
- Hautman D.J., Dryer F.L., Schug K.P., Glassman I. (1981) A Multiple-step Overall Kinetic Mechanism for the Oxidation of Hydrocarbons, *Combust. Sci. Technol.* **25**, 219-235.
- Heel H., Maly R., Weller H.G., Gosman A.D. (1998) Validation of SI Combustion Model over Range of Speed, Load, Equivalence Ratio and Spark Timing, *COMODIA*, 1998.
- Henriot S., Chaouche A., Cheve E., Duclos J.M., Leduc P., Menegazzi P., Monnier G., Ranini A. (1999) NSDI-3: A small bore GDI Engine, *SAE paper* 1999-01-0172.
- Heywood J.B. (1988) *Internal Combustion Engines Fundamentals*, Mc Graw Hill Publication.
- Jay S., Béard P., Pires da Cruz A. (2007) Modeling Coupled Processes of CO and Soot Formation and Oxidation for Conventional and HCCI Diesel Combustion, *SAE paper* 2007-01-0162.
- Knop V., Colin O., Benkenida A., Jay S. (2006) Adaptation of the ECFM combustion model to hydrogen internal combustion engines, *1st International Symposium on Hydrogen Internal Combustion Engines*, September 28-29, 2006.
- Knop V., Boulerie J.C., Bohbot J., Jay S. (2007) Controlling CAI™ combustion mode with VVA: A simulation approach, *SAE paper* 2007-01-0177.
- Leveque R.J. (1992) *Numerical Methods for Conservation Laws, Lectures in Mathematics*, ETH Zürich, Birkhäuser Verlag.
- Lewis J.G. (1982) Implementation of the Gibbs-Poole-Stockmayer and Gibbs King algorithm, *ACM T. Math. Software* **180**-189.
- Mantel T. (1992) Three-dimensional study of flame kernel formation around a spark plug, *SAE* 920587.
- Margolin L.G., Shashkov M. (2005) *Remapping, Recovery and Repair on a Staggered Grid*, Los Alamos National Laboratory Report, LA-UR-03-2230.
- Martinot S., Béard P., Roesler J., Garo A. (2001) Comparison and Coupling of Homogeneous Reactor and Flamelet Library Soot Modeling Approaches for Diesel Combustion, *SAE paper* 2001-01-3684.
- Naber J.D., Reitz R.D. (1988) Modeling engine spray/wall impingement, *SAE Technical Paper Serie*, March.
- Reitz R.D. (1987) Modeling atomization processes in high pressure vaporizing sprays, *Atomization Spray Technol.* **3**, 309-337.
- O'Rourke P.J., Amsden A.A. (1986) *Implementation of a Conjugate Residual Iteration in the KIVA Computer Program*, Report LA-10849-MS, Los Alamos National Laboratory.
- O'Rourke P.J., Sahota M.S. (1998) A Variable Explicit/Implicit Method for calculating Advection on Unstructured Meshes, *J. Comput. Phys.* **143**, 312-345.
- O'Rourke P.J., Zhang S., Sahota M.S. (1999) A Parallel, Unstructured-Mesh Methodology for Device-Scale Combustion Calculations, *Oil Gas Sci. Technol. – Rev. IFP* **54**, 2, 169-173.
- Patankar S.V. (1980) *Numerical Heat Transfer and Fluid Flow*, Hemisphere Publishing Corp., Washington D.C.
- Rabenseifner R., Hager G., Jost G. (2006) *Hybrid MPI/OpenMP Parallel Programming on Clusters of Multi-Core SMP Nodes*, Lecture Notes in Computer Science, Recent Advances in Parallel Virtual Machine and Message Passing Interface.
- Shashkov M., Wendroff B. (2004) The repair paradigm and application to conservation laws, *J. Comput. Phys.* **198**, 265-277.
- Sinha N., Cavallo P.A., Lee R.A., Hosangadi A., Kenzakowski D.C., Dash S.M., Affes H., Chu D. (1998) Novel CFD Techniques for In-Cylinder Flows on Tetrahedral Grids, *SAE paper* 980138.
- Sloan S.W. (1986) An algorithm for wave front reduction of sparse matrices, *Int. J. Numer. Meth. Eng.* **23**, 239-251.
- Tatschl R., Gabriel H.P., Priesching P. (2001) FIRE – A Genetic CFD Platform for DID Engine Mixture Formation and Combustion Simulation, *11th International Multidimensional Engine Modeling User's Group Meeting at the SAE Congress*, 2001.
- Torres A., Henriot S. (1994) 3D Modeling of Combustion in Lean Burn Four-Valve Engines: Influence of Intake Configuration, *COMODIA*, 1994.
- Torres A., Henriot S. (1996) Modeling the Effects of EGR Inhomogeneities Induced by Intake Systems in a Four-Valve Engine, *SAE paper* 961959.
- Walter B., Gatellier B. (2002) ent of the high power NADITM concept using dual mode Diesel combustion to achieve zero NOx and particulate emissions, *Proc. SAE Conf.* 2002-01-1744.

Walter B., Monteiro L., Miche M., Gatellier B. (2004) Improvement of Exhaust and Noise Emissions of the NADI™ concept using pre-mixed type combustion with multiple stages injection, *Proceedings of The Diesel Engine: TODAY and TOMORROW, SIA International Congress*, Lyon, 12-13 May 2004.

Zeldovitch YA.B., Sadovnikov P.YA., Frank-Kamenetskii D.A. (1947) *Oxidation of Nitrogen in Combustion*, translated by Shelef M., Academy of Science of USSR, Moscow.

Zolver M., Torres A., Klahr D. (2001) CFD and Combustion in Engines with an Unstructured Parallel Solver Based on KIVA, paper (1-14), *COMODIA*, 2001.

Zolver M., Klahr D., Torres A. (2002) An Unstructured Parallel Solver for Engine Intake and Combustion Stoke Simulation, *SAE paper* 2002-01-1120.

*Final manuscript received in September 2008
Published online in June 2009*

Copyright © 2009 Institut français du pétrole

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than IFP must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or a fee: Request permission from Documentation, Institut français du pétrole, fax. +33 1 47 52 70 78, or revueogst@ifp.fr.