High Performance 3D CFD Codes for Complex Piston Engine Applications
J. Bohbot, V. Knop, O. Laget, C. Angelberger, B. Réveillé
IFP, 1 et 4 avenue de Bois-Préau, 92852 Rueil-Malmaison, FRANCE

1 Introduction

Economic (fuel price), political (energetic independence), health (cancerous pollutants) and environmental (green house effect) concerns have greatly contributed to the evolution of pollutant emission regulations and fuel consumption regulations in the last decade. A better understanding of the combustion process for operating conditions near the stability limit will probably be the next step towards cleaner engines in order to use more aggressive control strategies. 3D CFD tools for engine modeling evolve to account for systems with increased complexity and to provide insight on the underlying physics of the stability limit. To compete with rapid engine prototyping, predictive simulation (therefore information) return times must be small, which is only possible with the efficient parallelization of 3D CFD codes with advanced physical sub-models. In the present paper, the recent effort for parallelizing IFP-C3D (RANS code) using MPI paradigm is described. This allows the use of larger meshes for reduced elapsed times. Combined with an improvement of the auto-ignition modeling, this allows to predictively compute a full cycle in the CAI combustion mode within 1.5 days. With a similarly linear speed-up, AVBP (LES code) was upgraded to multi-cylinder capabilities opening the way towards the study, understanding and control of cylinder-to-cylinder and cycle-to-cycle variability, which greatly influences engine efficiency.

2 Development of IFP-C3D-MPI

IFP-C3D [1], a hexahedral unstructured parallel solver of the RANS equations for unsteady turbulent chemically reactive flows in internal combustion engines was initially developed using the OpenMP paradigm (http://openmp.org). With the recent democratization of large super-scalar machines with more than 1000 processors and the will to greatly reduce simulation elapsed times, MPI parallelism was implemented in IFP-C3D while maintaining the OpenMP compiler directives so as to have a hybrid MPI/OpenMP code which may profit from the advantages of both paradigms (scalability for MPI and memory usage per node for OpenMP).

The cost of parallelizing using MPI is high, because parts of the code have to be completely restructured and the IO must be reorganized. In IFP-C3D, the process was time consuming and difficult because of the numerous embedded physical sub-models. The main difficulties came from:

- the combination of the Lagrangian discretization used for spray and spark ignition modeling with the Eulerian formulation of the Navier-Stokes equations;
- the complexity of the numerical scheme (time-splitting technique in 3 different phases);
- the Arbitrary Lagrangian Eulerian formulation used on staggered grid.

The MPI partitioning routine was first developed to manage the moving grid algorithm with remapping and the decomposition of the 3D mesh into sub-domains each managed by one processor. IFP-C3D uses the METIS generic graph partitioning library to decompose and distribute the 3D domains. MPI exchange routines were developed to handle data exchange at cells, vertex and face (fluid or wall) centers thereby allowing to deal with staggered data and all Lagrangian models. Spray and spark ignition were also parallelized using a domain decomposition strategy: each processor identifies the particles located in its part of the grid. This strategy was chosen because of the low CPU time cost of the Lagrangian part and its short lifetime (Lagrangian particles for liquid injection and spark ignition quickly vanish).

To assess the performance of the parallel version, the intake phase of a SI port fuel injection engine is calculated using a mesh containing 1.7 million hexahedra cells. Figure 1 presents the elapsed time improvement and the speed-up performance obtained with the MPI version of IFP-C3D on the CINES’ JADE superscalar machine (a 147 Tflop/s SGI Altix ICE 8200 supercomputer consisting of 1536 dual quad-core nodes, i.e. 12288 cores, with an Infiniband interconnect). The elapsed time is reduced by approximately a factor of 32 when using 256 cores instead of 8 cores. The speed-up is quasi linear and close to the ideal value. Because of the memory consumption of the implicit part of the solver, the speed-up can be greater than the theoretical value when the number of node used increase.
The next steps will be to implement dynamic load balancing and to activate hybrid MPI/OpenMP parallelization. Indeed, dynamic load balancing will help increase the parallelism efficiency by taking into account the CPU and memory load differences between domains due to the physics (injection combustion,...). Hybrid MPI/OpenMP seems a natural and promising solution Bova et al [18]; Rabenseifner et al. [19] on clusters made of interconnected SMP (Shared Memory Parallel) machines (case of multi node machine with several CPUs per node) with OpenMP on each SMP and MPI between the SMPs.

3 Chemical kinetics modeling

Auto-ignition is a fundamental process in internal combustion engines since it is the ignition process for Diesel engines and an efficiency-restricting process for Spark-Ignition engines (knock onset). Auto-ignition and the subsequent kinetics-driven combustion is also the ideal combustion process for HCCI and CAI engines. Therefore, the modeling of the whole kinetic ignition and combustion process is a key aspect for reactive 3D CFD applied to piston engines.

There is a general agreement on the need for accounting for complex chemistry to accurately predict auto-ignition in engines. Nevertheless, the simultaneous solving of flow and detailed chemistry within a fine mesh (more than 100,000 cells) is today unreachable in an industrial context because of the large number of species to transport (several hundreds). As a consequence, alternative methods such as kinetic mechanism reduction or implementation of look-up tables are necessary.

In the present paper, an update of an existing tabulation approach for the modeling of auto-ignition and subsequent heat release is proposed to improve the prediction accuracy. This model, called Tabulated Kinetics of Ignition (TKI), relies on the storing of auto-ignition delays and reaction rates of a progress variable computed with complex chemistry mechanisms. Compared to previous versions of this model [2,3], the definition of delays and reaction rates is updated in order to better match the chemical information.

In tabulation approaches [2-6], the chemistry information is rebuilt during the flow computation based on a progress variable. This variable describes the chemistry progress during the kinetics computation. For the TKI model [2,3], the progress variable is defined as a reduced temperature, following: $c = \frac{(T - T_{ini})}{(T_{fin} - T_{ini})}$ where $c$ is the progress variable, $T$ is the evolving temperature, $T_{ini}$ is the initial temperature and $T_{fin}$ is the final temperature, all temperatures being extracted from the a priori kinetic computations.

During experimental measurements of fuel reactivity, the auto-ignition delay is generally defined as the time to reach a predefined criterion that closely matches the peak heat release point. Such criteria can be the peak temperature gradient, OH light emission, CH* light emission,... In the previous versions of the TKI model, the peak temperature gradient was selected as the definition of the auto-ignition delay. However, early reactions and heat release appear before the peak heat release point. Therefore, such a definition introduces a lag between the heat release in the kinetics computation and that rebuilt during the 3D CFD computation since heat release in the CFD code only starts once auto-ignition has been detected. Such a discrepancy is particularly visible for engine operating points with auto-ignition during the expansion stroke. To provide a better description of the early heat release and an improved combustion timing, the auto-ignition delay of the new version of the TKI model is...
defined as the time to reach $c = 10^{-6}$. This value is a trade-off between the description of the very early heat release and the numerical precisions on the quantities defining $c$.

The new version of the TKI model also introduces a new way to define the reaction rates $\dot{c}$ that are the time derivatives of $c$. Previously, these reaction rates were directly the time derivative of $c$ coming from the kinetic computations at predefined values of the progress variable. These rates were stored in the look-up table and a linear interpolation along the progress variable was performed during the flow computation. This technique of reaction rate reconstruction induces a drift from the reference chemistry evolution because the reconstruction does not ensure to exactly evolve from one tabulation point to the following.

Therefore, a new and conservative definition of reaction rates is proposed: $\dot{c} = (c_{n+1} - c_n) / (t_{n+1} - t_n)$, where $c_n$ and $c_{n+1}$ are the lower and upper bounds of the progress variable interval wherein lies the instantaneous value of $c$ during the flow computation, while $t_n$ and $t_{n+1}$ are the instants at which $c_n$ and $c_{n+1}$ are reached during the kinetic computations, respectively. This update of the TKI model, a better match to the chemistry evolution, improves the complex chemistry description during flow computation. Figure 2 illustrates the great impact the update has on the heat release description for an academic test case (homogeneous constant pressure reactor).

### 4 Controlled Auto-Ignition

Among the existing concepts that help improve the efficiency of spark ignition engines at part load, Controlled Auto-Ignition™ (CAI™) is an effective way to lower both fuel consumption and pollutant emissions without major modifications of the engine design. The CAI concept is based on the auto-ignition of an air/fuel mixture, highly diluted with burnt gases, in order to achieve high indicated efficiency and low pollutant emissions through a low temperature combustion. The combustion mode entirely relies on auto-ignition and subsequent kinetics-driven heat release. Therefore, it is a concluding complex real-world test case for the TKI model.

<table>
<thead>
<tr>
<th>Engine Type</th>
<th>Single-cylinder</th>
</tr>
</thead>
<tbody>
<tr>
<td>Displacement</td>
<td>500 cm$^3$</td>
</tr>
<tr>
<td>Bore x Stroke</td>
<td>82.7 x 93 mm</td>
</tr>
<tr>
<td>Connecting rod</td>
<td>143 mm</td>
</tr>
<tr>
<td>Compression ratio</td>
<td>12.0:1</td>
</tr>
<tr>
<td>Engine speed [rpm]</td>
<td>2500</td>
</tr>
<tr>
<td>IMEP [bar]</td>
<td>1.6</td>
</tr>
<tr>
<td>Fuel/air eq. ratio</td>
<td>1.0</td>
</tr>
<tr>
<td>Injection system</td>
<td>Port Fuel Injection</td>
</tr>
<tr>
<td>Peak valve lift [mm]</td>
<td>3.0</td>
</tr>
</tbody>
</table>

Table 1: Main engine characteristics

Table 2: Operating point

Table 1 describes the test engine whereas Table 2 provides the operating point settings. For such a combustion mode, the trapping of burnt gases is of utmost importance and is controlled by the specific exhaust valve-lift profile. Therefore, the flow computation begins with the exhaust stroke, followed by the Negative Valve Overlap, the Induction and finally the power strokes. The main validation quantity for an engine without optical access is the mean in-cylinder pressure trace. Figure 3 indicates that the TKI model provides good agreement with experimental data for the chosen operating point. It is worth noting that such a result is largely dependent on the accurate modeling of the burnt gas trapping to store the right amount of thermal energy in the cylinder during the compression stroke.
Therefore, the prediction of CAI combustion mode imposes at least the modeling of a complete engine cycle and the accounting for the complete geometry (non-symmetrical combustion chamber) which is time consuming. Using hexahedral meshes with sizes ranging from 200,000 to 1,200,000 cells, the CPU time for 16 processors is about 12 days for the full cycle. Thanks to the nearly linear speed-up of the new MPI version of the IFP-C3D code, the elapsed time may be reduced to about 1.5 days (Table 3).

![Figure 3: In-cylinder pressure evolution](image)

<table>
<thead>
<tr>
<th>Mean cell nb</th>
<th>nCPUs</th>
<th>Duration [h]</th>
<th>nCPUs</th>
<th>Duration [h]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scavenging</td>
<td>700.000</td>
<td>16</td>
<td>42.5</td>
<td>128</td>
</tr>
<tr>
<td>NVO</td>
<td>300.000</td>
<td>16</td>
<td>5.6</td>
<td>64</td>
</tr>
<tr>
<td>Induction</td>
<td>950.000</td>
<td>16</td>
<td>221.6</td>
<td>128</td>
</tr>
<tr>
<td>Power strokes</td>
<td>400.000</td>
<td>16</td>
<td>19.4</td>
<td>64</td>
</tr>
<tr>
<td>Full cycle</td>
<td>550.000</td>
<td>16</td>
<td>~12 days</td>
<td>64 to 128</td>
</tr>
</tbody>
</table>

Table 3: CPU time consumption

5 Multi-cylinder & Multi-cycle LES computations

Cyclic combustion variability is a phenomenon that is still poorly understood. The prediction of its occurrence is complex due to an important number of global and local flow phenomena in the cylinder and ducts that potentially influence it. In this context, the usage of Large-Eddy Simulation (LES) to study and predict the occurrence of cyclic variability has received increasing interest, due to its inherent ability to address local, instantaneous flow phenomena, thus giving access to the cycle to cycle operation of an engine, a prerequisite for predicting cyclic variability. Published work has concerned the study of aerodynamics [7-9] and combustion [10,11] in single cylinder engines. Whilst such studies are helpful for understanding basic phenomena, they cannot address certain sources of combustion variability in real engines, and in particular the interactions between the different cylinders of a multi-cylinder engine. In this Section we present first results obtained applying the LES code AVBP to the study of flow and combustion in a four-cylinder spark-ignition engine.

5.1 Considered engine and working point

The engine considered here is a conventional in-line four cylinder gasoline engine. Figure 4 shows the computed domain comprising the four cylinders, their intake and exhaust ducts as well as intake and exhaust plena. This four stroke engine has a total displacement of 2 liters (single cylinder 500cc, stroke=86mm, bore=86mm and connecting rod length=152mm).

The computed operating point corresponds to a low load case at an engine speed of 2000 rpm. The fuel is gaseous propane injected far upstream the intake plenum, so that a homogeneous mixture with a fuel/air equivalence ratio of 0.7 is imposed at the inlet of the intake plenum.

Figure 4 shows an external view of the mesh. It is unstructured and comprises between 2.4 and 2.7 million hexahedral cells, with an approximate mesh size of 1mm inside the cylinder.
5.2 The AVBP code and the modeling approach

Computations are performed with the AVBP code [12]. AVBP is a fully compressible and explicit code that solves the multi-species reactive Navier-Stokes equations with realistic thermochemistry on hybrid unstructured, moving grids. AVBP is particularly well suited for exploiting the potential of massively parallel machines, using a domain decomposition method based on MPI.

For the present calculations, convective terms were discretized using a second order finite volume Lax-Wendroff scheme. Sub-grid stresses were modeled by a constant coefficient Smagorinsky model. Premixed combustion was addressed using the ECFM-LES model [10,11], combined with an evolution of the spark ignition model AKTIM-LES [10,14]. The code also includes Eulerian [15-17] and Lagrangian models for liquid sprays, which were not needed in the present computations.

For the multi-cylinder application 2 major evolutions of the code were required and implemented: an algorithm for the mesh movement of an arbitrary number of pistons and valves, and an ignition model capable of multi-location ignitions ISSIM 14].

5.3 Computations and results

The computations were carried out on the CINES' JADE cluster (see paragraph 2) on 256 cores. In terms of CPU time, a motored engine cycle lasts almost 1.5 day and a fired one 3.5 days.

A first step of the multi-cycle computations was to simulate 3 motored engine cycles (without combustion), in order to allow initiating the flow fields and establish physically sound starting conditions for the ensuing fired cycles and reduce the impact of the initialization procedure. Figure 6 shows the cylinder pressure traces of the motored and the fired cycles for the different cylinders as a function of crank angle. Although some level of variability between the 4 cylinders, and from cycle to cycle is apparent in the maximum cylinder pressure, their amplitude is small. Detailed analysis shows that these differences are due to small variations in the aerodynamics and turbulence between the cylinders, and from cycle to cycle. This is also apparent in Figure 5, which shows a FFT analysis of the time evolution of the pressure in the exhaust duct of cylinder 1. The observed cyclic variations have a frequency of twice and four times the rotational frequency of the engine. This indicates that the exhaust phase of this cylinder is influenced by the opening and closure of all the valves of the engine, and by pressure reflection.

After 3 motored cycles, the combustion model was switched on, and two fired cycles were computed. Figure 7 shows that the variations in aerodynamics and turbulence apparent from the motored cycles are amplified by combustion, leading to amplitudes of variations between the cylinders and from cycle to cycle that are higher than the ones observed in the motored case.

The presented simulations of fired cycles are still in progress, with the aim of computing a large number of cycles, allowing a detailed statistical analysis of the causes of the observed cyclic variations. The perspective opened up by such simulations is unique, as realizing such simulations in a competitive time will allow an unprecedented insight into the operation of a 4 cylinder engine under real operating conditions, with access to global and local; instantaneous flow phenomena (see Figure 7 for example) and their impact on engine efficiency and emissions.
Figure 7: Global view of the computational domain colored with velocity modulus, showing the flame front in cylinder two and the velocity field in cylinder four

6 Conclusion

This paper presents the modeling (auto-ignition, multi location ignition) and MPI parallelism efforts put by IFP in its 3D CFD codes IFP-C3D (RANS) and AVBP (LES) in order to be able to compute complex physical phenomena such as CAI combustion or complex systems such as a multi-cycle multi-cylinder configurations with rapid return times compatible with industry needs.

7 Acknowledgment

The LES multi-cylinder multi-cycle simulation was carried out in the frame of the CaMPaS project under grant ANR-06-CIS6-002 from the French National Research Agency (ANR). We acknowledge the strong support in terms of computing hours provided by the French supercomputing center CINES (http://www.cines.fr) under the DARI 2009 and 2010 programs of GENCI (http://www.genci.fr). Finally, the authors acknowledge the project partners: CERFACS (http://www.cerfacs.fr); EM2C (http://www.em2c.ecp.fr) and CINES, for their contributions to the success of the CaMPaS project.

8 References

7. P. Adomeit P., O. Lang, S. Pischinger, SAIE paper 2007-01-1412
17. L. Martinez, A. Benkenida, B. Cuenot, Fuel, 89 (2010), 219-228