Parallel Multi-Physics Simulation of Biomass Furnace and Cloud-based Workflow for SMEs

XAVIER BESSERON, University of Luxembourg, Luxembourg
HENRIK RUSCHE, Wikki GmbH, Germany
BERNHARD PETERS, University of Luxembourg, Luxembourg

Biomass combustion is a well-established process to produce energy that offers a credible alternative to reduce the consumption of fossil fuel. To optimize the process of biomass combustion, numerical simulation is a less expensive and time-effective approach than the experimental method. However, biomass combustion involves intricate physical phenomena that must be modeled (and validated) carefully, in the fuel bed and in the surrounding gas. With this level of complexity, these simulations require the use of High-Performance Computing (HPC) platforms and expertise, which are usually not affordable for manufacturing SMEs.

In this work, we developed a parallel simulation tool for the simulation of biomass furnaces that relies on a parallel coupling between Computation Fluid Dynamics (CFD) and Discrete Element Method (DEM). This approach is computation-intensive but provides accurate and detailed results for biomass combustion with a moving fuel bed. Our implementation combines FOAM-extend (for the gas phase) parallelized with MPI, and XDEM (for the solid particles) parallelized with OpenMP, to take advantage of HPC hardware. We also carry out a thorough performance evaluation of our implementation using an industrial biomass furnace setup. Additionally, we present a fully automated workflow that handles all steps from the user input to the analysis of the results. Hundreds of parameters can be modified, including the furnace geometry and fuel settings. The workflow prepares the simulation input, delegates the computing-intensive simulation to an HPC platform, and collects the results. Our solution is integrated into the Digital Marketplace of the CloudiFacturing EU project and is directly available to SMEs via a Cloud portal.

As a result, we provide a cutting-edge simulation of a biomass furnace running on HPC. With this tool, we demonstrate how HPC can benefit engineering and manufacturing SMEs, and empower them to compute and solve problems that cannot be tackled without.

CCS Concepts: • **Applied computing → Physical sciences and engineering**; • **Computing methodologies → Modeling and simulation**; **Parallel computing methodologies**.

Additional Key Words and Phrases: Biomass combustion, Multi-physics simulation, CFD-DEM, Parallel coupling, Cloudification

ACM Reference Format:
Xavier Besseron, Henrik Rusche, and Bernhard Peters. 2022. Parallel Multi-Physics Simulation of Biomass Furnace and Cloud-based Workflow for SMEs. In *Practice and Experience in Advanced Research Computing (PEARC '22), July 10–14, 2022, Boston, MA, USA.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3491418.3530294

1 INTRODUCTION

Biomass combustion is a well-established process to produce energy that offers a credible alternative to reduce the consumption of fossil fuel. Biomass is a renewable resource and is considered a carbon-neutral energy source because the carbon released into the atmosphere is used by the growing plants during the production of new biomass [6]. There exist numerous types of biomass furnaces, but typical installations on modern industrial setups are equipped with a forward acting grate that transports the fuel (i.e., the biomass particles like wood chips) through the combustion

Fig. 1. *Calidum Ember Furnace* produced by Enerstena UAB.

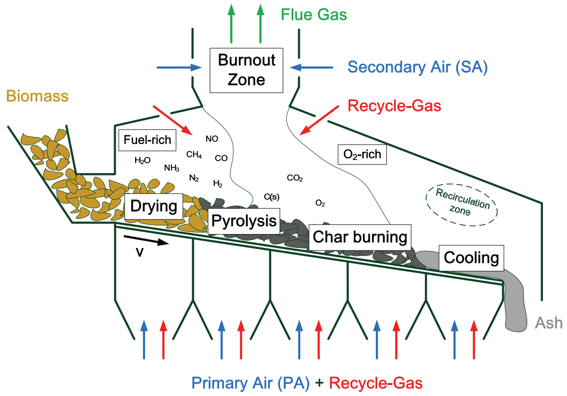

Fig. 2. Overview of the main physical processes occurring during the biomass combustion in a furnace, as proposed by Paces and Kozek in [23].

chamber (cf Figure 1). The performance of these biomass furnaces is evaluated in terms of thermal efficiency, harmful emissions, or remaining unburnt materials, and it is influenced by numerous parameters, including the design of the furnace, the operating conditions, and fuel composition [7]. To optimize the process of biomass combustion, numerical simulation is a more affordable and time-effective approach as opposed to the experimental method which is expensive. However, as a Dense Particulate Reaction System (DPRS) [33], the combustion in a biomass furnace involves complex physical phenomena that must be modeled (and validated) carefully, in the fuel bed (with particle motion and shrinking, heat transfer, drying, pyrolysis, gasification, etc.) and in the surrounding gas (with turbulence, combustion, radiation, etc.) [7, 11]. Figure 2 shows an example of a furnace modeling as proposed in [23]. With this high level of complexity, it is not surprising that the simulation of industrial biomass furnaces requires the use of High-Performance Computing (HPC) platforms and expertise to be conducted in a reasonable time [28, 33].

In this context, our research aims to make the simulation of complex DPRS available to small and medium-sized enterprises (SMEs) in a simple and flexible manner while leveraging HPC. Our approach is driven by a concrete industrial use-case and follows the technical specifications, requirements, and constraints defined by our industrial partner, the biomass furnace manufacturer Enerstena UAB. More concretely, our work closes the gap between Chemical Engineering and High-Performance numerical simulation by providing an advanced high-performance multi-physics simulation of a biomass furnace and integrating it in the CloudiFacturing EU project, which offers a *cloudified* digital marketplace targeting manufacturing SMEs.

Our contributions, which are novel related to the topic of biomass furnace simulations on HPC, are: (1) an original CFD-DEM coupling strategy using XDEM (parallelized with OpenMP) and FOAM-extend (parallelized with MPI); (2) a thorough performance evaluation on an industrial case; (3) an easy-to-use and fully automated workflow, from user input to results analysis, integrated into the CloudiFacturing digital marketplace.

This paper is organized as follows. First, we present an overview of the state-of-the-art related to the simulations of biomass combustion, their execution in parallel environments, and the integration of numerical simulations in cloud platforms. Section 3 presents our parallel implementation of the coupled CFD-DEM multi-physics simulation tool for biomass furnaces. Section 4 describes the complete execution workflow, from pre- to post-processing, and its technical integration in the Cloud. A detailed performance evaluation of our solution is carried out in Section 5, and then followed by the Conclusion.

2 RELATED WORKS

Biomass combustion is a complex physical process and it has been modeled in numerous ways in the literature [10]. The modeling covers different aspects [4, 7, 11]: the gas phase, the fuel bed and properties, and the different thermo-chemical processes involved in the conversion of biomass, like heat transfer, drying, pyrolysis, gasification, cooling, shrinking. The models can operate at the level of a single particle [4, 24] or on the complete packed bed [9, 19].

Original modeling based on a stochastic approach has been proposed [22], but biomass combustion is a multi-phase problem that falls in the category of Dense Particulate Reaction System (DPRS) and is typically addressed with one of the two techniques: Eulerian-Eulerian or Eulerian-Lagrangian [33]. In the **Eulerian-Eulerian** approach, the two phases (gas and solid particles) are considered as inter-penetrating continua. It is usually computationally efficient but fails to provide details on the behavior of individual particles. The alternative **Eulerian-Lagrangian** approach addresses this issue at the price of a more computation-intensive model in which the gas phase is still modeled as a continuum but the solid particles are modeled as discrete elements [32, 33]. In this category, the multi-phase particle-in-cell (MP-PIC) [17] creates lumps of particles representing the solid phase, and the CFD-DEM tracks each particle individually [4, 9, 25]. In summary, the CFD-DEM approach provides the highest accuracy but also the highest computation cost [7, 33] and it is the one we are using in this work.

In [33], Zhong and al. highlight that the use of CFD-DEM is not feasible for the simulation of industrial cases because it is too expensive in terms of computation and memory. Indeed, there are very few reports of CFD-DEM-based simulation of complete biomass furnace, even in an HPC configuration. [29] studies the scalability of a biomass reactor simulation up but using the MP-PIC approach arguing that the CFD-DEM approach is too costly for large number of particles. Recently in [26], a 2D model of a biomass furnace was tackled with the coupled code XDEM-OpenFOAM parallelized with OpenMP. The OpenFOAM-based code developed in [28] is used to compare the design of two different furnaces, and was run in parallel using 4 CPUs. [9] presented the simulation of biomass pyrolysis with MFiX running in parallel with 8 cores, and reported respectively 93.4% and 6.5% of the cost in the fluid model and in the DEM solver. It appears that the simulation of industrial-level biomass furnace is still [3] an open challenge and that could benefit from parallelization and High Performance Computing.

With such a level of complexity, it appears natural to offer HPC scientific simulations through a simplified interface in the Cloud for their use by non-experts. This process of *cloudification* is based on the concept of Application-as-a-Service (AaaS) that hides the technical aspects of high performance computing and automates the execution process. This approach has been demonstrated to have a positive impact in the development cycle of new products [16, 31] but the available options are not always known by the SMEs [14] that could benefit the most from it. Recent initiatives like CloudSME [30] and CloudiFacturing [13] aim at developing the cloudification and commercialization of scientific simulations and promoting its use among the SMEs. We follow on that new trend and present the technical details on the integration of our simulation tool in the CloudiFacturing platform.

3 PARALLEL MULTI-PHYSICS COUPLING FOR BIOMASS FURNACE

3.1 FOAM-extend and XDEM coupling

For the simulation of a complex industrial process of biomass combustion in a furnace, we based our approach on a coupled simulation between Computational Fluid Dynamics (CFD) for the gas phase and Discrete Element Method (DEM) for the particulate phase. While being computation intensive, the CFD-DEM approach gives the most suitable results thanks to the accurate description of the particle motion in DEM [33]. For our implementation, we rely on two well-known simulation frameworks FOAM-extend and XDEM. **FOAM-extend** is a fork of OpenFOAM open source library for Computational Fluid Dynamics (CFD). It is used to model the gas phase in the biomass furnace, including the fluid velocity and temperature, the species concentration and chemical reactions. **XDEM** (for eXtended Discrete Element Method) [25] is a numerical toolbox for the simulation of particulate materials. It extends the particles of the classical Discrete Element Method (DEM) with additional properties such as the thermodynamic state, internal temperature and species distribution. It is used to model the biomass fuel in the furnace, for the motion of the wood chips and for the internal temperature, chemical reactions, and composition of individual biomass particles.

The coupling between the two frameworks is implemented within a dedicated module in XDEM. In a simplified description, it covers three aspects. **Momentum transfer**: The presence (or not) of particles in the fluid creates a varying porosity which is applied to the CFD domain. Conversely, the gas velocity creates a drag force which is applied to the particles. **Heat transfer**: The temperature increases due to exothermic reactions within the particles are included in the CFD equation using an additional heat source. From the other side, the gas temperature solved in CFD is applied as a boundary condition at the surface of the particles. **Mass Transfer**: Gas species produced at the surface of the particles are injected in the CFD equations using dedicated mass source terms, one per gas species. Additionally, the species concentrations in the gas solved in CFD are applied as a boundary conditions at the surface of the particles.

In our implementation, the two libraries are coupled directly together using a dedicated simulation driver (i.e., a C++ program) that executes the respective parts of the simulation (CFD, DEM and coupling) in the appropriate way. This direct coupling approach provides a single-executable which allows the two libraries to reside in the same address space in the computer memory. Thanks to that, the data exchanges required for the coupling between CFD and DEM can be performed using direct memory access without extra copy or communication layer [27].

At the time of writing, there is no real-world experimental data available for the specific furnace modeled here, thus this article focuses mainly on the parallel coupling approach and performance assessment. However, the physics models and the implementation used for our simulation of biomass combustion (heating, drying, pyrolysis, gasification, etc.) have been previously developed, tested and validated for many years, at the level of single particles [2, 24], for particle packed beds [18, 19], and coupled with CFD [8, 21].

3.2 Parallelization Approach

In practice, for the HPC execution of our simulation, we rely on the parallel execution of each framework. FOAM-extend is parallelized using MPI. The cells of the CFD mesh are decomposed in independent meshes that are distributed to each process. On the other hand, XDEM supports parallel execution with MPI and OpenMP [20].

The CFD-DEM coupling is a **volume coupling** (as opposed to a surface coupling), which means that the amount of data exchanged is proportional to the volume of overlapping domain between CFD and DEM. As discussed in [27], this can have a significant impact on the performance of the execution. To address this issue, we adopt a **co-located partitioning strategy** [27] that ensures that the domain partitions between the independent physics modules are coordinated to favor the spatial locality and avoid complex communication patterns.

This approach was detailed in [27] where we also identified the constraints related to the partition alignment between the two coupled libraries. More precisely, because the internal calculation of each coupled library is unaware of the coupling with the other, the coupling layer has to make sure that all the coupled data is properly injected (e.g., as boundary conditions) in the respective solver. For parallel execution, it means that if the data required to update the boundary condition is available locally in the same process, the update can take place seamlessly using the sequential implementation and direct memory accesses. Otherwise, the coupling layer has to perform additional communications

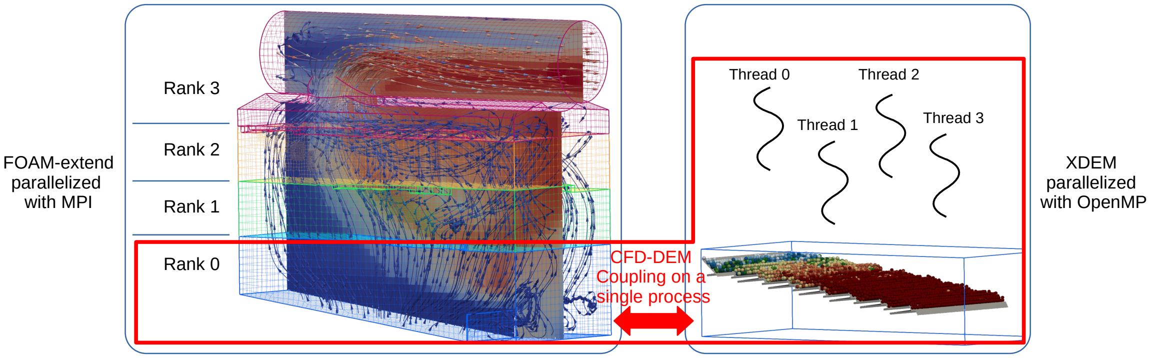
FOAM-extend parallelized with MPI
Rank 3
Rank 2
Rank 1
Rank 0
CFD-DEM Coupling on a single process
Thread 0
Thread 1
Thread 2
Thread 3
XDEM parallelized with OpenMP

Fig. 3. On the left, the CFD domain is parallelized with MPI. On the right, the DEM particles are parallelized with OpenMP. Because the particle bed is always located at the bottom of the furnace, we partition the CFD domain to guarantee that all the particles are in the same CFD domain.

(referred as *inter-partition inter-physics*) which are difficult and problematic to implement because they require accessing the internal data structures of the coupled libraries. In [27], we chose to enforce partition alignment between the coupled solvers to make sure that all the required data is available locally. We validated our approach with micro-benchmarks and highlighted the benefits of reduced communication with macro-benchmarks, in particular for volume coupling.

In the current work, we apply a similar technique to our industrial case in order to avoid inter-partition inter-physics communications. In this particular biomass furnace setup, the particle bed is always located at the bottom of the furnace. As illustrated on Figure 3, we define the CFD partitions to guarantee that the first partition is located at the bottom of the CFD domain and contains the whole particle bed (this is referred later as *co-location constraint*). On the side of XDEM, all the particles are assigned to the same process which executes its part of the computation in parallel using OpenMP. This results in a hybrid approach where FOAM-extend runs in parallel with MPI, XDEM runs in parallel with OpenMP, and all the coupling data exchanges take place in a single process using direct memory accesses. When starting the execution of coupled executable, process with rank 0 is started with multiple OpenMP threads and the other ranks are limited to one thread. To ensure that the first partition of the CFD decomposition always contains all the particles, we developed a dedicated decomposition method called *co-located*. It is loaded as a plugin in FOAM-extend and is used to partition the CFD mesh of the furnace. This decomposition method works on the graph representation of the CFD mesh, and merges together all the vertices corresponding to CFD cells surrounding the particle bed. Then this graph is partitioned with the builtin decomposition method based on SCOTCH. As the result, the bottom part of the CFD mesh and the DEM particle are always co-located in the same process. While this can sometimes cause imbalance in the decomposition, this approach avoid additional communication between CFD and DEM across processes.

Using the terminology defined in [27] (Fig. 1), we can summarize our parallel coupling approach based on the type of communications. **Inter-partition intra-physics** data exchange (i.e., within FOAM-extend) uses the MPI communication layer with the native implementation of FOAM-extend; **Intra-partition inter-physics** data exchange (i.e., within the same partition between FOAM-extend and XDEM) uses direct memory access because the two libraries are linked together into one executable and share the same address space; **Inter-partition inter-physics** data exchange are non-existent because the co-located partitioning strategy ensures that the coupled data from the two physics modules are located in the same process.

4 FULLY INTEGRATED WORKFLOW IN THE CLOUD

We have integrated our biomass furnace application as an experiment in the CloudiFacturing initiative. CloudiFacturing [13] is an EU-funded Innovation Action project that aims at developing the digitalization of the manufacturing sector. It brings together all the relevant actors (research organization, independent software vendors, manufacturing companies, technology consultants, resource providers, etc.) in order to promote the cloudification of services for the use by manufacturing SMEs. The integrated applications are available online on the CloudiFacturing Digital Marketplace on pay-per-use model. CloudiFacturing can execute complex scientific workflows and simulations on HPC platforms transparently for the users.

In practice, the automation of a scientific workflow like the biomass furnace simulation is a challenging task. First because the parameter space is huge and thus it must be limited and controlled carefully. Additionally, the complete workflow, from the inputs to the result analysis, is composed of numerous intermediate steps. Figure 4 shows an overview of the workflow that has been developed for the biomass furnace simulation in the CloudiFacturing platform. It can be decomposed in 5 parts. **Inputs** have been defined together with our industrial partner in the project, the furnace manufacturer Enerstena UAB. They consist of a set of parameters that represents relevant characteristics of a furnace. The goal here is not to be able to describe any furnace, but instead to focus on a specific type of experimental biomass furnace which is of interest for the manufacturer. The parameters cover a wide area of settings and offer a many hundred degrees of freedom in total: the furnace geometry under definite constraints; the grates (number, position, motion); the air inlets (number, position, composition, temperature,velocity, etc.); the fuel particles (size, composition,

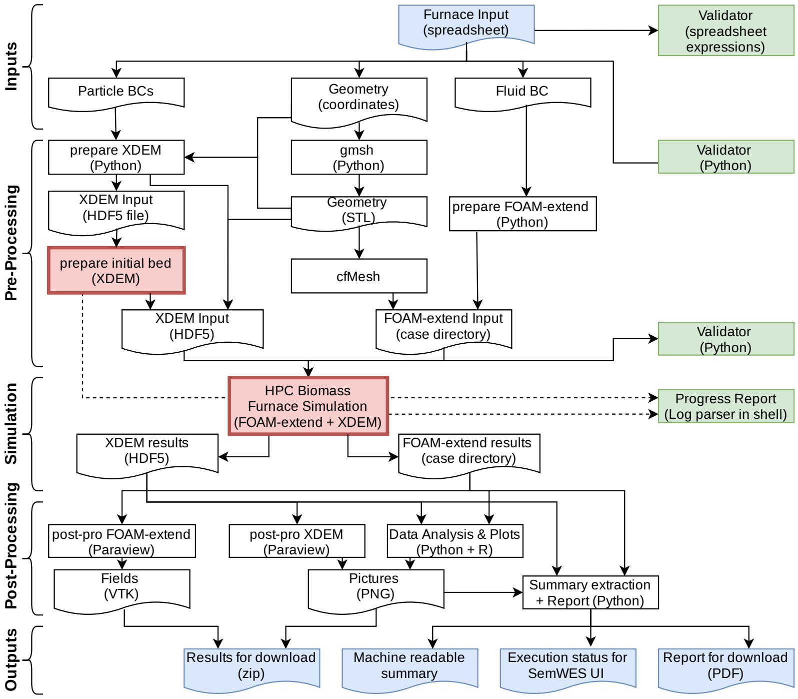
Inputs
Furnace Input (spreadsheet)
Validator (spreadsheet expressions)
Particle BCs
Geometry (coordinates)
Fluid BC
Pre-Processing
prepare XDEM (Python)
gmsh (Python)
Validator (Python)
XDEM Input (HDF5 file)
Geometry (STL)
prepare FOAM-extend (Python)
prepare initial bed (XDEM)
cfMesh
XDEM Input (HDF5)
FOAM-extend Input (case directory)
Validator (Python)
Simulation
HPC Biomass Furnace Simulation (FOAM-extend + XDEM)
Progress Report (Log parser in shell)
XDEM results (HDF5)
FOAM-extend results (case directory)
Post-Processing
post-pro FOAM-extend (Paraview)
post-pro XDEM (Paraview)
Data Analysis & Plots (Python + R)
Fields (VTK)
Pictures (PNG)
Summary extraction + Report (Python)
Outputs
Results for download (zip)
Machine readable summary
Execution status for SemWES UI
Report for download (PDF)

Fig. 4. Workflow of the *cloudified* multi-physics biomass furnace as it is executed by the CloudiFacturing platform. All the steps are automated, from the input file uploaded by the user (at the top) to the outputs available for download after the execution.

density, humidity, porosity, etc.); and more. All the possible variables are organized and documented in a spreadsheet template that can be conveniently prepared offline by the user and submitted to the CloudiFacturing portal when ready. The use of a spreadsheet file for the input allows keeping together all the parameters and simplifies the submission process. Furthermore, intermediate calculations can be used and basic validation rules of the input values are included in the template. **Pre-processing** parses all the parameters provided in the input spreadsheet and create all the input files for the simulation application. The CFD input requires the definition of the initial conditions, the generation of the mesh of the furnace and its decomposition for the parallel execution. The XDEM input requires the preparation of a realistic initial packed bed of particles. This task is performed by running a pure DEM solver to deposit particles on the furnace grates according to the geometry and settings of the input. This requires a few minutes of simulation using all the cores of a single computing node with OpenMP. The initial and generated inputs are validated at different levels to ensure the consistency of the data, and give descriptive error message when needed. The **Simulation** phase is the core of the workflow and represents the majority of the time. It runs the parallel solver described in Section 3 on the input generated using a full computing node of the IT4Innovations HPC platform. **Post-processing** gathers and processes the results of XDEM and FOAM-extend. Relevant metrics and analysis have been defined to answer the need of Enerstena UAB and are compiled into a summary report. It includes visualizations of the furnace (rendered with headless ParaView), quantitative metrics at specific points of the furnace (e.g., gas temperature at the exhaust pipe) and many plots representing of the evolution of values along the length of the grate or over the time. The **Outputs**, composed of the execution status, the report and additional materials, are sent back to the SemWES engine to be displayed and made available for download in the CloudiFacturing portal.

Under the hood, we provide all the required software, scripts and data in the form of a Singularity [15] image. The execution is delegated to the SemWES engine [13] that allocates the resources and manages the execution on the IT4Innovations HPC platform. The execution is continuously monitored and the progress is reported in the web-portal using the mechanisms provided by the SemWES engine. The SemWES engine executes a single `mpiexec` command with the Singularity image of our application with consists of the whole workflow described in Figure 4. With these startup settings (outside of our control), the executed processes are not attached to CPU cores, which allows flexibility to support a wide variety of applications for the CloudiFacturing platform. In our case, the monitoring process and the pre- and post-processing steps are only executed by the rank 0 while the simulation step is executed using all the ranks.

5 PERFORMANCE EVALUATION

For the performance evaluation, the input settings of the biomass furnace were defined by our partner to match an experimental furnace currently in construction at Enerstena UAB in Lithuania. The furnace dimensions are $2.51m \times 1.14m \times 2.07m$ $(L \times W \times H)$ topped with an exhaust pipe of $0.6m$ diameter. It features 6 primary air inlets from the bottom, 11 secondary air inlets on each side and 1 tertiary air inlet on the exhaust pipe. The CFD mesh generated with this configuration is composed of 60,001 cells. The furnace also contains 8 static grates and 6 moving grates with an average slope of 7.5 degrees. The initial fuel bed is $10cm$ high and composed of spherical wood particles of $3cm$ diameter with 40% humidity. It represents an initial number of 9141 particles. New wood chips are continuously injected at the top side of the grates at a rate of $439kg/h$.

For the software, we used a modified version of FOAM-extend 3.2 with the Git hash $3912d19b$, XDEM with the Git hash $7a100737$, and the CFG-BioOpt automated workflow described in Section 4 with the Git hash $11299e41$. They were all built together in a Singularity container based on Ubuntu 20.04. The executions were carried out on the *Barbora* cluster of the IT4Innovations HPC platform. Each independent execution was performed on a single standard computing node (i.e., non-GPU) that is equipped with 192 GB of memory and two 18-core Intel Cascade Lake 6240 2.6 GHz processors, for a total of 36 cores per computing node. In our default configuration, FOAM-extend is decomposed in 4 MPI processes and XDEM uses 28 OpenMP threads. Because some studies required to vary many parameters (e.g., number of nodes and cores, decomposition method), some measurements were limited on the first 10 seconds of the simulation to obtain results in a reasonable time.

With the configuration described above, the full automated workflow requires about 34 hours and 45 minutes on a single computing node to simulate 20 minutes of the functioning of the furnace. Table 1 shows a breakdown of the execution per stage of the workflow. Around 98% of the time is spent in the parallel coupled CFD-DEM simulation itself, while less than 2% is spent in pre- and post-processing. A large amount of time of the post-processing is spent in the reconstruction of the distributed output of FOAM-extend.

Table 1. Breakdown of the workflow execution for 20 minutes of simulated time.

Pre-processing	**5min 30s**	
- in XDEM bed packing	5min 10s	32 OpenMP threads
Simulation	**34h 10min 30s**	
- in FOAM-extend code	31h 51min 2s	4 MPI processes
- in XDEM code	2h 19min 28s	28 OpenMP threads
Post-processing	**29min 33s**	
- in FOAM-extend reconstructPar	25min 21s	

5.1 FOAM-extend decomposition and scalability

We study the impact of our *co-located* decomposition method in comparison with two other built-in decomposition methods of FOAM-extend: *scotch* which is based on the SCOTCH library and *simple* which makes a geometric decomposition on the vertical axis. Left of Figure 5 shows the maximum number of cells between the partitions after decomposition. For 2 and 4 partitions, all the decomposition methods give similar result in terms of load-balancing with less than 1% of imbalance. On 6 and 8 processes, the *co-located* method gives an imbalance (43% and 37% respectively) compared to the others, but this is the only one that can be used because the resulting partitioning with *scotch* and *simple* breaks co-location constraint imposed by our coupling approach presented in Section 3.2. On the right, we report the time spent in the CFD part for an execution limited to the first 10 seconds of simulated time. The execution in sequential (i.e., the *none* decomposition method) required 447.9s of CFD computation and only 122.6s, 125.0s and 129.8s

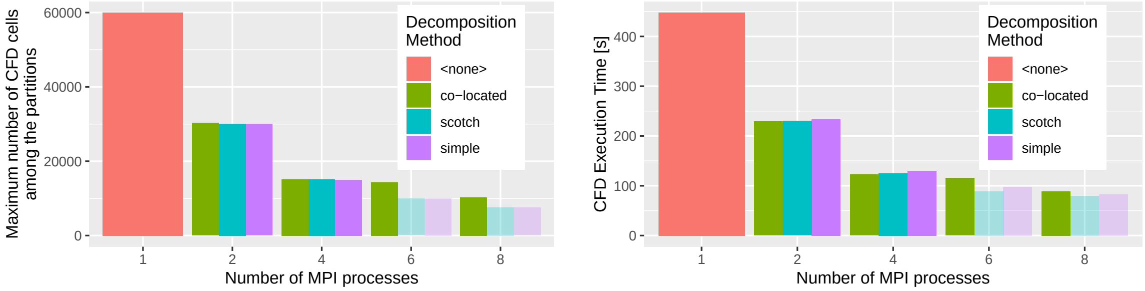

Fig. 5. Comparison of the maximum number of CFD cells among the partitions after decomposition (on the left) and the corresponding execution time of the CFD part for a simulated time of 10 seconds (on the right) for different number of processes and different decomposition methods. The lighter colors with 6 and 8 processors indicate the invalid configurations that might not give correct results for our coupling strategy.

with 4 MPI processes for the decomposition methods *co-located*, *scotch* and *simple* respectively. There is no significant impact of our constrained decomposition method at this level of parallelism.

With more processes, imposing the co-location constraint with *co-located* shows a performance impact of only 11% on 8 processes respectively compared to *scotch*, which relatively small compared to the unbalance of 37% shown on Figure 5 (left). But for this case, this is also expected because the size of the CFD problem is relatively small with only 60,000 cells and limits the scalability of the CFD part (OpenFOAM typically requires 10k to 50k CFD cells per core for best performance [12]). Finally, it is worth noting that to allow *scotch* and *simple* to work correctly with our coupling approach, additional communications (inter-partition inter-physics) would have to be implemented. It means that for these cases, the execution time indicated on Figure 5 (right) is a lowerbound of the actual time.

5.2 XDEM scalability

In this section, we focus on the performance of the XDEM computation. In this configuration, we still run the coupled application. The CFD part is executed by FOAM-extend on 4 MPI processes. The DEM part is executed by XDEM only on the first MPI process which contains all the particles. As mentioned in Section 4, the SemWES engine does not bind the MPI processes to the CPU cores which allows the OpenMP threads of XDEM in the rank 0 to use all the necessary cores. So for this experiment, we vary the number of OpenMP threads on the first MPI process, and the other processes are set to use only one thread. We measure the computation time spent in the XDEM simulation for an execution limited to the first 10 seconds of simulated time and the values are reported on Figure 6. We see an important benefit of the OpenMP parallelization in XDEM. With 36 OpenMP threads, there is an important penalty because the computing cores are already occupied by the other MPI processes solving the upper part of the CFD domain.

The speedup is calculated on Figure 7 and shows a maximum speedup of 16.8 when using 28 OpenMP threads. Ultimately, even though the CFD code cannot exploit efficiently all the cores of computing node (cf Section 5.1), the rest of the cores can be leveraged by XDEM to accelerate the particle simulation.

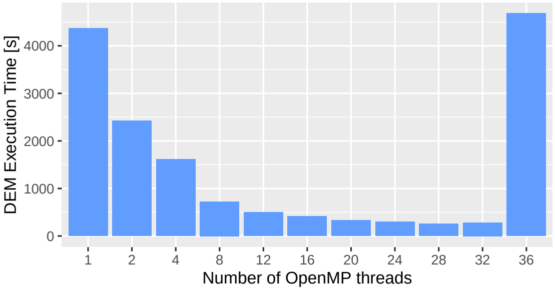

Fig. 6. Execution time spent in XDEM for different number of OpenMP threads. For 36 threads, the time increases greatly because some computing cores are shared with the CFD processes.

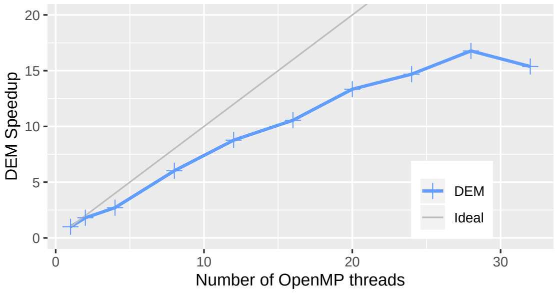

Fig. 7. Speedup of the XDEM computation.

5.3 Simulation Progress

With Figure 8, we look at the progress of the simulation over the time. We display here a simulation of 1600 seconds that required 56.2 hours of execution. The horizontal axis is the clock time since the start of the simulation. The black line (right Y-axis) indicates the progression of the simulation: the steeper is the slope, the faster the simulation is running. The filled colors (left Y-axis) show the amount of computation time required for each second of simulated time.

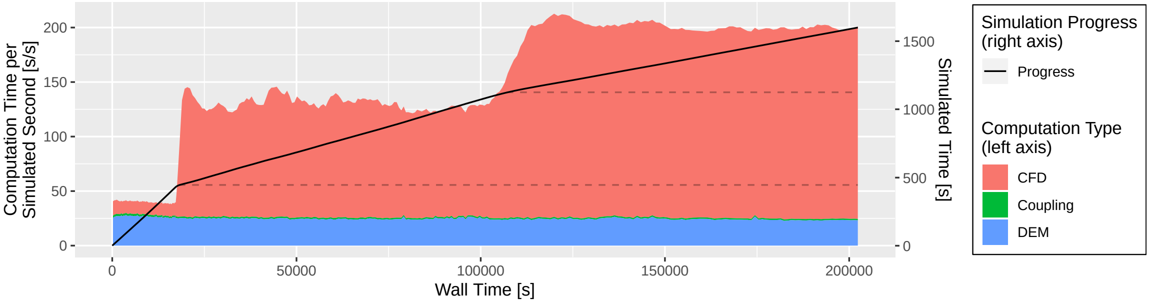

Fig. 8. Evolution of the computation load and progress with the time. The X-axis represents the wall time or clock time since the start of the simulation. The left Y-axis, with the filled colors, indicates the amount of computation time required for each second of simulation. The right Y-axis, with the black line, indicates the progress of the simulation.

We can notice two main events for which amount of computation work increases significantly and the simulation progress slows down: first around 445s and then around 1125s in the simulation time. At these points of interest, the computation required in CFD increases notably as shown with the colors (scale on the left axis) while the computation load for DEM and the Coupling stays relatively constant during the whole simulation. The first increase at 445s of simulated time is caused by the lighting-up of the furnace. The particles start burning (after drying) and the temperature in the furnace suddenly increases from below $600K$ to above $1400K$ which causes a higher pressure and gas velocity within the furnace, and thus the pressure equation requires more iterations to converge. At the second point around 1125s of simulated time, the furnace reaches its steady state with its maximum temperature and now all the hot gases are burning in the furnace. This change in the simulation speedup highlights the need for advanced dynamic load-balancing techniques that can re-adapt the workload distribution when the conditions change. For coupled simulations, this cannot solely rely on metrics like the number of particles or CFD cells because they cannot express the changes in the relative load between the coupled modules. In this example, a load-balancing done at the beginning of the simulation would probably favor the good balancing of DEM as it appears to be the most computation intensive part. However very quickly, CFD represents the highest load and should be the main concern of the load-balancing strategy.

6 CONCLUSION

In this work, we successfully developed a complex industrial-level simulation of a biomass furnace and integrated it in the CloudiFacturing platform to be used by manufacturing SMEs and our partner Enerstena UAB. The workflow is fully automated, from the upload of the furnace settings until the download of a summary report. This approach offers to SMEs the benefits of HPC simulations and hides all the complexity of using an HPC platform in the back-end. For the SME manufactuers, it also reduces the need of HPC expertise and upfront investments for HPC hardware.

Under the hood, the biomass simulation is based on a CFD-DEM approach that provides detailed results with a high accuracy but requires an important computation time. The solver developed for this industrial case relies on an original co-located coupling strategy and takes advantage of the parallelization capabilities of FOAM-extend with MPI and XDEM with OpenMP. A thorough performance evaluation has been carried out and highlights the computing cost of CFD and DEM, independently and together. The simulation of a realistic setup requires around 34 hours on a single HPC computing node for 20 minutes of simulated time. This high computing load is due to the complexity of the model which considers many thermo-chemical processes and reactions and the need for a long simulation time to light-up the furnace and reach a steady-state.

Our implementation follows the specifications of our industrial partner and supports hundreds of parameters related to the design of the furnace. Some technical decisions (e.g., the *co-located* partitioner that imposes the co-location constraint) are conservative choices that guarantee the correctness in the wide range of parameter values, rather that optimizing the performance of a specific design. Our parallel code is able to take advantage of all the computing cores of a node using MPI and OpenMP, but due to the relatively small size of the problem (60k CFD cells and 9k particles), it will probably not scale very well on multiple computing nodes. A more detailed model (e.g., a finer CFD mesh) would improve the scalability and the accuracy of the results, but it would not reduce the (already high) total execution time, which is an important criteria for our industrial partner. To make this type of simulations practical for SMEs, one has to find the right trade-off between the complexity of the model and the time required to obtain the results.

With this work, we believe that we have now made the first steps for simulating complex industrial biomass furnace using the CFD-DEM approach. The careful performance analysis conducted in Section 5 gives detailed insights about the behavior of the coupled simulation and guides the way for future improvements. Our coupling between FOAM-extend and XDEM is based on the co-located strategy that avoids the inter-partition inter-physics communication, but it suffers from constraints imposed on the domain decomposition and causes imbalance. To overcome this limitation, we are adapting [1] our approach to rely on the preCICE coupling library [5], that will handle transparently and efficiently the mapping between meshes and the data exchanges. However this does not solve the problem of the load-balancing itself. Our results bring to light the need for advanced dynamic partitioning and load-balancing techniques for coupled multi-physics problems. These approaches should account for the computing load of each physics module, the coupling interactions between them and their dynamic behavior instead of treating them independently.

ACKNOWLEDGMENTS

The authors would like to thank Ralfas Lukoševičius, Linas Paukštaitis and Laimonas Narbutas from Enerstena UAB for their help in the description of the biomass furnace and our the partners of the CloudiFacturing consortium. This work received funding from the European Union's Horizon 2020 research and innovation program through the CloudiFacturing project (Grant No. 768892) and the IT4Innovations infrastructure was supported by the Ministry of Education, Youth and Sports of the Czech Republic through the e-INFRA CZ (ID:90140).

REFERENCES

[1] Xavier Besseron, Alban Rousset, Alice Peyraut, and Bernhard Peters. 2021. Eulerian-Lagrangian Momentum Coupling between XDEM and OpenFOAM Using preCICE. In *14th WCCM & ECCOMAS Congress 2020.*

[2] Christian Bruch, Bernhard Peters, and Thomas Nussbaumer. 2003. Modelling Wood Combustion under Fixed Bed Conditions. *Fuel* 82 (2003). https://doi.org/10.1016/S0016-2361(02)00296-X

[3] José María Cela, Philippe O. A. Navaux, Alvaro L. G. A. Coutinho, and Rafael Mayo-García. 2016. Fostering Collaboration in Energy Research and Technological Developments Applying New Exascale HPC Techniques. In *16th IEEE/ACM Int. Symp. on Cluster, Cloud and Grid Comp. (CCGrid).* https://doi.org/10.1109/CCGrid.2016.51

[4] Tao Chen, Xiaoke Ku, Jianzhong Lin, and Hanhui Jin. 2019. Modelling the Combustion of Thermally Thick Biomass Particles. *Powder Technology* 353 (2019). https://doi.org/10.1016/j.powtec.2019.05.011

[5] Gerasimos Chourdakis, Kyle Davis, Benjamin Rodenberg, Miriam Schulte, Frédéric Simonis, Benjamin Uekermann, Georg Abrams, Hans-Joachim Bungartz, Lucia Cheung Yau, Ishaan Desai, Konrad Eder, Richard Hertrich, Florian Lindner, Alexander Rusch, Dmytro Sashko, David Schneider, Amin Totounferoush, Dominik Volland, Peter Vollmer, and Oguz Ziya Koseomur. 2021. preCICE v2: A Sustainable and User-Friendly Coupling Library. *ArXiv210914470 Cs* (2021).

[6] Ayhan Demirbas. 2005. Potential Applications of Renewable Energy Sources, Biomass Combustion Problems in Boiler Power Systems and Combustion Related Environmental Issues. *Progress in Energy and Combustion Science* 31 (2005). https://doi.org/10.1016/j.pecs.2005.02.002

[7] Andrea Dernbecher, Alba Dieguez-Alonso, Andreas Ortwein, and Fouzi Tabet. 2019. Review on Modelling Approaches Based on Computational Fluid Dynamics for Biomass Combustion Systems. *Biomass Conv. Bioref.* 9 (2019). https://doi.org/10.1007/s13399-019-00370-z

[8] Chiara Galletti, Niccolò Bitossi, Serena Santilli, Maria Grazia Gallo, Franco Sansone, and Bernhard Peters. 2017. Modeling of Biomass Combustion on a Reciprocating-Grate Super-Heater. In *40th Meeting of the Italian Section of the Comb. Inst.* Italy. https://doi.org/10.4405/40proci2017.IX1

[9] Xi Gao, Jia Yu, Liqiang Lu, and William A. Rogers. 2021. Coupling Particle Scale Model and SuperDEM-CFD for Multiscale Simulation of Biomass Pyrolysis in a Packed Bed Pyrolyzer. *AIChE J.* 67 (2021). https://doi.org/10.1002/aic.17139

[10] Gabriel Fernando García Sánchez, Jorge Luis Chacón Velasco, Yesid Javier Rueda-Ordoñez, David Alfredo Fuentes Díaz, and Jairo René Martínez Morales. 2021. Solid Biomass Combustion Modeling: Bibliometric Analysis and Literature Review of the Latest Developments in OpenFOAM Based Simulations. *Bioresource Technology Reports* 15 (2021). https://doi.org/10.1016/j.biteb.2021.100781

[11] Mohammad Hosseini Rahdar, Fuzhan Nasiri, and Bruno Lee. 2019. A Review of Numerical Modeling and Experimental Analysis of Combustion in Moving Grate Biomass Combustors. *Energy Fuels* 33 (2019). https://doi.org/10.1021/acs.energyfuels.9b02073

[12] Shannon Keough. 2014. *Optimising the Parallelisation of OpenFOAM Simulations.* Technical Report. Maritime Division, Defence Science and Technology Organisation, Australia.

[13] Tamas Kiss. 2019. A Cloud/HPC Platform and Marketplace for Manufacturing SMEs. In *11th Int. Workshop on Sci. Gateways (IWSG)*. Slovenia.

[14] Mirjana Kljajić Borštnar and Tomi Ilijaš. 2019. Assessment of High Performance Computing Services Potential of SMEs. In *42nd Int. Conv. on Inf. and Comm. Tech., Electronics and Microelectronics (MIPRO).* https://doi.org/10.23919/MIPRO.2019.8756681

[15] Gregory M. Kurtzer, Vanessa Sochat, and Michael W. Bauer. 2017. Singularity: Scientific Containers for Mobility of Compute. *PLOS ONE* 12 (2017). https://doi.org/10.1371/journal.pone.0177459

[16] Christopher Lange, Patrick Barthelmäs, Tobias Rosnitschek, Stephan Tremmel, and Frank Rieg. 2021. Impact of HPC and Automated CFD Simulation Processes on Virtual Product Development—A Case Study. *Appl. Sci.* 11 (2021). https://doi.org/10.3390/app11146552

[17] Liqiang Lu, Xi Gao, Jean-François Dietiker, Mehrdad Shahnam, and William A. Rogers. 2022. MFiX Based Multi-Scale CFD Simulations of Biomass Fast Pyrolysis: A Review. *Chemical Engineering Science* 248 (2022). https://doi.org/10.1016/j.ces.2021.117131

[18] Amir Houshang Mahmoudi, Xavier Besseron, Florian Hoffmann, Miladin Markovic, and Bernhard Peters. 2016. Modeling of the Biomass Combustion on a Forward Acting Grate Using XDEM. *Chem. Eng. Sci.* 142 (2016). https://doi.org/10.1016/j.ces.2015.11.015

[19] Amir Houshang Mahmoudi, Florian Hoffmann, Bernhard Peters, and Xavier Besseron. 2016. Numerical Study of the Influence of Particle Size and Packing on Pyrolysis Products Using XDEM. *Int. Comm. in Heat and Mass Transfer* 71 (2016). https://doi.org/10.1016/j.icheatmasstransfer.2015.12.011

[20] Abdoul Wahid Mainassara Chekaraou, Alban Rousset, Xavier Besseron, Sébastien Varrette, and Bernhard Peters. 2018. Hybrid MPI+OpenMP Implementation of eXtended Discrete Element Method. In *30th Int. Symp. on Computer Architecture and High Performance Computing (SBAC-PAD 2018).* https://doi.org/10.1109/CAHPC.2018.8645880

[21] Mohammad Mohseni, Bernhard Peters, and Mehdi Baniasadi. 2017. Conversion Analysis of a Cylindrical Biomass Particle with a DEM-CFD Coupling Approach. *Case Studies in Therm. Eng.* 10 (2017). https://doi.org/10.1016/j.csite.2017.08.004

[22] Corinna Netzer, Tian Li, Lars Seidel, Fabian Mauß, and Terese Løvås. 2020. Stochastic Reactor-Based Fuel Bed Model for Grate Furnaces. *Energy Fuels* 34 (2020). https://doi.org/10.1021/acs.energyfuels.0c02868

[23] Nicole Paces and Martin Kozek. 2011. Modeling of a Grate-Firing Biomass Furnace for Real-Time Application. In *Int. Symp. on Models and Modeling Methodologies in Sci. and Eng. (MMMse'11)*. Orlando, FL, USA.

[24] Bernhard Peters. 2011. Validation of a Numerical Approach to Model Pyrolysis of Biomass and Assessment of Kinetic Data. *Fuel* 90 (2011). https://doi.org/10.1016/j.fuel.2011.02.003

[25] Bernhard Peters, Maryam Baniasadi, Mehdi Baniasadi, Xavier Besseron, Alvaro A. Estupinan Donoso, Mohammad Mohseni, and Gabriele Pozzetti. 2019. XDEM Multi-Physics and Multi-Scale Simulation Technology: Review of DEM–CFD Coupling, Methodology and Engineering Applications. *Particuology* 44 (2019). https://doi.org/10.1016/j.partic.2018.04.005

[26] Bernhard Peters, Alban Rousset, Xavier Besseron, Abdoul Wahid Mainassara Chekaraou, Maria Grazia Gallo, Franco Sansone, Alessio Lupi, and Chiara Galletti. 2020. Process Analysis in Thermal Process Engineering with High-Performance Computing Using the Example of Grate Firing. In *12th European Conf. on Industrial Furnaces and Boilers (INFUB-12)*. Porto, Portugal.

[27] Gabriele Pozzetti, Xavier Besseron, Alban Rousset, and Bernhard Peters. 2018. A Co-Located Partitions Strategy for Parallel CFD–DEM Couplings. *Adv. Powder Technol.* 29 (2018). https://doi.org/10.1016/j.apt.2018.08.025

[28] Frank Ulrich Rückert, Daniel Lehser-Pfeffermann, Danjana Theis, Ju Pyo Kim, Andre Schargen, Ingo Zorbach, and Jens Sohnemann. 2021. A New Simulation Model for Grate Firing Systems in OpenFOAM. *Energy* 216 (2021). https://doi.org/10.1016/j.energy.2020.119226

[29] Salar Tavakkol, Niklas Weber, Thorsten Zirwes, Jordan Denev, and Henning Bockhorn. 2019. Performance of Large Scale Eulerian-Lagrangian Numerical Simulation for Particulate Flow in Rotating Reactors. In *6th bwHPC Symp.* Karlsruhe, Germany.

[30] Simon J. E. Taylor, Tamas Kiss, Anastasia Anagnostou, Gabor Terstyanszky, Peter Kacsuk, Joris Costes, and Nicola Fantini. 2018. The CloudSME Simulation Platform and Its Applications: A Generic Multi-Cloud Platform for Developing and Executing Commercial Cloud-Based Simulations. *Fut. Gen. Comp. Sys.* 88 (2018). https://doi.org/10.1016/j.future.2018.06.006

[31] Christian Vecchiola, Suraj Pandey, and Rajkumar Buyya. 2009. High-Performance Cloud Computing: A View of Scientific Applications. In *10th Int. Symp. on Pervasive Systems, Algorithms, and Networks.* https://doi.org/10.1109/I-SPAN.2009.150

[32] Jingyuan Zhang, Tian Li, Henrik Ström, and Terese Løvås. 2020. Grid-Independent Eulerian-Lagrangian Approaches for Simulations of Solid Fuel Particle Combustion. *Chemical Engineering Journal* 387 (2020). https://doi.org/10.1016/j.cej.2019.123964

[33] Wenqi Zhong, Aibing Yu, Guanwen Zhou, Jun Xie, and Hao Zhang. 2016. CFD Simulation of Dense Particulate Reaction System: Approaches, Recent Advances and Applications. *Chemical Engineering Science* 140 (2016). https://doi.org/10.1016/j.ces.2015.09.035