POLITECNICO MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Development of a lagrangian wall film approach for the modelling of spray-wall kinematic and thermal interaction

Laurea Magistrale in Mechanical Engineering - Ingegneria Meccanica

Author: Angelo Chiappa

Advisor: Prof. Gianluca Montenegro

Academic year: 2024-2025

1. Introduction

In internal combustion engines and exhaust after-treatment devices such as Selective Catalytic Reduction (SCR) systems, liquid films forming on solid surfaces strongly influence evaporation, chemical conversion, pollutant formation, and overall system efficiency [3]. The ability to model the dynamics of these wall films with both fidelity and computational efficiency is therefore essential for the design and optimization of modern propulsion and emissions control technologies.

OpenFOAM provides a flexible, Open-Source framework for multiphase flow simulation and lagrangian spray modelling. However, the only existing wall film treatments in *OpenFOAM* is based on lagrangian eulerian approach, in which the film is represented on the wall surface using eulerian fields while droplets are tracked in a lagrangian manner. Although this strategy has proven effective in many contexts, it introduces non-negligible computational overhead and complexities.

This thesis presents the development of a fully lagrangian wall film model implemented within the *OpenFOAM* framework. The proposed model represents the wall film as a collection of lagrangian parcels constrained to move along solid boundaries. This formulation enables easier setup by the user, reduces computational cost and is more robust with respect to mesh quality. To achieve this result, specific sub-models had to be formulated to simulate momentum exchange, drag forces, parcel impingement and heat exchange. The implementation follows the object-oriented architecture of *OpenFOAM* to ensure modularity and extensibility, allowing straightforward integration with existing spray and combustion solvers.

The development takes inspiration from established methodologies employed in widely used engine simulation codes such as KIVA and CONVERGE, which have demonstrated the effectiveness of lagrangian based surface film modeling. Through a verification and validation study, this work tests the performance of the new model in a representative configuration and quantifies its advantages in terms of computational efficiency.

2. Methodology

2.1. Pach Interaction Model

As a starting point for the creation of the lagrangian wall film, the patch interaction model had to be modified. In *OpenFOAM* the patch interaction model is called when a lagrangian particle reaches a boundary patch and describes the interaction law with that patch. The first version of this custom patch interaction model was done by modifying LocalInteraction.C, available in *OpenFOAM* 8. This model requires a specific interaction type input for each patch and it is possible to choose among:

- stick: the parcel is stopped at the contact point.
- rebound: the normal velocity of the parcel (with respect to the patch plane), Un, is reflected.
- escape: the parcel is removed from the computational domain.

The goal was to create and add a new type of interaction called film, used to keep the particles on the boundary, but letting them free to move on the plane of the patch. To achieve this, it was necessary to create two new parcel variables, the first one being the particle index regime (which was set to film for particles inside the wall film, "p.pIRegime") and the boundary face on which the particle was ("p.faceWF").
The setup just explained is enough to handle kinematics, but the impingement of a droplet on a surface is a complex phenomenon which involves also thermodynamic aspects. A thermodynamic patch interaction model was created and instantiated on the ThermoCloud, in this way it could have access to temperature, viscosity and surface tension of the parcel. If the parcel is not a film parcel, the Bai and Gosman splash model is applied [2], as was already done in *OpenFOAM* for lagrangian-eulerian wall films.

2.2. Tracking algorithm

In [4], the motion of film parcels is handled by exploiting the logical coordinate system adopted in KIVA-3. As it is shown in Figure 1, when advancing in time, firstly a new provisional particle position $\widetilde{x}_p^{n+1}$ is computed by advancing the parcel in physical space: this movement is unconstrained. Then, the provisional point is mapped to the corresponding logical coordinates, $(\widetilde{\xi}, \widetilde{\eta}, \widetilde{\zeta})$: If all the values of the triplet are between 0 and 1, the parcel has remained in the same cell and the new position x_p^{n+1} is simply the projection of $\widetilde{x}_p^{n+1}$ on the face on which the parcel was at x_p^n. Else, according to the values of the triplet, four cases may arise:

1. the particle moves to the same logical face of a neighbouring cell (like in Figure 1),
2. it turns an inside corner and changes face of the same cell,
3. it turns an outside corner and is assigned to a different face of a neighbouring cell,
4. crosses an open boundary and escapes the system.

In the first three cases, the cell and face indices are updated accordingly, and the logical coordinates are adjusted to remain consistent with the new cell mapping. In the last case, the parcel is removed from the simulation. In *OpenFOAM*,

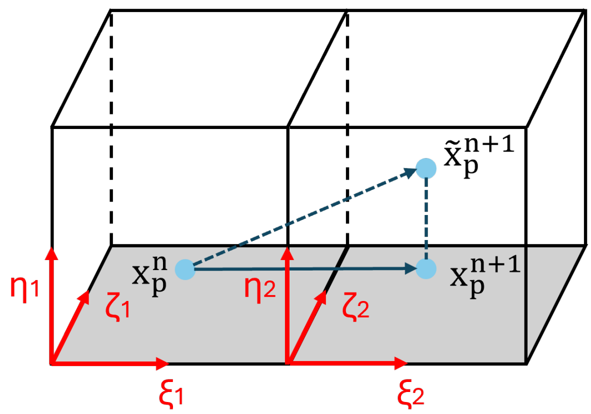

Figure 1: Representation of the projection algorithm implemented in [4].

the tracking algorithm of the particles works differently: it tracks particles on an implicit decomposition of each cell into tetrahedra as shown in Figure 2. This choice was done by the developers of the software in order to provide more robustness with respect to mesh quality. In this

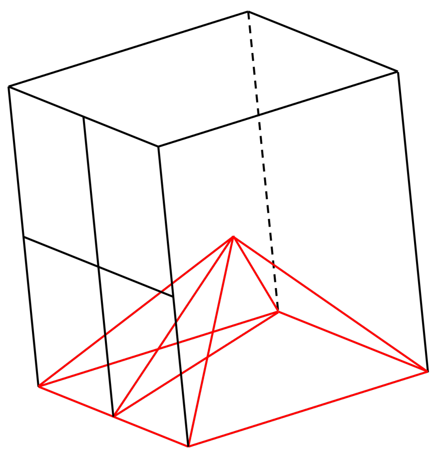

Figure 2: Tracking algorithm of *OpenFOAM*.

framework, trying to apply the "projection" algorithm of O'Rourke as it is, would raise portability issues for the library and generate a code very difficult to maintain. To overcome this, a "patch attraction" force has been created: in this way it was possible to obtain the same result by simply adding a new force model instead of acting inside the "basic" lagrangian class and the tracking algorithm.

2.3. Update Film

The main purpose of the function Update Film is to sum the mass of the parcels that are in film regime and to compute a representative tangential velocity on each boundary face. First, the function resets the film thickness and then loops over all parcels, identifying those whose interaction regime corresponds to film. For each of them, it updates the local film thickness contribution based on parcel volume and face area, while simultaneously accumulating the tangential momentum on that face. In a second stage, the function computes a mass-weighted mean tangential velocity for each face with film parcels on it. This averaged velocity is then imposed back onto all parcels residing on that face, ensuring consistency between parcel motion and the evolving wall-film layer. To perform this localized accumulation, the algorithm constructs a face-based occupancy structure that associates each boundary face with the parcels currently residing on it in film regime. Specifically, a list of dynamic pointer lists is created, where each outer list entry corresponds to a boundary face and stores the pointers to the parcels attached to that face. This data structure, referred to as filmOccupancy, is rebuilt at every time step. The adoption of this architecture provides several advantages. First, it enables direct face-wise access to parcels without requiring repeated global searches over the entire cloud, thereby reducing computational overhead. Second, by storing pointers rather than copies, the structure avoids memory duplication and guarantees that any modification applied to a parcel is immediately reflected in the cloud. Finally, reconstructing the structure at every time step ensures robustness in cases of parcel deletion, regime switching, or mesh updates.

2.4. Drag force

When inside the film, particles interact both with the carrier phase and the boundary, as shown in Figure 3. A film drag model has been implemented to account for the two contributions, making the hypothesis of linear velocity profile between the patch and the centroid of the cell, as shown in Figure 3.

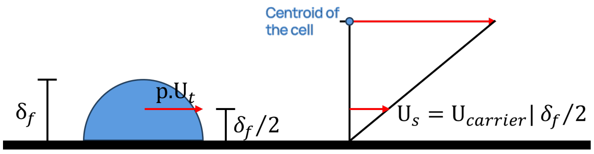

Figure 3: Drag force scheme.

2.5. Heat exchange

When a parcel enters the surface film, its heat exchange model must be modified both in terms of equations and in the definition of the effective heat-transfer area. In the free stream, the parcel exchanges heat only with the carrier phase and the characteristic surface area is evaluated from the parcel diameter. Once the parcel is deposited onto a wall film, however, it participates in a collective interaction over the wall face on which it resides. The relevant exchange area is therefore no longer the particle surface but a fraction of the wall-face area, proportional to the ratio between the parcel volume and the total volume of parcels present on that face. Within the film, the parcel simultaneously exchanges heat with the carrier phase through convection across the film interface and with the wall through conduction across the liquid layer. The resulting thermal balance becomes:

$$p.mass \cdot C_p \cdot \frac{dT}{dt} = h_g A_w \left(T_g - p.T\right) + \frac{k_l A_w}{\frac{\delta_f}{2}} \left(T_w - p.T\right) \quad (1)$$

where A_w denotes the effective wall-face area assigned to the parcel, $h_g = Nu \cdot \frac{K_g}{\frac{\delta_f}{2}}$ is the convective coefficient evaluated with half of the film half-thickness as characteristic length. The temperature evolution implemented in the function is written in a linear form that can be handled efficiently by the parcel temperature integrator. After dividing the energy balance by the parcel thermal inertia mC_p, the governing equation is rearranged as

$$\frac{dT}{dt} = a - b\,(p.T) \quad (2)$$

In this formulation, the coefficient b collects all contributions that are proportional to the parcel temperature $p.T$. For a parcel inside the wall film:

$$b = \frac{h_{tc} A_w}{m C_p} + \frac{k_l A_w}{(\delta_f/2)\, m C_p}. \tag{3}$$

The coefficient a gathers all terms that do not multiply the parcel temperature. These include the same heat–transfer mechanisms weighted by the external temperatures (carrier temperature T_c and wall temperature T_w), together with any additional source terms such as radiation or user-defined heating. In the film:

$$a = b_{\text{conv}} T_c + b_{\text{cond}} T_w + \frac{S_h + Q_{\text{rad}}}{m C_p}. \tag{4}$$

The temperature integrator, which is kept the same for both the models, advances the solution of the linear ordinary differential equation over the time step Δt.

$$\frac{dT}{dt} = a - bT \tag{5}$$

2.6. Film Cloud

All the models presented previously require the algorithm to determine whether a parcel is in the free stream or in the film regime. Performing this check repeatedly at every time step for every lagrangian element introduces a non-negligible overhead and can slow down the simulation. For this reason, a dedicated library has been developed specifically for the lagrangian wall–film treatment. The main feature of this library is the introduction of a `FilmCloud` and a corresponding `FilmParcel` class, implemented at the same abstraction level as the `SprayCloud` and `Parcel` classes. This separation allows each cloud to manage its own physics and avoids repeated regime checks within individual sub-models, as shown in Figure 4.

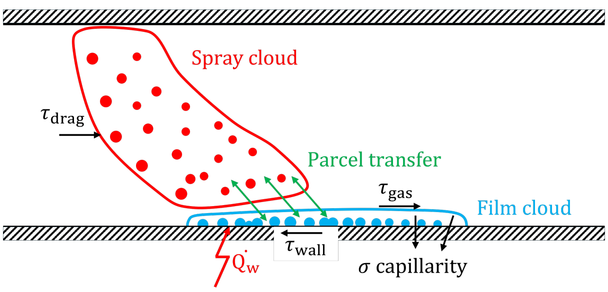

Figure 4: Film cloud.

Operationally, the solver, derived from a slightly modified version of `sprayFoam`, advances the solution as follows at each time step: first the `SprayCloud` is evolved, then the `FilmCloud`, and finally a transfer routine is invoked. This transfer function loops over all parcels and performs the following operations:

- if a spray parcel has `p.pIRegime == film`, it is transferred to the `FilmCloud`;
- if a film parcel has `p.pIRegime == freeStream`, it is transferred back to the `SprayCloud`.

This architecture eliminates the need to perform regime checks within each force model and patch-interaction model. Instead, model selection is handled independently for each cloud through their respective properties dictionaries, ensuring a clear separation at the configuration level. The Bai-Gosman patch interaction model described in subsection 2.1 has likewise been decomposed into two distinct implementations, one dedicated to the spray cloud and the other to the film cloud. Moreover, the introduction of the `FilmCloud` allows the film update routine to be encapsulated within that class. As a consequence, the update procedure operates exclusively on film parcels, avoiding unnecessary iterations over free-stream parcels.

2.7. Template specialization

The introduction of the second cloud enables a template specialization strategy.The basicFilmParcel type is templated on a FilmThermoParcel. This design allows the heat transfer model described in subsection 2.5 to be applied directly and exclusively to film parcels at compile time, while the base heat transfer model remains associated with free-stream parcels. Consequently, no runtime check on pIRegime() is required to determine which thermal model should be applied, improving both clarity and computational efficiency. Similarly, the Film Cloud is templated on a FilmKinematicCloud. This ensures that the construction of filmOccupancy is restricted to film parcels. As a result, the occupancy-building procedure can iterate directly over film parcels without repeatedly verifying their interaction regime before inserting them into the face-based pointer lists.

3. Model testing

As a first test, the model was used to simulate a spray inside a pipe. This configuration was specifically conceived for showing the capabilities of this model and does not rely on experimental validation; its purpose is to highlight the effects of heat transfer and gravity on the liquid film. Further details and images can be found in the thesis. A well known experiment, performed in [5] and numerically reproduced in [1] has been used to test the validity of the present model and to compare computational times of the present model and the lagrangian-eulerian model implemented in *OpenFOAM* 8. The domain is the same for both the cases and is represented in Figure 5. The k–ω *SST* turbulence

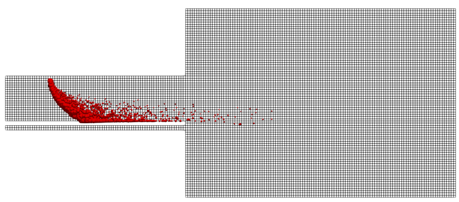

Figure 5: Domain dimensions.

model was selected due to its robustness with respect to near-wall treatment. To enable a consistent comparison between the two wall-film models, both simulations employed the same injection configuration, droplet breakup was disabled in all cases. Heat transfer was modelled only while parcels were in the free stream using the Ranz–Marshall correlation, whereas it was deactivated once the liquid entered the surface film. Both simulations were executed using a coupled solution approach in order to improve numerical accuracy. Figure 7 compare one of the experimental results with the corresponding numerical predictions from the presented model. In the referenced paper, the measurements were performed under steady-flow conditions. The numerical values were obtained by integrating the film thickness, (δ), over an area of (1mm^2) centred at the measurement location and dividing by the same area, thus yielding a local area-averaged thickness. Furthermore, the numerical data points were temporally averaged using a running-average window of (0.001s). The use of a running average was motivated by the in-

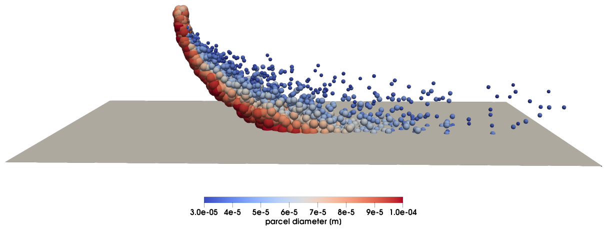

Figure 6: Snapshot taken at $t = 0,0008s$, the parcels are scaled and coloured by diameter.

herently unsteady nature of the simulation, for which reporting a single mean value would not adequately represent the local film dynamics. The time history starts from $t = 0s$, corresponding to the start of injection, and ends when the film thickness at that location decreases and the film effectively passes the probe position.

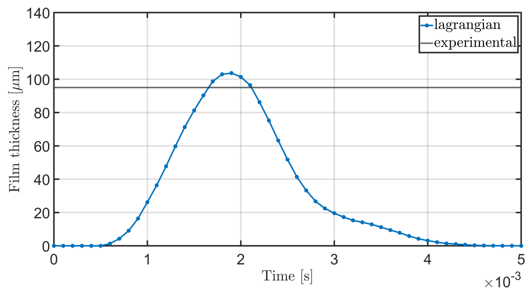

Figure 7: Comparison of the lagrangian model and the experimental result at $8mm$ from the jet nozzle.

In terms of computational time, the presented model is 33,5% faster with respect to the model present in *OpenFOAM*.

4. Conclusions

The motivation behind this work was the need for a fast and simple wall film model capable of simulating spray in Selective Catalytic Reduction (SCR) systems. SCR environments are characterized by high temperatures, high turbulence intensity, and high exhaust gas velocities. Accurately capturing wall film dynamics in such regimes is essential for predicting urea-water solution behaviour, deposition, evaporation, and overall system efficiency.
The development began with a detailed analysis of the wall film model already available in *OpenFOAM*. The existing implementation fol-

lows a lagrangian–eulerian formulation, where the spray is treated in a Lagrangian framework while the wall film is described as an eulerian thin film region coupled through an interface. Although physically comprehensive, this approach presents two major drawbacks:

1. High setup complexity, requiring careful configuration of the film region.
2. Limited mesh robustness, especially when using snappyHexMesh, which is not a boundary based meshing software by construction and may lead to inaccuracies in the thin-film region definition.

These limitations motivated the exploration of a fully Lagrangian wall film formulation as already adopted in established CFD solvers such as KIVA, CONVERGE, and ANSYS Fluent.

The core idea was to treat the wall film as a set of Lagrangian parcels constrained to move along the boundary surface. To achieve this, several key modelling components were developed:

- A new patch interaction model, initially designed to retain parcels on the wall after impact, mimicking the physical effect of capillarity. The model was subsequently extended into a more complete formulation capable of determining the impact outcome among deposition into the film, rebound or splash based on the local thermodynamic and Kinematic conditions of the droplet–wall interaction.
- An update film function designed to distribute the momentum inside the film parcels of the same face.
- A drag model, accounting for shear stress contributions from both the gas phase and the wall.
- A heat transfer model, incorporating conductive heat exchange with the solid wall.

This formulation completely removes the need for an interface between a lagrangian spray cloud and an eulerian wall film region. Initially, submodel activation was managed through conditional logic based on particle regime indices within the same cloud. However, this structure was later redesigned to improve clarity, modularity, and computational efficiency.

A dedicated filmCloud class was introduced, along with a cloud transfer function, allowing parcels to transition from a spray cloud to a film cloud. This architectural separation provided several advantages:

- Clear physical distinction between spray and film sub-models for the user.
- Removal of internal conditional branching based on particle indices.
- Possibility to specialize templates for each cloud type, resulting in improved computational performance.

The capabilities of the model were shown through a dedicated test case. Its response to external forces, such as gravity, was demonstrated, through correct motion of film parcels. The conductive heat transfer model showed the expected thermal behaviour. Furthermore, encouraging results were obtained from comparison with an established experimental benchmark case.

At the current stage, the model does not include the effect of the hydrostatic pressure gradient arising from spatial variations in film thickness between adjacent surface regions. Incorporating this effect in a physically consistent manner is particularly challenging within a discrete parcel framework. However, for SCR applications, this contribution is generally secondary compared to shear driven transport and thermal effects.

Future developments should focus on:

- Implementation of a curvature-induced separation (stripping) model to account for film breakup in the presence of a geometric discontinuity.
- Implementation of a more advanced capillarity formulation to better represent surface tension driven spreading.
- Exploration of Smoothed Particle Hydrodynamics (SPH) hybridization, where the filmCloud could be treated using a meshless approach. This would enable the evaluation of hydrostatic pressure gradients through kernel based interpolation, potentially overcoming current limitations related to thickness-driven flow redistribution.

In conclusion, this work demonstrates that a fully Lagrangian wall film model within *OpenFOAM* is both feasible and advantageous for SCR simulations. By eliminating the eulerian thin film region and its interface coupling, the proposed approach improves mesh robustness, reduces user setup complexity, and provides a flexible and extensible framework for future physical model enhancements.

References

[1] Giuliano Agati, Adriano Evangelisti, Serena Gabriele, Franco Rispoli, Paolo Venturini, and Domenico Borello. Liquid film formation: prediction accuracy of different numerical approaches. *Journal of Physics: Conference Series*, December 2022.

[2] Chengxin Bai and A. D. Gosman. Mathematical modelling of wall films formed by impinging sprays. *SAE Transactions*, 1996.

[3] Isabella Nova and Enrico Tronconi, editors. *Urea-SCR technology for deNOx after treatment of diesel exhausts*. Springer, 1 edition, March 2014.

[4] P.J. O'Rourke and A. A. Amsden. A particle numerical model for wall film dynamics in port-injected engines. In *SAE Technical Paper Series*, FFL. SAE International, October 1996.

[5] Timothy Shedd, May Corn, Jeffrey Cohen, Marco Arienti, and Marios Soteriou. Liquid film formation by an impinging jet in a high-velocity air stream. In *47th AIAA Aerospace Sciences Meeting*. American Institute of Aeronautics and Astronautics, January 2009.