

A Ghost Fluid/Level Set Method for boiling flows and liquid evaporation: Application to the Leidenfrost effect

Lucia Rueda Villegas, Romain Alis, Mathieu Lepilliez, Sébastien Tanguy*

IMFT CNRS UMR 5502, Université de Toulouse, France



ARTICLE INFO

Article history:

Received 26 June 2015

Received in revised form 2 March 2016

Accepted 16 April 2016

Available online 19 April 2016

Keywords:

Evaporation

Boiling

Level Set

Ghost Fluid

Leidenfrost

ABSTRACT

The development of numerical methods for the direct numerical simulation of two-phase flows with phase change, in the framework of interface capturing or interface tracking methods, is the main topic of this study. We propose a novel numerical method, which allows dealing with both evaporation and boiling at the interface between a liquid and a gas. Indeed, in some specific situations involving very heterogeneous thermodynamic conditions at the interface, the distinction between boiling and evaporation is not always possible. For instance, it can occur for a Leidenfrost droplet; a water drop levitating above a hot plate whose temperature is much higher than the boiling temperature. In this case, boiling occurs in the film of saturated vapor which is entrapped between the bottom of the drop and the plate, whereas the top of the water droplet evaporates in contact of ambient air. The situation can also be ambiguous for a superheated droplet or at the contact line between a liquid and a hot wall whose temperature is higher than the saturation temperature of the liquid. In these situations, the interface temperature can locally reach the saturation temperature (boiling point), for instance near a contact line, and be cooler in other places. Thus, boiling and evaporation can occur simultaneously on different regions of the same liquid interface or occur successively at different times of the history of an evaporating droplet. Standard numerical methods are not able to perform computations in these transient regimes, therefore, we propose in this paper a novel numerical method to achieve this challenging task. Finally, we present several accuracy validations against theoretical solutions and experimental results to strengthen the relevance of this new method.

© 2016 Elsevier Inc. All rights reserved.

1. Introduction

Boiling is the phase change of a pure liquid into a saturated vapor. It arises when the liquid is heated (or depressurized in the case of cavitation) beyond its boiling point. To start, it requires the activation of a nucleus, either in the bulk (homogeneous nucleation) or on a heated wall (heterogeneous nucleation). Many previous works on the direct numerical simulation of two-phase flows propose numerical methods which allow dealing with boiling flows [7,13,21,23,37,40,41,46,47].

Unlike boiling, evaporation does not require any nuclei activation to start. Indeed, it occurs spontaneously, whatever the pressure and the temperature conditions, at the interface between a liquid and a gas whose chemical compositions

* Corresponding author. Tel.: +33 5 34 32 28 08; fax: +33 5 34 32 28 99.

E-mail address: tanguy@imft.fr (S. Tanguy).

are different, such as the evaporation of liquid water in air, or the evaporation of fuel droplets in an ignitable mixture. A few previous works [6,15,18,26,28,39,45] propose numerical methods able to compute the evaporation of a liquid in the framework of the direct numerical simulation of two-phase flows with moving and deformable interfaces.

The direct numerical simulation of liquid–vapor phase change is a powerful and promising tool to perform accurate predictions and to improve the understanding of various physical phenomena which are involved in many industrial applications. For example, nucleate boiling is a topic of interest for the optimization of heat exchangers, since it is well-known that the heat transfer can be strongly favored by the nucleation of vapor bubbles on the wall of a heat pipe. On the other hand, if the wall heat flux reaches a critical value, the formation of a film of saturated vapor occurs and leads to a drastic decrease in the heat transfer coefficient. This phenomenon, known as the boiling crisis, is feared particularly in nuclear power plant. Other existing industrial applications, e.g. the fluid management of cryogenic liquids in microgravity conditions inside the tanks of space launchers, further motivate accurate studies of nucleate boiling.

The evaporation of droplets is also a topic of interest which is involved in many practical situations. For instance, it is an important step in the description of the combustion in automotive and aircraft engines. The prediction of this complex phenomenon requires an accurate description of the interaction between a cloud of moving and vaporizing droplets and flame fronts. As the dynamics of droplets can be strongly affected by thermal, chemical and collective effects, performing direct numerical simulations of such flows with a well-resolved description of the droplets would be a step forward in the description and the understanding of these flames. The evaporation of a droplet spray is also a significant phenomenon in the steel industry for cooling systems using liquid jet impingements. This latter involves the description of the interactions of vaporizing droplets with a very hot steel plate. This situation seems even more difficult than spray combustion. Indeed, many complex phenomena arise during the impact of a droplet on a hot wall. For example, in the so-called “Leidenfrost regime”, which occurs when the wall superheat is high, the formation of a thin layer of saturated vapor between the impinging droplet and the hot plate leads to the droplet levitation during its spreading; no contact line is formed during the impact. This regime is quite similar to the film boiling regime.

The main motivation of this paper is to present a novel numerical strategy to perform direct numerical simulations of the Leidenfrost droplets. Some previous works propose various strategies to achieve this difficult task. In [16,17,33], Volume-Of-Fluid simulations are carried out with a microscopic model for phase change. This microscopic model, which is based on the Schrage's law from the kinetic theory [2], allows deducing the local mass flow rate of phase changes from the discontinuities of local thermodynamics properties (temperature and pressure). Whereas it can provide acceptable results in some situations, many criticisms can be addressed to that kind of approaches. First, this microscopic model has been designed to describe the phase change in non-equilibrium thermodynamic conditions, thus the local mass flow rate of phase change will be different from zero if the interface temperature is different in the liquid and in the vapor. However, assuming the local thermodynamic equilibrium, the jump condition on the entropy points out that the temperature must be continuous at the interface [19]. Let us note that this last assertion is not strictly exact if the pressure is discontinuous at the interface, but the corrections resulting from the influence of the pressure jump condition on the temperature continuity are weak, see [21] for more details on this other topic.

Moreover, it appears that using the Schrage's law does not allow for accounting on the dependence of the local mass flow rate of phase change with the local temperature gradient at the interface. Instead, it depends on the temperature discontinuity and on an accommodation coefficient, whose value is not always perfectly known. To summarize, we think that using this law in numerical simulations, based on a local thermodynamic equilibrium assumption, leads to a disagreement with the first law of thermodynamic (mass flow rate of phase changes does not depend directly on the heat flux jump condition), and/or with the second law of the thermodynamic (the temperature is not considered continuous at the interface in contradiction with the jump condition on the entropy). Moreover, since the local mass flow rate of phase changes can be deduced from the jump condition on the thermal flux if the temperature is continuous at the interface, considering an additional law which involves an accommodation coefficient leads to an over-determined problem. Nevertheless, the Schrage's law is still frequently used in various contexts, since it allows determining much more easily a mass flow rate of phase change than by the classical approach based on the assumption of local thermodynamic equilibrium. Indeed, in the latter case the thermal gradient at the interface must be accurately computed which is a much more demanding task.

In [9,10], another approach has been developed to perform numerical simulations of Leidenfrost droplets. In these papers, a Level Set Method, used to describe the droplet motion in an isothermal domain, is coupled to a lubrication model to account to thermal transfer in the vicinity of the hot wall through an ALE auxiliary grid. Although many interesting results are provided in this study, this technique does not address the overall problem, since the thermal effects and the phase change are only solved in the vicinity of the hot wall. Moreover, a lubrication model in the thin vapor layer is not able to fully describe the complex and unsteady fluid dynamic in this microscopic region, which is strongly coupled to the interface motion.

Therefore, we present in this paper some new developments in order to perform simulations of Leidenfrost droplets with fully-resolved computations of the vapor layer dynamic, of the heat transfer and of the phase changes. The key-points to achieve these difficult simulations are the development of a phase change model which can deal both with evaporation and with boiling, as well as the use of very refined grids close to the hot wall. Such refined grids must be used with implicit algorithms for the temporal discretization of all the diffusion terms. Moreover, we will see that a semi-implicit temporal solution of the extrapolation techniques based on a PDE approach [1,12] can also be required to alleviate the number of temporal iterations of these extrapolations when the interface crosses grid cells with very different resolutions.

We will also present, in the numerical results section, several benchmarks in order to assess the accuracy of the numerical methods described in this paper. To our knowledge, these benchmarks have never been considered with an interface capturing method coupled to a numerical solver dedicated to droplet evaporation.

2. Formalism

In this section, we first present the conditions of local thermodynamic equilibrium. Next, the conservation equations for two-phase flows are presented. As we will use sharp interface methods for the computation of all singular terms, the conservation equations will be expressed with the “jump condition formulation” [13,38,46].

2.1. Thermodynamic local equilibrium conditions for a pure liquid in contact with a gas mixture

We consider an interface between a pure liquid and a gas mixture which is composed of the vapor of the liquid component (species 1) and of another component (species 2), for example steam water in nitrogen. We assume that the two gaseous components behave as an ideal gas mixture. Therefore, the total pressure P_0 of this gas mixture can be expressed as the sum of the partial pressures of the two components

$$P_0 = P_1 + P_2, \quad (1)$$

where P_1 and P_2 are respectively the partial pressures of the chemical species 1 and 2, which can be defined as

$$P_1\mathcal{V} = n_1RT, \quad (2)$$

$$P_2\mathcal{V} = n_2RT, \quad (3)$$

where \mathcal{V} and T are respectively the volume and the temperature of the gas mixture, R is the ideal gas constant ($R = 8.31 \text{ Jmol}^{-1} \text{ K}^{-1}$) and n is the number of moles of the chemical species. We now define Y_1 to be the mass fraction of the chemical species 1

$$Y_1 = \frac{n_1M_1}{n_1M_1 + n_2M_2}, \quad (4)$$

with M_1 and M_2 the molar masses of species 1 and 2, respectively. From these relations, we can express the mass fraction of chemical species 1 with the total pressure of the gas mixture and the partial pressure of this chemical species:

$$Y_1 = \frac{P_1M_1}{P_1M_1 + (P_0 - P_1)M_2}. \quad (5)$$

We consider now an interface Γ between a liquid and a gas. By writing the Clausius–Clapeyron law, we can determine the saturated vapor pressure of the liquid component at the interface, which provides the partial pressure of this chemical species at the interface:

$$P_1|_{\Gamma} = P_0e^{-\frac{L_{vap}M_1}{R}\left(\frac{1}{T|_{\Gamma}} - \frac{1}{T_{sat}}\right)}. \quad (6)$$

In this equation, L_{vap} is the mass latent heat of vaporization of chemical species 1, $T|_{\Gamma}$ is the interface temperature and T_{sat} is the saturation temperature (or boiling temperature) of chemical species 1. L_{vap} and T_{sat} depend on the external pressure which will be assumed constant in this study (isobaric phenomena). When the local thermodynamic equilibrium is respected, the inequality on the entropy jump condition resulting from the second law of thermodynamics implies that the temperature is continuous across the interface, provided the pressure is continuous at the interface. Let us notice that various effects can involve weak jump conditions in the interface temperature related to the pressure jump condition which can be induced by capillary, viscous or phase change effects. In [21], the authors present a detailed study to assess the corrections due to these different effects. As we have checked that the contribution of these different effects was negligible in the situations which are presented in this work, we will not consider any jump condition in the interface temperature in the present study. Therefore, we will consider in the rest of the paper that:

$$[T]_{\Gamma} = 0, \quad (7)$$

with $[.]_{\Gamma}$ denoting the jump condition operator defined by

$$[f]_{\Gamma} = f_{liq} - f_{gaz}. \quad (8)$$

2.2. Mass conservation

Strong thermal gradients and mixing of different chemical species can involve density variations in the gas phase. However, in the interest of simplification, we will restrict our study to incompressible flows with a constant density. Therefore, we will not need any equation of state for the gas phase or for the liquid phase. The mass conservation will be satisfied by imposing a divergence-free condition in the velocity field in each phase

$$\nabla \cdot \vec{V} = 0 \quad (9)$$

and the appropriate jump condition in the velocity field at the interface will be imposed by taking into account the phase change [32]:

$$[\vec{V}]_\Gamma = -\dot{m} \left[\frac{1}{\rho} \right]_\Gamma \vec{n}, \quad (10)$$

with \dot{m} the local mass flow rate of phase change, ρ the density, and \vec{n} the normal vector pointing to the liquid phase.

2.3. Momentum conservation

The momentum conservation is expressed with the Navier–Stokes equations for an incompressible flow:

$$\rho \frac{D \vec{V}}{Dt} = -\nabla p + \nabla \cdot (2\mu \mathbf{D}) + \rho \vec{g}, \quad (11)$$

where p is the pressure, μ the dynamical viscosity, \mathbf{D} the tensor of deformation, and \vec{g} the acceleration due to gravity. An appropriate jump condition in pressure must also be satisfied at the interface to account for capillary, viscous and phase change effects [13]:

$$[p]_\Gamma = \sigma \kappa + 2 \left[\mu \frac{\partial V_n}{\partial n} \right]_\Gamma - \dot{m}^2 \left[\frac{1}{\rho} \right]_\Gamma, \quad (12)$$

where σ is the surface tension coefficient, and κ is the interface curvature.

2.4. Energy conservation

Several formulations can be used for the energy conservation equation. Indeed, the first law of thermodynamics can be expressed with the total energy, the internal energy or the enthalpy. As only quasi-isobaric phenomena are considered in this paper, we will use a formulation based on the enthalpy to express the energy conservation. The contribution of the viscous dissipation on the energy balance will be neglected in this paper, therefore the following simplified relation can be used to compute the temperature evolution in each phase:

$$\rho C_p \frac{DT}{Dt} = \nabla \cdot (k \nabla T), \quad (13)$$

where T is the temperature, C_p is the specific heat at constant pressure, and k is the thermal conductivity. A jump condition in the thermal flux at the interface can be deduced from the first law of thermodynamics:

$$[k \nabla T \cdot \vec{n}]_\Gamma = \dot{m} (L_{vap} + (C_{pliq} - C_{pvap})(T_{sat} - T|_\Gamma)). \quad (14)$$

2.5. Chemical species conservation

The conservation of chemical species 1 can be expressed with the following convection–diffusion equation:

$$\rho \frac{DY_1}{D} t = \nabla \cdot (\rho D_m \nabla Y_1), \quad (15)$$

where D_m is the diffusion coefficient of the species 1 in the gas phase. The following jump condition must also be satisfied at the interface:

$$[\rho D_m \nabla Y_1 \cdot \vec{n}]_\Gamma = -\dot{m} [Y_1]_\Gamma. \quad (16)$$

For a mono-component liquid, this jump condition can be expressed as a Robin boundary condition at the interface:

$$\dot{m} Y_1|_\Gamma + \rho_g D_m \nabla Y_1 \cdot \vec{n}|_\Gamma = \dot{m}. \quad (17)$$

3. Solving the incompressible Navier–Stokes equations for two-phase flows with a prescribed velocity jump condition at the interface

In this section, we provide details on the implementation of the Navier–Stokes solver for incompressible flows with a prescribed velocity jump condition at a moving and deformable interface. The method used in this paper has been developed by Nguyen et al. [32] to describe the propagation of flame front. This method has also been used for the direct numerical simulation of boiling flows in [13,46]. In [45], the authors point out that it is not perfectly suited for the simulation of evaporating droplets. To overcome this issue they propose an improvement on the velocity field extension to insure the liquid velocity extension satisfies the divergence-free property at the interface. Since the simulations which are presented in this paper involve droplet evaporation, this divergence-free extrapolation will be used in all the computations presented in this paper.

3.1. On the computation of viscous terms with Level Set/Ghost Fluid Method

The computation of the viscous terms at the interface is another issue to be addressed for simulations with phase change in the framework of the Ghost Fluid Method. Indeed, in the first work of [22] on the Ghost Fluid Method for incompressible two-phase flows, the authors have developed a specific treatment of the viscous terms at the interface which is based on the jump condition of the viscous stresses. However these jump conditions cannot be applied to simulations with phase change because their determination is not possible if the velocity field at the interface is not continuous.

Let us notice that in the case of the simulations of boiling flows in [13], the authors propose to compute the viscous terms at the interface by using the smoothed delta function [43], whereas the overall algorithm for phase change is based on a sharp interface formulation. Also in the framework of boiling flows, the authors in [46] use a different Ghost Fluid approach to compute viscous terms. This method, which has been firstly presented in [44], does not require that the velocity field is continuous at the interface.

In fact, this Ghost Fluid approach is quite similar to the original Delta Function Method to compute the viscous terms, except that the sharpness of the method is preserved by using an interpolated viscosity on the border of the cells which are crossed by the interface. The interpolation of the viscosity consists in a harmonic average typical of the Ghost Fluid Method for incompressible flows [22,24,27].

In a recent study [25], an overview on the computation of viscous terms with Level Set Methods is presented. In this paper, it is demonstrated that the Delta Function Method [43], the Ghost Fluid Method of [22] (named GFPM for Ghost Fluid Primitive viscous Method), and the Ghost Fluid Method of [44] (named GFCM for Ghost Fluid Conservative viscous Method) are formally equivalent. Moreover, due to several test-cases it is shown that the two Ghost Fluid Methods have a comparable accuracy and that these two methods outclass the Delta Function Method. A new method is also proposed to deal with viscous terms in the framework of the Ghost Fluid Method.

The main interest of this new method (named GFSCM for Ghost Fluid Semi-Conservative viscous Method) lies in its ability to deal easily both with phase changes (as the GFCM) and with an implicit treatment of the viscous terms. Indeed, the direct numerical simulation of vaporizing droplets in the Leidenfrost regime requires an implicit temporal discretization of viscous terms due to the refined grid close to the hot wall.

We will present in what follows the adaptation of the original method of [32] to an implicit temporal discretization of the GFSCM with a jump condition on the velocity field.

3.2. Implicit temporal discretization of the Ghost Fluid Semi-Conservative viscous Method with a jump condition in the velocity field

As it is presented in [25], the following implicit temporal discretization of the viscous terms can be developed when the Ghost Fluid Semi-Conservative viscous Method is used with a projection method for solving incompressible flows

$$\rho^{n+1} \vec{V}^* - \Delta t \nabla \cdot (\mu \nabla \vec{V}^*) = \rho^{n+1} (\vec{V}^n - \Delta t (\vec{A}(\vec{V}^n) - \vec{g})) \quad (18)$$

with the following definition of $\vec{A}(\vec{V}^n)$ to account for convective terms if phase change occurs

$$\vec{A}(\vec{V}^n) = \vec{V}_l^n \cdot \nabla \vec{V}_l^n \quad \text{if } \phi > 0 \quad (19)$$

$$\vec{A}(\vec{V}^n) = \vec{V}_g^n \cdot \nabla \vec{V}_g^n \quad \text{if } \phi < 0. \quad (20)$$

Let us notice that this temporal discretization, Eq. (18), is like a Helmholtz equation. Its main advantage lies in its simplicity, since it allows to compute separately each component of the velocity field. This would not otherwise be possible by using the GFCM or the Delta Function Method. Indeed, a fully implicit treatment of viscous terms with the GFCM or the Delta Function Method leads to a large and complex linear system since all the components of the velocity field must be computed simultaneously.

Finally the projection of Eq. (18) on the x -axis and the y -axis leads respectively to the following relation on a staggered grid:

$$\rho_{i+1/2j}^{n+1} u_{i+1/2j}^* - \Delta t (\nabla \cdot (\mu \nabla u^*))|_{i+1/2j} = \rho_{i+1/2j}^{n+1} (u_{i+1/2j}^n - \Delta t ((\vec{A}(\vec{V}^n) \cdot \vec{e}_x)|_{i+1/2j} - \vec{g})), \quad (21)$$

$$\rho_{ij+1/2}^{n+1} v_{ij+1/2}^* - \Delta t (\nabla \cdot (\mu \nabla v^*))|_{ij+1/2} = \rho_{ij+1/2}^{n+1} (v_{ij+1/2}^n - \Delta t ((\vec{A}(\vec{V}^n) \cdot \vec{e}_y)|_{ij+1/2} - \vec{g})). \quad (22)$$

The following scalar equations, Eq. (21) and Eq. (22), must be respectively solved with the two following scalar jump conditions to respect the mass conservation across a reactive interface:

$$[u_{i+1/2j}^*] = -\dot{m}_{i+1/2j} \left[\frac{1}{\rho} \right]_\Gamma n_{i+1/2j}^x, \quad (23)$$

$$[v_{ij+1/2}^*] = -\dot{m}_{ij+1/2} \left[\frac{1}{\rho} \right]_\Gamma n_{ij+1/2}^y. \quad (24)$$

That can be done by using the spatial discretization proposed in [27] to impose jump conditions when solving a Poisson equation. The next step consists in computing an extension of the velocity field in each side of the interface:

$$\begin{cases} \vec{V}_l^* = \vec{V}^* & \text{if } \phi > 0 \\ \vec{V}_l^{*ghost} = \vec{V}^* - \dot{m} \left[\frac{1}{\rho} \right]_{\Gamma} \vec{n} & \text{if } \phi < 0 \end{cases} \quad (25)$$

$$\begin{cases} \vec{V}_g^* = \vec{V}^* & \text{if } \phi < 0 \\ \vec{V}_g^{*ghost} = \vec{V}^* + \dot{m} \left[\frac{1}{\rho} \right]_{\Gamma} \vec{n} & \text{if } \phi > 0 \end{cases} \quad (26)$$

Next, this extension of the velocity field will be used to compute the right hand side of the Poisson equation for the pressure:

$$\nabla \cdot \left(\frac{\nabla P^{n+1}}{\rho^{n+1}} \right) = f, \quad (27)$$

where the right hand side f is computed as follows:

$$\begin{cases} f = \frac{\nabla \cdot \vec{V}_l^*}{\Delta t} & \text{if } \phi > 0 \\ f = \frac{\nabla \cdot \vec{V}_g^*}{\Delta t} & \text{if } \phi < 0 \end{cases} \quad (28)$$

and with the following pressure jump condition which is imposed with the numerical methods proposed in [22,27]:

$$[P]_{\Gamma} = \sigma \kappa + \left[\mu \frac{\partial V_n}{\partial n} \right]_{\Gamma} - \dot{m}^2 \left[\frac{1}{\rho} \right]_{\Gamma}. \quad (29)$$

The final correction step can next be applied:

$$\vec{V}^{n+1} = \vec{V}^* - \frac{\Delta t}{\rho^{n+1}} \left(\nabla P^{n+1} - \left(\sigma \kappa + \left[\mu \frac{\partial V_n}{\partial n} \right]_{\Gamma} - \dot{m}^2 \left[\frac{1}{\rho} \right]_{\Gamma} \right) \vec{n} \delta_{\Gamma} \right). \quad (30)$$

A sharp approximation of the Dirac distribution δ_{Γ} , based on the work of [27], must be used to compute this correction step, see [25] for more details on this last point.

3.3. Divergence-free extrapolation of velocity field for evaporating droplets

Finally to restart the next time step, a velocity field extrapolation of the corrected velocity field is required, both to evolve the interface when the Level Set equation is solved, and also to compute $\vec{A}(\vec{V}^n)$. In [45], the authors have developed a specific extension of the liquid velocity field which preserves the divergence-free property at the interface. It consists in solving an additional Poisson equation for a ghost pressure P^{ghost} :

$$\nabla \cdot \left(\frac{\nabla P^{ghost}}{\rho^{n+1}} \right) = \frac{\nabla \cdot \vec{V}_l^*}{\Delta t}. \quad (31)$$

This ghost pressure enables the definition of an extrapolated velocity field for the liquid velocity field

$$\begin{cases} \vec{V}_l^{n+1} = \vec{V}^{n+1} & \text{if } \phi > 0 \\ \vec{V}_l^{ghost} = \vec{V}_l^* - \Delta t \frac{\nabla P^{ghost}}{\rho^{n+1}} & \text{if } \phi < 0 \end{cases} \quad (32)$$

whereas the extension of the gas velocity can be defined with Eq. (26). This divergence-free extrapolation of the velocity field leads to significant improvements on the mass prediction of evaporating droplets [45] and slight improvements on the mass prediction in the context of boiling flows [46].

4. Phase change model

In the previous section of this paper, the numerical methods involved to impose a jump condition on the velocity field have been detailed in the framework of the Ghost Fluid Method, assuming that the mass flow rate of phase change was known. In this section, we will present the different models that can be used to determine the mass flow rate of phase change.

4.1. Ghost Fluid Thermal Solver for Boiling (GFTSB)

We present here the numerical model which is proposed in [13] to deal with boiling flows. This sharp interface model will be referred in the rest of this paper as the Ghost Fluid Thermal Solver for Boiling (GFTSB). Considering that the interface temperature is continuous and uniform, the authors propose to solve separately the temperature in each phase by imposing a Dirichlet Boundary condition at the interface [11]. It consists in first solving the following equation in the liquid domain:

$$\rho_l C p_l T_l^{n+1} - \Delta t \nabla \cdot (k_l \nabla T_l^{n+1}) = \rho_l C p_l (T_l^n - \Delta t \vec{V}_l^n \cdot \nabla T_l^n) \quad \text{if } \phi > 0 \quad (33)$$

$$T|_{\Gamma} = T_{sat} \quad (34)$$

and the corresponding equation in the gas phase:

$$\rho_g C p_g T_g^{n+1} - \Delta t \nabla \cdot (k_g \nabla T_g) = \rho_g C p_g (T_g^n - \Delta t \vec{V}_g^n \cdot \nabla T_g^n) \quad \text{if } \phi < 0 \quad (35)$$

$$T|_{\Gamma} = T_{sat} \quad (36)$$

next with the jump condition Eq. (14), the boiling mass flow rate can be determined:

$$\dot{m} = \frac{[k \nabla T \cdot \vec{n}]_{\Gamma}}{L_{vap}}. \quad (37)$$

4.2. Ghost Fluid Thermal Solver Method for Evaporation (GFTSE)

We present now the phase change model proposed in [45] to deal with evaporating droplets in the framework of Level Set/Ghost Fluid Method. It will be referred to in the rest of this paper as the Ghost Fluid Thermal Solver for Evaporation (GFTSE). Let us notice that similar models have been developed next in [39] and [28] with Volume Of Fluid methods. As the evaporation of droplets strongly depends on the chemical composition of the gas mixture, an additional equation for the conservation of chemical species must be solved (for a binary mixture in the gas phase). The equilibrium thermodynamics conditions at the interface Eq. (5) and Eq. (6) define the mass fraction of evaporated species at the interface.

Therefore, the conservation equation of chemical species can be solved in the gas field with an imposed Dirichlet boundary condition at the interface

$$\rho_g Y_1^{n+1} - \Delta t \nabla \cdot (\rho_g D_m \nabla Y_1^{n+1}) = \rho_g (Y_1^n - \Delta t \vec{V}_g^n \cdot \nabla Y_1^n) \quad \text{if } \phi < 0, \quad (38)$$

$$Y_1|_{\Gamma} M_1 = \frac{P_1|_{\Gamma} M_1}{P_1|_{\Gamma} M_1 + (P_0 - P_1|_{\Gamma}) M_2}, \quad (39)$$

where $P_1|_{\Gamma}$ is the partial pressure of saturated vapor of species 1 which can be determined with the Clausius–Clapeyron law:

$$P_1|_{\Gamma} = P_0 e^{-\frac{L_{vap} M_1}{R} (\frac{1}{T|_{\Gamma}} - \frac{1}{T_{sat}})}. \quad (40)$$

Next, once the mass fraction field has been determined, the mass flow rate of phase change can be computed through the jump condition for the conservation of chemical species Eq. (16)

$$\dot{m} = -\frac{[\rho D_m \nabla Y_1 \cdot \vec{n}]_{\Gamma}}{[Y_1]_{\Gamma}}, \quad (41)$$

which can be simplified for a mono-component liquid which evaporates in contact with a binary mixture in the gas phase

$$\dot{m} = \frac{\rho_g D_m \nabla Y_1 \cdot \vec{n}|_{\Gamma}}{1 - Y_1|_{\Gamma}}. \quad (42)$$

Once \dot{m} has been determined, the energy conservation equation can be solved simultaneously in the two domains with the appropriate jump condition in the thermal flux

$$\begin{cases} \rho C p T^{n+1} - \Delta t \nabla \cdot (k \nabla T^{n+1}) = \rho C p (T^n - \Delta t B(\vec{V}^n, T^n)), \\ [k \nabla T \cdot \vec{n}]_{\Gamma} = \dot{m} (L_{vap} + (C_{pliq} - C_{pvap})(T_{sat} - T|_{\Gamma})), \end{cases} \quad (43)$$

with

$$\begin{cases} B(\vec{V}^n, T^n) = \vec{V}_l^n \cdot \nabla T_l^n & \text{if } \phi > 0 \\ B(\vec{V}^n, T^n) = \vec{V}_g^n \cdot \nabla T_g^n & \text{if } \phi < 0 \end{cases}. \quad (44)$$

4.3. Ghost Fluid Thermal Solver for Boiling and Evaporation (GFTSBE)

We describe here a novel numerical method which allows the treatment of both evaporation and boiling. Indeed, very heterogeneous conditions of vaporization occur during the interaction of vaporizing droplets with hot plates (above the boiling temperature) whether for levitating droplets in the Leidenfrost regime or for sessile droplets with a contact line. This new method captures implicitly the spatial transition between regions where evaporation occurs and regions where boiling occurs. It will be referred to in the rest of this paper as the Ghost Fluid Thermal Solver for Boiling and Evaporation (GFTSBE). The main idea consists of solving the thermal field in a similar way as in the GFTSB method. However, instead of a uniform Dirichlet boundary condition on the interface temperature, a non-homogeneous Dirichlet boundary condition is imposed. This boundary condition is determined by reversing Eq. (5) and Eq. (6), to define an interface temperature depending on the local mass fraction in the gas phase:

$$T|_{\Gamma} = \frac{L_{vap} M_1 T_{sat}}{L_{vap} M_1 - RT_{sat} \ln\left(\frac{P_1|_{\Gamma}}{P_0}\right)}, \quad (45)$$

with $P_1|_{\Gamma}$ determined from Eq. (5):

$$P_1|_{\Gamma} = \frac{-Y_1|_{\Gamma} P_0 M_2}{(M_1 - M_2) Y_1|_{\Gamma} - M_1}. \quad (46)$$

Let us remark that the temperature varies along the interface but remains continuous across the interface as stated by the second law of thermodynamics. Next, once the temperature field has been determined separately in each domain by using this non-homogeneous Dirichlet boundary condition, the mass flow rate of phase change can be computed in the same way as for the GFTSB with the Eq. (37). It is noteworthy that if the mass fraction is equal to 1 in the gas phase, this thermal solver becomes strictly identical as the one in the GFTSB since $T|_{\Gamma} = T_{sat}$ if $Y_1|_{\Gamma} = 1$.

As the mass fraction is required to determine the interface temperature, the next step consists of solving the equation of the mass fraction, Eq. (15), with the suitable jump condition, Eq. (16). The mass flow rate of phase change is known from Eq. (14) and the equations of thermodynamics equilibrium at the interface have already been used with the thermal solver. Then the mass fraction field can be computed by solving this equation with an imposed Robin boundary condition at the interface:

$$\rho_g Y_1^{n+1} - \Delta t \nabla \cdot (\rho_g D_m \nabla Y_1^{n+1}) = \rho_g (Y_1^n - \Delta t \vec{V}_g^n \cdot \nabla Y_1^n) \quad \text{if } \phi < 0, \quad (47)$$

$$\dot{m} Y_1|_{\Gamma} + \rho_g D_m \nabla Y_1 \cdot \vec{n}|_{\Gamma} = \dot{m}. \quad (48)$$

With this new formulation, we solve exactly the same set of equations as for the GFTSE, but the overall algorithm has been reversed. It avoids using Eq. (42) which can lead to indeterminacy when the gas phase is vapor saturated ($Y_1|_{\Gamma} = 1$). Indeed, in this critical case Eq. (42) diverges due to the division by 0. Unlike the GFTSE, this new method is stable even if the gas phase is saturated of vapor since in this situation $Y_1|_{\Gamma} = 1$ and $\nabla Y_1|_{\Gamma} = 0$. In this case, the Robin boundary condition is automatically satisfied since both sides of the equation are equal to \dot{m} . This method does not present any singularities if the interface temperature is equal to the saturation temperature, and we will see in the results section that it is perfectly suited for simulations involving heterogeneous regimes of vaporization with transient regions between evaporation and boiling.

5. Spatial discretization and details on the numerical implementation of the overall model

Usual spatial numerical schemes are used to solve the PDE far from the interface (fifth order WENO scheme [20] for the convective derivatives and second order finite volume for the diffusive derivatives). The temporal integration is carried out with a second order Runge–Kutta scheme. Classical time step constraints, in relation with convective terms, surface tension terms and viscous terms (except if an implicit temporal discretization is used for the latter), must be satisfied to ensure the temporal stability. The Black-Box MultiGrid [4] method has been implemented and used to speed up most simulations (compared to a ICCG solver) due to its great efficiency to solve linear system involving heterogeneous physical properties [29]. However, this solver is not efficient in the case of the Leidenfrost droplet where a very refined grid is used close to the hot wall.

5.1. Second order spatial discretization of a boundary condition on an immersed interface

In [11], the authors propose a rather simple discretization to impose Dirichlet boundary conditions on an immersed interface when a Poisson or a Helmholtz equation is considered. This method has many attractive features since it is second order in space and the matrix discretization remains symmetric positive-definite and simple (5 diagonals in 2D and 7 diagonals in 3D). This second order discretization is also used in the GFTSB [13] to compute the temperature field and in the GFTSE [45] to compute the mass fraction field. In this work, the method is used in a similar way as for the GFTSB, except that the interface temperature is not homogeneous and depends on the local mass fraction.

Moreover, as it has been previously explained, we need to solve the conservation equation of chemical species with an imposed Robin boundary condition at the interface. In [35], the authors have presented an efficient spatial discretization to

solve a Poisson or a Helmholtz equation with a Neumann or a Robin boundary condition at the interface. This new method, which is based on a finite volume formulation, also preserves the matrix symmetry and a simple stencil of discretization with 5 points in 2D and 7 points in 3D. A sub-cell interface reconstruction [30,31] is required to compute the interface geometrical properties for the computational cells which are crossed by the interface. For the purposes of this study, this method has also been implemented to solve Eq. (47) with the immersed Robin boundary condition, Eq. (48).

5.2. Temperature and mass fraction extension based on PDE resolution

Some discontinuous variables must be extrapolated across each side of the interface in order to populate ghost cells. This extrapolation can be done by solving iteratively some partial differential equations which propagate the information that must be extrapolated along the normal direction at the interface. More details on the extrapolation method and its implementation are provided in [1]. Let us notice that the influence of this extrapolation on the global accuracy of theoretical problems has been extensively and successfully assessed in [12,14] for Stefan problems and in [46] in the framework of boiling flows.

In this study, we will use linear extrapolations and quadratic extensions, which extend the extrapolated variable and respectively its first order normal derivative and its second order normal derivative, for the temperature field and for the mass fraction field.

For the purposes of this study on Leidenfrost droplets, a semi-implicit temporal integration has been used for the time advancement of these PDE in the longitudinal direction. Indeed, by using very refined grids close to a hot wall, if the interface crosses grid cells with very different sizes, the mesh points close to the interface in the very refined zones will be updated much more quickly than the mesh point located in a coarse region. This issue has been fixed by using an implicit temporal discretization in the longitudinal region to speed up the extrapolations in the whole domain. That semi-implicit temporal integration requires to solve a tri-diagonal linear system at each temporal iteration of the PDE resolution. It allowed us to reduce the number of temporal iterations from up to 2000 to about 50 to achieve a sufficiently steady state in the interface neighborhood in the case of the simulation of the Leidenfrost droplet.

5.3. Interface advection and re-initialization with a Level Set Method

The Level Set Method [8,34,43] is used to compute the interface motion. It consists of solving a convection equation for a Level Set Function ϕ :

$$\frac{\partial \phi}{\partial t} + \vec{V}_{int} \cdot \nabla \phi = 0. \quad (49)$$

Positive values of ϕ represent the liquid field and the negative values the gas field. \vec{V}_{int} is the interface velocity:

$$\vec{V}_{int} = \vec{V}_{liq} + \frac{\dot{m}}{\rho_{liq}} \vec{n} = \vec{V}_{gas} + \frac{\dot{m}}{\rho_{gas}} \vec{n}. \quad (50)$$

Next, a re-initialization step [43] can be performed to ensure that the ϕ function remains a signed distance in the computational domain. It can be done by solving iteratively the specific PDE

$$\frac{\partial d}{\partial \tau} = sign(\phi)(1 - |\nabla d|) \quad (51)$$

with d the reinitialized distance function, τ a fictitious reinitialization time and $sign(\phi)$ a smoothed signed function defined in [43]. Two temporal iterations of the redistancing equation, Eq. (51), are required at every time step.

Through this signed distance function, a good accuracy is preserved when computing the normal vector and the curvature at the interface, with the following simple relations:

$$\vec{n} = \frac{\nabla \phi}{|\nabla \phi|}, \quad (52)$$

$$\kappa(\phi) = -\nabla \cdot \vec{n}. \quad (53)$$

5.4. Time step constraints

A classical time-step constraint [22,43] accounting for the stability conditions for convection, viscosity (in the case of an explicit temporal discretization) and surface tension is imposed.

$$\Delta t_{conv} = \frac{\Delta x}{\max \|\vec{V}\|}, \quad (54)$$

$$\Delta t_{visc} = \frac{1}{4} \frac{\Delta x^2}{\max(v_{liq}, v_{gaz})}, \quad (55)$$

$$\Delta t_{surf_tens} = \frac{1}{2} \sqrt{\frac{\rho_{liq} \Delta x^3}{\sigma}}. \quad (56)$$

Finally, the global time step restriction can be computed with the following relation:

$$\frac{1}{\Delta t} = \frac{1}{\Delta t_{conv}} + \frac{1}{\Delta t_{visc}} + \frac{1}{\Delta t_{surf_tens}}. \quad (57)$$

6. Numerical results

We present in this section some numerical benchmarks in order to assess the relevance and the accuracy of the new method which is proposed in this paper. Firstly, in the next section we will describe an analytical theory of a static and isolated droplet in an infinite medium. This theory will provide us with an accurate reference solution that can be used to assess the different numerical solvers described previously.

6.1. Theory of the evaporation of a static droplet

The evaporation of a static droplet in an infinite medium can be modeled, see [42] for more details, by solving the following set of equations for the mass conservation, the energy conservation and the chemical species conservation in a 1D spherical coordinate system:

$$\frac{1}{r^2} \frac{d}{dr} (r^2 \rho_g u_r) = 0, \quad (58)$$

$$\frac{1}{r^2} \frac{d}{dr} (r^2 \rho_g C_{pg} u_r T) = \frac{1}{r^2} \frac{d}{dr} \left(k_g r^2 \frac{dT}{dr} \right), \quad (59)$$

$$\frac{1}{r^2} \frac{d}{dr} (r^2 \rho_g u_r Y) = \frac{1}{r^2} \frac{d}{dr} \left(\rho_g D_m r^2 \frac{dY}{dr} \right). \quad (60)$$

As the evaporation speed regression is very low for droplets (unlike boiling) in comparison to the Stefan flow which runs out in the normal direction at the interface, the unsteady terms can be neglected beside the convective and the diffusive derivatives.

The following boundary conditions, which can be deduced from the jump conditions Eqs. (10), (14), (16) must be applied on the droplet radius localized with $r = R_\Gamma(t)$:

$$V_r(r = R_\Gamma(t)) = \frac{\dot{M}(t)}{4\pi R_\Gamma^2(t) \rho_g}, \quad (61)$$

$$4\pi R_\Gamma^2(t) k_g \frac{dT}{dr} \Big|_{r=R_\Gamma(t)} = \dot{M}(t) L_{vap}, \quad (62)$$

$$4\pi R_\Gamma^2(t) \rho_g D_m \frac{dY}{dr} \Big|_{r=R_\Gamma(t)} = \dot{M}(t) (Y_\Gamma - 1), \quad (63)$$

where $Y|_\Gamma$ is the mass fraction at the interface, $T|_\Gamma$ is the temperature at the interface and $\dot{M}(t)$ is the total mass flow rate of evaporation. A first integration of these equations leads to the following relations:

$$u_r(r, t) = \frac{\dot{M}(t)}{4\pi r^2 \rho_g}, \quad (64)$$

$$\dot{M}(t) (C_{pg} (T(r, t) - T|_\Gamma) - L_{vap}) = 4\pi r^2 k_g \frac{dT}{dr}, \quad (65)$$

$$\dot{M}(t) (Y(r, t) - 1) = 4\pi r^2 \rho_g D_m \frac{dY}{dr}. \quad (66)$$

A second integration gives the temperature field and the mass fraction field:

$$\left(\frac{T(r, t) - T|_\Gamma - \frac{L_{vap}}{C_{pg}}}{T_\infty - T|_\Gamma - \frac{L_{vap}}{C_{pg}}} \right) = e^{-\left(\frac{\dot{M}(t) C_{pg}}{4\pi k_g}\right) \frac{1}{r}}, \quad (67)$$

$$\left(\frac{Y(r, t) - 1}{Y_\infty - 1} \right) = e^{-\left(\frac{\dot{M}(t)}{4\pi \rho_g D_m}\right) \frac{1}{r}}, \quad (68)$$

with the following far-field boundary conditions:

$$T(r \rightarrow \infty, t) = T_\infty,$$

$$Y(r \rightarrow \infty, t) = Y_\infty.$$

Table 1

Physical properties (SI) of fluids considered in the simulation presented in section 6.2 and section 6.3.

	ρ	M	C_p	K	σ	T_{sat}	L_{vap}	D_m	M
Liquid acetone	700	3.26E–6	2000	100	0.005	329	5.18E5	5.2E–5	0.058
Gas mixing	1	1E–7	1000	0.052	–	–	–	–	0.029

Next by pointing out that

$$\dot{M}(t) = -\frac{4\pi rk_g}{C_{pg}} \ln\left(\frac{T(r,t) - T|_\Gamma - \frac{L_{vap}}{C_{pg}}}{T_\infty - T|_\Gamma - \frac{L_{vap}}{C_{pg}}}\right) = -4\pi r\rho_g D_m \ln\left(\frac{Y(r,t) - 1}{Y_\infty - 1}\right), \quad (69)$$

we obtain the following equation for $T|_\Gamma$:

$$T|_\Gamma = T_\infty - \frac{L_{vap}}{C_{pg}} \left(1 - \left(\frac{Y_\infty - 1}{Y|_\Gamma - 1}\right)^{Le}\right), \quad (70)$$

with Le the Lewis number defined by:

$$Le = \frac{D_m \rho_g C_{pg}}{k}. \quad (71)$$

By using Eq. (39) and Eq. (40), we obtain two non-linear equations for the two unknowns $T|_\Gamma$ and $Y|_\Gamma$, for which a numerical solution can be computed easily.

Next, the mass flow rate $\dot{M}(t)$ of the droplet can be deduced from Eq. (69) and we obtain the following relation for the temporal evolution of the droplet diameter:

$$D_\Gamma(t) = \sqrt{D_0^2 - \frac{4\rho_g D_m}{\rho_l} \ln\left(\frac{Y_\infty - 1}{Y|_\Gamma - 1}\right)t}, \quad (72)$$

which is most commonly used as follows:

$$D_\Gamma(t) = \sqrt{D_0^2 - \frac{4\rho_g D_m}{\rho_l} \ln(1 + B_M)t}, \quad (73)$$

with B_M the mass transfer coefficient defined as:

$$B_M = \frac{Y|_\Gamma - Y_\infty}{1 - Y|_\Gamma}. \quad (74)$$

Let us remark that the global mass flow rate of evaporation can be expressed as follows:

$$\dot{M}(t) = 2\pi R_\Gamma(t)\rho_g D_m Sh B_M, \quad (75)$$

where Sh is the Sherwood number, which is a non-dimensional mass transfer flux, defined by the following expression in the case of a static evaporating droplet:

$$Sh = 2 \frac{\ln(1 + B_M)}{B_M}. \quad (76)$$

As this theoretical solution assumes that the temperature is uniform in the liquid phase, the average temperature of the droplet is equal to the interface temperature T_Γ . Let us remark that this theory provides a quasi-steady solution for the droplet evaporation since all unsteady terms have been neglected in the initial set of equations, however if the droplet is initialized with a temperature which is different from the steady temperature, the droplet temperature will initially vary (warming or cooling) until reaching the steady-state temperature. This last point will be illustrated in section 6.3.

6.2. Numerical simulations of the evaporation of a static droplet

A numerical test is proposed in this section in order to compare the accuracy and the relevance of the two numerical solvers for evaporation which have been described in this paper. The theoretical solution which has been derived in the previous section will provide a reference solution. As the characteristic time of droplets evaporation is very slow compared with the characteristic time of capillary waves, a large range of temporal scales must be solved to perform that kind of simulation. As a result, the number of temporal iterations, which are required to consider a sufficient duration of evaporation, is large (up to 100 000 iterations even if coarse grids are used). To reduce these drastic constraints, some physical properties of the fluid have been modified in order to speed up the numerical simulations. The physical properties of the liquid phase and of the vapor phase are summarized in the Table 1; these properties are close to those of the acetone for the liquid

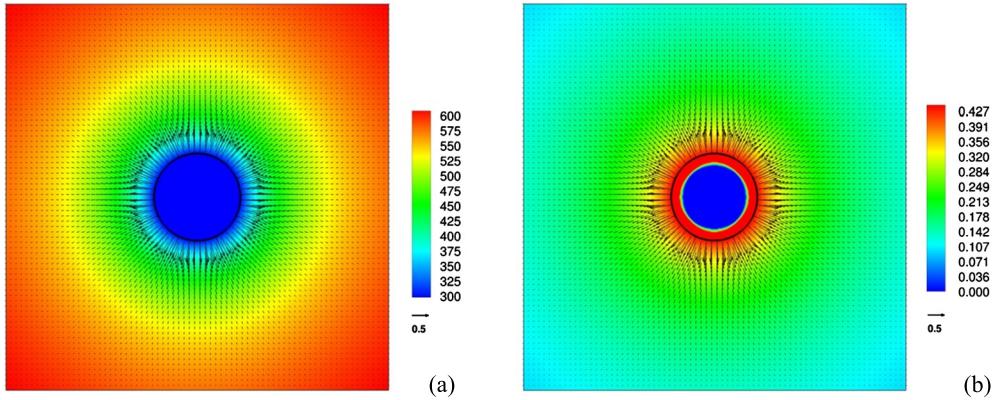


Fig. 1. Snapshots of the velocity field, of the interface location (black line), of the temperature field (a) or of the mass fraction field (b) on the computational grid (256×512) with the GFTSBE.

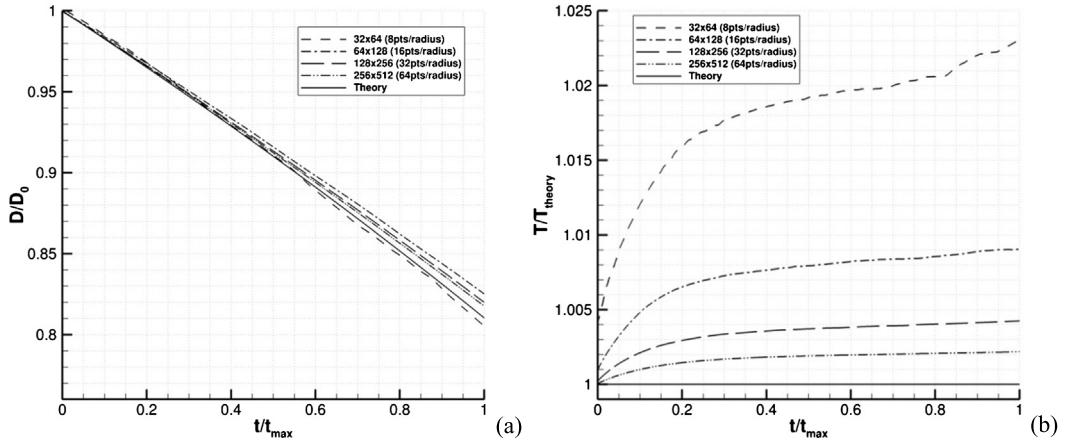


Fig. 2. Comparison between the GFTSE with linear extensions and theoretical results of the temporal evolution of the dimensionless droplet diameter (a) and on the dimensionless average droplet temperature (b).

phase and to those of a mixing of acetone vapor and air for the gas phase. Some values have been increased such as the mass diffusion coefficient, in order to speed the evaporation rate, and other properties have been decreased, e.g. the surface tension and the viscosity coefficients in order to increase the time step restriction depending on these variables. Since the numerical test proposed here is based on a theoretical solution which does not involve any capillary effects or viscous effects, the modification of these values does not change the expected reference solution. The thermal conductivity of the liquid has been strongly increased to guarantee that the temperature field will be uniform inside the droplet, consistent with the theory described in the previous section. The simulations are performed with an axisymmetric coordinate system. The initial diameter of the droplet is equal to $D_0 = 100 \mu\text{m}$. The dimensions of the computational field are $l_r = 2D_0$ and $l_z = 4D_0$. The boundary conditions for the temperature field and for the mass fraction field are imposed as the values of the theoretical solution. As a result these boundary conditions evolve slowly with time when the droplet diameter is decreasing. The temperature and the mass fraction in the far field (beyond the computational domain) are $T_\infty = 700 \text{ K}$ and $Y_\infty = 0$ respectively. The initial conditions on the temperature field and on the mass fraction field are also imposed by using the theoretical solution. In particular, the droplet temperature is initialized by using the theoretical steady-state temperature. For the velocity field, outflow boundary conditions are used to let the evaporated gas out of the computational domain.

In this configuration, the theoretical steady-state temperature is equal to $T_r = 294.94 \text{ K}$. Several computations have been performed in order to assess the accuracy of the GFTSE and of the new method, GFTSBE, with four different computational grids (32×64 , 64×128 , 128×256 , 256×512), which correspond respectively to 8 grid nodes/radius, 16 grid nodes/radius, 32 grid nodes/radius and 64 grid nodes/radius. Moreover, the influence of the order of the extensions which are used to extrapolate the temperature field and the mass fraction field has also been tested on the overall accuracy of the computation, since all the simulations presented in this section, have been performed with either linear extensions or with quadratic extensions. The computations are performed until the droplet diameter reaches $0.8D_0$, which corresponds to a mass evaporation of about 50% of the initial droplet mass.

In Figs. 1(a) and 1(b), snapshots of the simulations have been displayed to visualize the velocity field, the interface position and respectively the temperature field and the mass fraction field. In Figs. 2(a) and 2(b) the temporal evolution of

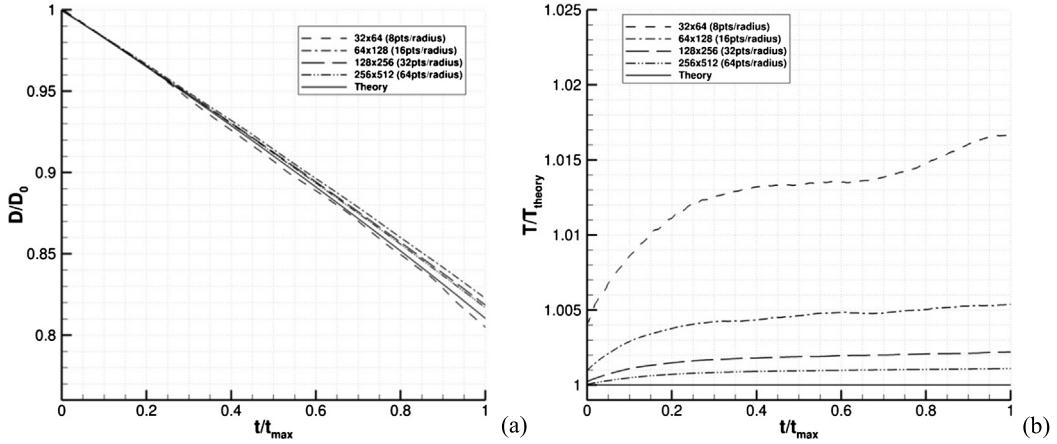


Fig. 3. Comparison between the GFTSE with quadratic extensions and theoretical results of the temporal evolution of the dimensionless droplet diameter (a) and on the dimensionless average droplet temperature (b).

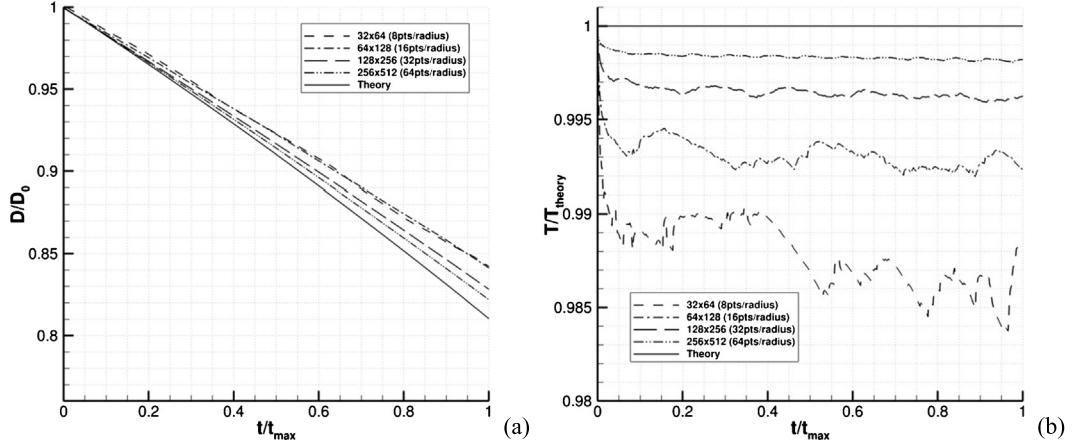


Fig. 4. Comparison between the GFTSBE with linear extensions and theoretical results of the temporal evolution of the dimensionless droplet diameter (a) and on the dimensionless average droplet temperature (b).

the non-dimensional droplet diameter and of the non-dimensional droplet temperature is plotted for different computational grids if GFTSE is used in conjunction with linear extensions on the temperature field and on the mass fraction field. These figures clearly indicate that the simulations converge to the theoretical solution. A first order accuracy can be roughly extrapolated from Fig. 2(b), whereas Fig. 2(a) does not allow evaluating an accuracy order since the error is lower on the coarsest grid. This behavior is typical for simulations where opposite errors can be compensated and can lead to misleading conclusions if sufficient attention is not paid to the analysis of the simulation results. Indeed, that can be explained by noting that the temperature droplet is strongly overvalued on the coarsest grid which increases spuriously the evaporation mass flow rate, whereas an undervaluation of the droplet diameter speed regression can result from other errors. However, this non-monotonic convergence disappears if thinnest grids are used.

In Figs. 3(a) and 3(b), the same variables have been plotted with the same solver but with quadratic extensions instead of linear extensions. These results are very similar to the results obtained with linear extensions, but we can observe a slight gain in accuracy since, on the thinnest grid, the error on the average droplet temperature is about 0.1% (about 0.3 K) instead of 0.2% (about 0.6 K). Such trends have already been reported in [14] for the Stefan problem or in [46] for boiling flows, whereas the gain in accuracy was much more important in the latter studies.

Next, we have plotted in Figs. 4(a) and 4(b), the numerical results obtained with GFTSBE and linear extensions. These numerical results also demonstrate that the GFTSBE converges to the expected solution. The order of accuracy seems to be approximately the same as with GFTSE. GFTSBE provides results which are slightly better on the average temperature but the error on the temporal evolution of the diameter is more important. We can remark that with this solver the temperature is slightly undervalued, unlike GFTSE which overestimates the average droplet temperature.

At last, in Figs. 5(a) and 5(b), the numerical results obtained with GFTSBE and with quadratic extensions are plotted. A gain in accuracy on the average temperature is observed in comparison with other solvers. For example, if the thinnest grid is used, the error on temperature is less than 0.1%, and the error on the temporal evolution of the droplet diameter is

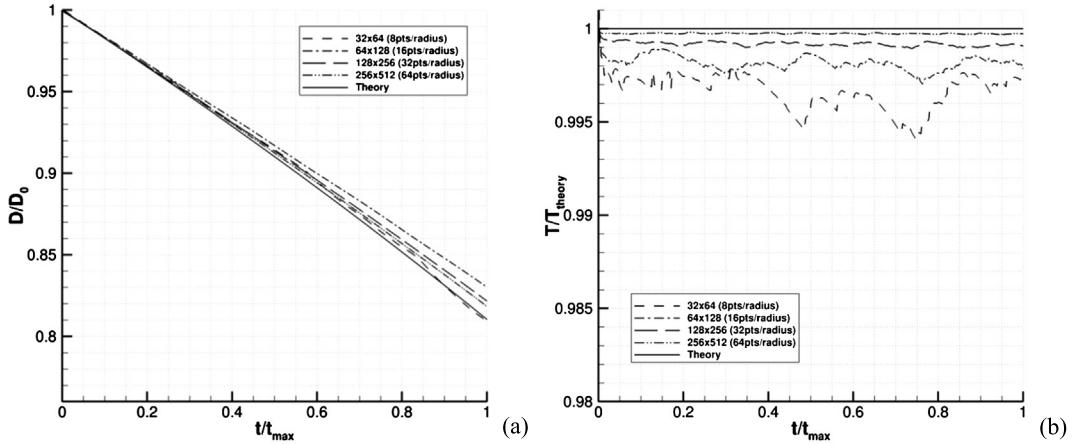


Fig. 5. Comparison between the GFTSBE with quadratic extensions and theoretical results of the temporal evolution of the dimensionless droplet diameter (a) and on the dimensionless average droplet temperature (b).

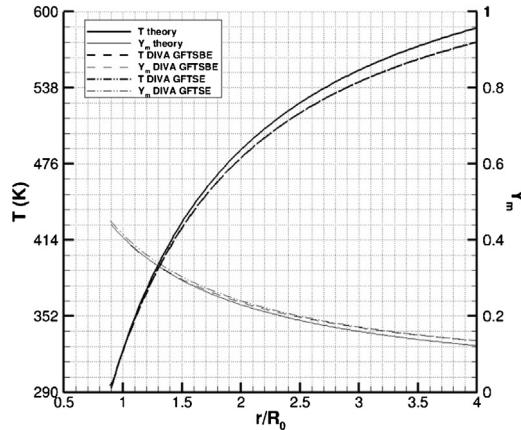


Fig. 6. Comparisons between theory and numerical simulations of the radial evolution of the temperature and of the mass fraction in the gas phase with the GFTSBE and with the GFTSE at a given time with the grid 128×256 .

approximately in the same range as with GFTSE (less than 5% at the end of the simulation). In conclusion of this section, we can assert that this combination of numerical methods (GFTSBE + quadratic extension) is the most accurate of the four which have been tested in this paper.

In Fig. 6, a comparison, between the theory and the numerical simulations, of the radial evolution of the temperature and of the mass fraction, has been plotted both for GFTSBE and GFTSE. This comparison is convincing since the theoretical curves and all the numerical curves are superimposed in the neighborhood of the interface. Let us notice that a slight discrepancy appears close to the boundary condition. This slight discrepancy can be explained by considering that the outflow boundary conditions, which are used in the simulations do not perfectly maintain the spherical symmetry of the theoretical velocity field.

We can conclude from this first benchmark that both GFTSE from [45] and GFTSBE, which is presented in this work, converge to a solution which is very close to the theoretical solution described in the previous section. A gain in accuracy is observed if GFTSBE and quadratic extensions are used. However, we must also remark that a more detailed quantification of the accuracy seems not to be possible in these simulations. Indeed the numerical simulations converge toward a solution which seems to be slightly different from the theoretical one, if the temporal evolution of the droplet diameter is considered. Various reasons could explain this slight discrepancy between the numerical simulations and the theory. For instance, unlike the numerical simulations, the unsteady terms are neglected in the theory. Another source of errors could come from the outflow boundary conditions which do not maintain perfectly the spherical symmetry of the problem. Finally, the droplet temperature is assumed to be uniform in the theory, whereas the simulations do not perfectly match this assumption.

6.3. Non-steady static droplet evaporation

In this section we present some numerical results which have been obtained with the same configuration as in the previous section, except for the droplet initial temperature which has been changed in order to check if after a transient step

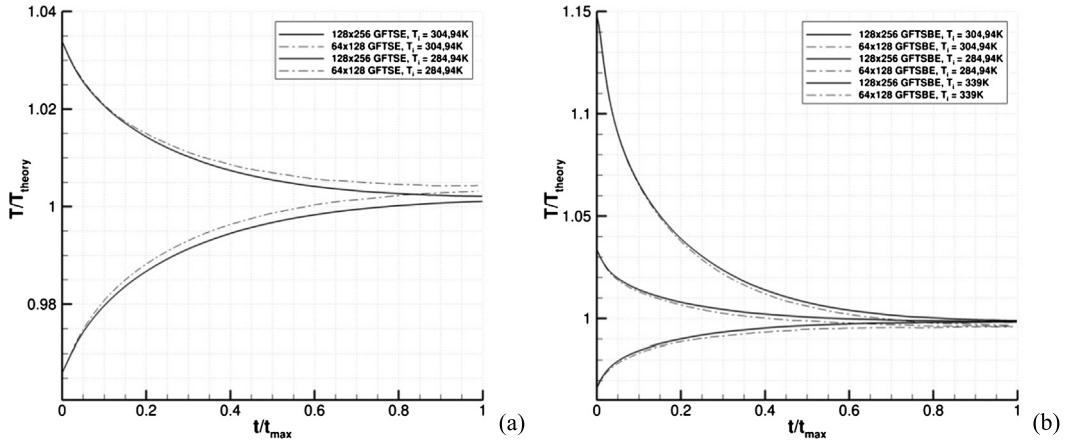


Fig. 7. Temporal evolution of the average dimensionless droplet temperature with the GFTSE (a) for two different initial temperatures and with the GFTSBE (b) for three different initial temperatures.

the droplet temperature will tend towards the theoretical steady-state temperature of evaporation. GFTSE has been tested with two different initial temperatures, one which is higher ($T = 304.94$ K) than the theoretical steady-state temperature and another which is lower ($T = 284.94$ K). The two computations have been performed by using quadratic extrapolations, since they provide more accurate results, and with two computational grids (64×128 and 128×256). The results are plotted in Fig. 7(a) and show clearly that all the computations tend to the correct theoretical steady-state temperature. Moreover, we can observe in this figure the good agreement between the simulations performed on the two grids, which further demonstrates the spatial convergence of the overall solver.

GFTSBE has been tested in the same conditions as GFTSE, but with three different initial temperatures, the same two as for GFTSE and a third temperature which is higher ($T = 339$ K) than the saturation temperature ($T = 329$ K). The objective of this last configuration is to show the stability of GFTSBE even in the case where superheated liquids are involved, unlike GFTSE which is unstable in this situation due to Eq. (42). The numerical results, summarized in Fig. 7(b), highlight the correct behavior of this numerical solver, even if a superheated liquid is considered. This second benchmark proves that GFTSBE is able to deal with both liquid evaporation and boiling whereas GFTSE fails to address this specific situation.

More detailed investigations on the spatial convergence are proposed in an appendix with three different computational grids 32×64 , 64×128 , 128×256 for all of these configurations.

6.4. Moving droplet evaporation

In this section, we are now interested by the evaporation of a moving spherical droplet. The simulations are still performed with an axisymmetric coordinate system. The initial diameter of the droplet is equal to $D_0 = 100 \mu\text{m}$. The dimensions of the computational field are $l_r = 4D_0$ and $l_z = 8D_0$ in order to minimize containment effects. We use a uniform grid inside the drop and in the gas phase up to a distance $D_0/2$ from the interface. Beyond this distance, a non-uniform grid is used to minimize the number of grid cells. In this way, the mesh is sufficiently refined with respectively 16 and 32 grid cells to discretize a droplet radius for the two computational grids which contain respectively 64×128 and 128×256 points. As the droplet will travel an important distance before a significant evaporation occurs, the computations are performed with a moving frame in order to ensure that the droplet is maintained at the center of the computational field. A constant relative velocity $V_z = 2 \text{ m s}^{-1}$ is imposed between the droplet and the gas phase throughout the computation. The gravity is neglected, and the computation is initialized with the static theoretical solution for the temperature and the mass fraction fields. When solving the Navier–Stokes equations, we use an inflow boundary condition at the bottom of the domain to blow the droplet with the correct velocity, slip boundary conditions are imposed on the right side of the computational field and outflow boundary conditions are used at the top of the domain. We solve the conservation of the energy and of the species equations with a Dirichlet boundary condition at the bottom of the domain $T_\infty = 700$ K and $Y_\infty = 0$, and for these equations a Neumann boundary condition can be imposed on the right side and on the top of the domain since the computational field is large enough to avoid containment effects in the far-field. In this configuration, the value of the Reynolds number is around 20, and the Weber number is sufficiently small to ensure that the droplet will remain spherical. Therefore, in such a configuration some correlations can be used to build a reference solution to assess our numerical methods with this more complex benchmark. For example, the following correlation has been proposed in [36] to include the Reynolds number, Re , to correct the Sherwood number compared to the static theory

$$\frac{Sh}{(1 + B_M)^{0.7}} = 2 + 0.87 Re^{0.5} Sc^{0.33}, \quad (77)$$

Table 2

Physical properties (SI) of fluids considered in the simulation presented in section 6.4.

	ρ	M	C_p	K	σ	T_{sat}	L_{vap}	D_m	M
Liquid acetone	700	3.26E-4	2000	0.161	0.0237	329	5.18E5	5.2E-5	0.058
Gas mixing	1	1E-5	1000	0.052	–	–	–	–	0.029

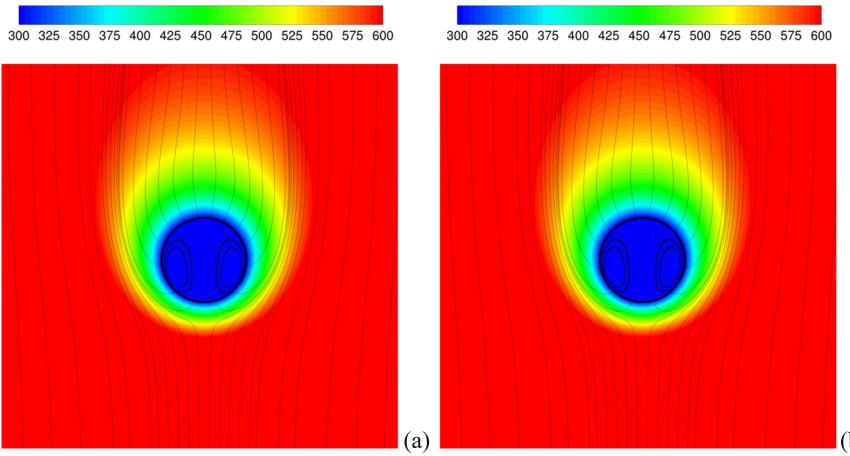


Fig. 8. Temperature field, streamlines and interface position (thick black line) around a moving droplet which evaporates in a hot gas. Comparison between the GFTSE (a) and the GFTSBE (b).

where the Sherwood number Sh takes into account the modification of the mass transfer flux due to the convection and its coupling with evaporation and Sc is the Schmidt number defined by

$$Sc = \frac{\mu_g}{\rho_g D_m}. \quad (78)$$

This modified expression of the Sherwood number can next be used to compute a modified mass flow rate, $\dot{M}(t)$, depending on Reynolds number and Schmidt number in Eq. (75). The correlation, Eq. (77), is based on boundary fitted numerical simulations which guarantee a high accuracy but which are limited to a single spherical or steady shape droplet, unlike the numerical methods presented in this paper which can be used with an interface of moving and arbitrary shape. The numerical simulations performed in [36] involve a volatile liquid (n-heptane) in a hot gas (800 K) for a range of Reynolds number varying between 100 and 13.4.

Our numerical simulations have been carried out with the physical properties which are given in the Table 2, which are nearly identical to the Table 1. A more realistic value than in the previous section is used here for the thermal conductivity, since we do not compare our numerical simulation against a theoretical study which assumes that the droplet temperature is perfectly uniform. The surface tension and the viscosity have been increased in order to obtain realistic values of the Reynolds number and of the Weibull number since this benchmark allows the computation of the influence of convection on the overall process of evaporation. As a result, the number of temporal iterations required to compute a significant evaporation of the droplet will be more important than in the static case. Therefore, these computations are more expensive than those presented in the previous section.

Comparisons between the correlation, Eq. (77), and direct numerical simulations with GFTSE and GFTSBE (with quadratic extensions in both cases) are now presented.

The temperature field, the streamlines and the interface location have been plotted at a given time in Fig. 8(a) for GFTSE and in Fig. 8(b) for GFTSBE to demonstrate the qualitative agreement between these two methods. Further results are presented in Fig. 9(a) and in Fig. 9(b) to show the agreement between the two methods for the mass fraction field. These depict the streamlines, inside the droplet and outside of the droplet, in order to bring out the coupling between the external flow and the radial velocity field due to evaporation (Stefan flow). Finally, in Fig. 10, the internal temperature field of the droplet is plotted to illustrate that it is slightly modified by the internal recirculating zone since a colder region is observed in the center of the axisymmetric vortex.

In Fig. 11(a) the temporal evolution of the normalized diameter is plotted until the diameter decreases by about 15%. This figure shows that the two numerical methods converge towards a solution which is very close to the results obtained by using the correlation in Eq. (77). The agreement between the numerical simulation and the correlation is good, especially if GFTSBE is used. The temporal evolutions of the average droplet temperature are plotted in Fig. 11(b). This figure further demonstrates the accurate behavior and the convergence of all numerical methods since the deviations from the theoretical steady temperature are very small (less than 0.5%) regardless of the solver and the computational grid used.

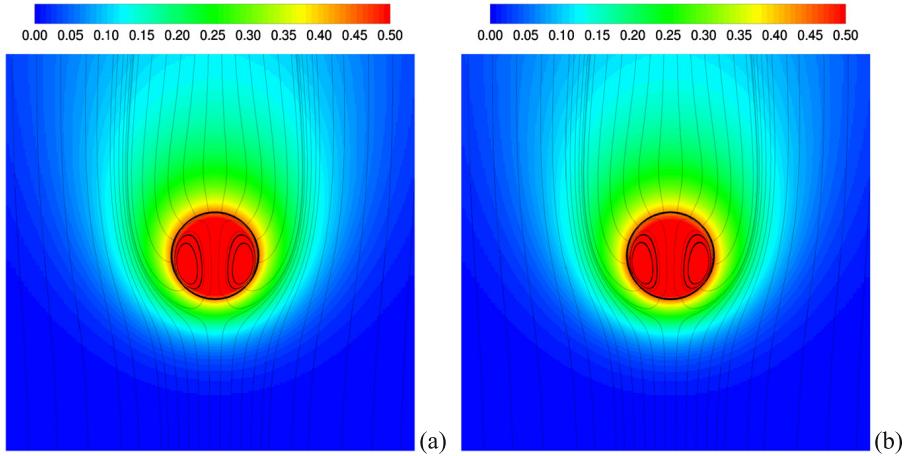


Fig. 9. Mass fraction field, streamlines and interface position (thick black line) around a moving droplet which evaporates in a hot gas. Comparison between the GFTSE (a) and the GFTSBE (b).

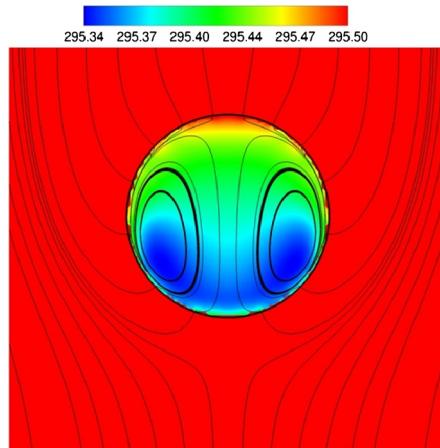


Fig. 10. Zoom on the temperature field in the liquid phase with the streamlines and the interface position (thick black line) around a moving droplet which evaporates in a hot gas with the GFTSE.

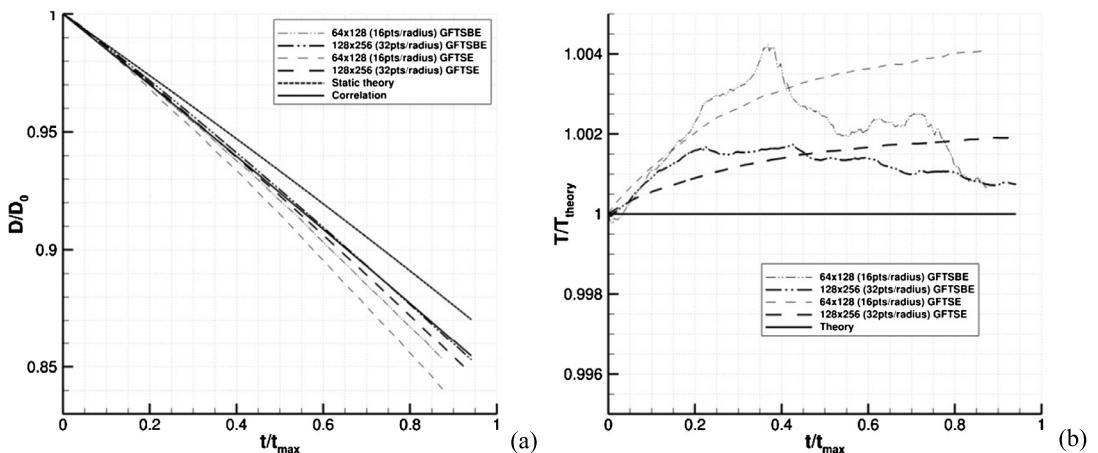


Fig. 11. Comparison between the GFTSBE, the GFTSE (both with quadratic extensions) and a semi-empirical correlation of the temporal evolution of the dimensionless droplet diameter (a) and of the dimensionless average droplet temperature (b).

Table 3

Physical properties (SI) of fluids considered in the simulation presented in section 6.5.

	ρ	M	C_p	k	σ	T_{sat}	L_{vap}	D_m	M
Liquid water	1000	0.00113	4180	0.6	0.07	373	2.3E6	2E-5	0.018
Gas mixing	1.226	3E-5	1000	0.046	-	-	-	-	0.029

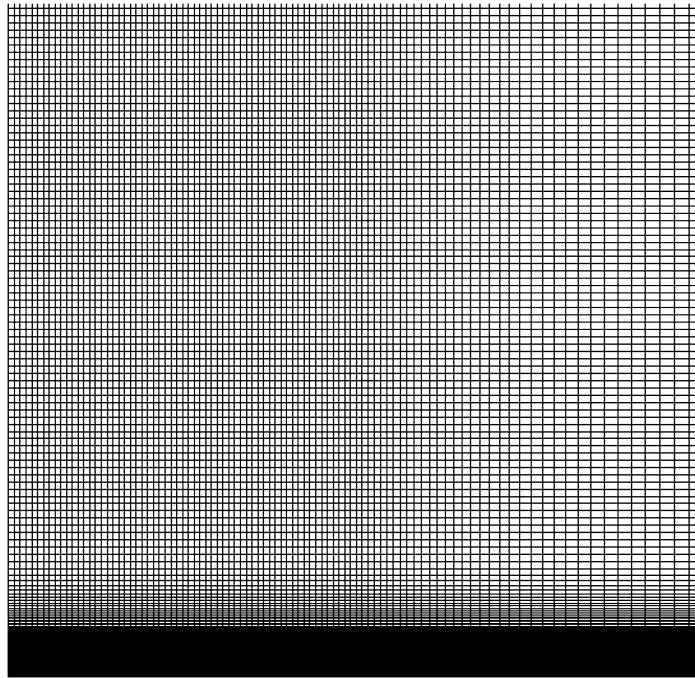


Fig. 12. Strongly refined computational grid used in the simulation of the Leidenfrost droplet.

Due to this third benchmark, the relevance of the two numerical solvers to compute the evaporation of a moving droplet has been demonstrated. As it is the case of the first benchmark, a gain in accuracy has also been observed if GFTSBE is used.

6.5. Levitation and bouncing of an impacting droplet on a hot plate (Leidenfrost effect)

We present here the numerical simulation of a droplet impacting a hot plate. In the Leidenfrost regime, a droplet never touches the plate because the high mass flow rate of vaporization leads to the formation of a very thin layer of saturated vapor. The pressure force resulting from the flow in this thin layer is sufficiently high to ensure the sustentation of the droplet above the hot plate. Several difficulties make the numerical simulation of this singular phenomenon challenging. A major issue to be overcome consists in designing a specific phase change model, which is suited both for evaporation and boiling, since the two phenomena occur simultaneously. Another difficulty, that must be addressed, results from the use of a locally refined grid to capture the formation of the thin vapor layer between the plate and the drop during the impact. Therefore, an implicit temporal discretization must be applied to compute all the diffusion terms (viscous terms, thermal conduction and mass fraction diffusion). In the same way, the resolution of the hyperbolic partial differential equations [1,14, 46], which results from extrapolation techniques, must also be discretized with an implicit scheme. For these equations, this implicit temporal discretization is useful only in the refined direction of the mesh. Thus, as it has been explained previously, that can be done by solving a tri-diagonal linear system for every time step of the extrapolation process. This implicit discretization allows the use of large time steps during the extrapolation. It ensures that the extrapolation will be advanced on a number of grid points sufficiently important both in the region where the grid is coarse and in the region where the grid is very refined.

In the simulation which is presented here, the initial droplet diameter is $D_0 = 130 \mu\text{m}$, the initial velocity is $V_0 = 2 \text{ m s}^{-1}$ which corresponds to a Weber number $We = \frac{\rho_1 V_0^2 D_0}{\sigma} = 7.4$. The initial temperature of the droplet is $T_0 = 290 \text{ K}$ and the hot plate is assumed to be isothermal $T_{wall} = 800 \text{ K}$. The following Neumann boundary condition, $\nabla Y_1|_{wall} = 0$, is imposed for the mass fraction on the hot wall in order to ensure that the chemical species do not diffuse inside the solid domain. The physical properties of the liquid phase and of the gas phase are summarized in Table 3. Let us notice that, for sake of simplicity, the physical variables of the model are constant and therefore do not depend on the local temperature, nor

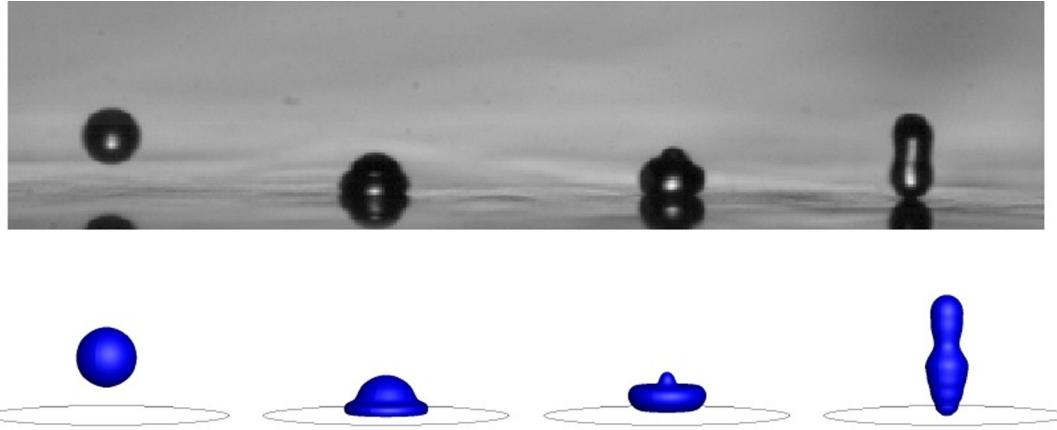


Fig. 13. Comparison between our numerical simulation and experimental data of the interface evolution during the impingement of a levitating droplet on a hot wall at different times with $We = 7$ ($t_1 = 9.83 \times 10^{-5}$ s, $t_2 = 1.36 \times 10^{-4}$ s, $t_3 = 1.57 \times 10^{-4}$ s, $t_4 = 2.69 \times 10^{-4}$ s).

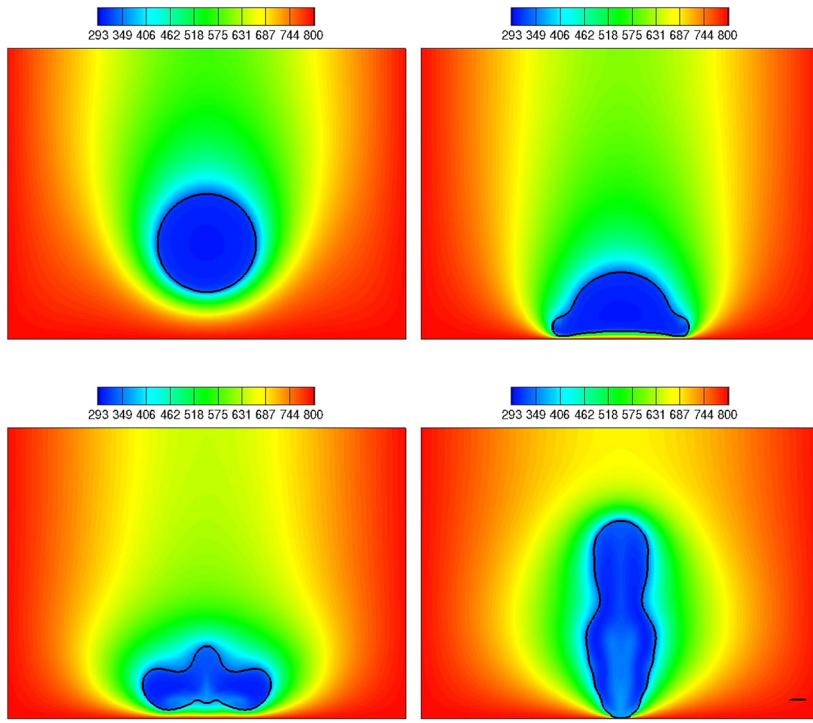


Fig. 14. Temperature field and interface evolution (in black) during the impingement of a levitating droplet on a hot wall at different times ($t_1 = 9.83 \times 10^{-5}$ s, $t_2 = 1.36 \times 10^{-4}$ s, $t_3 = 1.57 \times 10^{-4}$ s, $t_4 = 2.69 \times 10^{-4}$ s).

on the local mass fraction. The size of the computational domain is $l_r = 2D_0$ and $l_z = 3l_r/2$. The simulations have been performed with three computational grids 64×128 , 96×192 and 144×288 . As it is pointed out in Fig. 12, the simulations are performed on a non-uniform grid with an important concentration of grid cells in the region close to the wall in order to capture and solve accurately the thin vapor layer which is formed between the hot plate and the droplet. The radial direction is decomposed into two parts, a uniform zone between the center of the domain and the radial coordinate $r = l_r/2$ with 40, 60 and 90 points respectively for the coarsest grid, the medium grid, and the thinnest grid, and a continuously stretched zone up to $r = l_r$ which contains 24, 36 and 54 grid cells. In the longitudinal direction, the mesh grid is also decomposed into two parts: a very refined grid containing respectively 20, 40 and 80 grid cells in the region close to the wall between the coordinates $z = -l_z/2$ and $z = -l_z/2 + \varepsilon$ with $\varepsilon = 0.06l_z$ and a uniform one in the rest of the domain, which contains 108 points for the coarse grid, 152 points for the medium grid and 208 points for the refined grid. The transition between refined and uniform zone is continuous in order to preserve the accuracy of the spatial discretization.

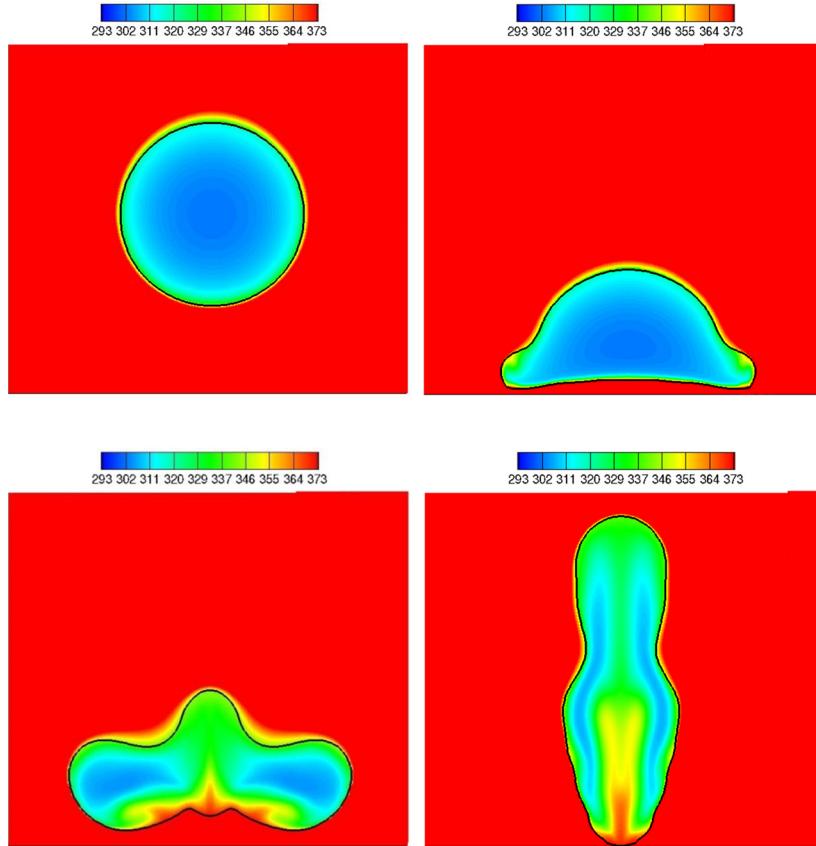


Fig. 15. Temperature field in the liquid phase and interface evolution (in black) during the impingement of a levitating droplet on a hot wall at different times ($t_1 = 9.83 \times 10^{-5}$ s, $t_2 = 1.36 \times 10^{-4}$ s, $t_3 = 1.57 \times 10^{-4}$ s, $t_4 = 2.69 \times 10^{-4}$ s).

In Fig. 13, a comparison is presented between the numerical simulation and experimental data [3,5] on the temporal evolution of the droplet shape. This comparison shows the qualitative agreement between the simulation and the experimental data. The temperature field has been plotted at different times of the simulation in the whole domain, Fig. 14, and inside the droplet, Fig. 15, to highlight the ability of the numerical method which has been designed for this purpose. Some snapshots have been plotted in Fig. 16 in order to visualize the temporal evolution of the mass fraction field. The transition between the evaporation regime and the boiling regime is clearly visualized on these snapshots since the formation of a thin saturated vapor layer is observed between the hot wall and the liquid-gas interface, during the droplet spreading. Let us notice that the description of this transition between two regimes of phase change has been possible through the new algorithm, GFSTBE, as opposed to GFTSE which is unstable in such drastic conditions whereas GFTSB is unable to describe the evolution of the mass fraction field and, thus, of the internal temperature of the droplet.

Finally, snapshots of the velocity field during the impact have also been plotted in Fig. 17 to point out the resolution of the simulation presented, since the velocity field in the very thin vapor layer is accurately solved all along the simulation. Some quantitative comparisons between our numerical simulations and the experimental data from [3] and [5] are presented in Table 4. Specifically, we have focused our attention on three variables which have been measured in the experiments: the non-dimensional maximum spreading diameter β , defined as the maximum spreading diameter divided by the initial diameter of the droplet, the normal restitution coefficient r_n , which is the ratio of the bouncing droplet velocity to the impacting droplet velocity, and the droplet heating ΔT , defined as the difference of the average droplet temperature after the impact and before the impact. The numerical simulation performed with the coarse grid (64×128) is unstable because the number of grid points in the refined zone is not large enough to capture the formation of the vapor layer between the hot wall and the droplet in the Leidenfrost regime. However this computation is used to check the spatial convergence of the simulations during the first part of the simulation (until $t = 0.00014$ s).

The comparison between the experimental results and the numerical simulations is satisfactory, especially for the dynamical parameters β and r_n since the differences reported are respectively around 12% and 7%. Moreover we can remark that the simulations are well converged since very low differences are reported between the computations performed on the three different grids, see Figs. 18(a) and 18(b) to visualize respectively the temporal evolution of the non-dimensional spreading diameter of the droplet and the non-dimensional average droplet velocity for the two different computational

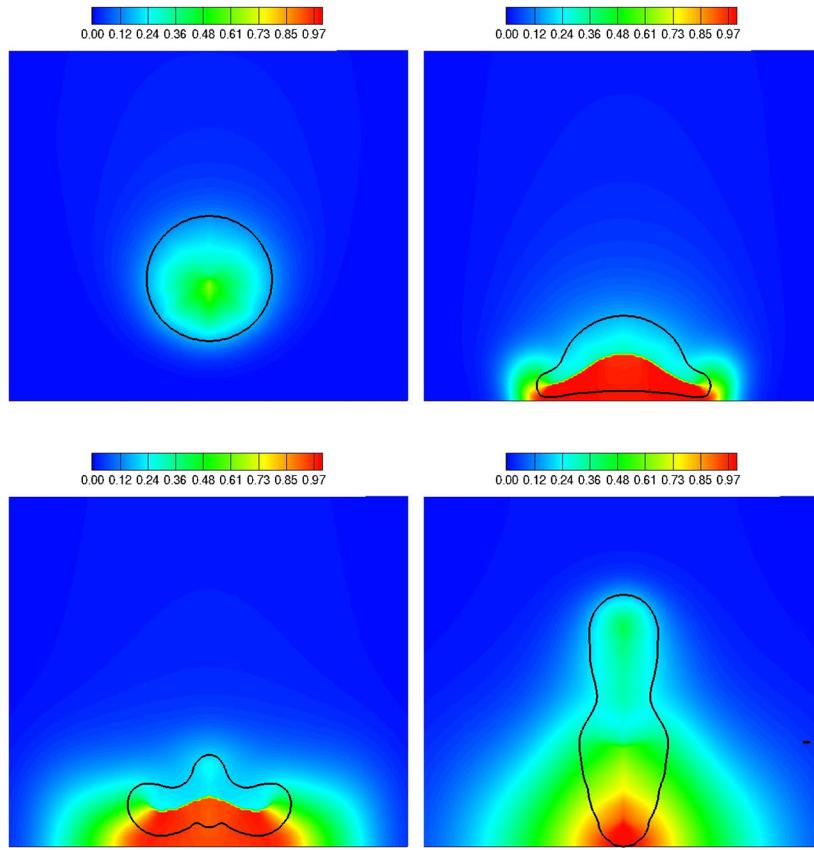


Fig. 16. Mass fraction field and interface evolution (in black) during the impingement of a levitating droplet on a hot wall at different times ($t_1 = 9.83 \times 10^{-5}$ s, $t_2 = 1.36 \times 10^{-4}$ s, $t_3 = 1.57 \times 10^{-4}$ s, $t_4 = 2.69 \times 10^{-4}$ s).

grids. Concerning the droplet heating plotted in Fig. 19, the experimental measurements (Table 4) are much more approximate due to the important spatial variations of the temperature field inside the droplet. Finally, we can observe that our numerical results are close to the range of droplet heating which has been measured. In the absence of more accurate experimental data, it denotes a qualitative agreement, between the simulations and the experiments, on the droplet heating.

7. Conclusion

The main motivation of this paper is to present a new numerical model to deal both with boiling flows and evaporation of droplets in the framework of the direct numerical simulation of two phase flows with deformable interface. This new method is a generalization of the Ghost Fluid Thermal Solver for Boiling presented in [13]. Indeed a further equation for mass fraction species conservation is incorporated in the overall algorithm with the suitable jump condition as in the Ghost Fluid Thermal Solver for Evaporation presented in [45]. However, unlike the numerical method presented in the latter, the new method, GFTSBE, deals with the droplet evaporation and with boiling flows. The key-point of this new method lies in the computation of the mass fraction field equation with an imposed Robin Boundary Condition at the interface. This formulation allows to capture continuously the transition between evaporation and boiling in thin regions of saturated vapor, as it can occur in Leidenfrost droplets or in superheated droplets. Let us notice that this numerical solver could also be used to perform numerical simulations involving a contact line between a liquid droplet and a superheated wall. The simulations, presented in this paper, demonstrate the efficiency of this method on several accurate benchmarks and demonstrate its relevance for simulating the dynamics of Leidenfrost droplets. Some numerical procedures, specifically developed to deal with the Leidenfrost droplet, are also presented in order to capture the thin vapor layer between the wall and the interface by using very refined grids.

Acknowledgements

The authors gratefully acknowledge funding by ANR (French Research National Agency) in the context of the IDHEAS (Interaction of Droplets with HEAted Surface) program ANR-NT09 432160 and by ONERA the French Aerospace Laboratory.

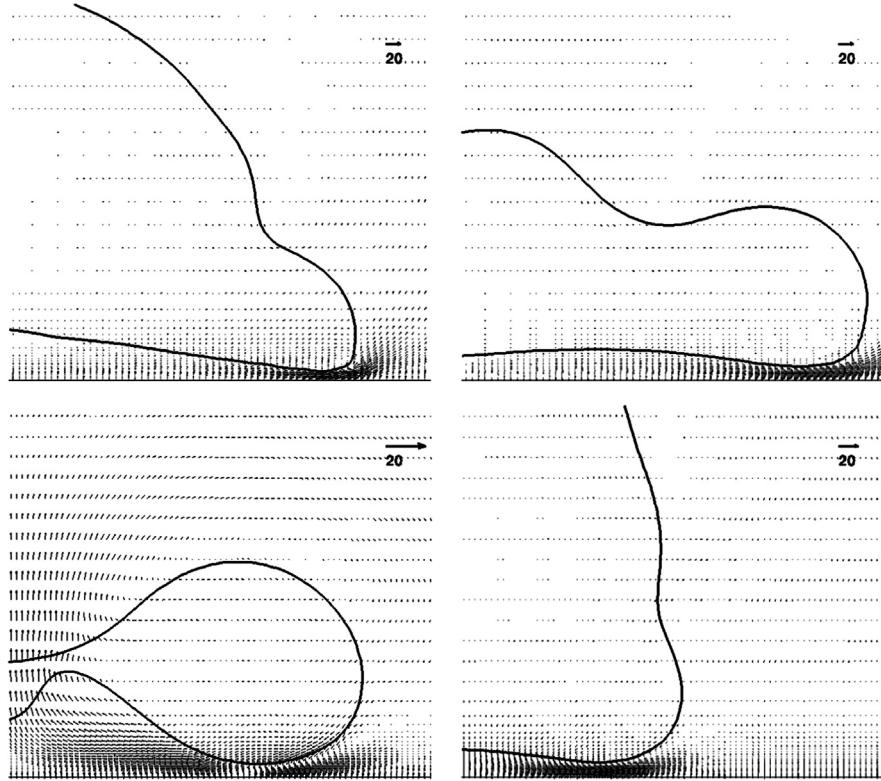


Fig. 17. Velocity field and interface evolution during the impingement of a levitating droplet on a hot wall at different times ($t_1 = 1.36 \times 10^{-4}$ s, $t_2 = 1.47 \times 10^{-4}$ s, $t_3 = 1.72 \times 10^{-4}$ s, $t_4 = 2.18 \times 10^{-4}$ s).

Table 4

Comparison between the numerical simulations and the experimental data from [3,5] for the Leidenfrost droplet with $We = 7.4$ K.

	B	r_n	ΔT
Experimental measurements	1.4	0.59	~10–20 K
Numerical simulation (96 × 192) (40 points in the refined zone)	1.57	0.61	19.45 K
Numerical simulation (144 × 288) (80 points in the refined zone)	1.57	0.63	20.6 K

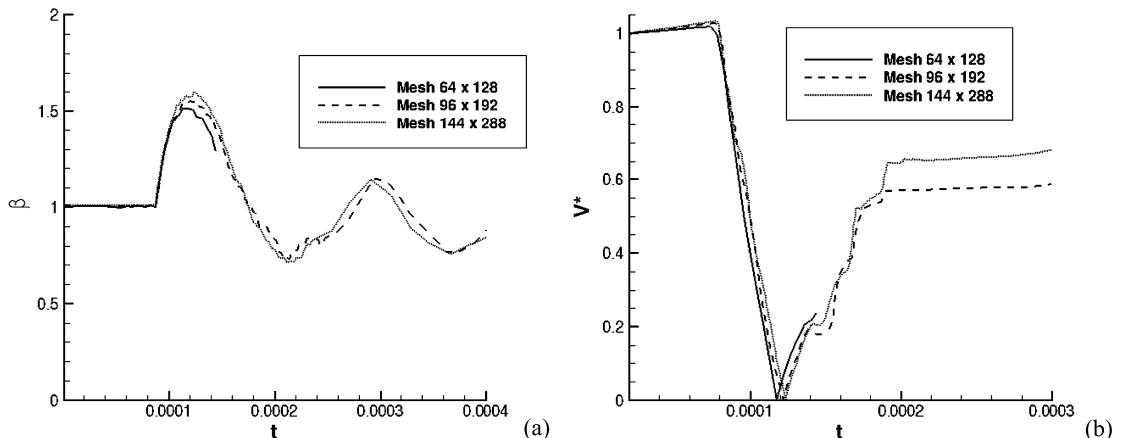


Fig. 18. Temporal evolution of the dimensionless droplet spreading diameter (a) and of the dimensionless average droplet velocity (b).

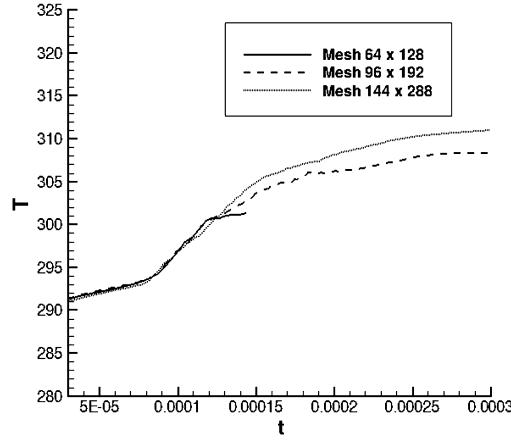


Fig. 19. Temporal evolution of the average droplet temperature (K).

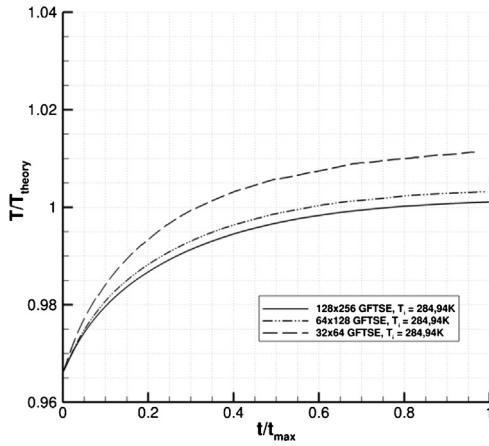


Fig. 20. Temporal evolution of the average dimensionless droplet temperature with the GFTSE and an initial temperature $T = 284.94$ K.

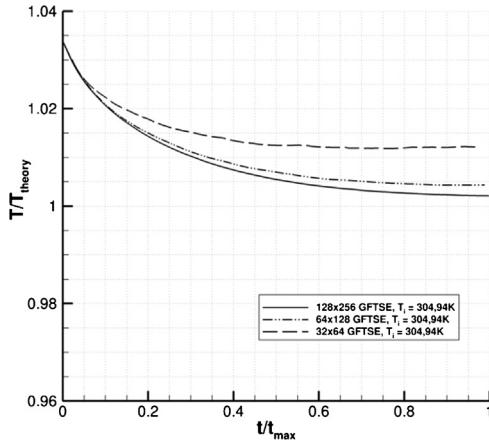


Fig. 21. Temporal evolution of the average dimensionless droplet temperature with the GFTSE and an initial temperature $T = 304.94$ K.

Appendix A. Spatial convergence for the benchmark of the non-steady static droplet evaporation

We present in this appendix more detailed results to point out the spatial convergence of GFTSE and GFTSBE with the benchmark of the non-steady static droplet evaporation. Simulations with a coarse grid, 32×64 points have been added and are compared with the medium and refined grid, with respectively 64×128 and 128×256 points. The numerical results

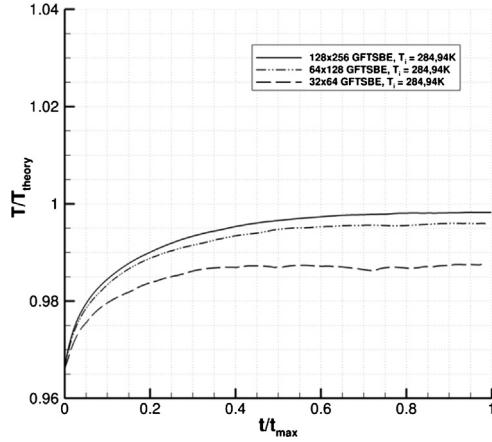


Fig. 22. Temporal evolution of the average dimensionless droplet temperature with the GFTSBE and an initial temperature $T = 284.94$ K.

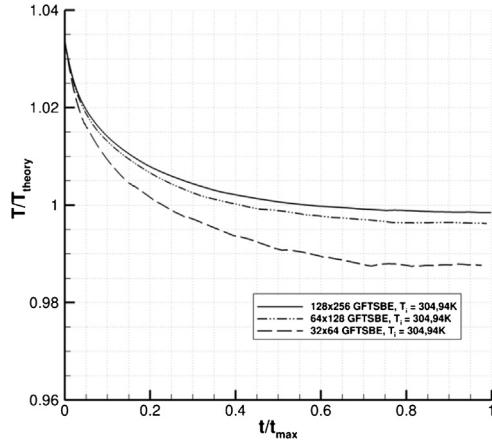


Fig. 23. Temporal evolution of the average dimensionless droplet temperature with the GFTSBE and an initial temperature $T = 304.94$ K.

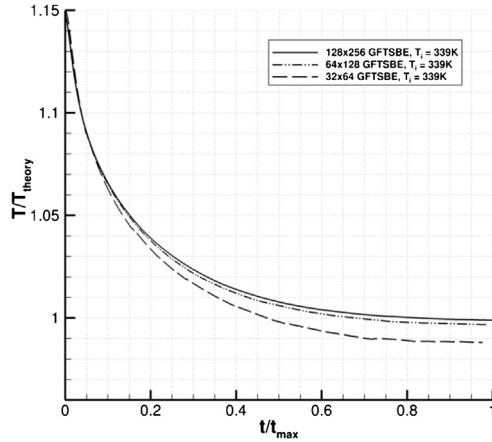


Fig. 24. Temporal evolution of the average dimensionless droplet temperature with the GFTSBE and an initial temperature $T = 339$ K.

obtained with GFTSE are presented in Fig. 20 and Fig. 21, and the results obtained with GFTSE are plotted in Fig. 22, Fig. 23 and Fig. 24. All these figures illustrate the spatial convergence of both methods for this benchmark.

References

- [1] T. Aslam, A partial differential equation approach to multidimensional extrapolation, *J. Comput. Phys.* 193 (2003) 349–355.

- [2] V.P. Carey, Liquid-Vapor Phase Change Phenomena: An Introduction to the Thermophysics of Vaporization and Condensation Processes in Heat Transfer Equipment, Taylor & Francis, 2007.
- [3] G. Castanet, P. Dunand, O. Caballina, F. Lemoine, High-speed shadow imagery to characterize the size and velocity of the secondary droplets produced by drop impacts onto a heated surface, *Exp. Fluids* 54 (2013) 1489.
- [4] J.E. Dendy, Black Box multigrid, *J. Comput. Phys.* 48 (3) (1982) 366–386.
- [5] P. Dunand, G. Castanet, M. Gradeck, D. Maillet, F. Lemoine, Energy balance of droplets impinging onto a wall heated above the Leidenfrost temperature, *Int. J. Heat Fluid Flow* 44 (2013) 170–180.
- [6] B. Duret, G. Luret, J. Reveillon, T. Menard, A. Berlemon, F.-X. Demoulin, DNS analysis of turbulent mixing in two-phase flows, *Int. J. Multiph. Flow* 40 (2012) 93–105.
- [7] A. Esmaeili, G. Tryggvason, A front tracking method for computations of boiling in complex geometries, *Int. J. Multiph. Flow* 30 (2004) 1037–1050.
- [8] R. Fedkiw, T. Aslam, B. Merriman, S. Osher, A non-oscillatory Eulerian approach to interfaces in multimaterial flows (The Ghost Fluid Method), *J. Comput. Phys.* 152 (1999) 457–492.
- [9] Y. Ge, L.S. Fan, Three-dimensional simulation of impingement of a liquid droplet on a flat surface in the Leidenfrost regime, *Phys. Fluids* 17 (2005) 027104.
- [10] Y. Ge, L.S. Fan, Three-dimensional direct numerical simulation for film-boiling contact of moving particle and liquid droplet, *Phys. Fluids* 18 (2006) 117104.
- [11] F. Gibou, R. Fedkiw, L.T. Chieng, M. Kang, A second-order-accurate symmetric discretization of the Poisson equation on irregular domains, *J. Comput. Phys.* 176 (2002) 205–227.
- [12] F. Gibou, R. Fedkiw, A fourth order accurate discretization for the Laplace and heat equations on arbitrary domains, with applications to the Stefan problem, *J. Comput. Phys.* 202 (2003) 577–601.
- [13] F. Gibou, L. Chen, D. Nguyen, S. Banerjee, A level set based sharp interface method for the multiphase incompressible Navier–Stokes equations with phase change, *J. Comput. Phys.* 222 (2007) 536–555.
- [14] F. Gibou, C. Min, R. Fedkiw, High resolution sharp computational methods for elliptic and parabolic problems in complex geometries, *J. Sci. Comput.* 54 (2013) 369–413.
- [15] S. Hardt, F. Wondra, Evaporation model for interfacial flows based on a continuum-field representation of the source terms, *J. Comput. Phys.* 227 (2008) 5871–5895.
- [16] D.J.E. Harvie, D.F. Fletcher, A hydrodynamic and thermodynamic simulation of droplet impacts on hot surfaces, part 1: theoretical model, *Int. J. Heat Mass Transf.* 44 (2001) 2633–2642.
- [17] D.J.E. Harvie, D.F. Fletcher, A hydrodynamic and thermodynamic simulation of droplet impacts on hot surfaces, part 2: validation and application, *Int. J. Heat Mass Transf.* 44 (2001) 2643–2659.
- [18] R.W. Houim, K.K. Kuo, A ghost fluid method for compressible reacting flows with phase change, *J. Comput. Phys.* 235 (2013) 865–900.
- [19] M. Ishii, T. Hibiki, Thermo-Fluid Dynamics of Two-Phase Flows, Springer, 2011.
- [20] G.S. Jiang, C.W. Shu, Efficient implementation of weighted essentially non-oscillatory schemes, *J. Comput. Phys.* 126 (1996) 202–228.
- [21] D. Juric, G. Tryggvason, Computations of boiling flows, *Int. J. Multiph. Flow* 24 (3) (1998) 387–410.
- [22] M. Kang, R. Fedkiw, X.-D. Liu, A boundary condition capturing method for multiphase incompressible flow, *J. Sci. Comput.* 15 (2000) 323–360.
- [23] C. Kunkelmann, P. Stephan, Numerical simulation of the transient heat transfer during nucleate boiling of refrigerant HFE-7100, *Int. J. Fluid Refrig.* 33 (2010) 1221–1228.
- [24] B. Lalanne, S. Tanguy, F. Risso, Effect of rising motion on the damped shape oscillations of drops and bubbles, *Phys. Fluids* 25 (2013) 112107.
- [25] B. Lalanne, L. Rueda-Villegas, S. Tanguy, F. Risso, On the computation of viscous terms for incompressible two-phase flows with Level Set/Ghost Fluid Method, *J. Comput. Phys.* 301 (2015) 289–307.
- [26] J. Lee, G. Son, A level-set method for analysis of particle motion in an evaporating microdroplet, *Numer. Heat Transf., Part B* 67 (2015) 25–46.
- [27] X.-D. Liu, R. Fedkiw, M. Kang, A boundary condition capturing method for Poisson's equation on irregular domains, *J. Comput. Phys.* 160 (2000) 151–178.
- [28] C. Ma, D. Bothe, Numerical modeling of thermo-capillary two-phase flows with evaporation using a two-scalar approach for heat transfer, *J. Comput. Phys.* 233 (2013) 552–573.
- [29] S.P. MacLachlan, J.M. Tang, C. Vuik, Fast and robust solvers for pressure-correction in bubbly flows problems, *J. Comput. Phys.* 227 (2008) 9742–9761.
- [30] C. Min, F. Gibou, Geometric integration over irregular domain with application to level-set methods, *J. Comput. Phys.* 226 (2007) 1432–1443.
- [31] C. Min, F. Gibou, Robust second order accurate discretization of the multi-dimensional Heaviside and Dirac delta functions, *J. Comput. Phys.* 227 (2008) 9686–9695.
- [32] D. Nguyen, R. Fedkiw, M. Kang, A boundary condition capturing method for incompressible flame discontinuities, *J. Comput. Phys.* 172 (2001) 71–98.
- [33] N. Nikolopoulos, A. Theodorakos, G. Bergeles, A numerical investigation of the evaporation process of a liquid droplet impinging onto a hot substrate, *Int. J. Heat Mass Transf.* 50 (2006) 303–319.
- [34] S. Osher, J.A. Sethian, Fronts propagating with curvature-dependent speed: algorithms based on Hamilton–Jacobi formulations, *J. Comput. Phys.* 79 (1988) 12–49.
- [35] J. Papac, F. Gibou, C. Ratsch, Efficient symmetric discretization for the Poisson, heat and Stefan-type problems with Robin boundary conditions, *J. Comput. Phys.* 229 (2010) 875–889.
- [36] M. Renksizbulut, R. Nafziger, X. Li, A mass transfer correlation for droplet evaporation in high-temperature flows, *Chem. Eng. Sci.* 46 (9) (1991) 2351–2358.
- [37] Y. Sato, B. Niceno, A sharp-interface phase change model for a mass-conservative interface tracking method, *J. Comput. Phys.* 249 (2013) 127–161.
- [38] R. Scardovelli, S. Zaleski, Direct numerical simulation of free-surface and interfacial flow, *Annu. Rev. Fluid Mech.* 31 (1999) 567–603.
- [39] J. Schlottke, B. Weigand, Direct numerical simulation of evaporating droplets, *J. Comput. Phys.* 227 (2008) 5215–5237.
- [40] G. Son, V.K. Dhir, Numerical simulation of film boiling near critical pressures with a level set method, *J. Heat Transf.* 120 (1998).
- [41] G. Son, V.K. Dhir, N. Ramanujan, Dynamics and heat transfer associated with a single bubble during nucleate boiling on a horizontal surface, *J. Heat Transf.* 121 (1999) 623–631.
- [42] D.B. Spalding, Combustion and Mass Transfer, Pergamon, 1979.
- [43] M. Sussman, P. Smereka, S. Osher, A level set approach for computing solutions to incompressible two-phase flow, *J. Comput. Phys.* 114 (1994) 146–159.
- [44] M. Sussman, K.M. Smith, M.Y. Hussaini, M. Ohta, R. Zhi-Wei, A sharp interface method for incompressible two-phase flows, *J. Comput. Phys.* 221 (2007) 469–505.
- [45] S. Tanguy, T. Menard, A. Berlemon, A level set method for vaporizing two-phase flows, *J. Comput. Phys.* 221 (2007) 837–853.
- [46] S. Tanguy, M. Sagan, B. Lalanne, F. Couderc, C. Colin, Benchmarks and numerical methods for the simulation of boiling flows, *J. Comput. Phys.* 264 (2014) 1–22.
- [47] S. Welch, J. Wilson, A volume of fluid based method for fluid flows with phase change, *J. Comput. Phys.* 160 (2000) 662–682.