Finite-element method based solver for an evaporating sessile droplet on a heated surface

MANISH KUMAR[1,2] and RAJNEESH BHARDWAJ[1,*]

[1]Department of Mechanical Engineering, Indian Institute of Technology Bombay, Mumbai 400076, India
[2]TIPs Laboratory, Université Libre de Bruxelles, Brussels 1050, Belgium
e-mail: rajneesh.bhardwaj@iitb.ac.in

Abstract. We report the development of a finite-element-based solver to compute transport of mass, momentum and energy during evaporation of a sessile droplet on a heated surface. The evaporation is assumed to be quasi-steady and diffusion-limited. The heat transfer between the droplet and substrate and mass transfer of liquid-vapor are solved using a two-way coupling. In particular, here, we develop and implement the formulation of fluid flow inside the droplet in the model. The continuity and Navier–Stokes equations are solved in axisymmetric, cylindrical coordinates. Jump velocity boundary condition is applied on the liquid–gas interface using the evaporation mass flux. The governing equations are discretized in the framework of the Galerkin weight residual approach. A mesh of finite triangular elements with six nodes is utilized, and quadratic shape functions are used to obtain the second-order accurate numerical solution. Two formulations, namely, penalty function and velocity pressure, are employed to obtain discretized equations. The numerical results are the same using both methods, and the latter is around 30–50% faster than the former for the cases of refined grid. Computed flow fields are in excellent agreement with published results. The solver's capability is demonstrated by solving the internal flow field for a case of a heated substrate.

Keywords. Droplet evaporation; finite-element method.

1. Introduction

Understanding the evaporation of a sessile droplet on a solid surface has been a much-studied problem in the last two decades [1]. It is central to several technical applications such as ink-jet printing, spray cooling, and bioassays. Recently, sessile droplet evaporation has been studied in the context of fomite transmission of COVID-19 [2–4]. The flow field inside the droplet is essential for designing the above-mentioned technical applications. In particular, self-assembly of colloidal particles is dictated by the internal flow during the evaporation [5–9]. The computational modeling of the internal fluid flow is challenging since fluid flow inside the droplet, heat transfer between the droplet and substrate, and mass transfer of liquid vapor from the droplet are coupled. The evaporation mass flux on the liquid–gas interface is non-uniform, and it influences the temperature and flow field inside the droplet.

Previous numerical studies reported evaporation dynamics of a sessile droplet on a non-heated substrate. These reports made certain assumptions for the sake of computational tractability. Girard *et al* [10] investigated Marangoni convection inside the evaporating droplet using

a commercial finite-element solver while neglecting the pressure gradient terms in the Naiver-Stokes equations. Widjaja *et al* [11] tested two meshing techniques, namely, spine and elliptic, in their FE model and reported that elliptic mesh provides more accurate results and resolves flow near the contact line. Barash *et al* [12] developed a finite-difference model to investigate evaporation of droplets and flow inside them. They compared their simulation results with experimental measurements for evaporation of toluene droplets. They reported different stages of vortices formation inside the droplet as it evaporates.

Bourantas *et al* [13] solved fluid flow inside the evaporating droplet using the velocity-correction method with meshless point collocation. They reported better accuracy and convergence rate compared to traditional mesh-based models. Using commercial FE solver, Barmi and Meinhart [14] simulated the convective flow inside the pinned evaporating droplet while assuming axisymmetric flow field, temperature, and vapor concentration distribution. They concluded that flow inside the droplet is responsible for particle deposition patterns, but it does not influence heat transfer much, and conduction remains the primary mode of heat transfer. Barash [15] investigated the dependence of vortex structures in the droplet on substrate properties using a transient Navier–Stokes solver. The

transient model solves the evaporation profile while neglecting the contribution of convection inside the droplet and uses it to calculate boundary conditions at the liquid–gas interface.

Al-Sharfi *et al* [16] used a finite element method-based software on studying flow and heat transfer characteristics inside evaporating droplets resting on hydrophilic and hydrophobic substrates. They reported strong dependence of the heat transfer rate on the droplet contact angle. Pradhan and Panigrahi [17] investigated flow inside sessile and pendant droplets of aqueous salt solution using commercial software. They showed that the orientation of the droplet strongly influences the convection inside it. Bouchenna *et al* [18] developed a finite volume model to study the effect of substrate heating on flow pattern inside the droplet with quasi-steady evaporation while accounting for thermal buoyancy and thermocapillary effect on the flow. Chen *et al* [19] reported the effect of substrate thermal properties on the bulk fluid flow inside the droplet, using a commercial finite element solver.

A brief literature survey shows that previously-reported computational methods developed in this context focused on the non-heated substrate. To the authors' knowledge, no studies have reported high-fidelity modeling of the sessile droplet evaporation on a heated surface thus far. To this end, we report a finite-element method based solver to compute diffusion-limited and quasi-steady droplet evaporation on a heated surface. In our previous work [20], we reported a finite-element method-based solver for calculating heat transfer between the droplet and substrate, and mass transfer of liquid-vapor, using a two-way coupling. In the present work, we extend this solver to account for fluid flow inside the droplet. Here we benchmark and validate the solver with previous studies and demonstrate the second-order accuracy of the solver. Lastly, the solver has been demonstrated to compute the flow field inside the evaporating droplet on a heated surface.

2. Computational model

2.1 *Assumptions and limitations of the model*

We present the computational model for quasi-steady, diffusion-limited evaporation of a sessile droplet, with a pinned contact line during the evaporation. The flow, thermal and concentration of liquid vapor fields are to be considered as axisymmetric. The assumptions in the model, justified in previous works [21–24] are briefly described as follows. The evaporation occurs by diffusion of liquid-vapor in quiescent air. The droplet's wetted diameter is lesser than the capillary length such that the droplet assumes a spherical cap during the evaporation. For the evaporation of microliter droplets, the internal flow is on the order of μm/s, and therefore, inertial terms in Navier–Stokes equation are neglected, due to small Reynolds number ($O(10^{-3})$). The

Reynolds number is based on the evaporation-induced velocity inside the droplet and wetted radius of the droplet. In the context of heat transfer, convection inside and outside of the droplet, and radiative heat transfer from the liquid–gas interface are neglected.

2.2 *Heat and mass transfer*

Using a two-way coupling, reported in our earlier work [20], we solved temperature field in the droplet and substrate, and concentration of liquid vapor outside the droplet in asymmetric cylindrical coordinates (figure 1). The details of the model and numerical methodology are given in Ref. [20] and here we briefly describe governing equations and boundary conditions used in this model. In the framework described in section 2.1, the energy equations for droplet and substrate are given by,

$$\nabla^2 T_i = 0 \qquad (1)$$

where T is temperature and, $i = 1$ and 2 correspond to the droplet and substrate, respectively.

The following heat transfer boundary conditions are applied. The substrate bottom surface is prescribed with a given temperature. The side and top surface of the substrate are insulated. A perfect thermal contact is considered at liquid–solid interface. At liquid–gas interface, the following jump energy condition is enforced [25]:

$$jL = -k\nabla T|_{LG} \cdot \mathbf{n} \qquad (2)$$

where j is the evaporative flux [kg m^{-2} s^{-1}] at the liquid–gas interface, L is the latent heat of the evaporation [J kg^{-1}], k is thermal conductivity of the droplet
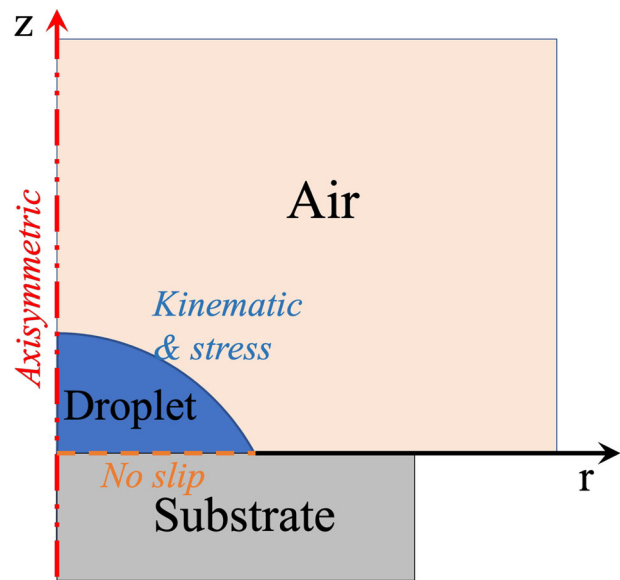

Figure 1. Computational domain and boundary conditions used in the present computational model.

[W m^{-1} K^{-1}], LG denotes liquid–gas interface and **n** is unit normal vector on the liquid–gas interface.

The concentration, c [kg m^{-3}] of liquid vapor outside droplet in ambient is governed by the following equation,

$$\nabla^2 c = 0 \tag{3}$$

The boundary conditions for mass transport of liquid vapor are described as follows. The value of c on the far-field boundaries is taken as $Hc_{sat,\infty}$, where $c_{sat,\infty}$ is the saturated concentration (kg m^{-3}) of the water vapor in the far-field and H is relative humidity. The concentration at the liquid–gas interface is prescribed at its saturated value at the interface temperature. The evaporative mass flux j at the liquid–gas interface is expressed in terms of c as follows [7, 21, 23],

$$j(r, T) = D(T)\nabla c|_{LG} \cdot \mathbf{n} \tag{4}$$

where r is radial coordinate and $D(T)$ is diffusion coefficient of liquid vapor in air as a function of temperature [7].

Eqs. 1–4 are solved using finite-element method using two-way coupling, as described in our previous work [20]. The thermal and concentration field are obtained after a successful convergence of the coupling. Using computed j, evaporation rate $\dot{m}$ is computed by integrating j over area of liquid–gas interface by the following expression:

$$\dot{m} = \rho \frac{\partial V}{\partial t} = \int_A 2\pi r j ds \tag{5}$$

where ρ is droplet density, V is instantaneous droplet volume and A is area of liquid–gas interface. j and $\dot{m}$ are used while applying boundary conditions of flow solver, as described in next section. The modules of the solver are schematically shown in figure 2.

2.3 *Fluid flow in the droplet*

In the framework described in section 2.1, the governing equation of the fluid flow inside the droplet are continuity and Navier–Stokes equations. The equations are written in axisymmetric cylindrical coordinates (r, z) as follows:

$$\frac{1}{r}\frac{\partial (ru)}{\partial r} + \frac{\partial v}{\partial z} = 0 \tag{6}$$
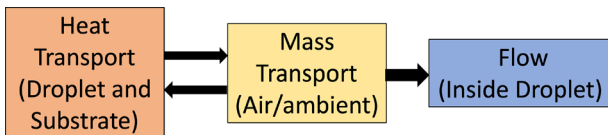

Figure 2. Different modules of the solver and data exchange between them.

$$\frac{1}{r}\frac{\partial (r\sigma_{rr})}{\partial r} + \frac{\partial \sigma_{rz}}{\partial z} - \frac{\sigma_{\theta\theta}}{r} = 0 \tag{7}$$

$$\frac{1}{r}\frac{\partial (r\sigma_{zr})}{\partial r} + \frac{\partial \sigma_{zz}}{\partial z} = 0 \tag{8}$$

where the components of the stress tensor are defined as follows:

$$\sigma_{rr} = 2\mu \frac{\partial u}{\partial r} - P \tag{9}$$

$$\sigma_{zz} = 2\mu \frac{\partial v}{\partial z} - P \tag{10}$$

$$\sigma_{\theta\theta} = 2\mu \frac{u}{r} - P \tag{11}$$

$$\sigma_{rz} = \sigma_{zr} = \mu \left(\frac{\partial v}{\partial r} + \frac{\partial u}{\partial z}\right) \tag{12}$$

The boundary conditions for the fluid flow (figure 1) are expressed as follows: At liquid–solid interface ($z = 0$, $r \leq R$), no slip boundary condition ($u = 0$, $v = 0$) is applied. At $r = 0$, axisymmetric boundary condition is applied: $u = 0$ and $\partial v / \partial r = 0$. At liquid–gas interface, two boundary conditions are applied: (a) zero shear stress boundary condition in tangential direction to the interface ($\tau_t = 0$) (b) Jump velocity or kinematic boundary condition, given by [26],

$$u_n = \frac{j}{\rho} - n_z \frac{\partial h}{\partial t} \tag{13}$$

where u_n is fluid velocity at liquid–gas interface, normal to the interface, $\partial h / \partial t$ is the velocity of the interface in axial direction and n_z is component of unit normal vector **n** in axial direction. The radial and axial components (n_r and n_z) of the normal vector are defined as follows [26],

$$\begin{aligned} n_r &= \frac{r \sin \theta}{R} \\ n_z &= \sqrt{1 - (n_r)^2} \end{aligned} \tag{14}$$

where θ and R are contact angle and wetted radius, respectively. The volume of the spherical cap droplet as a function of droplet height H is given by,

$$V = \frac{\pi h}{6}(3R^2 + H^2) \tag{15}$$

We consider a pinned contact line or constant contact radius (CCR) mode of evaporation. The velocity at the top ($r = 0$) of the droplet (dH/dt) is obtained by differentiating eq. 15 with respect to t, keeping R as constant [14]. After differentiation and using eq. 5, we obtain dH/dt as follows,

$$\frac{dH}{dt} = \frac{2\dot{m}}{\rho\pi(R^2 + H^2)} \tag{16}$$

The height of liquid–gas interface $h(r, t)$ is defined as follows [21],

$$h(r,t) = \sqrt{\left(\frac{R^2}{\sin^2\theta} - r^2\right)} - \frac{R}{\tan\theta} \tag{17}$$

Differentiating eq. 17 and using definition of contact angle, $\tan(\theta/2) = H/R$, we obtain expression of $\partial h/\partial t$ as follows [26]:

$$\frac{\partial h}{\partial t} = \frac{dH}{dt}\left[\frac{H^4 - R^4}{4H^3\sqrt{\left(\frac{H^2+R^2}{2H}\right)^2 - r^2}} + \frac{R^2 + H^2}{2H^2}\right] \tag{18}$$

In case of constant contact angle (CCA) mode of evaporation, V in eq. 15 is expressed in terms of θ and H and eq. 15 is differentiated, keeping θ as constant. In case of a mixed mode of evaporation, say CCR mode followed by CCA mode, formulations of both CCA and CCR can be combined together. In this case, receding contact angle θ_{rec}, the contact angle at which contact line recedes, may be available from companion experiment.

2.4 *FEM formulation*

We implement Galerkin weighted residual approach, in which, weight function (w_i) are the same as shape function or approximation function [27]. Using weight function, the following weak form of Navier–Stokes equation was obtained to lower the requirement of choosing higher-order approximation function [27]:

$$\int_{\Omega_e} w_i\left[\frac{1}{r}\frac{\partial(r\sigma_{rr})}{\partial r} + \frac{\partial\sigma_{rz}}{\partial z} - \frac{\sigma_{\theta\theta}}{r}\right]d\Omega = 0 \tag{19}$$

$$\int_{\Omega_e} w_i\left[\frac{1}{r}\frac{\partial(r\sigma_{zr})}{\partial r} + \frac{\partial\sigma_{zz}}{\partial z}\right]d\Omega = 0 \tag{20}$$

where $d\Omega = 2\pi r dr dz$ is differential volume in axisymmetric cylindrical coordinates and Ω_e represents a finite-element. After solving and rearranging differential terms in integrals in Eqs. 19–20, we get the following equations:

$$\int_{\Omega_e}\left[\frac{\partial w_i(\sigma_{rr})}{\partial r} + \frac{\partial w_i\sigma_{rz}}{\partial z} + \frac{w_i\sigma_{\theta\theta}}{r}\right]rdrdz$$
$$= \int_{\Gamma_e} w_i(t_r)rds \tag{21}$$

$$\int_{\Omega_e}\left[\frac{\partial w_i(\sigma_{zr})}{\partial r} + \frac{\partial w_i\sigma_{zz}}{\partial z}\right]rdrdz = \int_{\Gamma_e} w_i(t_z)rds \tag{22}$$

where $t_r = \sigma_{rr}n_r + \sigma_{rz}n_z$ and $t_z = \sigma_{zr}n_r + \sigma_{zz}n_z$ are the boundary stress components in radial and axial directions, respectively. Γ_e represents the boundary of the element and differential surface area ($d\Gamma$) of the axisymmetric geometry is $2\pi rds$, where ds is differential arc length of the boundary or liquid–gas interface.

2.4.1 *Penalty function formulation* In penalty function formulation, pressure (P) is eliminated from Navier–Stokes equations [27]. The elimination of pressure leads to a constrained problem and where constrained equation is the continuity equation (eq. 6). The constrained problem is reformulated as an unconstrained problem and pressure P is replaced by following function [27]:

$$P = -\gamma\left(\frac{\partial u}{\partial r} + \frac{\partial v}{\partial z} + \frac{u}{r}\right) \tag{23}$$

where γ is a penalty factor which should be a large arbitrary value [27], i.e. $\mu \times 10^{10}$. After calculating velocities, pressure (P) is recovered using eq. 23. Substituting σ_{rr}, σ_{zz}, $\sigma_{\theta\theta}$, σ_{rz} and P from eq. 9, 10, 11, 12 and 23, respectively, in eqs. 21 and 22, the following equations are obtained:

$$\int_{\Omega_e}\left[2\mu\frac{\partial w_i}{\partial r}\frac{\partial u}{\partial r} + \mu\frac{\partial w_i}{\partial z}\left(\frac{\partial u}{\partial z} + \frac{\partial v}{\partial r}\right) + 2\mu\frac{w_i}{r}\frac{u}{r}\right.$$
$$+ \gamma\frac{\partial w_i}{\partial r}\left(\frac{\partial u}{\partial r} + \frac{\partial v}{\partial z} + \frac{u}{r}\right)$$
$$\left. + \gamma\frac{w_i}{r}\left(\frac{\partial u}{\partial r} + \frac{\partial v}{\partial z} + \frac{u}{r}\right)\right]rdrdz = \int_{\Gamma_e} w_i(t_r)rds \tag{24}$$

$$\int_{\Omega_e}\left[\mu\frac{\partial w_i}{\partial r}\frac{\partial u}{\partial z} + \mu\frac{\partial w_i}{\partial r}\frac{\partial v}{\partial r} + 2\mu\frac{\partial w_i}{\partial z}\frac{\partial v}{\partial z}\right.$$
$$\left. + \gamma\frac{\partial w_i}{\partial z}\left(\frac{\partial u}{\partial r} + \frac{\partial v}{\partial z} + \frac{u}{r}\right)\right]rdrdz \tag{25}$$
$$= \int_{\Gamma_e} w_i(t_z)rds$$

To discretize continuous eqs. 24–25 within each finite element, the velocities (u, v) are approximated by the following trial solution:

$$u = \sum_{j=1}^{n}\psi_j(r,z)u_j$$
$$v = \sum_{j=1}^{n}\psi_j(r,z)v_j \tag{26}$$

where u_j and v_j are the radial and axial velocity, respectively, at the nodal points of the element. n and ψ_j are the number of nodes in the element and shape function, respectively. As mentioned earlier, in Galerkin weighted residual approach, weight functions are same as shape functions, therefore, $w_i = \psi_i$. Substituting trial solution and

weighting function, we write eq. 24 and 25 in compact matrix notation as follows [27]:

$$
\begin{aligned}
(2\mu[S^{rr}] &+ \mu[S^{zz}] + 2\mu[M] \\
&+ \gamma([S^{rr}] + [S^{ro}] + [S^{or}] + [M]))\{u_j\} \\
&+ (\mu[S^{zr}] + \gamma[S^{zr}] + \gamma[S^{oz}])\{v_j\} = \{F^r\}
\end{aligned}
\tag{27}
$$

$$
\begin{aligned}
(\mu[S^{rz}] &+ \gamma[S^{zr}] + \gamma[S^{zo}])\{u_j\} \\
&+ (\mu[S^{rr}] + 2\mu[S^{zz}] + \gamma[S^{zz}])\{v_j\} = \{F^z\}
\end{aligned}
\tag{28}
$$

Eqs. 27–28 are further simplified in matrix notation as follows:

$$
\begin{bmatrix} [K^{rr}] & [K^{rz}] \\ [K^{zr}] & [K^{zz}] \end{bmatrix} \begin{Bmatrix} \{u_j\} \\ \{v_j\} \end{Bmatrix} = \begin{Bmatrix} \{F^r\} \\ \{F^z\} \end{Bmatrix}
\tag{29}
$$

where,

$$
\begin{aligned}
[K^{rr}] &= 2\mu[S^{rr}] + \mu[S^{zz}] + 2\mu[M] \\
&\quad + \gamma([S^{rr}] + [S^{ro}] + [S^{or}] + [M]) \\
[K^{rz}] &= \mu[S^{zr}] + \gamma[S^{rz}] + \gamma[S^{oz}] \\
[K^{zr}] &= \mu[S^{rz}] + \gamma[S^{zr}] + \gamma[S^{zo}] \\
[K^{zz}] &= \mu[S^{rr}] + 2\mu[S^{zz}] + \gamma[S^{zz}]
\end{aligned}
\tag{30}
$$

The element coefficient matrices are defined as follows:

$$
M_{ij} = \int_{\Omega_e} \frac{\psi_i \psi_j}{r} dr dz
\tag{31}
$$

$$
S_{ij}^{rr} = \int_{\Omega_e} \frac{\partial \psi_i}{\partial r}\frac{\partial \psi_j}{\partial r} r dr dz \quad S_{ij}^{rz} = \int_{\Omega_e} \frac{\partial \psi_i}{\partial r}\frac{\partial \psi_j}{\partial z} r dr dz
\tag{32}
$$

$$
S_{ij}^{zr} = \int_{\Omega_e} \frac{\partial \psi_i}{\partial z}\frac{\partial \psi_j}{\partial r} r dr dz \quad S_{ij}^{zz} = \int_{\Omega_e} \frac{\partial \psi_i}{\partial z}\frac{\partial \psi_j}{\partial z} r dr dz
\tag{33}
$$

$$
S_{ij}^{ro} = \int_{\Omega_e} \frac{\partial \psi_i}{\partial r}\psi_j dr dz \quad S_{ij}^{or} = \int_{\Omega_e} \psi_i \frac{\partial \psi_j}{\partial r} dr dz
\tag{34}
$$

$$
S_{ij}^{zo} = \int_{\Omega_e} \frac{\partial \psi_i}{\partial z}\psi_j dr dz \quad S_{ij}^{oz} = \int_{\Omega_e} \psi_i \frac{\partial \psi_j}{\partial z} dr dz
\tag{35}
$$

$$
F_i^r = \int_{\Gamma_e} \psi_i(t_r) rds \quad F_i^z = \int_{\Gamma_e} \psi_i(t_z) rds
\tag{36}
$$

2.4.2 *Velocity-pressure formulation* In velocity-pressure formulation, a natural and direct formulation, the weak form of continuity equation is obtained using weight function, one order less than used for Navier–Stokes equations [27]. Both Navier–Stokes equation physically represents force, hence same weighting function can be used. However, continuity equation represents the volume change. Volume change occur under the action of

hydrostatic pressure, hence weight function (w_{2k}) for continuity, equation should like the pressure (P) or the shape function of pressure. The weak form of the continuity equation can be written as follows [27]:

$$
-\int_{\Omega_e} w_{2k}\left[\frac{\partial u}{\partial r} + \frac{\partial v}{\partial z} + \frac{u}{r}\right] r dr dz = 0
\tag{37}
$$

The minus sign is inserted to make the resulting finite element model symmetric. Equation 37 is rewritten as follows,

$$
-\int_{\Omega_e} \left[w_{2k}\frac{\partial u}{\partial r} + w_{2k}\frac{\partial v}{\partial z} + w_{2k}\frac{\partial u}{\partial r}\right] r dr dz = 0
\tag{38}
$$

Substituting components of stress tensor from eqs. 9, 10, 11 and 12 in eqs. 21 and 22, the following equations are obtained:

$$
\begin{aligned}
\int_{\Omega_e} &\left[2\mu\frac{\partial w_i}{\partial r}\frac{\partial u}{\partial r} + \mu\frac{\partial w_i}{\partial z}\left(\frac{\partial u}{\partial z} + \frac{\partial v}{\partial r}\right) \right. \\
&\left. +2\mu\frac{w_i}{r}\frac{u}{r} - \frac{\partial w_i}{\partial r}P - \frac{w_i}{r}P\right] r dr dz \\
&= \int_{\Gamma_e} w_i(t_r) rds
\end{aligned}
\tag{39}
$$

$$
\begin{aligned}
\int_{\Omega_e} &\left[\mu\frac{\partial w_i}{\partial r}\frac{\partial u}{\partial z} + \mu\frac{\partial w_i}{\partial r}\frac{\partial v}{\partial r} \right. \\
&\left. +2\mu\frac{\partial w_i}{\partial z}\frac{\partial v}{\partial z} - \frac{\partial w_i}{\partial z}P\right] r dr dz = \int_{\Gamma_e} w_i(t_z) rds
\end{aligned}
\tag{40}
$$

To discretize continuous eqs. 39–40 within each finite element, the velocities (u, v) are approximated by the trial solution defined in eq. 26 and pressure (P) is approximated by following trial solution, that is one order less than used for the velocities:

$$
P = \sum_{l=1}^{m} \phi_l(r, z)P_l
\tag{41}
$$

where P_l is the pressure at the corner nodal points of the element. m and ϕ_l are the number of corner nodes in the element and shape functions respectively. As explained before, in Galerkin weighted residual approach, $w_i = \psi_i$ and $w_{2k} = \phi_l$. On putting trial solution and weighting function, we can write eqs. 39, 40 and 38 in compact matrix notation as follows:

$$
\begin{aligned}
(2\mu[S^{rr}] &+ \mu[S^{zz}] + 2\mu[M])\{u_j\} + (\mu[S^{zr}])\{v_j\} \\
&- ([G^{ro}] + [G^{oo}])\{P_l\} = \{F^r\}
\end{aligned}
\tag{42}
$$

$$
\begin{aligned}
(\mu[S^{rz}])\{u_j\} &+ (\mu[S^{rr}] + 2\mu[S^{zz}])\{v_j\} \\
&- ([G^{zo}])\{P_l\} = \{F^z\}
\end{aligned}
\tag{43}
$$

$$- \left([G^{ro}]^T + [G^{oo}]^T\right)\{u_j\} - \left([G^{zo}]^T\right)\{v_j\} = 0 \qquad (44)$$

Further, we can write above three equations in more simplified way in matrix form as follows:

$$\begin{bmatrix} [K^{rr}] & [K^{rz}] & [K^{ro}] \\ [K^{zr}] & [K^{zz}] & [K^{zo}] \\ [K^{or}] & [K^{oz}] & [0] \end{bmatrix} \begin{Bmatrix} \{u\} \\ \{v\} \\ \{P\} \end{Bmatrix} = \begin{Bmatrix} \{F^r\} \\ \{F^z\} \\ \{0\} \end{Bmatrix} \qquad (45)$$

where,

$$\begin{aligned} [K^{rr}] &= 2\mu[S^{rr}] + \mu[S^{zz}] + 2\mu[M] \\ [K^{rz}] &= \mu[S^{zr}] \\ [K^{ro}] &= -([G^{ro}] + [G^{oo}]) \\ [K^{zr}] &= \mu[S^{rz}] \\ [K^{zz}] &= \mu[S^{rr}] + 2\mu[S^{zz}] \\ [K^{zo}] &= -[G^{zo}] \\ [K^{or}] &= -\left([G^{ro}]^T + [G^{oo}]^T\right) \\ [K^{oz}] &= -[G^{zo}]^T \end{aligned} \qquad (46)$$

where the $[S]$, $[M]$ and $\{F\}$ element coefficient matrices are defined as before from eq. 31 to eq. 36. The remaining element coefficient matrices are defined as follows:

$$G^{ro}_{kl} = \int_{\Omega_e} \frac{\partial \psi_k}{\partial r} \phi_l r \, dr \, dz \qquad (47)$$

$$G^{zo}_{kl} = \int_{\Omega_e} \frac{\partial \psi_k}{\partial z} \phi_l r \, dr \, dz \qquad (48)$$

$$G^{oo}_{kl} = \int_{\Omega_e} \psi_k \phi_l \, dr \, dz \qquad (49)$$

2.4.3 *Mesh generation and evaluation of element coefficient matrices* A six node triangular element (figure 3a) was chosen over three-node element as it improves the solution accuracy and reduce the overall solution time by reducing the number element required. Quadratic and linear shape functions were used to compute velocity and pressure field, respectively. A three-node triangular mesh inside the droplet domain was generated using Matlab® function 'initmesh'. To generate six-node mesh, we developed an in-house code, which places a node at the exact midpoint of the each side of triangular element and updates the point matrix, element connectivity matrix, boundary matrix. The mesh generated using MATLAB function 'initmesh' and modified using in-house code is shown in figure 3(b).

In order to evaluate the integrals of element coefficient matrices, numerical integration was employed. To do numerical integration, first we need to map real element in (r, z) coordinates to a master or parent element in generic (ξ, η) coordinates over which integration will performed using Gaussian quadrature points. As in our real element all three nodes located at side of triangle are located at mid-point of each triangle side, so it is sufficient to use linear shape function to do coordinate transformation. The following expression facilitates the (r, z) to (ξ, η) coordinate transformation:

$$r = \sum_{k=1}^{3} r_k \phi_k(\xi, \eta) \quad z = \sum_{k=1}^{3} z_k \phi_k(\xi, \eta) \qquad (50)$$

where r_k and z_k is k_{th} radial and axial coordinates of corner nodes of the triangle element. Shape functions ϕ_k can be defined as follows:

$$\begin{aligned} \phi_1 &= \xi \\ \phi_2 &= \eta \\ \phi_3 &= 1 - \eta - \xi \end{aligned} \qquad (51)$$

To evaluate integrals, the following expression is required to map infinitesimal area of real element to corresponding area of parent element:
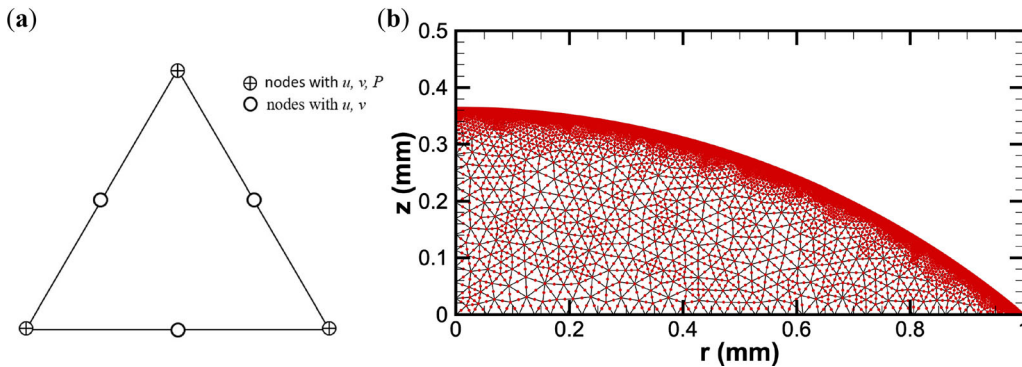

Figure 3. (a) Six-node triangular element. (b) Six-node mesh in droplet domain and red dots represent nodes.

$$drdz = \det(Jc)d\xi d\eta \tag{52}$$

where $\det(Jc)$ is the determinant of the Jacobian (Jc) for the element under consideration and is defined as follows:

$$[Jc] = \begin{bmatrix} \dfrac{\partial r}{\partial \xi} & \dfrac{\partial z}{\partial \xi} \\ \dfrac{\partial r}{\partial \eta} & \dfrac{\partial z}{\partial \eta} \end{bmatrix} \tag{53}$$

For linear mapping $\det(Jc)$ is equal to 2 times of the area of triangle in consideration. Next, we need to convert partial derivative of (r, z) into derivatives of (ξ, η) which is done using the following relationship:

$$\left\{ \begin{array}{c} \dfrac{\partial \psi_i}{\partial r} \\ \dfrac{\partial \psi_i}{\partial z} \end{array} \right\} = [Jc]^{-1} \left\{ \begin{array}{c} \dfrac{\partial \psi_i}{\partial \xi} \\ \dfrac{\partial \psi_i}{\partial \eta} \end{array} \right\} \tag{54}$$

where shape function ψ_i is quadratic shape function for a six-node triangle element as mentioned before. Quadratic shape function is defined as follows:

$$\begin{aligned} \psi_1 &= 2\xi^2 - \xi \\ \psi_2 &= 2\eta^2 - \eta \\ \psi_3 &= 1 - 3\xi - 3\eta + 4\xi\eta + 2\xi^2 + 2\eta^2 \\ \psi_4 &= 4\xi\eta \\ \psi_5 &= 4(\eta - \xi\eta - \eta^2) \\ \psi_6 &= 4(\xi - \xi\eta - \xi^2) \end{aligned} \tag{55}$$

Using eqs. 52, 54 and 55, the integrals specified in equations from eqs. 31 to 35 can be converted to (ξ, η) coordinates and numerical integration can be performed using Gaussian quadrature as follows:

$$\int_{\Omega_e} f(\xi, \eta)d\xi d\eta \cong \frac{1}{2}\sum_{l=1}^{gp} w_l f(\xi_l, \eta_l) \tag{56}$$

where $f(\xi, \eta)$ is any function in (ξ, η) coordinates and (ξ_l, η_l) are the Gaussian points [27]. gp is the number of Gaussian points used to evaluate the integral. Three and one-point Gaussian quadrature was used to evaluate integrals.

To calculate coefficient of force vector on the right hand side of eq. 29 or to apply stress boundary condition on the boundary of the domain, we need to transform the boundary or open side of the triangle to one dimensional line coordinate (s) whose origin at first point of boundary. As we are using six-node or quadratic triangular element, we have three nodes on each side of the triangle, so we need to use

quadratic shape function (ψ_{bi}). These three shape functions (ψ_{bi}) can be defined as follows:

$$\begin{aligned} \psi_{b1} &= \left(1 - \frac{s}{h}\right)\left(1 - \frac{2s}{h}\right) \\ \psi_{b2} &= \frac{4s}{h}\left(1 - \frac{s}{h}\right) \\ \psi_{b3} &= -\frac{s}{h}\left(1 - \frac{2s}{h}\right) \end{aligned} \tag{57}$$

where h is the length of the side or boundary of the element. The elements which are inside the domain, for them, the value of coefficients in force vector are need not to be calculated as the force acting on the side or boundary of the neighboring elements cancels out each other. Therefore, we need to calculate coefficient of force vector only at the boundary of elements that are located at boundary of the domain and some external force or stress is applied.

2.4.4 *Liquid–gas interface boundary conditions*   The boundary conditions at liquid–gas interface, as described in section 2.3, are applied using the following approach. As the shape of droplet is spherical cap, the liquid–gas interface is not parallel to r or z axis. To apply the both boundary conditions precisely, coordinate rotation must be done for the nodes at the liquid–gas interface of the droplet [27], which can be done with the help of rotation matrix [Q]. Equation 29 can be written in more compact form as follows:

$$[K]\{u\} = \{F\} \tag{58}$$

To perform the coordinate rotation on the nodes at the liquid–gas interface of the droplet following mathematical operation can be done:

$$[Q]^T[K][Q]\{u\} = \{F\} \tag{59}$$

where rotation matrix [Q] can be defined as follows:

$$[Q] = \begin{bmatrix} \ddots & \cdots & \cdots & \cdots \\ \cdots & n_{r,i} & n_{z,i} & \cdots \\ \cdots & -n_{z,i} & n_{r,i} & \cdots \\ \cdots & \cdots & \cdots & \ddots \end{bmatrix} \tag{60}$$

where $n_{r,i}$ and $n_{z,i}$ are the normal vector in radial and axial direction for i_{th} node on the liquid–gas interface.

2.4.5 *Algorithm*   The algorithm was implemented in Matlab® [28] and the flow chart of the algorithm is shown in figure 4. The sequential steps in the algorithm are as follows:
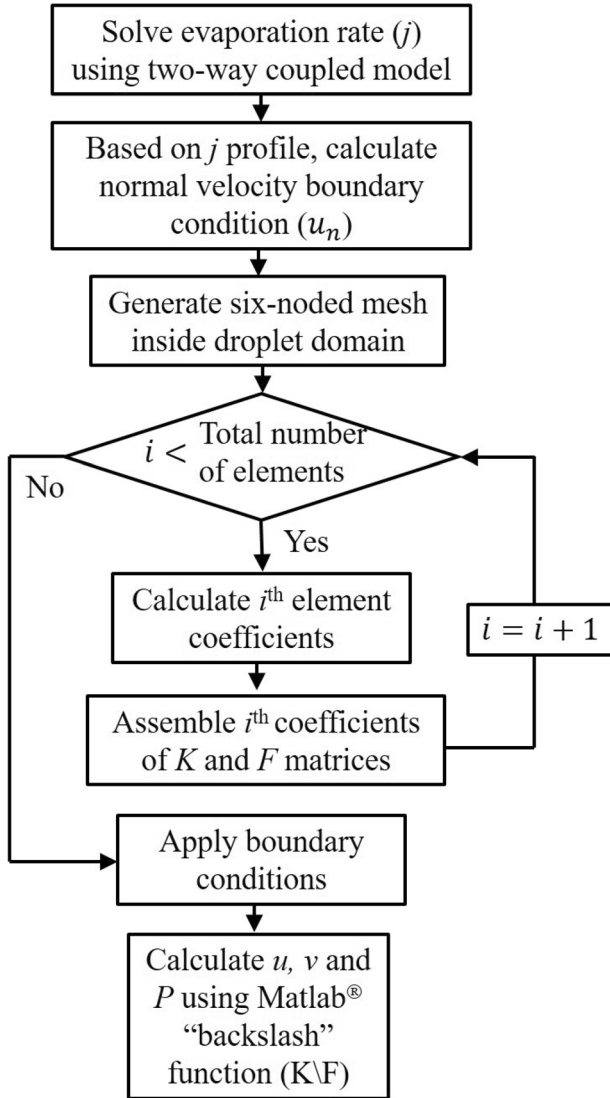
Figure 4. Flow chart of the solver.

1. Solve the evaporation mass flux on liquid–gas interface (j) by a two-way coupling [20] between heat transport inside droplet-substrate and mass transport of liquid vapor in surrounding air (figure 2).
2. Calculate $\dot{m}$, dH/dt and $\partial h/\partial t$ using eqs. 5, 16 and 18, respectively.
3. Calculate velocity boundary condition at liquid–gas interface using eq. 13 and data obtained in steps 1 and 2.
4. Generate six node mesh inside the droplet domain.
5. Calculate element coefficients defined in section 2.4.1 or section 2.4.2 using numerical integration technique explained in section 2.4.3 for six node triangular elements.
6. Assemble K and F matrices given by eq. 29 or eq. 45 for penalty function or Velocity-pressure formulation, respectively.

Table 1. Thermophysical properties and parameters used in the present study

Properties/parameters	Value
Viscosity of water	8.9×10^{-4} Pa s
Density of water	1000 kg m^{-3}
Latent heat of evaporation of water	2.2635×10^6 J kg^{-1}
Thermal conductivity of water	0.61 w m^{-1}K^{-1}
Thermal conductivity of copper	401 w m^{-1}K^{-1}
Saturated water vapor concentration (c_{sat}) at 25°C	0.023 kg m^{-3}
Diffusion coefficient of water vapor (D) at 25°C	2.6×10^{-5} m^2 s^{-1}
Variation of D and c_{sat} with temperature	Taken from Ref. [20]
Ambient temperature	25°C
Humidity	0.35
Contact angle	40°
Wetted radius	1 mm
Substrate width	5 mm
Substrate thickness	0.1 mm
Substrate temperature (non-heated case)	25°C
Substrate temperature (heated case)	40°C, 55°C, 70°C

7. Apply boundary conditions on the boundary nodes in the assembled matrices (section 2.4.4).
8. Solve system of equations ($Ax = b$) for u, v and P using inbuilt solver backslash (\) in Matlab®.

3. Testing of the solver

We have used two formulations (Penalty function formulation and Velocity-pressure formulation) to solve flow field inside the droplet. Here, we are presenting validation for these both formulations. We consider evaporation of a microliter water droplet on a heated and non-heated copper substrate. The values of the thermophysical properties and parameters used in the present study are given in table 1.

3.1 *Grid-size independence test*

In order to achieve grid size convergence, we test six grids with respect to most refined grid using penalty function formulation. Number of nodes in computational domain of droplet are listed in table 2. We compute L2-norm error (ϵ) with respect to the most refined mesh is expressed as follows:

$$\epsilon = \left[\frac{1}{N^2} \sum_{i=1}^{N} \left(u_i^N - u_i^{N_{max}} \right) \right]^{1/2} \tag{61}$$

where i represents the indices of grid points of the mesh and N is the number of nodes in the considered mesh. N_{max} is

Table 2. Number of nodes in different grids used in grid-size convergence study

S. No.	Nodes in droplet	Minimum grid size (mm)
Grid 1	728	4.0×10^{-2}
Grid 2	1348	2.0×10^{-2}
Grid 3	2909	1.0×10^{-2}
Grid 4	4005	6.5×10^{-3}
Grid 5	5253	5.5×10^{-3}
Grid 6	9490	3.8×10^{-3}
Grid 7	15865	3.8×10^{-3}
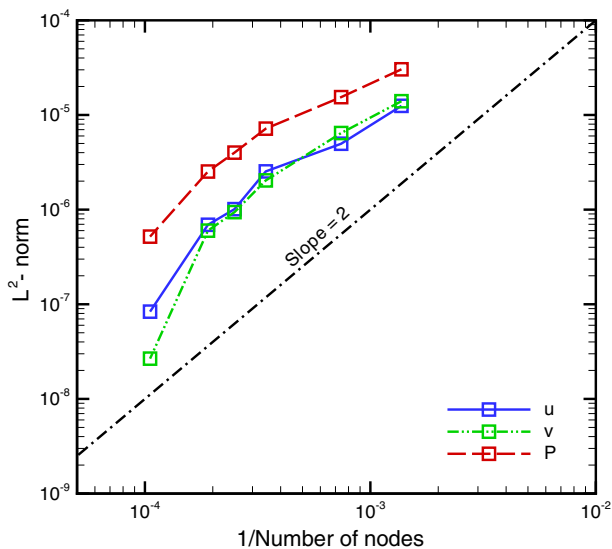
Figure 5. Grid-size independence test: L2-norm errors of radial velocity (u), axial velocity (v) and pressure (P) are plotted with respect to different grids tested. A dashed line of slope 2 is plotted for reference.

Figure 6. Validation of penalty function (PF) and velocity-pressure (VP) formulation. Comparison of radial velocity profiles at different radial locations obtained from present model with data of Hu and Larson [26].

Figure 7. Comparison of computational time taken by penalty function (PF) and velocity pressure (VP) formulation, with respect to number of nodes.

number of nodes in the most refined mesh ($N_{max} = 15865$) and represents the reference solution. Figure 5 shows L^2-norm error of a solution variable (radial velocity u, axial velocity v and pressure p) as function of number of nodes in the computational domain. As figure 5 shows, L^2-norm error for radial velocity (u), axial velocity (v) and pressure (P) solutions for different mesh follows the slope of the slope-2 line. Therefore, we can conclude that our solutions are second-order accurate and grid independent.

3.2 *Code validation*

We validate the code by comparing flow field inside the droplet with the published data. Computed velocity profiles obtained using different formulations are compared in figure 6. The published data is also plotted in this figure. At radial location $r = 0.1$ mm, maximum velocity is
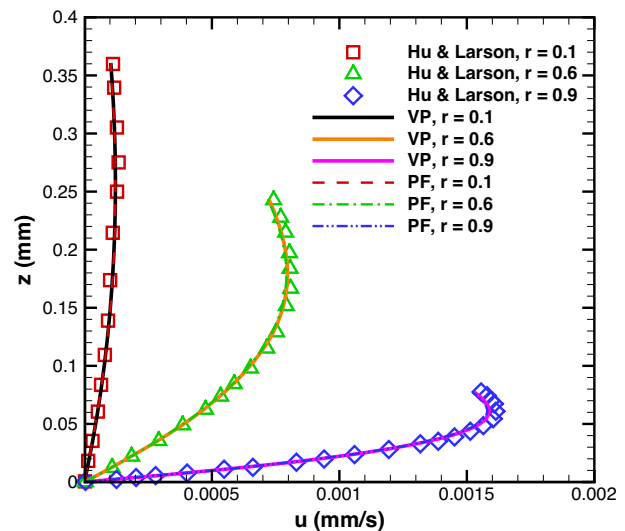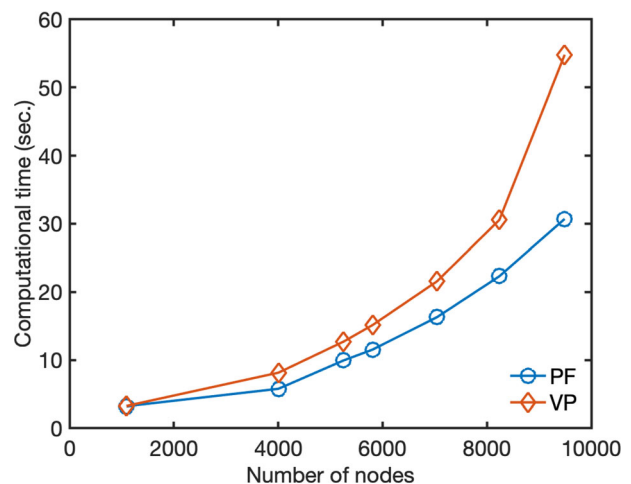
0.13×10^{-3} mm/s and as fluid move radially outward at location $r = 0.6$ and 0.9 mm radial velocity of fluid increases to 0.8×10^{-3} mm/s and 1.6×10^{-3} mm/s, respectively. However, in axial direction, from $z = 0$ to liquid-air interface of droplet, first, droplet velocity increases with the increase of axial distance and after attaining a maximum value it starts decreasing. To validate the code radial velocity at different radial location along the height of the droplet has been compared with results of model of Hu and Larson [26]. Figure 6 shows the radial velocity (u) profiles calculated along axial direction (z) at different radial locations, $r = 0.1, 0.6$ and 0.9 mm, using Velocity-pressure and Penalty function formulation with

direct approach of applying boundary conditions. These profiles were compared with the profiles obtained by Hu and Larson [26]. The radial velocity profile obtained from present models are in good agreement with the profiles obtained by Hu and Larson [26].

3.3 *Computational time*

Figure 7 shows the comparison of computational time taken by both formulations. These computations were performed on a computer having following specifications; 4.84 GHz, 8-core Intel core i9 processor, 64 GB RAM, SSD data storage. As the number of computational nodes increases computational time increases non-linearly for the both formulations. However, velocity pressure formulation consumes more computational time as compared to penalty function formulation and as the number of nodes increases computational time difference for velocity pressure formulation with respect to the penalty function formulation also increases for the computation performed on the same number of nodes. Which concludes that velocity pressure formulation gets slower with increment of number of nodes. Hence, it is preferable to use the penalty function formulation in order to save the computational cost and time.

3.4 *Flow and pressure field on a non-heated and heated substrate*

Figure 8a shows the computed streamlines and contours of radial velocity (u) using penalty function formulation. The streamlines start from the apex of the droplet and end in the the region close to contact line, implying a radially outward flow inside the droplet. Contours of radial velocity (u), plotted in figure 8a indicate that in the central region the flow velocity (~ 0.6 μm/s) is much slower than near the contact line region (~ 1.6 μm/s). Therefore, u increases from droplet interior to the contact line and the maximum u occurs near the contact line. This is attributed to the largest evaporation mass flux j near the contact line, which results in larger mass loss near the contact line. The fluid from the interior of the droplet rushes towards the contact line to maintain droplet spherical cap shape, dictated by the surface tension forces [5, 21]. On a heated substrate, a larger j and $\dot{m}$ accelerate the evaporation and reduces the total evaporation time of the droplet, consistent with prior experiments [20, 29].

Figure 8(b) shows the axial velocity (v) contour inside the evaporating droplet. The minus ($-$) sign represents the downward axial direction of the flow. The magnitude of v decreases from liquid–gas interface to liquids–solid interface, which implies that as the liquid moves from liquid–gas interface towards substrate or contact line region, the axial velocity decelerates. Comparison of
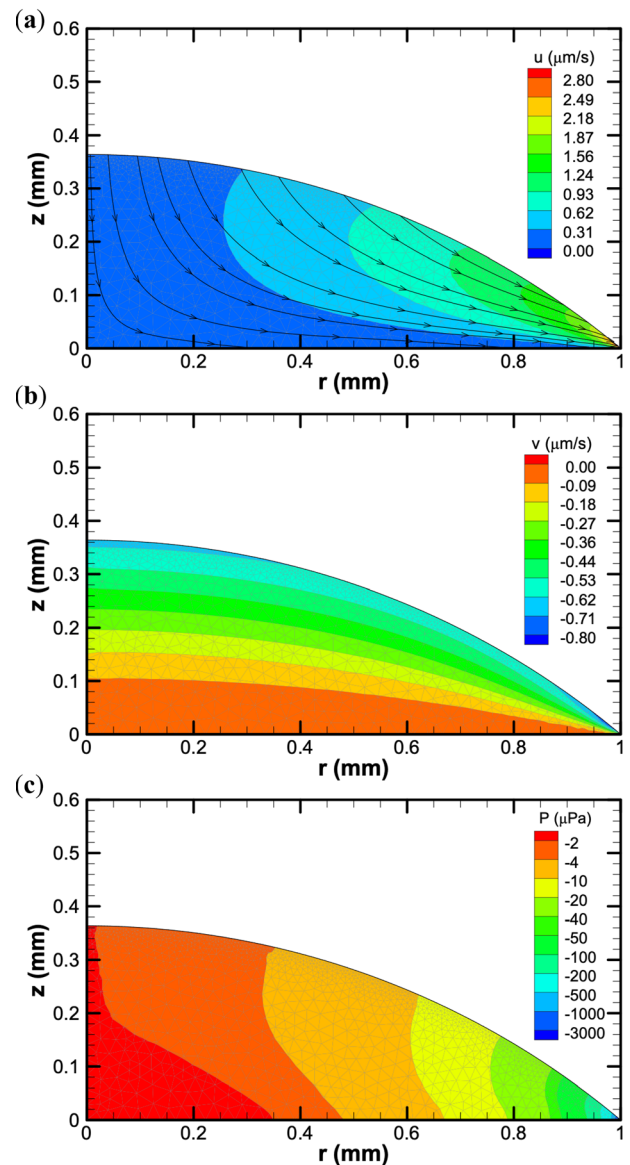

Figure 8. (**a**) Streamlines and contours of radial velocity (u) inside the evaporating droplet (**b**) Contours of axial velocity (v) inside the evaporating droplet (**c**) Contours of pressure field (P) inside the evaporating droplet.

figure 8(a) and b show that the axial velocity is smaller than radial velocity, suggesting a dominating flow in the radial direction compared to axial direction. This fact is also confirmed by direction of the streamlines. Figure 8(c) shows contours of computed pressure (P) inside the droplet. The pressure is the lowest near the contact line due to the largest velocity in this region.

Figure 9 shows radial velocity (u) profiles at different radial locations, when droplet is placed on a heated substrate. Each frame of figure 9 plots the variation of radial velocity with respect to axial coordinate of the droplet for different cases of substrate temperatures and fixed radial locations $r = 0.3, 0.5, 0.7$ and 0.9 mm. At axial location,
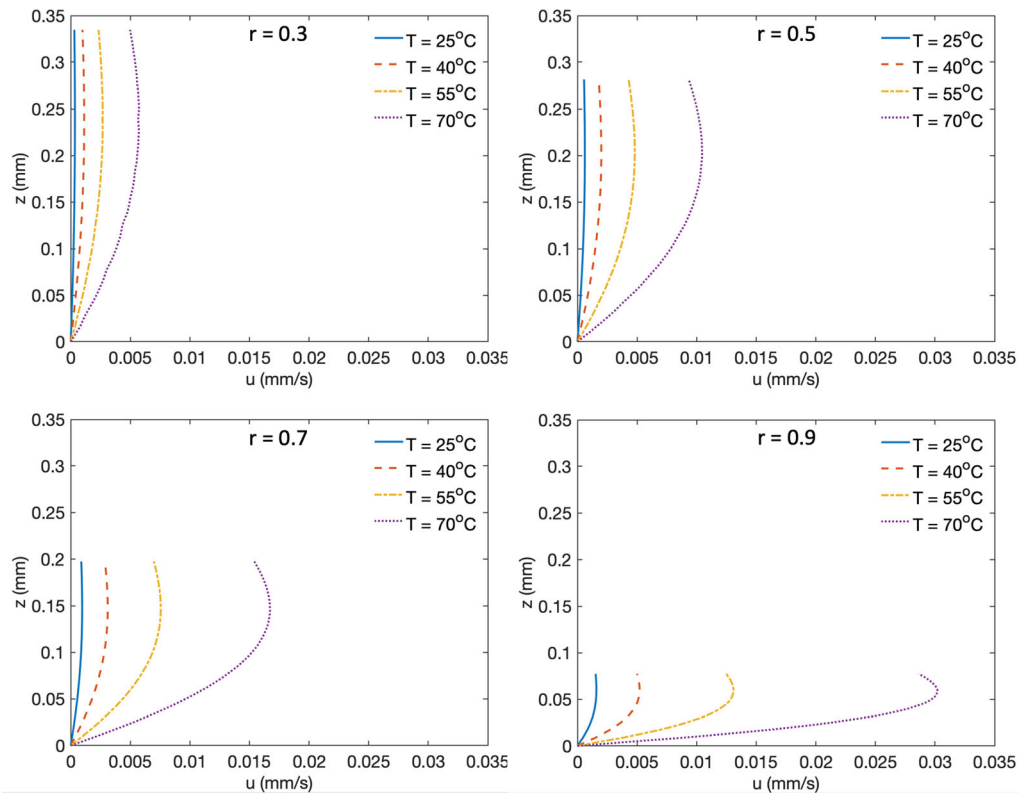
Figure 9. Radial velocity (u) profiles at different radial locations for a droplet evaporating on a heated substrate.

$z = 0$, radial velocity is zero for all cases and at all radial locations, due to zero slip condition. As axial location increases, radial velocity initially increases and reaches to a maximum value before it slightly decreases. With the increase in substrate temperature, radial velocity at any fixed axial location increases at all radial location inside the droplet. As the substrate temperature increases, it results in overall increase of liquid–gas interface temperature. This promotes the larger vapor concentration at the liquid–gas interface and consequently, mass loss is larger from the interface. Hence, it results in a larger evaporation rate (j), thereby increasing the liquid–gas interface normal velocity (u_n) defined by eq. 13 and resulting in larger radial velocity (u) inside the droplet.

Lastly, we discuss the effect of flow field inside a droplet on a heated substrate on self-assembly of colloidal particles, if present in the droplet. Studies have shown a departure from a typical ring-like pattern formed by colloidal particles to a thinner ring and central deposit of particles on a heated surface [29]. Similar patterns of deposits have been obtained for a initially-heated droplet on substrate at ambient temperature [30] and on a non-uniformly heated substrate [31]. In general, these changes in the pattern have been attributed to evaporation-driven flow and flow driven by Marangoni stress along the liquid–gas interface. The pinning or depinning of the contact line

by colloidal particles also play an important role in this context [31].

4. Conclusions

We have presented the development of a finite element solver for computing fluid flow inside an evaporating sessile droplet. The evaporation is considered as quasi-steady with a pinned contact line. Galerkin weighted residual approach is used to formulate weak form of the numerical equations. Penalty function and velocity-pressure formulation are applied to discretise the Navier–Stokes equation in cylindrical coordinates. Six-node triangular mesh is used for higher accuracy of the solution. The solver shows a second order accuracy in grid-size convergence study. Results obtained from the simulations are compared with prior numerical results and found to be in good agreement. Both formulations show similar accuracy in the computed results. However, since penalty function formulation uses one lesser degree of freedom on each node compared to velocity-pressure formulation, resulting in lesser computational time in the former as compared to latter. The solver is tested for simulating the evaporation of a sessile microliter droplet on a heated copper substrate. The future development will involve the implementation of Marangoni stress and buoyancy terms in the FE solver.

Acknowledgements

References

[1] Larson R G 2017 Twenty years of drying droplets. *Nature* 550(7677): 466–467.

[2] Bhardwaj R and Agrawal A 2020 Likelihood of survival of coronavirus in a respiratory droplet deposited on a solid surface. *Phys. Fluids* 32(6): 061704

[3] Bhardwaj R and Agrawal A 2020 Tailoring surface wettability to reduce chances of infection of COVID-19 by a respiratory droplet and to improve the effectiveness of personal protection equipment. *Phys. Fluids* 32(8): 081702

[4] Katre P, Banerjee S, Balusamy S and Sahu K C 2021 Fluid dynamics of respiratory droplets in the context of COVID-19: airborne and surfaceborne transmissions. *Phys. Fluids* 33(8): 081302

[5] Deegan R D, Bakajin O, Dupont T F , Huber G, Nagel S R, Witten T A 1997 Capillary flow as the cause of ring stains from dried liquid drops. *Nature* 389(6653): 827–829

[6] Deegan R D 2000 Pattern formation in drying drops. *Phys. Rev. E* 61(1): 475

[7] Bhardwaj R, Fang X and Attinger D 2009 Pattern formation during the evaporation of a colloidal nanoliter drop: a numerical and experimental study. *New J. Phys.* 11(7): 075020

[8] Bhardwaj R, Fang X, Somasundaran P and Attinger D 2010 Self-assembly of colloidal particles from evaporating droplets: role of DLVO interactions and proposition of a phase diagram. *Langmuir* 26(11): 7833–7842

[9] Li Y, Lv C, Li Z, Quéré D and Zheng Q 2015 From coffee rings to coffee eyes. *Soft Matter.* 11(23): 4669–4673

[10] Girard F, Antoni M, Faure S and Steinchen A 2006 Evaporation and Marangoni driven convection in small heated water droplets. *Langmuir* 22(26): 11085–11091

[11] Widjaja E, Liu N-C, Li M, Collins R T, Basaran O A, Harris M T 2007 Dynamics of sessile droplet evaporation: a comparison of the spine and the elliptic mesh generation methods. *Comput. Chem. Eng.* 31(3): 219–232

[12] Barash L Y, Bigioni T, Vinokur V and Shchur L 2009 Evaporation and fluid dynamics of a sessile drop of capillary size. *Phys. Rev. E* 79(4): 046301

[13] Bourantas G, Petsi A, Skouras E and Burganos V 2012 Meshless point collocation for the numerical solution of Navier–Stokes flow equations inside an evaporating sessile droplet. *Eng. Anal. Bound. Elem.* 36(2): 240–247

[14] Barmi M R and Meinhart C D 2014 Convective flows in evaporating sessile droplets. *J. Phys. Chem. B* 118(9): 2414–2421

[15] Barash L Y 2015 Dependence of fluid flows in an evaporating sessile droplet o the characteristics of the substrate. *Int. J. Heat Mass Transf.* 84: 419–426

[16] Al-Sharafi A, Yilbas B S, Sahin A Z, Ali H, Al-Qahtani H 2016 Heat transfer characteristics and internal fluidity of a sessile droplet on hydrophilic and hydrophobic surfaces. *Appl. Therm. Eng.* 108: 628–640

[17] Pradhan T K and Panigrahi P K 2017 Evaporation induced natural convection inside a droplet of aqueous solution placed on a superhydrophobic surface. *Colloids Surf. A Physicochem. Eng. Aspects* 530: 1–12

[18] Bouchenna C, Saada M A, Chikh S and Tadrist L 2017 Generalized formulation for evaporation rate and flow pattern prediction inside an evaporating pinned sessile drop. *Int. J. Heat Mass Transf.* 109: 482–500

[19] Chen X, Wang X, Chen P G and Liu Q 2017 Thermal effects of substrate on Marangoni flow in droplet evaporation: response surface and sensitivity analysis. *Int. J. Heat Mass Transf.* 113: 354–365

[20] Kumar M and Bhardwaj R 2018 A combined computational and experimental investigation on evaporation of a sessile water droplet on a heated hydrophilic substrate. *Int. J. Heat Mass Transf.* 122: 1223–1238

[21] Hu H and Larson R G 2002 Evaporation of a sessile droplet on a substrate. *J. Phys. Chem. B* 106(6): 1334–1344

[22] Hu H and Larsonv R G 2005 Analysis of the microfluid flow in an evaporating sessile droplet. *Langmuir* 21(9): 3963–3971

[23] Popov Y O 2005 Evaporative deposition patterns: spatial dimensions of the deposit. *Phys. Rev. E* 71(3): 036313

[24] Larson R G 2014 Transport and deposition patterns in drying sessile droplets. *AIChE J.* 60(5): 1538–1571

[25] Hu H and Larson R G 2005 Analysis of the effects of Marangoni stresses on the microflow in an evaporating sessile droplet. *Langmuir* 21 (9): 3972–3980

[26] Hu H and Larson R G 2005 Analysis of the microfluid flow in an evaporating sessile droplet. *Langmuir* 21(9): 3963–3971

[27] J Reddy 2005 *An Introduction to the Finite Element Method*, 3rd Edition, Tata McGraw Hill, New Delhi, 2005.

[28] https://www.mathworks.com/products/matlab.html.

[29] Patil N D, Bange P G, Bhardwaj R and Sharma A 2016 Effects of substrate heating and wettability on evaporation dynamics and deposition patterns for a sessile water droplet containing colloidal particles. *Langmuir* 32(45): 11958–11972

[30] Chatterjee S, Kumar M, Murallidharan J S and Bhardwaj R 2020 Evaporation of initially heated sessile droplets and the resultant dried colloidal deposits on substrates held at ambient temperature. *Langmuir* 36(29): 8407–8421

[31] Malla L K , Bhardwaj R and Neild A 2020 Colloidal deposit of an evaporating sessile droplet on a non-uniformly heated substrate. *Colloids Surf. A Physicochem. Eng. Aspects* 584: 124009