ORIGINAL ARTICLE

Simulation of large-scale additive manufacturing process with a single-phase level set method: a process parameters study

Anh-Duc Le[1] · Benoît Cosson[1] (ID) · André Chateau Akué Asséko[1] (ID)

Abstract

In this work, a numerical method is presented in order to simulate the deposition of molten polymer bead onto a substrate and its cooling down in the large-scale extrusion-based additive manufacturing process. The polymer flow is treated as a single-phase flow with a free surface. This method reduces the computation time without loss of accuracy as polymer behavior significantly dominates air behavior. The governing equations of the fluid motion are solved with the finite element method on moving mesh, whereas the free surface is captured based on the level set method on another fixed mesh. Since the free surface is captured "implicitly" by the zero level of the level set function, coalescence between filaments is well defined and naturally performed. Numerical algorithm and implementation method are described in detail. This model provides detailed information on the cooling process and the bonding formation during the molten polymer deposition process. The effects of control parameters (nozzle velocity, flow rate and extrusion temperature, etc.) on the final deformed shapes of the printed parts are investigated. And finally, the numerical result from 2D simulations is compared to optical micro-graphs of the longitudinal cross-section of the printed sample, which shows good agreement between numerical and experimental results.

1 Introduction

Thanks to its potential innovation, additive manufacturing (AM) technologies have recently considered a technological revolution in the world of materials, especially for polymer materials. The principal commonalities among all polymer AM methods are that they are thermally driven and the part is built layer-by-layer from the ground up. Material extrusion (ME), also known as Fused Deposition Modeling (FDM) or fused filament fabrication (FFF), is the most widely used technique for AM of thermoplastics, polymer blends, and polymer-based composite materials due to its low cost and accessibility for the consumer market. This technique relies on the fundamental functionality that a flexible filament-shaped polymeric material is heated above its glass transition temperature T_g to a semi-molten state in an extruder and is subsequently deposited in a controlled manner onto a substrate. Once a layer is completed, either

the support or the nozzle head moves vertically to deposit a new layer above the previous one. This process is repeated until the desired part is completed. Because of its inherent design, most of the traditional (ME) processes have offered small build volumes and have slow deposition rates for the last two decades [17]. Also, the use of polymer filament as feedstock is limited in the variety of printable materials [19]. As the technology matured, large-scale AM is a natural extension of the ME process. Recent changes in AM production size, speed, and cost make it a promising technology for many applications in the automotive and aerospace industries.

In this context, IMT Lille Douai has developed an extrusion-based AM platform dedicated to manufacturing large-scale parts for thermoplastics and composite materials: LASCALA (Large SCALe plAstics & composites 3D printing). This platform is based on robotic arm systems, which can print objects whose sizes up to 5 m length × 2 m width × 1 m height with build rates up to 5 kg/h. The robotic arm introduces the advantage of 7 degrees of freedom: 6 axes of rotation and 1 axis of translation. The nozzle attached to its end can, therefore, rotate in every direction and move in every direction in space. It operates using feedstock material in a pellet form, which is melted and extruded by a single screw extrusion system. This allows

✉ Benoît Cosson
benoit.cosson@imt-lille-douai.fr

1 IMT Lille Douai, Institut Mines-Télécom, Univ Lille, Centre for Materials and Processes, F-59000 Lille, France

LASCALA to enable faster and cheaper fabrication. The low cost of pellets with respect to filaments makes them suitable for the fabrication of large parts. This also enlarges the range of printable materials while all industrial polymers could be found as pellets. The use of a screw extrusion system provides higher extrusion speed as well as enables the depositing of fiber-reinforced polymers, multiple materials within a single component. LASCALA can, therefore, use 3D printing to produce composite materials with short, chopped fibers and even with continuous fibers.

Large-scale AM of polymer has been studied in academy and industry [29]. Several promising studies have been reported in the literature, focused on various aspects of the process, such as system designs of the process [24, 36], optimization of the process conditions [1, 19], improving the part strength [17], improving the part quality [6], and numerical modelling of the process [5, 8, 13]. However, to transfer from small scale to large scale, various challenges arise that need to be overcome [28]. First, compared to small-scale AM, the large-scale system significantly changes the thermal history of the printed part during the fabrication process which can affect thermal stresses and geometric distortions [24]. Large parts often result in a long cooling time which reduces the surface temperature of the layer below the T_g before the next one is deposited. Since the bonding mechanism is essentially driven by the thermal energy of the semi-molten polymer [30], in this case the inter-layer bonding can not occur properly, resulting in a low overall part strength. On the other hand, an opposing issue occurs when the cooling time is too short. As the material has no sufficient time to cool down, it will not be rigid enough to support its own weight and the upper layer during printing [3]. As a result, hot material may be deformed and be pushed to the side, which causes a dimensional inaccuracy or even a failure of the built part. Second, as the build rates increase, the problem of part quality becomes more critical. Large-scale AM yields large layer size, which reduces surface resolution and enlarges deflects from desired geometry. In small scale, these imperfections can be removed easily with light sanding or vapor polishing. In large scale, however, these defects are larger and more difficult to fix with post-processing [24]. Researches for solving such problems for large-scale AM are still rare; a few attempts have been reported recently, including using reinforcing fibers to improve stiffness and strength of the part [14], using infrared heating to improve the inter-layer bonding strength of the components [17], using thermal imaging for a control system to adjust layer build times [3], or optimizing the screw system design and the die shape of the nozzle to eliminate the void density during the process [36].

To overcome these challenges, the fundamental knowledge of the effect of the process parameters as well as the material parameters on the built part is required in order to optimize the manufacturing conditions. For the conventional AM process, such knowledge has mostly been determined experimentally by trial and error. For large scale, however, trial and error philosophy reduces the economical efficiency of the process due to the significant amount of material is required per time. Therefore, numerical simulations provide an effective alternative to figure out the influence of various parameters on the built part during the process. Extrusion-based AM process involves multi-physics simulation including thermal and mechanical phenomena. Several studies focused on the thermal aspect of the process. One of the first one-dimensional models of heat transfer analysis for the ceramic FFF process was presented in [39]. Li et al. [18] coupled a lumped capacity method to a 1D analytical transient heat transfer analysis of a single filament with an elliptical cross-section. Thomas and Rodriguez [35] performed a 2D heat transfer analysis transient of the solidification process in the FFF process. The analysis was simplified by assuming that the filaments have a rectangular cross-section. The model did not take into account heat transfer along the axis of the filament, and also neglected all contact resistances. Costa et al. [10, 11] developed a 3D analytical model of the transient heat transfer during deposition and cooling of the filament. A global heat transfer is modeled: convection and radiation with the environment, conduction with the substrate and between adjacent filaments, radiation between adjacent filaments, convection with the voids that are created between filaments, as well as all physical contacts between filaments during the progressive construction of a 3D structure. Zhang et al. [40] have developed an adaptable 3D model for calculating the temperature field during and after the FFF process by a numerical approach using the finite difference method. The numerical model allows researchers to study the influence of certain conditions on the FFF process (i.e., nozzle, substrate and environmental temperature, layer thickness, and printing speed) and on the evolution of the temperature field in space and time. 3D thermo-mechanical model of the FFF process was introduced by Zhang and Chou [41, 42], using the element activation technique available in commercial software ANSYS®. The same approach by element activation was used in [4], where a 3D simulation of the heat transfer and crystallization kinetics was sequentially coupled with mechanical analysis to model the residual stresses and the deformation state of the printed part.

In spite of these developments, most of these models only take into account a part of the physical phenomena that appear during the FFF process. They are also simplified from a geometric point of view; the filament shape is rectangular in global simulations or cylindrical, with circular or elliptical sections in filament-scale simulations. This significantly influences the accuracy of the thermal models.

Thereby, a couple of computational fluid dynamics (CFD) and heat transfer should offer a complete simulation for the whole process, which provides a detailed understanding of all the aspects. The geometry as well as the dynamics of bond formation among the built part are determined by the movements of the molten polymer. In addition, the solidification of polymer is modeled by the increase of its viscosity value while it cools down. Furthermore, taking into account the viscoelastic behavior and the thermal expansion in the material model will enable the possibility of compute residual stresses and shrinkage throughout the simulations.

Several efforts to model the full process have been presented recently, where the two-phase fluid flow (including molten polymer and air) and heat transfer were simulated. Du et al. [13] simulated a laser-assisted FDM extrusion process using a couple level set/volume of fluid (VOF) method to track the interface surface, where the viscosity dependence on the shear rate and temperature was taken into account. The same interface-capturing method has also been used in Comminal et al. [7, 25] to simulate the deposition of one molten polymer filament laid on a substrate. In spite of assuming isothermal flows of Newtonian fluid, the numerical model of Comminal et al. [7, 25] was able to predict accurately the dependence of the filament cross-sections on the process parameters, such as the extrusion velocity relative to the velocity of the moving substrate, and the distance of the nozzle from the substrate. Xia et al. [37] simulated the deposition of a filament of molten polymer onto a substrate and its cooling down, using the finite volume/front tracking method and the non-Newtonian fluid model. Their numerical model has further expanded to simulate the thermal expansion of the material and the development of the thermal stresses in [38]. However, both papers [37, 38] lack experimental validation; the accuracy of the method was investigated by convergence tests.

Despite the accuracy and effectiveness achievable, none of these models examines the effect of process parameters on the entire final part. They are also time-consuming, which reduces their robustness extending for the modeling of a large-scale process. This is due to the two main reasons. First, the high number of simulation steps required for the whole process. Second, two-phase fluid flows are simulated yielding a large computational domain (including both polymer and air) in which the conservation equations for mass, momentum, and energy must be solved. In addition, the discontinuity of the flow properties, as well as the thermal properties at the interface, requires special treatment to force them to be continuous in a narrow transition region.

In order to solve a part of this problem, a numerical solution is introduced by using a finite element/level set method to simulate the deposition process, aiming at

reducing computational time without loss of accuracy. The main feature of the method is that the melt flow is treated as a single-phase flow with a free surface. The free surface is captured based on the level set method at each time step on a background fixed mesh independent of the flow solver mesh. A Lagrangian scheme is performed for the advection of the boundary nodes located on the free surface in order to correct the free surface position and improve the mass conservation properties of the level set method. Governing equations including the conservation equations for mass, momentum, and energy are solved by the finite element method on the polymer domain only. The most important advantage of this method over the previous works is the gain in robustness resulting and computation cost from the fact that the computational domain is reduced and the problem of a single fluid is solved. Moreover, this also allows us to impose the correct boundary conditions on the free surface, e.g., the surface tension force.

In the conventional ME process, the material is deposited in the form of cylindrical filament. The elliptical shape of extruded filament cross-section leads to the void density problem and reduces the adhesive area between layers. This will affect the mechanical strength of the printed part. To solve this problem, with LASCALA, different nozzle shapes are examined in order to extrude filament with a flattened-shaped cross-section. In the experimental setup of this work, a flattened rectangle-shaped nozzle of cross-section 2 mm × 18 mm was used to print large-sized thin-walled parts. The motivation is to model the deposition process to analyze the effect of the process parameters on the shape of the final part. Considering the fact of using such nozzle, molten polymer is extruded as a ribbon with very high ratio between the width and the thickness. The melt flow is assumed to be laminar. The lateral flows are considered to be neglected. Because of the assumptions and for computational efficiency, the simulation could be simplified in a two-dimensional model. With the aim of a more accurate thermal analysis, this model also took into account the heat exchanger with the substrate which is often lacking from the recent works [37, 38].

2 Numerical procedures

2.1 Physical and numerical models

A two-dimensional numerical model is developed in order to investigate the molten polymer flow and heat transfer in the deposition process. In the present work, the numerical simulations are performed using the LiveLink^{TM} for Matlab of COMSOL Multiphysics software. The governing equations are solved using finite element–based solver of the COMSOL Multiphysics software, whereas

the interfaces evolution and reconstruction algorithms are implemented in Matlab. The molten polymer is considered to be a non-Newtonian and incompressible fluid. In order to simplify the simulation and limit the computation cost, other properties of the molten polymer such as density, specific heat capacity, thermal conductivity, and surface tension coefficient are assumed do not depend on the temperature.

The domain that is filled by the molten polymer is noted Ω_p, Ω_a is the domain filled by the air, and Ω_s is the substrate's domain (granite plate), whose thickness is 5 mm ($\Omega = \Omega_p \cup \Omega_a \cup \Omega_s$). The subscripts p, a, s, and n are used to refer to the physical properties of the molten polymer, air, substrate, and nozzle, respectively. Likewise, Γ_{ij} is used to denote the interface between materials i and j. The geometry model of computational domain is shown in Fig. 1. Obviously, the domain Ω_p extends during the process; hence, all domains and boundaries depend on time. The computational domain must be thus adjusted in order to always cover Ω_p. In this work, a cycle is chosen in order to update the computational domain in which the trend of change of Ω_p is known. Once the computational domain is updated at the beginning of each cycle, all the numerical solutions defined on the old mesh are transferred onto the new one through linear interpolation and linear extrapolation. Note that all the physical properties inside the domain Ω_p are conserved exactly by this manner. With this procedure, the updating process does not affect the inheritance of the simulation results. In the proposed model, the boundary conditions are given by a fixed position of the nozzle, while the substrate moves with a controlled velocity, which is denoted by $\mathbf{v}_s$. Effects of inertia are neglected. As discussed in [7], the movements of the printing head and substrate are relative to each other; hence, the model equally covers the opposite configuration where the printing head moves and the substrate is fixed. The flow inside the nozzle is assumed to be laminar and fully developed with the average (over the cross-section of the nozzle) velocity v_{in}.

The melted extruded polymer and the body of the nozzle are set to a fixed temperature of $T_n = 250$ °C, the surrounding air temperature is assumed to be $T_n = 20$ °C and remain constant throughout the simulation. The temperature of the substrate is initially set at the surrounding temperature and then updated by thermal exchanges between the substrate and the printed part.

2.2 Governing equations

As it has already be mentioned, the proposed model solves the Navier–Stokes equations only in the domain of polymer Ω_p. Equations 1 and 2 express the general form of the Navier–Stokes equations for single-phase incompressible flow problem.

Momentum conservation equation:

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho \left(\mathbf{u}.\nabla \mathbf{u} \right) + \nabla p - \nabla. \left(2\mu\varepsilon \left(\mathbf{u} \right) \right) = \boldsymbol{f} \tag{1}$$

Continuity equation:

$$\nabla.\mathbf{u} = 0 \tag{2}$$

Where $\mathbf{u}$ is the velocity field, ρ is the density, μ is the dynamic viscosity, p is the pressure, $\varepsilon(\mathbf{u})$ is the strain rate tensor given by $\varepsilon(\mathbf{u}) = \frac{1}{2}\left(\nabla\mathbf{u} + \nabla\mathbf{u}^T\right)$, and $\boldsymbol{f}$ is the vector of the external body forces, which includes the gravity force $\rho\mathbf{g}$ and buoyancy forces, if required.

Under the assumptions of a creeping flow with a very low Reynolds number, the inertial forces are very small compared to the viscous forces and they can be omitted . In

Fig. 1 **a** Schematic of geometry model for simulations. **b** Schematic of the nozzle with a 2×18 mm^2 inlet cross-section
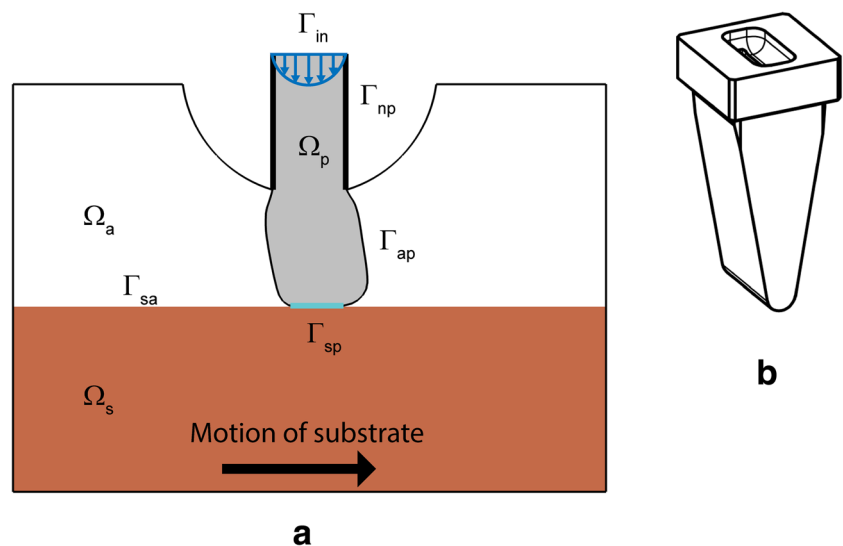
this work, the Navier–Stokes equations are simplified and transformed as Stokes equations:

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \nabla p - \nabla . (2\mu\varepsilon(\mathbf{u})) = \mathbf{f} \qquad (3)$$

Heat transfer is governed by energy conservation equation:

$$\rho C_p \left(\frac{\partial T}{\partial t} + (\mathbf{u}.\nabla) T \right) = \nabla . k \nabla T \qquad (4)$$

Where T is the temperature, and C_p and k are respectively the specific heat capacity (at constant pressure) and the thermal conductivity of the molten polymer. A coupling with the Navier–Stokes equations allows us to consider flow or stay motionless simulation domain.

2.3 Single-phase level set method

The single-phase level set method used to capture the location of the free surface is described here. In the present context, the free surface is the interface between polymer domain and air domain, denoted by Γ_{ap}. Since the original level set method [23, 31] is commonly used for two-phase flow simulation, some terminologies about the two-phase level set method are recalled for the sake of completeness. The free surface is captured implicitly as the zero level set of a smooth function so-called *level set* function $\phi(\mathbf{x}, t)$, which is defined in the entire physical domain (i.e., in both polymer and air phase) as the signed distance from the free surface:

$$\phi(\mathbf{x}, t) = \begin{cases} -d(\mathbf{x}, t), & \mathbf{x} \in \Omega_p \\ 0, & \mathbf{x} \in \Gamma_{ap} \\ d(\mathbf{x}, t), & \mathbf{x} \in \Omega_a \end{cases} \qquad (5)$$

Where $d(\mathbf{x}, t)$ represents the distance from the free surface. The evolution equation of the level set function $\phi(\mathbf{x}, t)$ is given by:

$$\frac{\partial \phi}{\partial t} + \mathbf{u}.\nabla \phi = 0 \qquad (6)$$

The initial condition $\phi(\mathbf{x}, t = 0) = 0$ is given in order to define the initial position of the free surface.

2.3.1 Re-initialization

Although the level set function is initialized to be a signed distance, it may not remain this property under the time evolution, which has undesirable effects on the evaluation of its normal vector and curvature fields [31]. Therefore, level set methods are often combined with a re-initialization technique. Different re-initialization techniques can be found in the literature; a popular method is the Fast Marching Method (FMM) [26, 27]. Another technique is based on solving the Eikonal equation $|\nabla \phi| = 1$ by introducing a time-dependent Hamilton–Jacobi equation [31]. In our level set method, re-initialization is achieved by

recomputing the value of ϕ at each node of the background fixed mesh at each time step according to the following expression:

$$\phi(\mathbf{x}) = sign\left(\phi^0(\mathbf{x})\right) d\left(\mathbf{x}, \Gamma_{ap}\right) \qquad (7)$$

Where $\phi^0(\mathbf{x})$ stands for the value of actual calculated level set function; $d\left(\mathbf{x}, \Gamma_{ap}\right)$ is the distance from the node under consideration to the free surface. A simple geometry-based approach has been used to calculate the distance d. Without loss of generality, in two dimensions using linear triangular element, the free surface Γ_{ap} is approximated by a set of line segments, whose component is denoted as Γ_T. Let T_Γ be the collection of triangles which are intersected by Γ_{ap}, then

$$\Gamma_{ap} = \bigcup_{T \in T_\Gamma} \Gamma_T \quad with \quad \Gamma_T = T \cap \Gamma_{ap} \qquad (8)$$

Γ_T is a line formed by segments in $\mathbf{R}^2$, for each grid node $p \in \mathbf{R}^2$, the distance between p and Γ_T can be defined as :

$$d(p, \Gamma_T) = \min_{x \in \Gamma_T} \|p - x\| \qquad (9)$$

v_1 and v_2 are two vertexes composing Γ_T. Details on the calculations of the coordinates of these vertexes are discussed in (2.3.2). It is straightforward to compute the perpendicular distance $d(p, \{v_1, v_2\})$ from p to the vector composed of v_1 and v_2, and the orthogonal projection of p on this vector. If this orthogonal projection is contained in Γ_T, $d(p, \Gamma_T) = d(p, \{v_1, v_2\})$. Otherwise, the distance from p to the two vertexes v_1 and v_2 has to be determined, by $d(p, \Gamma_T) = \min\{d(p, v_1), d(p, v_2)\}$. Once the distances from the nodal point $p(\mathbf{x})$ to each line segment Γ_T are computed, the minimum of these distance is identified to update the value of the level set function $\phi(\mathbf{x})$ on this point using Eq. 7

$$d\left(\mathbf{x}, \Gamma_{ap}\right) = \min \{d(p, \Gamma_T) : T \in T_\Gamma\} \qquad (10)$$

2.3.2 Free surface construction

The free surface is tracked as a boundary mesh at each time step using the Marching Triangles algorithm, which is based on the earlier work of R. Gantois et al. [16]. Input of the algorithm is a background triangular mesh and a set of scalar values of the level set function $\phi(\mathbf{x})$ at the vertexes of the triangles. Each triangle is successively tested to determine the collection of triangles T_Γ which are intersected by the *zero level* of the level set function. The free surface vertexes are located in edges of each triangle. The computation of the free surface vertexes coordinates is done by using linear interpolation. p and q are the two endpoints of a triangle edge of T_Γ, where $sign(\phi(p)) \neq sign(\phi(q))$. Each vertex

of the free surface v located on a triangle edge $[p, q]$ can be described as a linear combination of p and q:

$$v = (1 - \alpha) p + \alpha q \quad with \quad \alpha = \frac{\phi(p)}{\phi(q) - \phi(p)} \quad (11)$$

Within each triangle $T \in T_\Gamma$, a segment of the free surface Γ_T is created by connecting two nodes composing a line segment; these segments are also considered as a boundary elements. Thus, the boundary mesh is generated by stitching together these boundary elements. As connections occur within each triangle, it is straightforward to generate the connectivity table for the boundary mesh [16]. Special cases may occur if some of the values of the level set function on the vertexes of triangles are equal to 0. n_o is the number of these zero vertexes: it can be equal to 1, 2, or 3. For $n_0 = 1$, two cases are distinguished. In the first case, the two other non-zero values of vertex have the same sign, then the zero value vertex is a point of Γ_T, this case is then ignored. In the second case, the non-zero values have different signs yielding a zero vertex between them using linear interpolation, thus segment becomes a part of Γ_T. For $n_0 = 2$, Γ_T coincides with an edge of the triangle and this edge becomes a part of Γ_T. For $n_0 = 3$, the intersection of the free surface and the triangle are empty, or there is a degeneration. In the case $n_0 = 3$, the curvature κ of the free surface can be too large or the distance between free surfaces can be too small compared to mesh resolution.

According to the desired interface resolution, one or more nodes could be added linearly between the two zero vertexes calculated within each triangle. Figure 2 illustrates the generation of the boundary mesh in two cases where one and two nodes are added. In this work, two nodes are added and it is found out that it is good enough to preserve the accuracy of the interface during the movement.

The next step of the computation is the interfacing with COMSOL Multiphysics software for both fluid flow and heat transfer solvers. The geometry is created from an imported mesh. The imported mesh must be generated by combining the background fixed mesh and the boundary mesh Γ_T. In order to do the combination, the new boundary nodes of Γ_T are firstly numbered, then the sub-elements which are generated through the decomposition of the background mesh elements by the boundary mesh are defined and numbered in respect to the COMSOL Multiphysics mesh element numbering conventions. It is important to notice that this space decomposition generates an arbitrary scenario and gives the possibility ,in some cases, of the generation of sub-elements with bad qualities (sharp aspect ratio). Such undesirable elements may cause low convergence, or inaccurate solutions can be found. In the implemented method, the imported mesh is not used directly for the flow solver. This imported mesh is just used to generate a new geometry inside COMSOL Multiphysics. The flow solver mesh is also regenerated at each time step using the created geometry from the imported mesh. Then, information about material domains and boundary conditions which are predefined on the background mesh are taken as well.

2.3.3 Free surface evolution

The evolution of the level set function is governed by the partial differential equation (Eq. 6). Classical level set methods use this PDE to describe the evolution of the fluid interface. Thus, the interface is captured implicitly by the zero level set of the level set function $\phi(\mathbf{x}, t) = 0$. A well-known drawback associated with the use of level set method to simulate free surface flow (fluid interface in general) is a non-physical loss/gain of mass, especially in regions with a strong interface curvature [22]. This kind of mass loss/gain is partially caused by the advection schemes and partially by the re-initialization procedure. Many attempts to improve mass conservation of level set method have been proposed which include the high-order advection schemes [32, 33], and the improved re-initialization techniques [20, 21]. Hybrid methods that
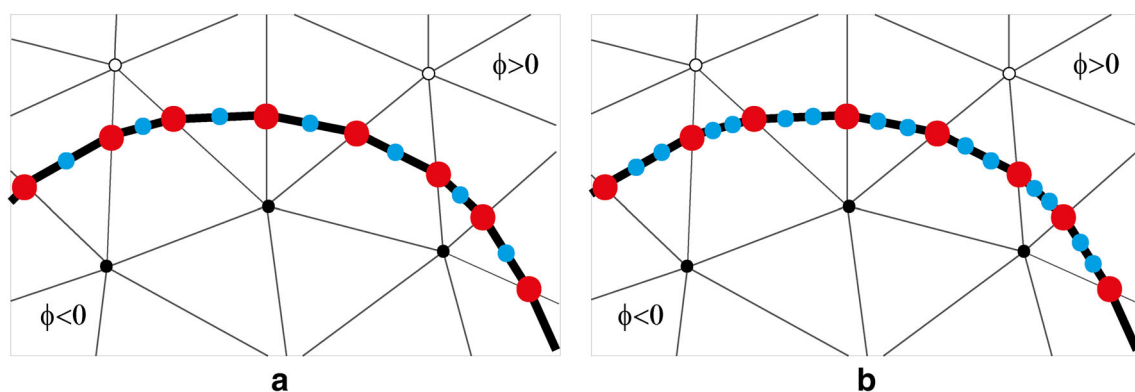

Fig. 2 Generation of boundary mesh. Vertexes with positive level set values in white, vertexes with negative values in black, vertexes with zero values in red, added nodes in cyan. Solid bold lines represent boundary elements of Γ_T. **a** One added node. **b** two added node

couple the level set method with other methods have also been developed such as the CLSVOF method [34] or the hybrid particle level set (PLS) method [15]. These methods exhibit significantly improved mass conservation properties and have been successfully employed in many applications. The complexity of them makes the implementation more complicated than the original level set method. In this work, an alternative approach is proposed to improve mass conservation properties by performing a parallel computation of the level set method and a Lagrangian scheme for the advection of the boundary mesh. The basic idea of this approach is fairly similar to the one-layer particle level set (OPLS) method used in [43]. However, this proposed approach offers a more straightforward implemented technique. In the OPLS method, the level set method is used to smoothly represent the interface, while the Lagrangian particles are used to directly track the interface and correct both the advection procedure and the re-initialization procedure of the level set function. Compared with the OPLS method, the particles are replaced by the boundary nodes which are used to track the interface position directly. As the interface evolves, the density of the boundary nodes is updated automatically by the interface reconstruction technique described previously in (2.3.2), then the resolution of the boundary mesh is always proportional to the resolution of the background mesh. Moreover, in this approach, there is no treatment required for adding particles when the interface undergoes stretching. The boundary nodes are advected with the following evolution equation:

$$\frac{d\mathbf{x}_\Gamma}{dt} = \mathbf{u}(\mathbf{x}_\Gamma) \tag{12}$$

Where $\mathbf{x}_\Gamma$ is the position of the boundary node and $\mathbf{u}(\mathbf{x}_\Gamma)$ is its velocity. The velocities of the boundary nodes are tri-linearly interpolated through the velocity field on the flow solver mesh (detail of the interpolation method can be found in [2]). A second-order TVD-Runge-Kutta scheme is used to evolve the node positions in time integration:

$$\begin{cases} \mathbf{x}_\Gamma^{n+1^{(1)}} = \mathbf{x}_\Gamma^n + \Delta t . \mathbf{u}(\mathbf{x}_\Gamma^n, t^n) \\ \mathbf{x}_\Gamma^{n+1^{(2)}} = \frac{1}{2}\mathbf{x}_\Gamma^n + \frac{1}{2}\left[\mathbf{x}_\Gamma^{n+1^{(1)}} + \Delta t . \mathbf{u}\left(\mathbf{x}_\Gamma^{n+1^{(1)}}, t^{n+1}\right)\right] \end{cases} \tag{13}$$

The solution of the PDE (6) is now considered. As already mentioned, the level set equation is solved over the whole background mesh, while the fluid flow is solved only in the polymer's domain Ω_p on another mesh; therefore, the velocity field on the background mesh must be interpolated. It is straightforward to retrieve the velocities inside the domain Ω_p through tri-linear interpolation from the velocity field on the flow solver mesh. However, the extended velocities are also needed on the air side of the interface to compute the advection of the level set field. In this work, these are defined by extrapolating the velocity field

across the interface using nearest neighbor extrapolation. It is worth to note that only the velocities of the nodes which are near the interface are required to propagate it accurately; the values of nodes far from the interface are not under consideration. The PDE (6) is discretized on the background mesh within a finite element context using the linear shape function. For the time integration, a second-order TVD-Runge– Kutta scheme is used (Eq. 14).

$$\begin{cases} \phi\left(\mathbf{x}, t^{n+1}\right)^{(1)} = \phi\left(\mathbf{x}, t^n\right) - \Delta t . \left(\nabla\phi\left(\mathbf{x}, t^n\right) . \mathbf{u}\left(\mathbf{x}, t^n\right)\right) \\ \phi\left(\mathbf{x}, t^{n+1}\right)^{(2)} = \frac{1}{2}\phi\left(\mathbf{x}, t^n\right) - \frac{1}{2}\left[\phi\left(\mathbf{x}, t^{n+1}\right)^{(1)} \right. \\ \qquad\qquad \left. - \Delta t . \left(\nabla\phi\left(\mathbf{x}, t^{n+1}\right)^{(1)} . \mathbf{u}\left(\mathbf{x}, t^{n+1}\right)\right)\right] \end{cases} \tag{14}$$

The spatial derivatives of the level set function in Eq. 14 can be readily calculated following a standard FEM procedure. The boundary nodes and the level set function are both advected by the velocity field. During the advection procedure, the velocity field inevitably distorts the level set function ϕ and it is no more the sign distance. In order to avoid this problem, the Lagrangian scheme preserves the accuracy of the advection of the boundary nodes. It means that after the advection, the zero level of the level set function no longer represents exactly the new interface, while the boundary node positions are exactly located on the new interface. Therefore, it is necessary to modify the level set function ϕ^{n+1} to ensure that $\phi^{n+1} = 0$ represents exactly the interface. Fortunately, in the proposed method, this can be readily performed thanks to a re-initialization procedure.

Due to the fact that the linear interpolation is used for the computations, together with the linear elements are used to reconstruct the approximation of the interface at each time step, the accurate interpretation of the interface may not be preserved perfectly, especially interfaces with high curvature. On the other hand, high-resolution interfaces with added nodes in the middle preserve properly the curvature of the interface after the movement. In order to counteract this problem, the new calculated boundary nodes are projected on the exact interface. To do the projection, for each calculated boundary node $\mathbf{P}$, an imaginary point $\mathbf{P}_1$ is generated along the normal vector $\mathbf{n} = \frac{\nabla\phi}{|\nabla\phi|}$ of the level set function with a certain delta distance and a point $\mathbf{P}_2$ on the reverse direction. The projection of $\mathbf{P}$ is calculated as the intersection of the line segment $\mathbf{P}_1\mathbf{P}_2$ and the exact interface. This technique also allows the detection and then deletes nodes that came into the merging region, where the two interfaces approach each other with a distance smaller than the delta distance. Hence, boundary nodes with two projections need to be deleted. Figure 3 illustrates the technique of projecting nodes and deleting nodes in the merging region. For the sake of illustration, the discrepancy between the exact and the approximated interface is amplified to some extent. In Fig. 3b, node $\mathbf{Q}$
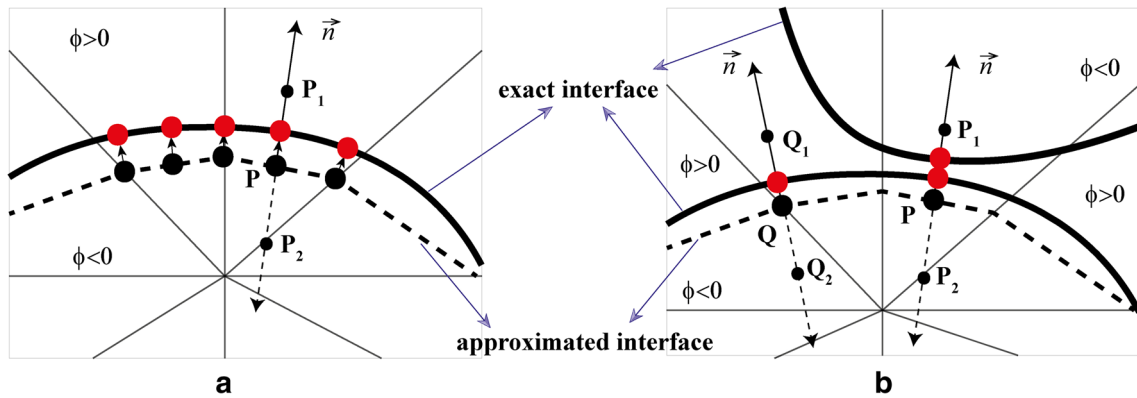
Fig. 3 Techniques of projecting and deleting nodes. Solid bold lines: Exacted interface (represented by the advected boundary nodes). Dashed bold line: approximated interface (calculated by the zero level of the level set function). Big black circle: New calculated boundary nodes. Small black circle: Imaginary points. Big red circle: Projections of the new calculated boundary nodes on the exact interface. **a** projecting nodes. **b** deleting nodes

with one projection will be kept, whereas node **P** with two projections will be deleted.

In summary, the interface evolution procedure is given by performing the following steps:

1. Advection of the boundary nodes $\mathbf{x}_\Gamma$ from time t^n to t^{n+1} with Eq. 13;
2. Advection of the level set function ϕ^n from time t^n to t^{n+1} with Eq. 14;
3. Re-initialization of ϕ^{n+1} with Eq. 7;
4. Reconstruction of the interface using ϕ^{n+1};
5. Exit the procedure.

It is clear that, in the re-initialization step 3, the calculation of the distance from nodes to the new interface is straightforward since the positions and the connectivity table of the boundary nodes are given.

2.4 Boundary conditions

Initial conditions and boundary conditions are described here for flow and thermal models.

Boundary conditions for fluid flow model:

The no-slip boundary condition is used for the flow inside the nozzle and on the substrate wall. At the interface with the substrate, fluid is enforced to move with the substrate velocity.

$$\mathbf{u} = 0 \quad on \quad \Gamma_{np} \tag{15}$$

$$\mathbf{u} = \mathbf{v}_s \quad on \quad \Gamma_{sp} \tag{16}$$

In the inlet Γ_{in}, a fully-developed flow with the average velocity v_{in} on the cross-section of the inlet is prescribed.

$$\mathbf{u}_{average} = v_{in} \quad on \quad \Gamma_{in} \tag{17}$$

On the free surface, the discontinuity of material properties and surface tension forces both cause a jump condition of the pressure across the interface. In the single-phase level set method, this jump condition can be treated by imposing Dirichlet pressure boundary conditions at the interface:

$$p = p_{hydro} + \sigma\kappa \quad on \quad \Gamma_{ap} \tag{18}$$

Where p_{hydro} is the hydrostatic pressure, σ is the surface tension coefficient, and κ is the local interface curvature which can be calculated readily from the level set function:

$$\kappa = \nabla.\mathbf{n} \quad with \quad \mathbf{n} = \frac{\nabla\phi}{|\nabla\phi|} \tag{19}$$

Where $\mathbf{n}$ is the normal vector of the level set function.

Boundary conditions for heat transfer model:

In the inlet and on the nozzle walls, a Dirichlet boundary condition is imposed with a temperature of $T_n = 250°C$:

$$T = T_n \quad on \quad \Gamma_{in} \cup \Gamma_{np} \tag{20}$$

On the free surface, the heat exchange with the surrounding environment is taken as a convective heat flux boundary condition:

$$q = h.(T - T_a) \quad on \quad \Gamma_{ap} \tag{21}$$

Where h is the thermal convection coefficient.

A perfect thermal contact between the substrate and fluid boundaries is considered to be inappropriate for the problems of 3D printing. This type of contact overestimates heat transfer by conduction to some extent. A thermal contact resistance (TCR) is defined at the interface between the substrate and fluid. Therefore, heat flux toward the domain can be imposed as heat flux boundary condition at the interface:

$$q = -\frac{1}{TCR}(T - T_s) \quad on \quad \Gamma_{sp} \tag{22}$$

Where T_s is the temperature of the substrate, which is initially set at the surrounding temperature.

2.5 Numerical algorithm

The implementation of the whole procedure is summarized below:

1. The level set function is initialized according to the initial free surface position.
2. The initial free surface is constructed to generate the imported mesh for the geometry construction in COMSOL Multiphysics.
3. The flow fields, i.e., the velocity and pressure, are computed by solving the Stokes equations (In COMSOL Multiphysics). Notice that since the Stokes equations are time independent, they are solved as a quasi-steady problem at any time instance; the temperature field is initialized by solving the energy equation as a quasi-steady problem at the beginning of a computation. However, since the energy equation is time dependent during the time evolution, a time-dependent solver is used for computing the temperature field at the next time steps.
4. Perform the first step of the second-order TVD-Runge-Kutta scheme to solve the intermediate level set function $\phi\left(\mathbf{x}, t^{n+1}\right)^{(1)}$ (in Matlab). Then, the imported mesh is generated for the intermediate calculation.
5. Compute the intermediate temperature field $T\left(\mathbf{x}_T^{n+1}, t^{n+1}\right)^{(1)}$ which is required for the intermediate calculation. This is done by projecting the temperature field $T\left(\mathbf{x}_T^n, t^{n+1}\right)$ onto the intermediate geometry.

$$T\left(\mathbf{x}_T^{n+1}, t^{n+1}\right)^{(1)} = T\left(\mathbf{x}_T^n + \Delta t.\mathbf{u}\left(\mathbf{x}_T^n, t^n\right), t^{n+1}\right) \tag{23}$$

Here, the temperature field $T\left(\mathbf{x}_T^n, t^{n+1}\right)$ is obtained by solving the energy equation on the geometry at time level n with the initial temperature field $T\left(\mathbf{x}_T^n, t^n\right)$ for a time period Δt.

6. Perform the intermediate calculation to compute $\mathbf{u}\left(\mathbf{x}, t^{n+1}\right)$.
7. Perform the second step of the second-order TVD-Runge-Kutta scheme to solve for ϕ^{n+1} then generate the imported mesh for the next calculation.
8. Project $T\left(\mathbf{x}_T^n, t^{n+1}\right)$ onto the new geometry:

$$T\left(\mathbf{x}_T^{n+1}, t^{n+1}\right)^{(2)} = T\left(\frac{1}{2}\mathbf{x}_T^n + \frac{1}{2}\left(\mathbf{x}_T^{n+1\,(1)}\right.\right.$$
$$\left.\left. + \Delta t.\mathbf{u}\left(\mathbf{x}_T^{n+1\,(1)}, t^{n+1}\right)\right), t^{n+1}\right) \tag{24}$$

Here: $\mathbf{x}_T^{n+1\,(1)} = \mathbf{x}_T^n + \Delta t.\mathbf{u}\left(\mathbf{x}_T^n, t^n\right)$.

9. The temperature field T^{n+1} is computed from two parts:

$$T^{n+1} = \frac{1}{2}.\left[T\left(\mathbf{x}_T^{n+1}, t^{n+1}\right) + T^*\left(\mathbf{x}_T^{n+1}, t^{n+1}\right)\right] \tag{25}$$

Where $T^*\left(\mathbf{x}_T^{n+1}, t^{n+1}\right)$ is the temperature field computed on the new geometry, where the initial temperature field $T^*\left(\mathbf{x}_T^{n+1}, t^n\right)$ is obtained by projecting $T\left(\mathbf{x}_T^n, t^n\right)$ in the same manner as $T\left(\mathbf{x}_T^{n+1}, t^{n+1}\right)$.

10. Steps 3–9 are repeated until a specified end time is reached.

2.6 Physical properties of material

Properties of acrylonitrile-butadiene-styrene (ABS) and granite are used in the simulations. ABS is used for the printed part and granite for the substrate materials. The properties of materials used in the numerical simulations are provided in Table 1. The thermal properties of ABS are taken from the literature [11]. The properties of the granite substrate are loaded from the library material contained in COMSOL Multiphysics 5.4. For this work, thermal properties of materials are assumed to be constant throughout the simulations. This reduces the complexity of the model and the computational time.

The rheological properties of the ABS are determined using an Anton Paar Physica MCR 301 rotational rheometer. All the measurements are performed using a 35-mm parallel-plate geometry and nitrogen is used as a protective gas to avoid the degradation of the polymers. Owing to the fact that the direct measurement of the shear viscosity over a broad range of shear rate is difficult (due to failure of the sample at high shear rate, i.e., $\dot{\gamma} > 10[s^{-1}]$), the measurements are conducted in the dynamic oscillatory mode. The shear viscosity is then extracted from the results using the Cox–Merz rule. For homogeneous polymer melts, this empirical rule equates an approximate agreement between the magnitude of the complex viscosity $|\eta^*(\omega)|$ and the steady shear viscosity $\eta(\dot{\gamma})$ if the angular frequency is set equal to the shear rate:

$$|\eta^*(\omega)| = \eta(\dot{\gamma}) \quad (\omega = \dot{\gamma}) \tag{26}$$

The Cross-WLF viscous model is used for the simulation of the non-isothermal flow, where viscosity depends on temperature and shear rate, as follows:

$$\eta = \frac{\eta_0}{\left(1 + \left(\frac{\eta_0\dot{\gamma}}{\tau^*}\right)\right)^{(n-1)}} \tag{27}$$

Table 1 Material properties used in the simulations

Symbol	Parameter	Value	Unit
ρ_p	Density of ABS [11]	1050	kg/m^3
C_p	Specific heat capacity of ABS [11]	2200	J/(kg·K)
k_p	Thermal conductivity of ABS [11]	0.18	W/(m·K)
TCR	Thermal contact resistance polymer to substrate [13]	1×10^{-3}	m^2·K/W
σ	Surface tension coefficient [13]	0.042	N/m
h	Thermal convection coefficient [37]	20	W·m^2/K
ρ_s	Density of the substrate	2600	kg/m^3
C_s	Specific heat capacity of the substrate	850	J/(kg·K)
k_s	Thermal conductivity of the substrate	2.9	W/(m·K)

$$\eta_0 = D_1 \exp\left[\frac{-A_1 (T - T_g)}{A_2 + (T - T_g)}\right] \qquad (28)$$

Where, $\boldsymbol{\eta_0}$ is the viscosity at zero shear rate, $\dot{\boldsymbol{\gamma}}$ is the shear rate. τ^*, n, D_1, A_1, A_2 are material-dependent constants. T_g in Eq. 28 stands for the glass transition temperature of polymer. The values of these model parameters are identified from experimental measurements by using a nonlinear least squares fitting method such with the minimization of the objective function $f(\tau^*, n, D_1, A_1, A_2)$:

$$min_X f(X) = ||\eta_f (T, \dot{\gamma}, X) - \eta_m (T, \dot{\gamma})|| \qquad (29)$$

Where, η_m and η_f are the measured and fitted viscosity, respectively, and $X = [\tau^*, n, D_1, A_1, A_2]$. The model parameter values giving the best fits to the experimental data are listed in Table 2. The glass transition temperature of ABS $T_g = 105$ °C is measured using differential scanning calorimetry (DSC). The value of measured viscosity and the corresponding fitted viscosity curves are given in Fig. 4.

3 Results and discussions

3.1 Convergence study

In order to show the mass conservation capability of the method and verify the convergence of the results with the size of the elements of the mesh, simulations of the

deposition of two short filaments, one on top of each other, are performed. The width of the inlet of the nozzle is 2 mm, the average extruded velocity at the inlet is $v_{in} = 5$ mm/s, the substrate moving at a constant speed $\mathbf{v}_s = 5$ mm/s along the x-axis. Small velocities were chosen for better visibility of temperature diffusion. The gap between the nozzle and the substrate is $g = 2$ mm for the lower layer and $g = 4$ mm for the top one. Once the first layer is completed, the substrate moves down in Z direction at a constant speed $\mathbf{v}_n = 5$ mm/s until the new position is reached. The temporal evolution of the simulation is shown in Fig. 5. It can be seen in this figure that the computational domain is updated as the polymer's domain Ω_p evolved. In these simulations, the cycle of updating is chosen for the purpose of covering the deposition of the next 5-mm filament along the x-axis. Initially, due to the large temperature difference between the hot polymer and the substrate, the polymer cools down immediately upon contact. Thus, the changing of the temperature of the first layer is visible. However, the cooling of the top layer is not as intuitive as the lower

Table 2 Parameter values of the Cross-WLF model

Parameter	Value	Unit
τ^*	2.87×10^4	Pa
n	0.42	–
D_1	1.4467×10^{12}	Pa s
A_1	29.7634	–
A_2	70.0139	K
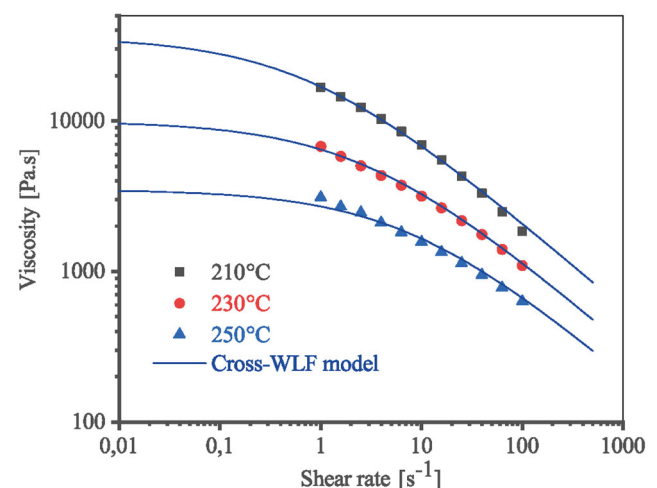
Fig. 4 Measured viscosity data (symbols) for ABS and fitted curves according to the Cross-WLF model

one. This is because the thermal diffusion rate is relatively low compared with the deposited velocity, and since short filaments are built for the purpose of convergence study, the temperature on the upper surface of the first layer is still high when the contact occurs. The ability to manage interface merging of the method can be seen in Fig. 5. On the contact area between the two layers, where the coalescence occurs, the interface disappears.

The precision of the numerical simulation depends on the mesh size and the time-step intervals. Since explicit time integration schemes are used, stability restrictions on time step, Δt, using the Courant-Friedreichs-Lewy condition (CFL condition), are given by the following inequality :

$$\Delta t < \frac{h}{\max \{|\mathbf{u}|\}} \tag{30}$$

Where $\max \{|\mathbf{u}|\}$ is the largest magnitude of the velocity on the interface and h is the mesh size. Equation 30 is enforced by choosing a *CFL number* α with:

$$\alpha = \Delta t \left(\frac{\max \{|\mathbf{u}|\}}{h} \right) \quad \text{with} \quad 0 < \alpha < 1 \tag{31}$$

Experimentally, the stability is reached by choosing $\alpha = 0.6 - 0.8$. Three different mesh resolutions, corresponding to three specific mesh sizes $h = 0.1$ mm, $h = 0.2$ mm, and $h = 0.3$ mm, respectively, are used for the simulations. A comparison of the polymer shapes at the end of the deposition process for various levels of mesh resolution is shown in Fig. 6. The results show that the polymer shapes are captured well for all reported mesh resolutions. As the mesh size decreases, the results have converged to one of the smallest mesh size. As expected, the two finest meshes give essentially the same shapes: the difference only appears in the highest curvature region.

In this paper, the relative global mass error E_r is defined as:

$$E_r \% = \left(1 - \frac{A_n}{A_t} \right) \times 100\% \tag{32}$$

Where A_t is the theoretical value of the area occupied by Ω_p; A_n is the numerical value of area enclosed in the zero contours of the level set function. The relative global mass error throughout the deposition process versus mesh sizes is plotted in Fig. 7. It can be seen that despite many complicated phenomena that occur throughout the deposition process (i.e., the contact between the polymer and the substrate, the coalescence between two layers of polymer), the proposed method shows good mass conservation even in coarse mesh size. During the first 4.65 s, the mass conservation results at the two finest mesh sizes are excellent at less than 0.2%. From time $t = 4.65$ s to the final time at $t = 8.25$ s, where the second layer is depositing, hence the interface between layers is merging, a certain amount of mass gain is found. This is due to the fact that the two interfaces are considered to be coalescence when the distance between them is smaller than a delta distance. For the sake of stabilization, delta distance must hold a certain value. Nevertheless, it can be seen that this mass gain can be improved significantly by using smaller mesh sizes.

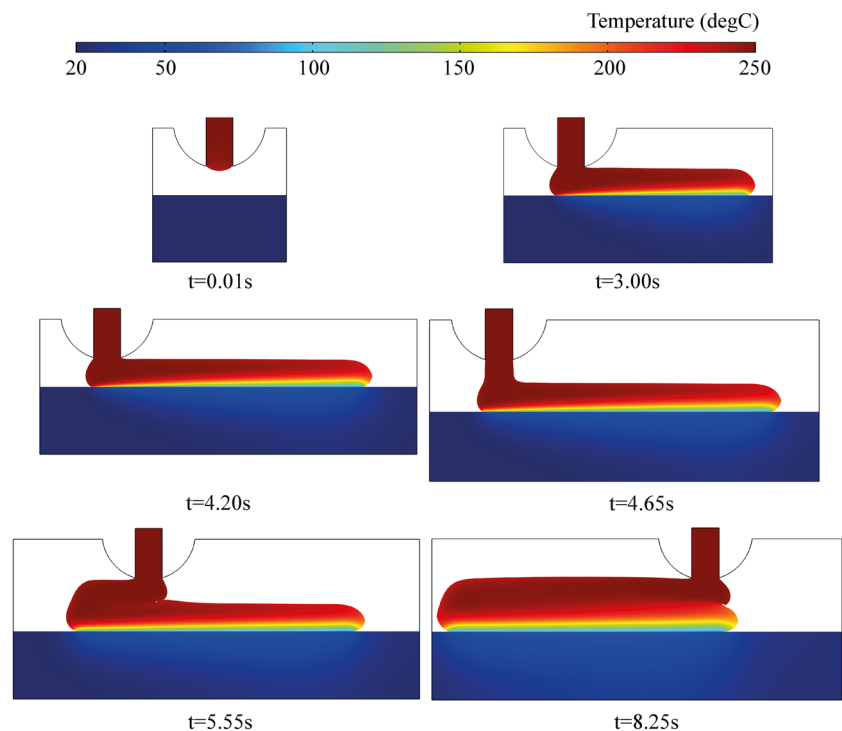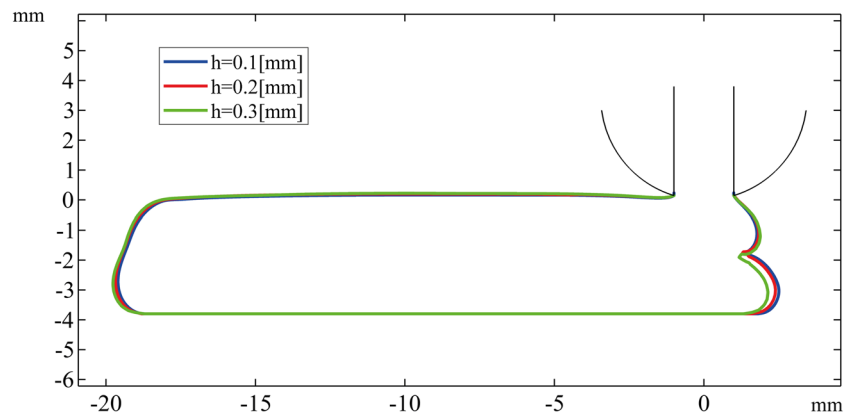**Fig. 5** Temporal evolution of the simulation

In addition to verifying the mesh convergence for the polymer shape, the convergence for the heat transfer solution had also investigated. The results are excellent since the discrepancy of the volumetric average temperature of the polymer between the coarsest and the finest mesh sizes was kept at less than 0.5 °C during the whole process.

3.2 Parametric effects

As already mentioned, previous studies in the literature [12, 37] have often simplified boundary conditions at the contact surface between the polymer and the substrate by prescribing a fixed temperature at the contact surface or using a perfect thermal contact between the substrate and the fluid. These assumptions are considered to be inappropriate for the problems of interest. It overestimates heat transfer by conduction to some extent [12]. In order to consider this argument, three simulations with different boundary conditions at the contact area with the substrate are performed. Figure 8 shows the temperature distribution
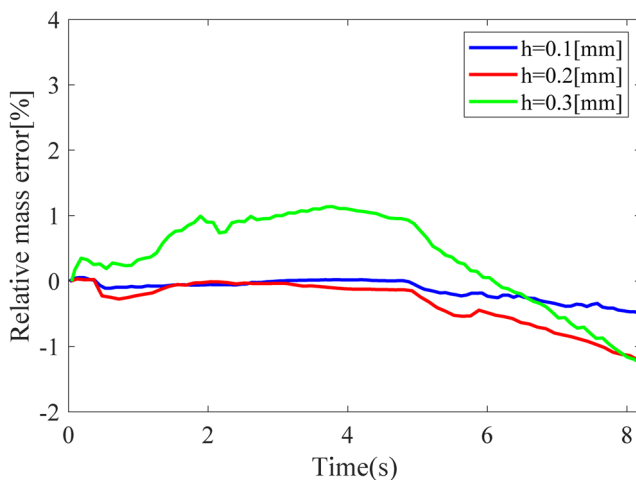
Fig. 7 Relative global mass error during the deposition process for different mesh sizes

of the polymer and the substrate when the first filament is deposited on the substrate. Figure 9 shows the evaluation of the temperature in the cross-section indicated by the vertical black line in Fig. 8. In the first two cases, we can see that the temperature fields are continuous across the surface contact, whereas a gap is appeared in the third case. Since a perfect thermal contact enhances the conduction through the substrate, the polymer cools down rapidly. The volumetric average temperature of polymer at the end of the position of the first layer for the first two cases are $T_m = 201.44$ °C and $T_m = 208.95$ °C, which are deeply smaller than the one of the third case $T_m = 225.23$ °C.

It is worth to note that heat transfer by conduction with the substrate occurs more rapidly in the early stages of the process when the temperature gap between the polymer and the substrate is still important. As time evolves, the polymer cools down and the substrate heats up, this temperature deviation gradually becomes smaller. Therefore, the heat loss of the printed part due to exchange with the substrate decreases over time. Nevertheless, control of the thermal history is a key success for the optimization of the FFF process. In reality, a perfect thermal contact between the polymer and the substrate is certainly not happening. Thus, the thermal contact resistance should be fully considered.

In this section, the effects of the printing conditions on the shape of the final printed part are examined. Here, the effects of three parameters are present: the layer thickness; the extrusion flow rate (i.e., the average velocity specified at the inlet of the nozzle v_{in}), and the printing speed (i.e., the magnitude of the velocity of the moving substrate $|\mathbf{v}_s|$). The shape of the thin-walled part can be characterized by the width W and the height H. Considering the current deposited layer, the thickness δ can be simply determined thanks to the mass conservation; the volumetric flow rate at the inlet boundary of the nozzle must be equal to the volumetric flow rate of material deposited.

$$\delta = \frac{v_{in}.D}{|\mathbf{v}_s|} \tag{33}$$
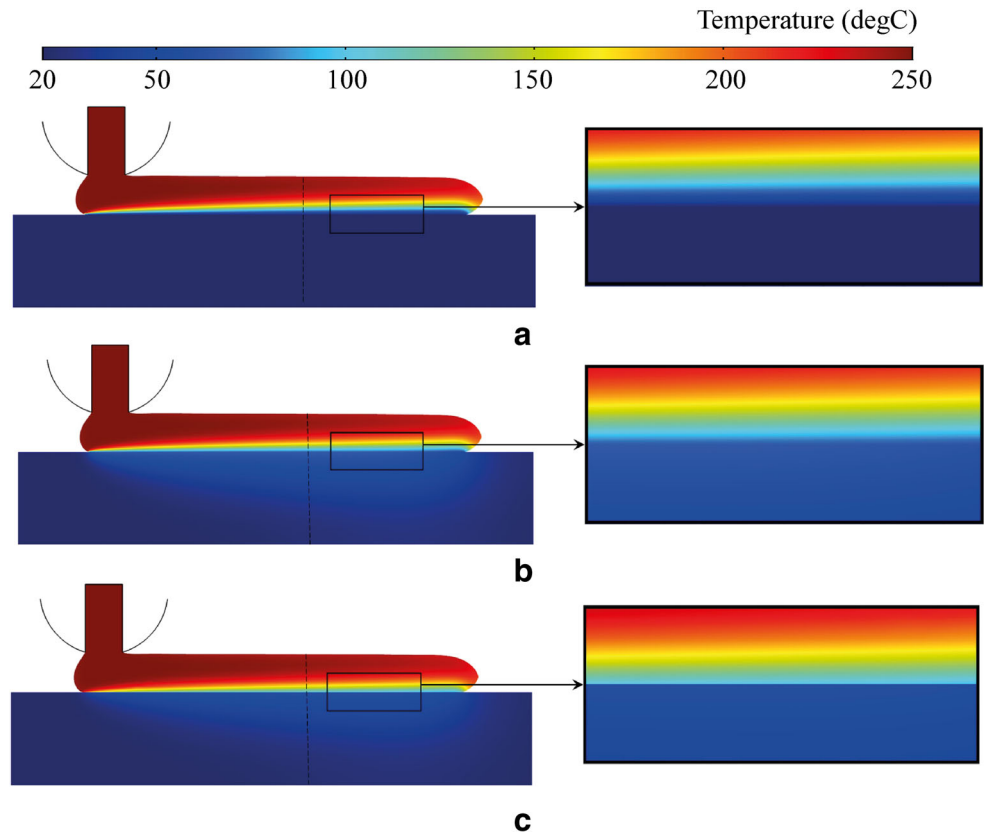
Here, D stands for the width of the nozzle. It is undoubted that if the vertical gap distance g between the nozzle and the upper surface of the previous layers (or between the nozzle and the planar substrate for the first layer) satisfies the condition $g > \delta$, then there is no interaction between the exterior walls of the nozzle and the polymer. In this case, the shapes of the part would become independent of the gap (Fig. 10c). In contrast, a gap value lower than that limit
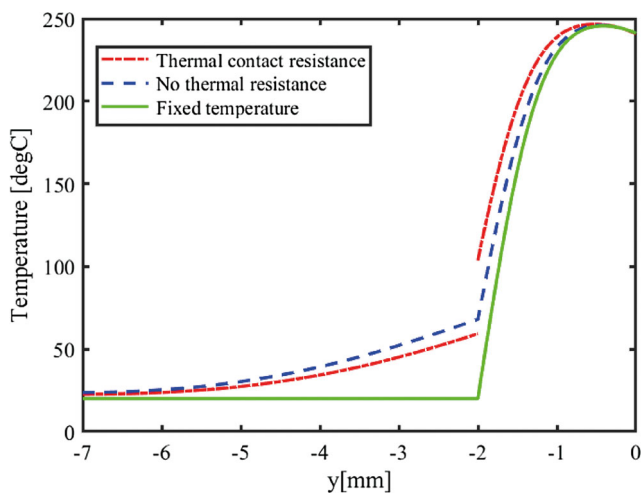

Fig. 9 Evaluation of the temperature

may distort the polymer in different levels as can be seen in Fig. 10. Nevertheless, controlling the gap is not easy since it depends on the setting of layer thickness, the printing speed, and the ratio between the extrusion flow rate and the printing speed. Figure 10a and b show the impact of the nozzle wall on the polymer in case the support material is rigid (i.e., (*a*) when the first layer is deposited on the substrate or (*b*) when the previous layer had been cooled down and become rigid). In this case, only the deposited material in the contact zone is restrained by the exterior wall of the nozzle. However, the printing material quickly reaches a stable state out of the affected area. Figure 10c and d illustrate the impact of the gap height on the printed material; in this case, the support material is still malleable. This occurs when the layer build time is too short; thus, the prior layer had no sufficient time to cool down. For large gap, in Fig. 10c, no impacts were found neither on the current layer nor on the prior layer. In Fig. 10d, a small gap leads to a pressure of the extruding material on the support material, which distorts the contact surface and pushes a certain amount of material to the side. This effect further reduces the height while increasing the width of the part. Figure 10e and f show the two typical cases for the interaction between the nozzle and the polymer when the gap is very small or/and the speed ratio is too high, where a considerable amount of material rises upwards along the exterior nozzle wall. It can be seen in Fig. 10e

Fig. 10 Various effects of the nozzle on the polymer during the deposition process
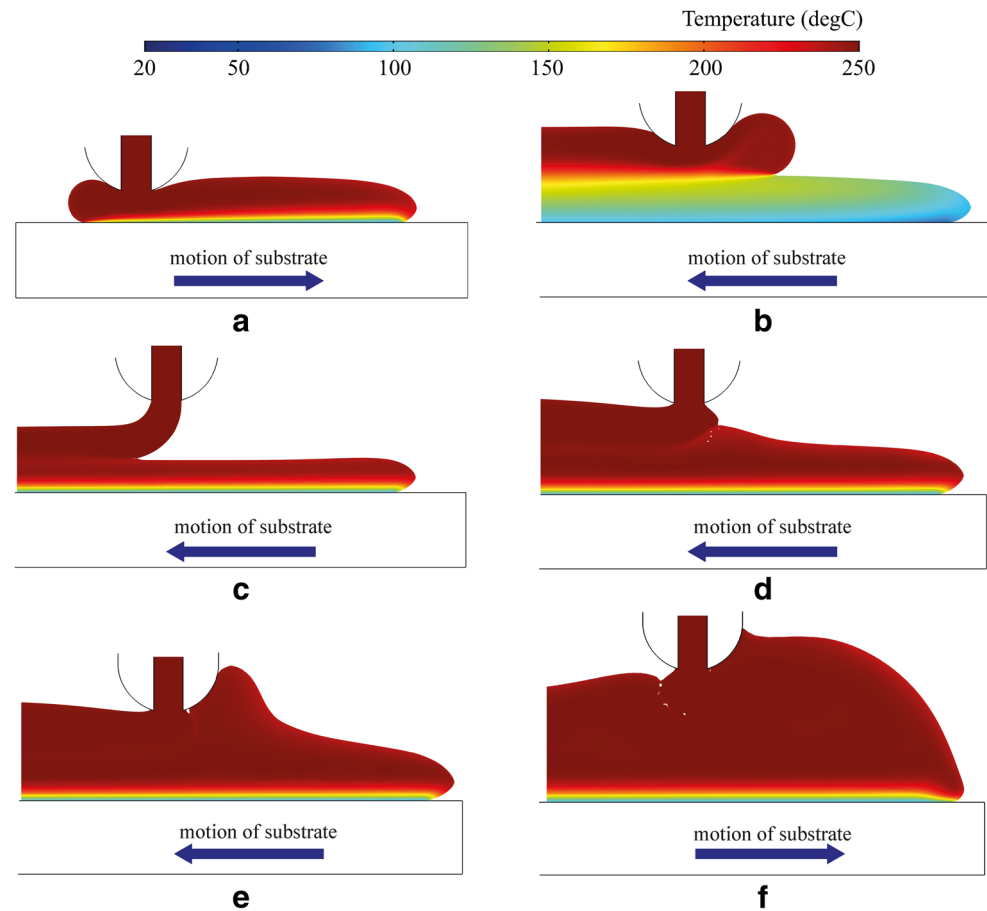
that the support material was strongly deformed, risen in front of the nozzle, and pushed to the side. As material rises upwards and adheres to the nozzle walls (Fig. 10f), it was then dragged with the movement relative to the nozzle.

The parametric study of the extrusion flow rate is investigated. Five simulations are performed corresponding to five extrusion flow rates $v_{in} = 30$ mm/s, $v_{in} = 40$ mm/s, $v_{in} = 45$ mm/s, $v_{in} = 50$ mm/s, and $v_{in} = 60$ mm/s, while the printing speed is fixed to $\mathbf{v}_s = 30$ mm/s. The simulation is of the printing of a thin-walled part of 4 layers, each layer of 50 mm width. The vertical gap g is initially set to 3 mm, then added 3 mm for each additional layer. The inlet flow is suspended when the support went down. Figure 11 shows the variety of all the geometric shapes that have been obtained within the numerical simulations. In the left column, the shapes of the first deposited layer are represented. It is evident that the layer thickness varies significantly with the change of the extrusion flow rate. Furthermore, the more the extrusion flow rate is important, the more amount of material spreads in front of the nozzle, which elongates the layer width to some extent. In the right column, as the extrusion flow rate increases, the parts tend to increase significantly the width, while increasing

slowly the height. This can be explained by the fact that the more the extrusion flow rate is important, the smaller the gap g is, and an important amount of material is pushed to the side. This issue was already explained in Fig. 10d. Moreover, in case the extrusion flow rate is too high, the impact of the nozzle may deform the part, which is shown in Fig. 11e.

The effect of the printing speed is also investigated here. The shape of the final part and the temperature fields obtained using two different printing speeds of $\mathbf{v}_s = 5$ mm/s and $\mathbf{v}_s = 30$ mm/s are shown in Fig. 12. The corresponding extrusion flow rates are chosen $v_{in} = 7.5$ mm/s and $v_{in} = 45$ mm/s respectively. These values ensure an identical speed ratio for the two simulations. Certainly, changing the printing speed leads to changing the cooling process. As expected, the lower printing speed results in a cooler part. Furthermore, the results indicate that the thermal diffusion to the substrate is major compared to the diffusion to the air surrounding. For lower printing speed, the previous deposited material has more time to cool down before depositing the new layer. This reduces the effects of the nozzle walls and the extrusion pressure on the part, hence resulting in a final part of better shape.
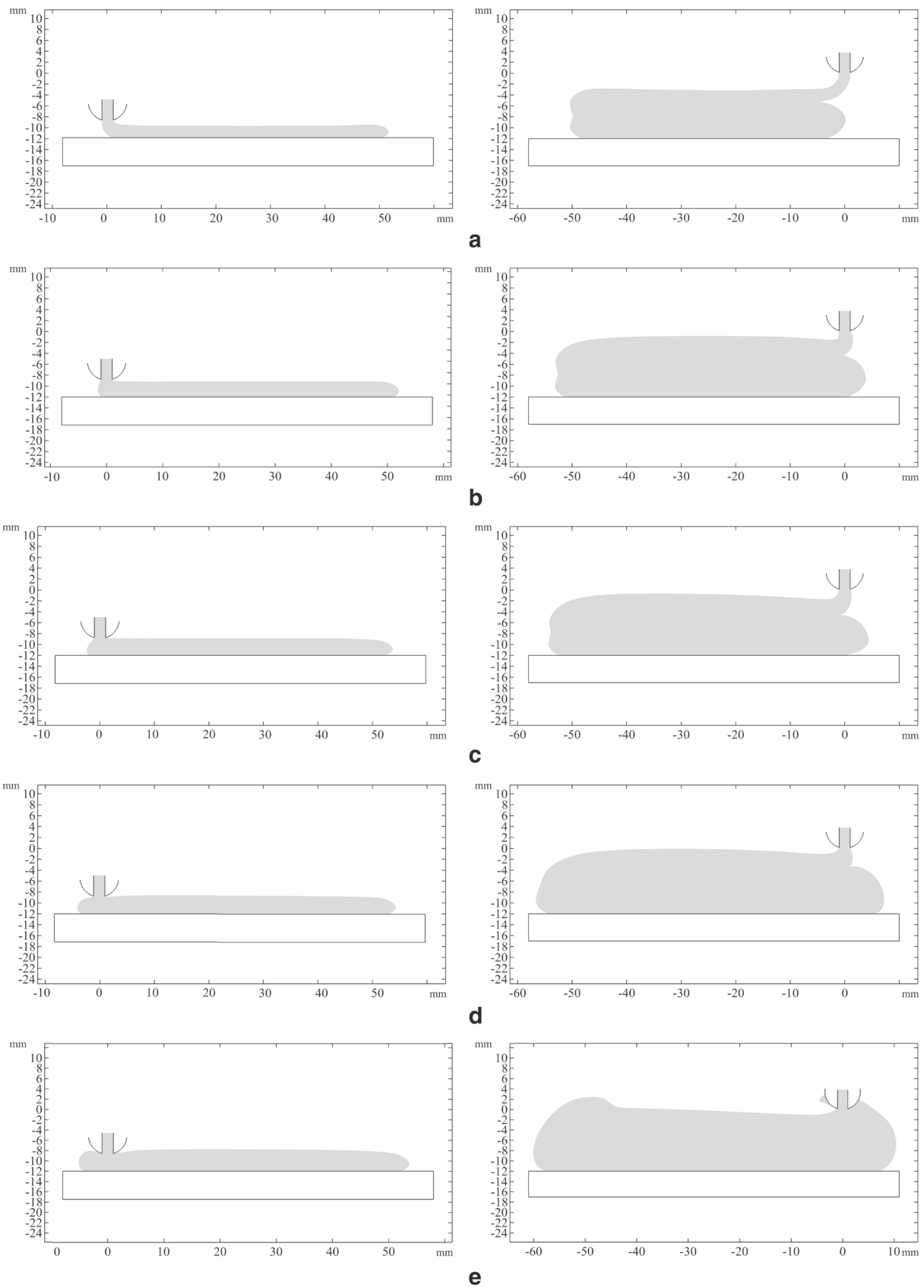
Fig. 11 Simulated shape of the deposited filament for different extrusion flow rates. **a** $v_{in} = 30$ mm/s; **b** $v_{in} = 40$ mm/s; **c** $v_{in} = 45$ mm/s; **d** $v_{in} = 50$ mm/s; and **e** $v_{in} = 60$ mm/s
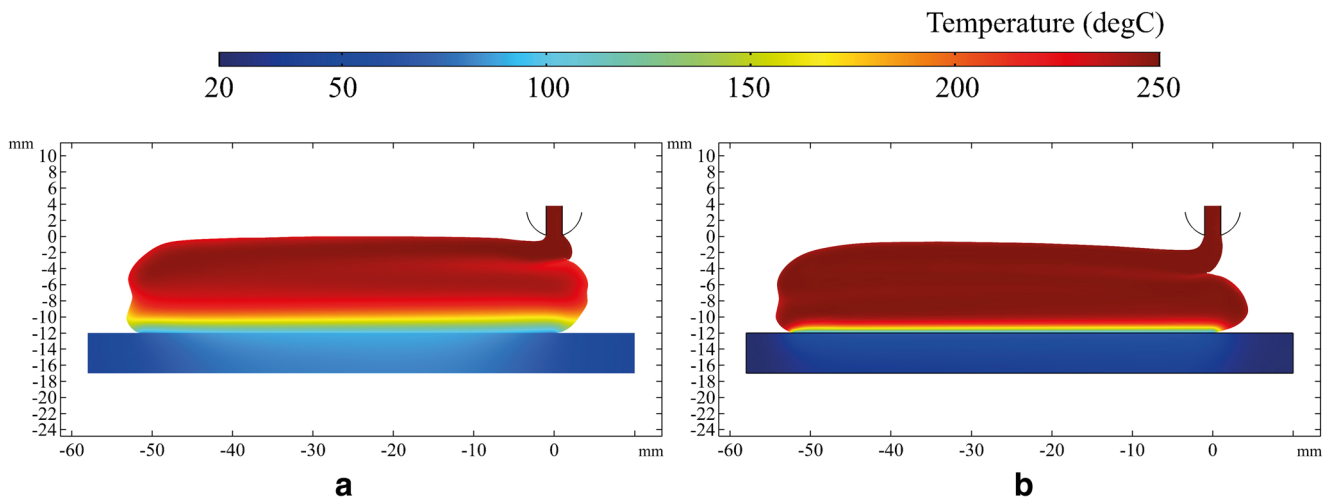
Fig. 12 Simulated shape of the deposited filament for different printing speeds. **a** $v_s = 5$ mm/s; **b** $v_s = 30$ mm/s

3.3 Comparison with experimental observation

The shape of the final printed part predicted by numerical simulation is compared with the longitudinal cross-section of the printed sample fabricated with LASCALA using the flattened rectangle-shaped nozzle of cross-section 2 mm×18 mm. For the sake of comparison, all process parameters in the simulation are directly derived from the experimental setup. In the present case, a continuous FFF process is performed. There is no stop during the deposition between two consecutive layers. In order to improve the adhesion between the part and the substrate, the first two layers are wider than the others with the widths of 70 mm and 60 mm respectively. Other layers are of 50 mm width. A thin-walled part of 7 layers is fabricated. Afterward, the

part is cut at the middle through a plane parallel to the printing direction. The longitudinal cross-section is polished and examined under an optical microscope. The comparison of the shapes of the simulated parts with the experimental observations is presented in Fig. 13. As is shown in the figure, the part printed at higher speed has an irregular layer thickness; this is due to the effect of the nozzle walls and the extrusion pressure during the printing process. In both cases, a small amount of material is pushed sideways during the printing process forming the trapezoidal parts. This effect is more clear in the case of higher printing speed as the part is hotter and more malleable. The predicted part shapes agree fairly well with those observed in the experiment. Bubbles can be seen at the interface of each layer in the experimental sample. They may occur with viscoelastic phenomena at the
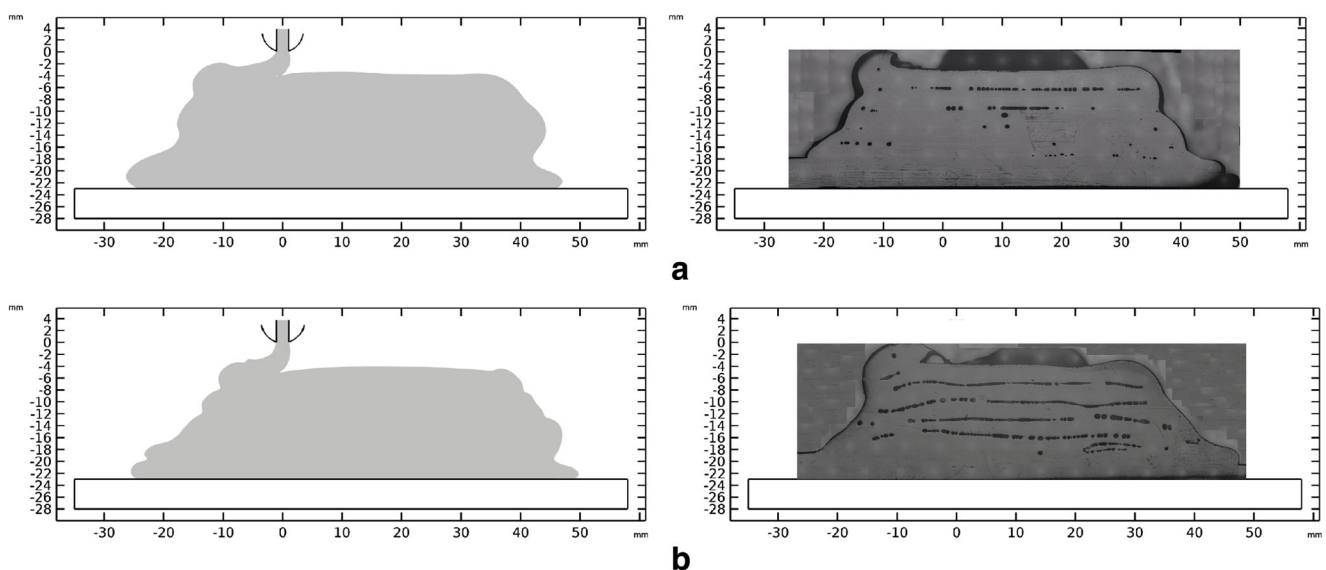

Fig. 13 Comparison of the simulated (left) and measured (right) shape of the final part. **a** $v_s = 5$ mm/s; **b** $v_s = 30$ mm/s

exit of the nozzle in the polymer. They can not be simulated by the proposed method.

4 Conclusion

A numerical method was introduced in detail for the simulation of fluid flow with a free surface. In this method, the key idea is that the free surface is simultaneously captured "implicitly" by the level set function and "explicitly" by the advection of the Lagrangian boundary nodes. The proposed method combines the advantages of the two techniques since it preservers the accuracy of the interface position during the movement and facilitates capturing complex topological changes when coalescence between interfaces occurs.

The method is combined with a heat transfer model to simulate the deposition of molten polymer bead onto a substrate and its cooling down in the extrusion-based additive manufacturing process. The accuracy and convergence of the method have been examined by mesh refinement studies. Despite the fact that 2D model has developed for a specific nozzle cross-section shape, the model, however, provides understanding of the various physical processes which occur in FDM/FFF. In addition, the model and the numerical method are capable of simulating the full process deposition of a thin-walled part and examining the effect of various process parameters on the final shape. Compared with experimental observation, the numerical result shows fairly good agreement. Simulations of the fabrication of a larger part are possible. Since the free surface of the part is captured well, the proposed model can be further extended to study the effect of the nozzle radiation on the part during the deposition process [9]. In addition, the thermal field in the contact region between layers may further be utilized for the prediction of the inter-layer bonding strength. 3D models have to be developed in the near future to extend the applications of the model and also improve the agreement of the results with experimental observations in order to improve the process optimization.

Declarations

References

1. Ajinjeru C, Kishore V, Liu P, Lindahl J, Hassen AA, Kunc V, Post B, Love L, Duty C (2018) Determination of melt processing conditions for high performance amorphous thermoplastics for large format additive manufacturing. Add Manuf 21:125–132. https://doi.org/10.1016/j.addma.2018.03.004
2. Amidror I (2002) Scattered data interpolation methods for electronic imaging systems: a survey. https://doi.org/10.1117/1.1455013
3. Borish M, Post BK, Roschli A, Chesser PC, Love LJ, Gaul KT, Sallas M, Tsiamis N (2019) In-situ thermal imaging for single layer build time alteration in large- scale polymer additive manufacturing. In: Procedia manufacturing, vol 34. Elsevier B.V., pp 482–488. https://doi.org/10.1016/j.promfg.2019.06.202
4. Brenken B, Barocio E, Favaloro A (2017) Experience, R.P.S.i.t.A.o., 2017, U.: Simulation of semi-crystalline composites in the extrusion deposition additive manufacturing process. Tech. rep
5. Brenken B, Barocio E, Favaloro A, Kunc V, Pipes RB (2019) Development and validation of extrusion deposition additive manufacturing process simulations. Add Manuf 25(October 2018):218–226. https://doi.org/10.1016/j.addma.2018.10.041
6. Chesser P, Post B, Roschli A, Carnal C, Lind R, Borish M, Love L (2019) Extrusion control for high quality printing on Big Area Additive Manufacturing (BAAM)systems. Add Manuf 28:445–455. https://doi.org/10.1016/j.addma.2019.05.020
7. Comminal R, Serdeczny MP, Pedersen DB, Spangenberg J (2018) Numerical modeling of the strand deposition flow in extrusion-based additive manufacturing. Add Manuf 20:68–76. https://doi.org/10.1016/j.addma.2017.12.013
8. Compton BG, Post BK, Duty CE, Love L, Kunc V (2017) Thermal analysis of additive manufacturing of large-scale thermoplastic polymer composites. Add Manuf 17:77–86. https://doi.org/10.1016/j.addma.2017.07.006
9. Cosson B, Asséko ACA (2019) Effect of the nozzle radiation on the fused filament fabrication process: Three-Dimensional numerical simulations and experimental investigation. J Heat Transfer, 141(8). https://doi.org/10.1115/1.4043674
10. Costa SF, Duarte FM, Covas JA (2015) Thermal conditions affecting heat transfer in FDM/FFE: a contribution towards the numerical modelling of the process: This paper investigates convection, conduction and radiation phenomena in the filament deposition process. Virtual Phys Prototyp 10(1):35–46. https://doi.org/10.1080/17452759.2014.984042
11. Costa SF, Duarte FM, Covas JA (2017) Estimation of filament temperature and adhesion development in fused deposition techniques. J Mater Process Technol 245:167–179. https://doi.org/10.1016/j.jmatprotec.2017.02.026
12. D'Amico A, Peterson AM (2018) An adaptable FEA simulation of material extrusion additive manufacturing heat transfer in 3D. Add Manuf 21:422–430. https://doi.org/10.1016/j.addma.2018.02.021

13. Du J, Wei Z, Wang X, Wang J, Chen Z (2016) An improved fused deposition modeling process for forming large-size thin-walled parts. J Mater Process Technol 234:332–341. https://doi.org/10.1016/j.jmatprotec.2016.04.005

14. Duty CE, Kunc V, Compton B, Post B, Erdman D, Smith R, Lind R, Lloyd P, Love L. (2017) Structure and mechanical behavior of Big Area Additive Manufacturing (BAAM) materials. Rapid Prototyp J 23(1):181–189. https://doi.org/10.1108/RPJ-12-2015-0183

15. Enright D, Fedkiw R, Ferziger J, Mitchell I (2002) A hybrid particle level set method for improved interface capturing. J Comput Phys 183(1):83–116. https://doi.org/10.1006/jcph.2002.7166

16. Gantois R, Cantarel A, Cosson B, Dusserre G, Felices JN, Schmidt F (2013) BEM-based models to simulate the resin flow at macroscale and microscale in LCM processes. Polymer Composites 34(8):1235–1244. https://doi.org/10.1002/pc.22531. https://hal.archives-ouvertes.fr/hal-01687314

17. Kishore V, Ajinjeru C, Nycz A, Post B, Lindahl J, Kunc V, Duty C (2017) Infrared preheating to improve interlayer strength of big area additive manufacturing (BAAM) components. Add Manuf 14:7–12. https://doi.org/10.1016/j.addma.2016.11.008

18. Li L, Bellehumeur C, Gu P (2002) Investigation of bond formation in FDM process 2002 international solid freeform fabrication symposium. https://doi.org/10.26153/tsw/4500

19. Moreno Nieto D, Casal López V, Molina SI (2018) Large-format polymeric pellet-based additive manufacturing for the naval industry. Add Manuf 23:79–85. https://doi.org/10.1016/j.addma.2018.07.012

20. Olsson E, Kreiss G (2005) A conservative level set method for two phase flow. J Comput Phys 210(1):225–246. https://doi.org/10.1016/j.jcp.2005.04.007

21. Olsson E, Kreiss G, Zahedi S (2007) A conservative level set method for two phase flow II. J Comput Phys 225(1):785–807. https://doi.org/10.1016/j.jcp.2006.12.027

22. Osher S, Fedkiw R (2003) Level set methods and dynamic implicit surfaces applied mathematical sciences, vol 153. Springer, New York. https://doi.org/10.1007/b98879

23. Osher S, Sethian JA (1988) Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations. J Comput Phys 79(1):12–49. https://doi.org/10.1016/0021-9991(88)90002-2

24. Roschli A, Gaul KT, Boulger AM, Post BK, Chesser PC, Love LJ, Blue F, Borish M (2019) Designing for big area additive manufacturing. Add Manuf 25:275–285. https://doi.org/10.1016/j.addma.2018.11.006

25. Serdeczny MP, Comminal R, Pedersen DB, Spangenberg J (2018) Experimental validation of a numerical model for the strand shape in material extrusion additive manufacturing. Addit Manuf 24(June):145–153. https://doi.org/10.1016/j.addma.2018.09.022

26. Sethian JA (1996) A fast marching level set method for monotonically advancing fronts. Proc Natl Acad Sci USA 93(4):1591–1595. https://doi.org/10.1073/pnas.93.4.1591

27. Sethian JA (1999) Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science. Cambridge University Press. https://doi.org/10.2277/0521645573. http://hrcak.srce.hr/file/69388?origin=publicationDetail

28. Shah J, Snider B, Clarke T, Kozutsky S, Lacki M, Hosseini A (2019) Large-scale 3D printers for additive manufacturing: design considerations and challenges. Int J Adv Manuf Technol 104(9-12):3679–3693. https://doi.org/10.1007/s00170-019-04074-6

29. Shen H, Pan L, Qian J (2019) Research on large-scale additive manufacturing based on multi-robot collaboration technology. Addit Manuf 100906:30. https://doi.org/10.1016/j.addma.2019.100906

30. Sun Q, Rizvi GM, Bellehumeur CT, Gu P (2008) Effect of processing conditions on the bonding quality of FDM polymer filaments. Rapid Prototyp J 14(2):72–80. https://doi.org/10.1108/13552540810862028

31. Sussman M (1994) A level set approach for computing solutions to incompressible two-phase flow. J Comput Phys 114(1):146–159. https://doi.org/10.1006/jcph.1994.1155

32. Sussman M, Almgren AS, Bell JB, Colella P, Howell LH, Welcome ML (1999) An adaptive level set approach for incompressible Two-Phase flows. J Comput Phys 148(1):81–124. https://doi.org/10.1006/jcph.1998.6106

33. Sussman M, Fatemi E (1999) Efficient, interface-preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow. SIAM J Sci Comput 20(4):1165–1191. https://doi.org/10.1137/S1064827596298245

34. Sussman M, Puckett EG (2000) A Coupled level set and volume-of-fluid method for computing 3D and axisymmetric incompressible two-phase flows. J Comput Phys 162(2):301–337. https://doi.org/10.1006/jcph.2000.6537

35. Thomas JP, Rodriguez JF (2000) Modeling the fracture strength between Fused-Deposition extruded roads 2000 international solid freeform fabrication symposium. https://doi.org/10.26153/tsw/2054

36. Wang Z, Liu R, Sparks T, Liou F (2016) Large-scale deposition system by an industrial robot (I): Design of fused pellet modeling system and extrusion process analysis. 3D Printing and Add Manuf 3(1):39–47. https://doi.org/10.1089/3dp.2015.0029

37. Xia H, Lu J, Dabiri S, Tryggvason G (2018) Fully resolved numerical simulations of fused deposition modeling. Part i: fluid flow. Rapid Prototyp J 24(2):463–476. https://doi.org/10.1108/RPJ-12-2016-0217

38. Xia H, Lu J, Tryggvason G (2018) Fully resolved numerical simulations of fused deposition modeling. Part II – solidification, residual stresses and modeling of the nozzle. Rapid Prototyp J 24(6):973–987. https://doi.org/10.1108/RPJ-11-2017-0233

39. Yardimci MA, Güçeri S (1996) Conceptual framework for the thermal process modelling of fused deposition. Rapid Prototyp J 2(2):26–31. https://doi.org/10.1108/13552549610128206

40. Zhang J, Wang XZ, Yu WW, Deng YH (2017) Numerical investigation of the influence of process conditions on the temperature variation in fused deposition modeling. Mater Des 130:59–68. https://doi.org/10.1016/j.matdes.2017.05.040

41. Zhang Y, Chou K (2008) A parametric study of part distortions in fused deposition modelling using three-dimensional finite element analysis. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture 222(8):959–968. https://doi.org/10.1243/09544054JEM990

42. Zhang Y, Chou YK (2006) 3D FEA simulations of fused deposition modeling process. In: Manufacturing science and engineering, parts a and b. ASME, pp 1121–1128. https://doi.org/10.1115/MSEC2006-21132

43. Zhao L, Khuc H, Mao J, Liu X, Avital E (2018) One-layer particle level set method. Comput Fluids 170:141–156. https://doi.org/10.1016/j.compfluid.2018.04.009