

# Finite volume simulation framework for die casting with uncertainty quantification

Shantanu Shahane\*, Narayana Aluru, Placid Ferreira, Shiv G Kapoor, Surya Pratap Vanka

Department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign, Urbana, Illinois 61801, USA

## ARTICLE INFO

### Article history:

Received 7 September 2018

Revised 21 March 2019

Accepted 16 April 2019

Available online 23 April 2019

### Keywords:

Solidification

Unstructured grid

Finite volume method

Validation

Uncertainty quantification

Casting

## ABSTRACT

The present paper describes the development of a novel and comprehensive computational framework to simulate solidification problems in materials processing, specifically casting processes. Heat transfer, solidification and fluid flow due to natural convection are modeled. Empirical relations are used to estimate the microstructure parameters and mechanical properties. The fractional step algorithm is modified to deal with the numerical aspects of solidification by suitably altering the coefficients in the discretized equation to simulate selectively only in the liquid and mushy zones. This brings significant computational speed up as the simulation proceeds. Complex domains are represented by unstructured hexahedral elements. The algebraic multigrid method, blended with a Krylov subspace solver is used to accelerate convergence. State of the art uncertainty quantification technique is included in the framework to incorporate the effects of stochastic variations in the input parameters. Rigorous validation is presented using published experimental results of a solidification problem.

© 2019 Elsevier Inc. All rights reserved.

## 1. Introduction

Die casting is an important manufacturing process used when high production rates and complex geometries are required to be manufactured. Fig. 1 shows a typical schematic of the die assembly. The die is generally made out of steel consisting of two halves which are separated along the parting line and held in place by multiple ejector pins. Cooling lines are designed in the die to flow a coolant, typically water. The mold cavity is sprayed with a lubricant which helps to control the die temperature and also reduces the sticking of the molten metal to the die during the removal. Then the two die halves are closed and liquid metal is filled in the cavity. The flowing coolant maintains temperature of the die and extracts heat from the molten metal. After solidification, the two halves are separated by sliding along the ejector pins. The last step is shakeout in which the scrap (gates, runners etc.) are separated from the casting and the casting is further cooled to room temperature either by quenching in water or leaving open to air. Cycle times vary from a couple of seconds for small components weighing less than one ounce, to thirty seconds for a casting of several pounds. Aluminum, magnesium and zinc alloys are the most popular materials used in die casting. Die cast products are more commonly used in automotive and housing industries. Several complex processes involving a large number of process parameters affect the final product quality. Due to the recent advances in computing hardware and software, it is now possible to simulate the physics of

\* Corresponding author.

E-mail address: [sshahan2@illinois.edu](mailto:sshahan2@illinois.edu) (S. Shahane).

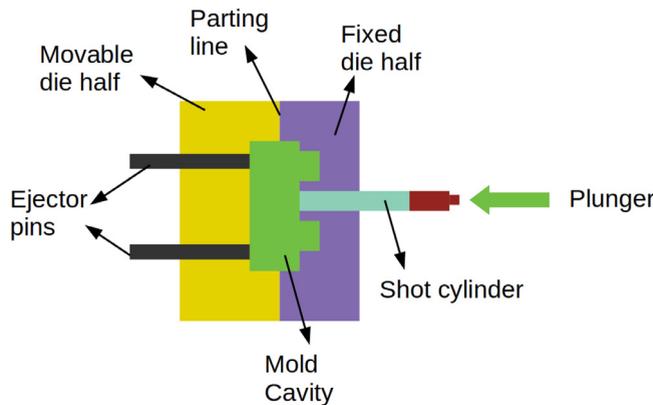


Fig. 1. Die casting assembly schematic.

these processes using numerical simulations. These simulations provide detailed flow and temperature histories and can be used to estimate the final product quality. Frequently, it is difficult to measure and tightly control the process parameters like initial melt temperature, mold temperature and alloy component concentration. However, they can have a significant impact on the process as well as the predicted product strength.

Temperature evolution and velocity distribution during solidification of pure metals and metal alloys has been analyzed by many researchers. [1] used a continuum mixture model to solve the momentum and energy equations and applied it to the solidification problem of a rectangular cavity filled with a binary aqueous solution [2]. From their work, it is clear that although the continuum formulation has a limitation of smearing the interface, it is more efficient than the multiple region method in which the governing equations are solved in each phase separately with appropriate interface conditions. Implementing multiple region method for a practical complex geometry with irregular interfaces is quite difficult and computationally expensive. Voller and Prakash [3] used the Darcy's law of drag and the enthalpy method to model the mushy zone during solidification of a square cavity. Two vertical walls were held below and above the melting point and the horizontal walls were thermally insulated. The damping of velocities in the mushy zone due to Darcy's law is clearly seen in the velocity vector plot. Vynnycky and Kimura [4] solved a similar problem on a rectangular enclosure analytically. They used an asymptotic approach to solve the non-dimensional momentum and energy equations for the case of the Rayleigh and Stefan numbers much larger and smaller than unity, respectively. Comparison with finite element transient numerical simulation showed that such an asymptotic simplification is possible. Bennett [5] used adaptive grid with local rectangular refinement (LRR) to model solidification with fluid flow in multiple rectangular geometries. The adaptive LRR grid needed half the storage and computational effort compared to the non-adaptive grid. Wang et al. [6] discussed a numerical model for melting in a rectangular cavity with natural convection at high Rayleigh number ( $10^8$ ). They used the consistent update technique (CUT) algorithm with a multigrid solver. Plotkowski et al. [7] simulated analytically and numerically heat transfer and fluid flow for solidification in a rectangular cavity. They used scaling analysis to simplify the governing equations of the mixture model followed by an analytical solution and comparison with a finite volume solution for an Al-Cu binary alloy. Hu et al. [8] studied the three-dimensional phase change problem with natural convection using the Lattice Boltzmann method. They simulated melting in a cubical cavity and in cavities with inner rectangular cylinders and sphere. Cleary et al. [9] used smoothed particle hydrodynamics to model filling and solidification in high pressure die casting. They simulated practical industrial case studies like differential cover, an electronic housing and a door lock plate. Gau and Viskanta [10] performed experiments to understand the importance of natural convection during solidification and melting of pure metal. Quillet et al. [11] and Hachani et al. [12] reported the temperature distribution and macro-segregation of solute with an experimental solidification of a Tin based alloy in an ingot. Such experimental work is useful for calibration and validation of the numerical methods.

Solidification phenomena of die casting involves interplay between heat transfer and flow due to natural convection. All the references discussed above simulated solidification on rectangular or cuboidal geometries and hence, structured Cartesian grids were used. Practical die casting geometries are complex and thus, unstructured hexahedral elements are utilized in this work. Multiple algorithms are discussed in the literature to simulate fluid flow on unstructured grids [13–18]. Strategy implemented in this research is an extension of the work by [17] and [18]. Solution of the fluid flow and the pressure Poisson equation involves a significant computational effort. As the solidification problem proceeds, the volume of liquid zone decreases continuously and it is not necessary to solve for fluid flow in the solid zone. We therefore, have developed a consistent procedure to modify the coefficients of the discretized momentum and pressure Poisson equations in order to satisfy the mass continuity equation. Algebraic multigrid [19,20] and Krylov subspace solvers from the open source library HYPRE [21] are used to solve the linear systems of equations.

For die casting, the microstructure parameters like grain size and dendritic arm spacing are important as they affect the final product quality. Phase field modeling [22–24] is a popular method used to study the evolution of the microstructure during solidification. The phase field method simulates the growth of each dendrite and thus, it is computationally expensive

at the length scale of die cast products. In this research, an empirical relation from the work of [25] is used to estimate secondary dendritic arm spacing. There are various models [26–28] suggested for grain size estimation during solidification. Here, the isothermal crystal growth model [27] is used.

Use of deterministic simulations alone to analyze the engineering systems is incomplete due to the lack of precisely defined input data. Thus, there has been a growing interest [29–38] in coupling uncertainty propagation techniques with the deterministic numerical simulations to estimate the effects of stochastic variations in the input process parameters on the outputs. The polynomial chaos expansion is a popular method used to estimate the relation between input and output parameters. Stochastic Galerkin projection [29–31] and collocation [32–34] are two strategies to estimate the coefficients of the polynomial chaos expansion. Stochastic Galerkin method is an intrusive method since it requires solution of a new set of equations and thus, modification of the underlying deterministic code which becomes a significant additional effort. Hence, recently non-intrusive stochastic collocation methods have been developed which need multiple evaluations of the deterministic simulation at predefined collocation points obtained by sampling from the probability distribution function of the input parameters. Values of outputs estimated at these samples are then used to estimate the coefficients of the polynomial chaos expansion.

In this paper, the traditional fractional step approach is modified to deal with the additional terms in the Navier-Stokes equation due to solidification. The discretized system of equations is altered in a way so as to simulate only in the liquid zone thus, increasing computational efficiency. In order to incorporate the effects of stochastic variations in the input process parameters, a parameter uncertainty propagation module has been developed in conjunction with the deterministic simulations. The method of polynomial chaos expansion is used to estimate the relation between input and output parameters and stochastic collocation is used as a wrapper on the underlying deterministic simulation. The framework has been validated against published experimental results followed by demonstration for solidification of two complex geometries. The novelty lies in the overall idea of coupling uncertainty quantification and sensitivity analysis techniques with the numerical simulation of solidification during die casting. We have shown the importance and advantages of such a coupling for validation and process improvement.

## 2. Numerical model description

### 2.1. Governing equations

Solidification, heat transfer and fluid flow due to natural convection are modeled. It is assumed that there is no macro-segregation during solidification and the metal is solidified at nominal composition. It is further assumed that the solutes do not contribute to buoyancy and the flow is incompressible. Thus, the set of governing equations consists of the standard Navier-Stokes equations with additional terms for solidification [7]. They are:

$$\nabla \cdot \mathbf{u} = 0 \quad (1)$$

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) = \nabla \cdot (\mu \nabla \mathbf{u}) - \nabla P - \frac{\mu}{K} \mathbf{u} - \mathbf{g} \rho \beta (T - T_{ref}) \quad (2)$$

$$K = \frac{\lambda^2 (1 - f_s)^3}{180 f_s^2} \quad (3)$$

Here,  $\mathbf{u}$  is the velocity vector,  $\rho$  is density,  $t$  is time,  $\mu$  is dynamic viscosity,  $\mathbf{g}$  is gravity vector,  $\beta$  is coefficient of thermal expansion,  $P$  is pressure,  $K$  is isotropic permeability of the dendritic array,  $\lambda$  is dendrite arm spacing and  $f_s$  is solid fraction.

To model the effects of natural convection, the Boussinesq approximation is used. This is a valid assumption for problems with moderate density variations in the domain. The fluid is modeled as a constant density fluid except for the additional buoyancy term  $-\mathbf{g} \rho \beta (T - T_{ref})$  in the momentum Eq. (2) [39].

The Darcy drag term  $(\frac{\mu}{K} \mathbf{u})$  represents increased resistance to the flow in the mushy zone. We have used the Blake-Kozeny model (Eq. (3)) which estimates the isotropic permeability ( $K$ ) of the dendritic array. In the liquid region, solid fraction is zero and permeability tends to infinity making the Darcy drag term to go to zero. When solid fraction is unity, permeability tends to zero and thus the coefficient of Darcy drag term goes to infinity. For stability, this coefficient is added to the diagonal term of the discretized momentum equations. As a result, the velocities in the solid region go to zero. In the mushy zone, the drag term reduces the velocities compared to the liquid zone.

The energy equation is written in terms of temperature as:

$$\rho C_p \frac{\partial T}{\partial t} + \rho C_p (\nabla \cdot \mathbf{u} T) = \nabla \cdot (k \nabla T) + \rho L_f \frac{\partial f_s}{\partial t} \quad (4)$$

where,

$$f_s(T) = \begin{cases} 0 & \text{if } T > T_{liq} \\ 1 & \text{if } T < T_{sol} \\ 1 - \left( \frac{T - T_f}{T_{liq} - T_f} \right)^{\frac{1}{p-1}} & \text{otherwise} \end{cases} \quad (5)$$

Here,  $T$  is temperature,  $C_p$  is specific heat,  $k$  is thermal conductivity,  $L_f$  is latent heat of fusion,  $k_p$  is partition coefficient,  $T_f$  is freezing temperature and  $T_{liq}$  is liquidus temperature. Note that the term dealing with advection of latent heat given by  $\rho L_f (\nabla \cdot \mathbf{u})$  [7] goes to zero under the incompressibility assumption. To close the system, the Gulliver–Scheil Eq. (5) [40] for the solid fraction is used.

### 2.2. Solution algorithm

We have developed a new software OpenCast in an object oriented C++ environment. A finite volume method on a collocated grid is used to discretize the governing equations. The fractional step method [41], modified to account for the solidification is used to integrate the equations. First, the momentum Eq. (6) without the pressure gradient term is solved to estimate an intermediate velocity ( $\mathbf{u}^*$ ) field. The buoyancy and Darcy drag terms are included in this step. Second order accurate Crank–Nicolson scheme for the diffusion terms and the Adams–Bashforth scheme for the convection terms are used for temporal discretization. The coefficient of the drag term ( $\frac{\mu}{K}$ ) in the mushy zone is treated fully implicitly as:

$$\rho \frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} + \frac{\mu}{K} \mathbf{u}^* = -Conv(\mathbf{u}^n, \mathbf{u}^{n-1}) + Diff(\mathbf{u}^*, \mathbf{u}^n) + Buoy(T^n) \tag{6}$$

where, the operators *Conv*, *Diff* and *Buoy* represent the discretized convection, diffusion and buoyancy terms respectively. The full momentum equation is similarly discretized with an implicit pressure gradient term, given as:

$$\rho \frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + \frac{\mu}{K} \mathbf{u}^* = -Conv(\mathbf{u}^n, \mathbf{u}^{n-1}) + Diff(\mathbf{u}^*, \mathbf{u}^n) - (\nabla P)^{n+1} + Buoy(T^n) \tag{7}$$

Subtracting Eq. (6) from (7) gives the velocity correction equation.

$$\mathbf{u}^{n+1} = \mathbf{u}^* - (\nabla P)^{n+1} \frac{\Delta t}{\rho} \tag{8}$$

Taking divergence of the velocity correction Eq. (8) and invoking the continuity constraint gives the equation for pressure:

$$\nabla \cdot \left( \frac{\nabla P}{\rho} \right)^{n+1} = \frac{\nabla \cdot \mathbf{u}^*}{\Delta t} \tag{9}$$

The overall solution algorithm to advance from time-step  $n$  to  $n + 1$  is as follows:

1. Solve for  $\mathbf{u}^*$  using Eq. (6). Since the diffusion term is implicit, solution is obtained iteratively
2. Solve the pressure Poisson Eq. (9) iteratively to estimate  $P^{n+1}$
3. Correct the velocities ( $\mathbf{u}^{n+1}$ ) using Eq. (8)
4. Solve the energy Eq. (4) together with solid fraction relation (Eq. (5)) to estimate temperature and solid fraction
5. Estimate micro-structure parameters such as grain size and yield strength using the empirical relations

In this work, we have used the fractional step method instead of the semi-implicit methods like SIMPLE, SIMPLER, PISO [42,43] etc. The fractional step method is computationally efficient when small time steps below unity Courant number are required from accuracy point of view. In this case, the temperature–time history is important to get the microstructure development during solidification.

### 2.3. Discretization on an unstructured grid

Practical die casting geometries are quite complex. Cartesian grids introduce high stair-casing errors near the boundaries. Thus, OpenCast uses unstructured grids with tetrahedral and hexahedral finite volumes. First, a tetrahedral mesh is generated using the open source software GMSH [44]. Tetrahedral elements have higher cross diffusion terms due to mesh skewness compared to hexahedral elements. Thus, we further use an open source tool TETHEX [45] to subdivide the tetrahedrons to hexahedrons.

The governing Eqs. (1), (2) and (4) can be written as a transport equation for a general scalar  $\phi$ :

$$\rho \frac{\partial \phi}{\partial t} + \nabla \cdot (\rho \mathbf{u} \phi) = \nabla \cdot (\Gamma \nabla \phi) + S_\phi \tag{10}$$

where,  $\phi$  is any scalar field,  $\Gamma$  is the diffusion coefficient, and  $S_\phi$  is the source term.

Fig. 2 shows two adjacent hexahedral control volumes sharing a common face with vertices  $V_1, V_2, V_3$  and  $V_4$ .  $C_1$  and  $C_2$  are cell centers and  $f$  is the face center.  $\hat{n}$  is the unit vector normal to face and in an outward direction with respect to cell  $C_1$ .  $\vec{d}$  is the distance vector from  $C_1$  to  $C_2$ . We use a collocated finite volume formulation with all the field variables stored at cell centers.

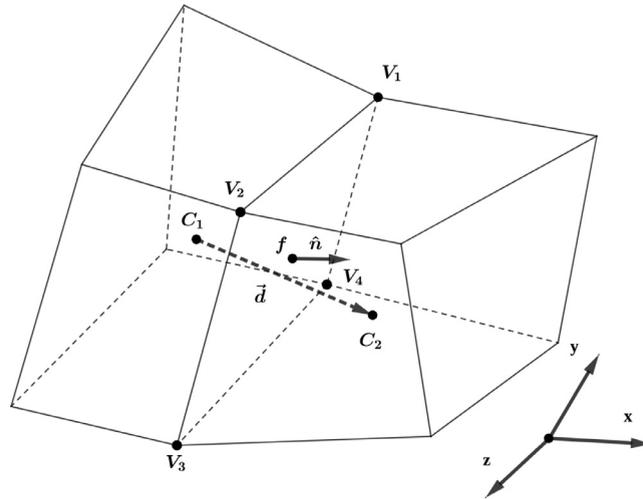


Fig. 2. Unstructured hexahedral control volumes.

The surface integral of the diffusion term is approximated as a summation over all the six faces of the cell. The inner product of the normal and the face centered gradient at each face is split into two terms [17]:

$$\begin{aligned} \hat{n} \cdot \nabla \phi|_f &= \left( \frac{\vec{d} \cdot \nabla \phi|_f}{\hat{n} \cdot \vec{d}} \right) - \left( \frac{\vec{d} \cdot \nabla \phi|_f}{\hat{n} \cdot \vec{d}} - \hat{n} \cdot \nabla \phi|_f \right) \\ &= \left( \frac{\phi_{C_2} - \phi_{C_1}}{\hat{n} \cdot \vec{d}} \right) + \left( \hat{n} - \frac{\vec{d}}{\hat{n} \cdot \vec{d}} \right) \cdot \nabla \phi|_f \end{aligned} \tag{11}$$

The first and second terms of Eq. (11) are direct and cross diffusion terms, respectively. For a structured grid,  $\hat{n}$  is parallel to  $\vec{d}$  and the cross diffusion term is identically zero as the direct diffusion term reduces to the central difference approximation of first derivative at face center.

In order to estimate the face centered gradient, the strategy used by [17] for two dimensional grids is extended. A local co-ordinate system is defined with  $\xi$ :  $(C_1C_2)$ ,  $\eta$ :  $(V_1V_3)$  and  $\zeta$ :  $(V_2V_4)$  (Fig. 2) as the three axes.  $x, y$  and  $z$  are the axes of the global frame of reference. The gradients in both these frames are related by the chain rule of differentiation.

$$\begin{bmatrix} \phi_\xi \\ \phi_\eta \\ \phi_\zeta \end{bmatrix} = \begin{bmatrix} x_\xi & y_\xi & z_\xi \\ x_\eta & y_\eta & z_\eta \\ x_\zeta & y_\zeta & z_\zeta \end{bmatrix} \begin{bmatrix} \phi_x \\ \phi_y \\ \phi_z \end{bmatrix} \tag{12}$$

where, the subscripts denote derivatives. The Jacobian matrix entries come from the co-ordinates of the cell centers and the vertices. Value of  $\phi$  at each vertex is estimated by averaging from the neighboring cells of the vertex. Thus, the face centered gradient  $\nabla \phi|_f = [\phi_x, \phi_y, \phi_z]^T$  is estimated by inverting the Jacobian matrix in Eq. (12) and multiplying by  $[\phi_\xi, \phi_\eta, \phi_\zeta]^T$ .

The surface integral of the convection term is approximated as a summation over all the six faces of the cell. The face value of the field  $\phi$  is estimated by interpolating from the two neighboring cells which share the face. The volume flux passing through the face  $(\hat{n} \cdot \mathbf{u} \Delta A)$  satisfies the discrete continuity equation. The cross diffusion term has to be accounted for in the computation of the volume flux. The details are given in Section 2.4.3.

2.4. Special modifications in solidification regions

For solidification problems, some additional steps are needed in order to handle the extra terms such as the Darcy drag and the latent heat terms in the momentum and energy equations respectively. The velocities in the solid region should go to zero and in the mushy zone, velocities should be significantly lower than the fully liquid region. Simultaneously, the continuity equation has to be satisfied by the face velocities for each control volume. Thus, special care has to be taken in the solution process of the pressure Poisson equation and the velocity correction step.

Fig. 3 shows a typical distribution of phases during the solidification process. For ease of visualization, a two dimensional schematic is shown and the same idea has been generalized to three dimensions. The dotted line shows a solid-mushy zone interface. The control volumes (cells) on the left of the line are solid and on the right are either liquid or in mushy zone. Cells are labeled with tags S:Solid, LM: Liquid or Mushy. The S or LM tag is assigned to each cell based on the temperature at the previous time step and the liquidus and solidus temperatures of the alloy.

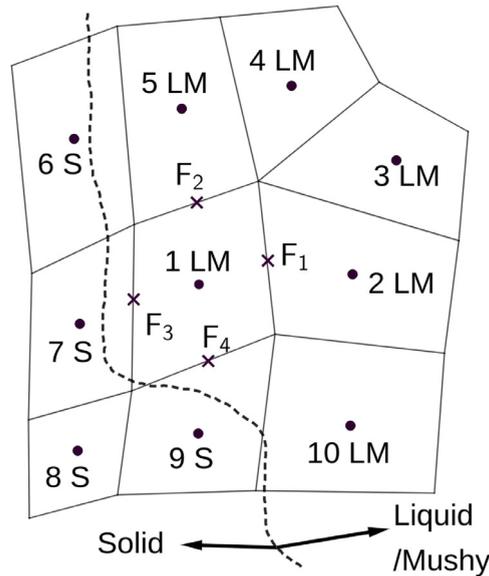


Fig. 3. Control volumes with solid liquid interface.

2.4.1. Momentum equation

As described in Section 2.2, the modified momentum Eq. (6) are solved to estimate the intermediate velocities ( $\mathbf{u}^*$ ). After discretization, the Darcy drag coefficient ( $\frac{\mu}{K}$ ) is added to the diagonal term of the linear equations. In the pure liquid region, this coefficient is zero and thus, it does not have any effect. In the mushy zone, it is finite and non-zero and thus, it acts like a resistance to the flow. In the fully solidified region, it is a large number and thus, the  $\mathbf{u}^*$  tends to zero. From computational efficiency point of view, it is not necessary to solve for  $\mathbf{u}^*$  in the solidified cells since it is zero. Therefore, at each time step, before solving the linearized system of equations, the matrix rows corresponding to solidified cells are removed. As these solidified cells are connected to the neighboring mushy or liquid zone cells, the rows corresponding to the neighboring cells have to be modified in a consistent manner. Consider the row corresponding to cell number 1 in Fig. 3:

$$[A_1, A_2, \dots, A_{10}] [\phi_1, \phi_2, \dots, \phi_{10}]^T = [S_1, S_2, \dots, S_{10}]^T \tag{13}$$

where,  $\phi$  is any component of  $\mathbf{u}^* = [u^*, v^*, w^*]$ . Originally,  $\phi_1$  is connected to all the neighboring cells from 2 to 10. But since cells 6, 7, 8 and 9 are solidified and their velocity is zero, those rows and columns are deleted from Eq. (13). Thus, the reduced row becomes:

$$[A_1, A_2, A_3, A_4, A_5, A_{10}] [\phi_1, \phi_2, \phi_3, \phi_4, \phi_5, \phi_{10}]^T = [S_1, S_2, S_3, S_4, S_5, S_{10}]^T \tag{14}$$

2.4.2. Pressure poisson equation

Similar to the momentum equations, the matrix rows corresponding to solidified cells are deleted from the discrete pressure Poisson equation. But in this case, the rows of the neighboring liquid or mushy cells cannot be updated by just deleting the connections of the solid cells as Neumann boundary conditions have to be applied. For any solidified cell, incoming or outgoing flow through all of its faces should be made zero. This is achieved as follows. Each face is shared by exactly 2 cells (face owners). All the cells which share a common vertex with a face are known as its neighbors. The face centered gradient is computed using the values at all connected neighboring cells. For example, cell numbers 1 and 2 are the owners of face  $F_1$  whereas, cells 1, 2, 3, 4, 5, 9 and 10 are its neighbors. The following cases arise for each face:

1. None of the neighbor cells is solidified: no change in the face centered gradient coefficient (eg. all the faces of cell number 3) is required
2. None of the owners are solidified but at least one neighbor cell is solidified: flow through the face is allowed but the face centered gradient coefficient has to be modified as the solidified cells are removed from the linear set of equation (eg. faces  $F_1$  and  $F_2$ )
3. At least one owner is solidified: flow through the face is blocked;  $\nabla P \cdot \hat{n} = 0$  thus, contribution of this face in the integrated diffusion term ( $\iint_S \Gamma \hat{n} \cdot \nabla P dS$ ) is zero (eg. faces  $F_3$  and  $F_4$ )

Consider the face  $F_2$  for modification of face centered gradient coefficient. Original coefficients for gradient computation at face  $F_2$  which are valid if none of its neighbor cells are solid are given by:

$$\left[ \frac{\partial P}{\partial x} \right]_{F_2} \approx \begin{bmatrix} A_{x1} & A_{x2} & A_{x3} & A_{x4} & A_{x5} & A_{x6} & A_{x7} \\ A_{y1} & A_{y2} & A_{y3} & A_{y4} & A_{y5} & A_{y6} & A_{y7} \end{bmatrix} [P_1 \ P_2 \ P_3 \ P_4 \ P_5 \ P_6 \ P_7]^T \tag{15}$$

Since cell numbers 6 and 7 have solidified, their contribution has to be removed from Eq. (15). Thus, the last 2 columns are deleted and those coefficients are smeared equally in the remaining columns for ex.,  $A_{x1}$  is modified to  $B_{x1} = A_{x1} + (A_{x6} + A_{x7})/5$  and  $A_{y1}$  to  $B_{y1} = A_{y1} + (A_{y6} + A_{y7})/5$ . Since there are 5 cells remaining, the division by 5 is required. After modification, Eq. (15) becomes:

$$\begin{bmatrix} \frac{\partial P}{\partial x} \\ \frac{\partial P}{\partial y} \end{bmatrix}_{F_2} \approx \begin{bmatrix} B_{x1} & B_{x2} & B_{x3} & B_{x4} & B_{x5} \\ B_{y1} & B_{y2} & B_{y3} & B_{y4} & B_{y5} \end{bmatrix} [P_1 \ P_2 \ P_3 \ P_4 \ P_5]^T \tag{16}$$

The steps for modification of the discretized pressure Poisson equation are as follows:

1. Identify the faces with none of the owners solidified but at least one neighbor cell is solidified and smear the coefficients as described above
2. If at least one owner of the face is solidified, set all of its coefficients to zero
3. Loop over all the cells:
  - If none of its face coefficients are modified, its coefficients do not change
  - If at least one of its face coefficients is modified, re-assemble its coefficients

These steps remove the contribution of the solidified cells carefully and reduce the computational effort significantly.

2.4.3. Face volume flux computation and velocity correction

The collocated finite volume formulation uses the face centered volume fluxes ( $\hat{n} \cdot \mathbf{u} \Delta A$ ) in the continuity equation so as to avoid the checker-boarding of pressure. If there is an inconsistency in the numerical formulation of the pressure Poisson equation and the flux computations, there can be a gain or loss of mass and convergence problems. This section describes a consistent method used in the current code to handle solidification.

The volume flux is obtained by taking inner product of the velocity correction Eq. (8) at the face center with face normal and multiplying by face area:

$$\hat{n} \cdot \mathbf{u}^{n+1} \Delta A|_f = \hat{n} \cdot \mathbf{u}^* \Delta A|_f - \hat{n} \cdot (\nabla P)^{n+1} \Delta A|_f \frac{\Delta t}{\rho} \tag{17}$$

$\mathbf{u}^*|_f$  is estimated by averaging the cell values from the two owner cells of the face.  $\hat{n} \cdot (\nabla P)^{n+1}|_f$  is computed exactly in the same way as the regular diffusion term by splitting it into direct diffusion and cross diffusion terms (Eq. (11) with  $\phi = P$ ). The face centered pressure gradient required in the cross diffusion term is estimated by the modified coefficients (Eq. (16)). This volume flux estimate satisfying the discrete continuity equation to a specified tolerance is used in the convection term (Eq. (6)).

The cell centered velocities do not satisfy the discrete continuity equation. They are computed from Eq. (8) and cell centered pressure gradient is estimated by averaging the face centered gradients.

2.4.4. Latent heat term

The Gulliver–Scheil Eq. (5) which relates temperature with solid fraction is a non-linear model. The easiest way to numerically couple this with the energy equation is to model the Gulliver–Scheil equation fully explicitly as a source term. The problem with an explicit approach is that the source term destabilizes the discretized energy equation due to high magnitude of the latent heat coefficient. Thus, we use the source term linearization concept discussed by Patankar [42]. The nonlinear term is split into a linear term and a remainder. The linear term is modeled implicitly and the remainder term is treated explicitly. From numerical stability point of view, the coefficient of the linear term which is added to the diagonal of the matrix should be positive. If the coefficient and the remainder term are functions of the unknown (temperature in this case), the equation has to be solved iteratively till convergence. Similar approach is described by Voller and Swaminathan [46]

The latent heat term of the energy Eq. (4) when integrated over time and control volume gives:

$$\int_V \int_t L_f \frac{\partial f_s}{\partial t} dV dt \approx L_f \Delta V (f_s^{m+1} - f_s^{old}) \tag{18}$$

where, superscripts *old* and *m* denote last time-step value and iteration number respectively. The value of solid fraction in the subsequent iteration ( $f_s^{m+1}$ ) can be estimated from its latest value ( $f_s^m$ ) by a first order Taylor expansion:

$$f_s^{m+1} \approx f_s^m + \left\{ \frac{df_s}{dT} \right\}^m [T_p^{m+1} - T_p^m] \tag{19}$$

Substituting Eq. (19) in (18) gives:

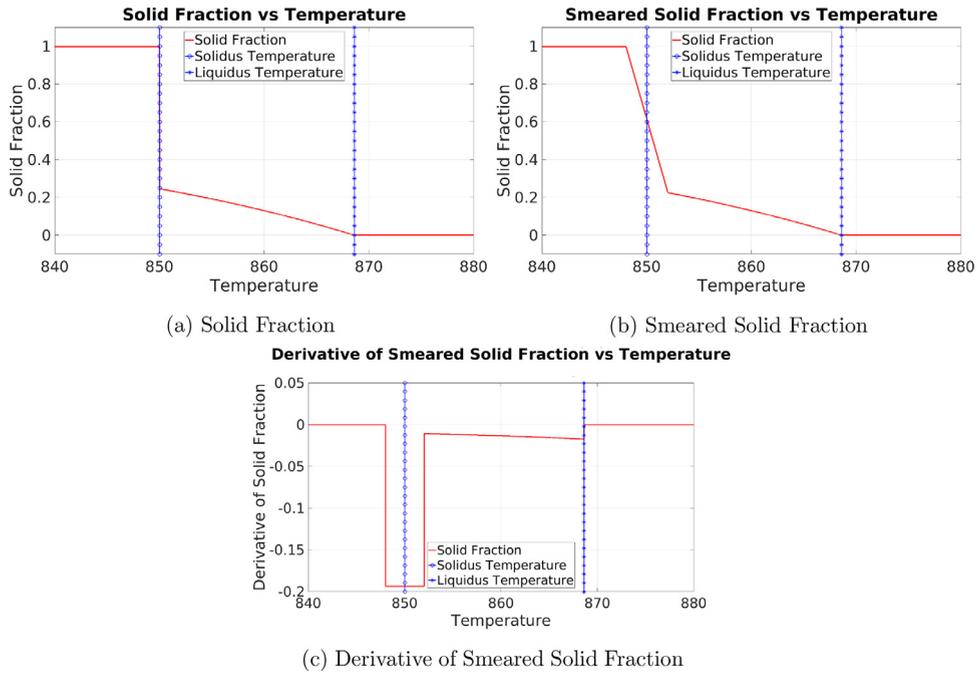


Fig. 4. Solid fraction temperature relation.

$$\begin{aligned}
 \int_V \int_t L_f \frac{\partial f_s}{\partial t} dV dt &\approx L_f \Delta V \left( f_s^m + \left\{ \frac{df_s}{dT} \right\}^m [T_p^{m+1} - T_p^m] - f_s^{old} \right) \\
 &= \left[ L_f \Delta V \left\{ \frac{df_s}{dT} \right\}^m \right] T_p^{m+1} + \left[ L_f \Delta V \left( f_s^m - f_s^{old} - \left\{ \frac{df_s}{dT} \right\}^m T_p^m \right) \right] \\
 &= S_p T_p^{m+1} + S_c
 \end{aligned} \tag{20}$$

$S_p$  and  $S_c$  are functions of last iteration and last time-step values and thus can be computed first. Note that  $S_p$  is always negative and when taken to the left hand side of the equation, it becomes positive and is thus added to the diagonal of the linear system matrix. Adding a positive term to the diagonal helps in stabilizing the system and speeds up convergence. Hence, this approach is found to be much better than the fully explicit method.

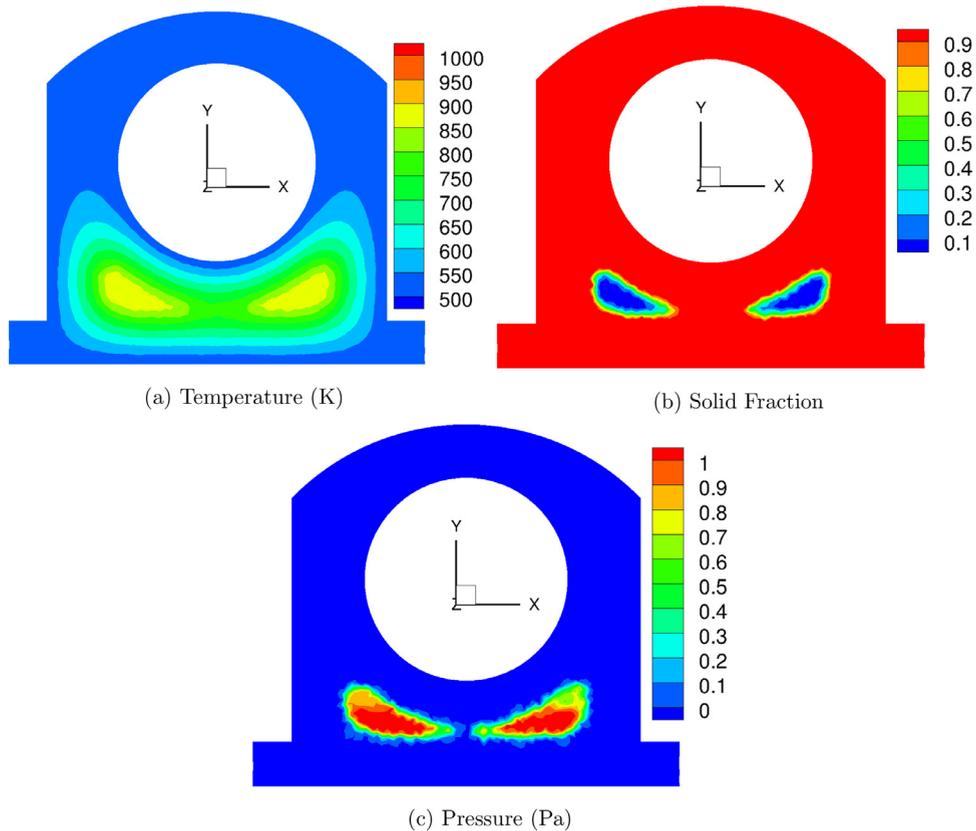
The Gulliver–Scheil Eq. (5) plotted in Fig. 4a for a typical aluminum alloy shows that there is a discontinuity at the solidus temperature. Thus, the derivative  $\frac{df_s}{dT}$  cannot be computed. To deal with this difficulty, the original equation is modified by smearing the discontinuity near the solidus temperature:

$$f_s(T) = \begin{cases} 0 & \text{if } T > T_{liq} \\ 1 & \text{if } T < T_{sol} - T_\epsilon \\ \hat{f}_s - (T - T_{sol} - T_\epsilon) \left( \frac{1 - \hat{f}_s}{2T_\epsilon} \right) & \text{if } T_{sol} - T_\epsilon < T < T_{sol} + T_\epsilon \\ 1 - \left( \frac{T - T_f}{T_{liq} - T_f} \right)^{\frac{1}{k_p - 1}} & \text{otherwise} \end{cases} \tag{21}$$

where,  $\hat{f}_s = 1 - \left( \frac{T_{sol} + T_\epsilon - T_f}{T_{liq} - T_f} \right)^{\frac{1}{k_p - 1}}$  and  $T_\epsilon$  is the width of linear smear which can be set to a reasonable value like 2 K. Thus, the derivative can be computed analytically. Fig. 4b and c plot the modified solid fraction relation and its derivative respectively.

The overall iterative procedure to obtain the variables at the new time step from values at the old time step can be summarized as:

1. Initialize:  $T_p^0 = T_p^{old}$  and  $f_s^0 = f_s^{old}$
2. Compute  $S_p$  and  $S_c$  using last iteration values ( $T_p^m$  and  $f_s^m$ ) by Eq. (20) and solve the linear system of equations to estimate next iteration value  $T_p^{m+1}$
3. Update the solid fraction:  $f_s^{m+1} = (1 - \lambda)f_s^m + \lambda f_s(T_p^{m+1})$  where,  $0 < \lambda \leq 1$  is an under relaxation parameter (Note that  $f_s(T_p^{m+1})$  is the solid fraction evaluated as a function of temperature at iteration  $m + 1$ )
4. Estimate the relative change between the successive iteration values of temperature and solid fraction



**Fig. 5.** Disjoint regions near the end of solidification.

Repeat steps 2–4 until the relative change drops below a desired threshold. For the aluminum alloy used here, it is found that under relaxation is not required i.e.,  $\lambda = 1$  and the solution converges in 5–10 iterations.

### 2.5. Solver for linear systems

Typical die cast geometries have high aspect ratios i.e., thin cross sections compared to the lateral dimensions. It is found that single grid iterative solvers for the elliptic pressure Poisson equation converge slowly for such geometries. Hence, in this work a multigrid solver is used. The central idea of a multigrid solver is to solve the equations on multiple coarse grids and couple the corrections from all the grids through prolongation and relaxation. The high frequency component of the residual converges fast on the fine grids while the coarse grids are used to accelerate convergence of the low frequency residual. Thus, the coarse grid solutions are used to accelerate the convergence while maintaining the discretization accuracy of the solution at the finest level.

Geometric multigrid is a technique in which multiple levels of grids are generated physically and the matrix vector system is estimated by discretizing the governing equations at each level. The main benefit of this approach is that the matrices at all the levels are obtained directly from the governing equations and thus, good convergence is observed. The main drawback is that generating coarse grids for a complex geometry with unstructured elements is non-trivial. Algebraic multigrid (AMG) tries to address this problem by coarsening the matrix using heuristics based algorithms. This is a black box approach which does not need any physical grids at coarse levels.

The BoomerAMG routine along with Krylov solvers of the open source library HYPRE [21] developed at the Lawrence Livermore National Laboratory is used in our work. The AMG solver is observed to solve the modified momentum Eq. (6) and the energy Eq. (4) without difficulty. However, some consistency issues have been observed during the solution of the pressure Poisson Eq. (9) when extensive solidification happens. In complex geometries, as solidification proceeds, there can be disjoint pockets of metal which are yet to solidify. Fig. 5 shows an example of the disjoint regions formed near the end of solidification. The area with solid fraction of unity (red in Fig. 5b) is fully solidified and thus, has zero velocity and the pressure is also set to zero. The blue regions indicate liquid/mushy zones which are yet to solidify. These regions are solved with zero normal velocities on their boundaries. Each region is solved with Neumann pressure boundary conditions corresponding to fixed velocities on the boundary. We have checked that the sums of the local sources/sinks of the pressure Poisson equation for individual regions are zero. Thus, the pressure Poisson equation is well formulated whether solved sep-

arately for each domain or as a single linear system. As these pockets are far enough from each other, they are decoupled numerically in the discrete reduced pressure Poisson equation (Section 2.4.2). This is seen to cause convergence difficulties with AMG. The AMG solver has been coupled with a single grid BiCGSTAB solver, also from HYPRE, to be used when AMG is unable to solve the pressure Poisson equation. This issue arises towards the end of the simulation when only a few cells are liquid (for instance 20%). Thus, the reduced system is much smaller in size compared to the original problem and a single grid solver is reasonably well convergent.

2.6. Grain growth and mechanical properties models

Grain size and Secondary Dendrite Arm Spacing (SDAS) are two important parameters used to characterize the microstructure. OpenCast uses empirical relations from the literature for estimation of microstructure parameters and mechanical properties such as yield strength.

SDAS is predicted based on the empirical relationship proposed by Backer and Wang [25],

$$SDAS = \lambda_2 = A_\lambda \left( \frac{\partial T}{\partial t} \right)^{B_\lambda} \quad [in \mu m]. \tag{22}$$

The model parameters  $A_\lambda$  and  $B_\lambda$  are chosen to be 39.4 and -0.317, respectively based on the model for microstructure in aluminum alloys [25]. Material behavior of the die cast alloy is predicted in terms of 2% yield strength ( $\sigma_{0.2}$ ) using empirical relationship proposed by Okayasu et al. [47].

$$\sigma_{0.2} = A_\sigma \lambda_2^{-1/2} + B_\sigma \tag{23}$$

Here,  $\sigma_{0.2}$  is in MPa,  $\lambda_2$  (SDAS) is in  $\mu m$ ,  $A_\sigma = 59.0$  and  $B_\sigma = 120.3$  [47].

Grain size estimation is based on the work of [27]. The grain growth rate is given by:

$$\frac{dr}{dt} = \frac{\lambda_s^2 D_s}{2r} \tag{24}$$

where,  $r$  is the grain size,  $D_s$  is the solute diffusion coefficient in the liquid and  $t$  is the time. The parameter  $\lambda_s$  is obtained by invariant size approximation:

$$\lambda_s = \frac{-S}{2\pi^{0.5}} + \left( \frac{S^2}{4\pi} - S \right)^{0.5} \tag{25}$$

$S$  is given by

$$S = \frac{2(C_s - C_0)}{C_s - C_l} \tag{26}$$

where,  $C_l = C_0(1 - f_s)^{(k_p-1)}$  is solute content in the liquid,  $C_s = k_p C_l$  is solute content in the solid at the solid-liquid interface and  $C_0$  is the nominal solute concentration. Thus, using the prescribed partition coefficient ( $k_p$ ) and estimated solid fraction ( $f_s$ ), Eqs. (24)–(26) are solved to get the final grain size.

2.7. Parameter uncertainty quantification

The final product quality in die casting is influenced by process parameters like alloy material properties, interface conditions at the mold, thermal boundary conditions etc. Due to the complexity of the process, accurate measurement and control of these parameters is difficult. Conventional deterministic simulations alone are not sufficient to estimate the effect of any stochastic variations on the product quality, thus parameter uncertainty quantification is important. From modeling point of view, parameter uncertainty quantification is a set of stochastic partial differential equations with variation in initial conditions, coefficients and boundary conditions. Stochastic variables are considered as dimensions of the problem in addition to space and time.

To estimate the relation between stochastic process parameters and output parameters, various methods have been proposed in the literature. A popular method is to use a linear combination of polynomial basis functions in the stochastic dimension to expand the output variables. Since orthogonality helps in convergence, orthogonal polynomials are used as basis functions. Wiener’s polynomial chaos [48] with Askey family of orthogonal polynomials leads to optimal convergence of the error. Xiu and Karniadakis [49] determined which polynomial leads to exponential convergence depending on the underlying probability distribution function that the stochastic variable follows. For instance, the Hermite polynomials are orthogonal with respect to the standard normal distribution as the weighting function. Hence, it is recommended to use Hermite polynomials as basis functions if the stochastic variable follows normal distribution. Using generalized polynomial chaos expansion developed by Xiu and Karniadakis [49], a second order random field ( $w$ ) can be expanded by polynomial basis (Eq. (27)). For all practical purposes, the series is truncated to order  $n$ .

$$w(\mathbf{x}, \xi(\theta)) = \sum_{i=0}^{\infty} w_i(\mathbf{x}) \Psi_i(\xi(\theta)) \approx \sum_{i=0}^n w_i(\mathbf{x}) \Psi_i(\xi(\theta)) \tag{27}$$

where,  $w$  is the quantity of interest,  $\mathbf{x}$  is a vector of all the deterministic inputs including space and time (if applicable),  $\Psi_i$  is the multi-dimensional orthogonal polynomial of order  $i$ ,  $\xi$  is the random variable vector  $(\xi_1, \xi_2, \dots, \xi_n)$  and  $\theta$  is an elementary event.

In this work, stochastic collocation method is used to estimate the deterministic coefficients of the polynomial chaos expansion. Collocation is a non-intrusive method and thus, modification of deterministic software is not necessary. It acts as a wrapper around existing deterministic software. The deterministic simulation is run at  $M$  sample points  $(\xi^m)$  and a condition  $w(\mathbf{x}, \xi^m) = w_{sim}(\mathbf{x}, \xi^m)$  is imposed. The right hand side comes from each deterministic simulation and left hand side from polynomial expansion. This gives  $M$  constraints written in a matrix vector form [50].  $M > n + 1$  ensures that the Vandermonde system (Eq. (28)) is overdetermined. Solving in the least-squares sense gives  $[w_0(\mathbf{x}) \ \dots \ w_n(\mathbf{x})]^T$ .

$$\begin{bmatrix} \Psi_0(\xi^1) & \dots & \Psi_n(\xi^1) \\ \vdots & & \vdots \\ \Psi_0(\xi^M) & \dots & \Psi_n(\xi^M) \end{bmatrix} \begin{bmatrix} w_0(\mathbf{x}) \\ \vdots \\ w_n(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} w_{sim}(\mathbf{x}, \xi^1) \\ \vdots \\ w_{sim}(\mathbf{x}, \xi^M) \end{bmatrix} \tag{28}$$

Sample points  $(\xi^m)$  have to be chosen wisely for successive implementation of stochastic collocation method. For instance, uniformly distributed samples can lead to highly oscillatory basis functions and thus poor convergence. Thus, for one dimensional stochastic problems [50], it is popular to choose roots of the basis orthogonal polynomial as sample points. For multiple stochastic dimensions, a simple idea is to use tensor product of the single dimensional samples. The problem with tensor products is that the number of samples grows exponentially with stochastic dimensions. Each sample corresponds to a deterministic simulation and thus the computational expense grows exponentially. Smolyak [51] came up with an algorithm to reduce number of samples in multi-dimensional space maintaining the accuracy of the interpolation. In this research, sparse grid nodes are taken from the work of of Heiss and Winsc [52]. A MATLAB based tool UQLab developed by Marelli and Sudret [53] is used for post processing the simulation outputs to estimate polynomial chaos coefficients and generate the response surfaces.

The strategy described above is quite general and can be applied to any numerical solution framework. In this case, the output variables ( $w$  of Eq. (27)) are temperature and microstructure parameters like grain size and dendritic arm spacing. The stochastic input parameters are boundary conditions, initial conditions and alloy and material properties. The polynomial chaos expansion with stochastic collocation is combined with the deterministic computational fluid flow and heat transfer solver to estimate the sensitivity and uncertainty propagation.

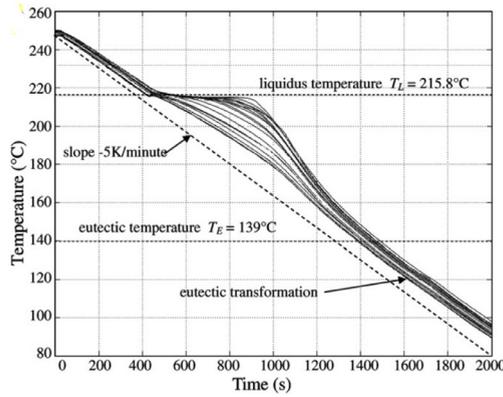
### 3. Validation with uncertainty

Due to the complexity of the die casting process, controlled experiments with accurate temperature measurements inside the casting during solidification are difficult. To validate our code, we have therefore used the experimental results of ingot solidification made of Sn–Bi alloy reported by Quillet et al. [11]. In their experiments, a  $50 \times 60 \times 10$  mm ingot is solidified by cooling one of the  $60 \times 10$  mm faces at a constant rate (5 K/min) and the remaining 5 faces are thermally insulated. Twenty five thermocouples on the ingot were used to measure temperatures during solidification. Due to the low cooling rate, natural convection velocities were significant and their effect can be seen from the curved contours of temperature. Thus, this validation study covers all the aspects of the software including heat transfer, natural convection and solidification.

As a first step, validation was attempted without incorporating the effects of uncertainty. Fig. 6a plots the experimental measurements of temperature from the 25 thermocouples as a function of time beginning from solidification. Fig. 6b is the computed plot assuming a constant conductivity at an averaged temperature, while Fig. 6 is the corresponding plot with temperature dependent thermal conductivity. It is seen that both the computed deterministic thermal histories do not compare well with the experimental data.

Local values of the thermo-physical properties such as thermal conductivity and density depend on grain structure. At the length scale of current simulation, it is not possible to predict the grain structure and hence the properties from first principles. Thus, there can be some uncertainty in the input properties. Also, when the alloy solidifies and cools, there is a thermal contraction. This creates a gap between the mold wall and the casting and adds thermal contact resistance, reducing the amount of heat extracted. Further, the experimental measurements are also subject to sensor noise. For example, the thermocouples used by Quillet et al. [11] have a reported accuracy of 1K, thereby adding another uncertainty in the predictions. Thus, the validation must account for these uncertainties. We have therefore included the stochastic variation in the boundary conditions and material properties and propagated them using the stochastic collocation method. A confidence interval is estimated about the mean output parameters and the experimental and numerical predictions are compared within the confidence interval.

The specific heat plays a role only in the energy equation. When the energy equation is divided by product of density and specific heat, the thermal diffusivity  $(\frac{k}{\rho C_p})$  is a coefficient of the diffusion term and the ratio  $(\frac{L_f}{C_p})$  is the coefficient of the latent heat term. Thus, introducing uncertainty in any two parameters is sufficient. Hence, thermal conductivity and latent heat are chosen to introduce uncertainty. Additionally, density is also modeled as a stochastic quantity since it is a part of the momentum equations. Considering stochasticity in specific heat is not required as it does not add any new information



(a) Experiment [11]

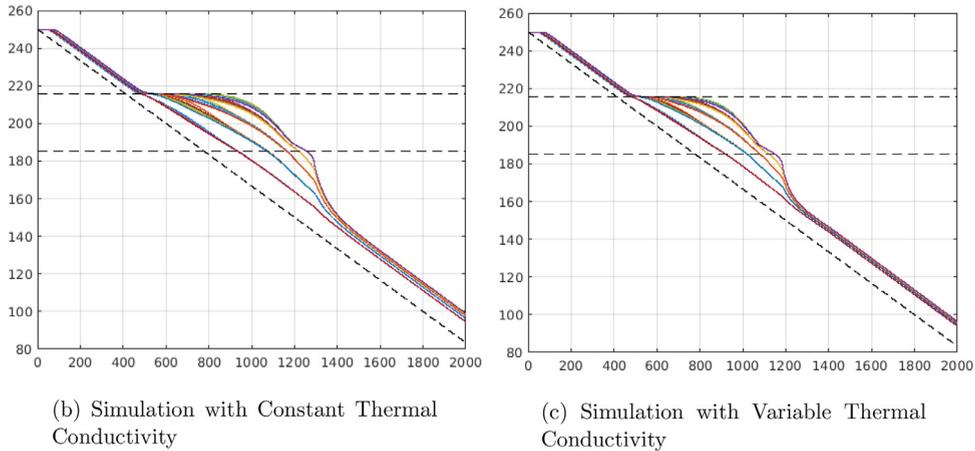


Fig. 6. Temperature-time plot for cooling rate 5 K/min.

to the problem. Uncertainty is therefore added only to wall temperature, latent heat, density and thermal conductivity. Since it is difficult to estimate the thermal contact resistance, the wall temperature is specified as a Dirichlet boundary condition by adding an offset to the cooling rate of 5 K/min. The offset is estimated from the experimental temperature plot (Fig. 6a). In order to take into account the errors in the temperature measurement, an uncertainty is added on top of the offset. All the four input stochastic parameters are assumed to follow a normal distribution with mean ( $\mu_0$ ) and standard deviation ( $\sigma_0$ ) as follows:

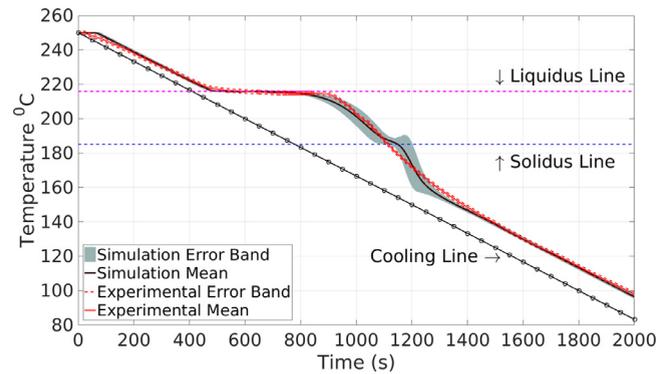
1. Wall temperature offset:  $\mu_0 = 0^\circ \text{ C}$ ,  $\sigma_0 = 0.5^\circ \text{ C}$
2. Latent heat:  $\mu_0 = 60000 \text{ J/kg}$ ,  $\sigma_0 = 2\% \mu_0$
3. Density:  $\mu_0 = 7300 \text{ kg/m}^3$ ,  $\sigma_0 = 2\% \mu_0$
4. Temperature dependent thermal conductivity:  $\mu_0 = [61.282, 57.42, 30.1, 37.7] \text{ W/mK}$ ,  $\sigma_0 = [2.5, 2.5, 2.5, 2.5] \text{ W/mK}$  at temperature [273.2, 373.2, 573.2, 973.2] K

All the thermo-physical properties are estimated as a weighted average of individual properties of Sn and Bi taken from the online version of Kaye and Laby [54]. These uncertainties are propagated using the stochastic collocation method described in Section 2.7.

In order to make a comparison with the experimental data, the experimental temperature-time data plot (Fig. 6a) is digitized. Since it is difficult to distinguish between the 25 temperatures, the thermocouple with the highest temperature is used for validation. Stochastic collocation is done with three accuracy levels of sample points in order to study its convergence. For estimating the interpolation error, 60 Latin Hypercube samples are used as test points. Deterministic simulations and polynomial chaos give two independent estimates of the same output parameter at the test points. The non-dimensional error is defined as the root mean square of difference between these two estimates divided by the maximum value of the parameter. First column of Table 1 denotes the accuracy level of sample points. Accuracy level  $l$  integrates polynomials upto order  $2l - 1$  exactly [52]. Second column is the number of sample points in two dimensional Smolyak sparse grid i.e., the number of deterministic simulations required ( $M$  in Eq. (28)). The last column lists the non-dimensional RMS error in computation of the temperature. It can be seen that the error is of order  $10^{-4}$  for all the accuracy levels and it decreases with increasing level thus, showing convergence. Hence, level 6 is used for validation.

**Table 1**  
Stochastic collocation convergence analysis.

Accuracy level	# Sample Pts.	Max temp. RMS error
4	137	7.27E-4
5	385	4.11E-4
6	953	2.60E-4



**Fig. 7.** Experimental and simulated temperatures with error bounds.

**Table 2**  
Clamp geometry: comparison with and without natural convection.

Output	With natural convection	Without natural convection
Total Solidification Time (s)	1.64	1.64
Max SDAS ( $\mu\text{m}$ )	10.30	10.26
Max Grain Size ( $\mu\text{m}$ )	18.34	17.51
Min Yield Strength (MPa)	138.72	138.75

The Polynomial-Chaos-Kriging (PCK) module of UQLab [53] is used to estimate the output (temperature–time history) as a function of the four input stochastic parameters. Kriging is a Gaussian process modeling tool based on stochastic interpolation algorithm to relate the inputs to outputs. PCK combines the polynomial chaos and kriging methods in a way that is more efficient than the individual methods.

Fig. 7 plots the maximum temperatures (with error bands) from OpenCast simulations and experiments [11] together with cooling, solidus and liquidus lines. A  $2\sigma$  confidence interval about the mean is plotted as the simulation error band. Since the accuracy of experimental temperature measurement is 1 K [11], an experimental error band is plotted as  $\pm 1$  K about the mean value. Before the solidification begins, the temperature drops at a rate similar to the cooling line (5 K/min). Thus, the temperature curve and cooling line are parallel. As mentioned before, an offset is added to the cooling line and set as a temperature boundary condition. For first 60 s, adding offset takes temperature above  $250^\circ\text{C}$  which is the initial temperature. Thus, to avoid heating, boundary temperature is limited to  $250^\circ\text{C}$ . That is the reason for the slight mismatch between the experimental and numerical estimates during the first 60 s. At the liquidus line, a plateau is observed in the temperature curve. This is due to the latent heat release inside the plate. In the region from 60 s to the mushy zone before solidus line, it can be seen that the experimental and numerical temperatures with error bars overlap. Near the solidus temperature, it can be seen that the numerical simulation shows a small kink in the temperature curve whereas, the experimental curve drops smoothly. We believe this may be due to the numerical smearing of the solid fraction temperature relation (Eq. (21)). This effect starts near the solidus line and continues for around 200 s beyond which again the curves overlap. Thus, except for the initial 60 s and near the solidus line, the agreement is good.

This study validates solidification with the fluid flow due to natural convection and heat transfer models implemented in OpenCast since all these phenomena occur in the experiments. It should however be noted that the microstructure parameters and yield stress empirical models are directly taken from the literature. Validation of these models is normally done by conducting solidification experiments.

#### 4. Deterministic solidification results of realistic geometries

In order to demonstrate the utility of the developed software, some complex die casting geometries were simulated. Most of the die casting geometries have thin cross section and thus, the solidification times are of the order of s. For any geometry, before starting uncertainty quantification, simulations are performed for two cases: with and without natural convection and a comparison is made to assess the effect of natural convection. Table 2 lists down various outputs for the clamp geometry

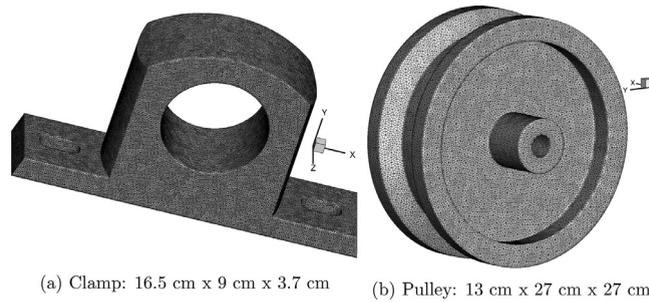


Fig. 8. Geometries and the hexahedral grids.

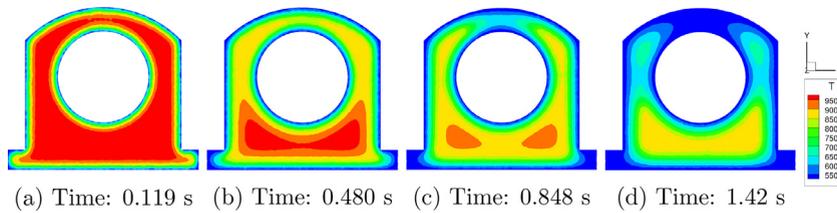


Fig. 9. Clamp temperature contours (K) (Solidification time: 2.40 s).

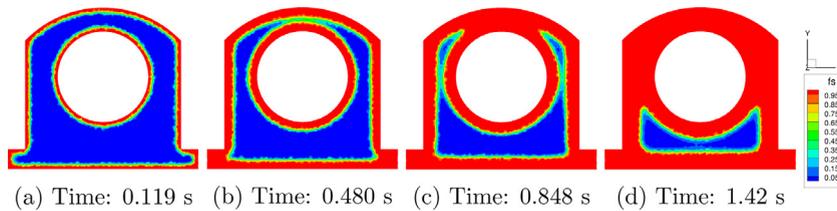


Fig. 10. Clampsolid fraction contours (Solidification Time: 2.40 s).

simulated with and without natural convection. The maximum and minimum are taken over the entire physical domain. The same outputs are used for uncertainty quantification in this work. It can be seen that the effect of natural convection is insignificant. Same thing is observed for the pulley geometry. Thus, simulating without natural convection is acceptable as it saves significant computational effort. Hence, in all the simulations of this section, natural convection was neglected. However, in other processes such as sand casting, natural convection can be important. Therefore, OpenCast is validated with natural convection so that it can be applied for simulation of other casting processes. Note that the validation is performed for a Sn–Bi alloy whereas, the subsequent results presented here are for an aluminum alloy. Aluminum alloys are the most common in die casting and thus, such an alloy is used for deterministic and stochastic simulations in this paper. For validation, it is difficult to measure temperatures inside the casting during solidification of die casting. Thus, we used experimental solidification data from the literature which was available for a Sn–Bi alloy. Since the focus of validation is on the temperature evolution with time, the alloy used is immaterial as long as appropriate material properties are specified in the simulation.

Fig. 8 shows the hexahedral grids of two selected geometries. Both the clamp and the pulley have approximately 300,000 control volumes each. Initially, the mould is filled with a liquid alloy at 1000 K and the walls are held at 500 K. Aluminum alloys with around 8–12% Silicon content are popular in die casting. Thus, for this simulation, an Al–10%Si binary alloy is chosen. The liquidus, solidus and freezing temperatures are taken from the phase diagram.

Fig. 9 plots the temperature contours for the clamp geometry along the mid-plane in Z-direction at different time-steps during the solidification. Fig. 10 shows the corresponding solid fraction contours. Fig. 11 plots micro-structure parameters SDAS, grain size and yield strength estimated using the previously described empirical models. As time progresses, the cooling rates and temperature gradients decrease. Hence, the regions with highest thickness or the core take longer time to solidify. The core regions typically have higher SDAS, grain size and lower yield strength as the grains have longer time to grow. Such trends are seen in Fig. 11. The pulley is axisymmetric in geometry and loading conditions. Thus, SDAS, grain size and yield strength are plotted only along the XY plane as shown in Fig. 12. Even for this case, the core region has higher SDAS and grain size and lower yield strength.

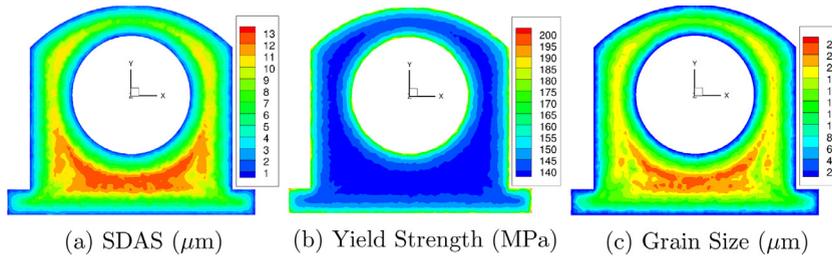


Fig. 11. Clamp microstructure parameters.

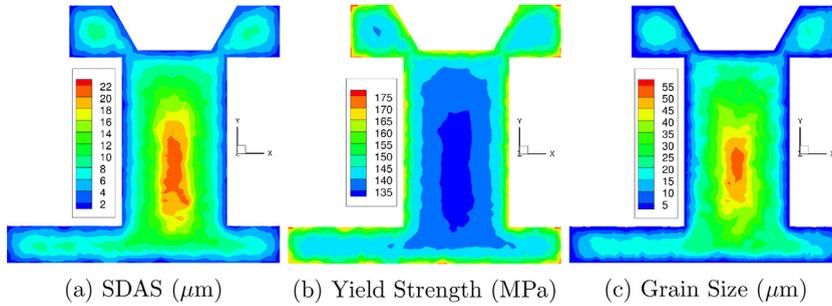


Fig. 12. Pulley microstructure parameters.

Table 3

Stochastic collocation convergence analysis.

Accuracy level	# Sample Pts.	Output 1	Output 2	Output 3	Output 4
3	27	1.20E-2	3.82E-2	2.70E-3	1.61E-2
6	216	7.41E-3	3.35E-2	2.42E-3	1.42E-2

## 5. Uncertainty quantification results

The clamp geometry is subsequently used to study the effect of stochastic variation in the process parameters on the outputs. The process parameters which can be tuned in a die casting foundry are alloy composition (the solute percentage), initial melt temperature and the wall temperature. It is assumed that these three parameters follow normal distribution ( $\mathcal{N}$ ) with the following means ( $\mu_0$ ) and standard deviations ( $\sigma_0$ ):

1. Solute concentration  $C_0 \sim \mathcal{N}(\mu_0 = 10, \sigma_0 = 0.2)$  wt %
2. Initial molten alloy temperature  $T_{init} \sim \mathcal{N}(\mu_0 = 1000, \sigma_0 = 1\% \mu_0 = 10)$  K
3. Wall temperature  $T_{wall} \sim \mathcal{N}(\mu_0 = 500, \sigma_0 = 1\% \mu_0 = 5)$  K

Thermal properties like thermal conductivity, density and specific heat are estimated as a weighted linear combination of the properties of aluminum and silicon with solute concentration as the weight. Since there is a significant variation with temperature [54], temperature varying properties are used. Alloy properties are obtained from the Al–Si phase diagram [55]. The freezing and solidus temperatures are  $660^\circ\text{C}$  and  $577^\circ\text{C}$ , respectively. The solidus and liquidus curves can be approximated by straight lines. The partition coefficient can be estimated as the ratio of liquidus to solidus line slope [40]. The liquidus temperature is calculated from the liquidus line as a function of the solute concentration.

The impact of the three input parameters is studied on the following four outputs:

1. Total solidification time
2. Maximum SDAS
3. Minimum yield strength
4. Maximum grain size

where, the maximum or minimum is taken over the entire domain. The polynomial chaos method with stochastic collocation described in Section 2.7 is used for uncertainty propagation. The error due to stochastic interpolation is estimated using 60 Latin hypercube samples. Estimates of the same output are obtained from the complete simulation and the polynomial chaos expansion (PCE) independently. The difference between these quantities non-dimensionalized by their mean value is defined as the error. Since the errors in estimating all the four outputs reduce by using higher accuracy levels (Table 3), it can be seen that the collocation method has converged. The results presented correspond to the accuracy level 6. In order to get an idea of the interpolation error visually, both the simulation and PCE estimates can be plotted on the same graph.

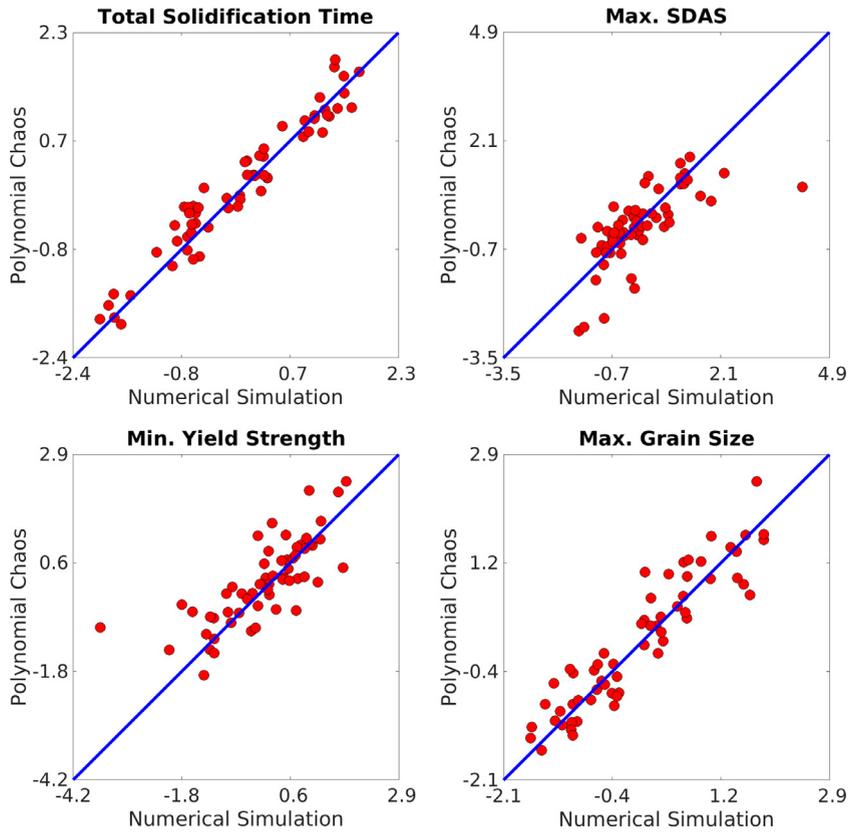


Fig. 13. Comparison of non-dimensional values ( $\frac{\text{value}-\text{mean}}{\text{std. dev.}}$ ) obtained from polynomial chaos expansion and actual numerical simulation estimates.

Table 4  
Clamp geometry: grid independence.

Output	334000 Elements	551000 Elements
Total Solidification Time (s)	2.40	2.42
Max SDAS ( $\mu\text{m}$ )	13.43	13.53
Max Grain Size ( $\mu\text{m}$ )	24.72	23.95
Min Yield Strength (MPa)	136.40	136.34

In the hypothetical scenario of exact interpolation, all the points should lie on the  $Y = X$  line. However, there is a deviation from the line because of the interpolation error. Fig. 13 shows plots for each of the four outputs. Each output is normalized by subtracting its mean and dividing by its standard deviation and thus, is non-dimensional. Most of the points follow the trend of the  $Y = X$  line except some outliers. The outliers correspond to those random samples which are too far from the means of the input parameters. Table 3 and Fig. 13 prove that the polynomial chaos has converged and is accurate enough for further use.

A brief grid independence study is performed in order to make sure that the mesh resolution is sufficient and that the truncation errors are smaller than the uncertainty. Two unstructured hexahedral grids are chosen and the four outputs are evaluated. Table 4 shows that the difference between the coarse and fine grid estimates of all the four outputs is much smaller compared to the final stochastic variation in the outputs.

Sensitivity of each output with respect to each input can be easily estimated once an accurate polynomial chaos expansion is obtained. The sensitivity analysis tool of the software UQLAB [53] is used to estimate the total Sobol indices (Fig. 14). It can be seen that each output is highly sensitive only to one input parameter. Sensitivity is practically important as it gives an idea as to which input parameter should be tightly controlled. Thus, other parameters can be loosely controlled saving cost but yielding desired product quality at the same time. The amount of heat extracted from the wall is proportional to the temperature gradient near the wall and hence, solidification time is highly sensitive to the wall temperature. On the other hand, the microstructure parameters like SDAS, yield strength and grain size depend on the solute concentration. Thus, these three parameters are more sensitive to solute concentration than the wall and initial temperatures.

In the response surface (Fig. 15), the most important input is plotted on the  $x$ -axis. For solidification time response surface, the wall temperature ( $x$ -axis) and initial temperature ( $y$ -axis) are chosen. For other three outputs, solute concentration

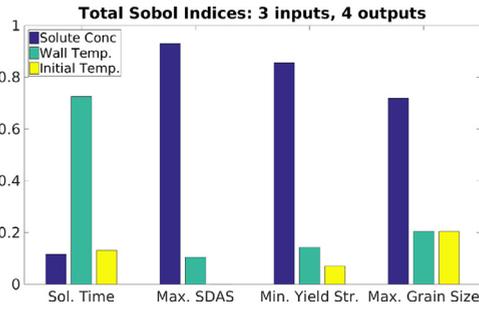


Fig. 14. Sensitivity of 4 outputs to 3 inputs.

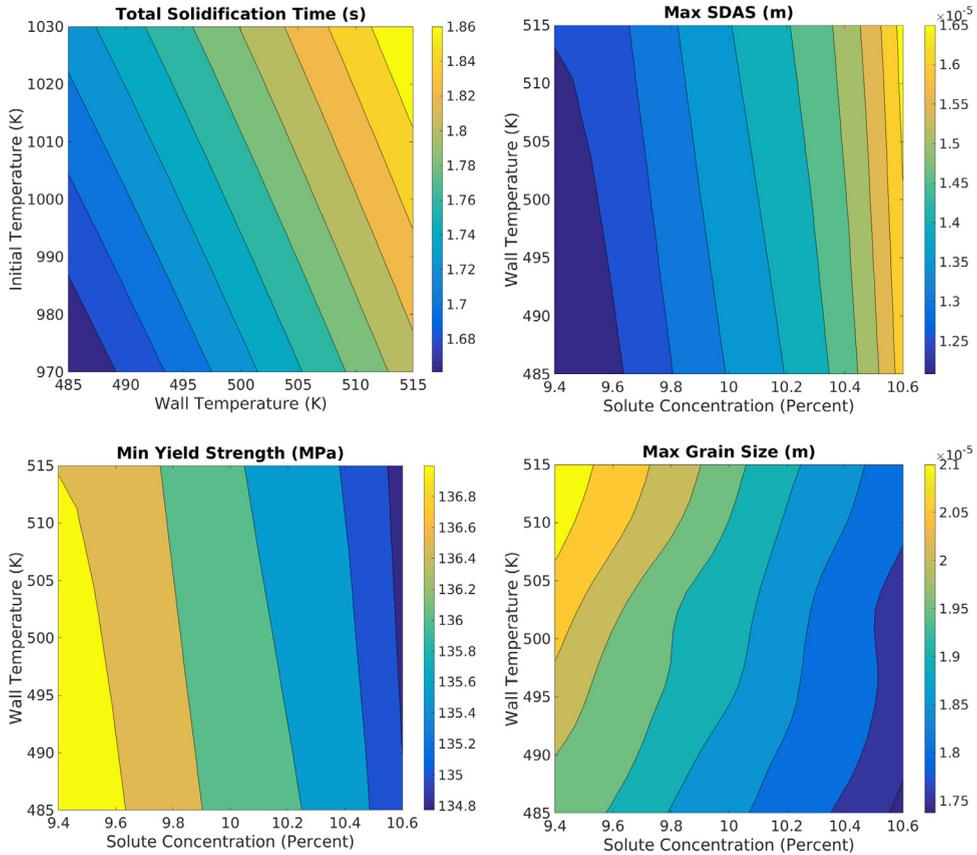


Fig. 15. Response surfaces.

(*x*-axis) and wall temperature (*y*-axis) are chosen. It can be seen that all the response surface contours are nearly vertical. This confirms that the output is most sensitive to the input parameter plotted on the *x*-axis. The maximum grain size contours are non-linear. This implies that the sensitivity varies locally in the input parameter space. For other three outputs, the contours are almost linear and thus, it can be concluded that the local sensitivity is similar everywhere and independent of the input parameter value.

**6. Conclusions**

This paper describes a numerical software framework OpenCast for simulations of solidification problems, including natural convection. Microstructure parameters and mechanical properties are estimated using published empirical relations. The flow equations are solved only in the liquid zone by modifying the coefficients of the discrete equations. The algebraic multigrid solver together with a Krylov subspace solver is used to solve the pressure Poisson equation. Complex geometries are meshed with unstructured hexahedral elements. Parameter uncertainty quantification is used as a wrapper over the deterministic simulations in order to assess the effect of stochastic variations in the inputs on the outputs.

The software is validated against published experimental results of solidification. This validation study shows the significance of stochastic analysis since it is observed that validation without uncertainty is unsuccessful. The validated software is used to simulate two practical die casting geometries. Sensitivity and uncertainty analysis is also performed. Sensitivity analysis shows that the product quality given by grain size and yield strength is highly sensitive to the solute concentration whereas, the productivity given by the solidification time is sensitive to the mold wall temperature. These results are practically useful as it gives an idea about the important input process parameters. The response surfaces show the variation of the outputs with the important input parameters. They can be used to get quick estimates of the outputs without running full deterministic simulations and also to get local sensitivities.

Although this paper demonstrates the ability of OpenCast to simulate die casting problems, it is a general purpose software. OpenCast can also be used to simulate other manufacturing processes such as sand casting, additive manufacturing, welding etc. The use of unstructured elements together with algebraic multigrid method adds complete flexibility to simulate arbitrary geometries. The coupling of uncertainty and sensitivity analysis tools enhance the power of the deterministic numerical method.

## Acknowledgment

Authors acknowledge the financial support from [Digital Manufacturing and Design Innovation Institute \(DMDII Grant No. 15-07-06\)](#). The authors also thank Steve Udvardy and Beau Glim of North American Die Casting Association (NADCA) for providing industry contacts. Technical discussions with Alex Monroe of Mercury Castings and his suggestions were helpful in this work.

## References

- [1] W. Bennon, F. Incropera, A continuum model for momentum, heat and species transport in binary solid–liquid phase change systems I. Model formulation, *Int. J. Heat Mass Transf.* 30 (10) (1987a) 2161–2170.
- [2] W. Bennon, F. Incropera, A continuum model for momentum, heat and species transport in binary solid–liquid phase change systemsII. Application to solidification in a rectangular cavity, *Int. J. Heat Mass Transf.* 30 (10) (1987b) 2171–2187.
- [3] V. Voller, C. Prakash, A fixed grid numerical modelling methodology for convection–diffusion mushy region phase–change problems, *Int. J. Heat Mass Transf.* 30 (8) (1987) 1709–1719.
- [4] M. Vynnycky, S. Kimura, An analytical and numerical study of coupled transient natural convection and solidification in a rectangular enclosure, *Int. J. Heat Mass Transf.* 50 (25–26) (2007) 5204–5214.
- [5] B. Bennett, Adaptive numerical modeling of natural convection, conduction, and solidification within mold cavities, *Numer. Heat Transf. Part A Appl.* 51 (4) (2007) 313–342.
- [6] S. Wang, A. Faghri, T. Bergman, A comprehensive numerical model for melting with natural convection, *Int. J. Heat Mass Transf.* 53 (9–10) (2010) 1986–2000.
- [7] A. Plotkowski, K. Fezi, M. Krane, Estimation of transient heat transfer and fluid flow for alloy solidification in a rectangular cavity with an isothermal sidewall, *J. Fluid Mech.* 779 (2015) 53–86.
- [8] Y. Hu, D. Li, S. Shu, X. Niu, Lattice Boltzmann simulation for three-dimensional natural convection with solid-liquid phase change, *Int. J. Heat Mass Transf.* 113 (2017) 1168–1178.
- [9] P. Cleary, J. Ha, M. Prakash, T. Nguyen, Short shots and industrial case studies: understanding fluid flow and solidification in high pressure die casting, *Appl. Math. Modell.* 34 (8) (2010) 2018–2033.
- [10] C. Gau, R. Viskanta, Melting and solidification of a pure metal on a vertical wall, *J. Heat Transf.* 108 (1) (1986) 174–181.
- [11] G. Quillet, A. Ciobanas, P. Lehmann, Y. Fautrelle, A benchmark solidification experiment on an Sn–10% wt Bi alloy, *Int. J. Heat Mass Transf.* 50 (3–4) (2007) 654–666.
- [12] L. Hachani, B. Saadi, X. Wang, A. Nouri, K. Zaidat, A. Belgacem-Bouzida, L. Ayouni-Derouiche, G. Raimondi, Y. Fautrelle, Experimental analysis of the solidification of Sn–3 wt.% Pb alloy under natural convection, *Int. J. Heat Mass Transf.* 55 (7–8) (2012) 1986–1996.
- [13] L. Davidson, A pressure correction method for unstructured meshes with arbitrary control volumes, *Int. J. Numer. Methods Fluids* 22 (4) (1996) 265–281.
- [14] A. Haselbacher, A grid-transparent numerical method for compressible viscous flows on mixed unstructured grids, 1999 Ph.D. thesis.
- [15] D. Kim, H. Choi, A second-order time-accurate finite volume method for unsteady incompressible flow on hybrid unstructured grids, *J. Comput. Phys.* 162 (2) (2000) 411–428.
- [16] J.H. Ferziger, M. Peric, *Computational Methods for Fluid Dynamics*, Springer Science & Business Media, 2012.
- [17] S. Mathur, J. Murthy, A pressure-based method for unstructured meshes, *Numer. Heat Transf.* 31 (2) (1997) 195–215.
- [18] S. Muzafferija, D. Gosman, Finite-volume CFD procedure and adaptive error control strategy for grids of arbitrary topology, *J. Comput. Phys.* 138 (2) (1997) 766–787.
- [19] U. Yang, V. Henson, BoomerAMG: a parallel algebraic multigrid solver and preconditioner, *Appl. Numer. Math.* 41 (1) (2002) 155–177.
- [20] J. Ruge, K. Stüben, Algebraic multigrid, in: *Multigrid Methods*, SIAM, 1987, pp. 73–130.
- [21] HYPRE: scalable linear solvers and multigrid methods, 2017, Open source library: <https://computation.llnl.gov/projects/hypre-scalable-linear-solvers-multigrid-methods>.
- [22] N. Provatas, N. Goldenfeld, J. Dantzig, Adaptive mesh refinement computation of solidification microstructures using dynamic data structures, *J. Comput. Phys.* 148 (1) (1999) 265–290.
- [23] A. Karma, W. Rappel, Quantitative phase-field modeling of dendritic growth in two and three dimensions, *Phys. Rev. E* 57 (4) (1998) 4323.
- [24] J. Ramirez, C. Beckermann, A. Karma, H.-J. Diepers, Phase-field modeling of binary alloy solidification with coupled heat and solute diffusion, *Phys. Rev. E* 69 (5) (2004) 051607.
- [25] G. Backer, Q. Wang, Microporosity simulation in aluminum castings using an integrated pore growth and interdendritic flow model, *Metall. Mater. Trans. B* 38 (4) (2007) 533–540.
- [26] I. Maxwell, A. Hellawell, A simple model for grain refinement during solidification, *Acta Metall.* 23 (2) (1975) 229–237.
- [27] A. Greer, A. Bunn, A. Tronche, P. Evans, D. Bristow, Modelling of inoculation of metallic melts: application to grain refinement of aluminium by Al–Ti–B, *Acta Mater.* 48 (11) (2000) 2823–2835.
- [28] P. Desnain, Y. Fautrelle, J.-L. Meyer, J.-P. Riquet, F. Durand, Prediction of equiaxed grain density in multicomponent alloys, stirred electromagnetically, *Acta Metall. Mater.* 38 (8) (1990) 1513–1523.
- [29] D. Xiu, G. Karniadakis, Modeling uncertainty in steady state diffusion problems via generalized polynomial chaos, *Comput. Methods Appl. Mech. Eng.* 191 (43) (2002) 4927–4948.

- [30] D. Xiu, G. Karniadakis, Modeling uncertainty in flow simulations via generalized polynomial chaos, *J. Comput. Phys.* 187 (1) (2003) 137–167.
- [31] O. Maitre, O. Knio, H. Najm, R. Ghanem, A stochastic projection method for fluid flow: I. basic formulation, *J. Comput. Phys.* 173 (2) (2001) 481–511.
- [32] M. Carnevale, F. Montomoli, A. D'Ammaro, S. Salvadori, F. Martelli, Uncertainty quantification: a stochastic method for heat transfer prediction using LES, *J. Turbomach.* 135 (5) (2013) 051021.
- [33] P. Marepalli, J. Murthy, B. Qiu, X. Ruan, Quantifying uncertainty in multiscale heat conduction calculations, *J. Heat Transf.* 136 (11) (2014) 111301.
- [34] B. Ganapathysubramanian, N. Zabarar, Sparse grid collocation schemes for stochastic natural convection problems, *J. Comput. Phys.* 225 (1) (2007) 652–685.
- [35] K. Fezi, M. Krane, Uncertainty quantification in modeling metal alloy solidification, *J. Heat Transf.* 139 (8) (2017) 082301.
- [36] S. Hosder, R. Walters, R. Perez, A non-intrusive polynomial chaos method for uncertainty propagation in CFD simulations, in: *Proceedings of the Forty Fourth AIAA Aerospace Sciences Meeting and Exhibit*, 2006, p. 891.
- [37] D. Kumar, M. Raisee, C. Lacor, An efficient non-intrusive reduced basis model for high dimensional stochastic problems in CFD, *Comput. Fluids* 138 (2016) 67–82.
- [38] N. Fajraoui, M. Fahs, A. Younes, B. Sudret, Analyzing natural convection in porous enclosure with polynomial chaos expansions: effect of thermal dispersion, anisotropic permeability and heterogeneity, *Int. J. Heat Mass Transf.* 115 (2017) 205–224.
- [39] E. Spiegel, G. Veronis, On the Boussinesq approximation for a compressible fluid., *Astrophys. J.* 131 (1960) 442.
- [40] J. Dantzig, C. Tucker, *Modeling in Materials Processing*, Cambridge University Press, 2001.
- [41] F. Harlow, J. Welch, Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface, *Phys. Fluids* 8 (12) (1965) 2182–2189.
- [42] S. Patankar, *Numerical Heat Transfer and Fluid Flow*, CRC Press, 1980.
- [43] R. Issa, Solution of the implicitly discretised fluid flow equations by operator-splitting, *J. Comput. Phys.* 62 (1) (1986) 40–65.
- [44] C. Geuzaine, J. Remacle, Gmsh: a 3-D finite element mesh generator with built-in pre-and post-processing facilities, *Int. J. Numer. Methods Eng.* 79 (11) (2009) 1309–1331.
- [45] M. Artemiev, TETHEX: conversion from tetrahedra to hexahedra, 2015, open source library: <https://github.com/martemyev/tetex/wiki>.
- [46] V. Voller, C. Swaminathan, ERAL source-based method for solidification phase change, *Numer. Heat Transf. Part B Fundam.* 19 (2) (1991) 175–189.
- [47] M. Okayasu, S. Takeuchi, M. Yamamoto, H. Ohfuji, T. Ochi, Precise analysis of microstructural effects on mechanical properties of cast ADC12 aluminum alloy, *Metall. Mater. Trans. A* 46 (4) (2015) 1597–1609.
- [48] N. Wiener, The homogeneous chaos, *Am. J. Math.* 60 (4) (1938) 897–936.
- [49] D. Xiu, G. Karniadakis, The Wiener–Askey polynomial chaos for stochastic differential equations, *SIAM J. Sci. Comput.* 24 (2) (2002) 619–644.
- [50] R. Smith, *Uncertainty Quantification: Theory, Implementation, and Applications*, 12, SIAM, 2013.
- [51] S. Smolyak, Quadrature and interpolation formulas for tensor products of certain classes of functions, in: *Doklady Akademii Nauk*, 148, Russian Academy of Sciences, 1963, pp. 1042–1045.
- [52] F. Heiss, V. Winschel, Likelihood approximation by numerical integration on sparse grids, *J. Econom.* 144 (1) (2008) 62–80.
- [53] S. Marelli, B. Sudret, UQLab: a framework for uncertainty quantification in MATLAB, in: *Vulnerability, Uncertainty, and Risk: Quantification, Mitigation, and Management*, 2014, pp. 2554–2563.
- [54] G. Kaye, T. Laby, *Tables of Physical and Chemical Constants: and Some Mathematical Functions*, Longmans, Green and Company, 1921.
- [55] J. Murray, A. McAlister, The Al–Si (Aluminum–Silicon) system, *Bull. Alloy Phase Diagr.* 5 (1) (1984) 74.