Towards the modelling of fluid-structure interactive lost core deformation in high-pressure die casting

S. Kohlstädta,b,∗, M. Vynnyckya, J. Jäckelb

aDivision of Processes, Department of Materials Science and Engineering, KTH Royal Institute of Technology, Brinellvägen 23, 100 44 Stockholm, Sweden
bVolkswagen AG - Division of Components Manufacturing, Dr. Rudolf-Leiding-Platz 1, 34225 Baunatal, Germany

Abstract

This paper investigates the fluid-structure interaction that would be expected to occur when a lost core deforms in high-pressure die casting. A two-phase compressible Volume of Fluid approach is used to model the fluid. The turbulence contribution to the Navier-Stokes equations is accounted for by using the Reynolds-averages Navier Stokes (RANS) Menter SST $k-\omega$ model, whilst an isotropic linear elastic model is assumed for the core material itself. The computed results for the core deformation were compared to those obtained for test bodies manufactured by high-pressure die casting, and good agreement was found.

Keywords: compressible two-phase flow; fluid-structure interaction; high-pressure die casting; lost salt cores; solver development; experimental validation

1. Introduction

High-pressure die casting (HPDC) is an important process for manufacturing high volume and low cost automotive components, such as automatic transmission housings, crank cases and gear box components [1–3]. Liquid metal, generally aluminium or magnesium, is injected through complex gate and runner systems and into the die at high speed, typically between 50 and 100 ms$^{-1}$ at ingate, and under pressures as high as 100 MPa [4].

From an economic point of view, the process is typically constrained by a huge base investment in machinery and tooling, but, on the other hand, low incremental costs for each additional unit produced. In other words, it scales very well with increasing output. However, this complicates the task for the design engineer, who has to be sure about the viability of a process and part beforehand, i.e. before the budget is invested in tooling and machinery. One

∗Corresponding author
Email address: skoh@kth.se (S. Kohlstädtt)

Preprint submitted to Elsevier March 28, 2019
technological constraint to date is that, so far, no serial production is in place where foundries produce high-pressure die casting parts with inlying hollow shapes or undercuts formed by lost cores. Several ideas for such products exist [5, 6], but a serial application is as yet unknown. One material that has been put forward to form these shapes is salt [7, 8].

For several application and process issues, it has to date proven difficult to employ lost salt cores within the process [9]. The basic idea of using salt cores is to block parts of the die volume by inserting cores as placeholders; in so doing, the melt will not penetrate into this space. The cores may then be removed after solidification and one creates undercuts or hollow sections with them, which may then later act as cooling or oil-flow channels [5, 7, 8]. This is still a disadvantage for HPDC, as other casting techniques have employed lost cores for decades [1, 4].

Given this process constraint in design freedom for the CAD-engineer, the idea of using salt as the material for lost cores has been put forward by machine manufacturers, as well as automotive companies [6, 10]. One way to determine whether this is indeed a viable option for a given geometry is to employ numerical simulation - in particular, computational fluid dynamics (CFD) [9, 11?, 12].

However, although some progress has been made using both the commercially available and widely-used MAGMA software [9, 12] and the open source code OpenFOAM [11, 12], primarily as regards determining the magnitude of the forces that a core would be exposed to during die filling, the very real issue of core deformation has yet to be touched upon. To tackle this requires the modelling of fluid-structure interaction, a topic that has already been successfully treated using OpenFOAM in a variety of other contexts [13–15]. Thus, the purpose of this paper is to apply OpenFOAM to the modelling of lost core deformation; this will be done with a combination of solver development and laboratory experiments, which are carried out in order to obtain physical properties for the model and results against which the model can be validated.

2. Model equations and physical properties

2.1. Fluid side

We model the two-phase flow of molten metal and air in high-pressure die casting by using the volume-of-fluid (VOF) method [16], wherein a transport equation for the VOF function, $\gamma$, of each phase is solved simultaneously with a single set of continuity and Navier-Stokes equations for the whole flow field; note also that $\gamma$, which is advected by the fluids, can thus be interpreted as the liquid fraction. Considering the molten melt and the air as Newtonian [17], compressible and immiscible fluids, the governing equations can be written as [18, 19]

$$\frac{\partial p}{\partial t} + \nabla \cdot (\rho \mathbf{U}) = 0,$$

and

$$\frac{\partial (\rho \mathbf{U})}{\partial t} + \nabla \cdot (\rho \mathbf{UU}) = -\nabla p$$
\[ + \nabla \cdot \left\{ (\mu + \mu_{\text{tur}}) \left( \nabla U + (\nabla U)^T \right) \right\} + \rho g + \mathbf{F}_s, \]  \hspace{1cm} (2)\\
\[ \frac{\partial \gamma}{\partial t} + \nabla \cdot (\gamma \mathbf{U}) + \nabla \cdot (\gamma (1 - \gamma) \mathbf{U}_r) = -\frac{\gamma}{\rho_g} \left( \frac{\partial \rho_g}{\partial t} + \mathbf{U} \cdot \nabla \rho_g \right), \]  \hspace{1cm} (3)\\
where \( t \) is the time, \( \mathbf{U} \) the mean fluid velocity, \( p \) the pressure, \( g \) the gravity vector, \( \mathbf{F}_s \) the volumetric representation of the surface tension force and \( ^T \) denotes the transpose. In particular, \( \mathbf{F}_s \) is modelled as a volumetric force by the Continuum Surface Force (CSF) method [20]. It is only active in the interfacial region and formulated as \( \mathbf{F}_s = \sigma \kappa \nabla \gamma \), where \( \sigma \) is the interfacial tension and \( \kappa = \nabla \cdot (\nabla \gamma / |\nabla \gamma|) \) is the curvature of the interface. The term \( \mathbf{U}_r \) is a supplementary velocity field for compressing the phase interface introduced by the solving scheme for the \( \gamma \)-field (MULES) [18, 21]. The material properties \( \rho \) and \( \mu \) are the density and the dynamic viscosity, respectively, and are given by

\[ \rho = \gamma \rho_l + (1 - \gamma) \rho_g, \]  \hspace{1cm} (4)\\
\[ \mu = \gamma \mu_l + (1 - \gamma) \mu_g, \]  \hspace{1cm} (5)\\
where the subscripts \( g \) and \( l \) denote the gas and liquid phases, respectively. We take \( \rho_l, \mu_g \) and \( \mu_l \) to be constant, but assume the air to be a barotropic fluid [22] as it only undergoes isentropic changes [23], i.e. its density changes linearly with pressure, and the process to be isothermal; hence, the equation of state for our model reads

\[ \rho_g = \frac{p}{R_s T} = \psi. \]  \hspace{1cm} (6)\\
\( R_s \) is the specific gas constant and \( T \) is the temperature. Since \( 1/R_s T \) is taken as constant, we introduce the constant compressibility factor \( \psi \) for the gaseous phase. The model parameters are given in table 1. Furthermore, \( \mu_{\text{tur}} \) in equation (2) denotes the turbulent eddy viscosity, which will be calculated via the Menter \( k-\omega \)-SST model [24]. The implementation of this model inside the OpenFOAM framework has previously been shown to be robust and also to give results that are in excellent agreement with experimental data [25].

Additionally to the material properties shown in table 1, the PDEs in equation (1) to (3) require also boundary and initial conditions. The boundary conditions for the fluid and solid side are shown in table 2. The initial conditions for the fluid domain are \( \mathbf{U} = [0 \ 0 \ 0]^T, \ p = \rho_{\text{amb}} \) and \( \gamma = 0 \).

2.2. Solid side

On the solid side, only the stress equation is evaluated for calculating the spatial displacement of the salt core. This process is physically governed by

\[ \rho_s \frac{\partial^2 \mathbf{D}}{\partial t^2} - \nabla \cdot [(2\mu_s + \lambda_s) \nabla \mathbf{D}] = \nabla \cdot \mathbf{q}. \]  \hspace{1cm} (7)\\
In equation (7), \( \mathbf{D} \) represents the displacement vector, \( \rho_s \) denotes the density of the solid, \( \lambda_s \) and \( \mu_s \) denote the solid’s first and second Lamé parameters,
\[
\mathbf{g} = [-9.81 \ 0 \ 0] \ \text{m s}^{-2}
\]
\[p_{\text{amb}} = 10^5 \ \text{Pa}\]
\[U_{\text{in}} = 20,30 \ \text{m s}^{-1}\]
\[\mu_g = 1.8 \times 10^{-5} \ \text{Pa s}\]
\[\mu_l = 1.62 \times 10^{-3} \ \text{Pa s}\]
\[\rho_l = 2520 \ \text{kg m}^{-3}\]
\[\psi = 1.893 \times 10^{-5} \ \text{s}^2 \ \text{m}^{-2}\]
\[\sigma = 0.629 \ \text{N m}^{-1}\]

Table 1: Fluid model parameters. The parameters for gas are those for air; those for metal are for the alloy AlSi9Cu3 [12, 17].

respectively, which are related to the Young’s modulus, \(E\), and the Poisson ratio, \(\nu\), by [26, 27]

\[
\lambda_s = \frac{\nu E}{(1 + \nu)(1 - 2\nu)} \quad \mu_s = \frac{E}{2(1 + \nu)},
\]

and \(\mathbf{q}\) is the load per unit area. \(\mathbf{q}\) consists of the pressure and viscous forces from the fluid side (see also section 2.3).

As the experienced reader in this field recognises immediately, this is an isotropic linear elastic model for the solid mechanics. This model is already available inside the fsi-library of the foam-extend toolbox. Therefore, no additional implementation on the solid side had to be taken into account.

Similar to the PDEs on the fluid side, equation (7) also requires boundary conditions. Those are also given in table 2. The initial displacement is \(\mathbf{D} = [0 \ 0 \ 0]^T\).

While the material properties of the aluminium alloy 6400, or AlSi9Cu3 by another name, are rather well documented in literature (see, for example, [17]), the properties of pressed salt cores are not; therefore, three-point bending experiments were conducted in order to measure them. The results of these experiments are shown in figure 1. Three core geometries were considered; in each case, the length of the core was 0.17 m, but the differences lay in their cross-sections: 10 mm \(\times\) 3 mm, 10 mm \(\times\) 5 mm, 10 mm \(\times\) 7 mm. For each core geometry, a batch of 20 cores was tested and the mean value and its standard deviation plotted. The figure shows the average Young’s modulus, \(E_{\text{av}}\), and the average bending strength, \(\sigma_{\text{bend}}^{\text{av}}\), plotted versus the relative density, which is a concept that is often used in the field of powder metallurgy. Although the core is not made from metal, the process of salt-core manufacturing can be considered to be similar to the powder metallurgy route, in which a green-body is formed by the pressing out of powder. Since the machinery capabilities for applying the necessary force are limited, the body will not have the perfect structure: there will still exist voids between the particles, here salt grains, due to friction between the grains. One therefore evaluates the relative density, which is defined
Table 2: Boundary conditions for the numerical simulations

<table>
<thead>
<tr>
<th>boundary</th>
<th>pressure</th>
<th>velocity</th>
<th>γ</th>
</tr>
</thead>
<tbody>
<tr>
<td>inlet</td>
<td>( \mathbf{n} \cdot \nabla p = 0 ) (zeroGradient)</td>
<td>( \mathbf{U} = [U_{in} \ 0 \ 0]^T ) (fixedValue)</td>
<td>( \gamma = 1 ) (fixedValue)</td>
</tr>
<tr>
<td>wall</td>
<td>( \mathbf{n} \cdot \nabla p = 0 ) (zeroGradient)</td>
<td>( \mathbf{U} = \mathbf{0} ) (fixedValue)</td>
<td>( \mathbf{n} \cdot \nabla \gamma = 0 ) (zeroGradient)</td>
</tr>
<tr>
<td>interface</td>
<td>( \mathbf{n} \cdot \nabla p = 0 ) (zeroGradient)</td>
<td>( \mathbf{U} = \mathbf{U}_i ) (movingWallVelocity)</td>
<td>( \mathbf{n} \cdot \nabla \gamma = 0 ) (zeroGradient)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>boundary</th>
<th>displacement</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>bearing</td>
<td>( \mathbf{D} = \mathbf{0} ) (fixedDisplacement)</td>
<td></td>
</tr>
<tr>
<td>interface</td>
<td>( \mathbf{D} = \mathbf{D}_i ), see section 2.3</td>
<td></td>
</tr>
</tbody>
</table>
as \[ \rho_{\text{rel}} = \frac{\rho_{\text{core}}}{\rho_{\text{NaCl}}} \] (9)

where \( \rho_{\text{core}} \) is the density of the core, which is easily calculated by dividing the mass of the core by its volume, and \( \rho_{\text{NaCl}} \) is the density of pure salt. It follows from equation (9) that all cores that were tested have less than perfect density (see figure 1), as \( \rho_{\text{rel}} \) did not reach 1 for any of the cores. The value for \( \rho_{\text{NaCl}} \) was found in literature to be 2165 kg m\(^{-3}\) [29].

Figure 1: Young’s modulus and bending strength plotted against relative density

One sees that, due to the differing shape of the salt grains before pressing, the material properties of the final body differ too, as the standard deviation in figure 1 shows. A weak inverse correlation between the relative density and \( \sigma_{\text{bend}}^{\text{av}} \) was found. For \( E_{\text{av}} \), the picture was not so straightforward. From 3 to 5 mm thickness, an increase in relative density was observed, while the Young’s modulus dropped by 3.5 GPa. Increasing the thickness further to 7 mm improved the relative density, while here the Young’s modulus also increased slightly to 15.2 GPa. However, given the indicated standard deviation for each batch, neither of these observations can be regarded as particularly significant.

It was interesting to observe that the different shapes of the tested cores had an effect on the material properties. Contradictory to the intuitive assumption that lower pressing heights should produce denser salt beams, the relative density increased from 0.9425 to 0.95 when the height of the core was increased from 3 to 7 mm. The core with a thickness of 5 mm lies exactly in the middle of the two. The values of relative density up to 95 % are in good agreement with published data for powder metallurgy, where limits of 95 % are reported for optimal process conditions [28].

As cores with a thickness of 7 mm proved to be more robust during the process handling, the major part of the modelling and testing work focused on this geometry. The conducted experiment indicates that the average maximum
Table 3: Solid model parameters according to the measured values in the experiments

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$E$</td>
<td>$1.5 \times 10^{10}$ Pa</td>
</tr>
<tr>
<td>$\rho_s$</td>
<td>$2056$ kg m$^{-3}$</td>
</tr>
<tr>
<td>$\nu$</td>
<td>$0.21$</td>
</tr>
</tbody>
</table>

bending strength was 11 MPa and a Young’s modulus of 15 GPa. All the tests were done at room temperature. Another important thing to note here is that the material properties of salt cores can differ significantly, if they are not produced via the pressing-sintering route. Yaokawa et al. [30] and Yamada [31] reported strengths up to 42 MPa. However, they tested salt cores that were manufactured by casting and not by pressing and sintering, such as the ones tested in this paper. One should therefore pay attention to the details of the manufacturing process when comparing these results with those in other publications involving salt cores. As far as the material data experiments are concerned, the findings of this paper are in reasonable agreement with previously published material data on pressed salt cores [97], although the former are somewhat smaller than the latter.

Table 3 summarises the values that were eventually used as input for the model.

2.3. Modelling of the interface processes

In all kinds of fluid-structure interactions, the process of what and how data is transferred at the interface is always of interest. Here, the physics involved in the model was shown in the equations above. The process that was chosen here is illustrated in figure 2. The solver of the solid region transfers the incremental displacement ($\delta D_I$) and velocity ($u_I$), while the solver of the fluid side then returns the pressure ($\delta p_I$), as well as the viscous forces ($\delta t_I$).

$\delta p_I$ and $\delta t_I$ are then fed into the equation (7) and force the solid body to deform accordingly. According to the physics described in equation (7), equations (1)-(3) and the process shown in figure 2, the temperature equation is not taken into account in the present model, as it is believed to be of negligible importance due to the short filling times in high-pressure die casting and hence the limited rise or decrease in temperature in the core and the melt, respectively.

3. Solver development and testing

Given the complexity of the physics involved and also the niche nature of the application, the C++ toolbox OpenFOAM [32, 33] was used to implement the model as it is freely available, rather suitable for being extended by the user and very well scalable for industrial application, as extra cores in parallelization do not require additional licenses. The spatial discretization inside the OpenFOAM environment takes place using the finite-volume-method (FVM) [34, 35]. The solver that was used for the simulations in this paper does not yet exist in either
Figure 2: Mechanism of transferring the field data at the fluid-solid interface

the OpenFOAM foundation release or the foam-extend project. However, the foam-extend release offers a toolbox called the fsi library [36].

Inside the fsi library, several solvers are available depending on the intensity of the coupling between the fluid and the solid. Here, as we expect to see rather large deformations, the solver fsiFoam was used. It follows the process chain depicted in figure 3. The names of the steps closely resemble the names given to the methods that are being called while the code is running. It may thus be easier to bridge between the code and the chart. One sees clearly that, at first, the displacement of the flow domain is evaluated, then the flow field according to the displacement and boundary conditions is evolved and then its eventual impact on the stress field in the solid is calculated. The blue loop indicates that this entire process is repeated until the specified residual is reached. After that, the solver moves on to the next time step.

As outlined in the previous section, already the fluid side is a rather complex fluid flow problem. There was thus previously no solver class available that could solve such a problem within the framework of the fsi library; however, it is available as a top-level solver inside the normal OpenFOAM solver toolbox. The first step was therefore to implement the same methodology of the solver compressibleInterDyMFoam into a class that fits the requirements of an fsi fluid solver class. The standard PIMPLE algorithm, a combination of PISO (Pressure Implicit with Splitting of Operator) and SIMPLE (Semi-Implicit Method for Pressure-Linked Equations), was modified for this purpose and an additional step at the beginning was added that solved the transport equation (3) at the beginning of the process chain, i.e. before the momentum predictor. The several process steps of the solution algorithm can be seen in figure 4 where, on the left, the original PISO-algorithm is shown and, on the right, the solution process
Figure 3: Solving process scheme of fsiFoam Solver
inside the newly developed solver class for the fsi library.

![Diagram of PISO algorithm]

Figure 4: PISO algorithm before and after the adjustments

Figure 4 shows basically a magnification on the flow.evolve() box in figure 3, i.e. what is happening when the flow solver inside the fsi solver is working. We see here that a general implementation of the PISO algorithm was modified to include the two-phase module and compressibility. This was done in order to resemble the physics defined by the governing equations inside the solver. The reader that is used to flow charts for stand-alone flow solvers might miss the loop for the proceeding time step. This was purposely left out as the time object inside the code is here no longer controlled by the flow solver, but by the top level fsi solver instead.

This newly coded solver class inside the fsi library was benchmarked against the standard version of the solver compressibleInterDyMFoam with the simple standard damBreak tutorial case (figure 5) with the coupling between the fluid and the solid, as well as with the mesh motion switched off. Figure 5 shows the result of this benchmark study. As the reader can easily see, the results for the phase field are entirely identical in the two pictures, indicating that both the standard compressibleInterDyMFoam, as well as the newly developed solver
class FSI:compInterFluid, produce the same flow field result. The pressure and velocity results were also identical. The flow field was also in excellent agreement with the results of Koshizuka et al. [37, 38].

![Comparison of the solutions obtained for $\gamma$ at 0.4 s using: (a) compressibleInterDyMFoam; (b) FSI:compInterFluid](image)

In addition to this graphical comparison, the solver has also been tested for mass conservation, volume conservation and conservation of $\gamma$ in the process on simple geometries and benchmarked with analytical models. The results of this testing were also in excellent agreement with previously developed codes such as compressibleInterFoam, which has previously been validated [18, 23, 39], and analytical considerations.

4. Results and discussion

The ultimate goal is to simulate the deformation of lost cores during the high-pressure die casting process and determine a priori how much core deformation will be seen in the eventual casting process. For this eventual aim, a simple structured mesh was constructed that resembles the setup of a three-point bending test [40, 41] within a casting mould. The dimensions of this casting part are shown in the technical drawing in figure 6.

One sees that the geometry is 100 mm long in the x-direction, and that it is also the direction of the mean inflow. The cross-section of this channel is a rectangle which is 40 mm wide in y-direction and 5 mm high (z-direction). The core is situated in the middle of the channel in y-direction, half way of the channel’s length. The thickness of the core in x-direction is 7 mm and 10 mm in the y-direction. The longest elongation of the core is in the z-direction. Its proportions make it eligible for applying beam theory [12]. Each mantle face of the core is surrounded by a 4 mm thick aluminium layer.
In order to keep the effects of meshing and spatial interpolation as low as possible, the mesh/geometry shown in figures 7 and 8 was constructed via the `blockMesh` utility of the OpenFOAM library. The `blockMesh` utility only creates meshes out of hexahedra, which are the best possible cells from a numerical point of view.

The different patches in the mesh are also shown in figures 7 and 8. In this depiction, the melt flows from left to right. All the other outer surfaces of the
fluid geometry are closed (walls). In the solid (figure 8), the mantle area acts as the interface between the fluid and the solid where the forces, the point motion and the velocities are being exchanged according to the scheme in figure 2. The reader should note that the chosen geometry does not have an outlet as the real-world die does not have one either. Given the eventually high injection pressure towards the end of the filling time, all residual air inside the die is compressed to a negligible fraction after the process is complete. Regarding the dimensions of this solid part: it is 70 mm long, 10 mm wide and 7 mm thick. It also resembles the dimensions of a test part that was produced in high-pressure die casting, which will be discussed shortly.

Three different meshes were tested in order to check the model for mesh independence. The starting mesh consisted of 19150 cells in the fluid domain which corresponds with cell spacing of around 1 mm. This mesh has 2760 cells in the solid featuring the same cell spacing. The OpenFOAM utility refineMesh, which divides each cell into eight pieces, was used to refine the mesh. The finest mesh therefore had 1225600 cells in the fluid and 176640 cells in the solid. The forces at the interface were already stable and mesh independent for the intermediate mesh with 153200 and 22080 cells in the fluid and solid, respectively. However, the simulations with the intermediate mesh in general proved to be unstable, so the results for the finest mesh are shown. As stated, the results for the forces and the flow pattern were already identical for the intermediate mesh. For stability reasons, the final and illustrated simulations are for the finest mesh with 0.27 mm cell spacing. This mesh data also corresponds with previously published mesh stats on the matter [12? ]. Although, Fuchs
and Körner [42] define an upper cap of 1 mm for mesh independence, which was too coarse for the simulations in this paper. Moreover, all computations were carried out on a workstation with a hexa-core processor of 12 threads (3.0 GHz) and a total of 32 GB random access memory (RAM). The computational time for the simulations with the finest mesh on this architecture was little short of 58 hours or 2.4 days.

Figure 9 shows a picture of the produced simulation result without the core in the upper part and also a picture of the deformation of the core in the lower part. The result is written out for an ingate velocity of $U_{in} = 30 \text{ ms}^{-1}$ and at a time of 0.016 s, shortly after the melt first hits the core. As stated in section 2.1, the die is initially filled with air and melt is flowing in at a constant velocity from the inlet.

It can be observed in figure 9 that the salt core undergoes a deformation in the range of 2.5 mm as soon as the melt hits it, and hence the momentum of the melt gets redirected due to the core blocking the initial channel flow. The core is initially moved in the positive x-direction due to an observable spike in the force when the interface hits the core [12] and is then displaced constantly by the bypassing flow, when the flow pattern around it becomes stable. The flow then bypasses the core, akin to what is known in literature as flow around an obstacle. It eventually hits the back wall of the geometry. There it changes direction, melt accumulates and the main stream now flows back and hits the core from the other side of the interface. It then starts to fill the arms in the z-direction of the geometry until the volume is eventually almost filled up.

As outlined in section 2.3, the linking element between the fluid and the solid in this FSI approach are the forces acting on the core with its pressure and viscous components (compare figure 2). Over all the simulations that were carried out, only the pressure force in x-direction was of significant order of magnitude to impact the core. These findings are also in line with previous papers that treat the core as stiff and model only the fluid side [9, 12]. We
therefore also evaluated the forces at the interface over time. The result of this simulation is shown in figure 10. As stated in table 1, simulations with two different ingate velocities of 20 and 30 ms$^{-1}$ were done and the results of both simulations are shown in figure 10.

As previous research on the subject has shown, the force plot first sees the typical spike in the force once the melt-air interface hits the core for the first time [12]. The melt then bypasses the core and accumulates in the back of the die before it then fills the two arms of the geometry, inverting the force on the core until the cavity is almost full. At this state then the melt is almost at complete standstill. Since most of the force is typically directly derived from the momentum of the melt that has to be altered in its direction, the force vector therefore approaches zero. One can also see these phenomena in the plot of figure 10.

Figure 11 shows the flow pattern as well as the core deformation when the cavity is almost filled up. There one sees that the core is now bent in the other
direction, i.e. towards the ingate and not in the direction of the melt inflow. The reason for this shape is also derivable from the force plot as, after an initial spike in the force, the absolute force on the patch interface points in the other direction. One sees that, at this state of 95 % fill fraction, the core is not bent as strongly as in figure 9 any more. It correlates also with the decreased force according to figure 10. The deformation of the core at the centre is now about 1.1 mm.

Figure 12 shows the filled die (100 % fill fraction) and the corresponding core deformation at this state. The reader sees in the picture that the core is nearly back in its original position. There is almost no more displacement observable in the x-direction, the direction of the mean flow. The reason is that there is no more melt flowing in from the ingate, as the cavity is filled with melt and the melt was modelled as incompressible. Since a simple linear elastic model was used for the solid mechanics, the core reacts to the decrease of the force by returning to its initial shape. That explains why the core is less displaced than at previous time steps. Another previously unseen deformation is, however, observable at this state. One sees that the overall dimensions of the core are slightly smaller than at the previous time steps. The reason for this is the overall increased pressure in the cavity due to it being continuously filled up with melt. The core therefore sees pressure from all sides on the interface patch. This has no effect on the forces which then eventually causes the displacement in one direction as the pressure from all sides is more or less identical. However, the core is compressed via this effect into a slightly smaller cross-section. The lateral displacement of the core centre is reduced to a value of less than 0.4 mm.

The solver is also capable of tracking the history of a specified point in time over the course of a simulation. This functionality was used to see which deformation the core undergoes during the process of die filling. The result of this simulation is shown in figure 13. Here, we see the process time on the
Figure 11: The filling pattern of the melt and core displacement at a fill fraction of 95%; $U_{in} = 30 \text{ ms}^{-1}$

abscissa while the ordinate bears the displacement in metres. Note that all values are multiplied by $10^{-3}$. As this plot is the result of the same simulation, one sees two different ingate velocities of 20 and 30 ms$^{-1}$ in figure 13.

One sees that while looking at the figure that the plot of the displacement follows the force which is impacting on it. Due to the linear material model that we used for the solid, the entire displacement is reversible: as soon as the forces
Figure 12: The filling pattern of the melt and core displacement at a fill fraction of 100 %; \( U_{in} = 30 \text{ ms}^{-1} \)

approaches zero towards the end, the core returns to its original position, i.e. a straight beam. This can be seen from the graph approaching zero as well.

For the sake of validating the solver, an experimental die was constructed and manufactured that resembles the geometry of the meshed body shown in figures 7 and 8. The die was then tested on a high-pressure die casting machine with the same process parameters that had previously been used for the simulation. One result of the so-cast prototypes is shown in its cross-section in figure 14. The casting was also produced with an ingate velocity of 30 ms\(^{-1}\). Its deformation is representative for a batch of 30, which were cast in the experiment.

One sees in this picture that, analogously to the shape that we saw in figure
the core is also symmetrically bent by the inflowing melt. However, one sees that, different from the intuitive assumption that the incoming mean flow would bend the core towards the upper end of the die, the core is bent towards the ingate. Comparing with figure 10 and also evaluating the results of the simulation at 95 % fill fraction, this becomes plausible. The reason for this can be the force changing sign over time. The spike in the force in the positive x-direction occurs at the beginning of the simulation. Afterwards, the sign of the force changes and the core receives load in the negative x-direction. This causes the deformation pattern illustrated in figure 11 as given the linear nature of the model, 100 % of the deformation is reversible as soon as the force vanishes. The displacement in the experiment was measured to be of 1.3 mm and is hence in the same order of magnitude as the CFD simulation at 95 % fill fraction (see figure 11). Although with a corresponding value of 1.1 mm, the CFD results predicts the deformation to be somewhat smaller.

Recent simulations by the authors, albeit without taking into account possible deformation and only considering the fluid domain, also show that the greatest impact on the core is expected to be at the transition from the air phase to the melt phase at the core interface [12]. However, different from what was believed in the previous publication, this is not the largest impact for producing the ultimately deformed shape of the core, as illustrated by the text and the figures in this section.

5. Conclusions

The research work documented in this paper showed how employing the approach of fluid-structure interaction can help the design engineer to solve an industrial problem. As it has been outlined that tooling in the high-pressure die
casting industry can be very costly, the incentives for the development engineer are high to design a process which is stable in the first place. As discussed, the CFD approach here is of fundamental importance as, \emph{a priori}, a conceivable design can be tested for its robustness before money has to be spent for tooling or machinery. As shown here and in previous papers, the results regarding displacement depending on ingate velocity are only valid for this particular setup. For example, if one changes the ingate area, the established connections are obsolete. Transferring the results is thus not possible and analysing a new geometry will always mean having to re-run the solver on it.

In this paper, the fluid-structure interaction concept was combined with a volume-of-fluid model with one compressible phase in the fluid area and a linear-elastic model in the solid area. Both were implemented for this purpose inside the computer-aided engineering (CAE) framework of OpenFOAM. It has been shown that even with this rather simple solid mechanics model, the system can predict the behaviour of the salt core during the die-filling process step of the casting process up to a certain degree of filling. As shown in the results section, the model presented here only predicts the behaviour of the core until a fill-fraction of about 95 \% is reached. Afterwards, physical effects that are beyond the scope of this model drastically gain importance – such as the increase in temperature of the core as well as the transition of the salt material into elasto-plastic behaviour. Both effects are not covered inside the principal model, but are of significant importance towards the end of the filling. Deformation in the CAE model was 100 \% reversible, so the core returned to its original shape as the melt lost momentum and came to a standstill. The core showed plastic deformation in the end, an effect which was previously unseen, i.e. in the three-
point bending tests documented in section 2.2.

One may thus conclude that the presented model is helpful for the design engineer if evaluated at a fill fraction of 95%. It will be of interest for future research undertakings to determine what an inclusion of the effects that have so far been left out into the model presented will yield. One idea would for example be to include temperature into the model. Another is to develop a more sophisticated material model that also captures the plasticity in the end. Brief testings showed that the economical costs in terms of computational time, however, do not so far outweigh the benefits in the prediction, as long as trained engineers with the necessary expertise and experience control the model and evaluate the results. The authors’ recommendation is therefore, in the present circumstances of available computational capacity and complexity of industrial geometries in casting, to stick with the simplified model and have the results interpreted by experienced design engineers.

References


