Discretisation procedures for multi-physics phenomena

C. Bailey*, G.A. Taylor, M. Cross, P. Chow

University of Greenwich, Ctr. Num. Modelling & Process Anal., London SE18 6PF, UK

Received 11 June 1997; received in revised form 2 November 1998

Abstract

Procedures are described for solving the equations governing a multi-physics process. Finite volume techniques are used to discretise, using the same unstructured mesh, the equations of fluid flow, heat transfer with solidification, and solid deformation. These discretised equations are then solved in an integrated manner. The computational mechanics environment, PHYSICA, which facilitates the building of multi-physics models, is described. Comparisons between model predictions and experimental data are presented for the casting of metal components. © 1999 Elsevier Science B.V. All rights reserved.

Keywords: Multiphysics modelling; Finite volumes; Computational fluid dynamics; Computational solid mechanics

1. Introduction

Since the 1950s a number of techniques have evolved which discretise and solve the governing equations of fluid flow, solid mechanics, heat transfer, etc. A computational mechanics analysis begins by breaking down the domain of interest into small elements forming a mesh. Discretisation techniques such as finite elements (FE) and finite volumes (FV) integrate the governing equations over this mesh and, using suitable approximations, provide systems of equations that can then be solved for dependent variables such as temperature, fluid velocities, solid displacements, etc., at specific points within the domain.

The requirement to discretise the governing equations over complex domains represented by unstructured-type meshes can be accommodated by both FE and FV techniques. The finite element method [18], originating in the structures and solid mechanics communities, has proved very successful in predicting a large number of solid mechanics based phenomena and is now being extended to model fluid flow, acoustics, heat transfer, and electromagnetics [18]. The finite volume method

* Corresponding author. E-mail: c.bailey@gre.ac.uk.

0377-0427/99/$– see front matter © 1999 Elsevier Science B.V. All rights reserved.
PII: S0377-0427(98)00236-2
[7, 10], developed from early finite difference techniques, is the dominant method within the computational fluid dynamics (CFD) community. It has gained such popularity, especially for highly nonlinear flow, because of its inherent ability to conserve physical quantities at both a local and global level. Indeed, it may be argued that FV techniques for highly nonlinear flows, such as free surface and multiphase flows, are superior to their FE equivalents because of their local conservation properties.

Current commercial simulation tools concentrate on specific physical phenomena. For example, codes such as ANSYS [1] and LUSAS [8] have proved very successful in predicting structural and solid mechanics phenomena using finite element techniques. In the fluids community codes based on the finite volume method such as PHOENICS [12] and FLUENT [6] are used extensively to simulate a large number of fluids related processes. As long as the dominant effects in the process can be classified as fluid or solid then these types of products and their algorithms are suitable as process modelling tools.

There is now an emerging requirement to simulate multi-physics processes that are governed by a number of interacting physical laws. An example of such a process is the casting of metal components [9] where a cold mould is filled with molten metal, which cools and solidifies. As the cooling progresses thermal gradients build up in the metal. In the liquid regions this promotes thermal convection that will redistribute heat around the casting. Before the onset of solidification the casting is in full contact with its surrounding mould, but as the casting solidifies it begins to pull away from the mould developing a gap. This gap restricts the flow of heat from the metal to the mould and influences the manner in which the cast component solidifies and develops. Such a process requires computational models that must address the following in an integrated manner:

- Thermal convection once the mould has filled.
- Thermal analysis of solidification and evolution of latent heat.
- Deformation and residual stress predictions of solidified component.

One way of representing, and simulating, all the governing physics in this process is to couple current CFD and solid mechanics codes together. In this scenario, FV-based results for thermal profiles are transferred at each time step into an FE-based stress analysis code to obtain predictions of stress magnitudes and cast-mould gap formation [15]. As well as being very time consuming such an approach, possibly using different meshes for the fluid and solid calculations, can lead to large errors in the analysis.

For multiphysics-based phenomena, like the casting process, a unified computational framework is required which can handle the interactions between the governing physics. Lately, finite volume methods have been used to analyse solid mechanics problems on unstructured meshes [3, 9, 16, 17]. Such methods have proved to be as accurate as classical finite elements in analysing both linear and nonlinear materials. Integration of these techniques with CFD algorithms for fluid flow and heat transfer with solidification has been reported in two dimensions [2]. This resulting two-dimensional multi-physics framework has been used to analyse a number of two-dimensional casting test cases providing very good comparisons. This paper describes finite volume procedures for fluid flow, heat transfer, and solid deformation calculations in three dimensions. Details of a multi-physics computational environment capable of solving complex casting simulations in three dimensions is described. Predictions from these models are presented and compared with experimental data for two casting processes.
2. Governing equations

2.1. Computational fluid dynamics (CFD)

The general equations for the conservation of momentum and mass are stated in their vector form below,

\[
\frac{\partial (\rho u)}{\partial t} + \nabla (\rho u u) = \nabla \cdot (\mu \nabla u) - \nabla p + s, \tag{1}
\]

\[
\frac{\partial \rho}{\partial t} + \nabla (\rho u) = 0, \tag{2}
\]

where \( u \) is the velocity vector and \( \mu, \rho, \) and \( p \) are the dynamic viscosity, density and pressure at time \( t \). The source term \( S \) contains the buoyancy and Darcy terms due to temperature changes and solidification [5].

The conservation of energy is given by

\[
\frac{\partial (\rho h)}{\partial t} + \nabla (\rho u h) = \nabla \cdot (k \nabla T) + S_h, \tag{3}
\]

where \( h, k \) and \( T \) are the enthalpy, thermal conductivity and temperature. The source term contains the contribution from latent heat release due to a change of phase:

\[
S_h = \frac{\partial (L \rho f)}{\partial t} - \nabla \cdot (L \rho u f), \tag{4}
\]

where \( L \) and \( f \) are the latent heat and liquid fraction, respectively.

2.2. Computational solid mechanics (CSM)

In tensor form the incremental equilibrium equations are:

\[
\Delta \sigma_{ij,j} = 0, \quad i,j = x,y,z, \tag{5}
\]

where the incremental stress is related to the incremental elastic strain via Hooke’s law

\[
\Delta \sigma = [D] \Delta \varepsilon_{\text{el}}, \tag{6}
\]

where \([D]\) is the elasticity matrix and \( \Delta \varepsilon_{\text{el}} \) are the elastic strains increments. For the deformation of metals the elastic strains are given by

\[
\Delta \varepsilon_{\text{el}} = \Delta \varepsilon - \Delta \varepsilon_t - \Delta \varepsilon_{\text{vp}}, \tag{7}
\]

where \( \Delta \varepsilon, \Delta \varepsilon_t \) and \( \Delta \varepsilon_{\text{vp}} \) are the total, thermal and viscoplastic incremental strains, respectively. The viscoplastic strains are represented by the Perzyna model [11]

\[
\frac{d \varepsilon_{\text{vp}}}{dt} = \lambda \left( \frac{\sigma_{\text{eq}}}{\sigma_y} - 1 \right)^N \frac{3}{2 \sigma_{\text{eq}}} s_{ij}, \tag{8}
\]
where \( \sigma_{eq}, \sigma_y, \lambda, N \) and \( s \) are the von-Mises stress, yield stress, fluidity, strain rate sensitivity and deviatoric stress, respectively. The viscoplastic strain increment is given by

\[
\Delta e_{vp} = \frac{d e_{vp}}{dt} \Delta t,
\]

where \( \Delta t \) is the time step. The total strain increment for infinitesimal strains is

\[
\Delta \varepsilon = [L] \Delta d,
\]

where \([L]\) is the differential operator matrix and \( \Delta d \) are the displacement increments.

3. Discretisation

Fig. 1 details two types of control volume that can be used in the finite volume discretisation procedure. Cell-centred control volumes are the same as the mesh elements used to represent the domain. Vertex based control volumes are built up around each node where the boundary for each control volume is the face that joins the element centre to the centre of the element faces. The governing equations for mass, momentum and heat transfer are integrated using the cell centred approach [5]. The stress equations are discretised over control volumes defined using the vertex centred approach where displacements are stored at the vertices and stresses at integration points on the control volume boundary [3, 9, 16, 17]. For the cell centred approach, velocity components, temperature, pressure, and liquid fractions are stored at the centres of each mesh element making up the control volume.

The general form of the conservation equations for mass, momentum, and energy can be written as

\[
\frac{\partial}{\partial t}(\rho \phi) + \text{div}(\rho \mathbf{u} \phi) = \text{div}(\Gamma \text{grad}(\phi)) + S,
\]

where, from left to right, the terms represent the transient, convective, diffusive and source contributions, respectively. The dependent variable is represented by \( \phi \) which is unity for mass, the velocity component for momentum, and enthalpy for energy. This general equation is integrated over cell-centred based control volumes. Using the divergence theorem for the convection and diffusion terms the resulting integrated equation is

\[
\int_v \frac{\partial}{\partial t}(\rho \phi) \, dv + \int_s \rho(\mathbf{u} \cdot \mathbf{n}) \phi \, ds = \int_s \Gamma \text{grad}(\phi) \mathbf{n} \, ds + \int_v S \, dv.
\]

Linear approximations are used for the transient term and source terms

\[
\int_v \frac{\partial}{\partial t}(\rho \phi) = \frac{\rho \phi_p - \rho \phi_p^{old}}{\Delta t} \Delta V,
\]

where \( \Delta V \) is the volume of the integrated control volume, and \( \phi_p \) is the dependent variable at this control volume.

\[
S_\phi = S_C - S_p \phi \Delta V.
\]
The diffusive and convective terms are approximated at the faces of the control volumes. For the diffusion term we have

\[
\int_s \Gamma \text{grad}(\phi) \cdot n \, ds = \sum_f \Gamma_f A_f \frac{\phi_n - \phi_p}{d_{np}},
\]

(15)

where the summation is over the number of faces. \(\phi_n, A_f\) and \(d_{np}\) are the dependent variables at neighbouring control volume, face area, and distance between centres of neighbouring cells. To obtain accurate values of the diffusion coefficient, \(\Gamma_f\), at the face the harmonic mean [2] is used.

The convection term is again approximated at the cell face, where

\[
\int_s \rho(\mathbf{u} \cdot \mathbf{n}) \phi \, ds = \sum_f \rho_f (\mathbf{u} \cdot \mathbf{n})_f A_f \phi_f.
\]

(16)
As with the diffusion term, care must be taken in approximating the face values. For \( \rho_f \) upwinding is used and to evaluate the normal component of the velocity at the face \((u.n)\), the Rhie–Chow interpolation method is used [5]. The value of \( \phi \) at the face is calculated using arithmetic averaging. Combining the above terms for \( \phi_p \) from the approximations will result in a system of equations relating the dependent variable at each control volume, \( \phi_p \), to its neighbours, \( \phi_n \).

Unlike the flow equations, above, the stress equations are integrated over vertex based control volumes. Integrating the equilibrium equations over a vertex-centred control volume and using the divergence theorem gives

\[
\int_v \Delta \sigma_{ij,j} \, dv = \sum_f \Delta \sigma_{ij} \cdot nA_f = \sum_f [D] \Delta \varepsilon_{el} A_f,
\]

where the stress components \( \Delta \sigma_{ij} \) are approximated at the faces of each vertex based control volume. The displacements are assumed to vary in a linear manner across each mesh element. Using shape functions, the displacement at any point within a mesh element can be written as a combination of the values stored at the nodes of that element [3, 9, 16, 17] therefore

\[
\Delta d = [N] \Delta u,
\]

where \([N]\) is the matrix of shape functions and \( \Delta u \) is the vector of vertex stored displacements. Substituting this equation into the strain displacement relationship, Eq. (9), results in the following approximation for total strain increment within an element:

\[
\Delta \varepsilon = [L][N] \Delta u = [B] \Delta u.
\]

Therefore the elastic strain, \( \Delta \varepsilon_{el} \), may be written as

\[
\Delta \varepsilon_{el} = [B] \Delta u - \Delta \varepsilon_t - \Delta \varepsilon_{vp}.
\]

The equilibrium equations, integrated over each control volume, can now be written in terms of nodal incremental displacements:

\[
\int_v \Delta \sigma_{ij,j} \, dv = \sum_f [D]([B] \Delta u - \Delta \varepsilon_t - \Delta \varepsilon_{vp}) \cdot nA_f.
\]

Combining the contributions from each control volume will result in a system of equations for all nodal incremental displacements.

4. Solution procedure

The above discretised equations for dependent variables at both cell-centred and vertex-centred control volumes can be assembled to provide systems of equations for the dependent variable \( \phi \) representing velocities, temperature, fluid pressure, and nodal displacements. The form of each system of
equations is
\[
[A] \phi = b,
\] (22)

where $[A]$ is the system matrix and $b$ contains source terms. These integrated systems of equations are solved using the following solution procedure.

1. Solve CFD equations using the SIMPLE procedure [10, 5].
2. Solve energy equation for temperatures and liquid fractions [5].
3. Evaluate other variable quantities, e.g. physical properties, etc.
4. Repeat steps 1–3 until convergence.
5. Solve incremental solid displacements [3, 9, 17, 16].
6. Update total stress variables. $\sigma_{\text{new}} = \sigma_{\text{old}} + \Delta \sigma$.
7. Recalculate geometrical quantities.
8. Repeat steps 1–7 for time-step advancement.

Note, in step 7, the geometrical quantities are recalculated due to the changes in displacement from the computational solid mechanics analysis, and the equations in steps 1 and 2 are rediscrteised over the updated mesh.

5. Boundary conditions for casting simulation

It is important when predicting how a cast shape develops, to be able to model the thermal changes correctly. To do this, account must be taken of the gap formation at the mould/cast interface. For this analysis, coincident nodes have been used to model the interface. This enables the cast component to move freely away from the mould. The boundary condition at this interface is given by

\[
\frac{\partial T}{\partial n} = h_{\text{eff}}(T_c - T_m),
\] (23)

where $T_c$, $T_m$ and $h_{\text{eff}}$ are the cast temperature, mould temperature and effective heat transfer coefficient which is given by

\[
h_{\text{eff}} = \frac{k_{\text{gap}}}{d_{\text{gap}}},
\] (24)

where $k_{\text{gap}}$ and $d_{\text{gap}}$ are thermal conductivity of the gap medium and the gap distance. During this analysis, the mould is assumed to be rigid and hence the mechanical contact analysis, although time consuming, is reasonably straightforward. A node is labelled as being in contact when it penetrates the mould and the displacement calculations are undertaken with this constraint in place. If the stress condition at a contacting node becomes tensile then the constraints at this node are released in further calculations.
6. PHYSICA: a multi-physics environment

A common element in all computational mechanics codes is the mesh representing the domain. Various discretisation methods can be used to discretise the equations governing the physics of interest across this mesh. For finite volume procedures the evaluation of fluxes across cell/element faces, volume sources and coefficients of the linear solvers are generic, being essentially based on mesh geometry and materials properties within a cell. PHYSICA [13] is a software framework which has generic tools that concentrate on objects (i.e. mesh faces, etc.) and their relations, in evaluating all the relevant terms (diffusion, convection, source, etc) at a generic level. Fig. 2 shows the design principles inherent within the PHYSICA framework. It is expected that the majority of modellers would only use the top three levels. So, for example, a new higher-order convection scheme may be inserted by the user at the ‘Algorithm’ level and the modular structure of the code will allow compatibility with all other routines.

Such a computational framework will aid in reducing model development times by many orders of magnitude. At present PHYSICA has the following capabilities:

- Data structures for tetrahedral, wedge, hexahedral and full polyhedral elements.
- SIMPLE based solution for incompressible Navier–Stokes equations.
- Free surface flow algorithm.
- Range of turbulence models.
- Enthalpy based solidification/melting procedures.
- Elastoviscoplastic solid mechanics algorithm.
- Solvers which include PCCG, SOR and Jacobi.
- Hooks to pre and post-processing packages.

PHYSICA is now being used to simulate numerous multi-physics-based phenomena for which the casting process is one.
Table 1
Material properties of the aluminium casting alloy

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T_L$ (Liquidus)</td>
<td>618.8°C</td>
</tr>
<tr>
<td>$T_S$ (Solidus)</td>
<td>566.4°C</td>
</tr>
<tr>
<td>$h$ (Latent)</td>
<td>440 kJ/kg</td>
</tr>
<tr>
<td>$k$ (Thermal)</td>
<td>150 W/(mK)</td>
</tr>
<tr>
<td>$\rho$ (Density)</td>
<td>2710 kg/m³</td>
</tr>
<tr>
<td>$c$ (Specific)</td>
<td>1160 J/(kgK)</td>
</tr>
<tr>
<td>$\alpha$ (Coefficient of thermal expansion)</td>
<td>$5 \times 10^{-5}$/K</td>
</tr>
<tr>
<td>$v$ (Poisson’s ratio)</td>
<td>0.33</td>
</tr>
<tr>
<td>$E$ (Young’s modulus)</td>
<td>60 000 MPa 20°C</td>
</tr>
<tr>
<td></td>
<td>34 000 MPa 450°C</td>
</tr>
<tr>
<td></td>
<td>$1 \times 10^{-2}$ MPa 566.4°C</td>
</tr>
<tr>
<td>$Y$ (Yield stress)</td>
<td>500 MPa 20°C</td>
</tr>
<tr>
<td></td>
<td>$1 \times 10^{-4}$ MPa 566.4°C</td>
</tr>
</tbody>
</table>

7. Results

The following results arise from the use of the multi-physics FV techniques described above, as embedded in the PHYSICA software, for the casting of two metal components. In each experiment data has been gathered and these are compared with the PHYSICA results. For both test cases fluid flow, heat transfer with solidification, and deformation of the solidified material are included in the analysis.

The first test case is based on an experimental analysis of a small aluminium casting which is solidifying inside a rigid steel mould [14]. Fig. 3 shows the experimental set-up detailing the location of thermocouples and a displacement transducer at the cast–mould interface. Table 1 details
the materials data used for this analysis. Initial temperatures of 800°C and 200°C are taken for the casting and mould materials, respectively. Fig. 4 shows the PHYSICA predictions for gap formation over time where solidified parts of the casting cool are pulling away from the mould. It should be noted that as this gap opens, the heat transfer coefficient at this interface decreases and restricts further heat losses. Including this effect in the analysis is detailed in Fig. 5.

Temperature profiles are much closer to the observed values when mechanical predictions are included. Fig. 6 shows the gap formation at cast–mould interface monitored by the displacement transducer. Although the PHYSICA predictions slightly over estimate this gap magnitude the manner in which it forms is clearly predicted.

The second test case details an industrial process where molten lead is poured into an iron mould. Heat transfer into the mould and surrounding atmosphere then cools and solidifies the lead. Fig. 7 shows the computational domain for this analysis. A quarter of the domain is shown with symmetry planes at the centre. Table 2 details the materials data used in this analysis. It is assumed that the
mould is instantly filled and that the cast and mould temperatures start at 730 and 300°C, respectively. This assembly is then air-cooled using a heat transfer coefficient of 15 at the cast–air and mould–air interfaces. As the molten lead cools thermal gradients will evolve resulting in thermal convection as lead flows from hotter to cooler regions. Fig. 8 shows the magnitude of thermal convection after 30 min of cooling. Fig. 9 shows the solidification profiles over time for two mould designs, one of 60 mm wall thickness and the other 100 mm. As expected the lead solidifies inwards from the mould walls towards the centre of the casting where the thicker mould provides a greater rate of solidification. Fig. 10 shows both PHYSICA and thermocouple temperature profiles at the centre of the casting. The effect of latent heat release during the phase change from liquid to solid can clearly be seen. The PHYSICA predictions compare very favourably with the gathered data. As the lead cools it heats the surrounding mould in a non-uniform manner. The thermal gradients occurring in the mould materials result in thermal stresses. Fig. 11 shows the magnitude of stress components throughout the mould after 100 min of cooling. From an industrial point of view these predictions are very useful in detailing the response of the mould to the solidification process. For this analysis it is clear that the thicker mould provides both faster solidification of the ingots as well as smaller stress magnitudes in the mould walls.

Fig. 5. Temperature profiles in casting and mould.
Fig. 6. Gap formation at cast/mould interface.

Table 2
Material properties for the steel mould material

<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k$</td>
<td>Thermal conductivity</td>
</tr>
<tr>
<td>$\rho$</td>
<td>Density</td>
</tr>
<tr>
<td>$c$</td>
<td>Specific heat capacity</td>
</tr>
</tbody>
</table>

8. Conclusions

Increasingly, there is a requirement in engineering processes to model multiphysics phenomena. The rationale behind the PHYSICA framework is to supply modellers with a tool which enables the development of computational mechanics based models where interactions between the participating physics can be accomplished in a robust and efficient manner. The Casting process is a classical example of such multiphysics phenomena which involves coupling between fluid flow, heat transfer and solid mechanics at the macroscopic level. This paper has detailed the discretisation and solution procedures used within the PHYSICA framework. Predictions for two casting simulations and their comparison with experimental observations are encouraging.

Development of PHYSICA is continuing. Future versions will include the physics of magnetic fields, radiation, combustion, etc.
Fig. 7. Computational domain for mould and lead ingot.

Fig. 8. Thermal convection after 30 min of cooling.
Fig. 9. Solidification profiles after 100 min of cooling.

Fig. 10. Temperature values at monitoring point in lead.

Acknowledgements

The financial assistance of the EPSRC, British Aerospace, Normalair-Garrett Ltd. and Britannia Zinc Ltd. is acknowledged in this work.
References

[1] ANSYS, Swanson Analysis Systems Inc, Houston PA, USA.
[13] PHYSICA, University of Greenwich, London, UK, E-mail physica@gre.ac.uk.