UNIVERSIDAD NACIONAL DEL LITORAL

Thermo-Mecano-Metallurgical modelling of welding: Application to welded joints in nuclear power plants

Alejandro Cosimo

FICH
FACULTAD DE INGENIERÍA
Y CIENCIAS HÍDRICAS

INTEC
INSTITUTO DE DESARROLLO TECNOLÓGICO
PARA LA INDUSTRIA QUÍMICA

Tesis de Doctorado **2014**

UNIVERSIDAD NACIONAL DEL LITORAL

Facultad de Ingeniería y Ciencias Hídricas

Instituto de Desarrollo Tecnológico para la Industria Química

THERMO-MECANO-METALLURGICAL MODELLING OF WELDING: APPLICATION TO WELDED JOINTS IN NUCLEAR POWER PLANTS

Alejandro Cosimo

Tesis remitida al Comité Académico del Doctorado

como parte de los requisitos para la obtención

del grado de

DOCTOR EN INGENIERÍA

Mención Mecánica Computacional

de la

UNIVERSIDAD NACIONAL DEL LITORAL

2014

Comisión de Posgrado, Facultad de Ingeniería y Ciencias Hídricas, Ciudad Universitaria, Paraje

"El Pozo", S3000, Santa Fe, Argentina.

UNIVERSIDAD NACIONAL DEL LITORAL
Facultad de Ingeniería y Ciencias Hídricas

Como miembros del Jurado Evaluador de la Tesis de Doctorado titulada

THERMO-MECANO-METALLURGICAL MODELLING OF WELDING: APPLICATION TO WELDED JOINTS IN NUCLEAR POWER PLANTS

preparada por

Alejandro Cosimo

Certificamos que hemos evaluado la Tesis y recomendamos que sea aceptada como parte de los requisitos para la obtención del grado de
DOCTOR EN INGENIERÍA

Dr. Gustavo Sánchez Sarmiento Miembro del Jurado	Dr. Enrique Pardo Miembro del Jurado
Dr. Mario Storti Miembro del Jurado	Dra. Patricia Dardati Miembro del Jurado

Certifico que he leído esta Tesis preparada bajo mi dirección y recomiendo que sea aceptada como parte de los requisitos para la obtención del título de DOCTOR EN INGENIERÍA.

Dr. Alberto Cardona
Director

17 de Noviembre de 2014

A Ceci, el gran amor de mi vida.
A mi familia, por su constante apoyo.

Finis Origen Pendent.

Agradecimientos

En este caso no se trata de agradecer solamente a las personas que estuvieron de alguna manera involucradas en este trabajo sino a todas aquellas personas que influyeron en mi formación tanto académica como humana. Debo agradecer en primer lugar a Dios por rodearme de buenas personas, que sin su paciencia y aliento poco hubiese hecho.

A mi esposa Cecilia y a mi familia les agradezco mucho por apoyarme siempre, incondicionalmente, en las decisiones que tomé y tomo a lo largo de mi vida. Gracias a ellos además porque me alegran la vida día a día. También, mi reconocimiento a mis amigos de la facu que de una manera u otra se hacen presente.

A mi director, Prof. Alberto Cardona, le agradezco por la paciencia que me tuvo y por los valiosos consejos que me dio, consejos que ayudaron en gran medida a la producción de esta tesis. También, debo agradecer al Prof. Sergio Idelsohn por su predisposición a discutir formulaciones de modelos de orden reducido.

Gracias a mis compañeros del CIMEC, especialmente a aquellos que me dieron una mano en charlas técnicas y otras no tan técnicas sino más bien de carácter humano. En este sentido, especialmente gracias a mis compañeros de oficina Fede, Martín y César, y a Alfredo Huespe y Lisandro Dalcin por el tiempo que me dedicaron cuando necesitaba discutir ciertos problemas de esta tesis. A Nestor Calvo y Pablo Novara les agradezco por la ayuda que me dieron con la malla del problema presentado en el Capítulo 7.

Toda mi formación la realicé en instituciones educativas estatales. Es por esto que estoy agradecido al estado argentino por su contribución a mi educación. En esta ocasión agradezco especialmente a la Autoridad Regulatoria Nuclear de la República Argentina por la beca doctoral que me otorgaron y con la cual pude solventar mis estudios doctorales.

Abstract

The Thermo-Mecano-Metallurgical (TMM) modelling of welding is considered in this thesis, where the high non-linearity and the multiphysics character of the problem makes necessary to study different areas of Computational Mechanics. Each of the main problems, specifically the thermal, the mechanical and the metallurgical problems, are separately investigated. In the context of Computational Welding Mechanics (CWM), their coupling is solved by means of a staggered approach making the hypothesis that they are weakly-coupled.

In the case of the thermal problem, the primary complication is stated by the solid/liquid phase change. Classical formulations dealing with the solution of this problem suffer from instabilities associated to the discontinuity of the temperature gradient at the phase change boundary. This issue is studied in this work by considering an enriched finite element formulation with the ability of representing the gradient discontinuity inside finite elements. It is remarked that the proposed method avoids the use of an auxiliary equation to determine the enrichment position, which is common for level set formulations.

The mechanical behaviour of bodies during solidification is revisited and implemented as part of the Finite Element (FE) framework OOFELIE. When possible, microstructure evolution must be considered in order to correctly predict Weld Residual Stresses (WRS). In this context, the implementation of a particular model for predicting microstructure evolution comes in association with the restriction that it can be applied to a reduced number of materials. In order to deal with this issue, the conception of a computational tool flexible enough to describe a wide range of materials is undertaken. Additionally, a model describing the Titanium alloy Ti6Al4V is particularly considered.

The high computational cost of welding problems is addressed by means of the formulation of Hyper-Reduced Order Models (HROMs), and the parallelisation of the FE framework OOFELIE. It is shown that the speed-ups obtained with the proposed HROMs are good enough, even when the speed-up is computed using the time taken by the full model run in parallel.

The developed computational framework is applied to the solution of a real life application problem, more specifically, to the simulation of a large welded structure of a nuclear power plant for the prediction of WRS.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation and background

The use of welding as a technology for joining components of large dimensions in nuclear power plant facilities, is a standard procedure. For example, this type of joint is found in the reactor pressure vessel, in the cold and hot legs, and in secondary pipe systems. All these structures are affected by welding processes, because of the characteristic high temperatures and gradients that, along with the high thermal deformations, induce solid/liquid and solid/solid phase transformations. As a result, the welded parts are subjected to complex patterns of Weld Residual Stresses (WRS) that highly influence the development of Stress Corrosion Cracking (SCC), a mechanism of failure characterised by the growth of long cracks. Therefore there is a clear need of developing techniques aimed at predicting WRS by means of the simulation of the welding process.

The study of WRS requires the coupled modelling of the Thermo-Mecano-Metallurgical (TMM) response of the welded parts, issue that is particularly addressed in this thesis. The couplings between the different involved physical problems can be observed in Figure 1.1. Ronda and Oliver [1] proposed to model these couplings by making use of a monolithic approach. Even though the methodology presented in that work is interesting, the computational cost and the numerical instabilities of monolithic schemes make necessary to study other alternatives. Specifically, a weakly coupled scheme is considered in this work by adopting an staggered approach for the description of the TMM problem [2].

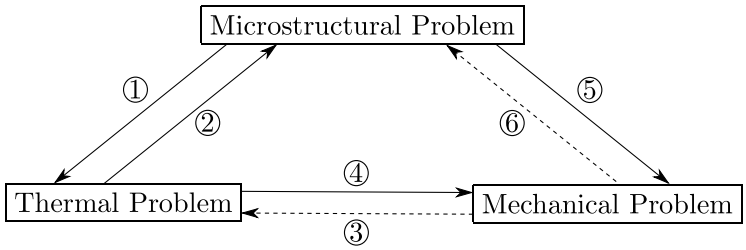


Figure 1.1: Thermo-Mecano-Metallurgical (TMM) coupling. Dashed lines couplings are neglected.

The heat generated by mechanical deformations (coupling (3)) is negligible in welding problems when compared to the heat input to the medium given by the welding torch (excepting the case of explosion welding [3]). Furthermore, metallurgical transformations induced by mechanical deformations are generally neglected. For example, in order

to model the formation of martensite, the Koistinen-Marburger model [4] is adopted by making the assumption that mechanical deformations do not influence the formation of martensite. Based on these considerations, couplings (3) and (6) are neglected.

It is essential to describe the variation of material's properties in terms of temperature, because thermal expansion is the driving force characterising the problem. Another factor that must be considered, when the material data are available, is the description of material properties variation in terms of microstructure evolution, or, equivalently, in terms of the thermal history. Oddy *et al.* [5] highlight the importance of considering microstructure evolution in order to correctly predict WRS.

Microstructure evolution can be modelled at different length scales [6], namely at the macro-scale and at the meso-scale. In this work, macro-scale based approaches are chosen for modelling microstructure evolution. This is justified in the low computational complexity of this kind of model, making tractable large scale welding problems [3]. Kirkaldy's model [7] is one of the most widespread models in the Computational Welding Mechanics (CWM) community [3, 8, 9] to describe microstructure evolution of low alloy steels. According to Watt *et al.* [10], this model can describe the microstructural behaviour of any low allow steel just by providing its chemical composition. However, it is restricted to low alloy steels only, and high alloy steels are typically used in nuclear power plants.

Microstructure evolution is mainly characterised by two types of transformations, known as diffusion controlled and diffusionless transformations. In this thesis, diffusion controlled transformations are described by means of the Johnson-Mehl-Avrami-Kolmogorov (JMAK) model [11, 12, 13], whereas diffusionless transformations, such as the formation of martensite, are modelled by the Koistinen-Marburger equation [4]. Despite the simplicity of these formulations, every time that a different material needs to be described, standard software requires the implementation of a new model. An alternative granting code reuse and more material independence of the implemented microstructural framework, is particularly studied.

The variation of material properties in terms of microstructure is modelled by making use of linear mixing rules. That is, for each property to be modelled its macroscopic value is computed as the linear combination of the values of the properties at a given temperature for each phase, with the coefficients of the linear combination given by the predicted phase fractions. It must be mentioned that for some material properties, such as hardening, linear mixing rules could be inappropriate making necessary to use a different mixing rule [14, 15].

Coupling (1) in Figure 1.1 reduces to the variation of thermo-physical properties in terms of microstructure. In the case of the mecano-microstructural coupling (5) the situation is more interesting, because it not only involves the mechanical properties variation in terms of microstructure, but also the deformations due to the microstructural volume change and the Transformation Induced Plasticity (TRIP) phenomenon. Oddy et al. [16] and some of the works mentioned there, showed that ignoring TRIP can significantly modify the computed WRS. Nevertheless, volume changes associated to microstructural transformations and TRIP are not considered because more experimental research is needed in this direction, subject which is out of the scope of this study.

It is frequently found in the CWM literature [17] that the mechanical behaviour of the material is modelled by different constitutive relations in each region of the thermal cycle. Isotropic elasto-plastic hardening models are adopted in the temperature range $T < 0.5T_m$, where T is the temperature field and T_m the melting temperature, whereas in the temperature range $0.5T_m < T < 0.8T_m$ viscous-plastic models are used, and in

the interval $0.8T_m < T < 1.0T_m$ linear viscous models are considered. In this study, as in the works of Anca *et al.* [18, 19], a different line is followed, the material response is modelled by an isotropic elasto-plastic hardening model in the whole temperature range. Additionally, the mechanical problem is delimited by the Zero Strength Temperature T_{zst} above which it is considered that the material has no strength.

When solving the thermal problem, the latent heat associated to solid/solid and solid-/liquid phase changes must be considered, although the assumption that the contribution of solid/solid latent heat is negligible is usually taken [3]. From a general point of view, two specialisations of the Finite Element Method (FEM) are aimed at solving phase change problems: moving mesh or front tracking methods and fixed mesh methods [20]. Moving mesh methods track the position of the phase change boundary; therefore, through remeshing, the mesh conforms to the position of the interface. In this way, the FEM formulation addresses the weak discontinuity present in the temperature field at the solidification front in a standard way, which is presented in several studies [21, 22, 23]. The problem with this type of methods is that in multidimensional problems, the mesh can become too distorted as the interface evolves. This does not occur with fixed mesh methods. Different types of fixed mesh methods were proposed, e.g., enthalpy methods [24, 25], capacitance methods [26] and temperature based methods [27, 28, 29, 30]. Despite the fact that none of these methods can exactly capture the discontinuity of the temperature gradient at the phase change boundary, in Fachinotti's thesis [31] is shown that when adopting the fixed mesh variant presented in [28] good results are obtained in comparison to other more standard methods. Nonetheless, it must be mentioned that, in certain situations, this kind of fixed mesh finite element formulation suffers from instabilities that lead to obtain spurious oscillations in the numerical solutions. One of the main reasons for these oscillations, is the incapacity of the method to represent the discontinuity of the temperature gradient at the position of the interface within the finite elements. This problem is particularly studied from the point of view of enrichment techniques.

The high complexity of CWM problems makes necessary to explore computationally efficient formulations, and to consider the application of High Performance Computing (HPC) techniques to already existing formulations. In this context, the use of Reduced Order Models (ROMs) appears as an elegant and promising solution. The manner in which the reduced order model is built is mainly given by two approaches [32]. In one approach, *a posteriori* model reduction techniques require the knowledge of the solution to a training problem. The most prominent *a posteriori* technique is based on the Proper Orthogonal Decomposition (POD) method [33, 34]. For the second approach, the *a priori* model reduction techniques require no previous knowledge of the solution. In this context, the leading technique is the Proper Generalised Decomposition (PGD), which has its roots in the works of Ladevèze [35, 36]. In this thesis, the formulation of reduced order models for the non-linear phase change problem with moving heat sources is considered in the context of *a posteriori* ROMs techniques. In the case of the POD method, in order to efficiently deal with the non-linearity of the problems, a second reduction is applied leading to obtain a Hyper-Reduced Order Model (HROM). It is worthwhile to mention that the development of HROMs specifically suited for phase change and welding problems has not yet been addressed by other authors.

Currently, the use of distributed shared memory architectures, also referred to as NUMA (Non-Uniform Memory Access) systems [37], represents an interesting option to deal with time consuming tasks. In order to exploit the computational resources offered by this architecture, the techniques developed in this thesis are parallelised. Part of the

work performed on this topic is based on the study of Farhat and Crivelli [38].

The work to be done in this thesis is summarised in the following paragraph. In the case of the thermal problem, the main complication is stated by the solid/liquid phase change. Classical formulations dealing with the solution of this problem suffer from instabilities associated to the discontinuity of the temperature gradient at the phase change boundary. This matter is studied in detail by considering an enriched finite element formulation with the ability of representing the gradient discontinuity inside finite elements. Then, the mechanical behaviour of bodies during solidification is revisited and implemented as part of the finite element framework OOFELIE [39]. When possible, microstructure evolution must be considered in order to correctly predict Weld Residual Stresses (WRS). In this context, the implementation of a particular model for predicting microstructure evolution comes in association with the restriction that it can be applied to a reduced number of materials. In order to deal with this problem, the development and design of a computational tool possessing enough flexibility to describe a wide range of materials is undertaken. Additionally, the microstructural behaviour under arbitrary thermal histories of the Titanium alloy Ti6Al4V is particularly analysed. This multiphysics description of welding is characterised by highly intensive computations. In order to address this problem, the formulation of Hyper-Reduced Order Models and the parallelisation of the Finite Element code OOFELIE, is performed. Then, the developed computational framework is applied to the solution of a real life application problem, more specifically, to the simulation of a large welded structure of a nuclear power plant for the prediction of WRS.

1.2 Objectives

The **general objective** of this thesis is to build robust and stable algorithms aimed at solving CWM problems, with special emphasis on computational efficiency ensured by the application of techniques of the HPC community. Specifically, the Thermo-Mecano-Metallurgical multiphysics modelling of welding is considered. In order to reduce the time required by the simulations, the formulation of hyper-reduced order models in the context of *a posteriori* ROM techniques is considered.

As a **particular objective** the developed tools are used for the prediction of Weld Residual Stresses of a nuclear power plant welded joint.

1.3 Overall structure of this thesis

As it was previously introduced, the TMM description of welding is numerically solved by using a weakly coupled scheme. Specifically, at each time step the solving sequence starts by the thermal problem, where the temperature field is computed by making use of the phase fractions characterising the microstructure of the material at the previous time step. Then, the mechanical problem is solved using as state the temperature field at the current time step and the phase fractions at the previous time step. Finally, by making use of the temperature field at the current time step, the microstructural problem is solved.

In this context, the thermal, the mechanical, and the microstructural problems are analysed in a separate manner, because of the weakly coupled scheme chosen for making computations. Therefore, this thesis is structured in chapters, each of which studies a specific problem. At the end of every chapter, the conclusions and the future work arising from the analysed topic are presented.

A basic overview of each chapter is next given. In Chapter 2, the thermal problem is introduced and the numerical issue stated by the discontinuity of the temperature gradient in solid/liquid phase change problems is studied in detail. Microstructure evolution of processes characterised by arbitrary thermal histories, and the general formulation and implementation of microstructural models are topics analysed in Chapter 3. In Chapter 4, the mechanical behaviour of bodies under solidification is revisited, and the material models to be used are introduced. The parallel implementation of the Finite Element framework, in the context of distributed shared memory architectures, is considered in Chapter 5. The cost of solving non-linear phase change problems with moving sources, is addressed in Chapter 6 by means of the formulation of Hyper Reduced Order Models. In Chapter 7, a nuclear power plant welded joint is studied in order to compute weld residual stresses. Finally, Chapter 8 presents the main conclusions of this thesis.

Chapter 2

Thermal Problem

The correct description of the thermal problem is essential in the simulation of welding processes, due to the fact that the thermal expansion is the driving force behind these kind of problems [40]. The dependence of thermophysical properties in terms of temperature must be considered, leading to a highly non-linear problem. Another phenomenon that must be taken into account is the latent heat contribution coming from the solid/liquid phase change, situation which highly complicates the numerical solution of the problem. The main difficulty is stated by the temperature gradient discontinuity at the solidification front. The numerical scheme presented by Fachinotti *et al.* [28] works well for a wide range of problems (in fact, it is used in this thesis to solve some of the studied problems obtaining good results). However, this method does not exactly represent the temperature gradient discontinuity leading to spurious oscillations in some situations.

In this chapter, the mathematical setting of the thermal problem is stated and a new enriched finite element formulation for solving isothermal phase change problems is presented. We propose a fixed mesh method, where the discontinuity in the temperature gradient is represented by enriching the finite element space through a function whose definition includes a gradient discontinuity. Generally, in these types of formulations, the enrichment location (the location of the solidification front) is determined through a level set auxiliary scheme. In this work, this position is computed implicitly by constraining the temperature at the phase change boundary to be equal to the melting temperature. Several numerical examples are presented to show the application of the method.

Remark: in what follows only the latent heat contribution coming from solid/liquid phase changes is considered. The contribution of solid/solid latent heats is studied in Chapter 3.

2.1 Introduction

Solidification processes are of interest in many areas of engineering, such as welding mechanics, nuclear engineering, metallurgy and metal casting processes [41, 42, 43, 40, 44]. When considering solidification of a pure substance, problems are described as isothermal phase change problems that are characterised mainly by two parameters: the material melting temperature and its latent heat. An inherent difficulty with these problems is the discontinuity in the temperature gradient at the solidification front.

From a general point of view, two specialisations of the Finite Element Method (FEM) are aimed at solving phase change problems: moving mesh or front tracking methods

and fixed mesh methods [20]. Moving mesh methods track the position of the phase change boundary; therefore, through remeshing, the mesh conforms to the position of the interface. In this way, the FEM formulation addresses the weak discontinuity present in the temperature field at the solidification front in a standard way [21, 22, 23]. The problem with this type of methods is that in multidimensional problems, the mesh can become too distorted as the interface evolves. This does not occur with fixed mesh methods. Different types of fixed mesh methods were proposed, e.g., enthalpy methods [24, 25], capacitance methods [26] and temperature based methods [27, 28, 29, 30]. However, none of these methods is able to represent the gradient discontinuity at the phase change boundary.

Because of the mesh distortion problem and the need for remeshing of the front tracking methods, we propose a fixed mesh scheme, which represents the aforementioned discontinuity. This method has the advantages of both groups of methods: the ability to represent the weak discontinuity without mesh distortions.

In classical fixed mesh methods, loss of stability and/or robustness, which are associated with the discontinuity at the interface, appears with low sensible to latent heat ratios (Stefan number) or when the initial temperature is close to the melting temperature [45]. These situations cannot be handled efficiently and require a large number of elements for accurate solutions. In several scenarios, the standard Newton-Raphson solver does not converge and a line search must be used [46].

Fixed mesh finite element formulations that perform discontinuous spatial integration, such as those proposed by Crivelli and Idelsohn [29] and subsequently refined by Storti *et al.* [30] and Fachinotti *et al.* [28], provide spurious oscillations in the computed solution for certain situations. This phenomenon appears for high latent heat values because the discontinuity in the temperature gradient at the interface is pronounced and cannot be represented when the position of the solidification front does not coincide with an element boundary due to the continuous interpolated gradient inside the element. This situation is worse with linear shape functions, which are used in most cases with discontinuous integration, resulting in a space of trial solutions that can reproduce up to piecewise constant gradients. One approach to avoid this situation is to introduce the representation of the gradient discontinuity by enriching the element. Another approach, proposed recently by Davey and Mondragon [47], involves removing the discontinuity through the introduction of a non-physical enthalpy.

Currently, enriching techniques are widely used in fluid and fracture mechanics [48, 49, 50]. These techniques are designed to represent the existing discontinuities and singularities in the fields of interest. If we apply the usual techniques in these situations (e.g., standard finite element formulations), the overall convergence rate is not optimal in the sense described by Fries [51].

Although new enriched formulations are constantly being proposed, it should be noted that this is not the case for solidification problems. There are few enriched formulations in the literature for solving the previously mentioned types of phase change problems. Chessa *et al.* [52] and Bernauer and Herzog [53] determined the position of the enrichment through a level set function that is evolved with the interface Rankine-Hugoniot condition (specialised to the Stefan problem) and the associated advection equation. Several steps are performed to compute the solution of the level set equation [54]. First, the velocity of the solidification front is computed from the Stefan condition, and a velocity field is built by extending the velocity of the solidification front to the whole domain. Next, the level set equation is solved to move the interface. Finally, the standard heat equation is solved for each domain (solid and liquid) separately by imposing the interface constraint

that dictates that the temperature at the phase change boundary must be equal to the melting temperature. The imposition of the mentioned constraint is enforced with a penalty formulation or a Lagrange multipliers formulation. Ji *et al.* [55] present a similar approach, differing from the previous one in the level set update and the energetically consistent way that they use to determine the jump in the heat flux at the interface. In contrast, Merle and Dolbow [56] proposed an enriched formulation in which they use an equation that is similar to a level set to track the interface position and the LATIN method [57] as iterative procedure to satisfy the local interface constraints stated by the problem.

In this work, the Stefan condition is satisfied with the weak formulation of the problem. This is accomplished with a weak formulation of the problem in each subdomain (liquid and solid). The Reynold's theorem is applied to extend the formulation to the whole domain and to satisfy the Stefan condition as an internal natural phase change boundary condition. Next, the interface position is determined implicitly during the simulation with the aforementioned constraint in the temperature at the solidification front. The formulation is performed while considering the enriched space. In this way, the overhead introduced by the classical level set methods is avoided and no auxiliary equation (e.g., the *level set* equation) is required.

The discrete formulation is completely stated for the one dimensional case and a discussion for the extension to two dimensions is given. The one dimensional implementation is tested for extreme values of temperature gradient discontinuity and for initial temperatures close to the melting temperature. The results obtained for one of the examples are contrasted against those obtained by other enriched formulations [56]. Also, comparisons with results obtained using a non enriched fixed mesh numerical scheme [28], where the temperature gradient discontinuity is not considered, are presented. Finally, conclusions are presented.

2.2 Mathematical setting

Isothermal phase change problems are governed by the first principle of thermodynamics. Assuming the contribution of the mechanical energy to the total energy negligible and considering the specific enthalpy $\mathcal{H}$ as a thermodynamic potential, the temperature field T is computed by solving the heat balance equation

$$\rho\dot{\mathcal{H}} = Q + \nabla\cdot(k\nabla T) \qquad \forall\ (\boldsymbol{x},t)\in\Omega_i\times(t_0,\infty) \tag{2.1}$$

where ρ is the density, k is the thermal conductivity, T is the temperature, Q is the external heat source per unit volume, and Ω_i for $i\in[s,l]$ are the solid and liquid sub-domains with $\Omega_s\cap\Omega_l=\{\emptyset\}$ and $\Omega=\Omega_s\cup\Omega_l$. The temperature field should verify the initial conditions

$$T = T_0 \qquad \forall\ \boldsymbol{x}\in\Omega,\ t=t_0 \tag{2.2}$$

where $T_0(\boldsymbol{x})$ is the given initial temperature field. Additionally, the following set of conditions must be verified at the disjoint portions $\Gamma_d,\Gamma_q,\Gamma_c$ of the external boundary:

$$T = T_d \qquad \forall\ (\boldsymbol{x},t)\in\Gamma_d\times(t_0,\infty) \tag{2.3}$$

$$-(k\nabla T)\cdot\boldsymbol{n} = q_w \qquad \forall\ (\boldsymbol{x},t)\in\Gamma_q\times(t_0,\infty) \tag{2.4}$$

$$-(k\nabla T)\cdot\boldsymbol{n} = h_f(T-T_f) \qquad \forall(\boldsymbol{x},t)\in\Gamma_c\times(t_0,\infty) \tag{2.5}$$

where $\Gamma_d\cup\Gamma_q\cup\Gamma_c=\partial\Omega$, and where T_d is the imposed temperature at the boundary Γ_d, q_w is the external heat flow at the boundary Γ_q, h_f is the heat convection coefficient, T_f is the

external fluid temperature at the portion the boundary Γ_c and $\boldsymbol{n}$ is the outward normal to the boundary under consideration. Finally, at the interface Γ between Ω_s and Ω_l (the phase change boundary), the constraint on the temperature and the Stefan condition hold

$$T = T_m \qquad \forall\ (\boldsymbol{x}, t) \in \Gamma \times (t_0, \infty) \tag{2.6}$$

$$[-(k\nabla T) \cdot \boldsymbol{n}_\Gamma]_\Gamma = \rho \mathcal{L} u_\Gamma \qquad \forall\ (\boldsymbol{x}, t) \in \Gamma \times (t_0, \infty). \tag{2.7}$$

Here, $\mathcal{L}$ is the latent heat, T_m is the melting temperature, $\boldsymbol{n}_\Gamma$ is the outward normal to the solidification front from the solid domain, $u_\Gamma = \boldsymbol{u}_\Gamma \cdot \boldsymbol{n}_\Gamma$ is the velocity of the interface in the direction of the normal $\boldsymbol{n}_\Gamma$ and the operator $[\centerdot]_\Gamma$ measures the jump of the quantity $\centerdot$ at the solidification front. Equation (2.6) is the constraint that imposes that the temperature at the phase change boundary must be equal to the melting temperature and equation (2.7) is the interface condition (the Stefan condition).

2.2.1 Variational temperature based formulation

Let $\mathcal{S} = \{T/T \in \mathcal{H}^1(\Omega),\ T|_{\Gamma_d} = T_d\}$ be the space of trial solutions and $\mathcal{V} = \{v/v \in \mathcal{H}^1(\Omega),\ v|_{\Gamma_d} = 0\}$ be the space of weighting or test functions, where $\mathcal{H}^1$ is the first order Sobolev space. By integrating equation (2.1) separately in each subdomain (solid and liquid), the following weak form is obtained

$$\sum_{i \in [s,l]} \int_{\Omega_i} w \left[\rho \dot{\mathcal{H}} - \nabla \cdot (k \nabla T) - Q\right]\ d\Omega = 0 \tag{2.8}$$

with $w \in \mathcal{V}$ and $T \in \mathcal{S}$. After applying the divergence theorem and replacing the boundary conditions from equations (2.3 - 2.5), we obtain

$$\sum_i \int_{\Omega_i} w \left[\rho \dot{\mathcal{H}} - Q\right]\ d\Omega + \sum_i \int_{\Omega_i} \nabla w \cdot (k \nabla T)\ d\Omega\ - \sum_i \int_{\Gamma_i} w k \nabla T \cdot \boldsymbol{n}_i\ d\Gamma + \int_{\Gamma_c} w h_f (T - T_f)\ d\Gamma\ + \int_{\Gamma_q} w q_w\ d\Gamma = 0. \tag{2.9}$$

Functions $w \in \mathcal{V}$ depend on time because the enrichment continuously adapts to the position of the interface. By applying the Reynold's theorem to the first term of equation (2.9), we obtain

$$\sum_i \int_{\Omega_i} w \rho \dot{\mathcal{H}}\ d\Omega = \sum_i \left[\frac{\partial}{\partial t} \int_{\Omega_i} w \rho \mathcal{H}\ d\Omega - \int_{\Omega_i} \rho \mathcal{H} \dot{w}\ d\Omega\ - \int_{\Gamma_i} w \rho \mathcal{H} \boldsymbol{u}_i \cdot \boldsymbol{n}_i\ d\Gamma\right] \tag{2.10}$$

where $\boldsymbol{u}_i$ is the velocity of the boundary Γ_i and $\boldsymbol{n}_i$ is the external unit normal vector of that boundary. Taking into account that the latent heat represents the jump of the enthalpy at the phase change boundary Γ, we have $\mathcal{L} = [\mathcal{H}]_\Gamma$. Then, the last term of the last equation becomes

$$\sum_i \int_{\Gamma_i} w \rho \mathcal{H} \boldsymbol{u}_i \cdot \boldsymbol{n}_i\ d\Gamma = \int_\Gamma w \rho \mathcal{L} u_\Gamma\ d\Gamma = -\sum_i \int_{\Gamma_i} w k \nabla T \cdot \boldsymbol{n}_i\ d\Gamma. \tag{2.11}$$

Replacing equation (2.11) into (2.10) and using this result in equation (2.9), we obtain

$$\sum_i \left[\frac{\partial}{\partial t} \int_{\Omega_i} w \rho \mathcal{H}\ d\Omega - \int_{\Omega_i} \rho \mathcal{H} \dot{w}\ d\Omega\ - \int_{\Omega_i} w Q\ d\Omega + \int_{\Omega_i} \nabla w \cdot (k \nabla T)\ d\Omega\right] + \int_{\Gamma_c} w h_f (T - T_f)\ d\Gamma + \int_{\Gamma_q} w q_w\ d\Gamma = 0. \tag{2.12}$$

The Stefan condition is therefore satisfied in weak form. Then, the addition of the first two terms of the last equation gives

$$\sum_i \left[\frac{\partial}{\partial t} \int_{\Omega_i} w\rho\mathcal{H}\, d\Omega - \int_{\Omega_i} \rho\mathcal{H}\dot{w}\, d\Omega \right] = \frac{\partial}{\partial t} \int_{\Omega} w\rho\mathcal{H}\, d\Omega - \int_{\Omega} \rho\mathcal{H}\dot{w}\, d\Omega = \int_{\Omega} w\rho\dot{\mathcal{H}}\, d\Omega. \quad (2.13)$$

Using this result into equation (2.12), we obtain

$$\int_{\Omega} w \left[\rho\dot{\mathcal{H}} - Q\right]\, d\Omega + \int_{\Omega} \nabla w \cdot (k\nabla T)\, d\Omega + \int_{\Gamma_c} w h_f (T - T_f)\, d\Gamma + \int_{\Gamma_q} w q_w\, d\Gamma = 0. \quad (2.14)$$

The specific enthalpy $\mathcal{H}$ can be expressed in terms of the temperature T as

$$\mathcal{H}(T) = \int_{T_{\text{ref}}}^{T} c(\tau) d\tau + \mathcal{L} f_l(T) \quad (2.15)$$

where T_{ref} is a reference temperature, $c(\tau) \equiv c$ is the heat capacity and $f_l(T)$ is the liquid fraction. For isothermal phase change, the liquid fraction is expressed as a Heaviside step, i.e. $f_l = H_{eav}(T - T_m)$.

Finally, by replacing equation (2.15) into (2.14), the following temperature based variational formulation is obtained

Find $T \in \mathcal{S}$ *such that* $\forall w \in \mathcal{V}$

$$\int_{\Omega} w \left[\rho c \frac{\partial T}{\partial t} + \rho\mathcal{L}\frac{\partial f_l}{\partial t} - Q\right]\, d\Omega + \int_{\Omega} \nabla w \cdot (k\nabla T)\, d\Omega + \int_{\Gamma_c} w h_f (T - T_f)\, d\Gamma + \int_{\Gamma_q} w q_w\, d\Gamma = 0\,, \qquad \text{for } t > 0;$$

$$\int_{\Omega} wT\, d\Omega = \int_{\Omega} wT_0\, d\Omega\,, \qquad \text{for } t = 0. \quad (2.16)$$

Remark: in the case of isothermal phase change, the term involving the time derivative of the liquid fraction $\frac{\partial f_l}{\partial t}$ should be interpreted in a distributional sense. In order to show that $\int_\Omega w\rho\mathcal{L}\frac{\partial f_l}{\partial t}\, d\Omega$ is well-defined, we first analyse the derivative $\frac{\partial f_l}{\partial t}$. For simplicity the following analysis is restricted to a solidification process. By expressing f_l in terms of time as $f_l = H_{eav}(t - t^*)$, where t^* is the time at which T_m is reached, and by letting ϕ be a test function in the time domain, the weak time derivative of f_l is given by

$$\int_{-\infty}^{\infty} \phi \dot{f}_l\, dt = -\int_{-\infty}^{\infty} \dot{\phi} f_l\, dt = -\int_{-\infty}^{\infty} \dot{\phi} H_{eav}(t - t^*)\, dt = -\int_{t^*}^{\infty} \dot{\phi}\, dt = \phi(t^*) = \int_{-\infty}^{\infty} \delta(t - t^*)\phi\, dt = \int_{-\infty}^{\infty} \delta(T - T_m)\phi\, dt. \quad (2.17)$$

From which the weak time derivative of f_l is

$$\frac{\partial f_l}{\partial t} = \delta(T - T_m). \quad (2.18)$$

Then, it can be concluded that the integral involving this derivative is well-defined by observing that

$$\int_\Omega w\rho\mathcal{L}\frac{\partial f_l}{\partial t}\,d\Omega = \int_\Omega w\rho\mathcal{L}\delta(T-T_m)\,d\Omega = \int_\Gamma w\rho\mathcal{L}\,d\Gamma, \tag{2.19}$$

and by noting that the last result is well-defined because $w \in \mathcal{H}^1$ has trace in Γ.

2.3 Enriched finite element formulation

For solidification problems, there is a weak discontinuity, i.e. only the gradient of the temperature field is discontinuous at the solidification front. The main features of this discontinuity are its weakness and local behaviour. For the local behaviour, we only need to enrich those elements that are crossed by the phase change boundary. For the weak discontinuity, the enrichment function only needs to have a discontinuity in its gradient.

A Galerkin finite element formulation is adopted for the discretisation of the continuous variational formulation. The enrichment functions are time dependent because of the change in position of the interface. Therefore, the space of weighting functions $\mathcal{V}$ depends on time, and the spatial and time discretisations need to be studied carefully. Following Fries and Zillian [58], the discretisation in time is first performed, and then the space discretisation is performed.

The functional space of the element intersected by the interface (or solidification front) is enriched with a weak discontinuous function denoted by E. From the previous comments, two features need to be considered to build this function: E should have a local character and should vanish at the element nodes, and ∇E must be discontinuous at the phase change boundary. An enrichment function with these features was proposed by Coppola-Owen and Codina [59].

2.3.1 Time discretisation

An unconditionally stable backward Euler scheme is used to accomplish the temporal discretisation, obtaining the following result

$$\int_\Omega w_n\left[\rho c_n\frac{T_n-T_{n-1}}{\Delta t}+\rho\mathcal{L}\frac{f_{l(n)}-f_{l(n-1)}}{\Delta t}\right]d\Omega\ -\int_\Omega\left[\nabla w_n\cdot(k_n\nabla T_n)+wQ_n\right]d\Omega+$$
$$\int_{\Gamma_q} w_n q_{w_n}\,d\Gamma\ +\int_{\Gamma_c} w_n h_{f_n}(T_n-T_{f_n})\,d\Gamma=0 \tag{2.20}$$

where $c_n \equiv c(T_n)$, $k_n \equiv k(T_n)$, $h_{f_n} \equiv h_f(T_n)$ and $q_{w_n} \equiv q_w(T_n)$.

The time level for the evaluation of the test function w was specified in a consistent way at time t_n, i.e. $w(\boldsymbol{x}, t_n) \equiv w_n$. We remark that if w was evaluated at the time level $n-1$, the regularity of the system matrix could not be guaranteed [58].

To elucidate this issue, consider the case where at time step $n-1$ the phase change boundary is within element e, and at time step n the phase change boundary is within the neighbouring element $e+1$. At time t_{n-1}, element e is enriched whereas element $e+1$ is not. When the interface tries to evolve to element $e+1$, the weight function w_{n-1} is zero at that element, providing a null equation for the enrichment. Therefore, the system matrix would be singular.

2.3.2 Spatial discretisation

Let $\mathcal{S}^h \subset \mathcal{S}$ and $\mathcal{V}^h \subset \mathcal{V}$ be N-dimensional subspaces of the trial and test functional spaces, formed by the usual finite element space *and* the enrichment functions. The discrete variational formulation is given by

Given T^h_{n-1}, *find* $T^h_n = v^h + T^h_d$, *where* $v^h \in \mathcal{V}^h$ *and* $T^h_n|_{\Gamma_d} = T^h_d$, *such that* $\forall w^h_n \in \mathcal{V}^h$

$$\begin{aligned}
&\int_\Omega w^h_n \left[\rho c_n \frac{T^h_n - T^h_{n-1}}{\Delta t} + \rho\mathcal{L}\frac{f_{l(n)} - f_{l(n-1)}}{\Delta t}\right] d\Omega \; - \int_\Omega \left[\nabla w^h_n \cdot (k_n \nabla T^h_n) + w^h_n Q_n\right] d\Omega + \\
&\qquad \int_{\Gamma_q} w^h_n q_{w_n}\, d\Gamma + \int_{\Gamma_c} w^h_n h_{f_n}(T^h_n - T_{f_n})\, d\Gamma = 0\,, \qquad \text{for } \; n = 1, 2, 3, \ldots \\
&\int_\Omega w^h_0 T^h_0 \, d\Omega = \int_\Omega w^h_0 T_0 \, d\Omega \qquad\qquad \text{for } \; n = 0. \qquad (2.21)
\end{aligned}$$

For conciseness, it is assumed without loss of generality that $T^h_d = 0$, such that $T^h_n = v^h$. Then equation (2.21) can be written as

$$\begin{aligned}
&\frac{1}{\Delta t}\int_\Omega \rho c_n w^h_n T^h_n \, d\Omega - \frac{1}{\Delta t}\int_\Omega \rho c_n w^h_n T^h_{n-1} \, d\Omega \; + \frac{1}{\Delta t}\int_\Omega \rho\mathcal{L} w^h_n f^h_{l(n)} \, d\Omega - \\
&\qquad \frac{1}{\Delta t}\int_\Omega \rho\mathcal{L} w^h_n f^h_{l(n-1)} \, d\Omega \; + \int_\Omega k_n \nabla T^h_n \cdot \nabla w^h_n \, d\Omega - \int_\Omega Q^h_n w^h_n \, d\Omega + \\
&\qquad\qquad \int_{\Gamma_q} w^h_n q_w \, d\Gamma \; + \int_{\Gamma_c} w^h_n h_{f_n} T^h_n \, d\Gamma + \int_{\Gamma_c} w^h_n h_{f_n} T_{f(n)} \, d\Gamma = 0. \qquad (2.22)
\end{aligned}$$

The discrete test and trial functions $v^h \in \mathcal{V}^h$ are the set of usual linear finite element functions covering the whole domain, plus the enrichment functions at the elements that are crossed by the interface. Thus, a typical enriched finite element in the one dimensional case has a total of three shape functions (including the enrichment one). In matrix notation, $T^h \in \mathcal{V}^h$ inside an enriched element is given by

$$T^h = \boldsymbol{N}^T \boldsymbol{T}, \qquad (2.23)$$

where $\boldsymbol{N}$ denotes the shape functions and $\boldsymbol{T}$ the degrees of freedom amplitudes.

The contribution of an enriched element to the residual at time t_n, is obtained next from equation (2.22):

$$\boldsymbol{\Pi} = \frac{\boldsymbol{C}\boldsymbol{T}_n}{\Delta t} - \frac{\boldsymbol{C}^*\boldsymbol{T}_{n-1}}{\Delta t} + \frac{\boldsymbol{L}_n - \boldsymbol{L}_{n-1}}{\Delta t} \; + \boldsymbol{K}\boldsymbol{T}_n + \boldsymbol{F} - \boldsymbol{Q} \qquad (2.24)$$

where

$$\boldsymbol{C} = \int_{\Omega} \rho c_n \boldsymbol{N}_n \boldsymbol{N}_n^T \, d\Omega \tag{2.25}$$

$$\boldsymbol{C}^* = \int_{\Omega} \rho c_n \boldsymbol{N}_n \boldsymbol{N}_{n-1}^T \, d\Omega \tag{2.26}$$

$$\boldsymbol{K} = \int_{\Omega} \nabla \boldsymbol{N}_n k_n \nabla \boldsymbol{N}_n^T \Omega + \int_{\Gamma_c} h_{f_n} \boldsymbol{N}_n \boldsymbol{N}_n^T \, d\Gamma \tag{2.27}$$

$$\boldsymbol{L}_n = \int_{\Omega} \rho \mathcal{L} \boldsymbol{N}_n f_{l(n)} \, d\Omega \tag{2.28}$$

$$\boldsymbol{L}_{n-1} = \int_{\Omega} \rho \mathcal{L} \boldsymbol{N}_n f_{l(n-1)} \, d\Omega \tag{2.29}$$

$$\boldsymbol{F} = \int_{\Gamma_q} \boldsymbol{N}_n q_{w_n} \, d\Gamma - \int_{\Gamma_c} h_{f_n} \boldsymbol{N}_n T_{f_n} \, d\Gamma \tag{2.30}$$

$$\boldsymbol{Q} = \int_{\Omega} \boldsymbol{N}_n Q_n \, d\Omega. \tag{2.31}$$

It is worthwhile to mention that function $\boldsymbol{N}_n$ depends on the interface position, which is an unknown of the problem, incrementing the degree of non linearity of the equations. The nonlinear problem (2.24) is solved using a Newton-Raphson scheme

$$\boldsymbol{\Pi}^{(i+1)} \simeq \boldsymbol{\Pi}^{(i)} + \frac{\partial \boldsymbol{\Pi}}{\partial \boldsymbol{T}}^{(i)} (\boldsymbol{T}^{(i+1)} - \boldsymbol{T}^{(i)}) = \boldsymbol{0} \tag{2.32}$$

where i represents the i^{th} iteration. Note that we omitted the subscript n to simplify notation.

Iterations proceed until convergence (the norm of the residual meets a prescribed tolerance). Due to the high non linearity of the problem, a line-search method must be used in conjunction with the Newton-Raphson scheme. This type of globally convergent method is quite standard and its formulation can be found in most textbooks that consider nonlinear optimisation problems [60, 61].

2.3.3 Determination of the interface position

The determination of the interface position is essential for this method because the enriched shape functions depend on it.

In other enrichment formulations, the interface position is computed using an auxiliary level set equation. After this position is determined, the standard heat conduction equation is solved in each subdomain, enriching the elements that are intersected by the phase change boundary. Additionally, the constraint (2.6) is imposed through the use of Lagrange multipliers or a penalty formulation.

We are proposing a new way to determine the interface position implicitly at each Newton iteration, in terms of the values of the degrees of freedom corresponding to that iteration and the constraint given by equation (2.6). Suppose that we are processing iteration i and we have the guess values $\boldsymbol{T}^{(i)}$. With these guess values, we determine if the element that is being processed is intersected by the solidification front. If this is the case, the position of the phase change boundary is computed by using the constraint equation (2.6). A detailed presentation of this issue is given for the one dimensional case in section 2.3.6.2 and a discussion for the two dimensional case is presented in section 2.3.7.

2.3.4 Discontinuous integration

When processing an enriched element, a weak discontinuity appears in the element. In order to evaluate integrals in equations (2.25) to (2.31), a discontinuous integration procedure is used. The number of integration regions depends on the nature of the integrand, i.e. whether the integrand depends only on the time t_n or whether it depends on both time stages t_{n-1} and t_n. In the former case we have two integration subregions, while in the latter case we typically have three integration subregions with a continuous integrand in each of them.

For instance, suppose that we are processing an enriched element, and we are computing the term described by equation (2.26) with three subregions. Then, the elemental contribution $\boldsymbol{C}_e^*$ is given by

$$\boldsymbol{C}_e^* = \int_{\Omega^e} \rho c \boldsymbol{N}_n \boldsymbol{N}_{n-1}^T \, d\Omega = \sum_{p=1}^{3} \int_{\Omega^p} \rho c \boldsymbol{N}_n \boldsymbol{N}_{n-1}^T \, d\Omega \tag{2.33}$$

where p indicates the partition or region number (ranging in this case from one to three), Ω^p denotes each partition and Ω^e denotes the region of the element. The integration is performed numerically using a Gaussian quadrature in each sub-region [62].

2.3.5 Tangent matrix

After differentiating the nonlinear residual function (2.24) with respect to the generalised degrees of freedom $\boldsymbol{T}$, we obtain

$$\frac{\partial \boldsymbol{\Pi}}{\partial \boldsymbol{T}} = \frac{\boldsymbol{C}}{\Delta t} + \boldsymbol{K} + \frac{1}{\Delta t}\frac{\partial \boldsymbol{C}}{\partial \boldsymbol{T}}\boldsymbol{T}^{(i)} - \frac{1}{\Delta t}\frac{\partial \boldsymbol{C}^*}{\partial \boldsymbol{T}}\boldsymbol{T}_{n-1} + \frac{\partial \boldsymbol{K}}{\partial \boldsymbol{T}}\boldsymbol{T}^{(i)} + \frac{1}{\Delta t}\frac{\partial \boldsymbol{L}_n}{\partial \boldsymbol{T}} - \frac{1}{\Delta t}\frac{\partial \boldsymbol{L}_{n-1}}{\partial \boldsymbol{T}} + \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{T}} - \frac{\partial \boldsymbol{Q}}{\partial \boldsymbol{T}}. \tag{2.34}$$

The first two terms on the right-hand-side are standard in any nonlinear thermal problem. The other terms have certain particularities in the enriched elements, which will be described.

As previously stated, the computation of the mentioned terms in an enriched element depends on the number of subregions. We analyse the case of the term $\frac{\partial \boldsymbol{C}^*}{\partial \boldsymbol{T}}$; the other terms in equation (2.34) are computed similarly.

There are three sources of dependency of $\boldsymbol{C}^*$ on $\boldsymbol{T}$:

- *Evaluation dependency*: the integration region depends on the position of the discontinuity at the time stage t_n. Therefore, the position of the Gauss points in the physical space depends implicitly on the degrees of freedom $\boldsymbol{T}$.
- *Enrichment dependency*: the position for the enrichment is always determined in terms of the degrees of freedom $\boldsymbol{T}$, so the enrichment function definition depends on $\boldsymbol{T}$.
- *Integration region dependency*: when discontinuous integration is applied and the integration region is determined by the position of the interface at time stage t_n. An implicit dependency on the degrees of freedom $\boldsymbol{T}$ is present.

In what follows, a complete description of the computation of the tangent matrix for the one dimensional case is given.

2.3.6 One dimensional formulation

To evaluate in more detail the proposed ideas, we restrict first the analysis to the one dimensional case. In the numerical examples section we show the application of the one dimensional formulation for a series of problems.

2.3.6.1 Enrichment function definition

For the definition of the enrichment function we need to know the position of the interface. To describe that position, we make use of a level set function ϕ defined by

$$\phi = x - x_a \tag{2.35}$$

where x_a is given by

$$x_a = x_1 + s(x_2 - x_1) = x_1 + sh \tag{2.36}$$

with x_1 the position of the left node, x_2 the position of the right node and h the element length. The interface is located at the point where the level set function ϕ equals zero. This position is provided locally by the parameter $s \in (0, 1)$, while its global position is tracked with x_a.

Then, the definition of the enrichment function is given by

$$E(x) = \begin{cases} \dfrac{x - x_1}{x_a - x_1} = \dfrac{\phi_1 - \phi}{\phi_1} & x \leq x_a \\ \dfrac{x_2 - x}{x_2 - x_a} = \dfrac{\phi_2 - \phi}{\phi_2} & x > x_a \end{cases} \tag{2.37}$$

where $\phi_1 = x_1 - x_a$ and $\phi_2 = x_2 - x_a$. The temperature field inside an enriched element is described as

$$T^h(x, t) = \sum_i N_i(x) T_i + E(x, t) a \tag{2.38}$$

where the term $\sum N_i T_i$ corresponds to the usual finite element discretisation with N_i the shape functions and T_i the nodal degrees of freedom. The term $E(x, t)a$ corresponds to the enrichment, where $E(x, t)$ is the enrichment function and a is the associated degree of freedom. In the subsequent development linear shape functions are used. In matrix notation, $T^h \in \mathcal{V}^h$ inside an enriched element is given by

$$T^h = \boldsymbol{N}^T \boldsymbol{T} \tag{2.39}$$

where

$$\boldsymbol{N} = \begin{bmatrix} N_1(x) \\ N_2(x) \\ E(x, t) \end{bmatrix} \qquad \text{and} \qquad \boldsymbol{T} = \begin{bmatrix} T_1 \\ T_2 \\ a \end{bmatrix}. \tag{2.40}$$

An example of the previously described enrichment function is provided in Figure 2.1, with a discontinuity at $s = 0.35$. For one dimensional cases, this enrichment is conforming but this is not true for higher dimensions.

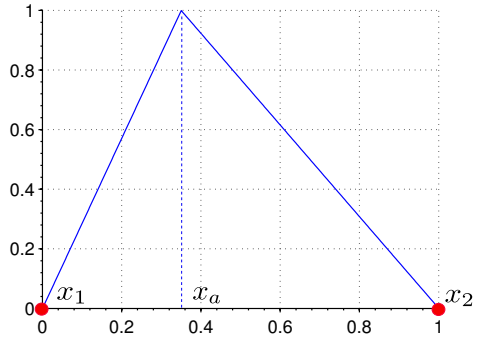


Figure 2.1: One dimensional enrichment function.

2.3.6.2 Determination of the interface position

As it was previously introduced, to determine the interface position we apply the constraint given by equation (2.6). When using the enrichment function defined in equation (2.37), a closed form of the value of the parameter s can be determined, which is given by

$$s = \frac{T_m - T_1^{(i)} - a^{(i)}}{T_2^{(i)} - T_1^{(i)}}. \tag{2.41}$$

Note that this procedure is based on physical features due to the determination of the interface position with the current temperature distribution.

2.3.6.3 Tangent Matrix

In this section we consider the case in which thermophysical properties do not depend on temperature. Contributions to the tangent matrix arising from temperature dependent thermophysical properties are considered in section 2.3.6.4.

Considering the dependencies outlined in section 2.3.5 and the example of the derivative $\frac{\partial \boldsymbol{C}^*}{\partial \boldsymbol{T}}$, the Gauss numerical discontinuous integration of equation (2.33) for the one dimensional case is given by

$$\boldsymbol{C}_e^* = \int_{\Omega^e} \rho c \boldsymbol{N}_n \boldsymbol{N}_{n-1}^T \, d\Omega = \sum_{p=1}^{3} \sum_{g=1}^{n_g} \rho c \boldsymbol{N}_n(x_g^{(p)}) \boldsymbol{N}_{n-1}^T(x_g^{(p)}) w_g \Omega^{(p)}. \tag{2.42}$$

where $\Omega^{(p)}$ is the area of the sub-region p. Then, the expression of the elemental contribution $\frac{\partial C^*_{(e)rk}}{\partial T_j}$ reads:

$$\begin{aligned}\frac{\partial C^*_{(e)rk}}{\partial T_j} = \sum_{p=1}^{3} \sum_{g=1}^{n_g} \rho c \Bigg[& \frac{\partial N_{n(r)}}{\partial x_g^{(p)}} \frac{\partial x_g^{(p)}}{\partial s} \frac{\partial s}{\partial T_j} N_{n-1(k)} w_g \Omega^{(p)} + \\ & N_{n(r)} \frac{\partial N_{n-1(k)}}{\partial x_g^{(p)}} \frac{\partial x_g^{(p)}}{\partial s} \frac{\partial s}{\partial T_j} w_g \Omega^{(p)} + \frac{\partial N_{n(r)}}{\partial x_a} \frac{\partial x_a}{\partial s} \frac{\partial s}{\partial T_j} N_{n-1(k)} w_g \Omega^{(p)} + \\ & N_{n(r)} N_{n-1(k)} w_g \frac{\partial \Omega^{(p)}}{\partial s} \frac{\partial s}{\partial T_j} \Bigg].\end{aligned} \tag{2.43}$$

The definition of x_a is given by equation (2.36). The first two terms in equation (2.43) arise from the *evaluation dependency*, the third term arises from the *enrichment dependency* and the fourth term arises from the *integration region dependency*. Computation of the terms $\frac{\partial x_g^{(p)}}{\partial s}$, $\frac{\partial x_a}{\partial s}$ and $\frac{\partial \Omega^{(p)}}{\partial s}$ is straightforward

$$\begin{aligned} \frac{\partial x_g^{(1)}}{\partial s} &= 0 & & & \frac{\partial \Omega^{(1)}}{\partial s} &= 0 \\ \frac{\partial x_g^{(2)}}{\partial s} &= \xi_2 h & \frac{\partial x_a}{\partial s} &= h & \frac{\partial \Omega^{(2)}}{\partial s} &= h \\ \frac{\partial x_g^{(3)}}{\partial s} &= (1-\xi_3)h & & & \frac{\partial \Omega^{(3)}}{\partial s} &= -h \end{aligned} \tag{2.44}$$

where $\xi_i, i = 1, 2, 3$ are the natural coordinates for each integration region.

The computation of the derivative $\frac{\partial s}{\partial T_j}$ is more cumbersome and is presented in the following paragraphs. This derivative expresses the core of the idea being exposed; it represents how the interface position varies in response to perturbations of the degrees of freedom $\boldsymbol{T}$.

The following preliminary results are needed

$$\frac{\partial E}{\partial x} = \begin{cases} -\dfrac{1}{\phi_1} & x \le x_a \\ -\dfrac{1}{\phi_2} & x > x_a \end{cases} \tag{2.45}$$

$$\frac{\partial E}{\partial x_a} = \begin{cases} \dfrac{\phi_1 - \phi}{\phi_1^2} & x \le x_a \\ \dfrac{\phi_2 - \phi}{\phi_2^2} & x > x_a \end{cases} \tag{2.46}$$

$$\frac{\partial E}{\partial s} = \begin{cases} h\dfrac{\phi_1 - \phi}{\phi_1^2} & x \le x_a \\ h\dfrac{\phi_2 - \phi}{\phi_2^2} & x > x_a \end{cases} \tag{2.47}$$

$$\frac{\partial E}{\partial s} + h\frac{\partial E}{\partial x} = \begin{cases} h\dfrac{\phi_1 - \phi}{\phi_1^2} - \dfrac{h}{\phi_1} = -\dfrac{h\phi}{\phi_1^2} & x \le x_a \\ h\dfrac{\phi_2 - \phi}{\phi_2^2} - \dfrac{h}{\phi_2} = -\dfrac{h\phi}{\phi_2^2} & x > x_a \end{cases} \tag{2.48}$$

The dependence of the temperature field on the position of the interface and on the degrees of freedom $\boldsymbol{T}$, was not explicitly specified in equation (2.38). Making these dependencies explicit and denoting T^h by T to simplify notation, we get

$$T(\boldsymbol{T}, x, s) = \sum_i N_i(x) T_i + E(x, s) a. \tag{2.49}$$

Considering the constraint given by equation (2.6), the temperature at the interface is

$$T(\boldsymbol{T}, x_a, s) = \sum_i N_i(x_a) T_i + a = T_m. \tag{2.50}$$

Linearising the increment of T, we obtain

$$dT = \frac{\partial T}{\partial \boldsymbol{T}} d\boldsymbol{T} + \frac{\partial T}{\partial x} dx + \frac{\partial T}{\partial s} ds. \tag{2.51}$$

Considering that the temperature at the interface should be equal to the melting temperature, we obtain

$$dT = \frac{\partial T}{\partial \boldsymbol{T}}\Big|_{x=x_a} d\boldsymbol{T} + \frac{\partial T}{\partial x}\Big|_{x=x_a} dx + \frac{\partial T}{\partial s}\Big|_{x=x_a} ds = 0. \tag{2.52}$$

Noting from equation (2.36) that $\frac{\partial x}{\partial s}\Big|_{x=x_a} = h$, we obtain

$$\frac{\partial T}{\partial \boldsymbol{T}}\Big|_{x=x_a} d\boldsymbol{T} + \Big(h\frac{\partial T}{\partial x} + \frac{\partial T}{\partial s}\Big)\Big|_{x=x_a} ds = 0. \tag{2.53}$$

Therefore, the partial derivative of s with respect to the generalised degrees of freedom $\boldsymbol{T}$ is

$$\frac{\partial s}{\partial \boldsymbol{T}} = -\left(\Big(h\frac{\partial T}{\partial x} + \frac{\partial T}{\partial s}\Big)\Big|_{x=x_a}\right)^{-1} \frac{\partial T}{\partial \boldsymbol{T}}\Big|_{x=x_a}. \tag{2.54}$$

The denominator in the equation (2.54) is computed as follows

$$h\frac{\partial T}{\partial x} + \frac{\partial T}{\partial s} = \sum_{i=1}^{2} h\frac{\partial N_i}{\partial x}T_i + \Big(\frac{\partial E}{\partial s} + h\frac{\partial E}{\partial x}\Big)a, \tag{2.55}$$

evaluating (2.55) at the interface $x = x_a$ becomes

$$\Big(h\frac{\partial T}{\partial x} + \frac{\partial T}{\partial s}\Big)\Big|_{x=x_a} = \sum_{i=1}^{2} h\frac{\partial N_i}{\partial x}(x_a)\, T_i. \tag{2.56}$$

By replacing the last result (2.56) into equation (2.54), and by evaluating $\frac{\partial T}{\partial \boldsymbol{T}}\Big|_{x=x_a}$ from equation (2.50), we obtain finally

$$\frac{\partial s}{\partial \boldsymbol{T}} = -\left(\sum_{i=1}^{2} h\frac{\partial N_i}{\partial x}(x_a)\, T_i\right)^{-1} \begin{bmatrix} N_1(x_a) \\ N_2(x_a) \\ 1 \end{bmatrix}. \tag{2.57}$$

2.3.6.4 Temperature dependent thermophysical properties

If we consider problems where the thermophysical properties depend on temperature, new contributions to the tangent matrix arise. These contributions to the elemental level, denoted by $\boldsymbol{\Upsilon}_e$, are given by

$$\boldsymbol{\Upsilon}_e = \sum_{p=1}^{3}\sum_{g=1}^{n_g}\Big[\rho \boldsymbol{N}_{n-1}^T \boldsymbol{T}_{n-1} w_g \Omega^{(p)} \boldsymbol{N}_n \frac{\partial c_n}{\partial \boldsymbol{T}}^T + \rho \boldsymbol{N}_n^T \boldsymbol{T} w_g \Omega^{(p)} \boldsymbol{N}_n \frac{\partial c_n}{\partial \boldsymbol{T}}^T + \nabla \boldsymbol{N}_n^T \boldsymbol{T} w_g \Omega^{(p)} \nabla \boldsymbol{N}_n \frac{\partial k_n}{\partial \boldsymbol{T}}^T\Big]. \tag{2.58}$$

The variation of the thermophysical property being considered with respect to $\boldsymbol{T}$ is calculated following the same procedure described before. As an example, we present the computation of the term $\frac{\partial c_n}{\partial \boldsymbol{T}}$. Considering that

$$\frac{\partial c_n(T)}{\partial \boldsymbol{T}} = \frac{\partial c_n(T)}{\partial T}\frac{\partial T}{\partial T_j}\boldsymbol{e}_j \tag{2.59}$$

we need to calculate the derivative $\frac{\partial T}{\partial \boldsymbol{T}}$ with the specific derivative $\frac{\partial c_p}{\partial T}$ for each considered material.

Taking the increment of the temperature field given by equation (2.49), we have

$$T(\boldsymbol{T}+d\boldsymbol{T}, x+dx, s+ds) = N_i(x+dx)[T_i+dT_i] + E(x+dx, s+ds)[a+da] \backsimeq \left[N_i(x) + \frac{\partial N_i}{\partial x}dx\right][T_i+dT_i] + \left[E(x,s) + \frac{\partial E}{\partial x}dx + \frac{\partial E}{\partial s}ds\right][a+da] \quad (2.60)$$

where N_i and E have been linearised. Considering that the differentials dx and ds are given by

$$dx = \frac{\partial x}{\partial s}ds \qquad ds = \frac{\partial s}{\partial T_j}dT_j + \frac{\partial s}{\partial a}da \quad (2.61)$$

and neglecting the higher order terms, we obtain

$$T(\boldsymbol{T}+d\boldsymbol{T}, x+dx, s+ds) = T(\boldsymbol{T}, x, s) + \left[N_j + \frac{\partial N_i}{\partial x}\frac{\partial x}{\partial s}\frac{\partial s}{\partial T_j}T_i + \frac{\partial E}{\partial x}\frac{\partial x}{\partial s}\frac{\partial s}{\partial T_j}a + \frac{\partial E}{\partial s}\frac{\partial s}{\partial T_j}a\right]dT_j + \left[E + \frac{\partial N_i}{\partial x}\frac{\partial x}{\partial s}\frac{\partial s}{\partial a}T_i + \frac{\partial E}{\partial x}\frac{\partial x}{\partial s}\frac{\partial s}{\partial a}a + \frac{\partial E}{\partial s}\frac{\partial s}{\partial a}a\right]da \quad (2.62)$$

Therefore, the partial derivative of T with respect to the generalised degrees of freedom $\boldsymbol{T}$ is

$$\frac{\partial T}{\partial \boldsymbol{T}} = \boldsymbol{N} + \left(\frac{\partial \boldsymbol{N}}{\partial x}\cdot\boldsymbol{T}\,\frac{\partial x}{\partial s} + a\,\frac{\partial E}{\partial s}\right)\frac{\partial s}{\partial \boldsymbol{T}} \quad (2.63)$$

2.3.6.5 Algorithmic implementation

Determination of when an element needs to be enriched follows an heuristic approach. The latent heat terms given by equations (2.28) and (2.29) are sources of high nonlinearity and this issue must be considered for implementation.

The determination of the integration region for the latent heat vector, equation (2.28), is presented as an example for the one dimensional case (algorithm 1). The degrees of freedom of the element are given by the vector `Tj=`$[T_1, T_2, a]$, the integration region is denoted by variable `xL`, and `x` represents the whole element domain.

The element is processed as a full solid element if the nodal temperatures are below the melting temperature, and if the absolute value of the enrichment degree of freedom, denoted by a, is below a certain threshold `minT3` (an element in which $a >$ `minT3` still has latent heat inside and cannot be considered as a full solid element). The same applies for the determination of the liquid state, but this time the nodal temperatures are above the melting temperature.

If the element is neither liquid nor solid, the element is supposed to be in a phase change state, i.e., one portion of the element is liquid and the other is solid. In this case, the interface position parameter s is determined using equation (2.41).

If parameter s is admissible, i.e., if s falls within the range $(0, 1)$, then the element is enriched (lines 36 to 46) and the integration region `xL` is computed. The number identifying the element being enriched and the local interface position parameter s are tracked by the variable `ifpos`.

When the element is supposed to be in phase change state, it could be possible for local parameter s to exceed the range $(0, 1)$ when it is computed. In this situation, to

Code 1 Integration Region

```
Tmax = max(Tj(1:2));
Tmin = min(Tj(1:2));
solid = false; %if the element is solid,
               %then solid=true.
               %Otherwise, solid=false
%the element is full solid
if (Tmax < Tm & abs(Tj(3))<minT3)
    xL = [x(2);x(2)];
    solid = true;
elseif (Tmin > Tm & abs(Tj(3))<minT3)
    %the element is full liquid
    xL=x;
else
    %parameter s determination
    s = parameterS(Tj,Tm);
    if (s>=1|s<=0) | ((s<minS | s>(1-minS))
                        & abs(Tj(3))<minT3)
        if(Tmax < Tm)
            solid = true;
        elseif (Tmin > Tm)
            solid = false;
        else
            diffT = Tj-Tm;
            index = max(abs(diffT(1:2)));
            if(sign(diffT(index))==-1)
                solid = true;
            else
                solid = false;
            end
        end
        xL = x;
        if solid
            xL = [x(2);x(2)];
        end
    else
        ifpos = [elementNumber,s];
        %the element is enriched
        eAct = true;
        %phase change element, x1 is solid
        if(Tj(2)>Tj(1))
            xL = [ x(1)+s*h; x(2)];
        else
            %phase change element
            %x1 is liquid
            xL = [ x(1); x(1)+s*h];
        end
    end
end
```

avoid the divergence of the Newton-Raphson scheme the element is not processed as a phase change element, instead it is considered liquid or solid. In this context to determine the element state, two cases must be considered:

- It could happen that the nodal temperatures are both below or above the melting temperature, but with the absolute value of the parameter a being greater than the threshold `minT3`. In this case, the element is considered liquid or solid based on the nodal temperatures values (lines 18 to 22).

- If the nodal temperatures are not either all below or all above the melting temperature, the element state is computed based on the relative value of the nodal temperatures with respect to the melting temperature. That is, we determine the maximum relative temperature taking into account absolute values. Then, if the sign associated to that maximum is negative, the element is processed as solid. If this is not the case, it is considered liquid. This issue is addressed in lines 22 to 30 of the code.

Finally, if the interface is located too close to an element node position and the enrichment parameter is too low, the Newton-Raphson scheme can lead to divergence (note that in this case the element is almost fully solid or fully liquid). In order to avoid this situation, whenever the parameter s is below the threshold `minS` or above the threshold 1-`minS`, and the absolute value of the enrichment parameter a is below the threshold `minT3`, the element is not enriched (second part of the conditional, line 16 of the code).

The threshold values are determined as follows. The threshold `minT3` for the enrichment a is given by

$$\texttt{minT3} = \text{tol}_T T_c n^{1/d} \tag{2.64}$$

where T_c is a characteristic temperature of the problem, n is the number of elements, d is the number of spatial dimensions and tol_T is a tolerance set equal to 10^{-5}. The threshold `minS` for the parameter s is set equal to 5×10^{-5}.

2.3.7 Extension to two dimensions

A discussion on the two dimensional extension of the discrete formulation is presented. Its implementation is left as future work. The enrichment shape functions E for the two dimensional case can be generated as described by Soghrati *et al.* [63].

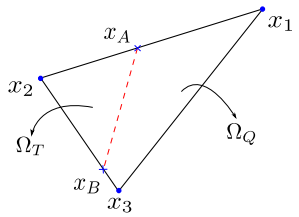


Figure 2.2: Example of a two-dimensional phase change element.

Let us consider a phase change element as displayed in Figure 2.2. For the current analysis, suppose that $T_2 > T_m$, $T_1 < T_m$ and $T_3 < T_m$; then, the interface intersects the sides l_{12} and l_{23} of the element as shown in the Figure. The intersection points of the interface with the boundary are denoted by $\boldsymbol{x}_A$ and $\boldsymbol{x}_B$.

We divide the element in its *bilinear* subregion Ω_Q and its *linear* subregion Ω_T. Then, two enrichment functions corresponding to points $\boldsymbol{x}_A, \boldsymbol{x}_B$ can be defined as

$$E_I(\boldsymbol{x}) = \begin{cases} Q_I(\boldsymbol{x}) & \boldsymbol{x} \in \Omega_Q \\ N_I(\boldsymbol{x}) & \boldsymbol{x} \in \Omega_T \end{cases}, \qquad I = A, B. \tag{2.65}$$

The temperature interpolation inside an enriched element is then given as:

$$T^h(\boldsymbol{x}, s_A, s_B) = \sum_i N_i(\boldsymbol{x}) T_i + E_A(\boldsymbol{x}, s_A) a + E_B(\boldsymbol{x}, s_B) b \, . \tag{2.66}$$

where $N_i(\boldsymbol{x})$ are the standard linear shape functions, and where parameters s_A, s_B express the position of points $\boldsymbol{x}_A, \boldsymbol{x}_B$ in terms of the nodal positions of the element:

$$\boldsymbol{x}_A = \boldsymbol{x}_1 + s_A(\boldsymbol{x}_2 - \boldsymbol{x}_1) \tag{2.67}$$

$$\boldsymbol{x}_B = \boldsymbol{x}_2 + s_B(\boldsymbol{x}_3 - \boldsymbol{x}_2). \tag{2.68}$$

Note that, from the definition of the enrichment function, the interface is straight inside a phase change element. To compute (2.65), we need to know positions $\boldsymbol{x}_A$ and $\boldsymbol{x}_B$.

The value of s_A, s_B will be determined from the requirement that the temperature at the interface should be equal to the melting temperature T_m, similarly to what we made in the one dimensional case. Then, we write

$$T_m = T_1 N_1^B(s_A) + T_2 N_2^B(s_A) + a, \tag{2.69}$$

where $N_1^B(s_A)$ and $N_2^B(s_A)$ are the restrictions of the shape functions $N_1(\boldsymbol{x})$ and $N_2(\boldsymbol{x})$ to the boundary l_{12}, and a is the amplitude of the enrichment function $E_A(\boldsymbol{x})$.

Following the same procedure used for the one dimensional case, we can show that:

$$s_A = \frac{T_m - T_1 - a}{T_2 - T_1}. \tag{2.70}$$

The expression for s_B is obtained in a similar form.

2.4 Numerical examples

The method was tested in a series of challenging one dimensional problems, i.e. problems with extreme values of temperature gradient discontinuity and problems with initial temperatures close to the melting temperature. The obtained results were compared with analytic solutions and with the results obtained from a fixed mesh numerical scheme, for which the temperature gradient discontinuity was not considered. Also, results for the first problem are compared against results obtained with an existing enriched formulation.

In the following examples, we use the normalised residual norm to ascertain the convergence of the Newton-Raphson scheme. This normalised residual norm is given by

$$\frac{\|\boldsymbol{\Pi}\|}{\|\boldsymbol{F_r}\| + \|\boldsymbol{F_c}\| + \|\boldsymbol{F_k}\|}, \tag{2.71}$$

where $\| \boldsymbol{.} \|$ denotes the L_2 norm of $\boldsymbol{.}$ and

$$\boldsymbol{F_r} = \frac{\boldsymbol{L}_n - \boldsymbol{L}_{n-1}}{\Delta t} + \boldsymbol{F} - \boldsymbol{Q} \tag{2.72}$$

$$\boldsymbol{F_c} = \frac{\boldsymbol{C}\boldsymbol{T}_n}{\Delta t} - \frac{\boldsymbol{C}^*\boldsymbol{T}_{n-1}}{\Delta t} \tag{2.73}$$

$$\boldsymbol{F_k} = \boldsymbol{K}\boldsymbol{T}_n. \tag{2.74}$$

2.4.1 Problem I: Dirichlet/Dirichlet boundary conditions

In this problem we show the performance of the proposed enriched formulation, comparing results with those obtained by Merle and Dolbow [56]. A solution obtained using the method without enrichment by Fachinotti *et al.* [28] is shown. All solutions are compared to the exact solution, computed for a semiinfinite bar. A similar problem was also treated in reference [21].

The problem consists in the freezing of a long slab that was initially at temperature T_0 above the melting temperature T_m and is suddenly cooled by imposing a constant temperature $T_1 < T_m$ to the slab end $x = 0$. The temperature at the slab end $x = L$ is held at $T_0 > T_m$, where L is the slab length. The parameters of the problem are given in table 2.1. The Stefan number St_s for this problem is given by

$$\mathrm{St}_s = \frac{c_s(T_m - T_1)}{\mathcal{L}} = 0.255. \tag{2.75}$$

Table 2.1: Problem I parameters

$\mathcal{L}$	c_s	c_l	k_s	k_l
$19.2\frac{\mathrm{J}}{\mathrm{kg}}$	$0.49\frac{\mathrm{J}}{^\circ\mathrm{Ckg}}$	$0.62\frac{\mathrm{J}}{^\circ\mathrm{Ckg}}$	$9.6\times10^{-3}\frac{W}{^\circ\mathrm{Ccm}}$	$6.9\times10^{-3}\frac{W}{^\circ\mathrm{Ccm}}$
ρ	T_m	T_1	L	T_0
$1\frac{\mathrm{kg}}{\mathrm{cm}^3}$	$0\,^\circ\mathrm{C}$	$-10\,^\circ\mathrm{C}$	10cm	$4\,^\circ\mathrm{C}$

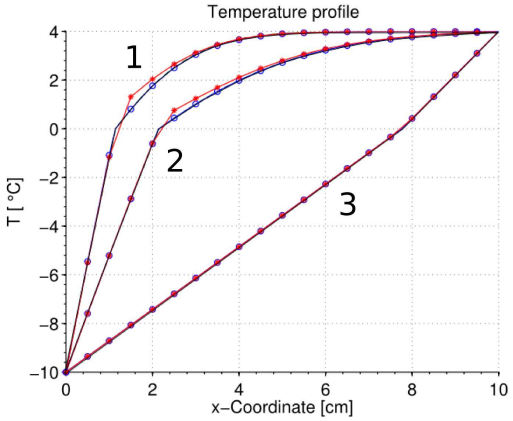


Figure 2.3: Temperature profile at different time instants: 1) $t = 180$ s; 2) $t = 360$ s; and 3) steady-state solution. Solid line: Merle and Dolbow's solution; dashed line: exact solution; asterisks: solution without enrichment; and circles: solution with the proposed enrichment. Merle and Dolbow's, exact and the solution with the proposed method are all coincident.

Twenty equally spaced elements and a time step of 12.76s are used to model this problem. The computed solution is shown in Figure 2.3 for different time instants. Perfect agreement with the results obtained by Merle and Dolbow [56] and with the exact solution can be observed. The solution obtained with the method without enrichment displays a small error near the interface between solid and liquid zones.

Table 2.2: Problem I: Normalised residual norm for selected time steps for the proposed method.

Iter.	t 12.76 s	t 127.6 s	t 446.6 s	t 1339.8 s
0	0.9995	0.6973	0.6677	0.6684
1	0.8778	0.0616	0.0236	0.0266
2	0.8820	0.0001	0.0002	0.0003
3	0.6078	2.7e-9	1.1e-8	1.7e-8
4	0.3627	1.2e-15	4.3e-15	1.1e-14
⋮	⋮			
6	0.0035			
7	3.7e-6			
8	2.5e-12			

The proposed numerical scheme took an average of 2.55 iterations per time to get the solution, with a maximum of 8 iterations in the first time step. On the other hand, Merle and Dolbow reported less than 15 iterations to get a solution with their XFEM method. In table 2.2 the evolution of the normalised residual norm for a number of representative time steps is given, showing quadratic convergence in all cases.

2.4.2 Problem II: Dirichlet / Dirichlet boundary conditions with a low Stefan number

Again, the freezing of a long slab is analysed; nevertheless, a much larger value of latent heat is now used. In this way, the behaviour of the proposed numerical formulation with very low Stefan numbers is studied. The slab is initially at temperature T_0 above the melting temperature T_m and is suddenly cooled by imposing a constant temperature $T_1 < T_m$ to the slab end $x = 0$. The temperature at the slab end $x = L$ is held at $T_0 > T_m$, where L is the slab length. The parameters of the problem are given in table 2.3. The Stefan number St_s for this problem is given by

$$\mathrm{St}_s = \frac{c_s(T_m - T_1)}{\mathcal{L}} = 0.025 \tag{2.76}$$

Table 2.3: Problem II parameters

$\mathcal{L}$	c_s	c_l	k_s	k_l
$190.26\frac{\mathrm{J}}{\mathrm{kg}}$	$0.49\frac{\mathrm{J}}{^\circ\mathrm{Ckg}}$	$0.62\frac{\mathrm{J}}{^\circ\mathrm{Ckg}}$	$9.6\times10^{-3}\frac{W}{^\circ\mathrm{Cm}}$	$6.9\times10^{-3}\frac{W}{^\circ\mathrm{Cm}}$
ρ	T_m	T_1	L	T_0
$1\frac{\mathrm{kg}}{\mathrm{m}^3}$	$0\,^\circ\mathrm{C}$	$-10\,^\circ\mathrm{C}$	10m	$4\,^\circ\mathrm{C}$

Sixteen equally spaced elements and a time step of 18s are used to model this problem. The computed solution is shown in Figure 2.4 for different time instants and is compared with the analytic solution for a semi-infinite medium [64] showing almost perfect

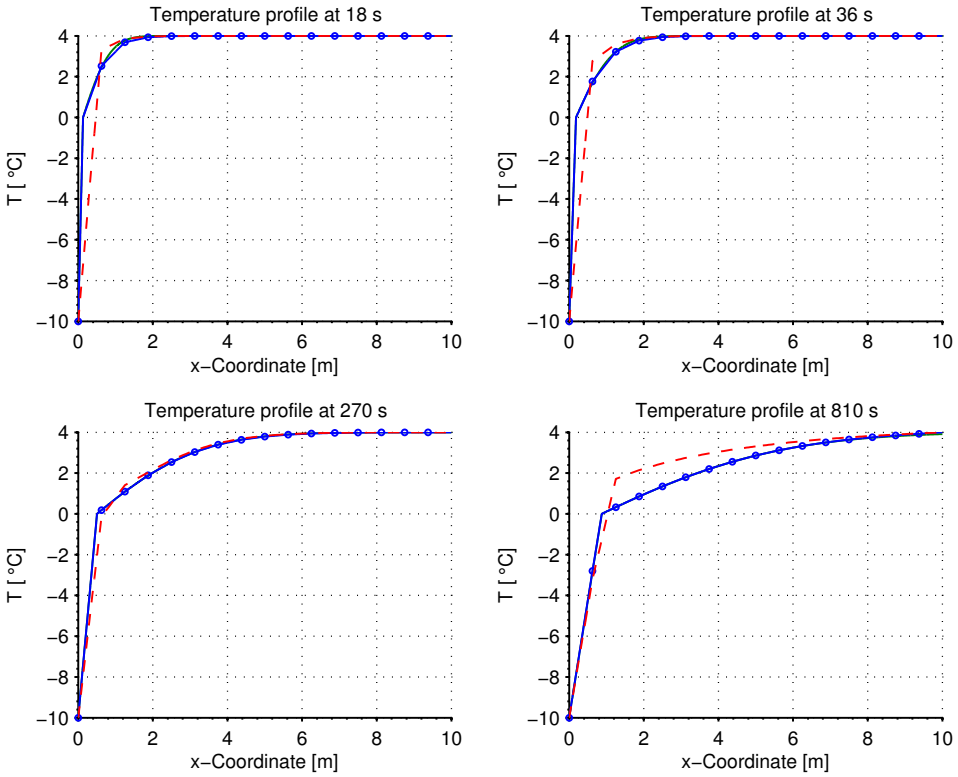


Figure 2.4: Solution of problem II at different time steps. Solid line: the exact solution; dashed line: approximate solution without enrichment; solid line with circles: approximate solution obtained with the proposed method.

Table 2.4: Problem II: Normalised residual norm evolution for the proposed formulation at selected time steps.

Iter.	Time 18 s	Time 126 s	Time 234 s	Time 324 s
0	0.9997	0.8926	0.8453	0.8208
1	0.9903	0.0159	0.0097	0.0076
2	0.9770	0.0001	4.7e-5	1.7e-5
3	0.9575	1.7e-9	3.7e-10	1.1e-10
4	0.9267	1.1e-15		
⋮	⋮			
10	0.0006			
11	5.8e-7			
12	4.8e-13			

agreement. Additionally, the solution obtained with the formulation without enrichment proposed by Fachinotti *et al.* [28] is shown in dashed lines. The solution without enrichment presents spurious oscillations and, in certain time instants, is quite different from the analytic solution.

Figure 2.5 displays the interface evolution over time. Again, the current method shows almost perfect agreement with the analytic solution, while the formulation without

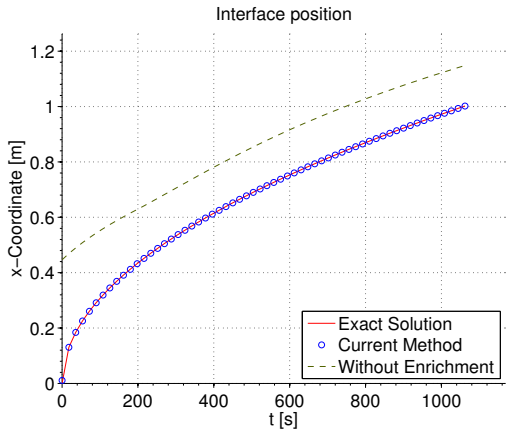


Figure 2.5: Interface position with exact and approximate values.

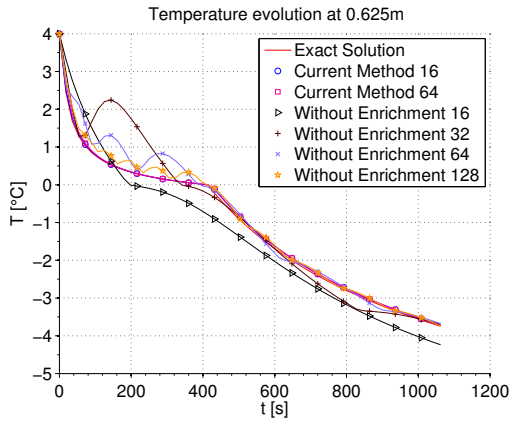


Figure 2.6: Temperature evolution. Exact and approximate values for different number of equally spaced elements.

enrichment differs from the analytic solution.

A quadratic convergence rate was observed in all time steps. The mean number of iterations per time step was 3.71, and the maximum number of iterations at a given time

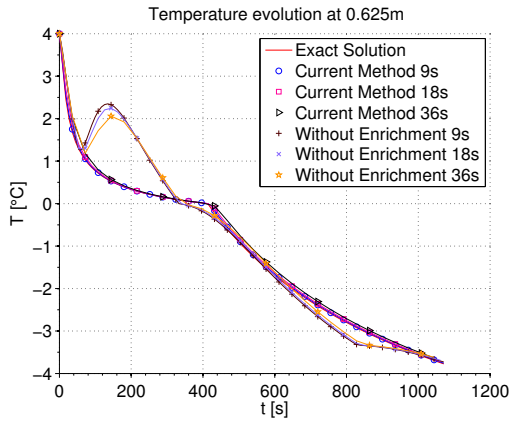


Figure 2.7: Temperature evolution. Exact and approximate values for different time steps.

step was 12. These values are similar to those for the fixed mesh technique, which required 3.32 iterations per time step, with a maximum of 7 iterations.

In table 2.4 the evolution of the residual for a number of representative time steps can be observed. The maximum number of iterations is attained at the first time step ($t = 18$ s) because at the initial time ($t = 0$ s), the interface is located too close to the first node to accurately represent the initial condition (a Heaviside at $x = 0$). It should be noted that the element's capability to represent this type of temperature distribution causes the observed accurate results. If the initial position of the interface is placed farther from $x = 0$, the number of iterations will be smaller, but the approximation to the exact solution of the problem will be less accurate. The errors with the method without enrichment are due, in part, to the impossibility of this formulation of representing this initial condition.

Figure 2.6 provides the temperature evolution at $x = 0.625$m for meshes of 16, 32, 64 and 128 equally spaced elements. In all cases a time step of 18s was used. Results for the method without enrichment are shown for all meshes. We can see that an error appears even for the mesh with 128 elements. The proposed method of enrichment, gives results which are almost coincident with the exact results, for a mesh of only 16 elements. Computations with 64 elements are completely coincident with the 16 elements results.

The method without enrichment using 32 equally spaced elements displays spurious oscillations of increasing amplitude with respect to the mesh with 16 elements. These oscillations are observed in all computations with the method without enrichment. This fact can hinder its use for the estimation of temperature time rates which are necessary, i.e. for the computation of microstructure.

Figure 2.7 provides the temperature evolution at $x = 0.625$m for time steps 9s, 18s and 36s, with a mesh of 32 equally spaced elements. The conclusions obtained for this experiment are similar to those obtained for the previous experiment. Our method gives accurate results without temporal oscillations, whereas the method without enrichment presents spurious oscillations.

2.4.3 Problem III: Dirichlet/Neumann boundary conditions

Here, we study the melting of a long slab initially at temperature T_0 below the melting temperature T_m that is suddenly heated by imposing a constant temperature $T_1 \gg T_m$ at the slab end $x = 0$. The slab end $x = L$ is insulated, where L is the slab length. The analytic solution for a semi-infinite medium associated with this problem can be found in [65].

The thermophysical parameters are considered constant and equal in both phases, such that $c = c_s = c_l$ and $k = k_s = k_l$. The problem parameters are given in table 2.5.

Table 2.5: Parameters of problem III

$\mathcal{L}$	c	k	ρ
$190.26\frac{\text{J}}{\text{kg}}$	$1\frac{\text{J}}{^\circ\text{Ckg}}$	$1.08\frac{W}{^\circ\text{Cm}}$	$1\frac{\text{kg}}{\text{m}^3}$
T_m	T_1	L	T_0
$-0.1\,^\circ\text{C}$	$45\,^\circ\text{C}$	4m	$-4\,^\circ\text{C}$

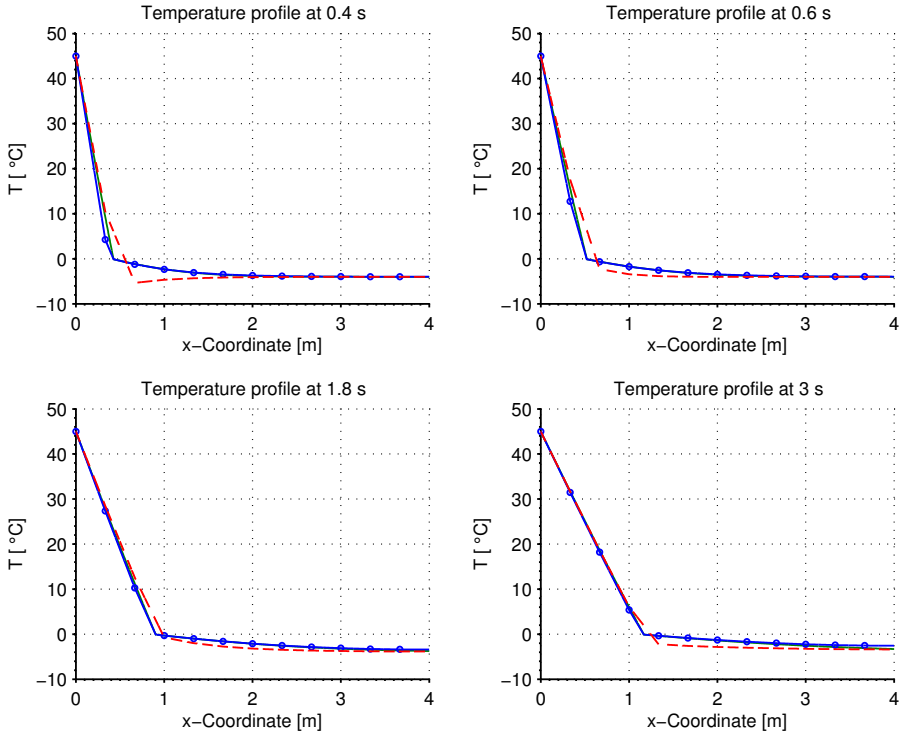


Figure 2.8: Solution of problem III at different time steps. The solid line is the exact solution. The dashed line is the approximate solution without enrichment. The solid line with circles is the approximate solution obtained with the proposed method.

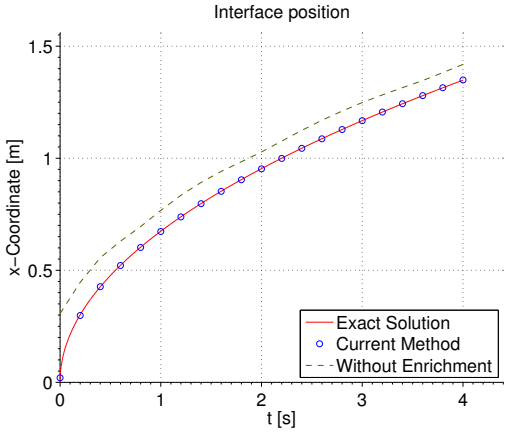


Figure 2.9: Interface position with exact and approximate values.

Twelve equally spaced elements and a time step of 0.2s are used to model this problem. The computed solution is shown in Figure 2.8 for different time instants and is compared with the analytic solution, showing almost perfect agreement. Additionally, the solution obtained using the formulation without enrichment, proposed by Fachinotti *et al.* [28], is shown with dashed lines. The solution without enrichment displays spurious oscillations, and in certain time instants is quite different from the analytic solution.

Figure 2.9 displays the interface evolution over time. Again, the method with enrichment shows nearly perfect agreement with the analytic solution, while the formulation

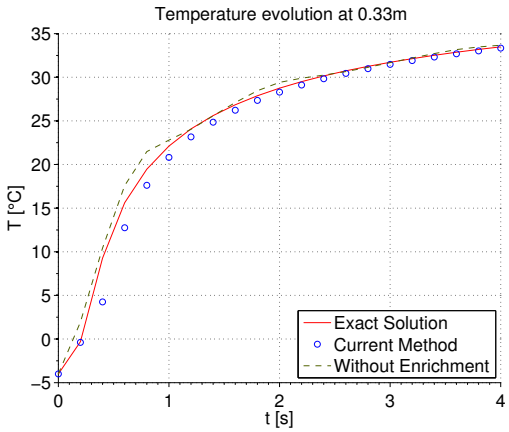


Figure 2.10: Temperature evolution with exact and approximate values.

without enrichment does not. In Figure 2.10 the temperature evolution at node 2 ($x = 0.33$ m) is provided. The enriched solution follows the analytic solution quite well, while the solution without enrichment has severe oscillations.

The Stefan number St_l for this problem is given by

$$\mathrm{St}_l = \frac{c_l(T_l - T_m)}{\mathcal{L}} = 0.23704. \tag{2.77}$$

The computational effort in this problem was important because of the high nonlinearity from our formulation. The mean number of iterations per time step was 12, and the maximum number of iterations at a given time step was 35. These values are much greater than those of the fixed mesh technique, which required 4.95 iterations per time step with a maximum of 8 iterations. Nevertheless, the computed solution showed high accuracy, such that even though the Stefan number was small, the solution was quite close to the analytic solution and did not have spurious oscillations (these oscillations are undesirable, e.g., when computing the microstructure, which depends on the temperature history and on the temperature rates).

2.4.4 Problem IV: simplified Tig-wash problem

This problem studies the behaviour of the proposed method under the presence of a moving heat source and with typical thermal dependent material properties. Figure 2.11 shows a sample square tube part made by Shaped Metal Deposition (SMD) of Ti-6Al-4V alloy that was built by deposing material with a welding Tig robot [41]. A simplified one dimensional model of this part is analysed during the "Tig-wash" procedure, where the part is heated using the welding torch without deposing the material.

The dimensions of the cross-section of the bead are $d_1 = 9.83$mm wide and $d_2 = 70$mm high. Initially the bead is at room temperature $T_e = 299$K. Suddenly the heat source starts to heat the body from the top, travelling at a known velocity. At the same time, the part is cooled through the top and side walls by air convection and radiation (Robin boundary conditions are assumed for combined convection and radiation). The base of the bead is considered to remain at constant temperature T_e.

We assume that heat conduction along the longitudinal and transversal directions can be neglected to build a simplified one-dimensional model. The heat source can be described

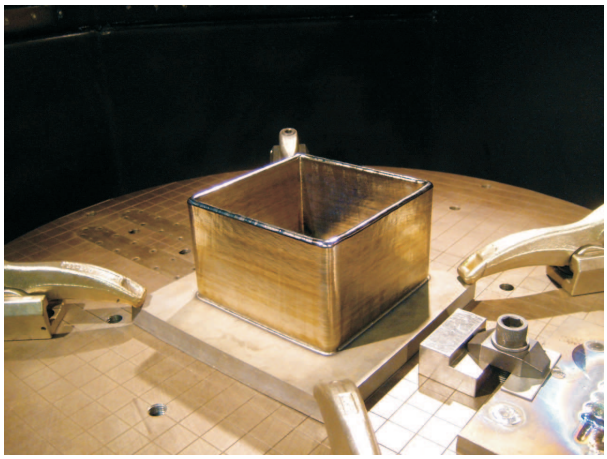


Figure 2.11: Square cylinder produced by SMD.

using a simplification of the Goldak heat source [66], which is obtained as follows for the one dimensional case [42]

$$Q_s(t) = \frac{\sqrt{3}Q}{\sqrt{\pi}d_1} \begin{cases} \frac{f_f}{c_f} \exp\left(-3\frac{z(t)^2}{c_f^2}\right) & \text{if } z(t) \leq 0 \\ \frac{f_r}{c_r} \exp\left(-3\frac{z(t)^2}{c_r^2}\right) & \text{if } z(t) > 0 \end{cases} \tag{2.78}$$

where c_f and c_r are the length parameters associated with the axis of the front and rear semi-ellipsoids, f_f and f_r are the portion of heat distributed in the front and rear semi-ellipsoids, respectively, and Q is the total heat input. The function $z(t)$ gives the variation of position of the heat source relative to the plane containing the cross section of the bead.

The material properties and other parameters of the problem are described in table 2.6. The heat capacity c, the thermal conductivity k and the thermal convection / radiation coefficient h_f depend on the temperature and are given by the expressions [41]

$$\begin{aligned} c\left[\frac{\mathrm{J}}{\mathrm{kg\ K}}\right] &= 0.17919\ T[\mathrm{K}] + 495.20 \\ k\left[\frac{\mathrm{W}}{\mathrm{mK}}\right] &= \begin{cases} 0.0107\ T[\mathrm{K}] + 4.6619 & T \leq 1905\mathrm{K} \\ 0.5733\ T[\mathrm{K}] - 1067.2 & T > 1905\mathrm{K} \end{cases} \\ h_f\left[\frac{\mathrm{W}}{\mathrm{m^2K}}\right] &= 0.26\ T[\mathrm{K}] - 55. \end{aligned}$$

The variation in time of the total heat input at the considered cross-section $Q_s(t)$ for the proposed parameter values is given in Figure 2.12.

Note that the Robin boundary conditions on the side walls are represented in the one dimensional model as a sink. These terms, denoted by $\boldsymbol{\Xi}$, are

$$\boldsymbol{\Xi} = \frac{2}{d_1}\int_\Omega h_{f_n} \boldsymbol{N}_n \boldsymbol{N}_n^T\, d\Omega \boldsymbol{T}\ - \frac{2}{d_1}\int_\Omega h_{f_n} \boldsymbol{N}_n T_f\, d\Omega, \tag{2.79}$$

with the elemental contribution $\boldsymbol{\Xi}_e$ given by

$$\boldsymbol{\Xi}_e = \frac{2}{d_1}\sum_{p=1}^{3}\sum_{g=1}^{n_g}\left[h_{f_n} \boldsymbol{N}_n \boldsymbol{N}_n^T w_g \Omega^{(p)} \boldsymbol{T}\ - h_{f_n} \boldsymbol{N}_n T_f w_g \Omega^{(p)}\right]. \tag{2.80}$$

Table 2.6: Parameters of problem IV

$\mathcal{L}$	T_m	ρ	Q	Torch speed
$292600\frac{\text{J}}{\text{kg}}$	1905K	$4430\frac{\text{kg}}{\text{m}^3}$	1512J	$5\frac{\text{mm}}{\text{s}}$
c_r	c_f	f_f	f_r	T_f
10mm	2.5mm	0.4	1.6	299K

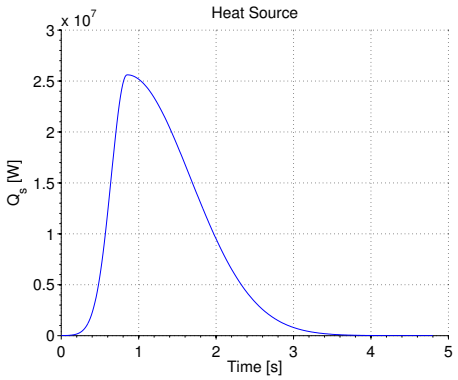


Figure 2.12: Heat source variation.

If we consider the temperature dependence of the thermal convection coefficient h_{f_n}, new contributions to the tangent matrix arise, which are given at the elemental level by

$$\boldsymbol{\Upsilon}_e^{h_f} = \frac{2}{d_1}\sum_{p=1}^{3}\sum_{g=1}^{n_g}\left[\boldsymbol{N}_n^T\boldsymbol{T}_n w_g \Omega^{(p)}\boldsymbol{N}_n\frac{\partial h_{f_n}}{\partial \boldsymbol{T}}^T - T_f w_g\Omega^{(p)}\boldsymbol{N}_n\frac{\partial h_{f_n}}{\partial \boldsymbol{T}}^T\right]. \tag{2.81}$$

A time step of 0.015s and 32 equally spaced elements are used to model this problem. The results obtained after running the simulation are shown in Figures 2.13 and 2.14. In Figure 2.14, a non physical variation can be observed in the initial transient due to a thermal shock problem. The maximum number of iterations was 4 iterations with a mean of 2.01 iterations per time step.

The temperature evolution over time at $x = 70$mm is shown in Figure 2.15a. To understand the phenomenon observed in this Figure, the temperature space profile at $t = 3.279$s is shown in Figure 2.15b. This behaviour is physically correct, as it represents the initial solidification of the weld surface with some liquid inside, which is a typical behaviour observed during the welding processes [67]. To show these details with high resolution a time step of 0.003s was used to obtain the results.

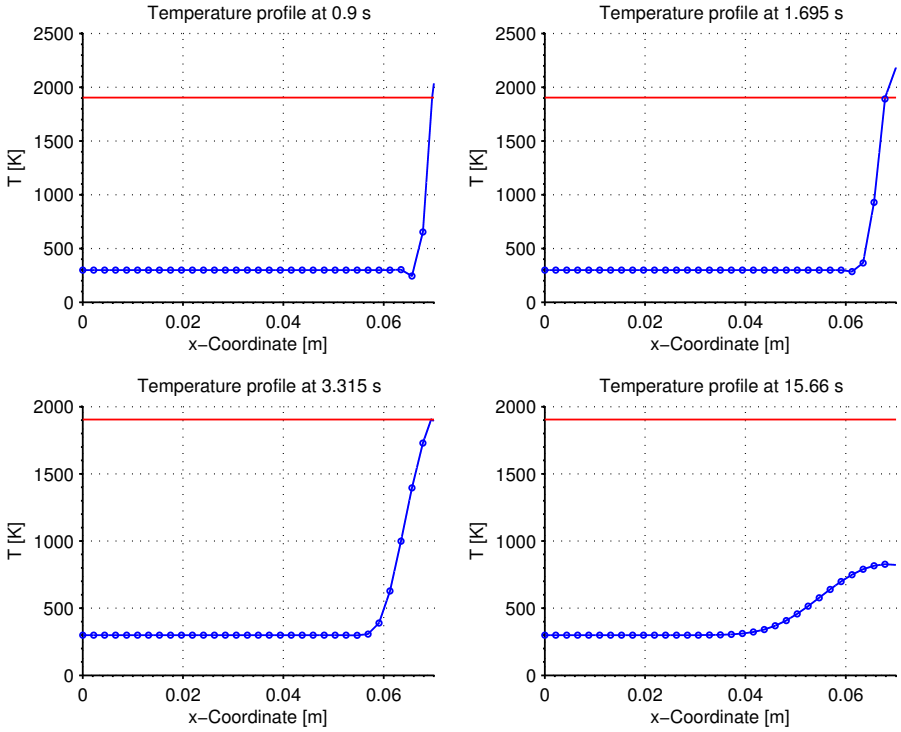


Figure 2.13: Temperature profile at different time steps.

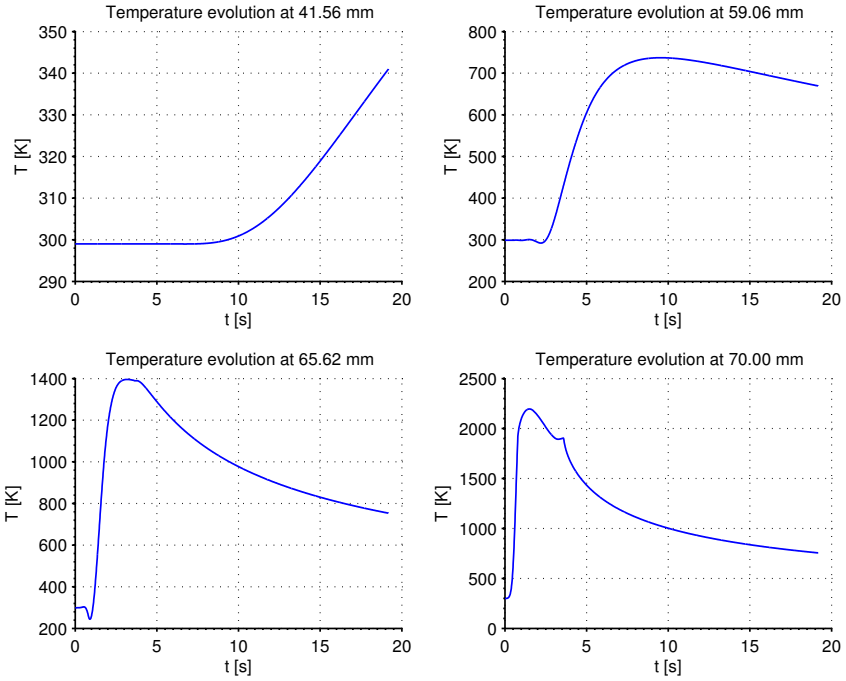
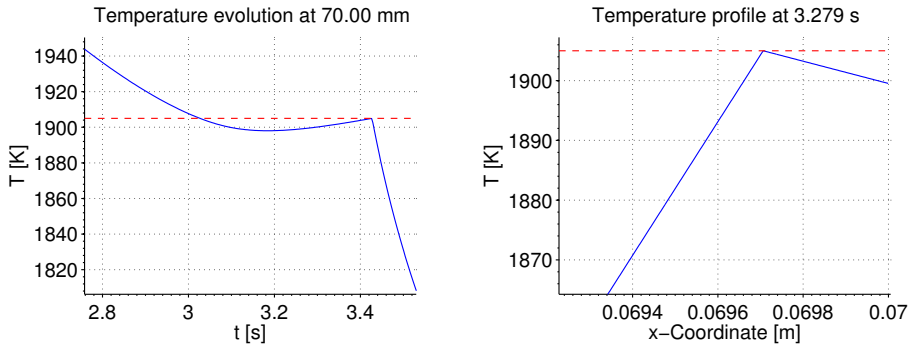


Figure 2.14: Temperature evolution at different points.



(a) Detail: temperature evolution at 70mm.

(b) Detail: temperature at 3.279s.

Figure 2.15: Details of problem IV (melting temperature is the dashed line and temperature evolution is the solid line).

2.5 Conclusions

A new approach for solving isothermal phase change problems was presented. The method has the advantages of a fixed mesh method that does not need remeshing to conform to the phase change interface, but also introduces the possibility to represent the discontinuity in the temperature gradient at the solidification front by enriching locally where it is necessary. The proposed method also avoids the use of an auxiliary equation to determine the enrichment position, which is common for level set formulations; instead, this position is determined with a constraint that imposes that the temperature at the phase change boundary must be the melting temperature.

Different test problems were investigated. The proposed method verified these tests giving results that were more accurate than with a formulation without enrichment and without presenting any spurious oscillations. Additionally, a simplified welding problem for a Titanium alloy was presented to study the behaviour of the current method under the influence of a moving heat source and for typical material values. This method is able to handle this type of problem. In future work, the formulation of two and three-dimensional implementations will be considered.

Chapter 3

Microstructure Evolution

The description of microstructure evolution in thermally driven processes is of fundamental importance, due to the fact that the thermal state induces changes at the microstructural level that strongly affects the material response. Usually, a particular model for predicting microstructure evolution is restricted to a reduced number of materials. When a material differs too much from that implemented, a new model must be coded. The main objective of this work is to design a computational tool with enough flexibility for representing a wide range of materials, thus reducing the need of recoding. The Object Oriented philosophy is chosen as a paradigm for the development of the solution. In order to gather the requirements of such a system, two different materials are analysed, the AISI 1060 steel and the Titanium alloy Ti6Al4V. In the case of Ti6Al4V, models of microstructure evolution for arbitrary thermal histories are considered. Application examples are presented to show the capability of the developed computational framework and models.

3.1 Introduction

Modelling material properties variations with temperature is of fundamental importance in thermally driven processes. Examples of application include heat treatment and welding. The latter are more challenging because they involve heating/cooling thermal histories. Moreover, in these problems the description of microstructure is important not only because thermal expansion is the driving force characterising the thermal/mechanical interaction but also because the velocity of changes in the temperature field may induce different final microstructure. It is well-known that in order to correctly determine residual stresses in welding processes, the effects of microstructure evolution on material properties must be taken into account [5].

Various models for describing microstructure evolution have been proposed in the literature. They can be classified into two large groups, depending on the length scale at which they are formulated [6]. The oldest ones, the more amenable for large scale simulations, are models formulated at the macro-scale. In these models, the microstructure is described in terms of a set of internal state variables with statistical information of the process, such as phase fractions, grain size and hardness. On the other hand, modern approaches are formulated at the microstructure length scale. They are computational intensive and, for this reason, difficult to be used in large scale applications. This kind of modelling is usually referred to as meso-scale modelling, and a representative technique is the Phase Field Modelling [68]. In this work, macro-scale based approaches are chosen for

modelling microstructure evolution because of their low computational complexity, which allows simulating large scale welding problems [3].

Each model for microstructure evolution is restricted to a given set of materials. Different materials have different models, which usually require coding of a new software or software part. The objective of this work is to design and implement a code able to handle a wide range of materials, whose behaviour is described from the problem data.

Flowcharts are of common use in computer science for specifying an algorithm or set of steps to solve a given problem [69]. In this work, we extend the use of flowcharts to fully define a material, by specifying in the dataset of the simulation problem a flowchart aimed at describing the microstructure kinetics. Two types of boxes or components are used in the material flowcharts: decisions, and processing steps which will be generally related to a type of transformation. Every component of the flowchart is coded just once, and eventually reused when defining a new material, thus ensuring code reuse.

In order to establish the software requirements, a set of microstructural evolution features common to several materials has been identified by analysing two models of microstructure evolution of two different materials: the AISI 1060 hypoeutectoid steel and the Titanium alloy Ti6Al4V. As a result, we get the design requirements of a computational tool with enough flexibility for representing a wide range of materials.

Microstructural transformations can be classified as diffusion and diffusionless transformations. For example, the formation of ferrite and pearlite in steels can be considered as diffusion controlled processes, whereas the formation of martensite is considered to be a diffusionless process. In steels, the bainite formation is usually referred to be a product of a diffusion-diffusionless process; however, in some works it is modelled as a diffusion controlled transformation [70].

A widely used model to describe diffusion controlled transformations in low alloy steels is the Kirkaldy's model [7, 3, 8, 9]. As it is mentioned in [10] this model is appealing because, once implemented, it can be applied to a wide range of low alloy steels just by providing a set of parameters, such as the chemical composition. A different approach is investigated in this work, because the restriction to low alloy steels limits the applicability of the current study. In this alternative, diffusion controlled transformations are described by means of the Johnson-Mehl-Avrami-Kolmogorov (JMAK) model [11, 12, 13], whereas diffusionless transformations, such as the formation of martensite, are modelled by the Koistinen-Marburger equation [4].

Firstly, a simple model for predicting microstructure evolution during cooling in the case of the hypoeutectoid steel AISI 1060, is analysed. Secondly, a model for describing the microstructure evolution of the Titanium alloy Ti6Al4V, is proposed and studied. This model is able to deal with the complexity associated to welding problems, which are characterised by arbitrary thermal histories featured by heating and cooling cycles where different critical temperatures may be reached. In the Titanium alloy, the heating stage is characterised by the formation of the β phase and the dissolution of other phases, identified as variants of the α phase. Some authors [71] assume that in welding problems this stage closely follows the equilibrium phase diagram; nevertheless, more complex dissolution models can be found, such as in [67] and [72]. In the cooling stage, the process is generally described as a continuous cooling process, where the β phase decomposes into the variants of the α phase. In the formulation of microstructural models, it is generally assumed that each transformation takes place during a characteristic temperature range in a sequential manner without overlapping [73, 74]; however, simultaneous transformations can also be modelled as shown further on.

This chapter is organised as follows. Section 3.2 gives a brief discussion on the different models used for describing transformation kinetics. In Section 3.3, the numerical treatment of the thermal-microstructural coupling is introduced. In Section 3.4, the computational description of the microstructure kinetics is studied in detail by introducing the notion of flowcharts. Then, Section 3.5 deals with the design and implementation of a general computational framework for modelling microstructure evolution. In Section 3.6, different tests are presented in order to assess the applicability of the proposed solution. Finally, conclusions are given in Section 3.7.

3.2 Modelling of transformations kinetics

Different models for predicting the microstructure evolution are introduced in this section. The general aspects of the materials to be studied are first presented. Then, models for describing diffusion and diffusionless transformations are described. In the general case, transformations are assumed to take place during a characteristic temperature range in a sequential manner without overlapping. An extension for simultaneous precipitation of phases is discussed in the case of diffusion controlled transformations.

3.2.1 Materials to be analysed

Two materials will be briefly described: the Titanium alloy Ti6Al4V and the AISI 1060 steel. They will be used as example of application of the software, and serve as a basis for specifying the software requirements.

The Ti6Al4V alloy is characterised by two solid phases: α and β [75]. When cooling from the solid phase β, the solid phase α begins to form below the β-transus temperature, $T_\beta = 1273\text{K}$. As temperature decreases, different α morphologies arise. The morphologies that involve diffusion controlled transformations during precipitation are the grain boundary α, α_g, and the Widmanstätten α, α_w. Also, a distinction between primary-α and secondary-α is usually made, referring to the formation of α during primary and secondary heat treatments, respectively [67]. Additionally, the Widmanstätten α has two forms referred to as intergranular basketweave α and α-colony. The formation of α martensite, α', and massive α, α_m, is associated to diffusionless transformations characteristic of rapid cooling: $\dot{T} < -410\text{Ks}^{-1}$ for α' and $-410\text{Ks}^{-1} \leq \dot{T} < -20\text{Ks}^{-1}$ for α_m.

The α morphologies to be modelled, associated to diffusion controlled transformations, are the grain boundary α_g and the Widmanstätten α_w. In the case of diffusionless transformations, both α martensite and massive α will be modelled as a single martensitic structure denoted for simplicity as α'. Arbitrary heating/cooling cycles will be considered.

In the case of the hypoeutectoid steel AISI 1060, we use the model proposed by Myoung-Gyu *et al.* [70] which is able to predict microstructure evolution only during cooling. The decomposition of austenite γ into the different structures is assumed to occur in a sequential manner without overlapping. It is also assumed that the bainite formation can be described as a diffusion controlled transformation. Therefore, the formations of α-ferrite, pearlite and bainite during cooling are considered to be diffusion controlled processes. In the temperature range $\text{Ae}_1 < T \leq \text{Ae}_3$, γ decomposes into α-ferrite; when $B_s < T \leq \text{Ae}_1$, it decomposes to pearlite; and when $M_s < T \leq B_s$, it decomposes into bainite. Below the martensite starting temperature M_s, martensite is formed as the result of a diffusionless transformation. In this case, Ae_3 denotes the temperature at which α-ferrite begins to precipitate, Ae_1 denotes the eutectoid temperature at which it is assumed

that pearlite begins to form, and B_s denotes the bainite starting temperature.

3.2.2 Diffusion controlled transformations: JMAK model

Diffusion controlled transformations are characterised by three stages: nucleation, growing and impingement [76]. Initially, during the nucleation stage, nuclei are formed and begin to grow. As they grow, the parent phase fraction decreases, and the number of nucleation sites for the formation of new nuclei reduces too. In what follows, a homogeneous model strategy for describing these phenomena is introduced.

Let $\alpha_1, \alpha_2, \cdots \alpha_n$ be the precipitating phases and let $y_1, y_2, ...y_n$ be the corresponding phase fractions. An accepted manner of describing the stages of nucleation and growth is by means of the law $y_i = b_i t^{m_i}$, where m_i and b_i are temperature dependent kinetic parameters [77]. The impingement stage is modelled by taking into account that the amount of the parent phase is given by $Y_i - y_i$, where Y_i is the maximum transformed fraction of phase α_i. Then, the rate equation describing a diffusion controlled process is given by

$$\begin{aligned} \dot{y}_i &= m_i b_i (Y_i - y_i) t^{m_i - 1}, \\ y_i(t_0) &= 0, \end{aligned} \tag{3.1}$$

where $(\dot{})$ denotes differentiation with respect to time. The JMAK model is then obtained by integrating the latter equation, giving:

$$y_i = Y_i (1 - e^{-b_i t^{m_i}}). \tag{3.2}$$

It follows that the time t is given by

$$t = \ln \left(\frac{Y_i}{Y_i - y_i} \right)^{\frac{1}{m_i}} b_i^{-\frac{1}{m_i}}. \tag{3.3}$$

By using this result in equation (3.1), the following autonomous equation is obtained

$$\begin{aligned} \dot{y}_i &= m_i b_i^{\frac{1}{m_i}} (Y_i - y_i) \ln \left(\frac{Y_i}{Y_i - y_i} \right)^{\frac{m_i - 1}{m_i}} \\ y_i(t_0) &= 0. \end{aligned} \tag{3.4}$$

The JMAK model given by equation (3.2) describes with good accuracy the kinetics of isothermal transformations. This model can be extended to non-isothermal transformations, such as those characterised by continuous cooling or heating, by assuming that a non-isothermal process can be described as a series of isothermal steps. This was first proposed by Scheil [78] and Cahn [79], giving rise to the so called Scheil-Cahn additivity rule, that in the words of Cahn is stated as follows: *"given an isothermal TTT (Time-Temperature-Transformation) curve for time τ, as a function of temperature, at which the reaction has reached a certain fraction of completion, x_0. Then, on continuous cooling, at that time t and temperature T, when the integral*

$$\int_0^t \frac{1}{\tau} ds \tag{3.5}$$

equals unity, the fraction completed will be x_0". In this work, the additivity rule is assumed to be valid for non-isothermal diffusive transformations.

A widespread method for the integration of equation (3.4) is the fictitious time method [80], in which the additivity rule is directly applied. For this reason, the method is also known as the "Additivity Rule Method". Every transformation undergoes an incubation time characteristic of the nucleation phase, subsequently followed by the growth stage. The anisothermal incubation time, that is the time delay to begin the formation of the considered phase, is estimated from isothermal data by applying the additivity rule (AR), equation (3.5). Let $\tau_s(T_n)$ be the incubation time for the isothermal case (datum that is extracted from a TTT diagram). Then, from the discrete AR, when

$$S = \sum_{i=1}^{n} \frac{\Delta t_i}{\tau_s(T_i)} \simeq 1, \tag{3.6}$$

the anisothermal incubation time is considered to be completed and the nuclei begin to grow. The kinetics of the growing phase is taken into account by computing a fictitious time τ^* from equation (3.2), which is given by

$$\tau_n^* = \ln\left(\frac{Y_{i,n}}{Y_{i,n} - y_{i,n-1}}\right)^{\frac{1}{m_{i,n}}} {b_{i,n}}^{-\frac{1}{m_{i,n}}}. \tag{3.7}$$

Then, to obtain the transformed fraction at time step t_n, equation (3.2) is evaluated at time $t = \Delta t_n + \tau_n^*$, that is

$$y_{i,n} = Y_{i,n}(1 - e^{-b_{i,n}(\Delta t_n + \tau_n^*)^{m_{i,n}}}). \tag{3.8}$$

In what follows, Y_i is considered to be given by the equilibrium fraction of phase α_i, and the additivity rule method will be adopted to describe non-isothermal diffusion controlled transformations.

3.2.2.1 Estimation of the JMAK parameters

The parameters m_i and b_i that control the kinetics of the transformation, equation (3.2), are generally determined from TTT diagrams, approach that is followed in this work as discussed below. We remark, however, that Geijsalers [77] proposed a method for computing these parameters from Continuous-Cooling-Transformation (CCT) diagrams.

The times t_s and t_f at which the transformation reaches $c_0\%$ and $c_1\%$ of completion can be extracted from TTT diagrams. Using this information in equation (3.2), and by considering that $Y_i = 1$ in the case of the isothermal TTT diagrams, the following relations arise

$$\ln\left(\frac{1}{1 - c_0}\right) = p_0 = b_i t_s^{m_i}, \tag{3.9}$$

and

$$\ln\left(\frac{1}{1 - c_1}\right) = p_1 = b_i t_f^{m_i}. \tag{3.10}$$

Then, these equations can be solved to get $m_i(T)$ and $b_i(T)$ as:

$$m_i(T) = \frac{\ln(p_0/p_1)}{\ln(t_s/t_f)}, \tag{3.11}$$

and

$$b_i(T) = \frac{p_1}{t_f^{m_i}}. \tag{3.12}$$

3.2.2.2 Inert phases behaviour

The JMAK model already introduced is modified to incorporate the inert behaviour of some phases during diffusive transformations. An example of this situation is the transformation $\beta \rightarrow \alpha_w + \alpha_g$ in the Ti6Al4V alloy, where the phase α' is inert. In this case, equation (3.8) is rewritten as

$$y_{i,n} = X_{i,n}(1 - e^{-b_{i,n}(\Delta t_n + \tau_n^*)^{m_{i,n}}}), \tag{3.13}$$

where the term $X_{i,n}$ is defined as

$$X_{i,n} = Y_{i,n} - \sum_{r \in R} y_{r,n-1},$$

and where R is the set of inert phases ($R = \{\alpha'\}$ in the case of the Ti6Al4V alloy).

3.2.2.3 Simultaneous precipitation of phases handling

The JMAK model does not consider the possibility of multi-phase transformations occurring simultaneously. Murgau *et al.* [72] proposed an extension to account for this situation. Consider the specific case of the Ti6Al4V alloy. Decomposition of the β phase into the two α morphologies α_w and α_g can be described simultaneously by taking into account that

$$y_{\alpha,n} = y_{\alpha_w,n} + y_{\alpha_g,n}. \tag{3.14}$$

By considering only the formation of α_g, from equation (3.13), we have

$$y_{\alpha_g,n} = X_{\alpha_g,n}(1 - e^{-b_{\alpha_g,n}(\Delta t_n + \tau_n^*)^{m_{\alpha_g,n}}}), \tag{3.15}$$

with

$$\tau_n^* = \ln\left(\frac{X_{\alpha_g,n}}{X_{\alpha_g,n} - y_{\alpha_g,n-1}}\right)^{\frac{1}{m_{\alpha_g,n}}} {b_{\alpha_g,n}}^{\frac{-1}{m_{\alpha_g,n}}}. \tag{3.16}$$

The equilibrium phase fraction of α_g, Y_{α_g}, which is needed to compute X_{α_g}, is not a generally available datum. Therefore, X_α is used instead, that is

$$y_{\alpha_w,n} + y_{\alpha_g,n} = X_{\alpha,n}(1 - e^{-b_{\alpha_g,n}(\Delta t_n + \tau_n^*)^{m_{\alpha_g,n}}}), \tag{3.17}$$

and by considering $y_{\alpha_w,n} = y_{\alpha_w,n-1}$, $y_{\alpha_g,n}$ is given by

$$y_{\alpha_g,n} = X_{\alpha,n}(1 - e^{-b_{\alpha_g,n}(\Delta t_n + \tau_n^*)^{m_{\alpha_g,n}}}) - y_{\alpha_w,n-1}, \tag{3.18}$$

with

$$\tau_n^* = \ln\left(\frac{X_{\alpha,n}}{X_{\alpha,n} - (y_{\alpha_g,n-1} + y_{\alpha_w,n-1})}\right)^{\frac{1}{m_{\alpha_g,n}}} {b_{\alpha_g,n}}^{-\frac{1}{m_{\alpha_g,n}}}. \tag{3.19}$$

After computing the phase fraction for α_g, a similar procedure is followed for α_w, that is

$$y_{\alpha_w,n} = X_{\alpha,n}(1 - e^{-b_{\alpha_w,n}(\Delta t_n + \tau_n^*)^{m_{\alpha_w,n}}}) - y_{\alpha_g,n-1}, \tag{3.20}$$

with

$$\tau_n^* = \ln\left(\frac{X_{\alpha,n}}{X_{\alpha,n} - (y_{\alpha_g,n-1} + y_{\alpha_w,n-1})}\right)^{\frac{1}{m_{\alpha_w,n}}} {b_{\alpha_w,n}}^{-\frac{1}{m_{\alpha_w,n}}}. \tag{3.21}$$

A second way of handling simultaneous formation of phases is analysed next. In the case of the Ti6Al4V alloy, it is experimentally observed that if martensite is present during heating, α' diffusively decomposes into $\beta + \alpha_w$ [81]. Therefore, it can be assumed that the phase fraction of martensite α' evolves according to

$$y_{\alpha',n} = e^{-b_{\alpha',n}(\Delta t_n + \tau_n^*)^{m_{\alpha',n}}} \equiv e^{f(\alpha')}, \tag{3.22}$$

where τ^* is similarly defined as in the previous cases, but particularly considering the kinetic parameters for α'. The amount of dissolved martensite $\Delta y_{\alpha',n}$ during the time increment Δt_n can be computed from this equation. Then, by assuming that the variation of phase α_g is negligible during this process, and that the amount of dissolved martensite contributes proportionally to the growth of phases α_w and β according to

$$\Delta y_{\beta,n} = \frac{1 - Y_{\alpha,n}}{1 - y_{\alpha_g,n-1}} \left(-\Delta y_{\alpha',n}\right), \tag{3.23}$$

$$\Delta y_{\alpha_w,n} = \frac{Y_{\alpha,n} - y_{\alpha_g,n-1}}{1 - y_{\alpha_g,n-1}} \left(-\Delta y_{\alpha',n}\right), \tag{3.24}$$

the reaction $\alpha' \rightarrow \beta + \alpha_w$ can be modelled as

$$y_{\beta,n} = \frac{1 - Y_{\alpha,n}}{1 - y_{\alpha_g,n-1}} \left(y_{\alpha',n-1} - e^{f(\alpha')}\right) + y_{\beta,n-1}, \tag{3.25}$$

$$y_{\alpha_w,n} = \frac{Y_{\alpha,n} - y_{\alpha_g,n-1}}{1 - y_{\alpha_g,n-1}} \left(y_{\alpha',n-1} - e^{f(\alpha')}\right) + y_{\alpha_w,n-1}, \tag{3.26}$$

$$y_{\alpha_g,n} = y_{\alpha_g,n-1}, \tag{3.27}$$

$$y_{\alpha',n} = e^{f(\alpha')}. \tag{3.28}$$

It should be noted that this procedure is different from that proposed by Murgau *et al.* [72]. They explicitly consider an expected martensite equilibrium, datum which is not usually available.

3.2.3 Diffusion controlled transformations: Kelly's model

In the case of the Ti6Al4V alloy, Kelly [67] modelled the dissolution of $\alpha_w + \alpha_g$ into β as a diffusion controlled transformation where the growing rate of β is assumed to be governed by a parabolic law. Details about this model are given in [67]. The model basically describes the growth of β according to

$$y_{\beta,n} = Y_{\beta,n} f_{p,n} (\Delta t + \tau_n^*)^{0.5}, \tag{3.29}$$

where, in this case, the fictitious time τ_n^* is given by

$$\tau_n^* = \left(\frac{y_{\beta,n-1}}{Y_{\beta,n} f_{p,n}}\right)^2, \tag{3.30}$$

and the parabolic thickening rate, $f_{p,n}$, is given by

$$f_{p,n} \equiv f_p(T_n) = 2.20 \times 10^{-31} T_n^{9.88821}. \tag{3.31}$$

3.2.4 Diffusionless transformations

Diffusionless transformations are characteristic of nucleation-controlled processes, where the nuclei growth occurs without diffusion and almost instantaneously [76]. The most important diffusionless transformation is the martensite transformation; therefore, the discussion is focused on the formation of this meta-stable phase. This transformation is mainly thermally-driven. However, it is observed in many situations that the stress state is the primary factor influencing its formation [3]. This influence is not considered here, although it is recognised that the stress state determines the formation of martensite in some situations, such as in Shape Memory Alloys (SMAs) [82].

The martensite transformation is characterised by two critical temperatures, the martensite start temperature M_s and the martensite end temperature M_f. In some models, the end of the martensite transformation is predicted in terms of the cooling speed instead of the critical temperature M_f. Alternatively, a combination of both can be used as a means of predicting the end of the transformation.

The kinetics of martensite formation is generally described by the Koistinen-Marburger (KM) equation [4], that is given by

$$y_{m,n} = y_0^{M_s} \left(1 - e^{r(T_n - M_s)}\right), \tag{3.32}$$

where $y_0^{M_s}$ is the amount of retained parent phase at the time when the temperature M_s is crossed, and the parameter r is a constant specific of the material under consideration. The usual value of this constant for steels is 0.011. In the case of the Ti6Al4V alloy, r is usually taken to be 0.003 [83, 84], although in the recent work of Murgau *et al.* [72] a value of $r = 0.005$ is used instead.

There are some situations where martensite forms under rapid cooling, but diffusion controlled transformations take place as well. In order to avoid a complete martensite transformation, Murgau *et al.* proposed to modify the KM model to

$$y_{m,n} = \left(y_0^{M_s} - Y_{\beta,n}\right)\left(1 - e^{r(T_n - M_s)}\right). \tag{3.33}$$

3.2.5 Instantaneous dissolution models and equilibrium phase diagrams

It is frequently found that during heating, an instantaneous dissolution model is used in welding problems for describing the behaviour of martensite. In this kind of model, when the dissolution temperature T_{dm} is reached, martensite instantaneously dissolves completely contributing to other phases. It must be specified how this contribution is done to each phase. We assume that martensite contributes to the other phases proportionally to the equilibrium of each phase at the current temperature. For example, in the case of the Ti6Al4V alloy, when T_{dm} is reached martensite α' decomposes completely into α_w and β according to

$$y_{\alpha_w,n} = y_{\alpha',n-1}(Y_{\alpha,n} - y_{\alpha_g,n-1}) + y_{\alpha_w,n-1}, \tag{3.34}$$

$$y_{\beta,n} = y_{\alpha',n-1} Y_{\beta,n} + y_{\beta,n-1}, \tag{3.35}$$

where it was additionally assumed that $y_{\alpha_g,n} = y_{\alpha_g,n-1}$,

If martensite is not present during heating, some works [71, 3, 17] assume that the transformation closely follows the equilibrium phase diagram, and the microstructure evolution is computed accordingly.

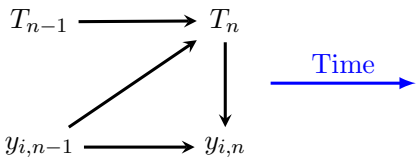


Figure 3.1: Thermal-microstructural staggered approach.

3.3 Thermal-microstructural coupling

The thermal problem is solved using a finite element discretisation, with the solid/liquid phase change tetrahedral element developed in [28] extended for taking into account solid state transformations. The phase fractions characterising microstructure are tracked at the element nodes as internal state variables. Phase fractions in other locations, for example at a Gauss point, are interpolated as usual by means of the expression

$$y_i = \boldsymbol{N}^T \boldsymbol{y}_i, \tag{3.36}$$

where $\boldsymbol{N}$ denotes the linear tetrahedral shape functions and $\boldsymbol{y}_i$ are the parameters describing the fraction of phase α_i at the element nodes. The temperature field T^h is interpolated accordingly as

$$T^h(\boldsymbol{x}, t_n) = \boldsymbol{N}^T \boldsymbol{T}_n, \tag{3.37}$$

where $\boldsymbol{T}_n$ are the nodal parameters describing the temperature field at time t_n.

A staggered approach is adopted for the thermal-microstructural coupling as depicted in Figure 3.1 [2]. The temperature field at time t_n is used for computing the phase fractions at the current time t_n. Phase fractions at the previous time step t_{n-1} are used for computing the thermal field at time t_n.

The microstructure coupling is taken into account in the thermal problem by means of two contributions. On one hand, we have the contribution of the latent heat terms associated to the solid/solid phase changes, and on the other hand we have the variation of thermophysical properties in terms of microstructure.

In order to consider the contribution of the latent heat terms associated to solid/solid phase transformations, the formulation given in [85, 28], dealing with the problem of solid/liquid phase change, is modified. A term S_s is added to the heat balance equation, which is given by

$$S_s = \rho \sum_{i=1}^{n_h} \mathcal{L}_i \dot{y}_i, \tag{3.38}$$

where n_h is the number of phases, $\mathcal{L}_i$ is the latent heat associated to phase α_i and ρ is the density. Then, if the residual of the finite element formulation considering only solid/liquid phase change is denoted by $\boldsymbol{\Pi}_n \equiv \boldsymbol{\Pi}(\boldsymbol{T}_n, t_n)$, the extension to solid/solid transformations reads

$$\boldsymbol{\Xi}_n \equiv \boldsymbol{\Xi}(\boldsymbol{T}_n, t_n) = \boldsymbol{\Pi}_n + \rho_n \boldsymbol{M} \sum_{i=1}^{n_h} \mathcal{L}_i \frac{\boldsymbol{y}_{i,n} - \boldsymbol{y}_{i,n-1}}{\Delta t}, \tag{3.39}$$

where $\boldsymbol{M}$ is the (unit density) mass matrix and $\rho_n \equiv \rho(\boldsymbol{T}_n, t_n)$.

Every material property depends on temperature and microstructure. That is, a given material property $\Psi(T, y_i)$, can be considered to vary in terms of the current temperature T and of the phase fractions y_i. If the variation of the material property in terms of temperature is known for each phase, the microstructural dependence of material properties

can be considered by a mixture rule. For instance, if a linear mixture rule is used, the macroscopic material property Ψ is given by

$$\Psi(T, y_i) = \sum_{i=1}^{n_h} y_i \Psi_i(T), \tag{3.40}$$

where y_i are the phase fractions, T the temperature and $\Psi_i(T)$ the function describing how the material property varies with respect to temperature for each phase α_i.

3.4 Computational description of microstructure kinetics

The objective of this work is to develop a general enough computational framework to handle the kinetics of a wide range of materials, thus reducing the need of recoding for the representation of microstructure evolution of new materials.

Flowcharts are of common use in computer science for specifying an algorithm or set of steps to solve a given problem [69]. In particular, the algorithm for computing the microstructure evolution of a given material can be specified in terms of a flowchart. In this work, we extend the use of flowcharts to fully define a material. Specifically, we propose to define the material behaviour as usual, through a set of material properties: the type of thermal response (isotropic, orthotropic, etc.), and the type of mechanical response (elastic, elasto-plastic, etc.); but additionally specifying as part of the dataset a flowchart aimed at describing the microstructure kinetics.

Every component of the flowchart is coded just once, and eventually reused when defining a new material. Two types of boxes are used in the material flowcharts: decisions, denoted by diamonds, and processing steps, denoted by rectangular boxes. Generally, each processing step will be related to a type of transformation. When drawing the material flowcharts, a set of labels denoted as LID and TID will be used. Its basic use is for component identification, and to facilitate the description of the flowchart in the program commands language. A detailed explanation will be given in Section 3.5, when describing the computer implementation.

In the rest of this section, arbitrary thermal histories handling and the kinetics of the materials under study are analysed from the point of view of flowcharts. Then, in Section 3.5, the design and implementation of the software for handling thermal-microstructural simulations is undertaken.

3.4.1 Arbitrary thermal histories handling

Microstructural models are generally designed for predicting either the dissolution or the formation of a parent phase (e.g., β for the Ti6Al4V alloy and γ for hypoeutectoid steels). In the case of monotonically cooling processes, dissolution models are applied, whereas in the case of monotonically heating processes, formation models are used.

When dealing with arbitrary thermal histories, a logic is needed to decide what kind of model should be applied at each time instant. This decision is taken by considering the equilibrium phase fractions at the current temperature. An example, specifically applied to the Ti6Al4V alloy, can be observed in Figure 3.2. The procedure **Start** is used to denote the start of the algorithm for computing the microstructure evolution. If equilibrium is reached (decision identified with the inscription TID 6), the system attains steady state

and nothing must be done. If this is not the case, solid/solid phase transformations can take place. Then, β decomposes if $y_{\alpha_w,n-1} < Y_{\alpha,n} - y_{\alpha_g,n-1} - y_{\alpha',n-1}$, and viceversa in the other case, conditional which is part of the processing step TID 7 as explained later.

This logic differs from other proposals in the literature [67, 72, 84], because in those works the decision about decomposition or formation of the parent phase is taken by comparing the amount of the parent phase with the equilibrium phase fraction at the current temperature. A different alternative is followed in this work, in order to include the effect of the formation of the meta-stable phase α' (martensite). We assume that the total amount of α is given by $y_\alpha = y_{\alpha_w} + y_{\alpha_g} + y_{\alpha'}$, and therefore, β decomposes if the amount of phase α_w at step t_{n-1} is less than the equilibrium phase fraction Y_α at time t_n minus the amount of phases α_g and α' at time t_{n-1} (see processing step TID 7).

It must be additionally noted that the proposed logic is able to handle the case posed by solid/liquid phase change, situation characteristic of temperatures above the β-transus temperature for the case of the Ti6Al4V alloy (the conditional $T_n > T_\beta$ associated to processing step TID 2). The solid/liquid phase change is modelled using the formulations presented in [85, 28].

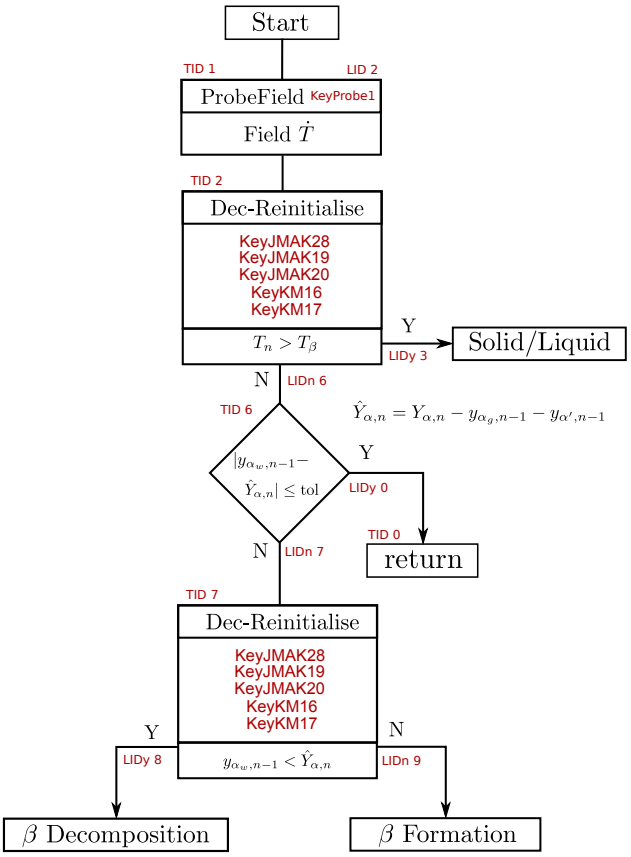


Figure 3.2: Logic for handling arbitrary thermal histories (Ti6Al4V).

A distinctive feature of arbitrary thermal histories characterised by cyclic cooling and heating, is the fact of having multiple cycles of decomposition and formation of the parent phase. In order to correctly model this behaviour, multiple transitions from decomposition models to formation models and viceversa, must be accomplished. We make the assumption that these transitions introduce the additional requirement of re-initialising certain state variables each time that a new decomposition or formation cycle starts. Examples

of these state variables are given by the Scheil sum (see equation (3.6)) for computing the incubation times of JMAK models, and by the amount of retained parent phase $y_0^{M_s}$ at the time of crossing the temperature M_s in the case of KM models.

In the case of the Ti6Al4V alloy, the re-initialisation of the state variables is controlled by means of the processing steps **Dec-Reinitialise** TID 7 and TID 2. In these procedures, the state variables to be re-initialised are listed in the first space of the box representing them, whereas in the second space the condition for controlling the re-initialisation process is specified. **Dec-Reinitialise** processing boxes can return either by branches **Y** or **N**, depending on the involved decision. The specified variables are re-initialised each time that the result of the condition at the current step of the simulation differs from the result of the condition at the previous time step. That is, when it is detected that the information flow changes of branch when compared to the branch followed at the previous time step, the variables are re-initialised. Note that at the beginning of the simulation the branch followed at the previous time step is undefined; therefore, an initial value must be specified. In the case of processing steps TID 7 and TID 2, the initial value is set to branch **N**.

Let us consider e.g. the case of a JMAK model describing the diffusive transformation $\beta \rightarrow \alpha$, whose Scheil sum is identified by means of the "Key" KeyJMAK19. Additionally, suppose that $T_n < T_\beta$ and that variables are not re-initialised by processing box TID 2. Then, if the transformation $\beta \rightarrow \alpha$ takes place, the **Y** branch of processing box TID 7 is followed. The first time the information flow exits by the **Y** branch (the branch value was initialised to **N**), the variable identified by KeyJMAK19 is re-initialised. The re-initialisation of this variable is not performed again until procedure TID 7 detects that the information flow changes of branch. In the current case, re-initialisation takes place again when the branch **Y** changes to the branch **N**, that is when β stops decomposing and begins to form. Note that in the re-initialisation process all what matters is the detection of branch changes. Details about the re-initialisation process for each of the state variables specified in TID 7 and TID 2 are given in what follows and in subsequent sections.

3.4.2 Steel AISI 1060

A simple model for predicting microstructure evolution of AISI 1060 steel during strict cooling is developed (note that since strict cooling is analysed, only the decomposition of austenite γ is studied). Additionally, the latent heat associated to solid state transformations, and temperature and microstructure dependent thermophysical properties, are taken into account. It is assumed that no solid/liquid transformation occurs, and that the microstructure of the body under analysis is 100 % austenite at the beginning of the simulation.

3.4.2.1 Decomposition of γ

As it was mentioned in Section 3.2.1, we assume that the decomposition of γ into product phases takes place during a characteristic temperature range in a sequential manner without overlapping. The flowchart for this behaviour is given in Figure 3.3. The name of the phase being updated is given inside each processing box. The JMAK processing steps implement the Johnson-Mehl-Avrami-Kolmogorov model for diffusion controlled transformations; the KM processing step implements the Koistinen-Marburger model for predicting the amount of martensite; and the PhaseUpdate procedure computes the retained austenite as $y_\gamma = 1 - y_\alpha - y_P - y_B - y_M$, where the subscripts α, P, B and M

denote α-ferrite, pearlite, bainite and martensite, respectively.

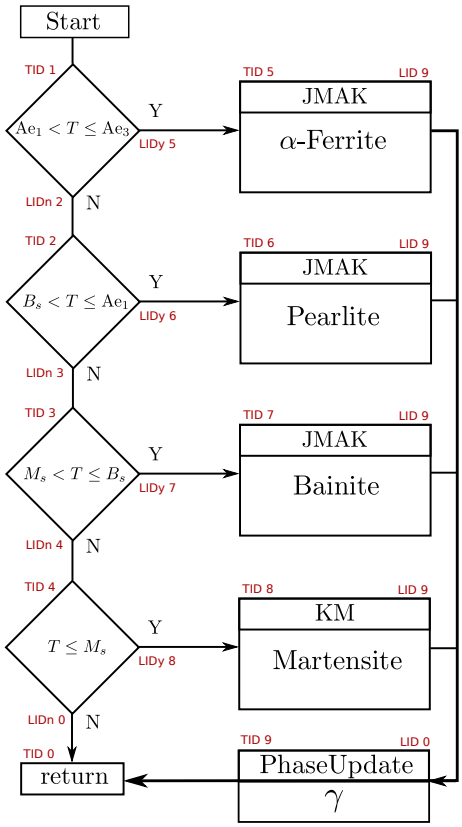


Figure 3.3: Flowchart for the description of the material AISI 1060 during cooling.

The kinetic parameters of the JMAK models were computed according to the procedure presented in Section 3.2.2.1 and by making use of the TTT diagram [86] specified in Figure 3.4. The obtained results can be observed in Figure 3.5. The critical temperatures were assumed to be $Ae_3 = 1023$K, $Ae_1 = 996$K, $B_s = 813$K and $M_s = 544$K. The equilibrium phase fractions were taken as

$$Y_\alpha = 1 + \frac{11407}{83(T - 1183)}, \tag{3.41}$$

$$Y_P = 0.735, \tag{3.42}$$

$$Y_B = 1, \tag{3.43}$$

where T is given in K. Here, Y_α and Y_P were computed as in the work of Geijsalers [77], and Y_B was assumed equal to 1. The exponent used in the Koistinen-Marburger model was taken equal to 0.011. It is assumed that martensite begins to form when $T \leq M_s$ and no temperature of end of transformation M_f is considered in this case.

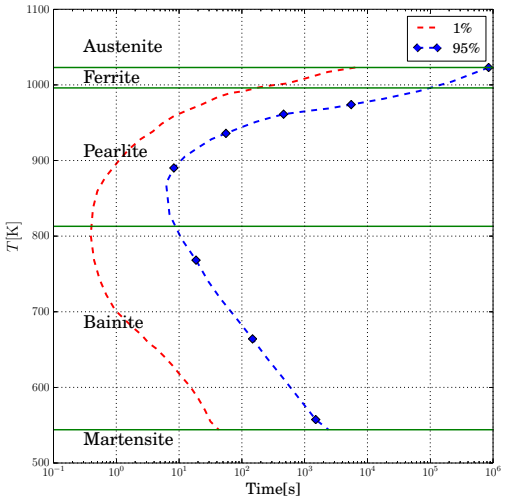


Figure 3.4: TTT diagram for the AISI 1060 steel [86].

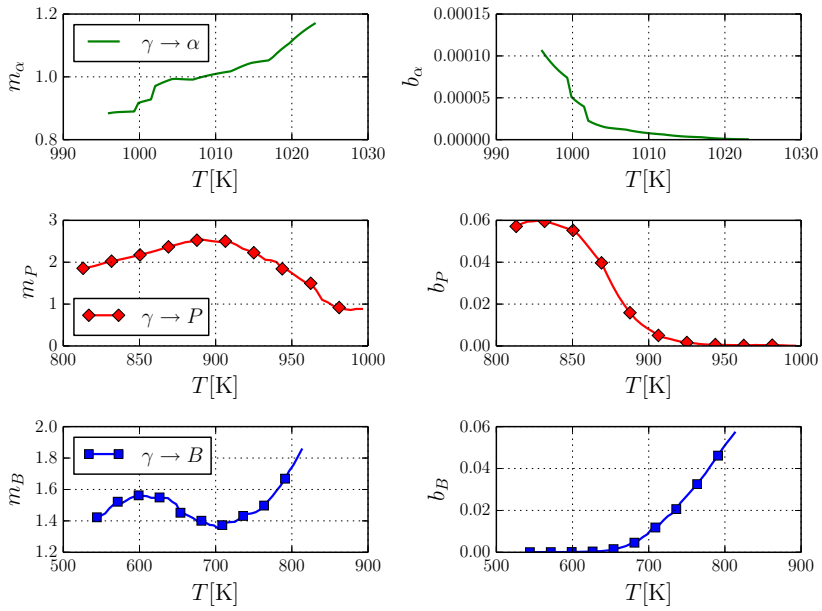


Figure 3.5: JMAK parameters for the AISI 1060 steel.

3.4.3 Titanium alloy Ti6Al4V

A microstructural model for the Ti6Al4V alloy is developed. This model is able to handle arbitrary thermal histories, including melting and solidification. The overall logic for dealing with arbitrary thermal histories is given in Figure 3.2. In what follows models for the decomposition and formation of β solid phase, taking into consideration the existence of a liquid phase, are proposed. This model is much more complex than the one presented previously. However, the latent heat associated to solid state transformations and the microstructure dependence of thermophysical properties are not considered, due to the lack of experimental data. Material data were extracted from the works of Kelly [67],

Murgau *et al.* [72] and Anca *et al.* [18].

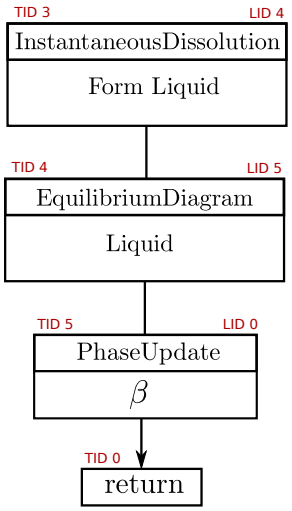


Figure 3.6: Flowchart for the solid/liquid logic.

The logic for handling the solid/liquid phase change is shown in Figure 3.6. Note that for temperatures greater than T_β, only β and possibly liquid can exist. For this reason, the processing step InstantaneousDissolution, TID 3, is used first to zero the fractions of all phases excepting the liquid phase. Then, the liquid fraction, f_l, is computed according to the equilibrium phase diagram, TID 4, whose expression is given by

$$f_l(T) = \begin{cases} 1 & \text{if } T > T_{liq}, \\ \dfrac{T - T_{sol}}{T_{liq} - T_{sol}} & \text{if } T_{sol} \leq T \leq T_{liq}, \\ 0 & \text{if } T < T_{sol}, \end{cases} \tag{3.44}$$

where $T_{sol} = 1877$K and $T_{liq} = 1933$K are the solidus and liquidus temperatures, respectively. Then, the amount of β phase is computed in the processing step PhaseUpdate, TID 5, according to $y_\beta = 1 - f_l$.

3.4.3.1 Decomposition of β

The flowchart characterising the decomposition of the β phase is depicted in Figure 3.7. The liquid phase vanish completely during solid state transformations, that is why the liquid phase is dissolved to zero through the processing step InstantaneousDissolution, TID 8.

The formation of α_g and α_w is described according to the kinetics of diffusion controlled transformations using specific JMAK models, where simultaneous formation of phases can take place. The parameters of the JMAK model are derived from the TTT diagram observed in Figure 3.8a by applying the procedure described in Section 3.2.2.1. The obtained parameters can be observed in Figure 3.9. The equilibrium phase fractions for α and β are shown in Figure 3.8b. It is assumed that α_w and α_g begin to form in the temperature range $M_s < T \leq T_\beta$ only if $y_{\alpha_w,n-1} + y_{\alpha_g,n-1} + y_{\alpha',n-1} < Y_{\alpha,n}$ (decision boxes TID 10 and 13). In this case T_β and M_s are taken equal to 1273K and 848K, respectively.

The diffusionless martensite transformation during cooling is modelled according to the Koistinen-Marburger (KM) model, with $r = 0.005$. This transformation is considered

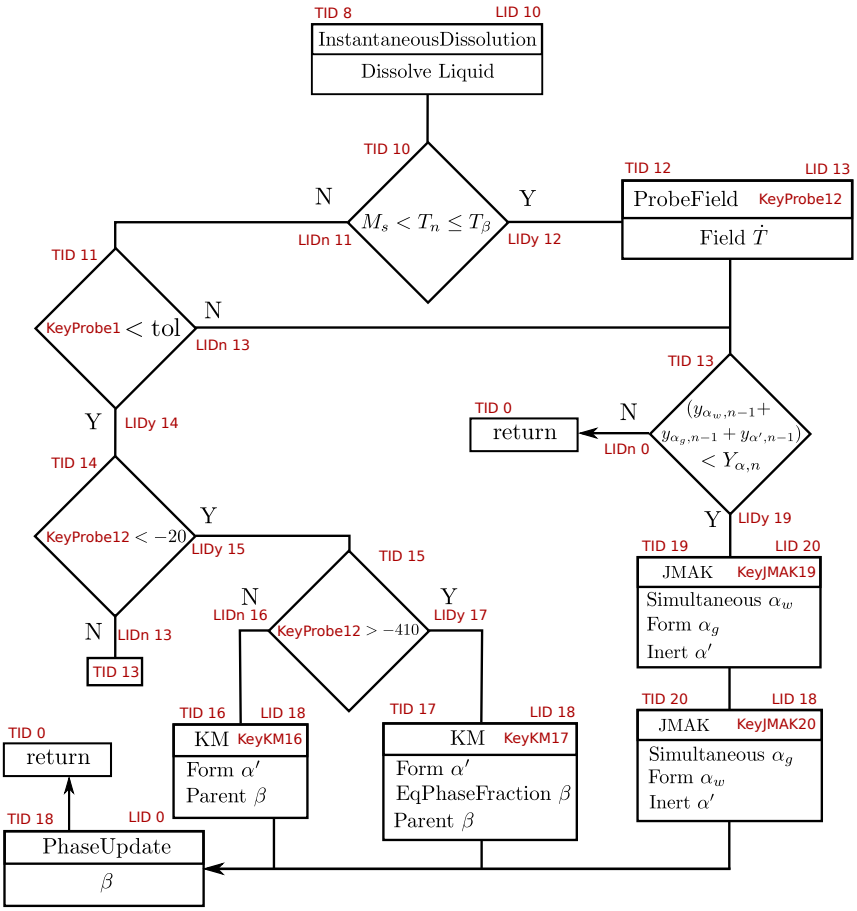
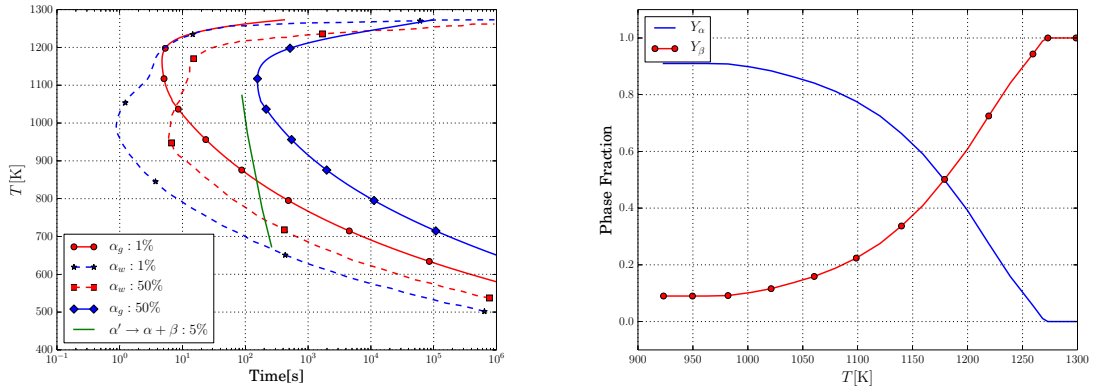


Figure 3.7: Flowchart description of the decomposition of β in the Ti6Al4V alloy.



(a) TTT diagram for the Ti6Al4V alloy.

(b) Equilibrium phase fractions for α and β.

Figure 3.8: Data for Ti6Al4V alloy.

to start at the critical temperature M_s. Full martensitic transformations are observed experimentally for temperature rates higher than 410Ks^{-1} [87, 72, 67], but for temperature rates in the range $-410\text{Ks}^{-1} < \dot{T} \leq -20\text{Ks}^{-1}$ the possibility of developing diffusion controlled transformations must be considered. In the former case, the KM model, equation

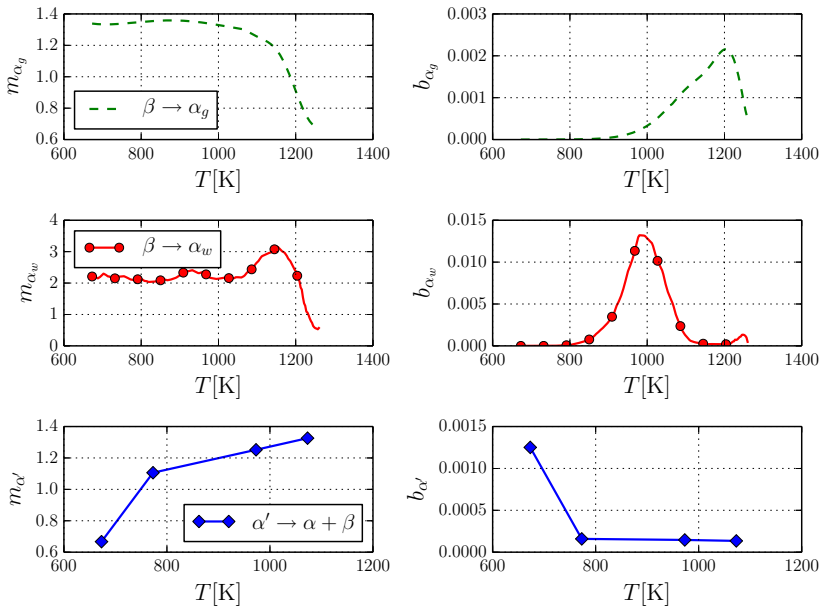


Figure 3.9: JMAK parameters for the Ti6Al4V alloy.

(3.32), is used (processing step KM, TID 16) while in the latter case, the modified KM model given by equation (3.33), is used (processing step KM, TID 17). The cooling rate is computed in the temperature range $M_s < T \leq T_\beta$ [67], by means of the processing step ProbeField, TID 12. This processing step computes and saves the specified field, in this case $\dot{T}$, in a variable identified by a Key, in this case KeyProbe12 (more details are given in Section 3.5). This KeyProbe12 is used by decision boxes TID 14 and TID 15 to decide whether martensite will form or not. Finally, the amount of retained β is computed by the processing step PhaseUpdate, TID 18.

The possibility of developing diffusion controlled transformations $\beta \rightarrow \alpha_w + \alpha_g$ for temperatures $T < M_s$ is also considered. Two different cases can be identified:

- The first case is that in which the cooling velocity at the time of crossing the temperature M_s is high enough to initiate the martensite transformation. As martensite forms, no diffusive transformation takes place. In this case, diffusion controlled transformations can start when the martensite transformation ends. This event is detected by means of decision TID 11, when the cooling velocity is lower than a given threshold denoted by `tol` (this tolerance is taken equal to -1Ks^{-1} for the current model). The cooling velocity in decision TID 11 is measured in the processing step ProbeField TID 1, and it is saved in the variable identified by the Key KeyProbe1 (see Figure 3.2).

- The second case can happen when the cooling velocity at the moment of crossing temperature M_s, is not enough to generate martensite, in which case it is assumed that β decomposes diffusively in $\alpha_w + \alpha_g$ (branch **N** of decision TID 14).

3.4.3.2 Formation of β

The flowchart for the formation of the β phase is shown in Figure 3.10. This process is partly characterised by diffusion controlled transformations. If α' is present (decision

box, TID 9) and $T_{d0} < T_n < T_{dm}$ (decision boxes, TIDs 22 and 27), it is observed that martensite begins to dissolve into $\alpha_w + \beta$ by a diffusive mechanism. The JMAK model given by equations (3.25-3.28) is employed to model the transformation $\alpha' \rightarrow \alpha_w + \beta$ (JMAK processing step TID 28). The corresponding JMAK kinetic parameters, which were taken from the work of Gil Mur *et al.* [81], can be observed in Figure 3.9. The function describing the incubation time for the isothermal case is shown in Figure 3.8a. The critical temperatures T_{dm} and T_{d0} were taken equal to 1073 K and 673 K, respectively.

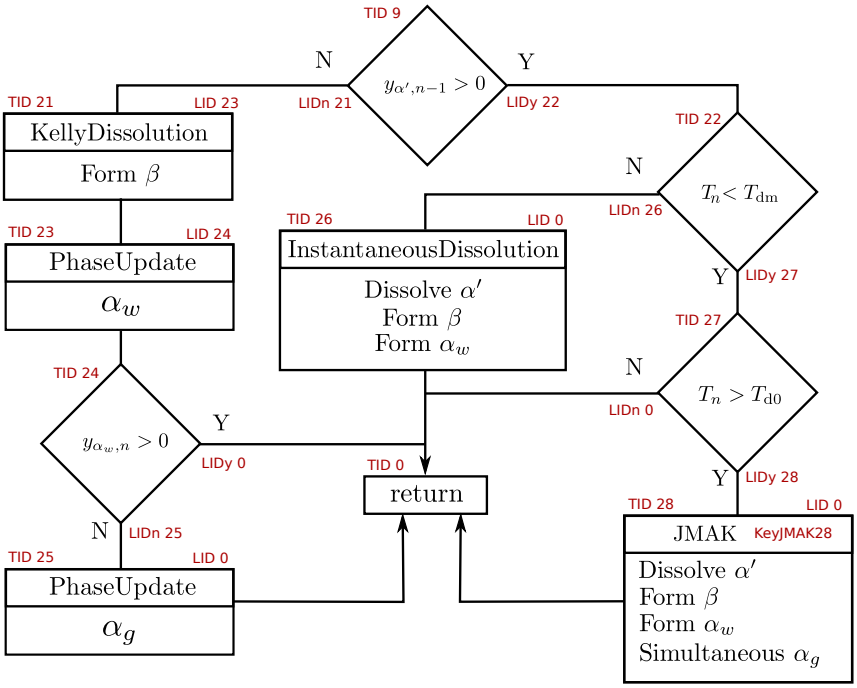


Figure 3.10: Flowchart for the description of the material Ti6Al4V for the formation of β.

The critical temperature T_{dm} is taken as dissolution temperature for martensite, meaning that after this temperature is exceeded, α' dissolves completely, i.e. $y_{\alpha'} = 0$, and the phase fractions of α_w and β are computed according to equations (3.34) and (3.35), respectively. This is reflected in the flowchart by the processing step InstantaneousDissolution TID 26.

If the phase fraction of α' is zero, α_w and α_g dissolve into β. The Kelly model, given by equation (3.29), is used to describe the reaction of $\alpha_w + \alpha_g \rightarrow \beta$ (processing step KellyDissolution TID 21). It is assumed that phase α_w dissolves first (processing step PhaseUpdate, TID 23), and when it vanishes (decision box TID 24), α_g begins to dissolve (processing step PhaseUpdate, TID 25).

3.4.3.3 Re-initialisation of state variables

The state variables of the JMAK models TIDs 19, 20 and 28, and of the KM models TIDs 16 and 17: Keys KeyJMAK19, KeyJMAK20, KeyJMAK28, KeyKM16 and KeyKM17, respectively, are re-initialised by procedures TIDs 7 and 2 (Figure 3.2). Processing step TID 7 re-initialises its state variables by detecting the transition between formation and decomposition of the parent phase β. Processing step TID 2 is another Dec-Reinitialise procedure, which handles variable re-initialisation in the extreme case where the temperature instantaneously jumps to $T_n > T_\beta$ while the β phase was decomposing. Take into

account that in this case, processing step TID 7 would not re-initialise the state variables when initiating the next decomposition cycle because the preceding formation cycle was instantaneous and followed the branch **Y** of processing step TID 2 without giving any notice to procedure TID 7 of its occurrence.

Remark: it should be noted that α_w and α_g are allowed to form below the temperature M_s and after the martensite transformation finishes. Take into account that the reaction $\beta \rightarrow \alpha_w + \alpha_g$ can take place before the $\beta \rightarrow \alpha'$ transformation. Nevertheless, once α' stops forming and α_w and α_g begin to precipitate again, the state variables related to the JMAK models describing the diffusive transformation $\beta \rightarrow \alpha_w + \alpha_g$ (see Figure 3.7) are not re-initialised during the same decomposition cycle. In the current computational framework, it is quite easy to incorporate the behaviour of re-initialising the required state variables during the same decomposition cycle. Just replace decision step TID 14, Figure 3.7, by a Dec-Reinitialise processing step specifying as state variables the Keys KeyJMAK19 and KeyJMAK20, and as condition the decision KeyProbe12 < -20.

3.5 Computer implementation

The design of a software able to handle a wide range of materials with different microstructural behaviour, is now presented. Two different materials were analysed in previous sections: a model of an hypoeutectoid steel for describing the microstructure evolution under cooling conditions, and a more complex model of a Titanium alloy involving arbitrary thermal histories. From these studies, a set of common features can be extracted and used as requirements to be met by the computational tool.

The fundamental idea is to specify the microstructure kinetics through flowcharts. It can be observed from the flowcharts given in Figures 3.2, 3.3, 3.6, 3.7 and 3.10, that two types of steps are necessary to model the materials behaviour: decision boxes and processing boxes.

Let F be a scalar field. Then, the different types of decision boxes are given by

- Interval decision box, where the value of field F is checked to lie within the interval $F_0 < F \leq F_1$ for arbitrary field values F_0 and F_1.
- Left decision box, where $F_0 < F$ is checked.
- Right decision box, where $F < F_1$ is checked.
- Right-tolerance decision box, where $|F - F_1| <$ tol is checked.
- Equality decision box, where $F == F_1$ is checked.

An Object Oriented (OO) philosophy is adopted for the design and implementation of the proposed computational framework. In this context, the involved physical fields as well as the internal variables, are identified by means of an instance of the class `Key`. It can be observed in the flowcharts shown in Figures 3.7 and 3.10, that the JMAK and KM models are associated to a Key. These Keys are used to identify which internal state variables have to be re-initialised. Therefore, an additional requirement is made evident: decision boxes must be able to handle the setting of states identified by specified Keys.

The different processing steps needed to build the described flowcharts are given by

- JMAK processing step: it is used for predicting the kinetics of diffusion controlled transformations.

- Instantaneous Dissolution processing step: it is used for describing the instantaneous and complete dissolution of a given phase or set of phases.

- Equilibrium Diagram processing step: it is used to describe microstructure evolution according to the equilibrium phase diagram.

- Koistinen-Marburger processing step: it is used to describe the instantaneous formation of phases related to diffusionless transformations.

- PhaseUpdate processing step: it is used to update a given phase fraction y_k by means of the law $y_k = 1 - \sum_{i \in I \setminus y_k}^{n_h} y_i$, where n_h are the number of phases and I is the set of indices identifying the different phase fractions.

- KellyDissolution processing step: it is used to describe diffusion controlled transformations, adopting a parabolic law for the growing of the phase under analysis.

- ProbeField processing step: it is used to sample a given field and save it into a variable identified by a specific Key (details are given below).

- Dec-Reinitialise processing step: it is used for re-initialising state variables.

- Return processing step: it is just used to denote the end of the current step.

These "processing steps" and "decision boxes" are coded only once, and reused at the different instances in which they appear in a flowchart. In this way, different material behaviours can be modelled by assembling the different flowchart components as required by the kinetics of the material under study. In order to do that, the different processing steps must be specified and configured with the phenomenological parameters corresponding to the analysed material. For example, when using a JMAK processing step, the Avrami parameters b_i and m_i, and the equilibrium phase fractions must all be given. Also, a way of describing the information flow between the different components of the flowchart must be given. For this purpose, each component of the flowchart is identified by a unique identifier (ID) denoted in the figures of the introduced flowcharts as TID (coming from Type ID). The links of the flowchart are specified by Link IDs and denoted in the flowcharts by LID.

We remark that additional processing steps can be coded if required, to enhance the range of application of the software. For example, another processing step is needed to simulate carbides precipitation, such as the precipitation of $Cr_{23}C_6$ in austenitic stainless steels [88]. Another feature usually considered is the grain size prediction, in which case a new state variable as well as another processing step must both be considered.

Although not introduced yet, there is an *initialisation step* where initial values must be given for the state variables. Generally, the microstructure evolution prediction will be intimately related to a Finite Element code used for thermal or thermo-mechanical calculations. If this is the case, the initialisation step will be carried out by the FEM solver. Each of the state variables that describe the microstructure evolution are taken as internal to the elements. These state variables can be either defined point-wise at each Gauss point, or in association to each thermal node. As already mentioned in Section 3.3, the last option is adopted in this work.

3.5.1 Solution design

A design based on the Object Oriented modelling (OOM) is presented. We assume that the associated Finite Element code is written using the Object Oriented (OO) philosophy, and that the microstructural model to be implemented will be part of this code. The design presented here follows the ideas used in the Finite Element program OOFELIE (Object Orient Finite Elements Led by Interactive Executor), which was described by Cardona *et al.* in [39].

From a global perspective, the proposed design is given in Figure 3.11. General FEM solvers are implemented based on algebraic structures characteristic of FEM discretisations, without considering any specific element. For example, in a nonlinear transient thermal problem discretised by FEM, which uses a Backward-Euler scheme for time integration and a Newton-Raphson scheme as non-linear solver, the type of element used for computing the associated tangent matrix and residual vector is independent of the operations done at a global level. For this reason, solvers are implemented independently of the element type. Time stepping in the code is controlled from the solver by means of the `setStep` method.

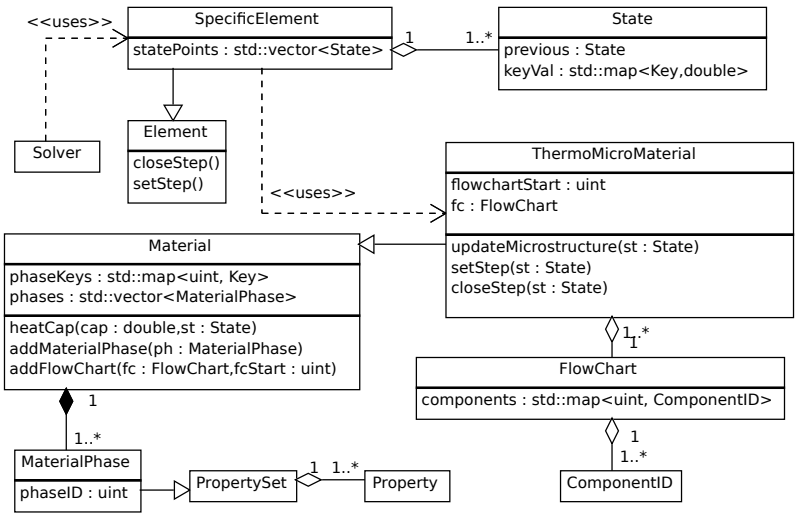


Figure 3.11: FEM global perspective of the proposed design.

Each specific element (for example, a thermal tetrahedral element) inherits from the mother class `Element` which defines the element main features from the FEM point of view. State variables, and in general any historical variable, are stored in instances of the class `State`. This class is specifically designed to track values of the state variables at the current and previous time steps. For example, scalar fields identified by a given `Key` are tracked through the `map<Key, double>` member `keyVal`.

Microstructure evolution is modelled as a response of the material to the current state of the system, given by T and $\dot{T}$. In the current computational framework, a given material response is taken into account through specific material types. Specifically, the material type `ThermoMicroMaterial` is responsible for computing the microstructure evolution. As it was mentioned in Section 3.3, a staggered approach is used, and because of this, the microstructure at time t_n is computed when the temperature field at that time is known. This logic is implemented by means of the following calling se-

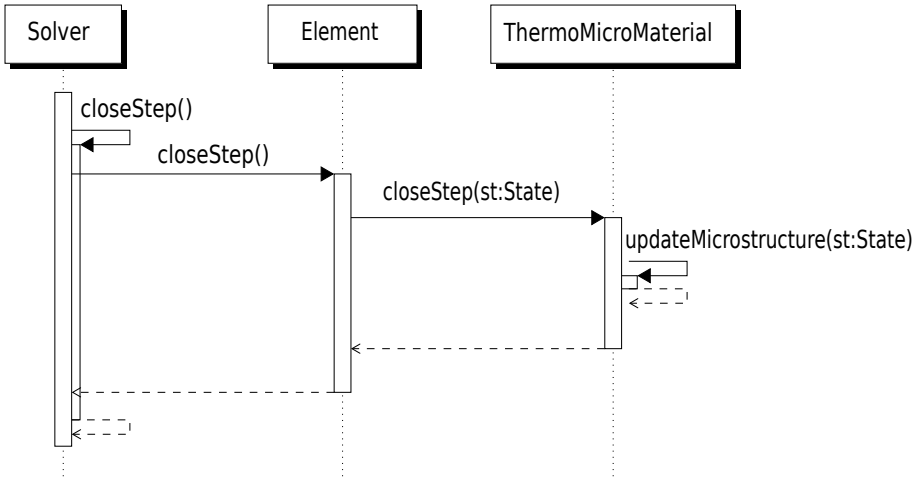


Figure 3.12: Calling sequence diagram for the computation of microstructure evolution.

quence. The start and the end of each time step is controlled from the solver, by making use of the methods `Solver::setStep()` and `Solver::closeStep()`. The calls of these methods are followed by the calls to the element functions `Element::setStep()` and `Element::closeStep()`, respectively, allowing to update the element states at a given instant. Because of the adopted staggered approach, the function `Element::closeStep()` is used to trigger the computation of phase fractions characterising the microstructure of the material at the current time step, after ending the thermal step solution. The `closeStep` function at the element level, loops through the element nodes calling to the `ThermoMicroMaterial::updateMicrostructure` routine (details are given below) that implements the model describing the microstructure evolution. This situation is summarised in the calling sequence diagram given by Figure 3.12.

The material type `ThermoMicroMaterial` must be able not only to compute the microstructure evolution from the material state, but to specify the material properties at a given point. It must be considered that the material properties can depend on the current state (in this case, temperature and microstructure) at the point where the property is computed. The `Material` class is intended to define the interfaces of methods used for computing material properties, such as the heat capacity. The associated function `Material::heatCap` is implemented in the `ThermoMicroMaterial` class and the macroscopic property computed according to the implemented mixture rule (in this work, linear mixture rules are used). In the following, we describe how to access the information needed for making these computations.

The class diagram shown in Figure 3.13, specifies the design for handling material properties. Every material type inherits from the class `Material`. Each material is composed of a set of material phases, concept which is modelled through the class `MaterialPhase`. Then, each material phase has an associated set of properties, behaviour that is introduced by inheriting from the class `PropertySet`. This class deals with each material property, or any other property in general, by acting as a container of properties that are identified through a specific `PropertyId`. The base class for each property is the `Property` class, and the different implementations will determine a new type of property. For example, one could deal with scalar properties of type double, by defining a new class (in the diagram named `DoubleProperty`). In case of considering properties that vary with microstructure and temperature, several instances of `MaterialPhase` are created first, one for each phase

of the actual physical material. Secondly, each material property in each material phase is set by specifying a curve that describes the property variation in terms of temperature (this is done, for example, by creating an object of type `PieceWiseLinearFunction` that implements, as suggested by its name, a piecewise linear function).

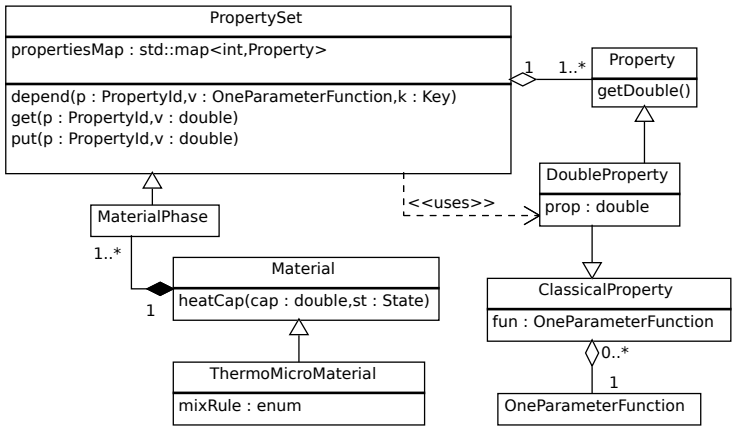


Figure 3.13: Material properties handling.

It can be observed from the class diagram shown in Figure 3.11, that the class `ThermoMicroMaterial` used for modelling the microstructure adds an instance of the class `FlowChart`. This class implements the notion of flowcharts for characterising the microstructure kinetics. Basically, this is done by adding a set of objects of the type `ComponentID`. The class `ComponentID` (details are given in Figure 3.14) specifies the Type ID, TID, and the Link IDs LID0 and LID1, used for building the connectivities defining the flowchart. Every component, identified by a unique ID (TID), has an assigned task to be accomplished when reached by the process flow. Once this task is finished, the next component in the process flow is specified by one of the Link IDs. The two Link IDs LID0 and LID1 are used in decision boxes; in the case of processing steps, only one LID is used (by default the LID0). In addition, there is an enumeration, `cmpType`, identifying the type of task assigned to the box. The different tasks or boxes are modelled as specialisations of the class `ComponentID`, making it necessary to implement the `ComponentID::compute` method for performing a specific task. The `return` processing box, always identified through the TID=0, is not modelled explicitly because it only denotes the end of the current time step. The attribute `flowchartStart` can be observed from a closer look at the

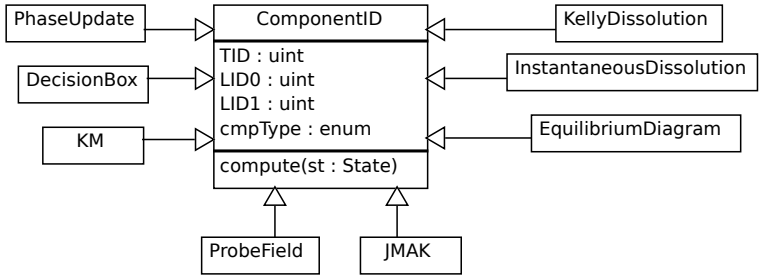


Figure 3.14: Layout of the flowchart's components for modelling microstructure.

class `ThermoMicroMaterial`. This attribute is used for specifying the component of the flowchart identified as the process starting point.

In what follows, a description of each of the flowchart boxes is given. Each of these components is defined in terms of a set of attributes which characterise its behaviour, and in terms of the specific implementation of the method `compute` which calculates the task assigned to the box. This method takes as argument the `State` of the system and returns the TID of the next box to be processed in the flowchart.

The class diagrams of each of the flowchart components can be observed in Figure 3.15. The design of each class is commented below:

- `InstantaneousDissolution`: this kind of component was used in three different situations: TID 3 on Figure 3.6 was used to zero all phases except from liquid; TID 8 on Figure 3.7 was used to zero only the liquid phase without doing anything with other phases; and TID 26 on Figure 3.10 was used to dissolve completely α' contributing to α_w and β according to equations (3.34) and (3.35). From this it is observed that there are situations in which all phase fractions must vanish, except from the one associated to a given phase, say α_i. This behaviour is modelled by specifying only the phase to form, in this case α_i. In other cases, a given phase must dissolve completely, situation that is modelled by specifying only the phase to dissolve (nothing is said about the behaviour of other phases). A final case is that in which a given phase dissolves completely contributing to other phases, situation that is modelled by specifying the phase to dissolve and the phases to form as product of that dissolution.

- `EquilibriumDiagram`: the attribute `phasesToPredict` specifies the Key identifying the phase to update according to the equilibrium phase diagram. The attribute `yeq0` is the function describing the phase fraction according to the equilibrium phase diagram of the phase under analysis.

- `DecisionBox`: the attribute `decisionType` is used to identify the different types of decision boxes, mainly the following boolean expressions: $F_0 < F$, $F < F_1$, $F_0 < F \leq F_1$, $F == F_1$ and $|F - F_1| <$ tol. The attribute `parameter` is a vector of Keys identifying the quantity/ies, denoted in general as F, that are used to evaluate the boolean expression. If the `parameter` size is greater than 1, then F is computed additively as $F = \sum_i$ `parameter[i]`. The attributes `func`, `keyFunc` and `previous` are used when F_1 is defined in terms of a function. The first attribute defines the function itself, the second attribute is the Key that identifies the function domain, and the third attribute specifies if the function must be evaluated at the current or at the previous time step. It must be noted that the **Dec-Reinitialise** processing step is a decision box, which apart from evaluating a boolean expression, can additionally set the specified state variables. Accordingly, this processing step is implemented as a part of a `DecisionBox`, thus avoiding the implementation of another component. The state variables to manage are specified by the attribute `keyToSet`. In order to implement the re-initialisation behaviour, a state variable identified by the Key `lastValue` is used. Basically, when the value of the boolean expression of the decision differs from the value evaluated at the previous time step (identified by the Key `lastValue`), the variables specified in the vector of Keys `keyToSet` are set to zero.

- `JMAK`: the basic JMAK parameters are given by `mi`, `bi`, `tauS` and `yeq`. The phase

or phases to predict are identified through the attribute `phaseToPredict` (remember that simultaneous formation of phases can take place). The Key `inertPhase` is used to identify the inert phase and to implement the behaviour presented in Section 3.2.2.2. The attributes `simPhase` and `phaseToDissolve` are used to model simultaneous precipitation of phases. If the Key `phaseToDissolve` is provided, it is assumed that the model given by equations (3.25-3.28) is used. In this case, `simPhase` identifies the phase that remains constant (α_g in the case of JMAK TID 28, Figure 3.10) and `phaseToPredict` identifies the simultaneously forming phases. If `phaseToDissolve` is not provided but `simPhase` is specified, a different model of simultaneous formation of phases is applied, in which the phase specified in `phaseToPredict` grows and the phase specified in `simPhase` remains constant, as done in JMAK models TIDs 19 and 20, Figure 3.7.

- `KM`: the Koistinen-Marburger equation parameters are given by `exponent` and `ms`. The attribute `retainedPhase` identifies the parent phase that is decomposing. This datum is needed to compute the fraction of the parent phase at the moment of beginning the transformation. The attribute `phaseToPredict` identifies the phase that is forming by means of a diffusionless process. If the attribute `yeq` is set with a given function, the KM model given by equation (3.33) is applied.

- `KellyDissolution`: the attribute `phaseToPredict` is used to identify the phase to predict. The attributes `yeq` and `rateFunction` are piecewise linear functions specifying the equilibrium phase fraction and the parabolic thickening rate, respectively.

- `ProbeField`: the attribute `fieldToProbe` identifies the variable or field to probe. The attribute `saveKey` specifies the Key to use when saving the probed value.

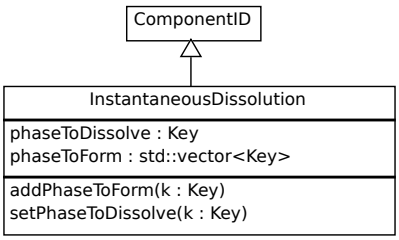
- `PhaseUpdate`: the attribute `phaseToUpdate` identifies the phase to update according to the law $\texttt{st[phaseToUpdate]} = 1 - \sum_{i \in I \setminus \texttt{phaseToUpdate}} \texttt{st[i]}$, where I is the set of indices specifying the different Keys that identify the phase fractions, and `st` denotes the state.

Once the routine `ThermoMicroMaterial::updateMicrostructure(st:State)` is called, the microstructure evolution is computed according to the specified flowchart. This is described by the C++ like pseudo-code presented in Code 2.

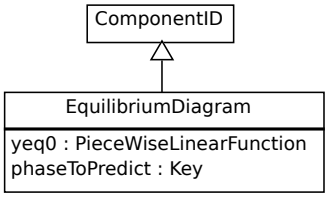
Code 2 Algorithm for updating microstructure by following the flowchart.

```
1 void ThermoMicroMaterial::updateMicrostructure(State &st){
2
3     ComponentID *C = (*flowchart)[flowChartStart];
4     int nextLID;
5     while(C->TID ! 0){
6         nextLID = C->compute(st);
7         C = (*flowchart)[nextLID];
8     }
9 }
```

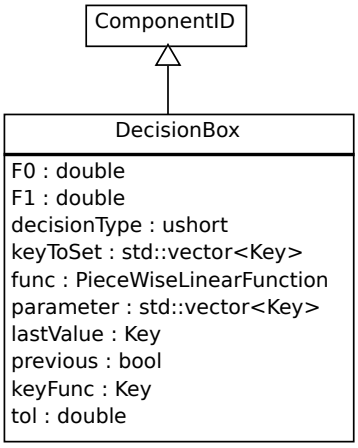
The flowchart and its components defining the microstructural model are provided as part of the data set describing the problem. In this manner, the hard-code programming of algorithms implementing new microstructural models is avoided, because the already



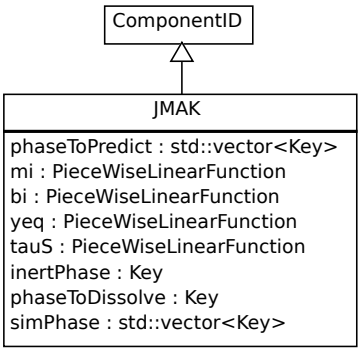
(a) Instantaneous dissolution design.



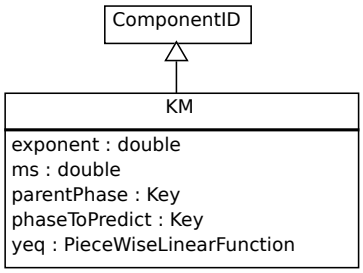
(b) Equilibrium Diagram design.



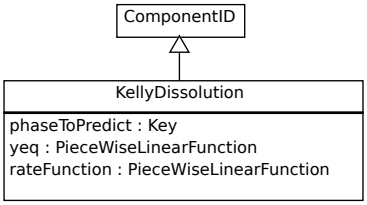
(c) Decision boxes design.



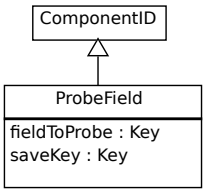
(d) JMAK design.



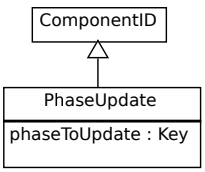
(e) KM design.



(f) KellyDissolution design.



(g) ProbeField design.



(h) PhaseUpdate design.

Figure 3.15: Boxes or flowchart components design.

programmed processing steps are enough for describing the behaviour of most materials. How the flowchart is supplied as part of the data set, highly determines the user experience, and thus the usability of the computational tool. In this point, the use of scripting or high level programming languages provide an interesting option for specifying the data set for a given problem. Some FEM codes, like OOFELIE [39], have an interpreter associated, which decodes each of the provided instructions into a set of objects and configurations defining the algorithms to be used for solving a given problem. Another alternative is the use of Python scripting by extending the Python language with functionalities of the FEM code at hand [89]. This can be done by using either the Cython language [90] or the SWIG interface compiler [91].

In this work, we used the OOFELIE own interpreter to decode the provided instructions. An example of part of the data set specification characterising a microstructural model is next given. The set of instructions defining and configuring a JMAK procedure can be observed in Code 3. In lines 1 and 2, the declaration of the flowchart used for specifying the microstructure kinetics and the JMAK procedure `jmak2` are respectively given. The following lines in the code configure the component `jmak2`. For example, in line 3 the identifier number 2 is assigned to this component, and in line 5 the kinetic parameter m_i is specified by providing the `PieceWiseLinearFunction miB2Agbf` previously defined as part of the data set. Then, the JMAK component is added to the flowchart in line 12. Finally, in lines 15-16 the material `thermalM` of type `ThermoMicroMaterial` is declared, and the flowchart is added by specifying the component with TID 1 as starting component of the flowchart.

Code 3 Example of a JMAK procedure specification.

```
1  FlowChart flowchart;
2  JMAK jmak2;
3  jmak2.TID(2);
4  jmak2.LID0(70);
5  jmak2.setJMAKmi(miB2Agbf);
6  jmak2.setJMAKbi(biB2Agbf);
7  jmak2.setIncubationTime(tAgbf);
8  jmak2.setEquilibriumPhaseFraction(yeqAf);
9  jmak2.setPhaseToPredict(phaseAlphaGB);
10 ...
11 ...
12 flowchart.addFlowChartComp(jmak2);
13 ...
14 ...
15 Material thermalM(ThermoMicroMaterial);
16 thermalM.addFlowChart(flowchart,1);
```

3.6 Application examples

In order to show the capability of the code for predicting microstructure evolutions, the developed models of AISI 1060 steel and Ti6Al4V alloy are applied to several test examples.

Firstly, the quenching of a hollow cylinder made of AISI 1060 steel is simulated. The latent heat associated to solid state transformations, and microstructural dependent prop-

erties are taken into account. This problem is intended to show the capability of solving thermal-microstructural problems where the contribution of solid/solid phase change latent heat is taken into account. This example is based partly in [92], although it must be mentioned that our model differs in some aspects from the model presented there.

Two application examples are afterwards shown, with samples made of Ti6Al4V alloy:

1. The experiment accomplished by Babu *et al.* [71] is reproduced, and the obtained results are compared with experimental measurements and with the results obtained by the model proposed by Murgau *et al.* [72].

2. In the Babu *et al.* experiment, the liquidus temperature is not exceeded. Therefore, in order to test the modelling of solid/liquid transformations, a 3D welding problem is solved. The problem consists on the simulation of a shape metal deposition problem where 4 layers of material are deposited. This problem is similar to that presented by Anca *et al.* [18]. However, in this case we also compute the microstructure evolution.

3.6.1 Quenching of a hollow cylinder.

The simulation of the quenching of a hollow cylinder made of AISI 1060 steel is performed. The cylinder is quenched from an initial temperature of 1103 K to ambient temperature. Its dimensions are: external radius 3×10^{-2}m, internal radius 9×10^{-3}m, and length 6×10^{-2}m. Symmetry conditions are used and only an octant of the geometry is simulated. A time increment of 0.1s is used and the total simulation time is 150s. The geometry is discretised into 20250 tetrahedra and 4096 degrees of freedom. The thermophysical properties, taken from [86], are listed below.

The density [kg/m^3], conductivity [W/K/m] and heat capacity [MJ/m^{-3}/K] of the AISI 1060 steel are given by the following expressions

$$\begin{aligned}
\rho_A &= -0.44789\ T + 8122.3 \\
\rho_F = \rho_P = \rho_B &= -0.42461\ T + 7972.9 \\
\rho_M &= -0.45409\ T + 7886 \\
k_A &= 5.96 \times 10^{-3}\ T + 15 \\
k_F = k_P = k_B &= -1.19 \times 10^{-2}\ T + 49 \\
k_M &= -1.17 \times 10^{-2}\ T + 43 \\
c_A &= 6.5 \times 10^{-4}\ T + 4.14 \\
c_F = c_P = c_B &= 1.7 \times 10^{-3}\ T + 3.76 \\
c_M &= 1.6 \times 10^{-2}\ T + 3.76
\end{aligned}$$

where T is given in Kelvin and the subscripts A, F, P, B and M denote austenite, ferrite, pearlite, bainite and martensite. The latent heat of each phase with respect to austenite can be observed in Figure 3.16. The cylinder is quenched from the external face, process that is modelled by a Robin boundary condition with temperature dependent wall heat convection coefficient as specified in table 3.1. On the internal face of the cylinder, an adiabatic boundary condition is assumed.

The microstructure obtained at the end of the process can be observed in Figure 3.17. The phase fraction of ferrite is not shown because its predicted amount is zero. Phase fractions along the straight line defined by points $(0, 0, 0.03)$ and $(0.03, 0, 0.03)$ can be

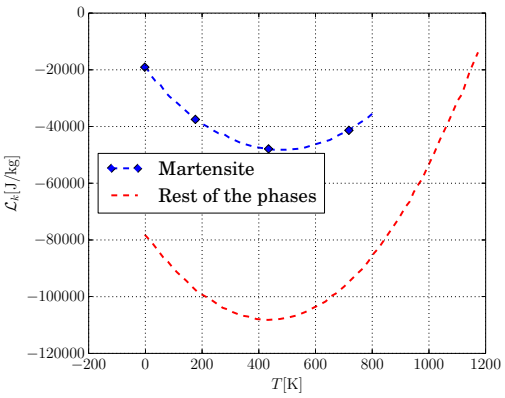


Figure 3.16: Latent heat of each phase with respect to Austenite.

Table 3.1: Wall heat convection coefficient

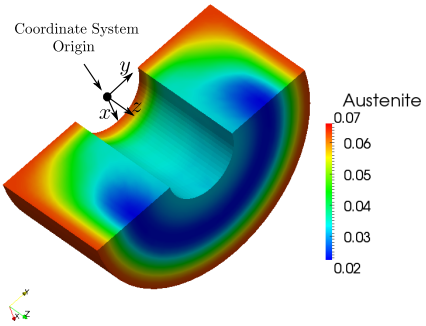
T[K]	473	673	703	773	833	873	973	1073	1173
h[W/m^2/K]	8207	11962	13492	12500	10206	7793	2507	437	200

observed in Figure 3.18, where the points coordinates are given in meters and computed with respect to the origin shown in Figure 3.17a. The computed results can be compared with the simulations made by Şimşir and Hakan [92] for the same problem. At the end of the process they predicted 0 % of austenite, whereas we predict a maximum of 6 %. This result agrees with the work of Wilson and Weins [93], who remarked that an amount of 6 % of retained austenite is expected when quenching the AISI 1060 steel to ambient temperature. Additionally, we predict the total absence of ferrite at the end of the simulation, whereas Şimşir and Hakan predicted a 5 % of ferrite at the midsection. It must be noted that in our model each transformation takes place during a characteristic temperature range in a sequential manner without overlapping. This hypothesis, along with the fast cooling characterising the current case, inhibits the formation of ferrite, that is considered to occur only in the temperature range $\mathrm{Ae}_1 < T \leq \mathrm{Ae}_3$, a quite narrow range as it can be observed from the TTT diagram shown in Figure 3.4. We remarked that in some works ferrite is allowed to form for temperatures below Ae_1, competing with the pearlite transformation [3]. In the current framework, this behaviour could have been similarly implemented as it was done for the simultaneous precipitation of α_w and α_g in the Ti6Al4V alloy.

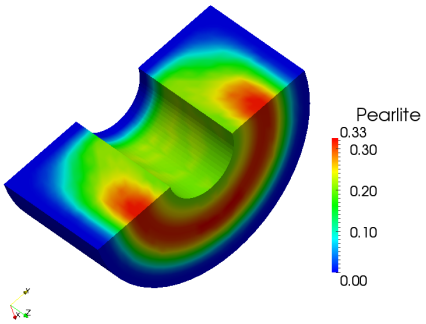
In order to study the contribution of solid/solid phase transformations latent heat, the same problem is simulated but with a wall heat convection coefficient fifty times smaller than the one provided by table 3.1. These coefficients are respectively denoted by h_2 and h_1. In this new simulation, a time increment of 1s and a total simulation time of 2000s are used. The comparison between the results obtained for the two different heat convection coefficients are shown in the plots of Figure 3.19. By observing the resulting temperature histories at the point $(0.009, 0, 0.03)$, it can be noted that in the case of coefficient h_2, the effects of the solid/solid phase transformations over temperature are much more pronounced (the recalescence phenomenon is noticeable at time $t = 800$s. This is because the body is quenched more slowly than in the case of using heat convection

coefficient h_1.

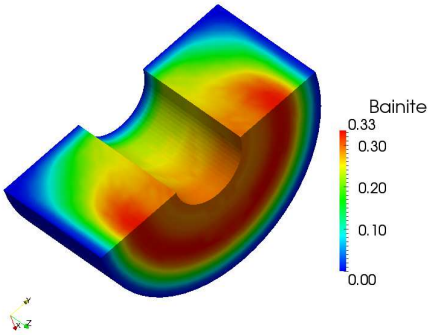
These results show that the proposed computational framework is able to effectively handle solid/solid phase transformations in conjunction with the thermal-microstructural coupling, from which it was possible to capture the recalescence phenomenon. In the next application test, a more complex microstructural model for the Ti6Al4V alloy is studied in detail.



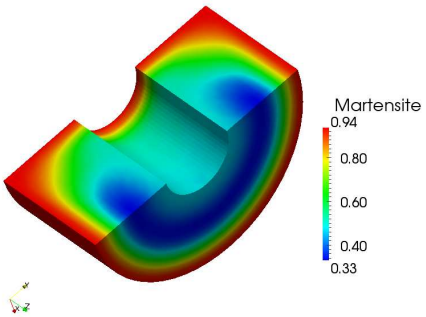
(a) Austenite phase fraction.



(b) Pearlite phase fraction.



(c) Bainite phase fraction.



(d) Martensite phase fraction.

Figure 3.17: Phase fractions at the end of the simulated time interval.

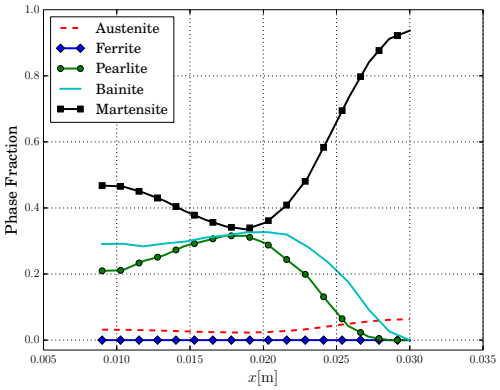
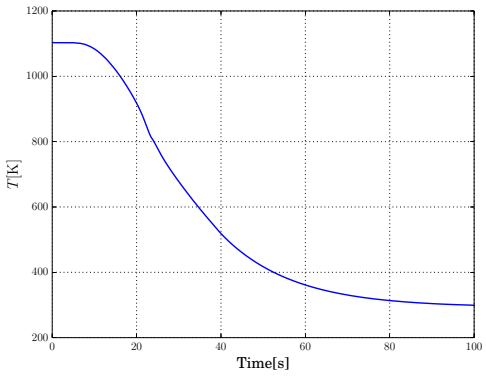
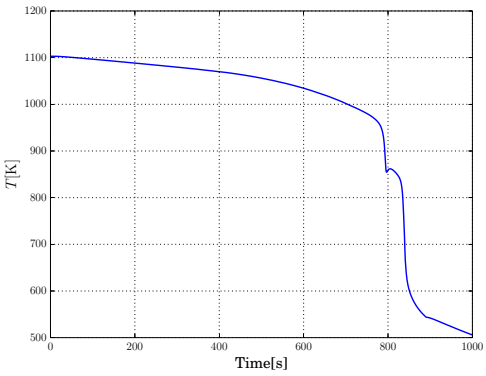


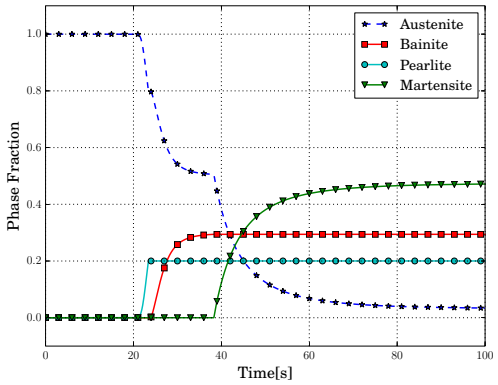
Figure 3.18: Phase fractions along the straight line defined by points $(0, 0, 0.03)$ and $(0.03, 0, 0.03)$.



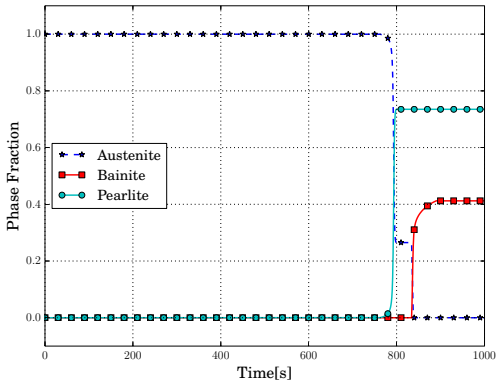
(a) Temperature history for h_1.



(b) Temperature history for h_2.



(c) Phase fractions histories for h_1.



(d) Phase fractions histories for h_2.

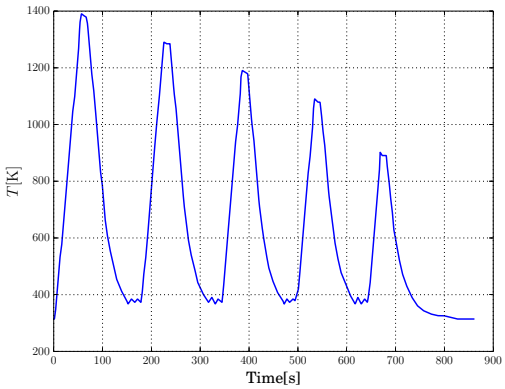
Figure 3.19: Temperature and phase fractions histories at the point $(0.009, 0, 0.03)$. No phase fractions variation is observed for times greater than 1000s in the case of using the small wall heat convection coefficient h_2.

3.6.2 Welding problem: four layers deposition.

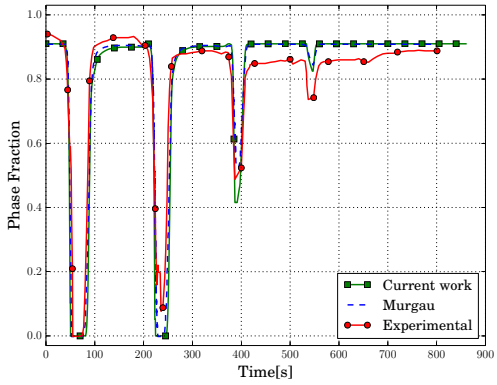
The experiment of Babu *et al.* [71] is reproduced first to validate the model proposed for the Ti6Al4V alloy. The temperature history in this experiment is given in Figure 3.20a. Figure 3.20b displays the comparison between the results of our model with those of the model of Murgau *et al.* and with the experimental measurements. As it can be seen, the obtained results are in good agreement with those of Murgau *et al.* and with the experiments. For completeness, the predicted phase fractions of the different α morphologies are shown in Figure 3.20c.

Additionally, in order to test the modelling of solid/liquid transformations, a 3D welding problem consisting on the simulation of a shape metal deposition of four material layers, is solved. The problem geometry and mesh are shown in Figures 3.21a and 3.21b, respectively.

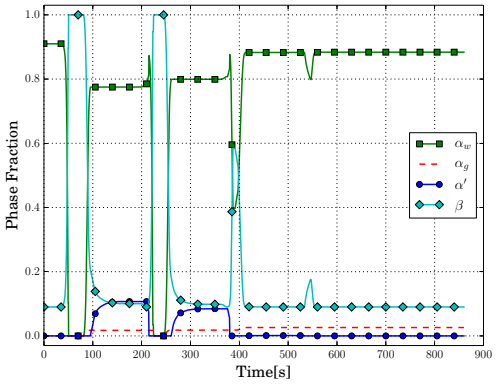
Symmetry conditions with respect to the y-z plane are used, and only one half of the welded piece is modelled. Robin boundary conditions are imposed at all the other faces, including those of the deposited material with the corresponding activation times. A time increment of 0.15s is used for the time interval $[0, 450]$ s. Temperature dependent thermo-



(a) Thermal history.

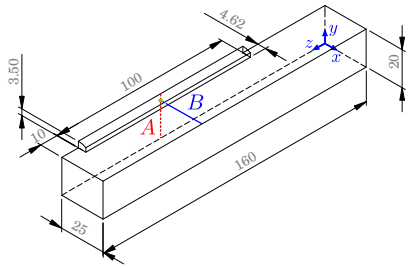


(b) Total amount of phase α, computed as $y_\alpha = y_{\alpha_g} + y_{\alpha_w} + y_{\alpha'}$. Results of our model compared to those of Murgau *et al.* [72] and with experimental measurements.

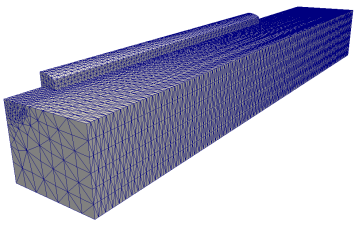


(c) Phase fractions predictions for all phases.

Figure 3.20: Temperature history and results obtained for the Babu *et al.* experiment.



(a) Details of the geometry. Dimensions given in mm.



(b) Details of the mesh.

Figure 3.21: Details of the geometry and of the mesh for the shape metal deposition problem.

physical properties are used, which are specified in Figures 3.24b and 7.9a. The material density is $\rho = 4430 \frac{\text{kg}}{\text{m}}$, and the latent heat associated to the solid/liquid transformation is $\mathcal{L} = 292600 \frac{\text{J}}{\text{kg}}$. The energy input to the medium is described in terms of the Goldak heat

source [66], given by

$$Q(\boldsymbol{\xi}(\boldsymbol{x}),t) = \frac{6\sqrt{3}Q_s}{\pi\sqrt{\pi}bc} \begin{cases} \dfrac{f_f}{a_f}\exp\Big(-3\dfrac{\boldsymbol{\xi}\cdot\boldsymbol{\xi}}{\boldsymbol{a}_f\cdot\boldsymbol{a}_f}\Big) & \text{if} \quad \xi > 0 \\ \dfrac{f_r}{a_r}\exp\Big(-3\dfrac{\boldsymbol{\xi}\cdot\boldsymbol{\xi}}{\boldsymbol{a}_r\cdot\boldsymbol{a}_r}\Big) & \text{if} \quad \xi \leq 0 \end{cases}, \tag{3.45}$$

where $\boldsymbol{a}_f = [a_f, b, c]$, $\boldsymbol{a}_r = [a_r, b, c]$, $\boldsymbol{\xi} = [\xi, \eta, \gamma]$, a_f and a_r are the length parameters associated with the axis of the front and rear semi-ellipsoids, respectively; b and c are the other axes of the semi-ellipsoids; f_f and f_r are the portion of heat distributed in the front and rear semi-ellipsoids, such that $f_f + f_r = 2$; t is the time and Q_s is the total heat input. Generally, Q_s is specified in terms of the electric current I, the voltage V and the heat source efficiency η, as $Q_s = \eta IV$. The local coordinate system $\boldsymbol{\xi}$ is attached to the heat source as shown in Figure 3.22. Initially the torch travels along the welding path without depositing material, performing what it is known as *tigwash*. Then, it travels successively depositing 4 layers of material. The welding plan and the total heat input can be observed in Figure 3.23a. The parameters defining the heat source are given by table 3.23b.

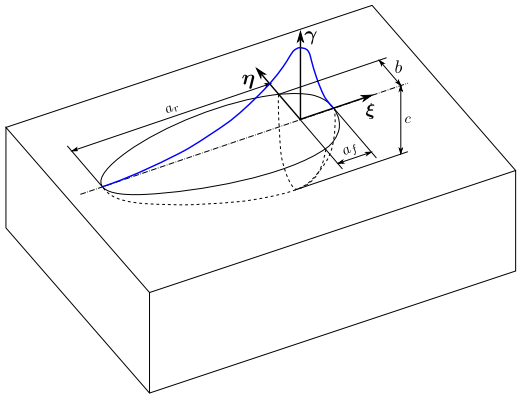
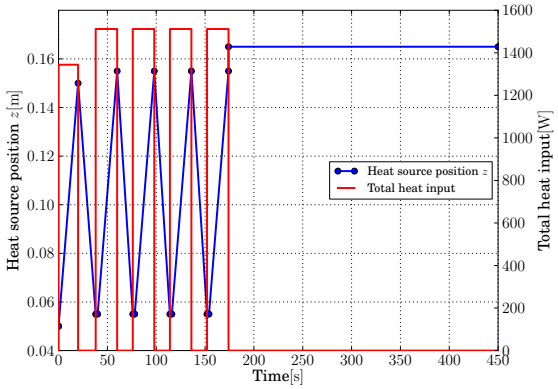


Figure 3.22: Goldak Heat Source.

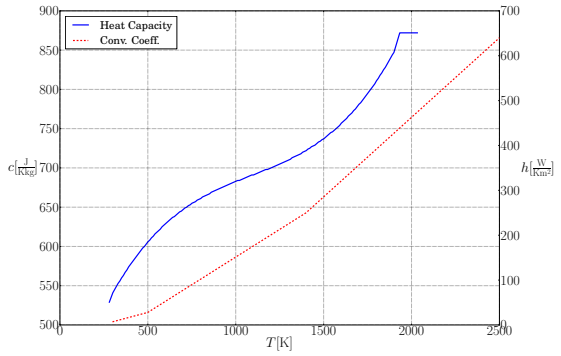


(a) Heat source definition: z coordinate of the torch and total heat input.

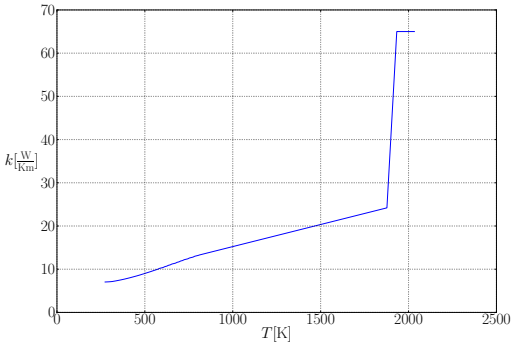
a_r	a_f	b	c	f_f
10.84mm	5mm	2mm	5mm	0.63131

(b) Parameters of the Goldak heat source.

Figure 3.23: Details of parameters defining the heat source.



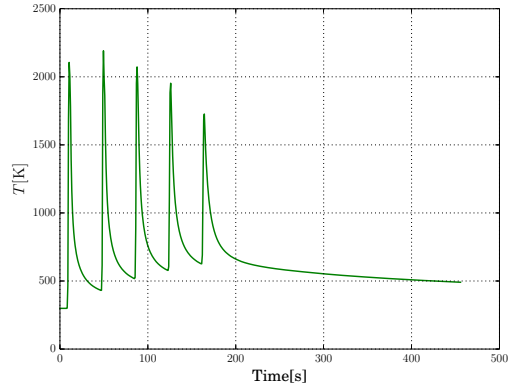
(a) Convection/radiation coefficient and heat capacity.



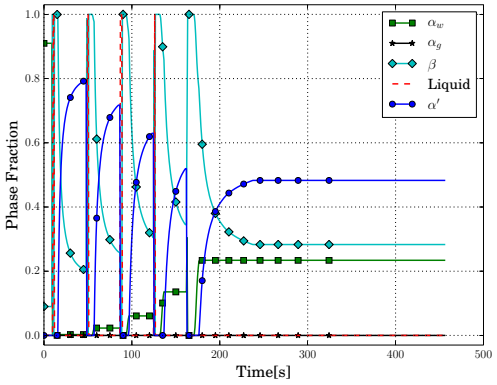
(b) Heat Conductivity.

Figure 3.24: Thermophysical properties.

The thermal history and the microstructure evolution at the point $(0, 0.019, 0.1)$ (coordinates given in m) can be respectively observed in Figures 3.25a and 3.25b. Also, the microstructure profiles along the lines A and B at the end of the simulation, i.e. at $t = 450$s, can be observed in Figure 3.21a. These results show that the proposed computational framework and model are able of handling totally arbitrary thermal histories, with repeated cooling/heating cycles and solid/liquid phase changes.

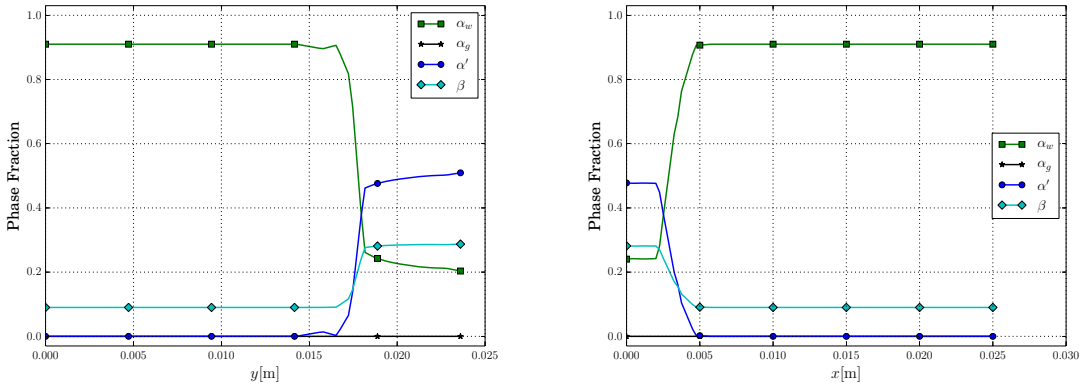


(a) Temperature history.



(b) Microstructure history.

Figure 3.25: Temperature and microstructure history at the point $(0, 0.019, 0.1)$. Coordinates given in m.



(a) Microstructure along the line A.

(b) Microstructure along the line B.

Figure 3.26: Microstructure at the end of the simulation along the lines A and B.

3.7 Conclusions

The microstructure evolution modelling was addressed by studying models formulated at the macro-scale. A basic model for the microstructure evolution prediction during cooling of the AISI 1060 steel was formulated and analysed. Also, a model for the Ti6Al4V alloy capable of predicting microstructure evolution in processes characterised by arbitrary thermal histories, was proposed. Both models were clearly described in terms of evolution flowcharts.

A set of design requirements were inferred from the formulation of the analysed models, to specify a computational tool with enough flexibility for representing a wide range of materials. The use of flowcharts was coined as a fundamental abstraction for defining the microstructure kinetics. The microstructural computational framework was designed as part of a general finite element program, using the Object Oriented Programming paradigm. The design guarantees generality by means of code reusing, because the different boxes that constitute the flowcharts to describe the microstructure kinetics, are hard coded only once and can be connected at will just by modifying the dataset.

The different models and the proposed computational framework were tested in a series of application examples. In the case of the AISI 1060 steel, the quenching of a hollow cylinder was studied. This test showed the ability of the computational tool of handling coupled thermal-microstructural problems, where the contribution of the latent heat terms associated to the solid/solid phase changes, and the variation of thermophysical properties in terms of microstructure, are taken into account. It must be noted that this feature allowed to capture the characteristic recalescence phenomenon.

In the case of the Ti6Al4V alloy, the validity of the proposed model was shown by accurately reproducing the experiment done by Babu *et al.* [71]. Then, the complexities posed by arbitrary thermal histories and solid/liquid phase changes were studied by solving a welding problem.

Future work will be addressed to further study microstructural models with a formulation at the meso-scale, looking for more physically based models. Also, much of the effort will be directed towards obtaining computational amenable meso-scale formulations.

Chapter 4

Mechanical Problem

The purpose of this chapter is to provide an introduction to the mechanical problem. First, the mathematical description of the problem is given by revisiting concepts dealing with the kinematics of body motion, the material behaviour definition through specific constitutive relations, and the statement of the field equations governing the mechanical problem. Then, the variational formulations associated to the irreducible and mixed versions of the mechanical problem are presented in order to be subsequently discretised by the Finite Element Method (FEM). A brief comment regarding the algorithm used to integrate the rate equations characterising plastic flow, is given. In the following, details specific to welding and phase change problems are presented, by providing an effective thermo-mechanical numerical framework able to handle the material deposition and the solid/liquid phase change of arbitrary regions of the body under analysis. At the end of this chapter, different examples are numerically solved in order to validate the implemented two-dimensional numerical tools.

4.1 Problem description: Mathematical setting

In this section, basic concepts regarding the mathematical description of the mechanical problem are introduced. First, in order to describe the kinematics of arbitrary bodies, the notions of configuration, deformation and strain are considered. Then, the constitutive relations of the different materials to be employed in this work are presented. Finally, the field equations defining the mechanical problem are introduced by considering the balance of linear and angular momentum.

All these topics are treated very briefly, because they are not at the core of the original aspects of this work. The interest reader is referred to specialised books [94, 60, 95, 96, 97].

4.1.1 Kinematics: configurations, deformation and strain

To study the motion of an arbitrary body B, it is assumed that it can be described as a continuum [94, 60]. The one-to-one mapping $\boldsymbol{\varphi} : B \rightarrow \mathbb{E}^3$, which maps particles $X \in B$ to the Euclidean space $\mathbb{E}^3$, is called configuration of the body B. The motion of the body is defined in terms of the one-parameter family of configurations $\boldsymbol{\varphi}_t = \boldsymbol{\varphi}(X, t)$, where t denotes time. The configuration used to identify or label the different particles

X comprising the body is referred to as reference configuration, and it is denoted by $\boldsymbol{\varphi}_0$. That is, for an arbitrary particle X we have

$$\boldsymbol{X} = \boldsymbol{\varphi}_0(X), \tag{4.1}$$

where $\boldsymbol{X}$ is the position of particle X according to the chosen reference configuration $\boldsymbol{\varphi}_0$. Using the last equation in the expression for the position $\boldsymbol{x}$ occupied by the particle X at time t, we have

$$\boldsymbol{x} = \boldsymbol{\varphi}_t(X,t) = \boldsymbol{\varphi}_t(\boldsymbol{\varphi}_0^{-1}(\boldsymbol{X}),t) = \boldsymbol{\varphi}_t(\boldsymbol{X},t). \tag{4.2}$$

By choosing appropriate observers $\boldsymbol{O}$ and $\boldsymbol{o}$, and bases $\boldsymbol{E}_i$ and $\boldsymbol{e}_i$ associated to the reference and current configurations, respectively, the location of particles in the Euclidean space can be assimilated to vectors lying in the vector space $\mathbb{R}^3$ by means of position vectors.

A material line element, or fibre, $\boldsymbol{dX}$ in the reference configuration of B maps to the material line element $\boldsymbol{dx}$ in the current configuration by means of the deformation gradient $\boldsymbol{F}$ as $\boldsymbol{dx} = \boldsymbol{F}\boldsymbol{dX}$. By defining the difference $\boldsymbol{x} - \boldsymbol{X}$ as the displacement $\boldsymbol{u}$ of particle X from the reference to the current configuration, the following expression for the deformation gradient arises

$$\boldsymbol{F} = \mathbf{1} + \boldsymbol{D}, \tag{4.3}$$

where $\boldsymbol{D}$ is the displacement gradient and $\mathbf{1}$ the second order identity tensor. A strain measure is provided by the Green-Lagrange strain tensor $\boldsymbol{E}$, which is

$$\boldsymbol{E} = \frac{1}{2}(\boldsymbol{F}^T\boldsymbol{F} - \mathbf{1}) = \frac{1}{2}(\boldsymbol{D} + \boldsymbol{D}^T + \boldsymbol{D}^T\boldsymbol{D}). \tag{4.4}$$

The small deformation hypothesis is adopted, from which the non-linear term $\boldsymbol{D}^T\boldsymbol{D}$ of the last equation vanishes, and $\boldsymbol{E}$ becomes

$$\boldsymbol{\varepsilon} = \frac{1}{2}(\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T) \doteq \nabla^s\boldsymbol{u}, \tag{4.5}$$

where the expression for the displacement gradient $\boldsymbol{D} = \nabla\boldsymbol{u}$ was used, and the symmetric gradient operator ∇^s was defined.

4.1.2 Material behaviour: constitutive relations

In welding problems [17, 40] the total strain $\boldsymbol{\varepsilon}$ is decomposed into different physically based terms, each of which additively contributes to $\boldsymbol{\varepsilon}$ as

$$\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}^e - \boldsymbol{\varepsilon}^{th} + \boldsymbol{\varepsilon}^i + \boldsymbol{\varepsilon}^{tr} + \boldsymbol{\varepsilon}^v, \tag{4.6}$$

where $\boldsymbol{\varepsilon}^e$ is the elastic strain, $\boldsymbol{\varepsilon}^{th}$ is the thermal strain, $\boldsymbol{\varepsilon}^i$ is the plastic strain, and $\boldsymbol{\varepsilon}^{tr}$ and $\boldsymbol{\varepsilon}^v$ are the strains associated to microstructural transformations, respectively, the strain due to Transformation Induced Plasticity (TRIP) and the strain due to microstructural volume change. Volume change associated to microstructural transformations and TRIP will be not taken into account. More experimental research is needed in this direction, subject which is out of the scope of this study.

In order to describe the elastic response of the material, the Hooke's Law is adopted [19, 18] relating $\boldsymbol{\varepsilon}^e$ to the stress tensor $\boldsymbol{\sigma}$ by the expression

$$\boldsymbol{\sigma} = \mathbb{C}\boldsymbol{\varepsilon}^e, \tag{4.7}$$

where $\mathbb{C}$ is the tensor of elastic moduli which is given, for the case of a hypoelastic Hookean material, by

$$\mathbb{C} = \lambda \mathbf{1} \otimes \mathbf{1} + 2\mu \mathbf{I} = \kappa \mathbf{1} \otimes \mathbf{1} + 2\mu \Big(\mathbf{I} - \frac{1}{3} \mathbf{1} \otimes \mathbf{1} \Big), \tag{4.8}$$

where $\mathbf{I}$ is the fourth order identity tensor, λ and μ are the Lamé parameters and $\kappa = \dfrac{3\lambda + 2\mu}{3}$ is the bulk modulus. In the applications presented in this work, the elastic material parameters are specified through the Young's modulus E and the Poisson ratio ν, which are in the general case considered temperature dependent parameters, and are related to μ and λ by the expressions

$$\lambda = \frac{E\nu}{(1+\nu)(1-2\nu)} \quad \text{and} \quad \mu = \frac{E}{2(1+\nu)}. \tag{4.9}$$

Welding problems are thermally driven processes characterised by big changes in the temperature field, situation which induces considerable thermal expansion in the piece being welded. In order to include this effect, equation (4.7) is modified leading to the following thermo-elastic relation

$$\boldsymbol{\sigma} = \mathbb{C}(\boldsymbol{\varepsilon}^e - \boldsymbol{\varepsilon}^{th}). \tag{4.10}$$

In the theory of linear thermo-elasticity the thermal strain $\boldsymbol{\varepsilon}^{th}$ is taken as $\boldsymbol{\varepsilon}^{th} = \alpha_T(T - T_{\text{ref}})\mathbf{1}$, where α_T is the thermal expansion coefficient, T is the temperature and T_{ref} is an arbitrary reference temperature. This is valid while $T - T_{\text{ref}}$ is small [31], assumption that is not met by welding problems. In order to solve this issue, $\boldsymbol{\varepsilon}^{th}$ is computed in terms of the Thermal Linear Expansion function ϵ^{th} which is defined as

$$\epsilon^{th}(T) = \int_{T_{\text{ref}}}^{T} \alpha_T(\tau) \mathrm{d}\tau, \tag{4.11}$$

leading to the following expression for the thermal strain

$$\boldsymbol{\varepsilon}^{th} = \epsilon^{th}(T)\mathbf{1}. \tag{4.12}$$

From this, the stress tensor takes the form

$$\boldsymbol{\sigma} = \kappa\big[\mathrm{tr}(\boldsymbol{\varepsilon}^e) - 3\epsilon^{th}\big]\mathbf{1} + 2\mu \boldsymbol{e}^e = p\mathbf{1} + \boldsymbol{s}, \tag{4.13}$$

where $\boldsymbol{e}^e = \mathrm{dev}(\boldsymbol{\varepsilon}^e)$ is the strain tensor deviator, $p = \mathrm{tr}(\boldsymbol{\sigma})/3$ is the mean stress and $\boldsymbol{s}$ is the deviatoric part of the stress tensor $\boldsymbol{s} = \mathrm{dev}(\boldsymbol{\sigma})$.

The inelastic part of the total strain, given by $\boldsymbol{\varepsilon}^i$ in equation (4.6), is associated to the irreversible process of plastic flow. Because of the characteristic time scales involved in welding problems, the use of rate independent material models is justified [98] and, then, no rate dependent models are considered in this work to describe the material response during welding.

A strain-driven formulation is adopted in order to describe plastic deformations. Consequently, the plastic flow at a given time t and position $\boldsymbol{x} \in \mathbb{R}^3$ is described by means of the history of the total strain $\boldsymbol{\varepsilon}$, its inelastic part $\boldsymbol{\varepsilon}^i$ and a set of hardening variables $\boldsymbol{\alpha}$ [96]. Then, the stress tensor $\boldsymbol{\sigma}$ depends on the variables $\boldsymbol{\varepsilon}$ and $\boldsymbol{\varepsilon}^i$ through the thermo-elastic relationship (4.10) by making use of the expression

$$\boldsymbol{\varepsilon}^e - \boldsymbol{\varepsilon}^{th} = \boldsymbol{\varepsilon} - \boldsymbol{\varepsilon}^i. \tag{4.14}$$

According to Mullins and Gunnars[99], when computing weld residual stresses in pipe welds, an isotropic hardening model better approximates experimental measurements than a kinematic or mixed hardening model. Following this observation, an isotropic elasto-plastic hardening model is adopted, and kinematic hardening is not considered. Then, just one hardening variable is needed and $\boldsymbol{\alpha}$ becomes the scalar quantity α.

By adopting a J2 associative plasticity model and the Von Mises yield criterion, the yield function $f(\boldsymbol{\sigma}, \alpha)$ is given by

$$f(\boldsymbol{\sigma}, \alpha) = ||\boldsymbol{s}|| - \sqrt{\frac{2}{3}} K(\alpha), \tag{4.15}$$

where $||\boldsymbol{s}|| = \sqrt{\boldsymbol{s} : \boldsymbol{s}}$ denotes the norm of the stress deviator $\boldsymbol{s}$, and $K(\alpha)$ is the hardening function. In this context, the flow rule is given by

$$\dot{\boldsymbol{\varepsilon}}^i = \gamma \frac{\boldsymbol{s}}{||\boldsymbol{s}||} = \gamma \boldsymbol{n}, \tag{4.16}$$

and the evolution law for isotropic hardening is

$$\dot{\alpha} = \gamma \sqrt{\frac{2}{3}} = \sqrt{\frac{2}{3}} ||\dot{\boldsymbol{\varepsilon}}^i||, \tag{4.17}$$

where γ is the consistency parameter satisfying the Kuhn-Tucker conditions

$$\gamma \geq 0, \quad f(\boldsymbol{\sigma}, \alpha) \leq 0, \quad \gamma f = 0 \tag{4.18}$$

and, from there, the relation $\gamma \dot{f} = 0$. By integrating the hardening law (4.17), α becomes

$$\alpha = \sqrt{\frac{2}{3}} \int_0^t ||\dot{\boldsymbol{\varepsilon}}^i|| dt, \tag{4.19}$$

expression formally defined as the *equivalent plastic deformation*, a measure of the amount of plastic distortion [60].

Specific expressions for the hardening function $K(\alpha)$ depend upon the adopted law for hardening. The exponential law proposed by Voce [100] is general enough for describing the behaviour of materials encountered in real life applications. This law is given by

$$K(\alpha) = \sigma_Y + k_0 \alpha + k_1 (1 - e^{-\delta \alpha}), \tag{4.20}$$

where σ_Y is the yield stress, and $k_0 \geq 0$, $k_1 \geq 0$ and $\delta \geq 0$ are material constants that can depend on temperature. It must be noted that when $k_1 = 0$ the typical linear relation for isotropic hardening is recovered, and when additionally $k_0 = 0$ a perfectly plastic material behaviour is obtained.

4.1.3 Field equations of the mechanical problem

We assume that when the neighbourhood of a material point is at a temperature above the Zero Strength Temperature (ZST), T_{zst}, it has no longer mechanical strength, and that it only begins to develop strength when it cools down to a temperature below T_{zst}. Let Ω_m be the region in $\mathbb{R}^3$ formed by all particles whose temperature is below T_{zst}, or equivalently, by particles that have mechanical strength. The initial boundary value

problem characterising the mechanical problem is obtained by considering equilibrium of linear momentum and angular momentum. Accordingly,

$$\nabla \cdot \boldsymbol{\sigma} + \rho \boldsymbol{b} = \mathbf{0} \quad \forall (\boldsymbol{x}, t) \in \Omega_m \times [t_0, \infty), \tag{4.21}$$

$$\boldsymbol{\sigma} = \boldsymbol{\sigma}^T, \tag{4.22}$$

$$\boldsymbol{u} = \bar{\boldsymbol{u}} \quad \forall (\boldsymbol{x}, t) \in \Gamma_u \times [t_0, \infty), \tag{4.23}$$

$$\boldsymbol{\sigma n} = \bar{\boldsymbol{t}} \quad \forall (\boldsymbol{x}, t) \in \Gamma_\sigma \times [t_0, \infty), \tag{4.24}$$

where $\boldsymbol{b}$ are given body forces, ρ is the mass density, t_0 is the initial time, $\bar{\boldsymbol{u}}$ and $\bar{\boldsymbol{t}}$ are boundary conditions respectively imposed on non-overlapping regions Γ_u and Γ_σ of $\partial\Omega_m \equiv \Gamma_m$, and $\boldsymbol{n}$ is the external normal to Γ_σ.

Remark: in a general case, Ω_m can be a not simply connected region; however, this case will not be considered.

4.2 Problem discretisation

In this section the discretisation of the mechanical problem is addressed. The first issue to be discussed is the variational formulation and the FEM spatial discretisation of the problem stated by equations (4.21-4.24).

Let the trial solution and test spaces be given by $\mathcal{S} = \{\boldsymbol{u} | \boldsymbol{u} \in \mathcal{H}^1, \boldsymbol{u}|_{\Gamma_u} = \bar{\boldsymbol{u}}\}$ and $\mathcal{V} = \{\boldsymbol{u} | \boldsymbol{u} \in \mathcal{H}^1, \boldsymbol{u}|_{\Gamma_u} = \mathbf{0}\}$, respectively where $\mathcal{H}^1 = [\mathcal{H}^1]^3$ with $\mathcal{H}^1$ as the first order Sobolev space. Then, the variational formulation of the mechanical problem, irreducible in displacements, reads: Find $\boldsymbol{u} \in \mathcal{S}$ such that for every $\boldsymbol{v} \in \mathcal{V}$

$$\int_{\Omega_m} \boldsymbol{\sigma} : \nabla^s \boldsymbol{v} \, d\Omega = \int_{\Omega_m} \rho \boldsymbol{b} \cdot \boldsymbol{v} \, d\Omega + \int_{\Gamma_\sigma} \bar{\boldsymbol{t}} \cdot \boldsymbol{v} \, d\Gamma. \tag{4.25}$$

It is well-known that discrete versions of (4.25) can lead to instabilities for an incompressible material [97]. Pathologies of that kind can emerge in welding problems, because the material behaves as quasi-incompressible in the elastic range as ν approaches 0.5 and, additionally, incompressible plastic deformations are assumed in the context of the J2 flow theory. In order to deal with this kind of problems, a mixed formulation can be used. In this work, this type of formulation is built by adopting the displacement $\boldsymbol{u}$ and the mean stress p as unknowns of the problem. The trial or solution space for the mean stress is given by $\mathcal{P} = \{p | p \in L_2\}$, where L_2 is the space of square integrable functions, which agrees with the test space because no explicit boundary conditions are imposed on p [97]. The test and trial spaces for $\boldsymbol{u}$ remain the same as before. Then, the corresponding mixed variational formulation reads: Find $(\boldsymbol{u}, p) \in \mathcal{S} \times \mathcal{P}$ such that for every $(\boldsymbol{v}, q) \in \mathcal{V} \times \mathcal{P}$

$$\int_{\Omega_m} \boldsymbol{s} : \nabla^s \boldsymbol{v} \, d\Omega + \int_{\Omega_m} p\mathbf{1} : \nabla^s \boldsymbol{v} \, d\Omega - \int_{\Omega_m} \rho \boldsymbol{b} \cdot \boldsymbol{v} \, d\Omega - \int_{\Gamma_\sigma} \bar{\boldsymbol{t}} \cdot \boldsymbol{v} \, d\Gamma = 0, \tag{4.26}$$

$$\int_{\Omega_m} q \Big[\mathrm{tr}(\boldsymbol{\varepsilon}^e) - 3\epsilon^{th} - \frac{p}{\kappa} \Big] d\Omega = 0, \tag{4.27}$$

where use of the expression (4.13) was made.

The finite element spatial discretisations, in the irreducible and mixed forms, are based on the corresponding variational formulations previously introduced. Let $\mathcal{S}^h \subset \mathcal{S}$ and $\mathcal{V}^h \subset \mathcal{V}$ be discrete versions of the trial and test spaces, and let $\boldsymbol{v}^h \in \mathcal{V}^h$ be given by[1]

[1]Note that $\mathcal{S}^h$ is an affine translation of span$\{\mathcal{V}^h\}$.

$$\boldsymbol{v}^h(\boldsymbol{x}) = \sum_{i=1}^{n_v} N_i(\boldsymbol{x})\boldsymbol{U}_i \doteq \boldsymbol{N}^T\boldsymbol{U}, \tag{4.28}$$

where n_v is the number of nodes and $\boldsymbol{U}_i$ is the displacement associated to node i. Denoting by matrix $\boldsymbol{B}$ the discrete version of the kinematic strain-displacement relation, the discretisation of the variational formulation (4.25) is given by

$$\boldsymbol{\Pi} = \int_{\Omega_m} \boldsymbol{B}^T\boldsymbol{\sigma} d\Omega - \int_{\Omega_m} \rho\boldsymbol{N}^T\boldsymbol{b} d\Omega - \int_{\Gamma_\sigma} \boldsymbol{N}^T\bar{\boldsymbol{t}} d\Gamma = \boldsymbol{0}. \tag{4.29}$$

Then, the problem consists in solving the nonlinear problem $\boldsymbol{\Pi} = \boldsymbol{0}$, which is made using a Newton-Raphson scheme

$$\boldsymbol{\Pi}^{(i+1)} \simeq \boldsymbol{\Pi}^{(i)} + \frac{\partial \boldsymbol{\Pi}}{\partial \boldsymbol{U}}^{(i)} (\boldsymbol{U}^{(i+1)} - \boldsymbol{U}^{(i)}) = \boldsymbol{0} \tag{4.30}$$

where i represents the i^{th} iteration. Iterations proceed until convergence (the norm of the residual meets a prescribed tolerance). Due to the high non linearity of welding problems, a line-search method must be used in conjunction with the Newton-Raphson scheme [60, 61].

The discrete version $\mathcal{P}^h$ of $\mathcal{P}$ is introduced for the discretisation of the mixed formulation (4.26-4.27). As similarly done for the displacement field, the mean stress test functions $p^h \in \mathcal{P}^h$ are given by

$$p^h(\boldsymbol{x}) = \sum_{i=1}^{n_p} \bar{N}_i(\boldsymbol{x})p_i \doteq \bar{\boldsymbol{N}}^T\boldsymbol{p}, \tag{4.31}$$

where n_p is the number of mean stress nodes and p_i is the mean stress associated to node i. In this case, the discretisation of the variational formulation leads to

$$\underbrace{\int_{\Omega_m} \boldsymbol{B}^T\boldsymbol{s} d\Omega}_{\boldsymbol{F}_u} + \underbrace{\int_{\Omega_m} \boldsymbol{B}^T\bar{\boldsymbol{N}} d\Omega}_{\boldsymbol{G}}\boldsymbol{p} - \underbrace{\int_{\Omega_m} \rho\boldsymbol{N}^T\boldsymbol{b} d\Omega - \int_{\Gamma_\sigma} \boldsymbol{N}^T\bar{\boldsymbol{t}} d\Gamma}_{\boldsymbol{F}_{\text{ext}}} = \boldsymbol{0}, \tag{4.32}$$

$$\underbrace{\int_{\Omega_m} \bar{\boldsymbol{N}}^T\boldsymbol{B} d\Omega}_{\boldsymbol{G}^T}\boldsymbol{U} - \underbrace{\int_{\Omega_m} 3\epsilon^{th}\bar{\boldsymbol{N}}^T d\Omega}_{\boldsymbol{F}_{th}} - \underbrace{\int_{\Omega_m} \frac{1}{\kappa}\bar{\boldsymbol{N}}^T\bar{\boldsymbol{N}} d\Omega}_{\bar{\boldsymbol{M}}}\boldsymbol{p} = \boldsymbol{0}. \tag{4.33}$$

The latter equations can be written in matrix form as

$$\boldsymbol{F}_u + \boldsymbol{G}\boldsymbol{p} = \boldsymbol{F}_{\text{ext}}, \tag{4.34}$$

$$\boldsymbol{G}^T\boldsymbol{U} - \bar{\boldsymbol{M}}\boldsymbol{p} = \boldsymbol{F}_{th}, \tag{4.35}$$

from which the following expression for the mean stress is obtained

$$\boldsymbol{p} = \bar{\boldsymbol{M}}^{-1}(\boldsymbol{G}^T\boldsymbol{U} - \boldsymbol{F}_{th}). \tag{4.36}$$

Then, this last equation is used in (4.34) in order to statically condense the mean stress. Therefore, $\boldsymbol{\Pi}$ is redefined to

$$\boldsymbol{\Pi} = \boldsymbol{F}_u + \boldsymbol{G}\bar{\boldsymbol{M}}^{-1}\boldsymbol{G}^T\boldsymbol{U} - \boldsymbol{F}_{\text{ext}} - \boldsymbol{G}\bar{\boldsymbol{M}}^{-1}\boldsymbol{F}_{th}. \tag{4.37}$$

The choice for the basis functions $\boldsymbol{N}$ and $\bar{\boldsymbol{N}}$ for interpolating the displacement field and the mean stress must satisfy the Babuška-Brezzi (BB) condition [101, 97], otherwise

instabilities can take place. Note that the static condensation performed to arrive to equation (4.37) can be done at the element level if discontinuous mean stress elements are used. In practise, for two dimensional problems the Q1-P0 mixed finite element is frequently used. This element is a quadrilateral, where displacements are bilinearly interpolated and the mean stress is discontinuously interpolated by constant shape functions. Although its performance in elasto-plastic problems is quite good, it must be kept in mind that it does not satisfy the BB condition.

In the current work, the Q1-P0 element is implemented for two dimensional problems and used for solving the mechanical problem. In the case of three dimensional problems, mechanical hexahedral elements are used, where the displacements are trilinearly interpolated and the mean stress is discontinuously interpolated by constant shape functions.

For two dimensional problems the plane strain, the generalised plane strain (GPS) and the axisymmetric hypotheses are implemented and tested. In the case of plane strain, from the point of view of the implementation the strain-displacement matrix $\boldsymbol{B}_i$ associated to node i takes the form

$$\boldsymbol{B}_i = \begin{bmatrix} \frac{\partial N_i}{\partial x} & 0 \\ 0 & \frac{\partial N_i}{\partial y} \\ \frac{\partial N_i}{\partial y} & \frac{\partial N_i}{\partial x} \end{bmatrix}, \tag{4.38}$$

where (x, y) are the spatial coordinates. For axisymmetric problems, this matrix is given by

$$\boldsymbol{B}_i = \begin{bmatrix} \frac{\partial N_i}{\partial x} & 0 \\ 0 & \frac{\partial N_i}{\partial y} \\ \frac{\partial N_i}{\partial y} & \frac{\partial N_i}{\partial x} \\ \frac{N_i}{x} & 0 \end{bmatrix}, \tag{4.39}$$

where in this case x and y denote the radial and axial coordinates, respectively.

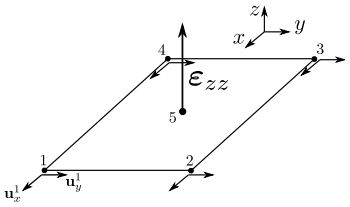


Figure 4.1: Generalised plane strain element details.

When considering the generalised plane strain kinematic hypothesis, $\boldsymbol{\varepsilon}_{zz}$ is assumed to be different from zero and a free traction boundary condition in the z direction is imposed by means of the following constraint

$$\int_{\Omega_m} \boldsymbol{\sigma}_{zz} d\Omega = 0. \tag{4.40}$$

Following the work of Risso *et al.* [102], in order to incorporate the component $\boldsymbol{\varepsilon}_{zz}$ to the computations, a new degree of freedom is added as shown in Figure 4.1, and the matrix $\boldsymbol{B}$ at the element level takes the form

$$\boldsymbol{B}^e = \begin{bmatrix} \boldsymbol{B}_1 & \boldsymbol{B}_2 & \boldsymbol{B}_3 & \boldsymbol{B}_4 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & 1 \end{bmatrix}. \tag{4.41}$$

The additional degree of freedom $\boldsymbol{\varepsilon}_{zz}$ is global in nature. Lagrange multipliers are used to ensure that every element in the mesh shares the same degree of freedom.

4.2.1 Overall logic for elasto-plasticity

The different FEM formulations to be used for solving the balance equations (4.21-4.24) were previously discussed. When considering dissipative material laws, the problem becomes non-linear and the Newton-Raphson scheme specified by equation (4.30) is used for finding the solution. At each Newton iteration, an incremental displacement field is calculated, from which the associated incremental strain is computed. Knowing the strain history, the state variables are updated according to a specified elasto-plastic model. Next, we test if the computed stresses are at equilibrium. If this is the case, convergence to the solution is concluded, otherwise a new iteration is accomplished in order to correct the approximation. By following the work of Simo and Hughes [96], the evolution laws associated to an elasto-plastic model are integrated by making use of a Backward Euler scheme, and the state variables are updated by means of a classical return mapping algorithm.

4.3 Material deposition and phase change handling in thermo-mechanic problems

In the context of welding problems, the mesh of the full geometry comprised by the base metal and the layers or weld beads to be deposited is generated at the beginning of the simulation. How these layers are considered in the thermal and mechanical analyses depends on the type of method used for modelling material deposition. Two main methods are aimed at modelling material deposition of multipass welding problems, the *quiet* and the *inactive* element methods [103, 104]. At a given time t of the simulation of the welding process, these methods are able to handle elements which have not yet been deposited according to the welding plan, referred to as "*quiet*" or "*inactive*" elements. In the case of the *quiet* element method, quiet elements are considered as part of the thermo-mechanical analysis, but with assigned material properties which makes their contribution to the problem negligible. The obvious big disadvantage of this method is the induced ill conditioning in the linear system of equations.

In the *inactive* element method, each element has an activation time associated which allows to distinguish between active elements, which are considered part of the computations, and inactive elements, which do not take part of the analysis. In this context, the management of inactive elements does not generate an ill conditioning of the system equations, and the numerical scheme as a whole is more robust. The only disadvantage is that the profile of the matrix of the linear system of equations changes every time that an inactive element is activated. In this work, the *inactive* element method is used to manage elements which have not been yet deposited, although some minor modifications distinguish it from classic inactive element techniques.

Let t_n be the current time of analysis. The status of an arbitrary element is considered to be *inactive* when its activation time t_a (provided by the welding plan) is greater than t_n. When an inactive element changes its status from inactive to active, because $t_a \leq t_n$, it is incorporated to the thermal analysis but not to the mechanical one. The activation of mechanical elements is based on the temperature of the element nodes. By considering that the neighbourhood of a given material point develops strength only when its temperature T is below the ZST, a mechanical element is activated only when all of its nodes fulfil the requirement $T < T_{\text{zst}}$, and, additionally, one of the following requirements is satisfied

- In the two dimensional case the element to be activated shares at least one edge

with another element whose status is active.

- In the three dimensional case the element to be activated shares at least one face with another element whose status is active.

This last condition is checked in order to avoid the case detailed in Figure 4.2.

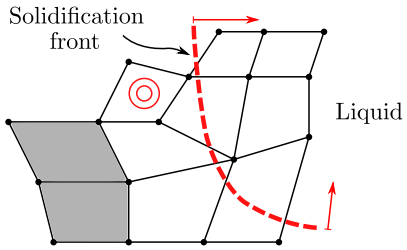


Figure 4.2: Activation/deactivation of mechanical elements. Every element shown in the figure is activated in the thermal analysis, although mechanical elements are only activated if they have enough strength. In grey activated mechanical elements, in white and identified with a double circle, a solid element which is discarded because it does not share at least an edge with one of the activated elements. If this element were activated, it could artificially rotate.

From the numerical point of view, in the case of thermo-mechanical problems where solid/liquid phase changes take place, the use of stress free or natural configurations is considered in this work to correctly compute thermal stresses. When a mechanical element is activated, the stress, the strain and in general any internal variable, must all be zero. This is done by allowing the neighbourhood of every material point to adopt a natural configuration at the time instant at which the considered material point begins to develop strength. The idea is depicted in Figure 4.3. The displacement that actually causes the body to deform is denoted by $\boldsymbol{u}$ and defined in terms of the displacements $\boldsymbol{u}_t$ and $\boldsymbol{u}_0$ as

$$\boldsymbol{u} = \boldsymbol{u}_t - \boldsymbol{u}_0, \tag{4.42}$$

where $\boldsymbol{u}_0$ is the displacement from the reference to the natural configuration and $\boldsymbol{u}_t$ is the displacement from the reference to the current configuration. Accordingly, under the assumption of small deformations, we have the following expression for the strain

$$\boldsymbol{\varepsilon}^{th} + \boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}_t - \boldsymbol{\varepsilon}_0, \tag{4.43}$$

where $\boldsymbol{\varepsilon}_t = \nabla^s \boldsymbol{u}_t$ and $\boldsymbol{\varepsilon}_0 = \nabla^s \boldsymbol{u}_0$.

Remark: It should be noted that the use of the *inactive element* method for handling material deposition is the reason of introducing stress free configurations. In order to make this clearer observe Figure 4.4. Suppose that every element in grey undergoes a motion because it is activated. After that motion, element E, that was not active, will have an initial deformation because the displacement associated to the two nodes shared with the activated elements is different from zero. Then, a natural configuration is built for that element by removing its initial deformation. It must be taken into account that this solution will be valid as the magnitude of the displacement field is smaller than the elements size. Another alternative for dealing with this previous limitation is using an *updated Lagrangian* formulation by additionally considering an algorithm for correctly position newly activated elements. This alternative will be studied in future work.

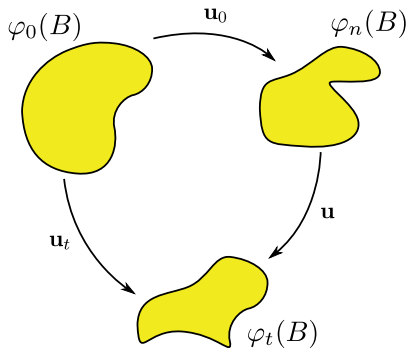


Figure 4.3: Reference, current and natural configurations, respectively denoted $\boldsymbol{\varphi}_0(B)$, $\boldsymbol{\varphi}_t(B)$ and $\boldsymbol{\varphi}_n(B)$.

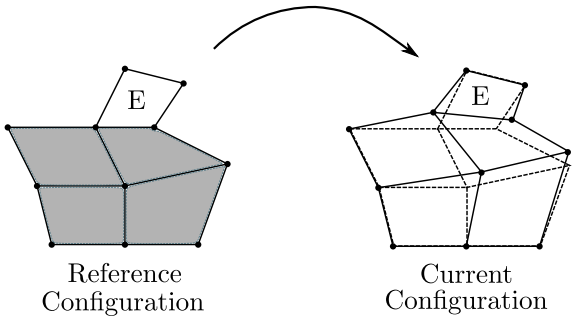


Figure 4.4: Inactive element approach and the need of stress free configurations.

4.4 Validation tests

Several tests are performed in order to validate the performance of the introduced thermo-mechanical numerical framework. In the first example, the problem of a solidifying slab, semi-analytically solved by Weiner and Boley [105], is adopted as reference problem in order to test the correct computation of thermal stresses of elasto-plastic bodies undergoing solid/liquid phase changes. Then, for assessing the correct implementation of elastic-perfectly plastic axisymmetric problems, the stresses in response to an internal pressure applied to a thick wall cylinder are numerically computed and compared against the analytic solution. Finally, at the end of this section, a test checking the correct implementation of the Voce hardening law is presented.

4.4.1 Weiner-Boley test

The Weiner-Boley test consists in the computation of thermal stresses of a semi-infinite elastic-perfectly plastic slab which is initially at melting temperature T_s and is suddenly cooled from one of its borders with a temperate drop $T_0 = T_s - T_w$, where $T_w < T_s$. The thermal problem consists in a Neumann phase change problem. In the mechanical problem, the material is assumed as elastic-perfectly plastic, with the yield stress σ_y varying linearly with temperature as

$$\sigma_y(T) = \sigma_y^0 \frac{T_s - T}{T_0}, \quad T \leq T_s,$$

and constant Young's modulus E and Poisson ratio ν. According to the solution of the thermal problem [106], the solidification front at time t is located at $\hat{x} = p\sqrt{t}$, where p is

given by

$$p = 2\sqrt{\alpha_\kappa}\gamma, \quad \alpha_\kappa = \frac{k}{\rho c_p} \quad \text{and} \quad \gamma \approx \sqrt{\frac{T_0 c_p}{2\mathcal{L}}} \quad \text{with} \quad \gamma^2 \ll 1,$$

with $\mathcal{L}$ denoting the latent heat, ρ the mass density, c_p the heat capacity and k the heat conductivity. By taking $\hat{x}$ as a characteristic length, the following dimensionless quantities are introduced

$$\bar{x} = \frac{x}{\hat{x}}, \quad \bar{T} = \frac{T - T_0}{T_0},$$
$$\bar{\sigma} = \frac{(1-\nu)\sigma}{\alpha_T E T_0}, \quad \bar{\sigma}_y = \frac{(1-\nu)\sigma_y}{\alpha_T E T_0} = -m\bar{T}, \quad m = \frac{(1-\nu)\sigma_y^0}{\alpha_T E T_0},$$

where α_T is the thermal expansion coefficient.

As stated by Weiner and Boley, if the slab is considered to solidify in the x direction, $\bar{\epsilon}_{yy} = \bar{\epsilon}_{zz} =$ constant in the transverse directions to x. In the three dimensional case, this is ensured by imposing extended plane strain conditions by constraining the y and z nodal displacements to be equal [18]. In the two dimensional case, a generalised plane strain state is adopted as kinematic hypothesis, thus allowing $\bar{\epsilon}_{zz}$ to be different from zero, and the condition $\bar{\epsilon}_{yy} =$ constant is imposed through the use of Lagrange multipliers [102].

The discrete problem was solved in a two dimensional slab whose dimensions are 2mm long and 0.67mm high. Q1-P0 quadrilaterals meeting the generalised plane strain kinematic hypothesis were used, dividing the slab geometry into one element in the y-direction and 400 elements in the x-direction. A time step equal to 0.01s was used to compute the evolution of thermal stresses in time. In Table 4.1 the material parameters used to run this test are shown. In Figure 4.5 a comparison between the computed numerical solution and the semi-analytic solution of Weiner and Boley at time $t = 0.47$s, can be observed. The adimensional parameters m and γ are equal to 0.097 and 0.465, respectively. The results show that the performance of the introduced thermo-mechanical framework is good enough to approximate thermal stresses in processes with liquid/solid phase change.

Table 4.1: Parameters of the Weiner-Boley test

$\mathcal{L}$	T_w	T_s	ρ	α_T
$272000\frac{\text{J}}{\text{kg}}$	1573K	1741K	$7400\frac{\text{kg}}{\text{m}^3}$	$7.6297 \times 10^{-5}\text{K}^{-1}$
k	c_p	E	ν	σ_y^0
$33\frac{\text{W}}{\text{Km}}$	$700\frac{\text{J}}{\text{Kkg}}$	210GPa	0.3	0.373GPa

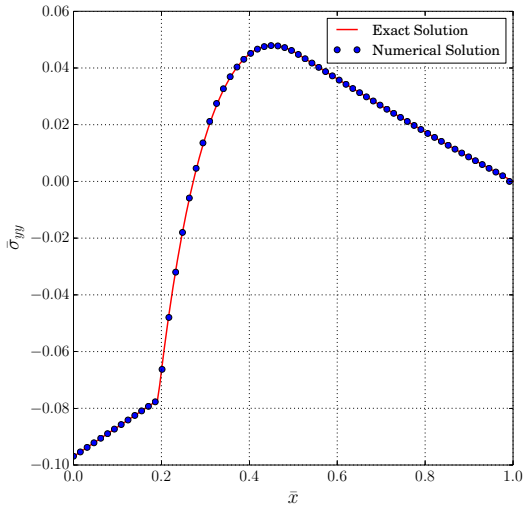


Figure 4.5: Results obtained for the Weiner-Boley test, $t = 0.47$s.

4.4.2 Elasto-plastic thick wall cylinder test

A pressurised elasto-plastic thick wall cylinder is numerically solved in order to test the correct implementation of elasto-plastic axisymmetric problems. The exact solution to this problem can be found in [95, 107]. The problem is solved using Q1-P0 quadrilaterals meeting the axisymmetric kinematic hypothesis.

The parameters used to run the test are specified in Table 4.2. A rectangular longitudinal section of the cylinder is modelled, whose dimensions are 1m in the axial direction and $r_e - r_i$ in the radial direction, where r_i and r_e are the internal and external radius, respectively. The radial direction is divided into 400 elements and the axial direction into 4 elements. The internal pressure is denoted by p_i, the Young's modulus by E, the Poisson ratio by ν and the yield stress by σ_y. The comparison between the exact and the numerical solution for the hoop stress and the radial stress is shown in Figure 4.6. From the obtained results, the good performance of the numerical scheme is concluded.

Table 4.2: Parameters of the Thick wall cylinder test

p_i	E	ν	σ_y	r_e	r_i
10kPa	21000kPa	0.49999	24kPa	1.5m	1m

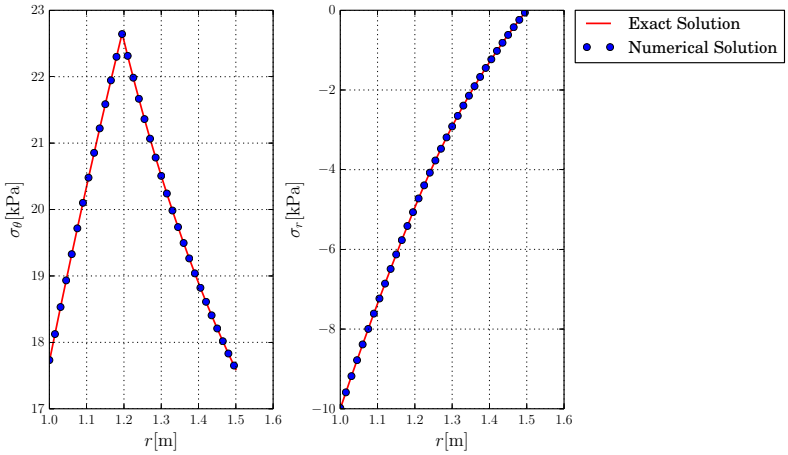


Figure 4.6: Thick wall cylinder test: exact and numerical solution comparison.

4.4.3 Voce hardening law implementation test

The implementation of the Voce hardening law, used for the description of isotropic hardening associated to elasto-plastic materials, is now tested. The problem to be solved is a uni-axially loaded bar. Then, the computed stress σ_{yy} along the longitudinal axis, when expressed in terms of the equivalent plastic strain α, must be equal to the curve describing the Voce hardening law, which uni-axially specifies the evolution of the Von Mises yield surface in terms of the equivalent plastic strain α.

Using symmetry, one fourth of the bar is modelled as an axisymmetric two dimensional solid by just one element. The dimensions of this section are 16mm in the axial direction and 4mm in the radial direction. A displacement of 4mm is imposed in the y-direction at the end of the bar and traction free boundary conditions are imposed at the other borders. The parameters used for the Voce hardening law are specified in Table 4.3. The Young's modulus and the Poisson ratio are taken equal to 210GPa and 0.29, respectively. The comparison between the numerical results and the expected solution can be observed in Figure 4.7, leading to good agreement and thus verifying the correct implementation of the Voce hardening law.

Table 4.3: Parameters of the Voce hardening law

k_0	k_1	δ	σ_y
907.8389MPa	173.1MPa	86.212	507.12MPa

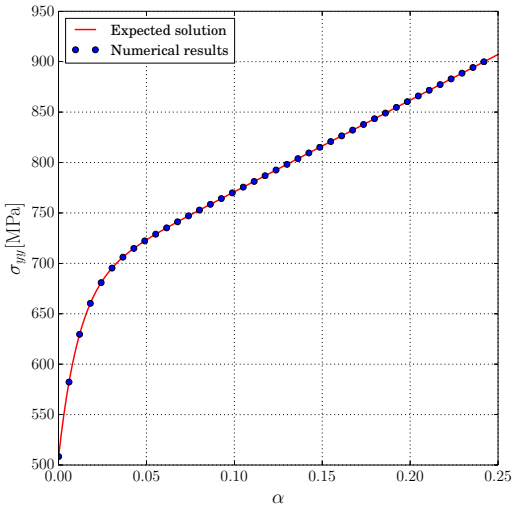


Figure 4.7: Test of the Voce hardening law: comparison of the numerical results and the exact solution.

4.5 Conclusions

The statement of the mechanical problem and its numerical treatment was revisited in this chapter from the point of view of welding and phase change problems. A thermo-mechanical framework able to handle material deposition and solid/liquid phase change was established by introducing the discrete version of the problem, and by considering the use of stress free configurations in order to correctly compute thermal stresses. An isotropic elasto-plastic hardening model was chosen for describing plastic flow, because, as observed by Mullins and Gunnars [99], this model best matches experimental measures of weld residual stresses. In this context, the small deformation kinematic hypothesis was adopted, although it is recognised that it may be inappropriate for describing some welding processes.

Different validation tests were investigated in order to assess the numerical performance of the presented tools. From the Weiner-Boley test, the high accuracy of the activation/deactivation of mechanical elements in conjunction with the use of stress free configurations was demonstrated. Additionally, the good results obtained with the implemented tools when solving the thick wall cylinder and the Voce hardening law tests, take to the conclusion that the introduced thermo-mechanical framework achieves a good performance. In future work, the use of an *updated Lagrangian* formulation by additionally considering an algorithm for correctly position newly activated elements will be studied.

Chapter 5

Parallel FE implementation

Multiphysics problems, such as welding problems, are computationally expensive applications. In order to reduce the computational complexity, some simplification hypotheses can be adopted in the formulation of these problems. Nonetheless, generally this is not enough and the number of computational resources required to solve them does not significantly decrease. In this chapter, a different alternative is followed in order to deal with this issue. Specifically, High Performance Computing (HPC) techniques are explored and implemented, towards obtaining more efficient numerical algorithms in the context of distributed shared memory architectures. Several performance tests are run to measure the effectiveness of the proposed parallel framework.

5.1 Introduction

Finite element computations of problems of technological importance are quite usually characterised by a large number of degrees of freedom (DOFs). One of the most computationally expensive problems is the Direct Numerical Simulation of turbulent flows [108]. In this kind of problem, the spatial and temporal scales must be considered when solving the Navier-Stokes equations directly, making necessary to use a large number of DOFs and a small time increment. Similarly, welding problems can be very time consuming because of its multiphysics character. In this context, HPC techniques represent an interesting alternative to deal with the high computational complexity exhibited by some problems.

Different solutions can be found in the realm of HPC. The one to be adopted depends upon the features of the available hardware and software. Nowadays, the characteristics of modern hardware makes necessary to become parallel in order to efficiently use the available resources. To understand this comment take into account that modern processors, even desktop processors, are built with more than four cores. Then, in order to take advantage of all the available hardware, programs must be coded making use of parallel programming models.

The memory in MIMD [1] (Multiple Instruction, Multiple Data) multiprocessor systems can be organised in two different manners [109]. There are systems where a single centralised memory is shared by every processor, an inherent feature of the so-called *centralised shared-memory architectures*. These systems are usually known as *Uniform Memory Access* (UMA) systems, because the memory latency is the same for every processor.

[1] In this work, only MIMD architectures are discussed.

From this point of view these multiprocessors are also known as *Symmetric MultiProcessors* (SMP).

Another manner of organising the memory in multiprocessor systems is by physically distributing all the memory, giving rise to the so-called *distributed memory architectures*. From the point of view of how the *address space* is organised, two different kind of distributed memory systems can be identified. If processors have its own *private* address space, the multiprocessors are usually referred to as *message-passing multiprocessors*. This is because in these systems data communication is usually accomplished through message-passing [109]. On the other hand, if the address space is the same for every processor, it is said that the address space is shared and, consequently, these multiprocessors are called *shared memory multiprocessors* or *distributed shared memory architectures*. In these systems the memory access is asymmetric, that is processors do not share the same memory latency. When this feature wants to be emphasised, the term *Non Uniform Memory Access* (NUMA) is used to refer to these kind of multiprocessors.

Message-passing multiprocessor architectures are more scalable than shared memory ones (UMA and NUMA systems), although the program to be run in the former must be written from scratch in order to satisfy some design requirements and keep pace with performance. On the other hand, shared memory architectures are more common to find, and pre-existing software can be more easily adapted to them without completely re-writing the program. As it was mentioned before, the fundamental difference between the shared memory architectures UMA and NUMA, resides in memory access latency. This is a very important parameter to be analysed when scalability is an issue. In the case of symmetric multiprocessor systems, the overall complexity increases as the number of processors does, making it very difficult to sustain the performance of the whole system. This is not the case for asymmetric shared memory architectures, because memory management does not increase considerably as processors are added to the system, thus ensuring the scalability of the architecture.

Much of the work presented in this thesis is implemented in the OOFELIE Multiphysics software [39], a finite element code written in C++ making use of the object oriented philosophy. This software was conceived to run sequentially in mono-processor machines, thus not taking advantage of the parallelism exhibited by the Finite Element Method (FEM). Specifically, the parallelism exposed by the computation of element contributions to corresponding global matrices or vectors, and even by the assembling of these structures, is not exploited. Additionally, the resulting system of the linear equations can be solved in parallel. In OOFELIE, this last point is already accomplished in parallel by making use of the routines provided by the Intel MKL library [110].

In order to exploit the parallelism exposed by Finite Element (FE) discretisations, this chapter will be focused on the parallelisation of OOFELIE by adopting a shared memory paradigm. Message-passing multiprocessors are not taken into account in this study, because the whole structure of OOFELIE would have to be re-thought and re-written. Additionally, based on the fact that on message-passing multiprocessors only iterative solvers are scalable, the problems to be solved would have to show certain properties, such as a well conditioning. The fulfilment of this property for the type of problems considered in this thesis is not an easy task. Clearly, more research is needed in this direction, which is out of the scope of the current work. In this thesis, a distributed shared memory architecture is adopted as point of departure. In order to make clearer the discussion of subsequent sections, a brief review of NUMA systems is next given.

An oversimplified representation of a NUMA system can be observed in Figure 5.1.

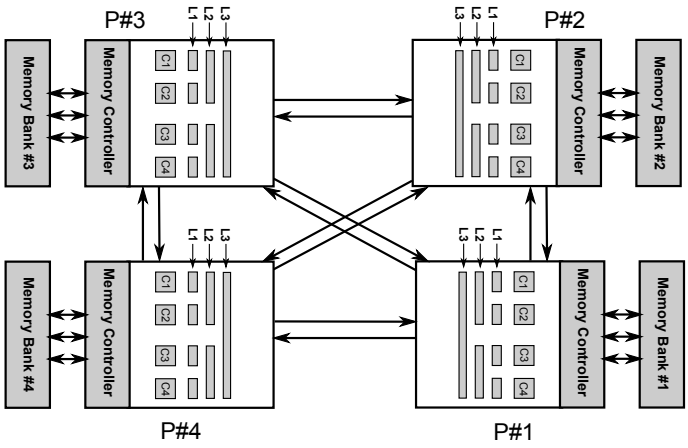


Figure 5.1: NUMA system comprised of four processors with four cores each. Each processor has a three-level cache hierarchy and a local memory bank interfaced through a local memory controller. For more details see [111].

The system is comprised of four processors with four cores in each one. Each processor has a three-level cache hierarchy, which at the same time can be correlated to a level of sharing, that is the L1 cache is assigned to only one core, the L2 cache is shared by two cores and the L3 cache is shared by every core in each processor die. In a NUMA system, the whole memory is physically distributed among processors in a set of memory banks. Each of these memory banks is interfaced to a given processor by means of a memory controller located at the processor die. In this way, all the memory in the system can be addressed by every processor, although in an asymmetric manner because the memory latency is smaller for access issued to memory banks local to the considered processor.

This design ensures the scalability of the system, although careful attention must be paid when coding applications targeting these kind of architectures. Specifically, if threads are used to process a given task, *processor and memory affinity* must be ensured in order to attain a good overall performance. In the last sentence, the term *processor affinity* was used to refer to the binding of threads to cores, and the term *memory affinity* was used to refer to the binding of a given processor to a specific memory bank. By means of processor affinity the number of cache-misses is reduced, and by means of memory affinity, memory access latency is controlled. Generally, processor affinity can be set by means of environment variables. Ensuring memory affinity is not an easy task when programming parallel applications in NUMA systems, because it is the work of the programmer to place the data that a given thread is probably to use in the memory bank closest to that thread.

Although in this work the solution is implemented by making use of a multi-threading programming model, specifically through the API OpenMP [112], it should be kept in mind that memory affinity is ensured automatically when adopting a *"message-passing paradigm"* such as the Message Passing Interface (MPI) standard. It must be noted that highly efficient implementations of MPI targeting distributed shared memory systems exist, such as the MPICH2 Nemesis [113, 114]. This very interesting path is not followed in this work because the design of OOFELIE does not allow to be treated in a distributed "message-passing" sense.

This chapter is organised as follows. In Section 5.2, a description of the problem in the context of FE nonlinear discretisations is given. Then, the proposed FE parallel framework is introduced, giving rise to the discussion of the problems posed by the thread-

safe processing of FE structures and the associated dual graph colouring. In Section 5.4, the performance obtained with the implemented solution is studied by running two tests in four machines with different specifications. Finally, conclusions are given.

5.2 Problem description

The discrete version of general nonlinear problems can be described by the following set of algebraic equations

$$\boldsymbol{\Pi}^{(i+1)} \simeq \boldsymbol{\Pi}^{(i)} + \frac{\partial \boldsymbol{\Pi}}{\partial \boldsymbol{U}}^{(i)} (\boldsymbol{U}^{(i+1)} - \boldsymbol{U}^{(i)}) = \boldsymbol{0}, \tag{5.1}$$

where the vector $\boldsymbol{U}$ denotes the unknown of the problem, $\boldsymbol{\Pi}$ is the associated residual function, and $\frac{\partial \boldsymbol{\Pi}}{\partial \boldsymbol{U}}$ the tangent matrix. This discrete formulation is generally obtained by considering the spatial and temporal discretisation of a given problem, and subsequently linearising it by making use of an iterative procedure such as a Newton-Raphson scheme. In this context, superscripts i and $i+1$ in equation (5.1) are used to denote the value of a given variable at iterations i and $i+1$, respectively.

Then, from a *global* point of view, a general nonlinear solver should deal with equation (5.1) at every time step of the time interval under analysis. In the FEM context, each time that the residual $\boldsymbol{\Pi}$ and tangent matrix $\frac{\partial \boldsymbol{\Pi}}{\partial \boldsymbol{U}}$ need to be evaluated, the contribution of each element is computed and assembled in the corresponding algebraic structure.

In OOFELIE, a distinction between *structural* and *non-structural* quantities is done, specifically applied to vectors and matrices. *Structural* quantities have the ability to "talk" to elements and thus initiating the assembling process of the corresponding quantity. A *non-structural* quantity does not have the ability of initiating the assembling process, but it implements the corresponding mathematical structure by means of arrays possessing a continuous memory layout. A continuous memory layout ensures that the components of a given quantity are accessed without indirectly addressing them, feature that is not met, for example, by *structural* vectors. In this context, the parallelisation of the assembling process deals with structural quantities, identified in OOFELIE by the classes `VectorStr` and `CSRMatrixStr`. It is supposed that the involved matrices are sparse, making use of the Compressed Sparse Row (CSR) format for storing its components. This matrix storage format specifies the data by means of three arrays [110]:

- `values`: array with non-zero entries of the matrix. The size of this array is equal to the number of non-zero entries.
- `columns`: element i of this array specifies the column corresponding to the value with index i in the array `value`. The size of this array is equal to the number of non-zero entries.
- `rowindex`: element i of this array specifies an index to array `values` identifying the first element of row i. The size of this array is number of rows + 1, where the last entry is equal to the number of rows if zero base is assumed.

It must be taken into account that in some situations, as that posed by staggered thermo-mechanical problems, the mesh of each problem type can be formed by different kind of elements, but sharing the same nodes that define the geometry. It is like if two different meshes were describing the same geometry, but each of them is used for a specific

problem type, e.g. one mesh for the thermal problem and another for the mechanical one. This feature must be taken into account in the subsequent sections when implementing the strategy for dividing the work among threads.

As it was already mentioned, the solution is implemented by making use of a multi-threading programming model, specifically through the API OpenMP. Some requirements must be satisfied in order to ensure reliability and performance of the coded solution, specifically:

- Strictly forbid, or at least avoid, the use of polymorphic methods in core operations.
- Avoid using static variables in order to ensure *thread-safe* implementations of methods.
- Modification of structures that are global in nature must be avoided in the implementation of elements, unless it is explicitly ensured that it involves a *thread-safe* operation.

It is said that a function is *thread-safe* "if it can be safely invoked by multiple threads at the same time" [115]. There are various methods to ensure that a given function is thread-safe. One of them is the use of *mutexs* or *locks* (*barriers* in general) in order to suppress any critical region, or, equivalently, any region of code in which thread-safety cannot be guaranteed. Another term that is used in what follows is the term *race-condition*, which denotes a non-deterministic task whose result depends on the order in which threads are executed.

5.3 Description of the FE parallel framework

The implementation of the FE parallel framework is based and adapted from the work of Farhat and Crivelli [38]. The objective is not only to compute in parallel the element contributions, but also to assemble them concurrently. In order to ensure thread-safety, these two operations should be carefully analysed. Specifically, the following comments apply:

- **Computation of element contributions**. This concerns to element methods which compute element contributions. In particular, it must be ensured that operations over global structures from these element functions are *read-only*, thus avoiding race-conditions when processing elements concurrently. In the FE context, this is easy to guarantee if the local support of FE structures makes possible to use local variables in the implementation of element functions.
- **Assembling of element contributions**. Ensuring that the assembling process is thread-safe is not so simple to verify as the previous requirement. We make the assumption that the support of each FE basis function extends only on a set of mutually adjacent elements, denoted by $\mathbb{S}_i$. In this context, it is understood that two elements are adjacent if they share at least a node. Then, if elements in $\mathbb{S}_i$ are processed concurrently, a race-condition arises because elements in $\mathbb{S}_i$ share at least one degree of freedom d. If different threads process elements sharing d, the race-condition arises when element contributions are added in the component/s of the structural matrix or vector corresponding to DOF d. In order to avoid this race-condition, a *lock* or *critical section* could be used. However, the parallel execution is serialised when this point is reached, with a loss in performance. It must

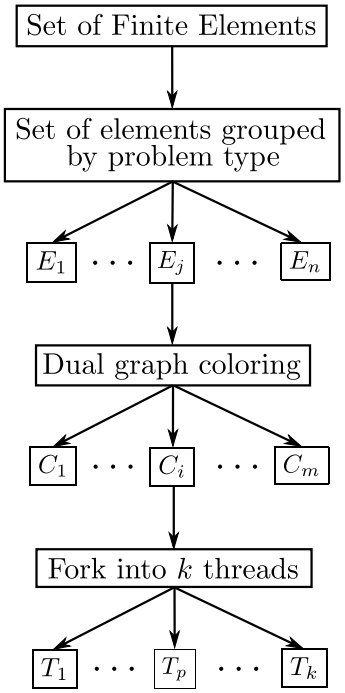


Figure 5.2: Schematic representation of the solution.

be remembered that it is known from the Amdahl's law that the improvement in performance is, in part, limited by the portion of code that is sequential [109]. To alleviate this limitation, critical regions are avoided by means of processing elements in a smart form as presented next.

In the interest of avoiding critical sections, the elements to be processed can be organised in groups of elements that do not share DOFs. In this manner, each group of elements can be processed in parallel ensuring thread-safety. In order to build these groups of elements, the dual graph G to the mesh is computed. The dual graph $G = \{V, E\}$, where V and E are the sets of vertexes and edges, respectively, is defined in the following manner: let each element of the mesh be identified with a vertex $v_i \in V$, then the edge $e_{ij} \in E$ is defined as

$$e_{ij} = \begin{cases} \{v_i, v_j\}, & \text{if elements } v_i \text{ and } v_j \text{ are adjacent} \\ \emptyset, & \text{if elements } v_i \text{ and } v_j \text{ are not adjacent} \end{cases} \tag{5.2}$$

where $\emptyset$ is the empty set.

Then, once the dual graph G is computed, it is coloured. Graph colouring consists in assigning a colour to every vertex in the graph such that adjacent vertexes have different colours [116]. Then, the complete set of elements is partitioned into different groups of elements sharing the same colour, ensuring that these elements are not adjacent and do not have any DOF in common. Therefore, thread-safety is guaranteed and each set can be processed in parallel.

The overall logic used for implementing the parallel assembling of structural vectors and matrices, along with the computation of the corresponding element contributions, can be observed in Figure 5.2. In order to be general enough and to consider weakly coupled problems which are characterised by meshes of different elements that share the

same nodes, the whole set of finite elements is first subdivided into a number of subsets, E_i, according to the problem type associated to each element. Then, the dual graph G_i of each of these sets E_i is computed and coloured, resulting into a number of subsets C_i which can be processed concurrently ensuring thread-safety. The computations to be done for the processing of each colour set C_i, are subdivided into T_k mutually exclusive tasks assigned to different threads. In the sections that follow, the problem posed by the graph colouring and the details of the implementation of the parallel framework are given.

5.3.1 The graph colouring problem

The computation of the dual graphs of the FE meshes (as many as problem types) is accomplished by making use of the METIS library [117]. These graphs are considered to be sparse, therefore they are stored by making use of the Compressed Storage Format (CSR). In this context, let a given graph G to have n vertexes and m edges, then G is stored making use of two arrays, denoted by `xadj` and `adjncy`, whose sizes are $n+1$ and $2m$, respectively. By assuming zero base, the adjacency list of vertex i is given by the array that spans from `adjncy[xadj[i]]` to `adjncy[xadj[i+1]-1]`, included [117].

Once the dual graph is computed, its colouring is undertaken. The following features are desirable in the obtained colouring:

A - Minimum number of colours in order to obtain the maximum performance: in this way more parallelism is exposed by the data to be processed.

B - Minimum number of resources and time to compute the colouring.

It is known that a graph colouring meeting feature A is an NP-HARD problem. Satisfaction of feature B is not possible in this context, then some compromise must be taken by relaxing both requirements. A greedy algorithm is used in order to obtain an approximate solution (in the sense that the resulting number of colours does not necessarily coincide with the chromatic number of the graph[2]) in a reasonable time. The greedy algorithm used to colour the graph is given in Algorithm 4. Each colour is identified with an integer starting from 0. Then, at the beginning of the algorithm the colour 0 is assigned to the first vertex (`node` in the code). Following, it loops over the remaining vertexes. At iteration k, the minimum colour number that can be assigned to vertex k is computed and correspondingly assigned by taking into account the already coloured elements.

This algorithm works quite well for sparse graphs. In other words, this algorithm performs well if the average number of adjacencies $\bar{n}_a$ that an element from the mesh can have is much smaller than the number of elements n, that is if $\bar{n}_a \ll n$. Then, the complexity of the proposed algorithm is $O(n)$, as $\bar{n}_a$ can be considered to be constant and independent of n. It must be noted that this analysis implies that the number of colours will be independent of n.

A test was run in order to evaluate the complexity of the algorithm [118]. The test consists in a two dimensional squared domain discretised with triangular elements taking d_e divisions for the directions x and y, resulting in d_e^2 nodes and $2(d_e-1)^2$ elements. The obtained results are shown in Figure 5.3, from which it can be effectively observed that the complexity of the algorithm is $O(n)$, and that the number of colours can be considered to be independent of n.

[2]The exact minimum number of colours needed to colour the graph.

Code 4 Greedy Graph Colouring Algorithm

```
void greedyGraphColoring(vector< vector<int> > &colorTable){

    vector<int> tablaColor(verticesNumber, -1);
    int node = 0, minCol = 0, cj;

    colorTable.push_back(vector<int>());
    colorTable[minCol].push_back(node);

    tablaColor[node] = minCol;
    vector<int> alCol;
    for (node=1; node<verticesNumber; node++){

        minCol = 0;
        alCol.clear();
        for(int k=xadj[node];k<xadj[node+1];k++){
            cj = tablaColor[adjncy[k]];
            if (cj >=0)
                alCol.push_back(cj);
        }

        //sort colors of adjacent elements
        sort(alCol.begin(), alCol.end());
        for(int i=0;i<alCol.size();i++)
            if (minCol==alCol[i])
                minCol++;

        tablaColor[node] = minCol;

        if (colorTable.size() <= minCol)
            colorTable.push_back(vector<int>());

        colorTable[minCol].push_back(node);

    }
}
```

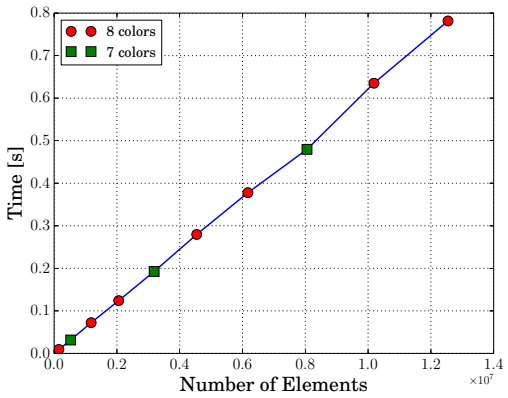


Figure 5.3: Graph colouring test.

5.3.2 Details of the parallel framework implementation

At the beginning of this section the overall logic behind the FEM parallel framework was presented, where the basic steps were summarised in Figure 5.2. Although not explicitly mentioned, it was assumed that every structure to be modified concurrently was pre-allocated. Then, from a global perspective, the adopted strategy reads as following:

1. Pre-allocate the involved global structures, such as (structural) matrices and vectors.
2. Organise the elements to be processed in such a way that critical sections are avoided: compute the dual graph associated to each problem type and colour them.
3. Process each colour in parallel.

Pre-allocating structural vectors is not a problem, because their dimensions are known, more specifically, their dimensions are equal to the number of DOFs. In the case of structural matrices, the issue is more complicated because the profile of each matrix to be assembled is not a-priori known. For example, there are some nonlinear problems in which element activation and deactivation depends on the value of a field being calculated, situation which prevents from knowing which elements are going to contribute, leading to a variable matrix profile.

In order to implement this strategy, two operations are defined: the `dryUpdate` and the `update` operations, where the first one computes the profile of the matrix, and the second assembles and computes the element contributions in parallel. That is, the `dryUpdate` operation implements point 1 of the enumerate, and the `update` operation implements point 3.

The `dryUpdate` operation consists in determining the components of the global matrix that are hit by element contributions. This is done by sequentially visiting every element without computing their actual contributions, but computing the components of the matrix where they contribute.

5.4 Testing the implemented solution

In order to test the parallel implementation presented in the preceding sections, two tests were performed: a simple two dimensional linear transient heat conduction problem implemented in an iterative manner (a Newton like solver) making use of 500000 triangular

elements and 251001 DOFs; and a three dimensional non-isothermal phase change problem making use of 192000 tetrahedral elements and 44541 DOFs. The speedup and the efficiency of the tangent matrix assembling and of the MKL Pardiso solver were measured in four different machines whose specifications are given below.

MachineA: Intel i7

- Processor: Intel(R) Core(TM) i7 CPU 930 @ 2.80GHz
- Number of physical cores: 4. Hyperthreading deactivated.
- Cache L1, L2, L3: 32KB, 256 KB, 8MB.
- Memory: 6GB
- Operative system: `Linux 2.6.34.10-0.6 #1 SMP PREEMPT x86_64 GNU/Linux`

MachineB: Intel Xeon X5680

- Processor: 2 x [Intel(R) Xeon(R) CPU X5680 @ 3.33GHz]
- Number of physical cores: 6 cores on each processor. Hyperthreading deactivated.
- Cache L1, L2, L3: 32KB, 256 KB, 12MB.
- NUMA nodes #0, #1: 2 x 48GB
- Operating system: `Linux 3.5.1-1.fc17.x86_64 #1 SMP x86_64 GNU/Linux`

MachineC: Intel Xeon E5620

- Processor: 2 x [Intel(R) Xeon(R) CPU X5620 @ 2.4GHz]
- Number of physical cores: 4 cores on each processor. Hyperthreading deactivated.
- Cache L1, L2, L3: 32KB, 256 KB, 12MB.
- NUMA nodes #0, #1: 2 x 8GB
- Operating system: `Linux 2.6.35.6-45.fc14.x86_64 #1 SMP x86_64 GNU/Linux`

MachineD: Intel Xeon E5430

- Processor: 2 x [Intel(R) Xeon(R) CPU E5430 @ 2.66GHz]
- Number of physical cores: 4 cores on each processor. Hyperthreading deactivated.
- Cache L1, L2: 32KB, 6 MB.
- Memory: 32GB (Symmetric Memory Access / FSB based).
- Operating system: `Linux 3.5.4-2.fc17.x86_64 #1 SMP x86_64 GNU/Linux`

Machines `MachineB` and `MachineC` are NUMA systems, differing from `MachineD` that has a single centralised memory with two (SMP) processors. The code OOFELIE is compiled making use of the Intel C++ compiler. Then, the examples are run setting the environment variable `KMP_AFFINITY` to `KMP_AFFINITY=verbose,compact`, in order to ensure processor affinity at the granularity of cores.

The speedup and efficiency for the tangent matrix assembling for the two problems and for the different machines can be observed in Figures 5.4 and 5.5. `MachineA` and `MachineC` have a good performance, attaining almost the same speedup up to 4 threads. In `MachineC`, when the number of threads is greater than 4, NUMA misses arise and the speedup decays due to the latency increment related to the communication between

NUMA nodes. In the case of `MachineB`, the speedup has a different slope to the expected one, with a loss of efficiency. The observation that the microarchitecture of `MachineA` is Nehalem, and the microarchitecture of `MachineB` and `MachineC` is Westmere (improved Nehalem), leads to think that the reason of the behaviour of `MachineB` is a malfunction that it is not related to the parallel implementation when taking into account that good results are obtained in the case of `MachineC`.

In the case of `MachineD`, it can be observed that its performance is quite bad. The main factor explaining this behaviour is the symmetric access of memory and its implementation by making use of the Front Side Bus (FSB) technology, which attains up to a maximum of 1.6 GT/s. In order to make some comparisons with the results obtained with the other machines, take into account that memory access is asymmetric in these machines and it is implemented making use of the Quick Path Interconnect (QPI) technology which can attain up to 6.4-4.8 GT/s [111].

In Figures 5.6 and 5.7 the speedup and efficiency for the MKL Pardiso solver for the two problems and for the different machines can be observed. In these cases the obtained efficiency and speedup up to 4 threads for the different machines are not so different as for the matrix assembly operation. However, the efficiency for the MKL Pardiso solver is quite far from 1. A possible explanation of these results, is the fact that in the case of the matrix factorisation the access to memory is more aggressive than in the case of the matrix assembling, affecting the scalability of the operation. For completeness, in Figure 5.8 a comparison of the wall clock times for the matrix assembling and the MKL Pardiso solver in the different machines using 4 threads is given.

Remark: it should be taken into account that the reduction in the total time required by the simulation will be controlled by the solution of the system of linear equations as the problem size grows. For example, in the 2D example (251001 DOFs) the time needed for solving the system is almost the same to the time required for assembling the tangent matrix. Despite this, a total speedup per iteration of 2.52 was measured for this test when using 4 threads in the case of **MachineA**. However, it should be kept in mind that as the problem size grows the efficiency of the direct solver used for solving the system of linear equations will decrease and so the total speedup. It can be roughly said that if the size of the problem is bigger than 400000 DOFs, a message-passing paradigm and the use of iterative solvers should be adopted. In our case the adopted parallelisation scheme is enough for the problems that we are considering; quite frequently, we need to solve transient problems characterised by a large number of time steps ($\simeq$ 12000 in some cases), but with few DOFs ($<$ 150000).

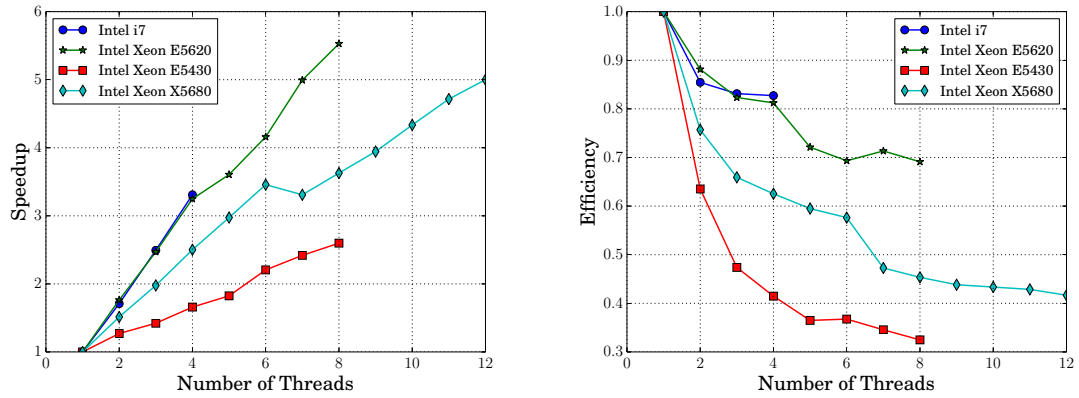


Figure 5.4: Speedup and efficiency for the matrix assembling for a 3D non-isothermal phase change problem (tetrahedral elements).

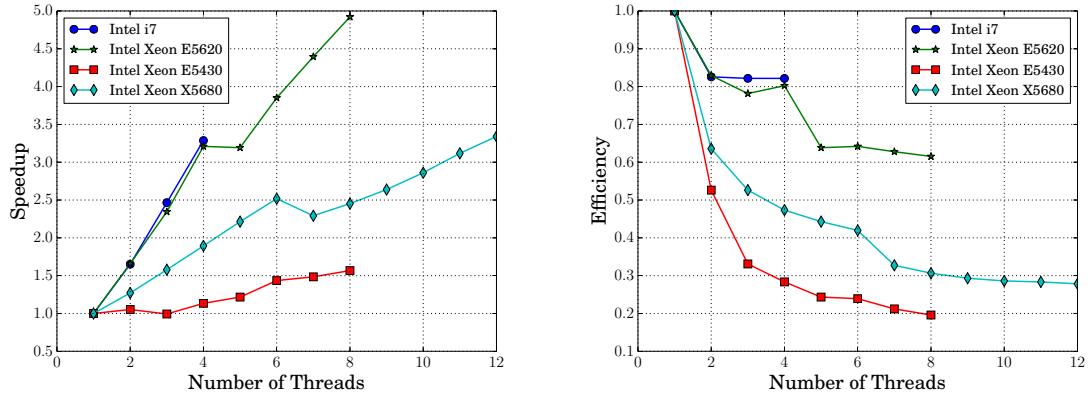


Figure 5.5: Speedup and efficiency for the matrix assembling for a 2D linear heat conduction problem (triangular elements).

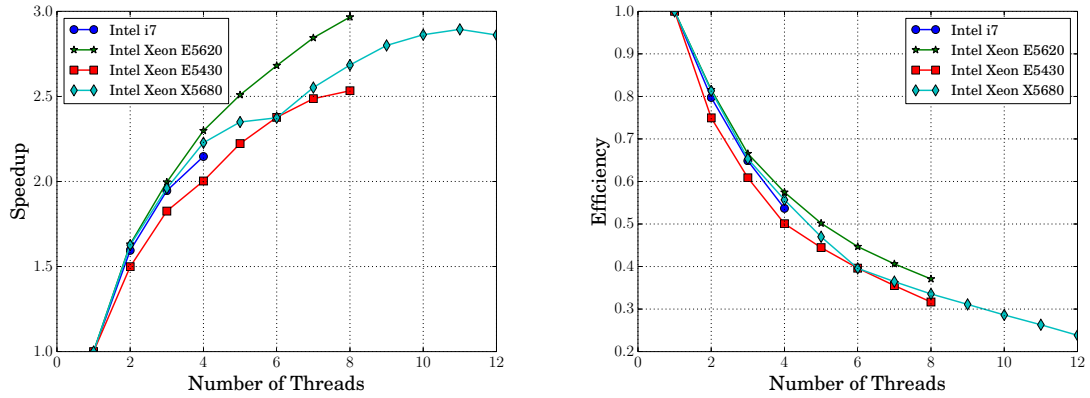


Figure 5.6: Speedup and efficiency of the MKL Pardiso for a 3D non-isothermal phase change problem (tetrahedral elements).

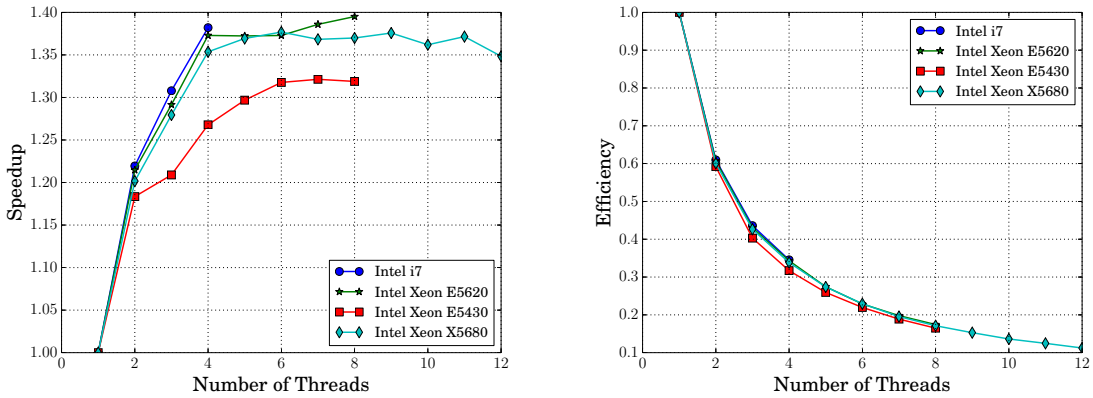
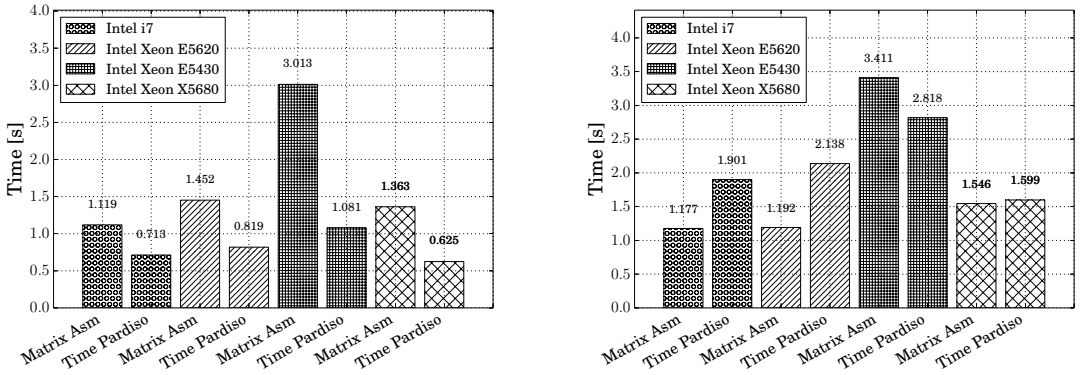


Figure 5.7: Speedup and efficiency of the MKL Pardiso for a 2D linear heat conduction problem (triangular elements).



(a) Times for the 3D problem (192000 tetrahedral elements).

(b) Times for the 2D problem (half a million triangular elements).

Figure 5.8: Wall clock time comparison for the tested problems. Data obtained using 4 threads in all cases.

5.5 Conclusions

The problem posed by the parallelisation of FE computations was addressed. In particular shared memory architectures were considered, because the design of the pre-exiting software to be parallelised (OOFELIE) is not well-suited for message-passing multiprocessor architectures. The concept of *thread-safety* was introduced and taken as feature to be guaranteed in the proposed parallel framework. The requirement of computing the element contributions and assemble them in parallel, is satisfied by means of computing the dual graph of the mesh of each problem type. Then, these graphs are coloured in order to build groups of elements that can be concurrently assembled in a thread-safe manner, avoiding critical sections and the use of locks or barriers that serialise the parallel execution of the code. The graph colouring problem, an NP-HARD problem, was efficiently solved by making use of a greedy strategy.

Two different tests were performed in different machines in order to study the performance of the implemented parallel framework. As expected, good performance was observed for the processor `i7` (Nehalem) and for the processor `Xeon E5620` (Westmere). Results exposing bad performance were obtained in the case of the processor `Xeon X5680`, which are probably related to a malfunction of the machine as good results were observed in the case of the processor `Xeon E5620` which has a similar microarchitecture. Additionally, these tests were run making use of a machine with a single centralised memory with two (SMP) processors `Xeon E5430`. The results obtained with this machine showed a bad performance, in part due to the symmetric access of memory and the use of the FSB technology.

Chapter 6

Reduced Order Models for the thermal problem

The simulation of engineering problems is quite often a complex task that can be very time consuming. In this context, the use of Hyper Reduced Order Models (HROMs) is a promising alternative for real-time simulations. In this work, we study the design of HROMs for non-linear problems with a moving source. Applications to nonlinear phase change problems with temperature dependent thermophysical properties are particularly considered; however, the techniques developed can be applied in other fields as well.

A basic assumption in the design of HROMs is that the quantities that will be hyper-reduced are *k-compressible* in a certain basis in the sense that these quantities have at most k non-zero significant entries when expressed in terms of that basis. To reach the computational speed required for a real-time application, k must be small. We examine different strategies for addressing hyper-reduction of the nonlinear terms with the objective of obtaining *k-compressible* signals with a notably small k. To improve performance and robustness, it is proposed that the different contributing terms to the residual are separately hyper-reduced. Additionally, the use of moving reference frames is proposed to simulate and hyper-reduce cases that contain moving heat sources. Two application examples are presented: the solidification of a cube in which no heat source is present and the welding of a tube in which the problem posed by a moving heat source is analysed.

6.1 Introduction

Modelling problems that are intractable from the point of view of reaching real time computational speed are quite frequently found in science and in engineering. Two particularly time-consuming cases are the problems of welding and nonlinear phase change. This paper presents hyper-reduction methods for these two problems, but the methods developed can be extended to many other problems with similar characteristics.

A welding problem is essentially a thermally driven process. Due to this fact, a correct description of the heat source that represents the energy input is of great importance. Generally, this input is described in terms of a standardised and highly concentrated moving heat source. This feature sets up a problem whose main characteristics are rather rapid changes in the involved fields as well as the rapid variation of material properties. The concentrated behaviour of the heat source introduces constraints in the size of the mesh and time step used to run the simulation, thus forcing an increment in the number

of degrees of freedom of the problem.

Simulation of a welding problem is a highly complex task described as a Thermo-Mechanical-Metallurgical (TMM) process [1]. To reduce this complexity, certain simplifications are applied, such as describing the governing physics of the problem with a staggered thermo-mechanical model [19]. Despite efforts to render treatment of the problem more amenable, its complexity makes it unaffordable from the perspective of real-time simulations. In this context, the design of Reduced Order Models (ROMs) is an elegant and promising alternative to the classical high fidelity solutions.

Currently, the use of separate representations to build ROMs has caught the attention of the engineering community. The manner in which the separated representation is built is described by two approaches [32]. In one approach, *a posteriori* model reduction techniques require knowledge of the solution to a training problem. The most prominent *a posteriori* technique is based on the Proper Orthogonal Decomposition (POD) method [33, 34]. For the second approach, the *a priori* model reduction techniques require no previous knowledge of the solution, concept which was introduced by Ryckelynck in [119]. In this context, the leading technique is the Proper Generalised Decomposition (PGD), which has its roots in the works of Ladevèze [35, 36].

In both approaches, to successfully address the high dimensionality of the problem, it is assumed that the solution can be described in terms of a reduced number of functions in the separate representation. In a certain sense, this situation can be referred to as the *separate representation hypothesis*. A consequence of this idea is the supposition that the solution of the problem is *k-compressible* in a certain basis, with a notably small k. The solution is said to be *k-compressible* if it has at most k non-zero significant entries when expressed in terms of that *magic* basis. An extensive analysis of similar concepts is offered in the context of compressed sensing, see, for example [120, 121].

The present work considers the *a posteriori* ROM technique based on the POD method. The reduction of the problem begins by reducing the dimensionality of the discrete versions of the test and trial spaces. This process is generally carried out by finding a basis, say $\boldsymbol{\Theta}$, in which the solution to the problem is *k-compressible*, which is obtained by computing the Singular Value Decomposition of a set of snapshots of the solution. Next, the trial solution space is defined as an affine translation of $span\{\boldsymbol{\Theta}\}$. If a Bubnov-Galerkin projection is used, the test functions are in $span\{\boldsymbol{\Theta}\}$. It must be mentioned that a Petrov-Galerkin projection is recommended for problems where the Jacobian is not Symmetric Positive Definite (SPD), which involves the solution of a least squares problem [122].

After reducing the dimensionality of the test and trial spaces, the size of the system of linear equations to be solved is reduced from a size of $N \times N$, where N is the size of the High Fidelity (HF) model, to a size of $k \times k$, with $k \ll N$. With this approach, although the computational cost of solving the system of linear equations is reduced, the cost of assembling the residual and the tangent matrix at each Newton iteration is still of order N. It is widely known that to significantly reduce the computational complexity of the problem, the cost of assembling the residual and the tangent matrix at each Newton iteration must be reduced [123, 124, 122, 125, 126, 127]. To accomplish this objective, a second reduction is performed by evaluating the involved quantities in few points of the domain. It can be found extensions of this idea in the context of *a priori* and *a posteriori* reduction methodologies. For instance, in [128] the extension to nonlinear Finite Element models making use of an *a priori* approach is presented and the term hyper-reduction is coined to refer to the general procedure of performing a second reduction. Another work following this line is that of Sarbandi *et al.* [129].

In the context of *a posteriori* reduction techniques, the hyper-reduction method can be found to be applied widely. Generally, the ideas following this path are based on the gappy data reconstruction method introduced by Everson and Sirovich [130] in the image processing community. For example, an extension of this idea to Finite Volume equations was accomplished by Astrid [131], extensions to nonlinear mechanical models are found in [132, 124] and, in the case of computational fluid dynamics, treatments of this kind are found in [126, 125]. We use the term Hyper-Reduced Order Models (HROMs) to refer to the reduced models arising from the hyper-reduction method. To the author' best knowledge, the design and application of HROMs specifically suited for phase change and welding problems have not yet been addressed.

Different approaches are studied in detail for the design of HROMs with particular application to the nonlinear phase change problem. Schemes in which the residual is hyper-reduced as a unit, taking the history of the residual as snapshots for the gappy data reconstruction procedure, are usually found in the literature [122, 125]. This strategy is used as a reference technique for comparison. As observed from numerical experiments, poor *k-compressibility* and tangent matrix conditioning are obtained when applying this technique for the design of HROMs in the considered problems. To improve the performance and robustness, it is proposed that the different contributing terms to the residual are separately hyper-reduced. These terms are assumed to be physically based nonlinear generalised contributing forces, features that lead to a well-posed HROM. In the case of welding problems, the moving heat source represents an issue that can severely affect the *k-compressibility* of the involved terms. This complication is addressed by considering both moving and fixed frames of reference respect to the welded piece.

This chapter is organised as follows. Section 6.2 states the mathematical formulation of the solid-liquid phase change problem, and Section 6.3 presents the formulation of Reduced Order Models. The cost of assembling the nonlinear forces and tangent matrices is reduced by means of Hyper Reduced Order Models in Section 6.4, significantly reducing the computational complexity of the problem. In Section 6.5, the issue presented by the moving heat source is considered, and two application examples are presented in Section 6.6 to assess the performance of the introduced HROMs. Specifically, the solidification of a cube and welding of a tube without material deposition are analysed. Finally, Section 6.7 describes the main conclusions of this chaper.

6.2 Problem setting

The physical problem under consideration is a nonlinear transient heat conduction problem in which the liquid-solid phase change and thermophysical properties that depend on temperature are taken into account. The mathematical setting of this problem was already introduced in Chapter 2, but it is briefly revisited in this section for the sake of understanding. Assuming that the contribution of the mechanical energy to the total energy is negligible and considering the specific enthalpy $\mathcal{H}$ as a thermodynamic potential, the temperature field T is computed by solving the heat balance equation

$$\rho\dot{\mathcal{H}} = Q + \nabla\cdot(k\nabla T) \qquad \forall\ (\boldsymbol{x}, t) \in \Omega_i \times (t_0, \infty) \tag{6.1}$$

where ρ is the density, k is the thermal conductivity, T is the temperature, Q is the external heat source per unit volume, and Ω_i for $i \in [s, l]$ are the solid and liquid sub-domains with $\Omega_s \cap \Omega_l = \{\emptyset\}$ and $\Omega = \Omega_s \cup \Omega_l$. The temperature field should verify the initial conditions

$$T = T_0 \qquad \forall\ \boldsymbol{x} \in \Omega,\ t = t_0 \tag{6.2}$$

where $T_0(\boldsymbol{x})$ is the given initial temperature field. Additionally, the following set of conditions must be verified at the disjoint portions $\Gamma_d, \Gamma_q, \Gamma_c$ of the external boundary

$$T = T_d \quad \forall\ (\boldsymbol{x}, t) \in \Gamma_d \times (t_0, \infty) \tag{6.3}$$

$$k\nabla T \cdot \boldsymbol{n} = q_w \quad \forall\ (\boldsymbol{x}, t) \in \Gamma_q \times (t_0, \infty) \tag{6.4}$$

$$k\nabla T \cdot \boldsymbol{n} = h_f(T_f - T)\ \forall (\boldsymbol{x}, t) \in \Gamma_c \times (t_0, \infty) \tag{6.5}$$

where $\Gamma_d \cup \Gamma_q \cup \Gamma_c = \partial\Omega$, T_d is the imposed temperature at the boundary Γ_d, q_w is the external heat flow at the boundary Γ_q, h_f is the heat convection coefficient, T_f is the external fluid temperature at the portion the boundary Γ_c and $\boldsymbol{n}$ is the outward normal to the boundary under consideration. Finally, at the interface Γ between Ω_s and Ω_l (the phase change boundary), the following constraints must hold

$$T = T_m \qquad \forall\ (\boldsymbol{x}, t) \in \Gamma \times (t_0, \infty) \tag{6.6}$$

$$[-(k\nabla T) \cdot \boldsymbol{n}_\Gamma]_\Gamma = \rho\mathcal{L}u_\Gamma \qquad \forall\ (\boldsymbol{x}, t) \in \Gamma \times (t_0, \infty). \tag{6.7}$$

In these equations, $\mathcal{L}$ is the latent heat, T_m is the melting temperature, $\boldsymbol{n}_\Gamma$ is the outward normal to the solidification front from the solid domain, $u_\Gamma = \boldsymbol{u}_\Gamma \cdot \boldsymbol{n}_\Gamma$ is the velocity of the interface in the direction of the normal $\boldsymbol{n}_\Gamma$ and the operator $[\centerdot]_\Gamma$ measures the jump of the quantity $\centerdot$ at the solidification front. Equation (6.6) imposes the constraint that the temperature at the phase change boundary must be equal to the melting temperature and equation (6.7) is the interface condition (the Stefan condition).

The specific enthalpy $\mathcal{H}$ can be expressed in terms of the temperature T as

$$\mathcal{H}(T) = \int_{T_{\text{ref}}}^{T} c(\tau)d\tau + \mathcal{L}f_l(T) \tag{6.8}$$

where T_{ref} is a reference temperature, $c(\tau) \equiv c$ is the heat capacity and $f_l(T)$ is the liquid fraction. For an isothermal phase change, the liquid fraction is expressed as a Heaviside step, i.e., $f_l = H_{eav}(T - T_m)$. In the case of non-isothermal phase change, the liquid fraction can be described in terms of a linear function of temperature with solidus temperature T_{sol} and liquidus temperature T_{liq} as parameters and given by

$$f_l(T) = \begin{cases} 1 & \text{if } T > T_{liq} \\ \dfrac{T - T_{sol}}{T_{liq} - T_{sol}} & \text{if } T_{sol} \leq T \leq T_{liq} \\ 0 & \text{if } T < T_{sol} \end{cases}. \tag{6.9}$$

The energy input to the medium is described in terms of the Goldak heat source previously introduced by equation (3.45) in Chapter 3.

6.2.1 Variational formulation and finite element discretisation

The variational formulation of the problem and its finite element discretisation is briefly described in this section. Further details can be found in references [28, 85]. Let $\mathcal{S} = \{T/T \in \mathcal{H}^1(\Omega),\ T|_{\Gamma_d} = T_d\}$ be the space of trial solutions and $\mathcal{V} = \{v/v \in \mathcal{H}^1(\Omega),\ v|_{\Gamma_d} =$

$0\}$ be the space of weighting or test functions, where $\mathcal{H}^1$ is the first order Sobolev space. Next, the variational formulation is given as follows:
Find $T \in \mathcal{S}$ *such that* $\forall w \in \mathcal{V}$

$$\int_\Omega w\left[\rho c\frac{\partial T}{\partial t} + \rho\mathcal{L}\frac{\partial f_l}{\partial t} - Q\right]\, d\Omega + \int_\Omega \nabla w \cdot (k\nabla T)\, d\Omega + \int_{\Gamma_c} w h_f (T - T_f)\, d\Gamma + \int_{\Gamma_q} w q_w\, d\Gamma = 0\,, \qquad \text{for } t > 0;$$

$$\int_\Omega wT\, d\Omega = \int_\Omega wT_0\, d\Omega\,, \qquad \text{for } t = 0. \tag{6.10}$$

We remark that in the case of isothermal phase change, the time derivative of the liquid fraction $\frac{\partial f_l}{\partial t}$ should be interpreted in a distributional sense.

Let $\mathcal{S}^h \subset \mathcal{S}$ and $\mathcal{V}^h \subset \mathcal{V}$ be subspaces of the trial and test functional spaces, respectively. Therefore, in matrix notation, $T^h \in \mathcal{V}^h$ is given by

$$T^h(\boldsymbol{x}, t_n) = \boldsymbol{N}^T \boldsymbol{T}_n, \tag{6.11}$$

where $\boldsymbol{N}$ denotes the finite element basis, such that $\mathbb{R}^N \in span\{\boldsymbol{N}\}$, and $\boldsymbol{T}_n \in \mathbb{R}^N$ denotes the FEM (Finite Element Method) nodal degrees of freedom, with N as the dimension of the FEM space. Linear shape functions are used in this work. Next, using a Bubnov-Galerkin projection and a Backward-Euler scheme for time integration, the discrete form of the residual of the nonlinear thermal problem reads

$$\boldsymbol{\Pi} = \boldsymbol{G}_c + \boldsymbol{G}_k + \boldsymbol{G}_l + \boldsymbol{F} - \boldsymbol{Q} = \boldsymbol{0}, \tag{6.12}$$

where

$$\boldsymbol{G}_c = \int_\Omega \rho c_n \boldsymbol{N}\boldsymbol{N}^T\, d\Omega\, \frac{\boldsymbol{T}_n - \boldsymbol{T}_{n-1}}{\Delta t}, \tag{6.13}$$

$$\boldsymbol{G}_k = \Big(\int_\Omega \nabla\boldsymbol{N} k_n \nabla\boldsymbol{N}^T \Omega + \int_{\Gamma_c} h_{f_n}\boldsymbol{N}\boldsymbol{N}^T\, d\Gamma\Big)\, \boldsymbol{T}_n, \tag{6.14}$$

$$\boldsymbol{G}_l = \frac{1}{\Delta t}\Big(\int_\Omega \rho\mathcal{L}\boldsymbol{N} f_{l(n)}\, d\Omega - \int_\Omega \rho\mathcal{L}\boldsymbol{N} f_{l(n-1)}\, d\Omega\Big), \tag{6.15}$$

$$\boldsymbol{F} = \int_{\Gamma_q} \boldsymbol{N} q_{w_n}\, d\Gamma - \int_{\Gamma_c} h_{f_n}\boldsymbol{N} T_{f_n}\, d\Gamma, \tag{6.16}$$

$$\boldsymbol{Q} = \int_\Omega \boldsymbol{N} Q_n\, d\Omega. \tag{6.17}$$

In what follows the discussion will be directed towards obtaining highly *compressible* signals in the adopted formulation. That is why we present a variation of the formulation given by equation (6.12), that will be advantageous for reasons given more fully below. Each term in equation (6.12) is kept unmodified, except for the terms $\boldsymbol{G}_c$ and $\boldsymbol{G}_l$, which involve discrete time derivatives and represent reactions in response to temperature increments. Equation (6.12) is reformulated by re-writing these terms as reaction increments during the time step. In this case, the reformulated problem reads

$$\boldsymbol{\Pi} = \frac{\boldsymbol{G}_n^c - \boldsymbol{G}_{n-1}^c}{\Delta t} + \boldsymbol{G}_k + \frac{\boldsymbol{G}_n^l - \boldsymbol{G}_{n-1}^l}{\Delta t} + \boldsymbol{F} - \boldsymbol{Q} = 0, \tag{6.18}$$

where

$$\boldsymbol{G}_n^c = \int_\Omega \rho c_n \boldsymbol{N}\boldsymbol{N}^T\, d\Omega\, \boldsymbol{T}_n, \tag{6.19}$$

$$\boldsymbol{G}_n^l = \int_\Omega \rho\mathcal{L}\boldsymbol{N} f_{l(n)}\, d\Omega, \tag{6.20}$$

and the remainder of the contributions are the same as those given in equation (6.12). It must be noted that $\boldsymbol{G}_l = (\boldsymbol{G}_n^l - \boldsymbol{G}_{n-1}^l)/\Delta t$ only when, as considered in this work, ρ does not depend on temperature. Also, note that in the nonlinear case $\boldsymbol{G}_c \neq (\boldsymbol{G}_n^c - \boldsymbol{G}_{n-1}^c)/\Delta t$. Despite these facts, it is assumed that the introduced modifications do not affect the accuracy. Take into account that the only difference is that in the term $\boldsymbol{G}_{n-1}^c$, the material property, given by the heat capacity, is evaluated with $\boldsymbol{T}_{n-1}$ instead of $\boldsymbol{T}_n$, as it is done in equation (6.12).

6.3 Formulation of the Reduced Order Model

The application of spatio-temporal separated representations for the formulation and design of the Reduced Order Model is studied next (for a general treatment of separated representations, see for instance [133]). In our case, to describe the unknown temperature $T(\boldsymbol{x}, t)$, we use the separated representation given by

$$T^h(\boldsymbol{x}, t) \simeq \sum_{j=1}^{M} S_j(\boldsymbol{x}) R_j(t) + T_d(\boldsymbol{x}, t), \tag{6.21}$$

where T_d denotes the non-homogeneous essential boundary conditions.

Both approaches, the *a posteriori* and *a priori* ROM techniques, make use of this idea. These methods assume that the response of the system under study is *k-compressible* in a certain basis, meaning that only the first k components, with $k \ll M$, condense all the significant information about the system behaviour. The manner in which this basis is determined gives the particular features of the ROM formulation. The concept of *k-compressibility* of the signals is therefore highly important. Next, we focus on the POD-ROM, which is classified as an *a posteriori* technique.

Remark: In what follows all the HROMs for the case in which a fixed frame of reference respect to the welded piece is used, are based on the HF model given by equation (6.18). Detailed explanation of this choice is given in Section 6.6.

6.3.1 The Proper Orthogonal Decomposition approach

In an *a posteriori* approach, the response of the system is assumed to be known in advance, e.g., by running a training problem or performing experimental tests. Assuming that the response is k-compressible, we look for a procedure to build a basis $\boldsymbol{\Theta}$ for $\mathbb{R}^N$ with the peculiarity that only the first $k \ll N$ vectors of the basis provide significant information to reproduce the system response. From a physical point of view, we hope to be able to capture the dynamics of the problem with a reduced number of basis vectors. Solutions to problems similar to that of the training problem lie in $span\{\boldsymbol{\Theta}\}$. However, if we take into account that the first k vectors of the basis $\boldsymbol{\Theta}$ condense the significant information on the problem dynamics, we can form a reduced basis $\boldsymbol{\Xi}$ with only k of these vectors, and subsequently look for the solution in $span\{\boldsymbol{\Xi}\}$.

To build the basis, we proceed in a manner similar to that of Principal Component Analysis (PCA) [134]. The main objective is to reduce the dimensionality of the solution while retaining as much information as possible about the system response. To achieve this goal, first, a set of *snapshots* is built using time instances of the spatial distribution of the solution of the training problem [135]. Next, PCA is carried out by making use of

the Proper Orthogonal Decomposition, better known in the context of linear algebra as Singular Value Decomposition (SVD) [136].

Remark: When non-homogeneous essential boundary conditions T_d are imposed, snapshots of the form $T^h - T_d$ are considered. In what follows, it is assumed for conciseness and without loss of generality that $T_d = 0$. The implementation of non-homogeneous essential boundary conditions is left for detailed analysis in a future work. It is taken into account that when $T_d = 0$, the trial and test function spaces are the same.

Let matrix $\boldsymbol{A}$ be a matrix formed by the set of collected snapshots. From the SVD, we have

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T, \tag{6.22}$$

where the columns $\boldsymbol{u}$ of $\boldsymbol{U}$ are the left singular vectors of $\boldsymbol{A}$, the columns $\boldsymbol{v}$ of $\boldsymbol{V}$ are the right singular vectors of $\boldsymbol{A}$ and the diagonal entries σ of $\boldsymbol{\Sigma}$ are the singular values of $\boldsymbol{A}$. If the snapshots $\boldsymbol{A}$ correspond to a function $T_{sp}(\boldsymbol{x}, t)$, with $\boldsymbol{A}_{ij} = T_{sp}(\boldsymbol{x}_i, t_j)$, from equation (6.22) we have

$$T_{sp}(\boldsymbol{x}_i, t_j) = \Big[\sum_{j=1}^{M} \boldsymbol{u}_j \otimes \boldsymbol{v}_j \ \sigma_j\Big]_i. \tag{6.23}$$

If the response of the system is *k-compressible* with respect to the basis $\boldsymbol{u}_j$, we can write

$$T_{sp}(\boldsymbol{x}_i, t_j) \simeq \Big[\sum_{j=1}^{k} \boldsymbol{u}_j \otimes \boldsymbol{v}_j \ \sigma_j\Big]_i, \tag{6.24}$$

where $k \ll M$, and then, the separate representation given by equation (6.21) can be used to describe T^h in terms of $\boldsymbol{u}_j$, leading to

$$\Big[\boldsymbol{T}_n\Big]_i = T^h(\boldsymbol{x}_i, t_n) \simeq \Big[\sum_{j=1}^{k} \boldsymbol{u}_j R_j(t_n)\Big]_i. \tag{6.25}$$

In matrix notation, this equation is given in a more compact form by

$$T^h(\boldsymbol{x}, t_n) = \boldsymbol{N}^T\boldsymbol{T}_n = \boldsymbol{N}^T\boldsymbol{X}\boldsymbol{a}_n, \tag{6.26}$$

where $\boldsymbol{X}$ denotes the change of basis matrix whose columns are the first k vectors $\boldsymbol{u}_j$ and $\boldsymbol{a}_n$ denotes the vector of coefficients $R_j(t)$ evaluated at the time instant t_n.

From equations (6.25) and (6.26), it can be concluded that the basis sought is given by the vectors $\boldsymbol{u}_j$ (commonly referred to as POD modes of the approximation). Moreover, the test functions v^h in this new basis are given by

$$v^h = \boldsymbol{N}^T\boldsymbol{X}\boldsymbol{\eta}, \tag{6.27}$$

with $\boldsymbol{\eta} \in \mathbb{R}^k$.

Making use of the test functions given by the last equation and using a Bubnov-Galerkin projection, we arrive at a discrete variational formulation whose residual is given by

$$\boldsymbol{\Pi}^p(\boldsymbol{a}_n) = \boldsymbol{X}^T\boldsymbol{\Pi} = \boldsymbol{X}^T\frac{\boldsymbol{G}^c_n - \boldsymbol{G}^c_{n-1}}{\Delta t} + \boldsymbol{X}^T\boldsymbol{G}_k + \boldsymbol{X}^T\frac{\boldsymbol{G}^l_n - \boldsymbol{G}^l_{n-1}}{\Delta t} + \boldsymbol{X}^T\boldsymbol{F} - \boldsymbol{X}^T\boldsymbol{Q} = \boldsymbol{0}. \tag{6.28}$$

This result is equivalent to that obtained by projecting the residual given by equation (6.18) on the space spanned by $\boldsymbol{X}$; therefore, we refer to the modes $\boldsymbol{u}_j$ as projection modes of the approximation. Due to the high nonlinearity of the problem, a line-search method must be applied in conjunction with a Newton-Raphson scheme. This type of globally convergent method is quite standard and its formulation can be found in most textbooks in nonlinear optimisation [60, 61].

The tangent matrix corresponding to the residual given by equation (6.28) is

$$\frac{\partial \boldsymbol{\Pi}^p}{\partial \boldsymbol{a}} = \boldsymbol{X}^T \frac{\partial \boldsymbol{\Pi}}{\partial \boldsymbol{a}} = \boldsymbol{X}^T \frac{\partial \boldsymbol{\Pi}}{\partial \boldsymbol{T}} \frac{\partial \boldsymbol{T}}{\partial \boldsymbol{a}} = \boldsymbol{X}^T \frac{\partial \boldsymbol{\Pi}}{\partial \boldsymbol{T}} \boldsymbol{X}. \tag{6.29}$$

As can be observed from the last equation, the system of equations to be solved in the ROM is much smaller in size than the system to be solved in the HF model. In the former case, the size of the problem is $k \times k$, and in the latter case, the size is $N \times N$. Although the computational cost of solving the system of linear equations is reduced, the cost of assembling the residual and the tangent matrix in each Newton iteration is still $O(N)$. To significantly reduce the computational cost, we must reduce the cost of assembling the residual and the tangent matrix at each Newton iteration. In the following sections, we introduce approximations that lead to the formulation of the so-called *hyper-reduced models*.

6.3.1.1 Optimality and consistency

Carlberg *et al.* [122] introduced the notions of optimality and consistency to ensure that *good* approximations are obtained when designing ROMs. According to Carlberg *et al.* [122], "an approximation is said to be *consistent* if, when implemented without data compression, it introduces no additional error in the solution of the same problem for which data was acquired... The approximation is said to be *optimal* if it leads to approximated quantities that minimise some error measure".

The ROM presented in this work satisfies consistency because, when the POD basis is not truncated, T^h is in $span\{\boldsymbol{X}\}$, and the solution of the training problem is recovered. With respect to the optimality of the approximation, it must be mentioned that a Bubnov-Galerkin projection is optimal if the Jacobian or the tangent matrix of the system is Symmetric Positive Definite (SPD). If it is not the case, a Petrov-Galerkin projection should be used [122]. In our ROM, we use a Bubnov-Galerkin projection despite the fact that the system tangent matrix is not SPD because non-symmetry is not large for typical phase change problems.

6.4 Formulation of the Hyper Reduced Order Model

To reduce the cost of assembling the residual and the tangent matrix, we apply the gappy data reconstruction method introduced by Everson and Sirovich [130] in the image processing community and applied by many other researchers in the computational mechanics community [127, 137, 138]. Specifically, suppose only n_s components $\widehat{\boldsymbol{u}}$ of a vector $\boldsymbol{u}$ are known. The objective is to build an approximation $\widetilde{\boldsymbol{u}}$ to $\boldsymbol{u}$, starting from the known components $\widehat{\boldsymbol{u}}$.

First, a set of snapshots corresponding to $\boldsymbol{u}$ is collected. After computing the SVD of this set of snapshots, we obtain a set of POD modes for $\boldsymbol{u}$. Assume that only the first n_g modes are significant; these modes are referred to in this work as gappy modes and are

given as columns of a matrix $\boldsymbol{\Psi}$. Therefore, an approximation to $\boldsymbol{u}$ can be written as

$$\boldsymbol{u} \simeq \boldsymbol{\Psi}\boldsymbol{b}. \tag{6.30}$$

Let us assume that we know only n_s values of $\boldsymbol{u}$. This situation is represented by multiplying equation (6.30) by the sampling matrix $\boldsymbol{R}$ of size $n_s \times N$, yielding

$$\widehat{\boldsymbol{u}} = \boldsymbol{R}\boldsymbol{u} \simeq \boldsymbol{R}\boldsymbol{\Psi}\boldsymbol{b} = \widehat{\boldsymbol{\Psi}}\boldsymbol{b}, \tag{6.31}$$

where the operator $\widehat{\boldsymbol{.}}$ denotes the sampling of $\boldsymbol{.}$. Subsequently, this problem can be stated as that of finding $\boldsymbol{b} \in \mathbb{R}^{n_g}$ such that

$$\boldsymbol{b} = \underset{\boldsymbol{c}\in\mathbb{R}^{n_g}}{\arg\min} \, ||\widehat{\boldsymbol{\Psi}}\boldsymbol{c} - \widehat{\boldsymbol{u}}||_2. \tag{6.32}$$

To gain a unique solution, we require the columns of $\widehat{\boldsymbol{\Psi}}$ to be linearly independent. For this requirement to hold, the constraint $n_g \leq n_s$ must be met. Problem (6.32) subsequently can be solved either by computing the Moore-Penrose pseudo-inverse or by applying the QR decomposition. The latter technique is recommended for ill-conditioned matrices $\widehat{\boldsymbol{\Psi}}^T\widehat{\boldsymbol{\Psi}}$ [139]. In our problems, we have observed that although the condition of $\widehat{\boldsymbol{\Psi}}^T\widehat{\boldsymbol{\Psi}}$ could be somehow deteriorated in certain cases, the results obtained by applying either the QR decomposition or the pseudo-inverse did not differ significantly. In this work the pseudo-inverse is used, giving

$$\boldsymbol{b} = (\widehat{\boldsymbol{\Psi}}^T\widehat{\boldsymbol{\Psi}})^{-1}\widehat{\boldsymbol{\Psi}}^T\widehat{\boldsymbol{u}}. \tag{6.33}$$

By replacing this result into equation (6.30), we arrive at

$$\boldsymbol{u} \simeq \widetilde{\boldsymbol{u}} = \boldsymbol{\Psi}(\widehat{\boldsymbol{\Psi}}^T\widehat{\boldsymbol{\Psi}})^{-1}\widehat{\boldsymbol{\Psi}}^T\widehat{\boldsymbol{u}}. \tag{6.34}$$

In other words, an approximation to the entire vector $\boldsymbol{u}$ is reconstructed, with only the knowledge of the n_s components $\widehat{\boldsymbol{u}}$ of $\boldsymbol{u}$. By extending these ideas to our problem, we can compute the residual $\boldsymbol{\Pi}$ only knowing its values at n_s points, thus reducing the order of the assembly process from N to n_s, with $n_s \ll N$.

Several techniques aimed at finding the n_s points for which to evaluate $\boldsymbol{\Pi}$ have been proposed in the literature. A number of techniques operate at the discrete level, whereas others are expressed at the continuum level. Currently, two approaches that operate at the continuum level are widely used and can be mentioned: the *Best Point Interpolation Method (BPIM)* [140], which optimally selects the points, and the *Empirical Interpolation Method (EIM)* [141], which yields good but not optimal performance. Discrete versions of these algorithms have also been proposed [126, 127]. This work uses an extension of the *Discrete Empirical Interpolation Method (DEIM)* [127]. This method is briefly described in algorithm (5), in which the sampling points are returned in vector $\boldsymbol{p}$.

Code 5 DEIM extension: return sampling points in vector $\boldsymbol{p}$

```
function DEIMEXT(Ψ)
    z := Ψ(:, 1)
    p(1) := maxAbs(z)              ▷ Find index of maximum absolute value of z
    for i = 1 to n_s − 1 do
        n := min(i, n_g)
        Λ := Ψ(p, 1 : n)
        U := Λ^T Λ
        v := mod(i, n_g) + 1
        z := Ψ(:, v)
        c := U^{-1} Λ^T z(p)
        r := z − Ψ(:, 1 : n)c
        p(i + 1) := maxAbs(r, p)   ▷ Find index of maximum absolute value
                                   ▷ of r which is not in the set of indices p
    end for
    return p
end function
```

Different approaches for the design of HROMs have been proposed in the literature [122, 125, 127, 137, 141, 138]. The main differences among them lie in the choice of the set of snapshots used to compute the POD modes of the approximation. In addition, the proposal of Carlberg *et al.* [122, 125] differs from the others in the fact that they use one set of POD modes for interpolating the residual and another set for interpolating the tangent matrix. Instead, in our proposal, we derive the method for interpolation of the tangent matrix from the procedure used to interpolate the residual. For this reason, we first focus our study on the treatment of the residual.

In the following sections, two different approximations for collecting the snapshots are analysed. The expected performance of each approximation is evaluated in terms of the *Relative Information Content (RIC)*, indicator, which is defined as

$$\text{RIC} = \frac{\sum_{i=1}^{N_s} \lambda_i}{\sum_{i=1}^{N_r} \lambda_i}, \tag{6.35}$$

where λ_i are the singular values of the SVD, N_s is the number of singular values kept in the truncation and N_r is the total number of singular values of the SVD.

6.4.1 Snapshot collection by one nucleating nonlinear term

A strategy used to formulate the HROM divides the residual into two contributions: one term that nucleates the linear terms and another that nucleates the nonlinear terms [137]. Next, the gappy method is applied to the nonlinear term. In this sense, it can be said that the residual is divided into one term, which is hyper-reduced, and another term, which is assembled just once without needing to be hyper-reduced. Certain other authors, such as Carlberg *et al.* [122, 125], hyper-reduce the residual as a single entity.

In our case, the terms that comprise the residual $\boldsymbol{\Pi}$ in equation (6.18) are all nonlinear except for the heat source term. However, because the heat source is in motion and follows a given trajectory, an assembly process is required for this term at each time instant,

operation which is $O(N)$. Therefore, this term also must be hyper-reduced by gappy data reconstruction, making it necessary to hyper-reduce the residual as a unit, similar to what was performed by Carlberg *et al.*.

We take the residuals of the HF model in each Newton iteration as snapshots, diverging from Carlberg *et al.* at this point because they take the residuals of the reduced problem as snapshots instead (formulation presented in Section 6.3.1). The POD modes for interpolation are computed by applying the SVD to the set of collected snapshots. If we denote $\boldsymbol{\Phi}$ the first n_g POD modes of the residual, we have

$$\boldsymbol{\Pi} \simeq \widetilde{\boldsymbol{\Pi}} = \boldsymbol{\Phi}(\widehat{\boldsymbol{\Phi}}^T\widehat{\boldsymbol{\Phi}})^{-1}\widehat{\boldsymbol{\Phi}}^T\widehat{\boldsymbol{\Pi}}. \tag{6.36}$$

Next, to obtain the hyper-reduced residual $\boldsymbol{\Pi}^p$, we project this equation with the POD basis $\boldsymbol{X}$ as in equation (6.28); that is

$$\boldsymbol{\Pi}^p = \boldsymbol{X}^T\widetilde{\boldsymbol{\Pi}} = \boldsymbol{X}^T\boldsymbol{\Phi}(\widehat{\boldsymbol{\Phi}}^T\widehat{\boldsymbol{\Phi}})^{-1}\widehat{\boldsymbol{\Phi}}^T\widehat{\boldsymbol{\Pi}} = \boldsymbol{A}_r\widehat{\boldsymbol{\Pi}}. \tag{6.37}$$

Take into account that matrix $\boldsymbol{A}_r = \boldsymbol{X}^T\boldsymbol{\Phi}(\widehat{\boldsymbol{\Phi}}^T\widehat{\boldsymbol{\Phi}})^{-1}\widehat{\boldsymbol{\Phi}}^T$ is computed just once in the *off-line* stage.

The hyper-reduced tangent matrix is finally obtained by differentiation of the last equation to produce

$$\frac{\partial\boldsymbol{\Pi}^p}{\partial\boldsymbol{a}} = \boldsymbol{A}_r\frac{\partial\widehat{\boldsymbol{\Pi}}}{\partial\boldsymbol{a}} = \boldsymbol{A}_r\frac{\partial\boldsymbol{R}\boldsymbol{\Pi}}{\partial\boldsymbol{a}} = \boldsymbol{A}_r\boldsymbol{R}\frac{\partial\boldsymbol{\Pi}}{\partial\boldsymbol{T}}\boldsymbol{X}. \tag{6.38}$$

In the following the consistency of this approximation is studied. If all modes are retained in the gappy basis $\boldsymbol{\Phi}$ and in the basis $\boldsymbol{X}$, i.e., $\boldsymbol{\Phi} = \widehat{\boldsymbol{\Phi}}$ and $\boldsymbol{\Pi} = \widehat{\boldsymbol{\Pi}}$, then in equation (6.36) produces

$$\widetilde{\boldsymbol{\Pi}} = \boldsymbol{\Phi}(\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^T\boldsymbol{\Pi} = \boldsymbol{\Phi}\boldsymbol{I}^{-1}\boldsymbol{\Phi}^T\boldsymbol{\Pi} = \boldsymbol{\Phi}\boldsymbol{\Phi}^T\boldsymbol{\Pi} = \boldsymbol{I}\boldsymbol{\Pi} = \boldsymbol{\Pi}, \tag{6.39}$$

where $\boldsymbol{I}$ is the identity matrix. Therefore, when no truncation is carried out, the residual $\boldsymbol{\Pi}$ is fully recovered. Using this result in equation (6.37), the projected residual given by equation (6.28) is obtained, and the solution of the HROM and the solution of the ROM coincide. Moreover, it was shown in Section 6.3.1.1 that the solution of the ROM converges to the solution of the training problem when no truncation is used, making it possible to conclude that the solution of the HROM and the solution of the training problem are the same, and therefore the method is consistent.

Remarks:

- To ensure that the projected tangent matrix given by equation (6.38) is full-rank, we require the number of gappy modes n_g to obey the constraint $n_g \geq k$. The justification for this constraint is given in Section 6.4.3.

- Equation (6.37) can be re-written as

$$\boldsymbol{\Pi}^p = \boldsymbol{X}^T\boldsymbol{\Phi}\boldsymbol{c}, \tag{6.40}$$

  where $\boldsymbol{c}$ is the vector that condenses the rest of the terms involved in the HROM. Next, it is evident that matrix $\boldsymbol{X}^T\boldsymbol{\Phi}$ should have rank k for the formulation of the HROM to be well-posed. This condition is satisfied by the current hyper-reduced formulation, and details are given in Section 6.4.3.

- The columns of $\boldsymbol{\Phi}$ are left singular vectors of the SVD of the matrix built by the snapshots taken at all Newton iterations of the residual. Therefore, they should normally be linearly independent, which is the condition required for $\boldsymbol{\Phi}$ to be a basis. However, in certain cases, these vectors can be close to parallel. An example is a case in which snapshots are collected from a training problem with a notably small time step; in this situation, all residuals will be close to zero and the computed left singular vectors will be almost linearly dependent. This type of situations leads to ill-conditioning and problems of convergence.

6.4.2 Snapshot collection by individual nonlinear terms

An alternative approach for the snapshot collection is proposed in this section to avoid the inconveniences of the previous procedure. We consider the residual $\boldsymbol{\Pi}$ as the summation of various physically based nonlinear generalized contributing forces plus one term that nucleates the linear contributions. Next, each nonlinear term will be individually stored and hyper-reduced. The heat source term will be hyper-reduced as well to reduce its assembly cost from $O(N)$ to $O(n_s)$.

It must be kept in mind that to obtain an optimal method, we need to individualise terms that are highly *compressible*. Each term given in equation (6.12) will be computed using the gappy data technique, except for terms $\boldsymbol{G}_c$ and $\boldsymbol{G}_l$.

The latter terms can be handled in two different forms. The first form hyper-reduces $\boldsymbol{G}_c$ and $\boldsymbol{G}_l$ directly, thus basing the HROM on the HF model given by equation (6.12). The other option is to based the HROM on the formulation given by equation (6.18), that re-writes the terms $\boldsymbol{G}_c$ and $\boldsymbol{G}_l$ as reaction increments during the time step, and subsequently hyper-reducing the reactions at the current time step. Snapshots are taken for each individual contribution at each time step after convergence of the Newton-Raphson scheme.

The SVD spectrum of $\boldsymbol{G}_c$ and $\boldsymbol{G}_l$ is expected to be less compact than that of the terms $\boldsymbol{G}_n^c$ and $\boldsymbol{G}_n^l$ because the former terms can be viewed as integrated forms of the latter. Indeed, this is the case that was verified by numerical experiments. Moreover, it was verified that the condition of the HROM tangent matrix in the former case is worse than in the latter case. These comments justify the adoption of the formulation given by equation (6.18), instead of that given by equation (6.12), as HF model for the HROM.

Each term of equation (6.18) has an associated POD basis for its gappy data reconstruction. In what follows, suffixes $i \in \{c, k, l, f, q\}$ are used to identify the different POD bases $\boldsymbol{\Phi}_i$ and sampling matrices $\boldsymbol{R}_i$, corresponding to each term. We emphasise that the sampling matrices are different for each term, but the number of sampling points n_s and the number of gappy modes n_g are always the same for all. Then, the approximation to each term from the gappy data reconstruction is given by

$$\boldsymbol{G}_n^c \simeq \boldsymbol{\Phi}_c(\widehat{\boldsymbol{\Phi}}_c^T\widehat{\boldsymbol{\Phi}}_c)^{-1}\widehat{\boldsymbol{\Phi}}_c^T\widehat{\boldsymbol{G}}_n^c, \tag{6.41}$$

$$\boldsymbol{G}_k \simeq \boldsymbol{\Phi}_k(\widehat{\boldsymbol{\Phi}}_k^T\widehat{\boldsymbol{\Phi}}_k)^{-1}\widehat{\boldsymbol{\Phi}}_k^T\widehat{\boldsymbol{G}}_k, \tag{6.42}$$

$$\boldsymbol{G}_n^l \simeq \boldsymbol{\Phi}_l(\widehat{\boldsymbol{\Phi}}_l^T\widehat{\boldsymbol{\Phi}}_l)^{-1}\widehat{\boldsymbol{\Phi}}_l^T\widehat{\boldsymbol{G}}_n^l, \tag{6.43}$$

$$\boldsymbol{F} \simeq \boldsymbol{\Phi}_f(\widehat{\boldsymbol{\Phi}}_f^T\widehat{\boldsymbol{\Phi}}_f)^{-1}\widehat{\boldsymbol{\Phi}}_f^T\widehat{\boldsymbol{F}}, \tag{6.44}$$

$$\boldsymbol{Q} \simeq \boldsymbol{\Phi}_q(\widehat{\boldsymbol{\Phi}}_q^T\widehat{\boldsymbol{\Phi}}_q)^{-1}\widehat{\boldsymbol{\Phi}}_q^T\widehat{\boldsymbol{Q}}, \tag{6.45}$$

where $\widehat{\boldsymbol{\cdot}}$ denotes the vector of n_s components sampled from the associated full term.

To obtain the hyper-reduced residual $\boldsymbol{\Pi}^p$ we project the approximation of $\boldsymbol{\Pi}$ with the POD basis $\boldsymbol{X}$, as in equation (6.37) which results in

$$\boldsymbol{\Pi}^p = \boldsymbol{A}_c \frac{\widehat{\boldsymbol{G}}^c_n - \widehat{\boldsymbol{G}}^c_{n-1}}{\Delta t} + \boldsymbol{A}_k \widehat{\boldsymbol{G}}_k + \boldsymbol{A}_l \frac{\widehat{\boldsymbol{G}}^l_n - \widehat{\boldsymbol{G}}^l_{n-1}}{\Delta t} + \boldsymbol{A}_f \widehat{\boldsymbol{F}} - \boldsymbol{A}_q \widehat{\boldsymbol{Q}}, \tag{6.46}$$

where $\boldsymbol{A}_i = \boldsymbol{X}^T \boldsymbol{\Phi}_i (\widehat{\boldsymbol{\Phi}}_i^T \widehat{\boldsymbol{\Phi}}_i)^{-1} \widehat{\boldsymbol{\Phi}}_i^T$, with $i \in \{c, k, l, f, q\}$. Again, take into account that matrices $\boldsymbol{A}_i$ are computed in the *off-line* stage.

Finally, the hyper-reduced tangent matrix is obtained by differentiating the residual with respect to the global unknown parameters

$$\frac{\partial \boldsymbol{\Pi}^p}{\partial \boldsymbol{a}} = \boldsymbol{A}_c \boldsymbol{R}_c \frac{\partial \boldsymbol{G}^c_n}{\partial \boldsymbol{T}} \boldsymbol{X} + \boldsymbol{A}_k \boldsymbol{R}_k \frac{\partial \boldsymbol{G}_k}{\partial \boldsymbol{T}} \boldsymbol{X} + \boldsymbol{A}_l \boldsymbol{R}_l \frac{\partial \boldsymbol{G}^l_n}{\partial \boldsymbol{T}} \boldsymbol{X} + \boldsymbol{A}_f \boldsymbol{R}_f \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{T}} \boldsymbol{X}. \tag{6.47}$$

The consistency of the current HROM can be shown by following the same procedure that was applied in the previous method (Section 6.4.1).

6.4.3 Relationship between the projection and the hyper-reduction spaces

In the previously developed HROMs, we addressed the hyper-reduced quantities $\boldsymbol{\Pi}^p_i$ of the form

$$\boldsymbol{\Pi}^p_i = \boldsymbol{X}^T \boldsymbol{\Phi}_i \boldsymbol{c}, \tag{6.48}$$

where $\boldsymbol{X}$ is the basis spanning the solution space, $\boldsymbol{\Phi}_i$ is the POD basis used in the gappy data reconstruction and $\boldsymbol{c}$ is a vector condensing the rest of the terms involved in the HROM.

To obtain a well-posed formulation of the HROM, we need matrix $\boldsymbol{X}^T \boldsymbol{\Phi}_i$ to have maximum rank k. Note that terms $\boldsymbol{\Phi}_i$ and $\boldsymbol{X}$ are energetically conjugated. Therefore, their mutual product is equal to an energy increment, and it is intuitively expected that $\boldsymbol{X}^T \boldsymbol{\Phi}_i$ has maximum rank k. Next, we show that this expectation is effectively verified in our case.

For the solidification problem under consideration, we take the temperature values $\boldsymbol{T}_j$ of the system at a sequence of S time instants t_j, $j = 1, \dots S$ as snapshots for building the basis $\boldsymbol{X}$. To design the HROM, we computed the residual as the summation of various physical contributions and took snapshots for each of these quantities to compute the corresponding POD basis for the gappy data reconstruction procedure.

For instance, let us consider the thermal conductivity term: the nonlinear contribution to the residual reads

$$\boldsymbol{G}_k = \int_\Omega \nabla \boldsymbol{N} \cdot (k \, \nabla \boldsymbol{N} \, \boldsymbol{T}) \, d\Omega. \tag{6.49}$$

For the problem at hand, it is well known that

$$\int_\Omega \nabla \boldsymbol{T} \cdot (k \nabla \boldsymbol{T}) \, d\Omega = \boldsymbol{T}^T \boldsymbol{K} \boldsymbol{T} \; > \; 0 \, , \tag{6.50}$$

with given Dirichlet boundary conditions. If we assume that the temperature field signal is *k-compressible*, we know that for a certain basis $\boldsymbol{X}$ there exists $\boldsymbol{\xi} \in \mathbb{R}^k$ such that

$$\boldsymbol{T} = \boldsymbol{X} \boldsymbol{\xi}, \tag{6.51}$$

where the equality is used for conciseness (it is in fact $\boldsymbol{T} \simeq \boldsymbol{X} \boldsymbol{\xi}$).

Now, let $\boldsymbol{B} = [\boldsymbol{G}_{k_1}, \boldsymbol{G}_{k_2}, \ldots, \boldsymbol{G}_{k_S}]$ denote the matrix formed by S snapshots $\boldsymbol{G}_{k_j} = \boldsymbol{K}\boldsymbol{T}_j$. If we assume that $\boldsymbol{B}$ is n_g-*compressible* and $n_g \geq k$, we have

$$\boldsymbol{B}\boldsymbol{B}^T = \boldsymbol{\Phi}_k \boldsymbol{\Delta}^2 \boldsymbol{\Phi}_k^T, \tag{6.52}$$

where $\boldsymbol{\Delta}$ is the matrix of the first n_g singular values of $\boldsymbol{B}$, and $\boldsymbol{\Phi}_k$ are the first n_g left singular vectors of $\boldsymbol{B}$.

Let us now consider matrix $\boldsymbol{C}$ defined as the projection of $\boldsymbol{B}^T\boldsymbol{B}$ on the solution space. From the considerations above, we obtain

$$\boldsymbol{C} = \boldsymbol{X}^T \boldsymbol{\Phi}_k \boldsymbol{\Delta}^2 \boldsymbol{\Phi}_k^T \boldsymbol{X} = \boldsymbol{X}^T \boldsymbol{B}\boldsymbol{B}^T \boldsymbol{X} = \boldsymbol{X}^T \boldsymbol{K}\boldsymbol{J}\boldsymbol{J}^T \boldsymbol{K}^T \boldsymbol{X}, \tag{6.53}$$

where $\boldsymbol{J} = [\boldsymbol{T}_1, \boldsymbol{T}_2, \ldots, \boldsymbol{T}_S]$. The term $\boldsymbol{J}\boldsymbol{J}^T$ can be expressed as

$$\boldsymbol{J}\boldsymbol{J}^T = \boldsymbol{X}\boldsymbol{\Sigma}^2\boldsymbol{X}^T, \tag{6.54}$$

because the temperature field is assumed to be k-*compressible*, where $\boldsymbol{\Sigma}$ is the matrix of the first k singular values of $\boldsymbol{J}$. Therefore, $\boldsymbol{C}$ reads

$$\boldsymbol{C} = \boldsymbol{X}^T \boldsymbol{K}\boldsymbol{X}\boldsymbol{\Sigma}^2\boldsymbol{X}^T\boldsymbol{K}^T\boldsymbol{X} = \overline{\boldsymbol{K}}^T \boldsymbol{\Sigma}^2 \overline{\boldsymbol{K}}, \tag{6.55}$$

where $\overline{\boldsymbol{K}} = \boldsymbol{X}^T\boldsymbol{K}^T\boldsymbol{X}$. From equation (6.50), $\overline{\boldsymbol{K}}$ is symmetric positive definite. Moreover, if the left singular vectors associated with zero singular values are not part of $\boldsymbol{X}$, $\boldsymbol{\Sigma}^2$ is a diagonal matrix with positive entries. Therefore, $\boldsymbol{C}$ is symmetric positive definite and invertible. It follows therefore that the matrix $\boldsymbol{X}^T\boldsymbol{\Phi}$ has rank k.

A similar analysis can be carried for the other terms of the HROM presented in Section 6.4.2 with the conclusion that it is well-posed. A similar conclusion can be obtained for the HROM given in Section 6.4.1.

We mention for completeness that an HROM formulation in which the projection space and the hyper-reduction space are mutually orthogonal leads to an ill-posed problem. A solution to this type of singularity was proposed by Hernández *et al.* [124]

It was previously mentioned that to ensure the projected tangent matrix is full-rank, the number of gappy modes must obey the constraint $n_g \geq k$. The reason for this constraint is obvious with the development presented in this section. If the number of gappy modes is smaller than the number of POD projection modes, i.e., $n_g < k$, the matrix $\boldsymbol{B}\boldsymbol{B}^T$ given by equation (6.52) would have rank n_g, and the invertibility of $\boldsymbol{C}$ would not be guaranteed, and it cannot be assured that the projected tangent matrix would have rank k.

6.5 Alternative formulation: The moving frame approach

A formulation of HROMs for nonlinear transient heat conduction problems expressed in terms of a moving frame of reference is introduced next, specifically, a frame of reference attached to a heat source. In the case of a highly concentrated moving heat source, such a formulation permits us to obtain a problem whose solution is characterised by a high *compressibility*. In other words, a small number of projection modes and gappy points are required when using a moving frame of reference that follows the high gradients present in the solution.

6.5.1 Variational formulation and discretisation

The following considers the energy balance equations in a moving frame of reference attached to the heat source. Let $\boldsymbol{e}_i$, with $i \in \{1,2,3\}$, be unit vectors associated with a body fixed frame centered at O. Let $\boldsymbol{r}(t)$ be the parametrisation of the curve describing the trajectory of the heat source with respect to $\boldsymbol{e}_i$. We describe this trajectory by the position of M_c points $\boldsymbol{c}_i$, interpolated by shape functions $R_i(t)$

$$\boldsymbol{r}(t) = \sum_{i=1}^{M_c} R_i(t)\boldsymbol{c}_i, \tag{6.56}$$

where the time $t \in [0, t_f]$ is subdivided into $M_c - 1$ sub-intervals $I_k = (t_k, t_{k+1})$ with $k \in \{1, \ldots, M_c - 1\}$, such that $\boldsymbol{r}(t_k) = \boldsymbol{c}_k$. In the case of a moving frame, observe

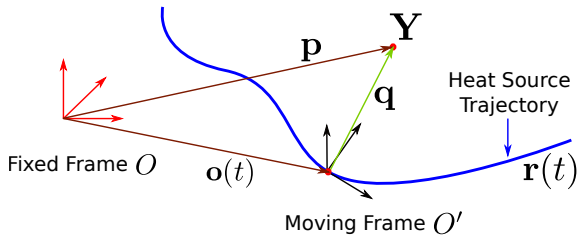


Figure 6.1: Fixed and Moving frames.

Figure 6.1, the position of the heat source at each time instant is taken as the origin of this moving welding frame or the position of the observer O'. We denote by $\boldsymbol{u}_i$, with $i \in \{1,2,3\}$, the orthonormal unit vectors associated to the moving frame. We use a cubic spline to interpolate the trajectory of the heat source. Let $\boldsymbol{p}_s$ and $\boldsymbol{p}_e$ be the given starting tangent at $t = 0$ and the given ending tangent at $t = t_f$, respectively. Next, $\boldsymbol{r}(t)$ in an arbitrary interval I_k is given by

$$\boldsymbol{r}(t) = R_{00}(t)\boldsymbol{c}_k + R_{10}(t)\boldsymbol{p}_k + R_{01}(t)\boldsymbol{c}_{k+1} + R_{11}(t)\boldsymbol{p}_{k+1}, \tag{6.57}$$

where

$$R_{00}(t) = (1 + 2\zeta)(1 - \zeta)^2, \tag{6.58}$$
$$R_{10}(t) = (t - t_k)(1 - \zeta)^2, \tag{6.59}$$
$$R_{01}(t) = (3 - 2\zeta)\zeta^2, \tag{6.60}$$
$$R_{11}(t) = (t - t_k)(1 - \zeta)\zeta, \tag{6.61}$$
$$\zeta = \frac{t - t_k}{t_{k+1} - t_k}. \tag{6.62}$$

The tangents $\boldsymbol{p}_k$ are computed according to the Catmull-Rom formula

$$\boldsymbol{p}_k = \frac{\boldsymbol{p}_{k+1} - \boldsymbol{p}_{k-1}}{t_{k+1} - t_{k-1}}. \tag{6.63}$$

The unit vectors $\boldsymbol{u}_i$ of the welding frame are given by (i) $\boldsymbol{u}_1(t) = \boldsymbol{r}'(t)/||\boldsymbol{r}'(t)||$ the tangent to the trajectory; (ii) $\boldsymbol{u}_2(t)$ the outward normal to the welding surface; and (iii) $\boldsymbol{u}_3(t) = \boldsymbol{u}_1(t) \times \boldsymbol{u}_2(t)$. The set of coordinates in this frame is denoted by $\boldsymbol{\xi} = [\xi_1, \xi_2, \xi_3]$.

The position vector $\boldsymbol{p}$ of an arbitrary point $\boldsymbol{Y}$ with respect to the fixed frame O can be computed as

$$\boldsymbol{p}(t) = \boldsymbol{o}(t) + \boldsymbol{\Lambda}(t)\boldsymbol{q}(t), \tag{6.64}$$

where $\boldsymbol{q}(t)$ is the position of $\boldsymbol{Y}$ with respect to the moving frame O', vector $\boldsymbol{o}(t)$ is the position of the moving frame with respect to the fixed frame and $\boldsymbol{\Lambda}$ is the rotation matrix formed by the unit vectors $\boldsymbol{u}_i$ as columns.

The material derivative in the moving frame of the scalar field T is given by

$$\frac{DT}{Dt} = \frac{\partial T}{\partial t} + \frac{\partial \boldsymbol{\xi}}{\partial t} \cdot \nabla_{\boldsymbol{\xi}} T, \tag{6.65}$$

where $\nabla_{\boldsymbol{\xi}}$ is the gradient operator given by $\nabla_{\boldsymbol{\xi}} = \frac{\partial}{\partial \boldsymbol{\xi}}$, and

$$\frac{\partial \boldsymbol{\xi}}{\partial t} = \frac{\partial \boldsymbol{\Lambda}^T}{\partial t} \boldsymbol{\Lambda}\boldsymbol{q} - \boldsymbol{\Lambda}^T \frac{\partial \boldsymbol{o}}{\partial t}. \tag{6.66}$$

The Jacobian of the transformation relating both frames is given by

$$\frac{\partial \boldsymbol{\xi}}{\partial \boldsymbol{x}} = \boldsymbol{\Lambda}, \tag{6.67}$$

and therefore, the conductivity term reads

$$\nabla \cdot \boldsymbol{k} \nabla T = \nabla_{\boldsymbol{\xi}} \cdot (\boldsymbol{\Lambda}\boldsymbol{k}\boldsymbol{\Lambda}^T) \nabla_{\boldsymbol{\xi}} T. \tag{6.68}$$

The material derivative of the liquid fraction is analysed next. Note that this term must be understood in a distributional sense. The material derivative of the liquid fraction in a variational framework can be expressed as

$$\int_\Omega w \frac{Df_l}{Dt}\, d\Omega = \int_\Omega w \Big[\frac{\partial f_l}{\partial t} + \frac{\partial \boldsymbol{\xi}}{\partial t} \cdot \nabla_{\boldsymbol{\xi}} f_l \Big]\, d\Omega. \tag{6.69}$$

By applying Green's theorem to the second term, we obtain

$$\int_\Omega w \frac{\partial \boldsymbol{\xi}}{\partial t} \cdot \nabla_{\boldsymbol{\xi}} f_l \, d\Omega = -\int_\Omega f_l \frac{\partial \boldsymbol{\xi}}{\partial t} \cdot \nabla_{\boldsymbol{\xi}} w \, d\Omega - \int_\Omega f_l w \nabla_{\boldsymbol{\xi}} \cdot \frac{\partial \boldsymbol{\xi}}{\partial t} \, d\Omega + \int_{\Gamma \setminus \Gamma_d} f_l w \frac{\partial \boldsymbol{\xi}}{\partial t} \cdot \boldsymbol{n} \, d\Gamma. \tag{6.70}$$

Taking into account that $\nabla_{\boldsymbol{\xi}} \cdot \frac{\partial \boldsymbol{\xi}}{\partial t} = 0$ and that $\frac{\partial \boldsymbol{\xi}}{\partial t} \cdot \boldsymbol{n} = 0$, because the torch motion lies on the body surface, we obtain

$$\int_\Omega w \frac{\partial \boldsymbol{\xi}}{\partial t} \cdot \nabla_{\boldsymbol{\xi}} f_l \, d\Omega = -\int_\Omega f_l \frac{\partial \boldsymbol{\xi}}{\partial t} \cdot \nabla_{\boldsymbol{\xi}} w \, d\Omega. \tag{6.71}$$

Finally, the variational formulation for the welding problem expressed in a frame moving with the heat source reads:
Find $T \in \mathcal{S}$ *such that* $\forall w \in \mathcal{V}$

$$\begin{aligned}
&\int_\Omega w \left[\rho c \frac{\partial T}{\partial t} + \rho c \frac{\partial \boldsymbol{\xi}}{\partial t} \cdot \nabla_{\boldsymbol{\xi}} T + \rho \mathcal{L} \frac{\partial f_l}{\partial t} - Q(\boldsymbol{\xi}) \right] \, d\Omega + \int_\Omega \nabla_{\boldsymbol{\xi}} w \cdot \left[(\boldsymbol{\Lambda}\boldsymbol{k}\boldsymbol{\Lambda}^T) \nabla_{\boldsymbol{\xi}} T - \rho \mathcal{L} f_l \frac{\partial \boldsymbol{\xi}}{\partial t} \right] \, d\Omega \\
&+ \int_{\Gamma_c} w h_f (T - T_f) \, d\Gamma + \int_{\Gamma_q} w q_w \, d\Gamma = 0 \,, \qquad \text{for } \; t > 0; \\
&\int_\Omega wT \, d\Omega = \int_\Omega w T_0 \, d\Omega \,, \qquad \text{for } \; t = 0.
\end{aligned} \tag{6.72}$$

Note that the heat source $Q(\boldsymbol{\xi})$ does not depend on time anymore. Note also that two new terms appear compared with equation (6.10): one originating from the material time derivative of the temperature field and another from the material time derivative of the liquid fraction. Additionally, the conductivity properties are affected by the rotation of the computation frame.

The finite element discretisation is quite similar to the one introduced in Section 6.2.1. The equation to be solved is briefly expressed as

$$\boldsymbol{\Pi} = \frac{\boldsymbol{G}_n^c - \boldsymbol{G}_{n-1}^c}{\Delta t} + \boldsymbol{G}_k + \boldsymbol{G}_{vc} + \frac{\boldsymbol{G}_n^l - \boldsymbol{G}_{n-1}^l}{\Delta t} - \boldsymbol{G}_{vl} + \boldsymbol{F} - \boldsymbol{Q} = \boldsymbol{0}, \tag{6.73}$$

where the new terms $\boldsymbol{G}_{vc}$ are taken from the discretisation of the term $\rho c \frac{\partial \boldsymbol{\xi}}{\partial t} \cdot \nabla_{\boldsymbol{\xi}} T$, and $\boldsymbol{G}_{vl}$ expresses the discretisation of the term $\nabla_{\boldsymbol{\xi}} w \cdot \rho \mathcal{L} f_l \frac{\partial \boldsymbol{\xi}}{\partial t}$.

6.5.2 HROM formulation

The HROM formulation in the moving frame is constructed using the same developments as for the fixed frame formulation. The procedure for dividing the residual $\boldsymbol{\Pi}$ into a set of physically based nonlinear generalized forces that contribute to the residual and one term that nucleates the linear contributions, is followed. Next, the only difference is that the hyper-reduced versions of the new terms $\boldsymbol{G}_{vc}$ and $\boldsymbol{G}_{vl}$ must be incorporated, resulting in

$$\boldsymbol{\Pi}^p = \boldsymbol{A}_c \frac{\widehat{\boldsymbol{G}}_n^c - \widehat{\boldsymbol{G}}_{n-1}^c}{\Delta t} + \boldsymbol{A}_{vc} \widehat{\boldsymbol{G}}_{vc} + \boldsymbol{A}_k \widehat{\boldsymbol{G}}_k + \boldsymbol{A}_l \frac{\widehat{\boldsymbol{G}}_n^l - \widehat{\boldsymbol{G}}_{n-1}^l}{\Delta t} - \boldsymbol{A}_{vl} \widehat{\boldsymbol{G}}_{vl} + \boldsymbol{A}_f \widehat{\boldsymbol{F}} - \boldsymbol{X}^T \boldsymbol{Q}. \tag{6.74}$$

Note that because the heat source term does not depend on time, it is not hyper-reduced and its projection is computed only once. In addition, note that the same notation of Section 6.5.2 has been used to identify the POD modes and the gappy modes. Clearly, these modes have been computed from snapshots obtained from a HF model calculated using a moving frame of reference formulation and therefore they are different from the modes used in Section 6.5.2.

6.6 Application examples

We present two examples for assessing the performance of the introduced HROMs. The first example consists of the solidification of a cube. In the second example, we study a welding-like problem of a tube without material deposition.

To ensure the *compressibility* of the involved signals, the HF model given by equation (6.18) was used in the formulation of the HROM presented in Section 6.4.2. This HF model was also chosen to based the HROM of Section 6.4.1. This setting makes easier the comparison between the results obtained with these two HROMs, in the sense that at least they are based on the same HF model. Nonetheless, results obtained with the HROM formulation based on a moving frame is still comparable with the other HROMs because, although the HF model is different, the same physical problem is being solved for.

To assess the performance and robustness of the introduced HROMs, the following parameters are evaluated:

- The condition number of the matrices $\boldsymbol{M} = \widehat{\boldsymbol{\Phi}}_i^T \widehat{\boldsymbol{\Phi}}_i$ that are part of the gappy data reconstruction method. The condition number cond($\boldsymbol{M}$), is computed as

$$\text{cond}(\boldsymbol{M}) = \frac{\lambda_e}{\lambda_0}, \tag{6.75}$$

  where λ_0 and λ_e are the maximum and minimum singular values of $\boldsymbol{M}$, respectively.

- The evolution of the condition number of the HROM tangent matrix. This information is plotted in terms of the global iteration number (the iterations are numbered globally, i.e., the iterations of all time steps are consecutively numbered).

- The relative error ϵ introduced by the HROM. This error is computed as

$$\epsilon = \frac{||T_R - T_H||_F}{||T_H||_F}, \tag{6.76}$$

  where T_R is the matrix of the solution in time obtained with the HROM, T_H is the HF solution for the same problem, and $||\,\boldsymbol{.}\,||_F$ is the Frobenius norm of matrix $\boldsymbol{.}$. To measure this error, the temperature field is computed for every 20 time steps of the simulation.

- The speedup S_H of the HROM with respect to the HF model, defined as

$$S_H = \frac{t_H}{t_R}, \tag{6.77}$$

  where t_H and t_R are the total times needed to compute the solution by the HF model and by the HROM (online stage only), respectively. The computations were carried out by saving the results for every 20 time steps.

A normalised residual norm was used to verify convergence of the Newton-Raphson scheme, given in the HF case by

$$\frac{\|\boldsymbol{\Pi}\|}{\|\boldsymbol{F}_l\| + \|\boldsymbol{G}_k\| + \|\boldsymbol{F}_c\| + \|\boldsymbol{F}\| + \|\boldsymbol{Q}\|}, \tag{6.78}$$

where $\|\,\boldsymbol{.}\,\|$ denotes the L_2 norm of $\boldsymbol{.}$ and

$$\boldsymbol{F}_l = \frac{\boldsymbol{G}_n^c - \boldsymbol{G}_{n-1}^c}{\Delta t}, \tag{6.79}$$

$$\boldsymbol{F}_c = \frac{\boldsymbol{G}_n^l - \boldsymbol{G}_{n-1}^l}{\Delta t}. \tag{6.80}$$

The normalised residual norm for the HROM is defined similarly with the corresponding hyper-reduced quantities. Convergence of the Newton-Raphson scheme is verified when the normalised residual norm is below 1×10^{-6}.

The tests were performed in a computer with the following characteristics:

- Processors: 2 x [Intel(R) Xeon(R) CPU X5680 @ 3.33GHz]
- Number of physical cores: 6 cores on each processor; hyper-threading deactivated.
- Cache L1, L2, L3: 32KB, 256 KB, 12MB.
- NUMA nodes #0, #1: 2 x 48GB
- Operating system: `Linux 3.5.1-1.fc17.x86_64 #1 SMP x86_64 GNU/Linux`

6.6.1 Solidification of a cube

Solidification of a cube of side $L = 0.41$m is analysed first. The body is located at the positive octant, with a corner at the origin of coordinates $(0, 0, 0)$. It is initially at constant temperature $T_0 = 2500$K and is cooled from sides $s_1 = (0, y, z)$ and $s_2 = (x, 0, z)$. Cooling is described by means of a Robin boundary condition, with fluid temperature $T_f = 299$K. The other sides of the cube are perfectly insulated. The heat capacity c, the thermal conductivity k and the heat convection coefficient h_f depend on the temperature and are given by the expressions

$$\begin{aligned} c \left[\frac{\mathrm{J}}{\mathrm{kgK}}\right] &= 0.19566\ T[\mathrm{K}] + 474.04, \\ k \left[\frac{\mathrm{W}}{\mathrm{mK}}\right] &= 0.011808\ T[\mathrm{K}] + 3.7066, \\ h_f \left[\frac{\mathrm{W}}{\mathrm{m^2K}}\right] &= 2.6\ T[\mathrm{K}] - 55. \end{aligned} \tag{6.81}$$

The material density is $\rho = 4430\frac{\mathrm{kg}}{\mathrm{m}}$, the solidus temperature is 1877K, the liquidus temperature is 1933K and the latent heat is $\mathcal{L} = 292600\frac{\mathrm{J}}{\mathrm{kg}}$. The geometry is discretised using 162000 equally spaced tetrahedra and 29791 degrees of freedom. A constant time step of 1s is used for a simulation in the time interval $[0, 800]$ s. The computation of the HF solution required 4318.6 s of CPU time.

6.6.1.1 Results obtained by hyper-reducing the residual

The HROM presented in Section 6.4.1 is applied in this section to solve the solidifying cube problem. Figure 6.2 shows the SVD spectrum of the temperature and of the residual histories. From this figure, a good approximation is expected using a small number of projection modes and sampling points. However, as shown in the following, this is not the case, and the error could not be decreased below 1×10^{-4}.

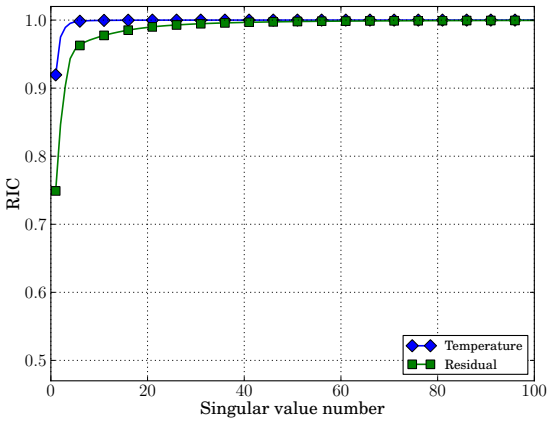
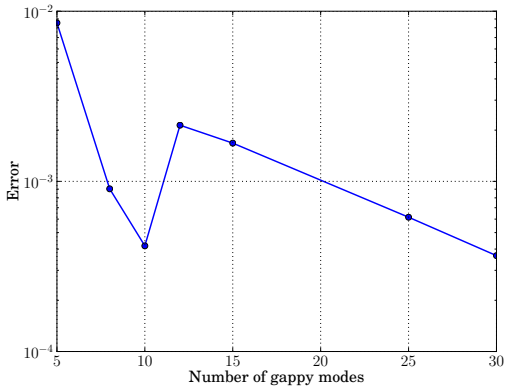


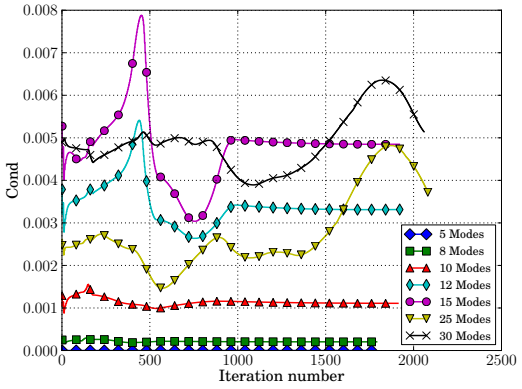
Figure 6.2: SVD spectrum for each of the involved terms.

Figure 6.3 displays the results obtained using $k = 4$ projection modes, and a varying number of gappy modes n_g and sampling points n_s with $n_s = n_g$. Even with a large number of gappy modes, i.e., $n_g = 30$, the error of the approximation is always greater than 1×10^{-4}

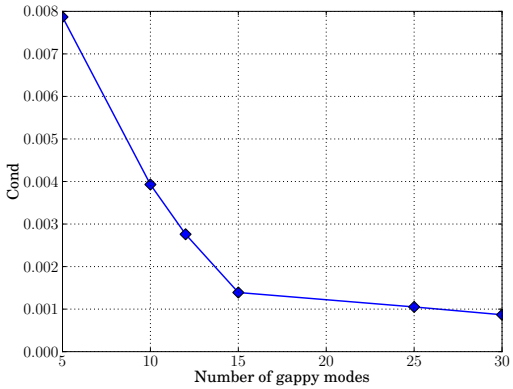
and therefore, the performance is not satisfactory. In this sense, the expectation of using a few sampling points to obtain a good approximation to the residual is not met. Moreover, the condition numbers of the tangent matrix and of the matrix $\widehat{\Phi}^T\widehat{\Phi}$ are quite small, which affects the robustness of the HROM.



(a) Relative error.



(b) Tangent matrix condition.

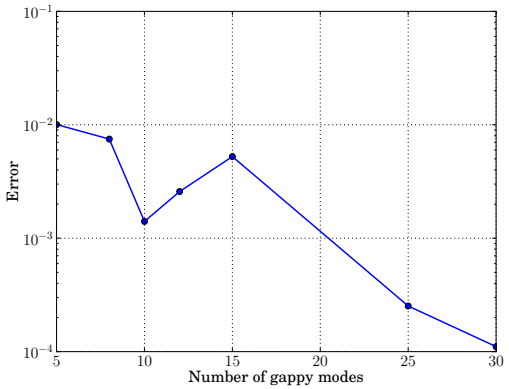


(c) Condition of $\widehat{\Phi}^T\widehat{\Phi}$.

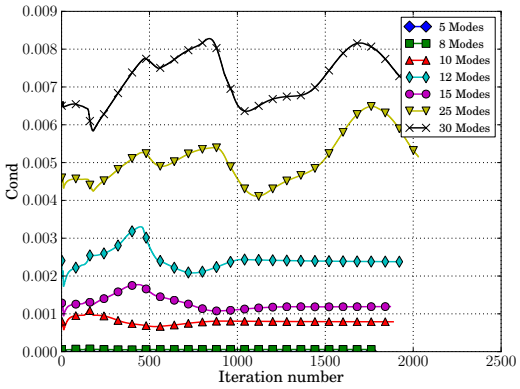
Figure 6.3: Results obtained by varying the number of gappy modes and sampling points with $k = 4$ and $n_s = n_g$.

To improve the performance of this HROM, we double the number of gappy points, i.e., we run the tests using $k = 4$ and $n_s = 2n_g$. As observed in Figures 6.4a, 6.4b and 6.4c, better results are obtained but the condition numbers of the involved matrices are rather small, and the error reached is almost 1×10^{-4} when using 60 sampling points, which is much more than expected from the SVD spectrum.

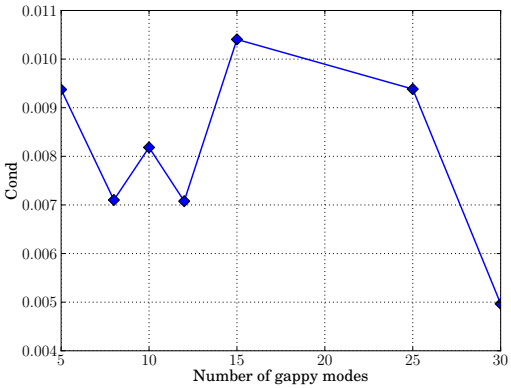
Finally, a speedup test was run, using $k = 4$, $n_g = 30$ and $n_s = 60$ which corresponds to an error of 1.10×10^{-4} and produces a speedup of 74.9.



(a) Relative error.



(b) Tangent matrix condition.



(c) Condition of $\widehat{\Phi}^T\widehat{\Phi}$.

Figure 6.4: Results obtained by varying the number of gappy modes and sampling points with $k = 4$ and $n_s = 2n_g$.

6.6.1.2 Results obtained by hyper-reducing the separate contributions to the residual

The cube solidification test is solved next using the HROM that separately hyper-reduces each contribution to the residual. Figure 6.5 displays the SVD spectrum for each of the involved terms in the formulation. Again, a good HROM approximation is expected using a small number of sampling points, because the spectrum shows high compression. As shown in the following, this expectation is confirmed by the experiments.

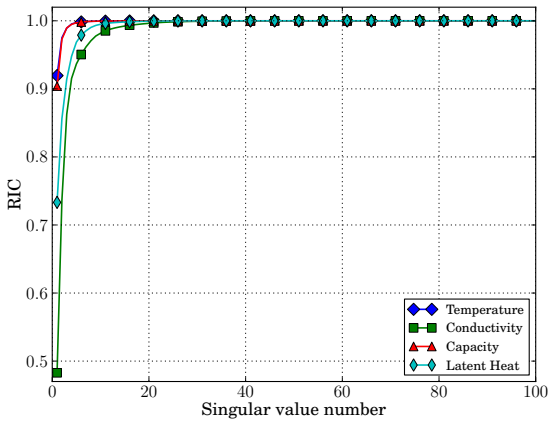
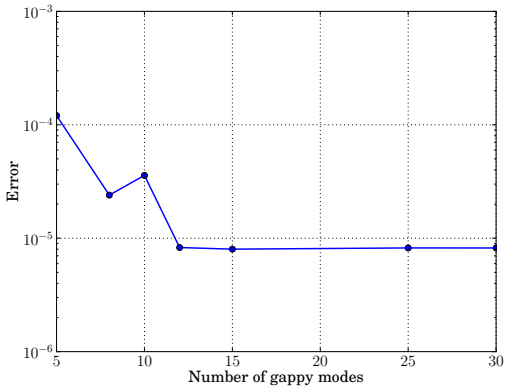


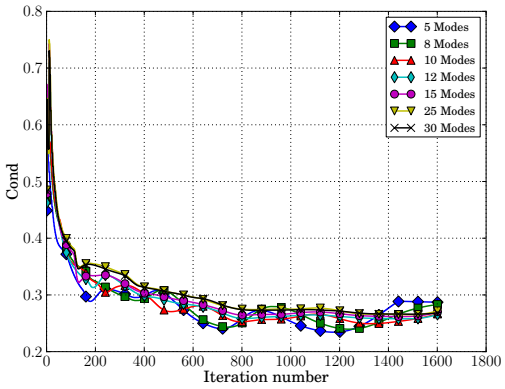
Figure 6.5: SVD spectrum for each of the involved terms.

Figure 6.6 plots the results obtained using $k = 4$ projection modes, and a varying number of gappy modes n_g and sampling points n_s with $n_s = n_g$. By comparing these results with those presented in figure 6.3, we conclude that this HROM performs much better than the previous one, not only because the number of gappy points required to reach a given error is smaller, but also because the condition of the involved matrices is larger, thus contributing to the robustness of the method. Moreover, using $k = 4$ and $n_g = n_s = 5$, a relative error of 1.20×10^{-4} is obtained with a speedup of 336.25. When using $k = 4$ and $n_g = n_s = 12$, the relative error is 8.27×10^{-6} and the speedup is 224.9.

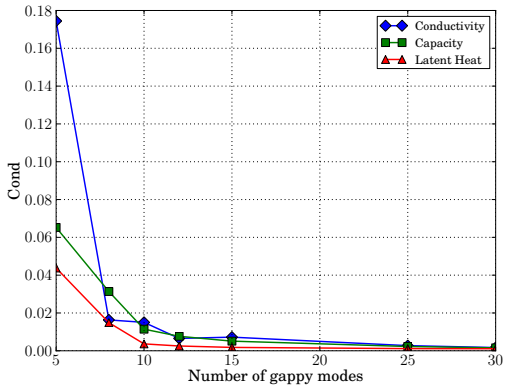
To further analyse the behaviour of this HROM, we run a test that augments the number of gappy points to $n_s = 2n_g$. The number of projection modes was held equal to $k = 4$, and the results can be observed in Figures 6.7a, 6.7b and 6.7c. The condition number of the involved matrices remain essentially unchanged. The computational error does not decrease significantly, making it necessary to increment the number of projection modes k if better results are required, e.g., by incrementing k to 8 and using $n_s = n_g = 12$, an error of 1.82×10^{-7} is obtained. We can conclude that in this case the error was governed by the number of projection modes and that the computation of the contributions to the residual were sufficiently accurate with $n_s = n_g$.



(a) Relative error.

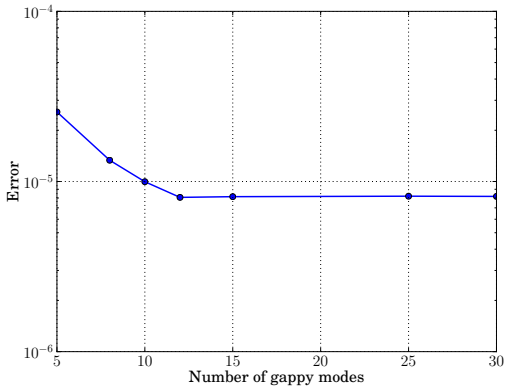


(b) Tangent matrix condition.

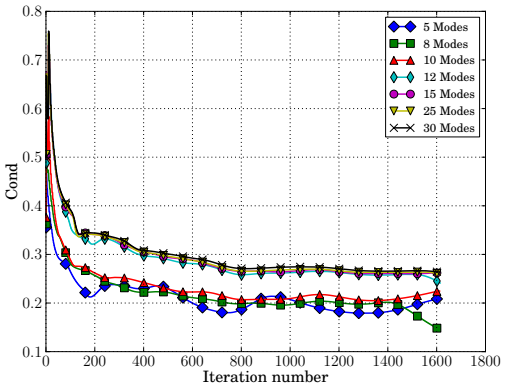


(c) Condition of $\widehat{\Phi}_i^T \widehat{\Phi}_i$.

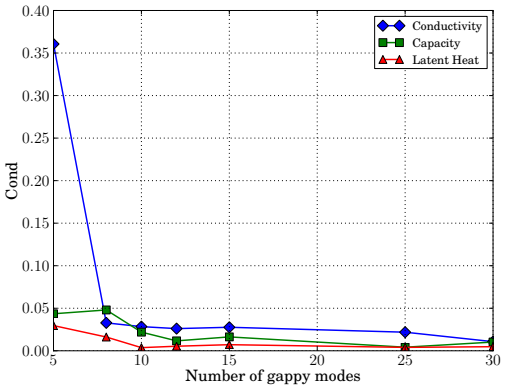
Figure 6.6: Results obtained by varying the number of gappy modes and sampling points with $k = 4$ and $n_s = n_g$.



(a) Relative error.



(b) Tangent matrix condition.



(c) Condition of $\widehat{\Phi}_i^T \widehat{\Phi}_i$.

Figure 6.7: Results obtained by varying the number of gappy modes and sampling points with $k = 4$ and $n_s = 2n_g$.

6.6.2 Welding of a tube

This problem consists of welding of a tube without material deposition. The external radius of the tube is 0.075m and the internal radius is 0.05615m. Symmetry conditions with respect to the $x-y$ plane are applied, and only one half of the tube is modelled with a length of 0.0377m. Robin boundary conditions are imposed at the internal and external surfaces. Figure 6.8 shows the discretisation composed of 388800 tetrahedra and 74100 degrees of freedom. A time step of 0.15s is used for the time interval $[0, 100]$ s. Temperature dependent thermophysical properties are used as specified in Figures 6.9b and 6.9a. The material density is $\rho = 4430\frac{\text{kg}}{\text{m}}$, the solidus temperature is 1877K, the liquidus temperature is 1933K, and the latent heat is $\mathcal{L} = 292600\frac{\text{J}}{\text{kg}}$. The heat source parameters are listed in Table 6.1. Initially, the heat source is located at the position given by the point $(-0.075, 0, 0)$. At the time instant $t = 0$, the heat source begins to travel with a constant tangential velocity equal to $5 \times 10^{-3}\frac{\text{m}}{\text{s}}$. The CPU time consumption required to obtain the HF solution was 11993.1 s. In the following, the reported speedups are obtained relative to this value.

Table 6.1: Parameters of the Goldak heat source

a_r	a_f	b	c
10.84mm	5mm	5mm	2mm
f_f	I	V	η
0.63131	260A	12V	0.7

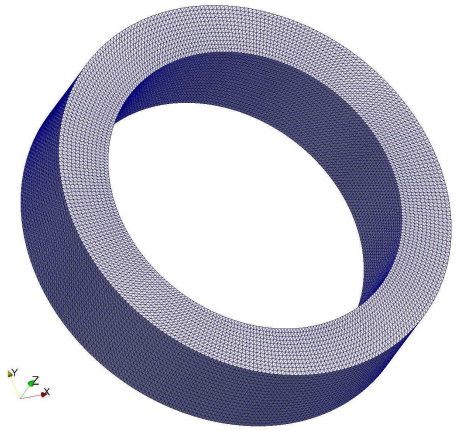
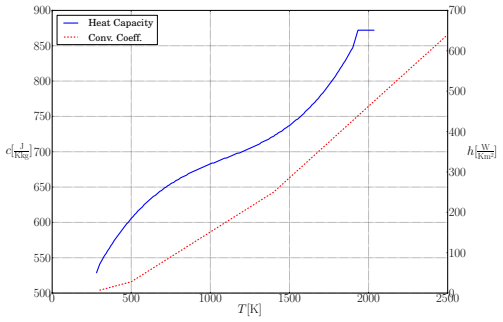


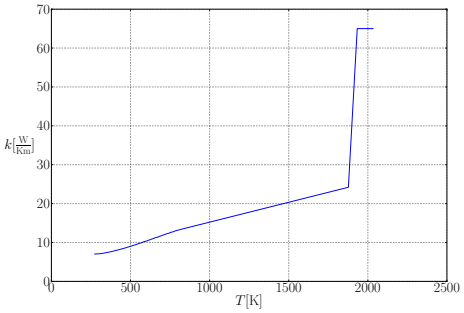
Figure 6.8: Tube discretisation.

6.6.2.1 Results obtained by hyper-reducing the residual

The welding problem is first solved by application of the HROM presented in Section 6.4.1 with a fixed reference frame. Figure 6.10 presents the SVD spectrum of the temperature and the residual histories. It can be observed that the spectrum of the residual is not as



(a) Convection/radiation coefficient and heat capacity.



(b) Heat Conductivity.

Figure 6.9: Thermophysical properties.

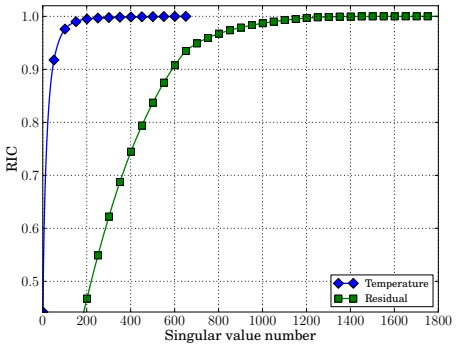
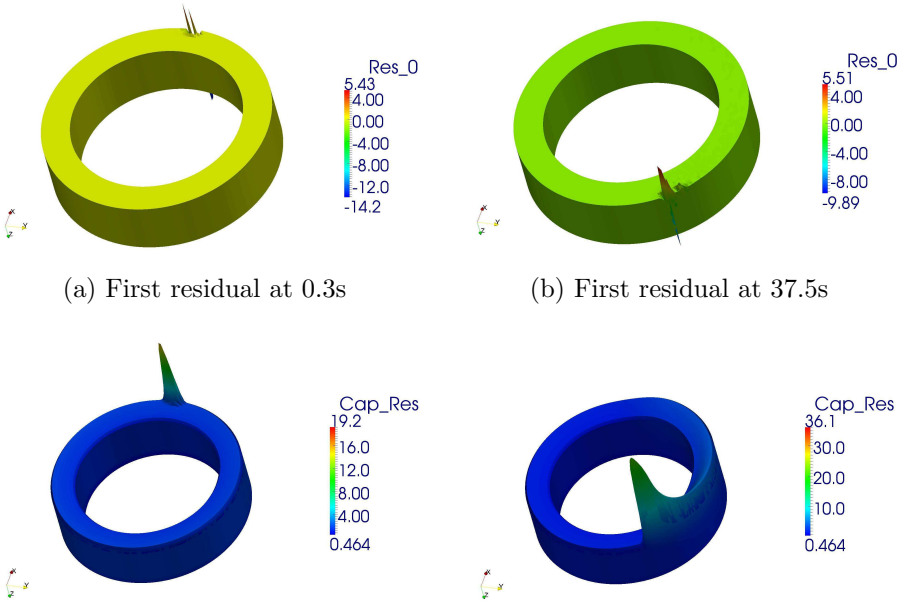


Figure 6.10: SVD spectrum for each of the involved terms.

compact as the spectrum of the temperature, which is an indication of an inconvenience in the HROM, implying that a large number of sampling/gappy points must be used to build good approximations of the residual. Therefore, this HROM must use a large number of sampling points to capture the residual behaviour. For example, to obtain a relative error of 0.0010254, $k = 180$ projection modes, $n_g = 1200$ gappy modes and $n_s = 2400$ sampling points are required, thus affecting the performance (the speedup factor is only 18.06).

In this case, the condition number of $\widehat{\Phi}^T\widehat{\Phi}$ is 0.2919, although the situation is worse for the conditioning of the tangent matrix. For the iterations of the Newton-Raphson scheme, the average, the maximum and the minimum value of this condition number are given by 2.04×10^{-5}, 4.82×10^{-5} and 1.70×10^{-5}. In this context, even if the Newton-Raphson scheme converges, it cannot be guaranteed that a good approximation to the solution is obtained with the HROM. The bad performance and the ill-conditioning of the tangent matrix are related to the fact that the snapshots of the residual are zero everywhere except for a small region, where the residual's behaviour tends to be noisy (see, e.g., the residual corresponding to the first iteration at two different time steps in Figures 6.11a and 6.11b).



(a) First residual at 0.3s

(b) First residual at 37.5s

(c) Heat Capacity at 0.3s

(d) Heat Capacity at 37.5s

Figure 6.11: Comparison of the quantities hyper-reduced by each of the introduced formulations. The visualisation used a deformation factor of 0.005 for wrapping the considered scalar quantities in the $-z$ direction.

6.6.2.2 Results obtained by hyper-reducing the separate contributions to the residual

The welding problem is solved next by the application of the HROM that separately hyper-reduces each contribution to the residual, with a fixed reference frame. Figures 6.11c and 6.11d show the heat capacity contribution to the converged residual at two different time steps. The pattern is less noisy than the residual that was used to build the projection basis in the previous HROM. Therefore, the resulting SVD spectrum for each of the involved terms in the formulation is more compact than in the previous HROM, as shown in Figure 6.12.

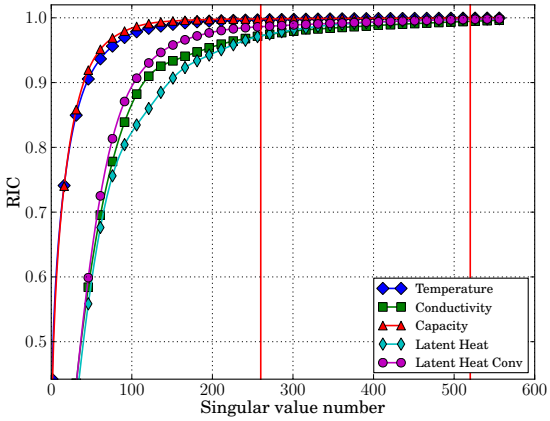
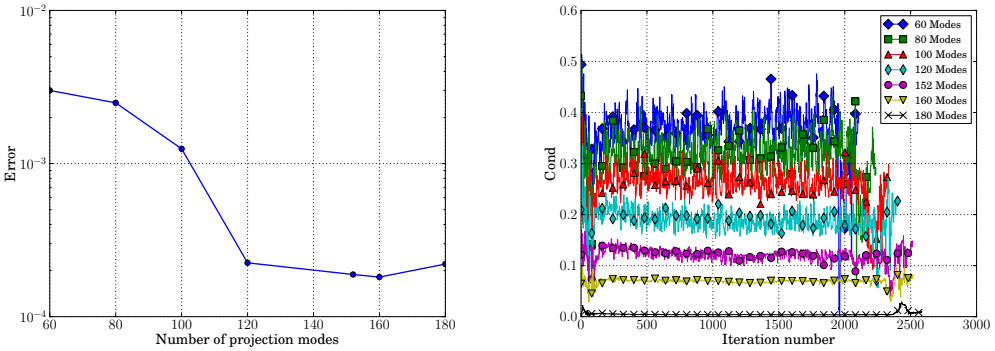


Figure 6.12: SVD spectrum for each of the involved terms.

In an initial test, we selected a number of modes and sampling gappy points that were both equal to 200, i.e., $n_g = n_s = 200$, and analysed the variation of the relative error in terms of the selected number of projection modes k. Figure 6.13a shows the evolution of this error. Initially, the error decreases when incrementing the number of projection modes but stagnates for $k > 120$. Figure 6.13b gives an explanation for this phenomenon. First, the condition number of the tangent matrix decreases with the increment of the number of projection modes, indicating that the problem condition deteriorates and errors increase. Even worse, the condition numbers of the matrices associated with the gappy data reconstruction process are also quite low: $\text{cond}(\widehat{\Phi}_c^T \widehat{\Phi}_c) = 5.75 \times 10^{-5}$ and $\text{cond}(\widehat{\Phi}_k^T \widehat{\Phi}_k) = 0.007$. The error induced by the interpolation of the internal forces (gappy data reconstruction) is therefore quite large; it deteriorates the condition number of the tangent matrix, and the algorithm is not able to decrease the computation error when augmenting the number of projection modes.

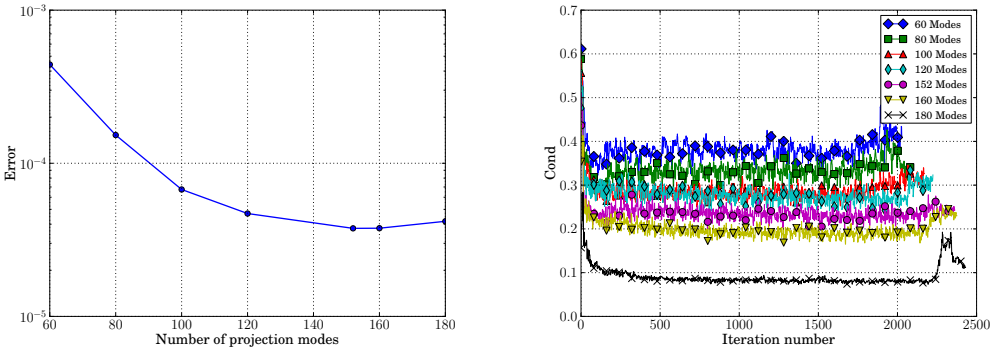
The same experiment was performed but with the doubling of the number of sampling points: $n_g = 200$ and $n_s = 400$. The results can be observed in figures 6.14a and 6.14b. The relative error of the solution is much smaller than before whereas the condition number of the tangent matrix increases with respect to the previous computation. In this case, the condition number of the matrices associated with the gappy procedure are $\text{cond}(\widehat{\Phi}_c^T \widehat{\Phi}_c) = 0.00104$ and $\text{cond}(\widehat{\Phi}_k^T \widehat{\Phi}_k) = 0.0686$, showing an increase of at least one order of magnitude with respect to the previous computation.

Next, we analyse how the HROM behaves when varying the number of modes and sampling points in the gappy data procedure. We leave the number of projection modes fixed at $k = 180$, and we progressively increment the number of gappy modes n_g and



(a) Relative error for $n_g = 200$ and $n_s = 200$. (b) Tangent matrix condition.

Figure 6.13: Results obtained by varying the number of projection modes for $n_g = 200$ and $n_s = 200$.

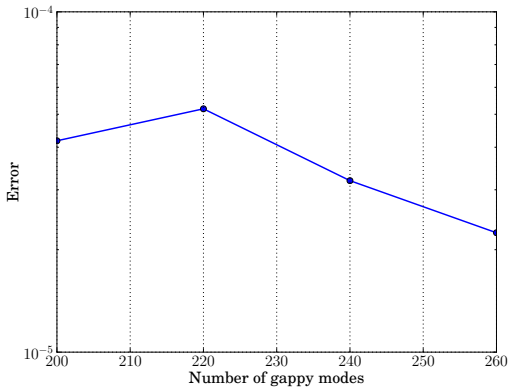


(a) Relative error for $n_g = 200$ and $n_s = 400$. (b) Tangent matrix condition.

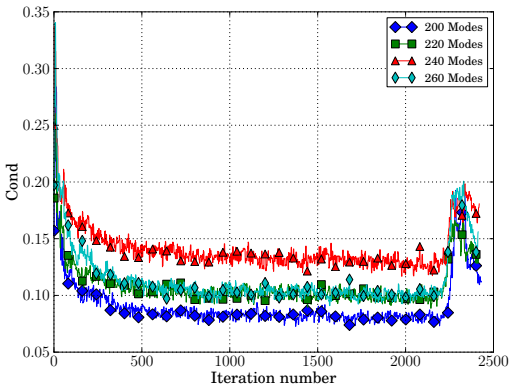
Figure 6.14: Results obtained by varying the number of projection modes for $n_g = 200$ and $n_s = 400$.

sampling points n_s while retaining the rule $n_s = 2n_g$. The results are shown in Figures 6.15a, 6.15b and 6.15c. In this case, the condition of the tangent matrix improves in response to an increment of the number of sampling points. However, the error of the approximation does not show any improvement because it is controlled by the number of projection modes.

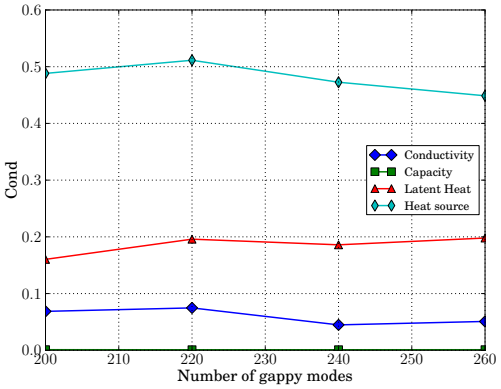
To assess the computational performance, we measured the speedup using $n_g = 260$ and $n_s = 520$ (vertical lines in Figure 6.12) for a varying number of projection modes. The result is shown in Figure 6.16: the speedup S_l decreases slightly when the number of projection modes is increased, attaining maximum and minimum values of 83.98 and 66.75, respectively.



(a) Relative error.



(b) Tangent matrix condition.



(c) Condition of $\widehat{\Phi}_i^T \widehat{\Phi}_i$.

Figure 6.15: Results obtained by varying the number of gappy modes and sampling points with $k = 180$ and $n_s = 2n_g$.

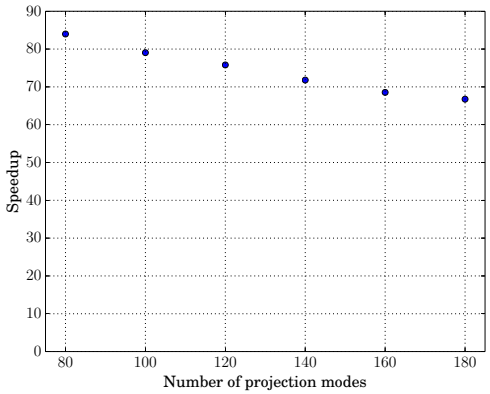


Figure 6.16: Speedup.

6.6.2.3 Results obtained with the moving frame approach

Finally, the welding problem is solved by application of the HROM that separately hyper-reduces each contribution to the residual using a moving reference frame attached to the heat source. Figure 6.17 shows the SVD spectrum for each term to be hyper-reduced. Comparing these spectra with those of Figure 6.12, it can be expected that the performance of this model would be improved because these spectra are more compact than the previous spectra.

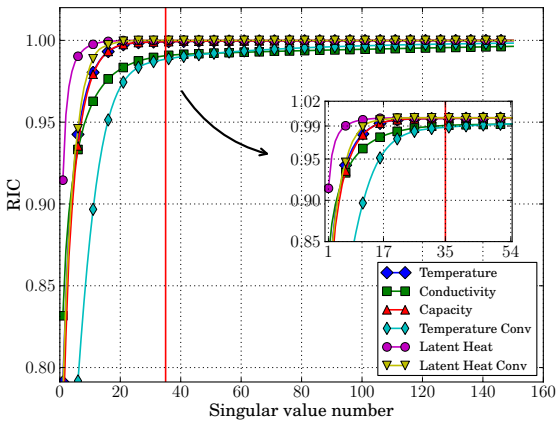
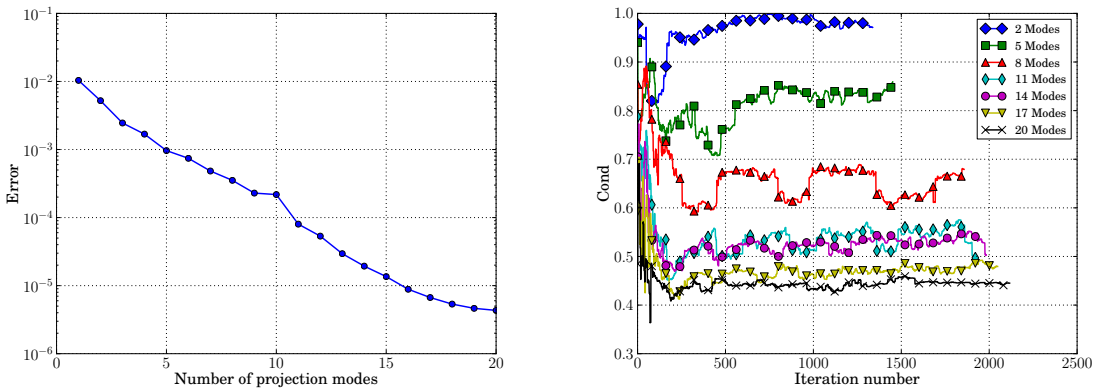


Figure 6.17: SVD spectrum for each of the involved terms.



(a) Relative error for $n_g = n_s = 35$.

(b) Tangent matrix condition.

Figure 6.18: Results obtained by varying the number of projection modes for $n_g = n_s = 35$.

When 20 modes are used for the projection, a RIC of 99.7% is obtained; this indicator shows that the dynamics can be captured with a lower number of modes. Based on this plot, a number of $n_g = n_s = 35$ gappy modes and sampling points were used to interpolate the different contributions to the residual (see the vertical line in Figure 6.17). Therefore, we analysed the variation of the relative error of the HROM in terms of the number of projection modes k from 1 to 20, holding $n_g = n_s = 35$. From the results shown in Figure 6.18, it can be concluded that the HROM performs quite well with an acceptable condition

number for the tangent matrix. The same can be said of the computational savings. The CPU time consumption for the HF solution was 9817.4 s. Figure 6.19 displays the variation of the speedup in terms of the number of projection modes k using $n_g = n_s = 35$. In this case the speedup, denoted by S_e, attains a maximum value of 246.9. This speedup is higher than that obtained in the previous case, particularly because the number of gappy modes and sampling points is much lower than before.

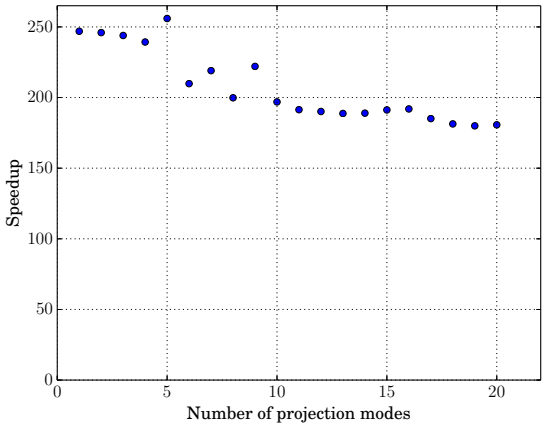


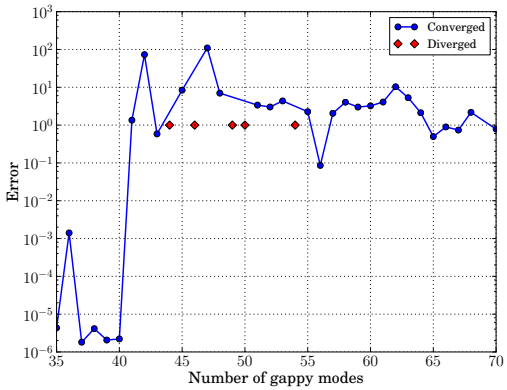
Figure 6.19: Speedup in terms of the number of projection modes for $n_g = n_s = 35$.

To investigate the limits of this formulation, we studied how the increment in the number of gappy modes and sampling points affects the performance if the number of projection modes is fixed. Clearly, the computational performance will be affected because we are incrementing the number of sampling points. However, the objective is to evaluate what happens with the relative error and the condition of the tangent matrix when varying the number of modes and sampling points in the interpolation of the contributions to the residual.

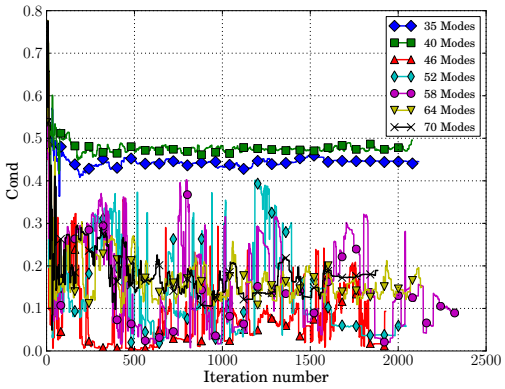
Figure 6.20a shows the variation of the relative error in terms of the number of gappy modes and sampling points ($n_g = n_s$) and always using $k = 20$ projection modes. It can be observed that for $n_g = n_s > 40$, the HROM solution displays a large error and does not converge in certain cases. This behaviour can be attributed to the fact that with the increment of the number of gappy modes, the condition of the tangent matrix and of the matrices cond($\widehat{\Phi}_i^T \widehat{\Phi}_i$) deteriorates, as shown in Figures 6.20b and 6.20c.

This situation is corrected using a greater number of sampling points, n_s, than gappy points, n_g. For instance, the number of sampling points is now selected as twice the number of gappy points ($n_s = 2n_g$). Figures 6.21a, 6.21b and 6.21c show that for $n_g \geq 42$ and $k = 20$, the error decreases almost monotonically with the number of selected gappy points. The increment in the number of sampling points decreases the error of interpolation in the gappy procedure, and the condition of the resulting matrices improves. Note that the error was already quite low for $n_g = 40$ and that by adding additional points, only a small decrease in the error is produced because no significant information is added to the model.

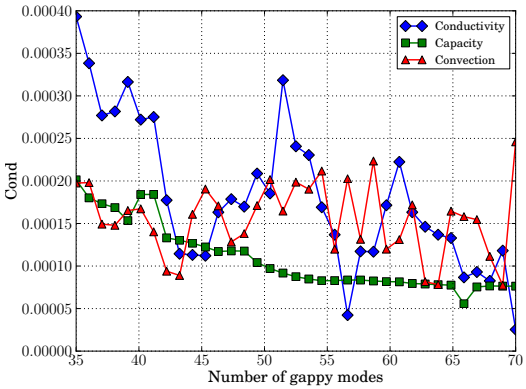
For completeness, we present in Figure 6.22 the location of the sampling points selected for the term $\boldsymbol{G}_k$ when using $n_g = n_s = 35$. We show that most of the sampling points are located notably close to the zone in which the high temperature gradients are produced, but several points are also almost uniformly distributed in the rest of the domain.



(a) Relative error.



(b) Tangent matrix condition.

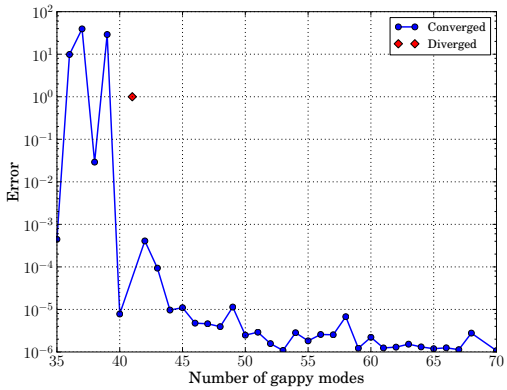


(c) Condition of $\widehat{\Phi}_i^T \widehat{\Phi}_i$.

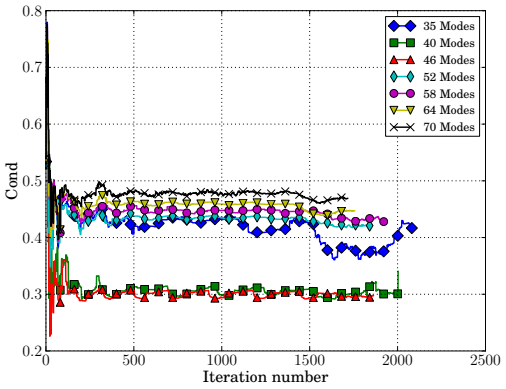
Figure 6.20: Results obtained by varying the number of gappy modes and sampling points with $k = 20$ and $n_g = n_s$.

The previous tests were intended to call the attention on the behaviour of each of the presented HROMs. Properties, such as, error, speedup, compressibility and conditioning, were studied for each HROM, showing the ability of each one of reproducing the results of the HF training model. In these tests parameter variation was not considered, that is, the problem solved by the HROM is the same as that used for training. For the present work this analysis suffices for pointing out the potential of each HROM, and more specifically for illustrating the different alternatives for ensuring the *compressibility* of the involved terms. Despite this scenario, it would be interesting to study how parameter variations affect the performance, more specifically the accuracy, of the presented HROMs. It is not the purpose of this work to give a complete answer to this matter, but to give an insight by analysing the error obtained with the current HROM when a set of parameters different than those of the training problem are used. Specifically, a welding problem with the following characteristics is run:

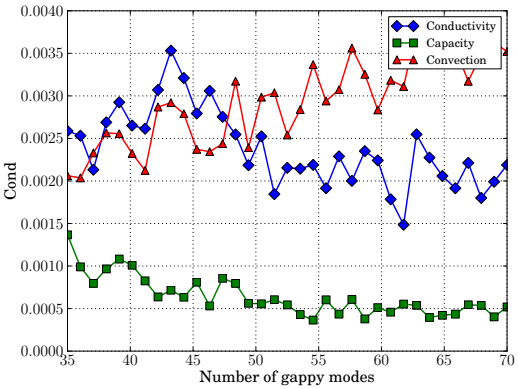
- Conductivity $k = 0.9k_r$,
- Heat capacity $c = 0.9c_r$,



(a) Relative error.



(b) Tangent matrix condition.



(c) Condition of $\widehat{\Phi}_i^T \widehat{\Phi}_i$.

Figure 6.21: Results obtained by varying the number of gappy modes and sampling points with $k = 20$ and $n_s = 2n_g$.

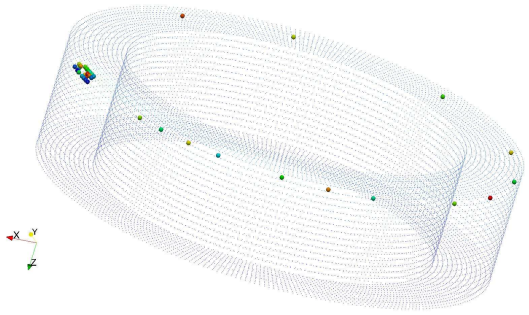


Figure 6.22: Sampling points selected for the term $\boldsymbol{G}_k$ using $n_g = n_s = 35$.

- Heat source velocity $v = 0.9v_r$,
- Heat source intensity $I = 0.9I_r$,

where the subscript r denotes the value of the considered property in the training problem. The other parameters that define the problem are the same as those of the training problem. The relative error obtained with the current HROM using $n_g = n_s = 35$ and varying the number of projection modes is shown in Figure 6.23. As it can be observed from the figure, the error is acceptable when recognising that no parameter sampling was accomplished for building the snapshots set, other than the training problem itself. It is interesting to observe that when the number of projection modes is greater than 11, the error keeps almost the same and no better results are obtained. That is, these extra modes are not contributing with significant information to the solution of this specific problem, at least when using $n_g = n_s = 35$. Although, notice that when running the training problem, the error keeps decreasing for a number of projection modes greater than 11 (see Figure 6.18a).

Remark: The last test shows that the current HROM has the ability to predict the solution to a problem different from the training problem. Although, a deeply study about this topic is needed and left as future work.

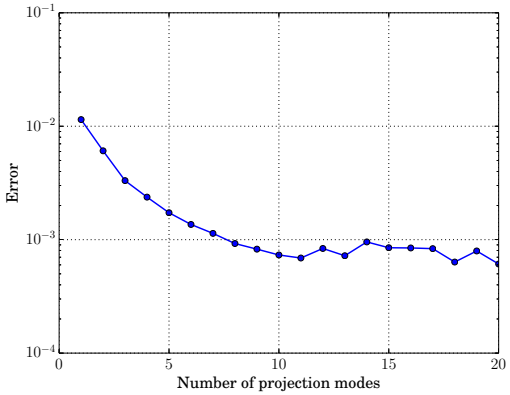


Figure 6.23: Relative error for $n_g = n_s = 35$.

6.7 Conclusions

In this chapter, different strategies were analysed for building Hyper-Reduced Order Models to solve nonlinear thermal problems, with applications for welding modelling. The considered methods are classified within the *a posteriori* techniques and are based on the Proper Orthogonal Decomposition approach.

Several aspects were examined. First, the manner in which the hyper-reduction is performed was considered. Two different methods were analysed: the hyper-reduction of the residual as a whole and a new proposal based on the separate hyper-reduction of the different terms contributing to the residual, in which the subdivision in terms is based on the physical characteristics of the problem. It was shown that the compressibility of the information was improved by the latter technique, allowing us to obtain good representations with a smaller number of gappy modes than with the former technique. As a result, faster computations could be performed, and the formulation also led to better conditioning of the nonlinear equations to be solved, thus improving the convergence properties.

A second aspect considered was the use of moving frames to simulate welding-like problems. A moving frame approach was applied in which the frame was attached to the concentrated heat source. The experiments showed that the moving frame approach increased the compressibility of the information with a further reduction in the number of modes required to obtain accurate approximations to the solution. Again, the conditioning of the problem was improved with better convergence properties.

The formulations are based on the choice of three parameters to obtain a hyper-reduced order model: (i) the number of projection modes, k; (ii) the number of gappy modes, n_g; and (iii) the number of sampling points, n_s. It was shown that these numbers must be selected such that $k \leq n_g \leq n_s$ to verify consistency. Several experiments were carried out to demonstrate the effect of varying these parameters. It was also demonstrated how these parameters must be chosen to avoid inconveniences produced by ill-conditioned equations.

Two three-dimensional nonlinear application examples were extensively covered: a solidifying cube and a welding-like problem in which a concentrated heat source travels to simulate a welding torch. Speedups of up to 250 times with respect to the high fidelity solutions were observed in these examples showing the potential of the proposed techniques.

Future work will be carried out to further study the ill conditioning of the matrices involved in the HROM formulation and to find a method that can detect and avoid problems in the computations. Additionally, the hyper-reduction of welding problems with material deposition will be analysed and ensuring the validity of the HROMs under variations of the parameters that define the problem will be addressed.

Chapter 7

Case of study: a nuclear power plant welded joint

This thesis has the particular objective of applying the developed tools for the prediction of Weld Residual Stresses (WRS) of a nuclear power plant welded joint. This type of application allows to test the robustness of the introduced numerical framework, and consequently improve existing techniques by answering the questions that emerge from the analysis of the obtained results. In this chapter, the welding process of the nozzle weld that connects the Reactor Pressure Vessel (RPV) and the cold leg of a nuclear reactor facility is studied and modelled with the numerical framework introduced in the preceding chapters. In Figure 7.1 details of the nozzle to be modelled can be observed.

Figure 7.1: Detail of the nozzle weld that connects the reactor pressure vessel and the cold leg.

The technological impact of this application example is of great importance, because the knowledge of WRS helps to predict the growth of long circumferential cracks in the welded pipe. Basically two mechanisms of failure can be identified in welded joints of nuclear power plant structures: failure by fatigue, and failure by Stress Corrosion Cracking

(SCC). The worst case scenario is stated by the SCC, mechanism that is highly influenced by the WRS. The objective of this chapter is the prediction of WRS by simulating the whole welding process. Modelling of Post Weld Heat Treatment (PWHT) is left as future work.

The chapter is organised as follows. First a description of the problem is given. In Section 7.2, details of the numerical model are provided. The thermal and mechanical properties of the involved materials are specified as part of Section 7.3. In Section 7.4 the obtained results are exposed and discussed. Finally, in Section 7.5 the conclusions of the current chapter are given.

7.1 Problem description

As it was already mentioned the problem consists in the simulation of the welding process of the nozzle weld that connects the RPV and the cold leg. Initially, a cladding layer of approximately 5mm is deposited in the interior of the pipe. Once the cladding deposition is finished, a thermal treatment is applied in order to remove the generated residual stresses. Therefore, it is considered that the cladding process does not contribute to residual stresses, and that the initial configuration is a stress free configuration. Before the welding of the nozzle and the pipe is started, a preheating stage takes place rising the temperature of the zone to be welded from ambient temperature to 160 °C. In order to model the preheating stage, an initial homogeneous temperature equal to 160 °C is taken for the whole domain.

In what follows details of the problem to be modelled are given. A longitudinal section of the involved pipes can be observed in Figure 7.2, where the inner diameter of the nozzle and the pipe is 750mm. Details of the geometry and the welding plan can be observed in Figure 7.3a. A macrograph specifying the material denomination of each of the components of the weld is shown in Figure 7.3b. Based on the dimensions characterising the problem, it must be noted that a three dimensional analysis would be very computational intensive making this option prohibitive and impractical. That is why a two dimensional axisymmetric analysis seams more reasonable for the current case. The error introduced by the adoption of this hypothesis, is not high if it is considered that the longitudinal section under analysis is away from the starting and ending points of the welding process.

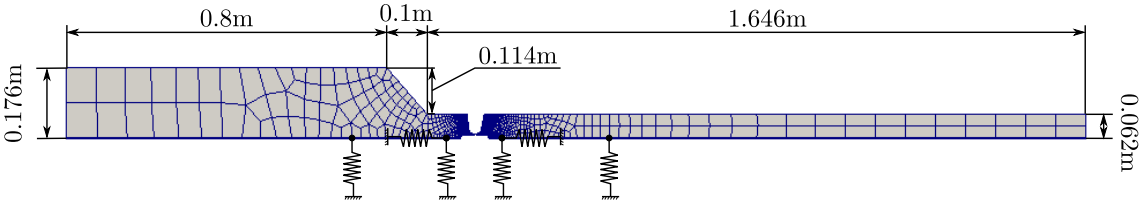
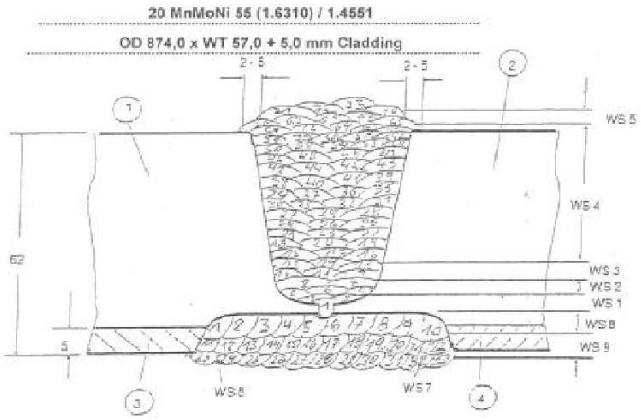
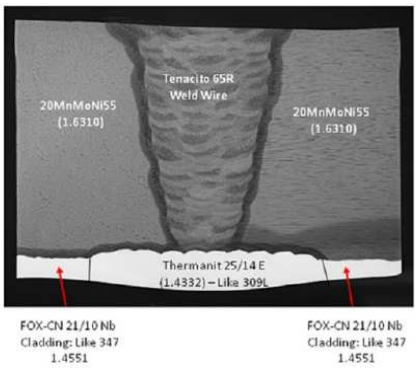


Figure 7.2: Longitudinal section of the involved pipes. Specification of the domain and mesh to be used in the two dimensional analysis of the problem.

The boundary conditions of the thermal problem are given by Robin boundary conditions at the internal and external faces of the nozzle and the pipe, including the evolving boundary of the deposited weld beads. Both ends of the tube are modelled with adiabatic boundary conditions. In the mechanical problem, free traction boundary conditions are imposed everywhere, and the system is supported by means of the springs depicted in Figure 7.2. This is justified in the observation that in real facilities, in order to deal with



(a) Graphical longitudinal section of the welding plan and expected final geometry.



(b) Specification of the materials used in each region of the weld.

Figure 7.3: Details of the problem: welding plan and material denominations.

thermal deformations every structural component of the plant is elastically supported to the RPV by means of springs and bellows pipes. Examples of this supporting system are observed in Figures 7.4a-7.4b. The springs used for the numerical analysis of the welding process, have a very low stiffness constant and are used to avoid singularities.

(a) Spring system supporting a pipe.

(b) Bellows pipe system.

Figure 7.4: Details of a spring supporting system.

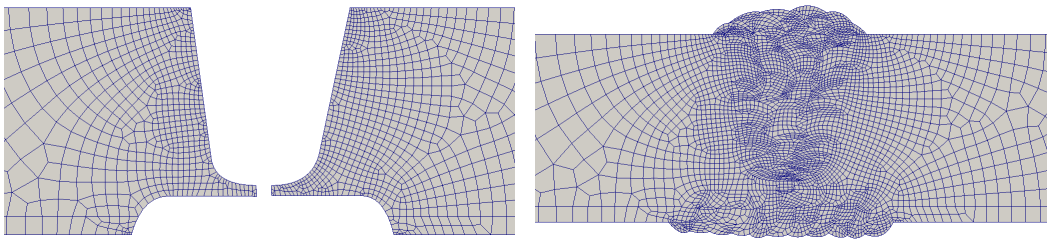
7.2 Details of the numerical model

The welding problem is numerically solved by making use of a staggered thermo-mechanical scheme, where in each time step, the thermal problem is solved first, and the mechanical problem is solved next by taking as thermal state the previously computed temperature

field. The thermal problem is considered to be decoupled from the mechanical problem, because the energy variation generated by mechanical deformations is negligible when compared to the heat input to the medium given by the welding torch.

No microstructural changes were taken into account. The contribution of solid/liquid transformations is computed as part of the thermal problem by making use of the linear triangular element presented by Fachinotti *et al.* in [28]. It was observed by Mullins and Gunnars [99] that when computing weld residual stresses in pipe welds, an isotropic hardening model approximates better experimental measurements than a kinematic or mixed hardening model. Following this observation, the mechanical behaviour of the material is described by an isotropic elasto-plastic hardening model. Quadrilateral Q1/P0 elements are used to solve the mechanical problem. Material deposition is handled by making use of the *inactive* element approach where, additionally, the concept of stress free configurations is taken into account.

As depicted by Figure 7.5a the domain of the problem was discretised using 4722 quadrilaterals and 4885 nodes. At the same time the welding plan specified by Figure 7.3a was taken into account, and accordingly meshed as shown in Figure 7.5b. For the thermal problem, the triangular elements were generated from the quadrilateral mesh by dividing each of them into two triangles by the biggest angle criterion. A mesh of line segments defining the skin elements for the thermal Robin boundary condition was built and considered in the computations. This task requires some careful work, as skin elements associated to weld beads that are being deposited must be activated and subsequently deactivated as new weld beads are deposited. Figure 7.6 shows the different skin elements used as weld bead boundaries, and the order in which each weld bead is deposited, information that is specified by the numbers inside each weld bead.



(a) Detail of the mesh without weld beads deposited.

(b) Detail of the mesh with all the weld beads deposited.

Figure 7.5: Details of the domain discretisation.

In this two dimensional axisymmetric analysis, a new layer of material or weld bead is deposited each time the weld torch crosses the section under analysis. Then, the material deposition and the heat input to the medium can be modelled by means of a simplified version of the Goldak heat source [66] as similarly done in [42, 142]. In this heat source model, it is supposed that the total heat power is uniformly distributed across the area Ω_L of the section of the last added layer of material. Then, heat input to the medium is given by the expression

$$Q_s(t) = \frac{\sqrt{3}Q}{\sqrt{\pi}\Omega_L} \begin{cases} \frac{f_f}{c_f} \exp\left(-3\frac{\theta(t)^2}{c_f^2} \right) \text{ if } \theta(t) \leq 0 \\ \frac{f_r}{c_r} \exp\left(-3\frac{\theta(t)^2}{c_r^2} \right) \text{ if } \theta(t) > 0 \end{cases} \tag{7.1}$$

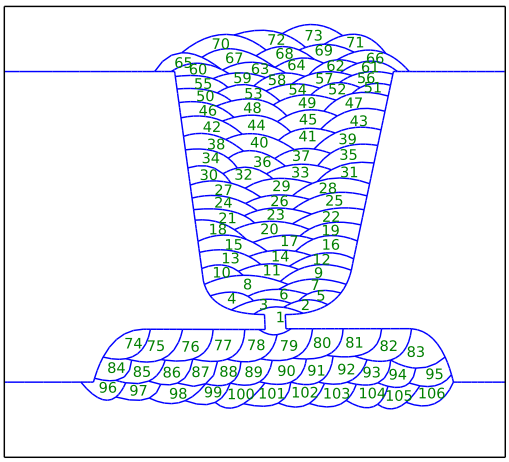


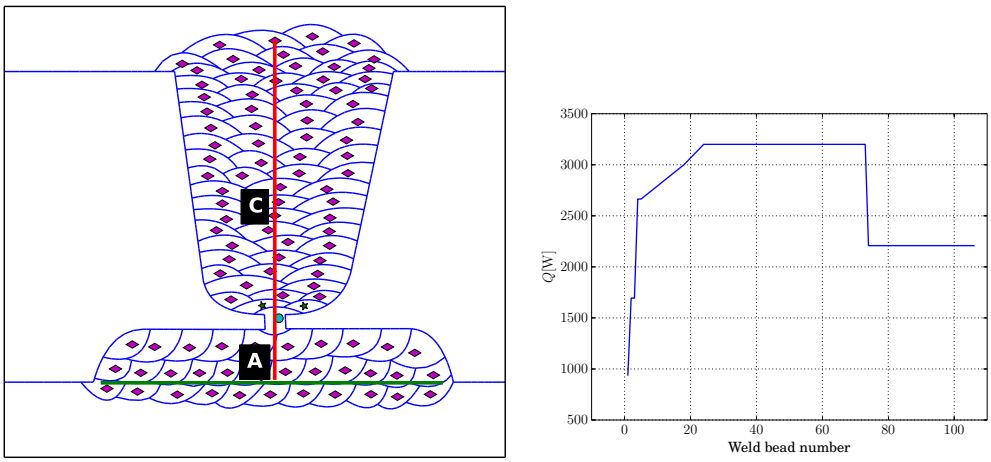
Figure 7.6: Skin elements used as part of the Robin boundary condition (take into account that these skin elements have associated activation and deactivation times). The numbers are used to denote the order in which weld beads are deposited, serving as definition of the weld plan.

where c_f and c_r are the length parameters associated with the axis of the front and rear semi-ellipsoids, f_f and f_r are the portions of heat distributed in the front and rear semi-ellipsoids, and Q is the total heat input. The function $\theta(t)$ gives the variation of position of the heat source in the circumferential direction. For the current application problem the heat source parameters where taken as $c_f = 5 \times 10^{-3}$m, $c_r = 4c_f$, $f_f = 2c_f/(c_f + c_r)$ and $f_r = 2 - f_f$. The tangential velocity of the heat source and the heat source parameter Q can be observed in Figures 7.7a and 7.7b.

The time increment Δt used in the time discretisation must be small enough in order to ensure that the heat source is well integrated. In order to correctly capture the behaviour of the heat source, it must traverse the section under analysis in 10 to 20 time steps [142, 143]. In other words, when the heat source is crossing the analysed section the time increment should be of the order

$$\Delta t = O\Big(\frac{c_f + c_r}{nv}\Big), \tag{7.2}$$

where n takes a value in the interval $n \in [10, 20]$ and v denotes the tangential velocity of the heat source. Once the heat source has crossed the section of analysis, the time increment can be increased. In this work the parameter n was taken equal to 10 and the time increment was linearly increased up to a maximum of 100s.



(a) Heat source velocity at each weld bead. Circles: 0.83mm/s. Stars: 1.86mm/s. Diamonds: 2mm/s.

(b) Specification of the heat source parameter Q as a function of the bead weld number.

Figure 7.7: Details of the heat source parameters.

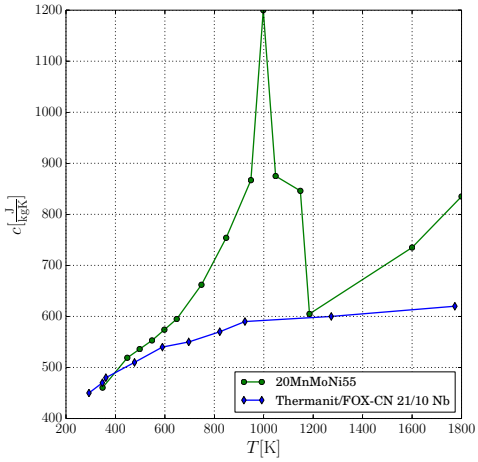
7.3 Material properties

As previously observed in Figure 7.3b, the material of the nozzle and the pipe is the alloy 20MnMoNi55, the weld metal material is Tenacito 65R, the cladding material is FOX-CN 21/10 Nb and the back weld metal is Thermanit 25/14E. In the case of 20MnMoNi55 and Tenacito 65R, it is supposed that both materials share the same thermal and mechanical properties except for the curves σ-α describing the plastic response.

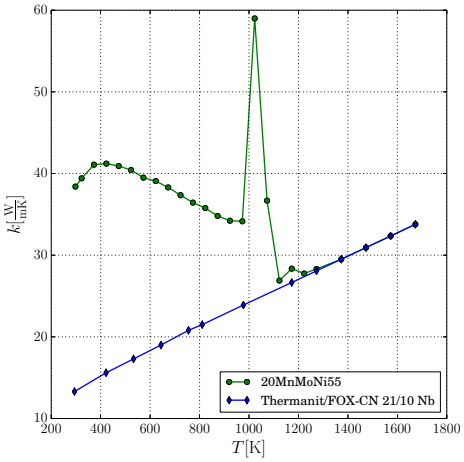
The materials designations 20MnMoNi55 (1.6310) and Tenacito 65R are similar to the US A508 material, the designation Thermanit 25/14E (1.4332) is similar to the steel 309L, and the material FOX-CN 21/10 Nb is similar to the steel 347. In what follows the thermal and mechanical properties of these materials are specified. In Table 7.1 the reference temperature T_{ref}, and the material parameters given by the zero strength temperature T_{zst}, the mass density ρ (taken as constant), the melting temperature T_m and the latent heat $\mathcal{L}$ can be observed for each of the considered materials. The mass density for the materials 20MnMoNi55 and Thermanit 25/14E were estimated from the Material Property Database [144]. The material parameters ρ, T_m and $\mathcal{L}$ were taken from [145]. The material parameters T_m and $\mathcal{L}$ for the other materials were estimated from data available at the website of `Code Aster` (`www.code-aster.org`). The parameter T_{zst} was taken as 98% of the melting temperature T_m. The other material properties are given in Figures 7.8a, 7.8b, 7.9b, 7.10a and 7.10b. The heat convection coefficient, specified in Figure 7.9a, was supposed to be equal to the one presented in [19]. The Voce expression is adopted as hardening law. Accordingly, the curves σ-α are given in Figures 7.11a, 7.11b, 7.12a and 7.12b were built by fitting the experimental curves. This fitting was done using the function `curve_fit` of the `python` package `scipy.optimize` [146].

Table 7.1: Some of the material parameters

Material	T_{zst}	T_{ref}	ρ	T_m	$\mathcal{L}$
20MnMoNi55	1715K	423K	$7870\frac{\text{kg}}{\text{m}^3}$	1750K	$273190\frac{\text{J}}{\text{kg}}$
FOX-CN 21/10	1639.5K	423K	$7750\frac{\text{kg}}{\text{m}^3}$	1673K	$271960\frac{\text{J}}{\text{kg}}$
Thermanit 25/14E	1592.5K	423K	$7980\frac{\text{kg}}{\text{m}^3}$	1625K	$276940\frac{\text{J}}{\text{kg}}$

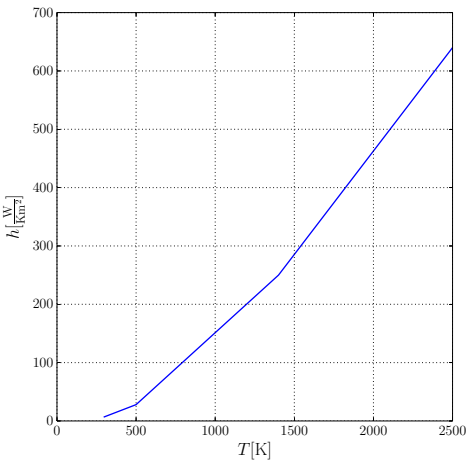


(a) Temperature dependent heat capacity.

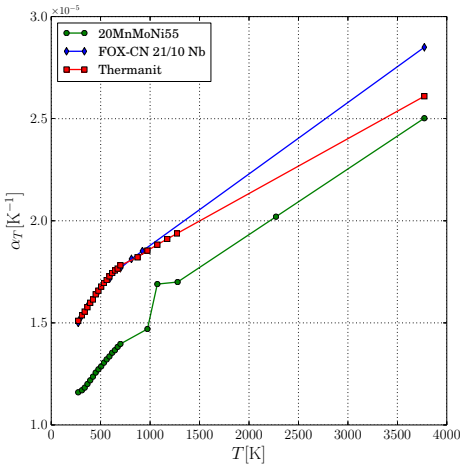


(b) Temperature dependent heat conductivity.

Figure 7.8: Thermal properties specification: heat capacity and heat conductivity.

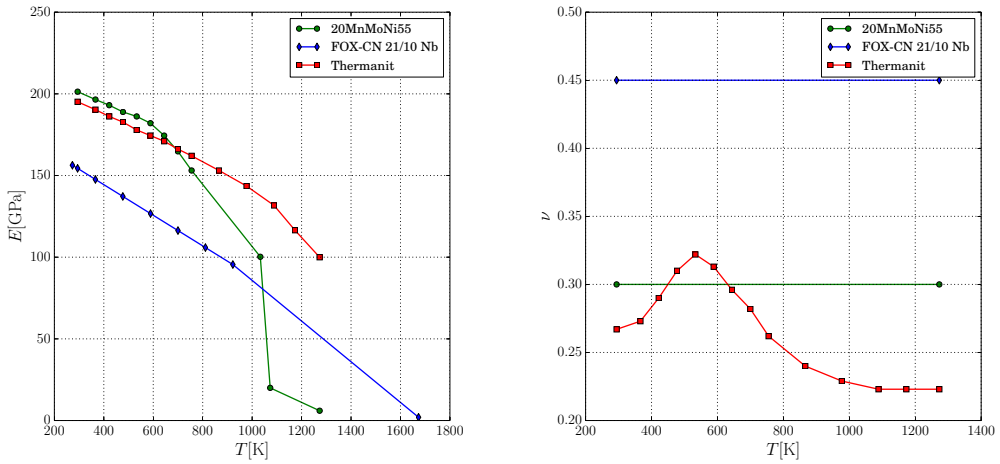


(a) Temperature dependent heat convection coefficient.



(b) Temperature dependent thermal expansion coefficient.

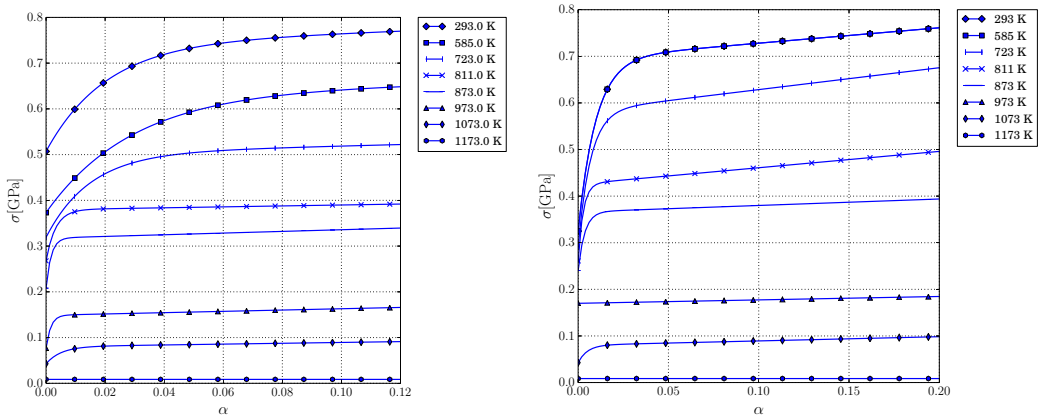
Figure 7.9: Heat convection coefficient and thermal expansion coefficient specification.



(a) Temperature dependent Young's modulus.

(b) Temperature dependent Poisson ratio.

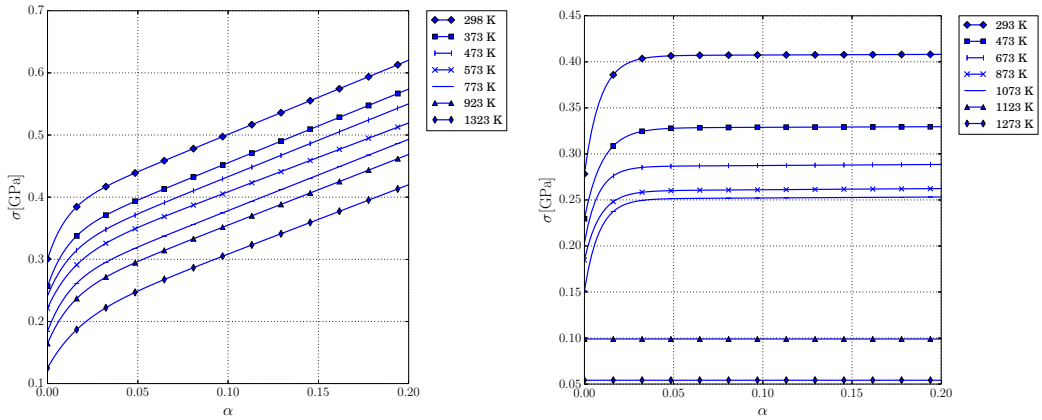
Figure 7.10: Mechanical properties specification: Young's modulus and Poisson ratio.



(a) Temperature dependent σ/α curves for the 20MnMoNi55 alloy.

(b) Temperature dependent σ/α curves for the weld metal.

Figure 7.11: Mechanical properties specification: σ/α curves for the materials 20MnMoNi55 and Tenacito 65R.



(a) Temperature dependent σ/α curves for the FOX-CN 21/10 Nb alloy.

(b) Temperature dependent σ/α curves for the Thermanit alloy.

Figure 7.12: Mechanical properties specification: σ/α curves for the materials FOX-CN 21/10 Nb and Thermanit 25/14E.

7.4 Results

The simulation was run up to attaining a maximum temperature of 302K. Approximately 12100 time steps were computed, taking an approximate computation time of 5887.4s = 1h 38min in a machine with the following specifications using four threads:

- Processor: Intel(R) Core(TM) i7 CPU 930 @ 2.80GHz.
- Number of physical cores: 4 cores; hyper-threading deactivated.
- Cache L1, L2, L3: 32KB, 256 KB, 8MB.
- Memory: 6GB.
- Operating system: `Linux 2.6.34.10 #1 SMP x86_64 GNU/Linux`.

The exportation of data to ParaView, a post-processing software [147], took approximately 1025.9s = 17min.

In Figures 7.13, 7.14 and 7.15 temperature profiles at different time instants of the welding process are shown. From these results it can be observed that the predicted temperature field is of good quality from the numerical point of view, and the heat source contribution is uniformly distributed as expected. In Figure 7.16 the amount of plastic distortion characterised by means of the equivalent plastic deformation α is shown. In the subsequent figures, specifically Figures 7.17, 7.18, 7.19 and 7.20, the components of the stress tensor can be observed, which specify the WRS at the end of the welding process.

In order to determine the influence of the solid/liquid phase change in the computed residual stresses, a simulation where no phase change is taken into account was performed. For comparison purposes the residual stress distribution along lines **A** and **B** depicted in Figure 7.7a was studied. The axial and hoop stresses along line **C** can be observed in Figures 7.21a and 7.21b, whereas these stresses along line **A** are shown in Figures 7.22a and 7.22b. A first conclusion is that solid/liquid phase change does not contribute in a considerable manner to the final residual stresses. This is an interesting finding that allows to simplify the thermal problem considerably, thus reducing the computation time. For example, in two dimensional problems the use of triangular elements is no longer needed, making possible to use the cheaper quadrangular elements. The same is true for three dimensional problems, were thermal tetrahedral elements can be replaced by thermal hexahedral elements. Take into account that in three dimensional problems, when using tetrahedral elements, the element count is six times larger than the number of required hexahedral elements. To understand why it is necessary to use triangles and tetrahedra when solving thermal problems with solid/liquid phase change, remember that in order to predict the position of the phase change boundary for a given set of nodal temperatures, the field should be interpolated linearly otherwise this estimation is not possible or, at least, much more complicated [30]. Another evident advantage of not considering the solid/liquid phase change is the increase in robustness.

Remark: the thermo-mechanical framework used in this work is not able to predict a displacement field compatible with the computed mechanical deformation. This is because mechanical deformation is calculated with respect to a natural configuration instead of the reference configuration. Remember that in this case the reference configuration is defined in terms of the geometry of the whole welding plan, that is built at the beginning of the computation without taking into account the (unknown) distortion suffered by the elements as the welding plan advances. Finding a displacement field compatible with the computed mechanical deformations is left as future work.

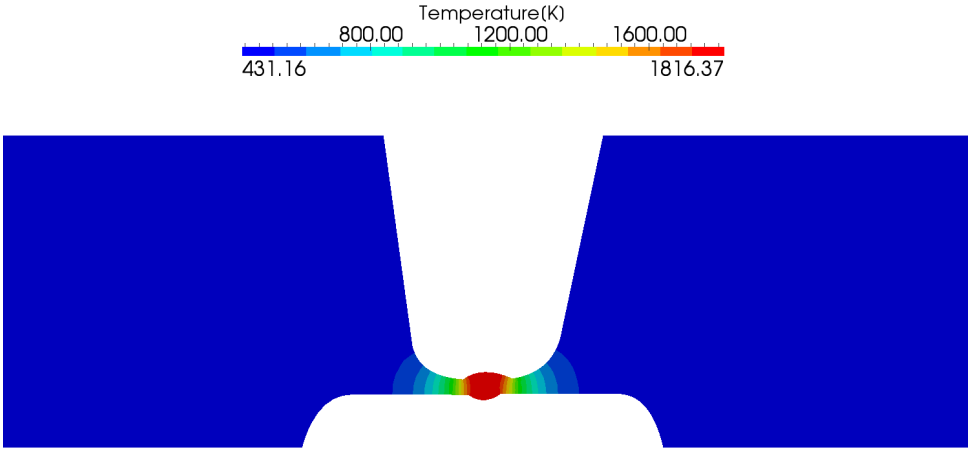


Figure 7.13: Temperature profile at the beginning of the process.

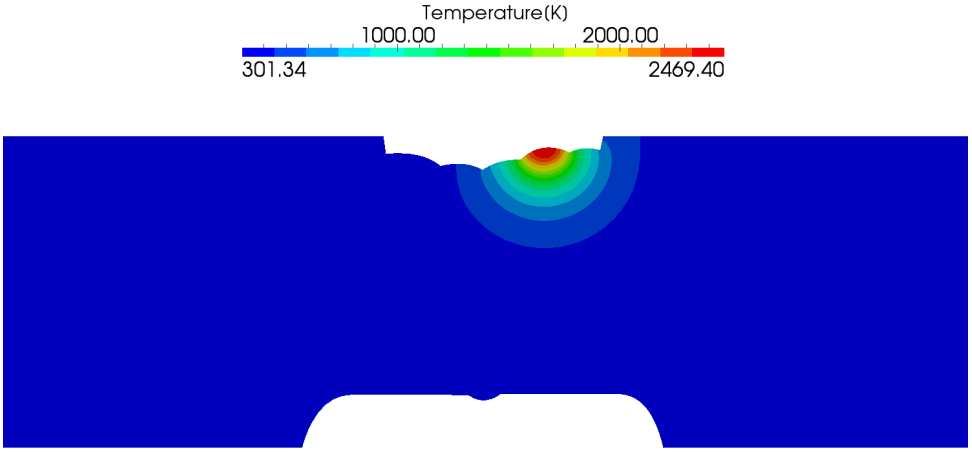


Figure 7.14: Temperature profile at the middle of the process.

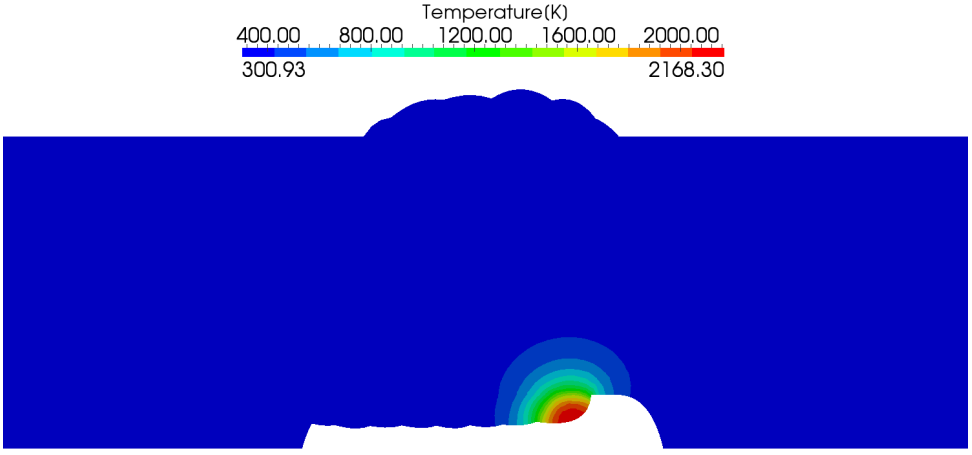


Figure 7.15: Temperature profile when depositing a weld bead at the back weld.

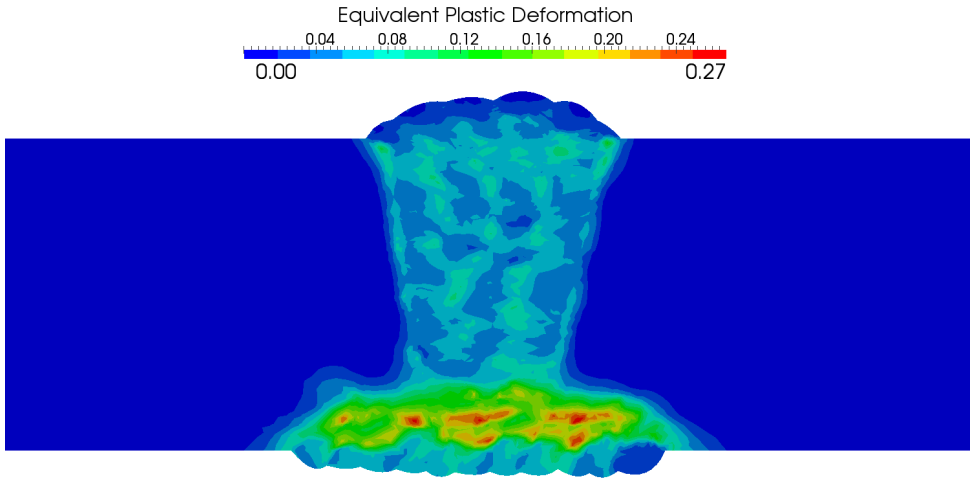


Figure 7.16: Resultant equivalent plastic deformation.

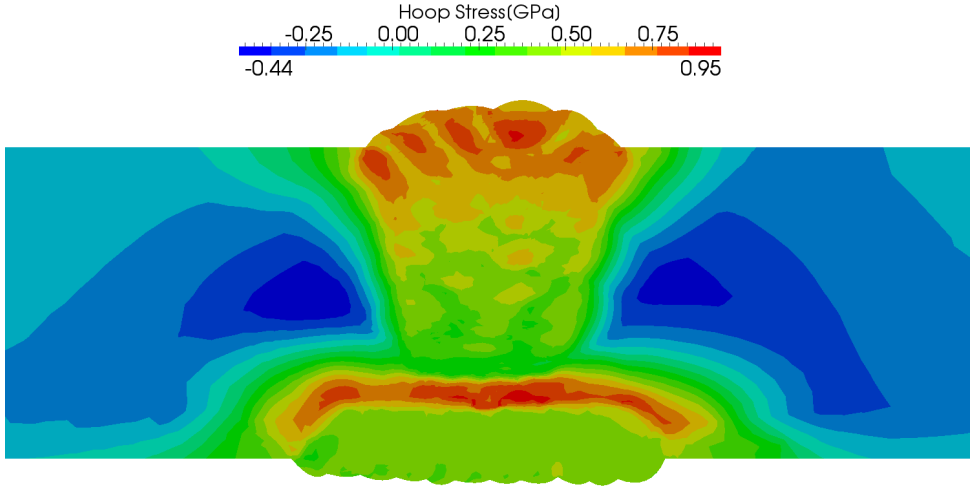


Figure 7.17: Residual stresses: Hoop stress.

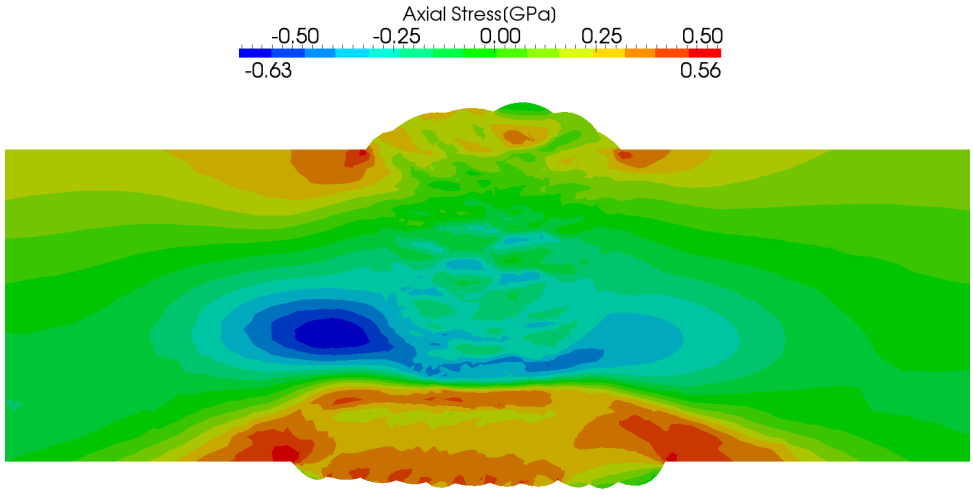


Figure 7.18: Residual stresses: Axial stress.

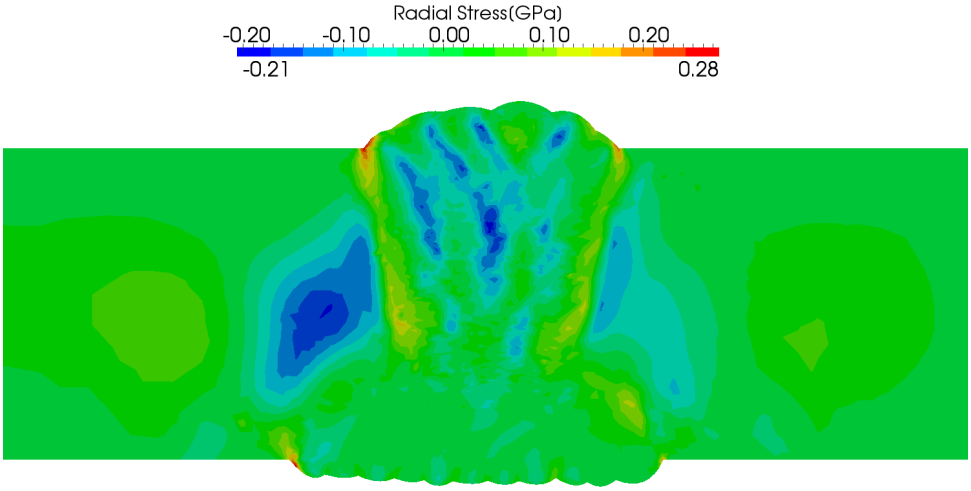


Figure 7.19: Residual stresses: Radial stress.

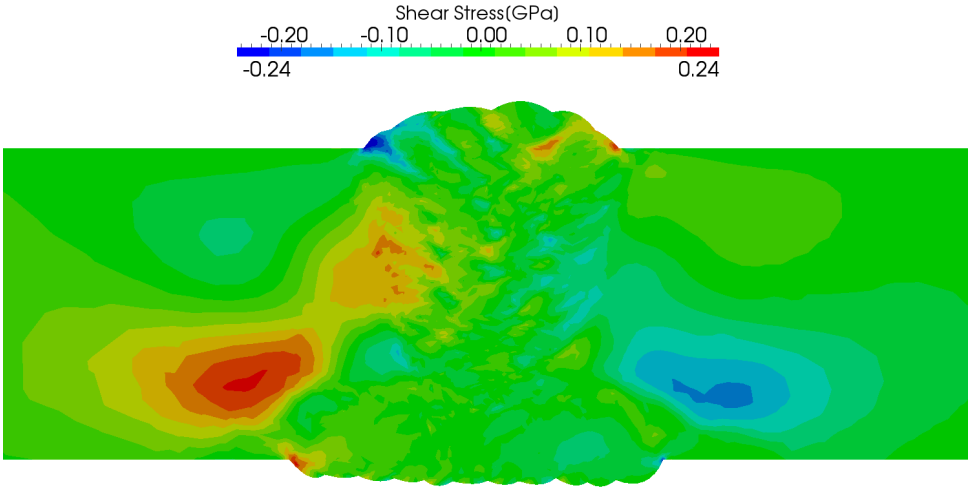
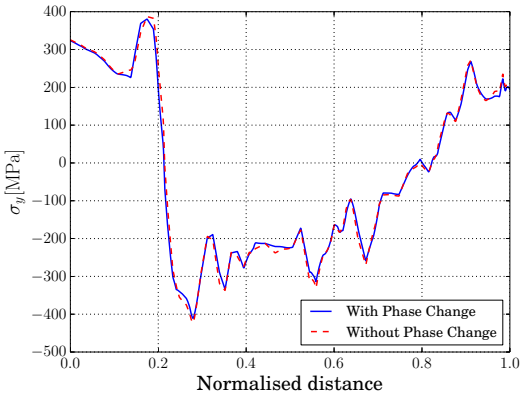
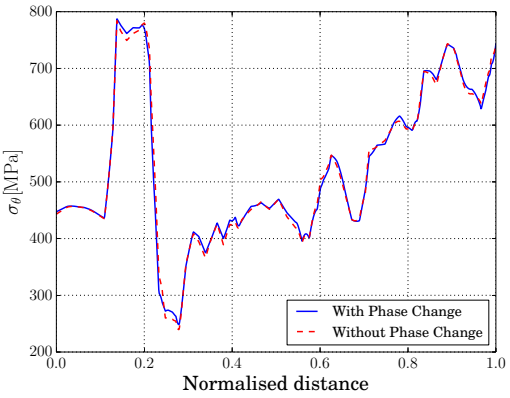


Figure 7.20: Residual stresses: Shear stress.

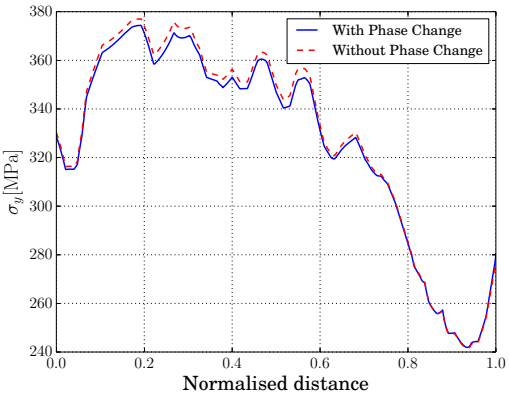


(a) Axial stress along line **C**.

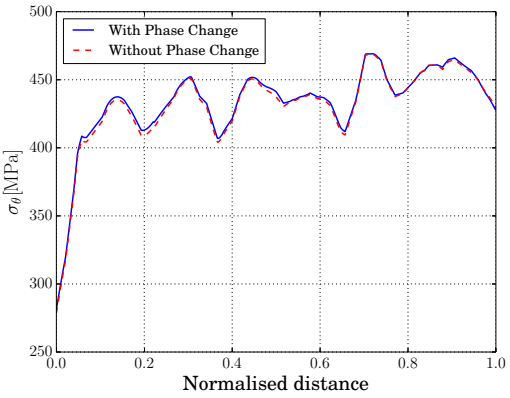


(b) Hoop stress along line **C**.

Figure 7.21: Residual stress components along the line **C**. Comparison of results obtained computing the contribution of solid/liquid latent heat and without it.



(a) Axial stress along line **A**.



(b) Hoop stress along line **A**.

Figure 7.22: Residual stress components along the line **A**. Comparison of results obtained computing the contribution of solid/liquid latent heat and without it.

7.5 Conclusions

In this chapter the particular technological objective of applying part of the developed tools to the prediction of weld residual stresses of a nuclear power plant welded joint was treated. Specifically, the simulation of the welding process of the nozzle weld that connects the RPV and the cold leg of a nuclear reactor facility was addressed. This problem was solved using a two dimensional axisymmetric model in order to reduce the computational complexity, approximation that is valid if the section under analysis is far enough from the starting and ending points of the weld. Following the remark of Mullins and Gunnars [99], an isotropic elasto-plastic hardening model was used. Additionally, the hypothesis of small deformations was adopted. In the case of the thermal problem, two kinds of models were taken into account. On one model, the contribution of the latent heat associated to the solid/liquid phase change was considered, and in the other this contribution was neglected. When comparing the obtained residual stresses it is evident that the latent heat of the solid/liquid phase change does not contribute significantly, driving to think that it would be advisable not to compute solid/liquid phase change effects in order to gain in terms of computational efficiency and robustness. Future work will be addressed in order to model the Post Weld Heat Treatment.

Chapter 8

Final remarks

8.1 Contributions of this work

In this thesis different aspects of the multiphysics modelling of welding were considered. The main original contributions are listed below:

- Thermal Problem: a new enriched finite element for solving isothermal phase change problems was presented. It is important to remark that the proposed method avoids the use of an auxiliary transport equation to determine the enrichment position, which is common for level set formulations. This point establishes a new modern approach in enrichment techniques.

- Microstructure Evolution: the use of flowcharts was coined as fundamental abstraction for defining the microstructure kinetics, thus allowing to design a flexible computational tool for representing a wide range of materials. Additionally, a model for the Titanium alloy Ti6Al4V capable of predicting microstructure evolution of processes characterised by arbitrary thermal histories was proposed. Despite the fact that this model differs in minor details with existing models, it allowed to thoroughly analyse the microstructural behaviour of the alloy Ti6Al4V from the point of view of flowcharts. In the context of the alloy AISI 1060, the recalescence phenomenon was studied by considering the contribution of the latent heat of solid/solid transformations.

- Reduced Order Modelling: different strategies were analysed for building Hyper-Reduced Order Models to solve nonlinear thermal problems, with applications for welding modelling. A new proposal based on the separate hyper-reduction of the different terms contributing to the residual was introduced. Additionally, the use of moving frames to simulate welding problems was considered.

Mechanical concepts involved in the description of bodies during solidification were revisited and implemented as part of the FE code OOFELIE. In order to deal with the computational cost of welding simulations, the parallelisation of the FE framework was accomplished, thus taking advantage of distributed shared memory architectures.

A problem of technological importance was solved. Specifically, the simulation of the welding process of the nozzle weld that connects the Reactor Pressure Vessel and the cold leg of a typical nuclear power plant was performed. This application example allowed to predict the Weld Residual Stresses in the welded joint, a factor that highly influences the

formation of long cracks by means of the Stress Corrosion Cracking mechanism. Additionally, it was possible to establish that, for the present case, the contribution of solid/liquid latent heat to WRS can be neglected.

8.2 Future work

The following topics are subject of future research:

- Mechanical behaviour of bodies during solidification: in future work the use of an *updated Lagrangian* formulation by additionally considering an algorithm for correctly position newly activated elements will be studied. Also, the modelling and simulation of the Post Weld Heat Treatment will be treated.
- Reduced Order Modelling: future work will be carried out to further study the ill conditioning of the matrices involved in the HROM formulation and to find a method that can detect and avoid problems in the computations. Additionally, the hyper-reduction of welding problems with material deposition will be analysed.
- Microstructure evolution modelling: future work will be addressed to further study microstructural models with a formulation at the meso-scale, looking for more physically based models. Also, much of the effort will be directed towards obtaining computational amenable meso-scale formulations. If these objectives are met, it is believed that more general computational tools are going to be formulated, making possible the description of a wider range of materials.
- Solid/liquid phase change modelling: in future work, the extension of the proposed enriched finite element to two and three dimensional problems will be considered.

Appendix A

Publications

During the course of this thesis the following articles were published in international refereed journals:

- Cosimo, A., Cardona, A., and Idelsohn, S., 2014. "Improving the k-compressibility of hyper reduced order models with moving sources: Applications to welding and phase change problems". *Computer Methods in Applied Mechanics and Engineering,* **274**(0), pp. 237 – 263

- Cosimo, A., Fachinotti, V., and Cardona, A., 2013. "An enrichment scheme for solidification problems". *Computational Mechanics,* **52**(1), pp. 17–35

- Dalcin, L. D., Paz, R. R., Kler, P. A., and Cosimo, A., 2011. "Parallel distributed computing using Python". *Advances in Water Resources,* **34**(9), pp. 1124 – 1139

Additionally, the following works were presented and submitted to international and national conferences on Computational Mechanics:

- Cosimo, A., and Cardona, A. "Aspectos de implementación informática para el modelado computacional de la evolución microestructural de materiales". *Complete article accepted - XXI Congress on Numerical Methods and their Applications, 23th-26th September 2014, Bariloche, Patagonia, Argentina.*

- Cosimo, A., Cardona, A., Novara, P., and Calvo, N. "Weld residual stresses modelling. Application to a nuclear power plant welded joint.". *Complete article accepted - XXI Congress on Numerical Methods and their Applications, 23th-26th September 2014, Bariloche, Patagonia, Argentina.*

- Cavalieri, F. J., Cosimo, A., and Cardona, A. "Estudio de desgaste en válvulas de motores de combustión interna mediante el método de los elementos finitos y ensayos experimentales". *Complete article accepted - XXI Congress on Numerical Methods and their Applications, 23th-26th September 2014, Bariloche, Patagonia, Argentina.*

- Cosimo, A., and Cardona, A. "Modelación microestructural computacional de procesos con historias térmicas arbitrarias. Aplicación: aleación de Titanio Ti6Al4V". *Complete article accepted - International Congress of Metallurgy and Materials SAM-CONAMET / IBEROMAT 2014, 21th-24th October 2014, Santa Fe, Argentina.*

A. Publications

- Cosimo, A., Cardona, A., and Idelsohn, S., 2013. "Modelos de orden reducido para el problema térmico de soldadura". In Mecánica Computacional, C. García Garino, A. Mirasso, M. Storti, and M. Tornello, eds., Vol. XXXII, pp. 3151–3163 (complete article).

- Cosimo, A., and Cardona, A., 2012. "Problemas de choque térmico en el método de los elementos finitos". In Mecánica Computacional, A. Cardona, P. H. Kohan, R. Quinteros, and M. Storti, eds., Vol. XXXI, pp. 1937–1950 (complete article).

- Cosimo, A., Fachinotti, V., and Cardona, A., 2012. "Temperature gradient discontinuity aware numerical scheme for solidification problems (complete article)". In Proceedings of the 10th World Congress on Computational Mechanics.

- Rojas Fredini, E., Benitez, F., Cosimo, A., and Cardona, A., 2012. "Paralelización de un código de elementos finitos en multiprocesadores de memoria compartida". In Mecánica Computacional, A. Cardona, P. H. Kohan, R. Quinteros, and M. Storti, eds., Vol. XXXI, pp. 3153–3164 (complete article).

- Poster: Cosimo, A., Rojas Fredini, E., Benitez, F., and Cardona, A., 23-24 July 2012. Parallelization of a finite element code on shared memory multiprocessors. RISC High-Performance Computing Latin America Symposium (HPCLATAM). Buenos Aires, Argentina.

- Fachinotti, V., Cardona, A., Cosimo, A., Baufeld, B., and Van der Biest, O., 2010. "Evolution of temperature during shaped metal deposition: Finite element predictions vs. observations". In Mecánica Computacional, E. Dvorkin, M. Goldschmit, and M. Storti, eds., Vol. XXIX, pp. 4915–4926 (complete article).

Appendix B

Resumen extendido en Español

En esta tesis se considera el modelado Térmico-Mecánico-Microestructural de procesos de soldadura. La no-linealidad y el carácter multifísico del problema hacen necesario tomar como objeto de estudio distintas áreas de la Mecánica Computacional. Es por esto que cada uno de los problemas principales, en particular los problemas térmico, mecánico y microestructural, serán investigados de manera separada. Luego, en el contexto de la Mecánica Computacional de Soldadura, el acoplamiento de los mismos será descripto por medio de un esquema débilmente acoplado.

En el caso del problema térmico, la principal complicación surge del cambio de fase sólido/líquido. Las formulaciones clásicas que tratan este problema sufren de inestabilidades asociadas a la discontinuidad del gradiente de temperatura en la interfaz de cambio de fase. En este trabajo, esta cuestión se estudia por medio de una formulación de elementos finitos enriquecida con la habilidad de representar la discontinuidad del gradiente dentro de los elementos finitos. Se debe remarcar que en la solución propuesta no se hace uso de una ecuación auxiliar para determinar la posición del enriquecimiento, lo que es usual en formulaciones "level set".

El comportamiento mecánico de cuerpos que experimentan cambios de fase sólido/líquido se estudia e implementa en el software de multifísica OOFELIE. Cuando sea posible, se deben considerar los efectos de la evolución microestructural para predecir correctamente las tensiones residuales de soldadura. En este contexto, se encuentra que la implementación de un modelo microestructural en particular tiene asociada la restricción de que puede ser aplicado a un número reducido de materiales. Para lidiar con este problema, se estudia el diseño y la implementación de una herramienta computacional que posea flexibilidad suficiente como para poder describir la cinética de las transformaciones microestructurales de un amplio rango de materiales. Adicionalmente, se considera de manera particular la aleación de Titanio Ti6Al4V.

La alta complejidad computacional asociada a problemas de soldadura hace necesario considerar soluciones que ataquen este problema. Así, por un lado se formulan Modelos de Orden Hiper-Reducido (HROMs), y por el otro, se paraleliza el código de elementos finitos OOFELIE. Se demuestra que por medio de los HROMs propuestos se obtienen buenos factores de aceleración o speedups, inclusive cuando el factor de aceleración se calcula con el tiempo de ejecución requerido por el modelo sin reducir corriendo en paralelo.

El marco computacional desarrollado se aplica a la solución de un problema de soldadura de la vida real. Específicamente, se resuelve la soldadura de una unión soldada

de una central nuclear tipo, con el objeto de predecir las tensiones residuales resultantes. Adicionalmente, en este problema se estudia la contribución del calor latente proveniente del cambio de fase sólido/líquido al cálculo de tensiones residuales.

A continuación se detalla un resumen de cada uno de los problemas estudiados y los resultados obtenidos. Las principales contribuciones fueron realizadas en el marco de los problemas térmico y microestructural, y en la formulación de Modelos de Orden Hiper-Reducido para el problema térmico. En este contexto también se desarrolló un entorno de elementos finitos paralelo que permite resolver problemas térmicos-mecánicos-microestructurales de manera eficiente en máquinas de memoria compartida distribuida o NUMA (Non-Uniform Memory Access).

B.1 Problema Térmico

En el contexto de la Mecánica Computacional de Soldadura, el problema térmico se debe plantear considerando tanto los calores latentes provenientes de los cambios de fase sólido/líquido, como de los cambios de fase sólido/sólido. Sin embargo, las mayores complicaciones numéricas provienen del tratamiento del cambio de fase sólido/líquido, razón por la cual en esta sección se analizará sólo este caso, ignorándose las contribuciones de los demás calores latentes. Desde un punto de vista general, existen dos especializaciones del Método de los Elementos Finitos (FEM) para resolver problemas térmicos con cambio de fase sólido/líquido: métodos de malla móvil o métodos de seguimiento de la interfaz de cambio de fase, y métodos de malla fija [20]. Los métodos de malla móvil siguen la posición de la interfaz. De ahí que, por medio del remallado, la malla se hace conforme a esta posición y la formulación FEM puede tratar de manera estándar la discontinuidad débil presente en el campo de temperaturas en la interfaz [21, 23]. El problema con este tipo de métodos es que en problemas multidimensionales, la malla puede distorsionarse mucho a medida que avanza la interfaz. Esto no ocurre así en métodos de malla fija. Existen distintos métodos de este tipo, e.g., métodos de entalpía [24], métodos de capacidad [26] y métodos basados en temperatura [27, 28, 29]. A pesar de que ninguno de éstos puede capturar de manera exacta la discontinuidad del gradiente de temperatura en la interfaz, en la tesis de Fachinotti [31] se muestra que una variante [28] del método presentado en [29] obtiene buenos resultados en comparación a otros métodos allí estudiados.

Cuando se aplica el método presentado por Fachinotti *et al.* [28] a cambios de fase isotérmicos, la solución numérica presenta oscilaciones espurias debido a la incapacidad del método de poder representar la discontinuidad del gradiente de temperatura en la posición de la interfaz dentro de los elementos finitos. En respuesta a este problema, se propone introducir una formulación enriquecida de malla fija que tiene la capacidad de representar la discontinuidad del gradiente dentro de los elementos finitos. En este trabajo se presenta y analiza la idea restringiendo la formulación a problemas 1D, con lo cual se tiene como trabajo futuro extender esta formulación a problemas multidimensionales.

La idea se basa en describir el campo temperatura T^h dentro de los elementos finitos por medio de la expresión

$$T^h(x,t) = \sum_i N_i(x) T_i + E(x,t)a \tag{B.1}$$

donde el término $\sum N_i T_i$ corresponde a la discretización usual FEM con N_i denotando las funciones de forma y T_i denotando los grados de libertad. El término $E(x,t)a$ corresponde al enriquecimiento, donde $E(x,t)$ es la función de enriquecimiento y a es el grado de

libertad que tiene asociado. La función de enriquecimiento se define por medio de

$$E(x) = \begin{cases} \dfrac{x - x_1}{x_a - x_1} = \dfrac{\phi_1 - \phi}{\phi_1} & x \leq x_a \\ \dfrac{x_2 - x}{x_2 - x_a} = \dfrac{\phi_2 - \phi}{\phi_2} & x > x_a \end{cases} \tag{B.2}$$

donde ϕ es una función de "level set" o nivel, que localiza la posición de la interfaz de cambio de fase, y se define por

$$\phi = x - x_a \tag{B.3}$$

donde x_a está dado por

$$x_a = x_1 + s(x_2 - x_1) = x_1 + sh, \tag{B.4}$$

siendo x_1 la posición del nodo izquierdo, x_2 la posición del nodo derecho y h la longitud del elemento. El parámetro $s \in (0, 1)$ es utilizado para determinar localmente la posición de la interfaz sin hacer uso de una formulación auxiliar que transporte el "level set". Específicamente, para determinar la posición de la interfaz se hace uso de la restricción del problema de cambio de fase que dicta que la temperatura en la posición de la interfaz debe ser igual a la temperatura de fusión T_m. Así, si se utilizan funciones de forma FEM lineales, una fórmula cerrada caracteriza el parámetro s, y está dada por

$$s = \frac{T_m - T_1^{(i)} - a^{(i)}}{T_2^{(i)} - T_1^{(i)}}, \tag{B.5}$$

donde el súper-índice (i) denota la i-ésima iteración de un esquema Newton-Raphson. El no utilizar una formulación auxiliar de transporte para determinar la posición de enriquecimiento representa una de las mayores contribuciones de este trabajo. Notar que el basamento físico de este procedimiento es mayor que el de otros métodos, debido a que la posición de la interfaz se determina por medio de la distribución de temperatura actual.

Puede observarse un ejemplo de una función de enriquecimiento con una discontinuidad en $s = 0.35$ en la Figura B.1. Para apreciar el comportamiento numérico del enriquecimiento propuesto, se resuelve el problema de cambio de fase líquido/sólido en una barra en cuyos extremos se imponen condiciones de borde Dirichlet. Inicialmente la barra se encuentra a una temperatura por encima de la temperatura de fusión y se enfría por uno de sus bordes a una temperatura por debajo de la de fusión. Los resultados obtenidos en distintos pasos de tiempo se pueden observar en la Figura B.3, en donde además se muestran los resultados obtenidos con un esquema sin enriquecimiento [28] y la solución exacta. Adicionalmente, en la Figura B.2 se muestra la evolución temporal del campo temperatura en la posición 0.625m. Como puede apreciarse, el hecho de poder representar la discontinuidad del gradiente de temperatura tiene un gran impacto sobre la calidad de la solución obtenida. Se observa así el buen rendimiento del enriquecimiento propuesto.

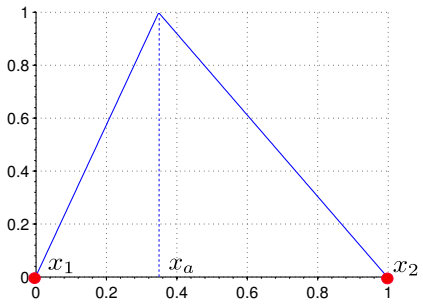


Figura B.1: Función de enriquecimiento.

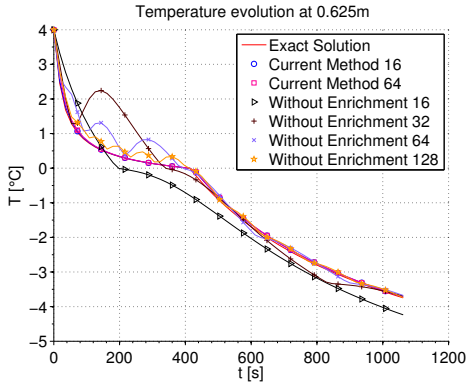


Figura B.2: Evolución de la temperatura. Resultados aproximados para distinto número de elementos igualmente espaciados y solución exacta.

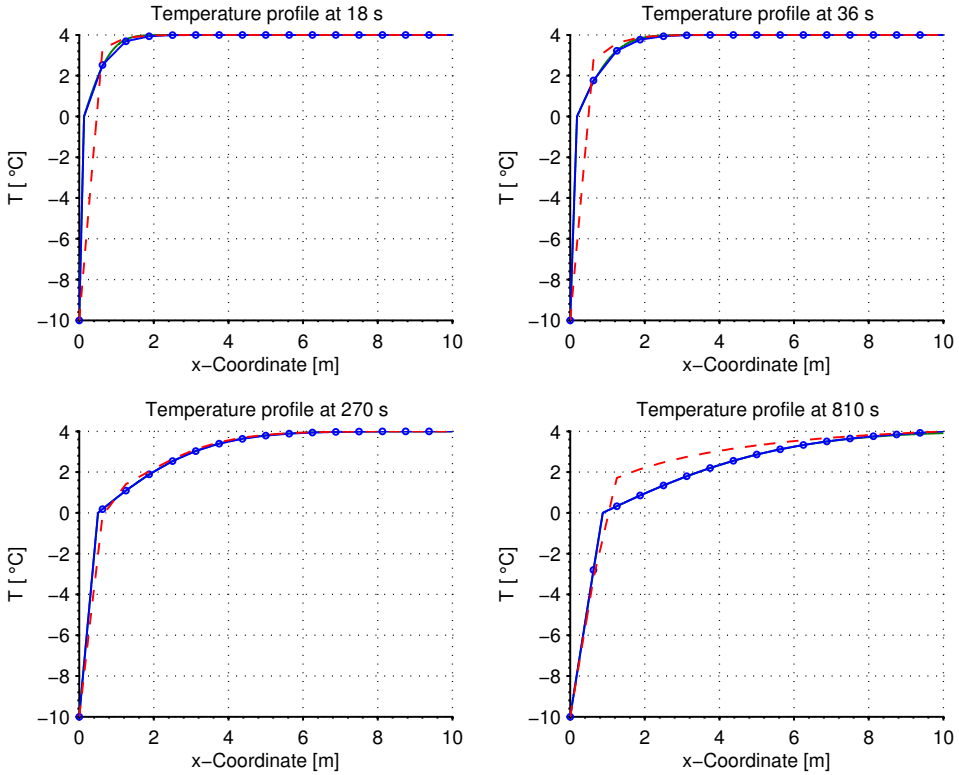


Figura B.3: Solución al problema de cambio de fase en una barra en distintos pasos de tiempo. Línea sólida: solución exacta; línea discontinua: solución aproximada sin enriquecimiento; línea sólida con círculos: solución aproximada obtenida con el método propuesto.

B.2 Evolución Microestructural

En procesos donde la energía térmica es la fuerza impulsora de cambios microestructurales y de la deformación del cuerpo, es esencial modelar las variaciones de las propiedades materiales en términos de la temperatura y de la historia térmica. Un buen ejemplo de

este tipo de procesos son los tratamientos térmicos. Sin embargo, son los procesos de soldadura los más complejos de describir desde el punto de vista microestructural, debido a que se caracterizan por historias térmicas arbitrarias de enfriamiento/calentamiento.

Generalmente, la implementación de un modelo microestructural en particular tiene asociada la restricción de que puede ser aplicado a un número reducido de materiales. Cuando un material difiere mucho de aquel modelo implementado, debe codificarse uno nuevo. Por ejemplo, modelos que describen el comportamiento microestructural de aleaciones de Titanio, como la aleación Ti6Al4V, son implementados de tal manera que sólo aquel material puede ser considerado. El objetivo de este trabajo es diseñar e implementar un marco computacional que sea capaz de representar el comportamiento microestructural de un amplio rango de materiales, sin requerir re-escribir código o, al menos, reducir esta tarea. En este contexto, se adoptan modelos macro-escala para modelar la evolución microestructural, debido a que son los más económicos desde el punto de vista de la complejidad computacional [3].

A fin de recabar los requerimientos del marco computacional a implementar, en esta tesis se analiza el comportamiento microestructural de la aleación de Titanio Ti6Al4V y del acero AISI 1060. Para el propósito del actual resumen, alcanza describir el comportamiento microestructural de la aleación Ti6Al4V para historias térmicas arbitrarias, omitiéndose así la descripción del acero AISI 1060.

Las transformaciones microestructurales pueden ser clasificadas en transformaciones difusivas y no difusivas. También se pueden encuentran transformaciones mixtas, como lo es la formación de bainita, sin embargo, hay trabajos [70] en donde esta transformación se modela como una transformación difusiva. En las siguientes secciones se describe el modelado de estas transformaciones.

B.2.1 Modelado de las transformaciones

En esta sección se describe brevemente el material a ser estudiado y los modelos que se utilizan en la descripción de los diferentes procesos involucrados en la evolución microestructural de materiales.

La aleación Ti6Al4V es una aleación de Titanio que se caracteriza por tener dos fases, α y β, la cual cuando se enfría desde la fase sólida β, comienza a formarse la fase ślida α debajo de la temperatura β-transus, T_β, considerada igual a $T_\beta = 1273\text{K}$ [75]. A medida que baja la temperatura, se conforman diferentes morfologías de α. Las morfologías que involucran transformaciones difusivas son: α borde de grano, α_g, y α Widmanstätten, α_w [67]. La formación de martensita, α', y α masiva, α_m, está asociada a transformaciones no difusivas características de un enfriamiento rápido: $\dot{T} < -410\text{Ks}^{-1}$ para α', y $-410\text{Ks}^{-1} \leq \dot{T} < -20\text{Ks}^{-1}$ para α_m.

En el presente trabajo, las morfologías de α asociadas a transformaciones difusivas que serán modeladas son α borde de grano y α Widmanstätten. En el caso de transformaciones no difusivas, se modelan martensita y α masiva como una sola estructura microestructural denotada por α'.

Transformaciones difusivas: modelo JMAK. El modelo de Johnson-Mehl--Avrami-Kolmogorov (JMAK) [12] es un modelo ampliamente utilizado para describir transformaciones difusivas isotérmicas. El mismo puede extenderse a transformaciones no-isotérmicas aplicando la regla de la aditividad por medio del método del tiempo ficticio [80]. Así, denotando por $\tau_s(T_n)$ al tiempo de incubación isotérmico, se tiene que el tiempo

de incubación no-isotérmico se completará cuando $\sum_{i=1}^{n} \frac{\Delta t_i}{\tau_s(T_i)} = 1$. En este caso Δt_i es el incremento temporal y T_i la temperatura. Luego, la fracción y_i de la fase α_i en el tiempo actual t_n está dada por

$$y_{i,n} = Y_{i,n}(1 - e^{-b_{i,n}(\Delta t_n + \tau_n^*)^{m_{i,n}}}), \tag{B.6}$$

donde el tiempo ficticio τ_n^* toma la siguiente expresión

$$\tau_n^* = \ln\left(\frac{Y_{i,n}}{Y_{i,n} - y_{i,n-1}}\right)^{\frac{1}{m_{i,n}}} b_{i,n}{}^{\frac{-1}{m_{i,n}}}, \tag{B.7}$$

y donde y_i son las fracciones de las fases α_i que presenta el material, Y_i es la máxima fracción transformada de la fase α_i considerada igual a la fracción de equilibrio de la fase en cuestión, y m_i y b_i son parámetros cinéticos que dependen de la temperatura. Estos parámetros cinéticos se estiman de diagramas TTT.

En este trabajo se modifica el modelo JMAK para incorporar el comportamiento inerte de ciertas fases durante el transcurso de la transformación difusiva modelada. Un ejemplo de esta transformación es la transformación $\beta \rightarrow \alpha_w + \alpha_g$, en donde la fase α' es inerte. En este caso la ecuación (B.6) se re-escribe como

$$y_{i,n} = X_{i,n}(1 - e^{-b_{i,n}(\Delta t_n + \tau_n^*)^{m_{i,n}}}), \tag{B.8}$$

donde se ha introducido el término $X_{i,n} = Y_{i,n} - y_{p,n-1}$ siendo p la fase inerte.

En la formulación anterior no se considera la posibilidad de que ocurran transformaciones simultáneas. Sin embargo, en el caso de la aleación de Ti6Al4V, la descomposición de la fase β en las morfologías α_w y α_g es simultánea. De acuerdo con [72], este fenómeno se puede describir evaluando primero $y_{\alpha_g,n}$ por medio de

$$y_{\alpha_g,n} = X_{\alpha,n}(1 - e^{-b_{\alpha_g,n}(\Delta t_n + \tau_n^*)^{m_{\alpha_g,n}}}) - y_{\alpha_w,n-1}, \tag{B.9}$$

con

$$\tau_n^* = \ln\left(\frac{X_{\alpha,n}}{X_{\alpha,n} - (y_{\alpha_g,n-1} + y_{\alpha_w,n-1})}\right)^{\frac{1}{m_{\alpha_g,n}}} b_{\alpha_g,n}{}^{\frac{-1}{m_{\alpha_g,n}}}, \tag{B.10}$$

para luego calcular $y_{\alpha_w,n}$ con expresiones similares.

A continuación se considera otro tipo de precipitación simultánea de fases. En el caso de la aleación de Titanio Ti6Al4V se observa experimentalmente que durante etapas de calentamiento, si hay martensita presente, α' descompone en $\beta + \alpha_w$ difusivamente [81]. Adoptando la hipótesis de que es posible despreciar la variación de la fase α_g durante este proceso, la reacción $\alpha' \rightarrow (\beta + \alpha_w)$ se describe por medio de

$$y_{\beta,n} = \frac{1 - Y_{\alpha,n}}{1 - y_{\alpha_g,n-1}}(y_{\alpha',n-1} - e^{f(\alpha')}) + y_{\beta,n-1}, \tag{B.11}$$

$$y_{\alpha_w,n} = \frac{Y_{\alpha,n} - y_{\alpha_g,n-1}}{1 - y_{\alpha_g,n-1}}(y_{\alpha',n-1} - e^{f(\alpha')}) + y_{\alpha_w,n-1}, \tag{B.12}$$

$$y_{\alpha_g,n} = y_{\alpha_g,n-1}, \tag{B.13}$$

$$y_{\alpha',n} = e^{f(\alpha')}. \tag{B.14}$$

Notar que Murgau *et. al* modelan $\alpha' \rightarrow (\beta + \alpha_w)$ introduciendo una fracción de equilibrio esperada de martensita, un dato adicional difícil de conseguir.

Transformaciones difusivas: modelo de Kelly. En el caso de la aleación de Titanio Ti6Al4V, Kelly [67] describió la disolución de $\alpha_w + \alpha_g$ en β como una transformación difusiva donde el crecimiento de β se supone gobernado por una ley parabólica. Así, el crecimiento de β se describe por

$$y_{\beta,n} = Y_{\beta,n} 2.20 \times 10^{-31} T_n^{9.88821} \left(\Delta t + \left(\frac{y_{\beta,n-1}}{Y_{\beta,n} f_{p,n}} \right)^2 \right)^{0.5}. \tag{B.15}$$

Transformaciones no difusivas. La formación de martensita se modela por medio de la extensión del modelo KM propuesta por Murgau *et. al* [72]:

$$y_{m,n} = \left(y_0^{M_s} - Y_{\alpha_p,n} \right) \left(1 - e^{r(T_n - M_s)} \right), \tag{B.16}$$

donde $y_0^{M_s}$ es la fracción de la fase matriz retenida en el tiempo correspondiente a M_s, el parámetro r es una constante específica del material modelado y α_p es la fase matriz.

Modelos de disolución instantánea y diagramas de equilibrio. En problemas de soldadura se encuentra frecuentemente la utilización de modelos de disolución instantánea para describir el comportamiento de la martensita durante la etapa de calentamiento. Esto es, cuando se alcanza la temperatura de disolución, T_{dm}, la martensita presente se disuelve instantáneamente contribuyendo a las otras fases.

B.2.2 Descripción computacional de la cinética de las transformaciones microestructurales

En ciencias de la computación está muy extendido el uso de diagramas de flujo para especificar un algoritmo o conjunto de pasos para resolver un problema. Es por esto que, por ejemplo, el algoritmo que se sigue para calcular la evolución microestructural de un material puede ser especificado en términos de un diagrama de flujo. En este trabajo, se propone extender el uso de diagramas de flujo para definir el comportamiento material a nivel microestructural. Esto es, se propone definir el comportamiento material por medio de un conjunto de propiedades materiales: el tipo de respuesta térmica (isótropa, ortotrópica, etc.), el tipo de respuesta mecánica (elástica, elasto-plástica, etc.) y un diagrama de flujo para especificar la cinética de las transformaciones microestructurales. De esta manera, cada componente del diagrama de flujo se codifica sólo una vez, eventualmente reutilizado cuando se define un nuevo material.

Para la confección de diagramas de flujo se utilizan dos tipos de cajas o representaciones gráficas: decisiones, denotadas por medio de diamantes, y procedimientos, denotados por medio de cajas rectangulares. Generalmente, cada procedimiento estará describiendo una transformación, o parte de una transformación. Con el fin de relacionar cada uno de los componentes del diagrama de flujo, cada componente es identificado por medio de un identificador único (ID) denotado en las figuras por TID. Los lazos o links entre los diferentes componentes se especifican por medio de Link IDs y son denotados por LIDs. La implementación informática de los modelos se basa en interpretar automáticamente el diagrama de flujo que se especifica como propiedad adicional de los materiales involucrados en el problema. En esta sección, el material Ti6Al4V será analizado desde el punto de vista de diagramas de flujo.

Lógica para describir historias térmicas arbitrarias. Generalmente, los modelos microestructurales son diseñados para predecir la disolución o formación de una fase matriz característica, β en el caso de Ti6Al4V. Los modelos de disolución son estrictamente aplicados a procesos de enfriamiento, y los modelos de formación son aplicados a procesos de calentamiento. En el caso de historias térmicas arbitrarias la decisión de qué modelo aplicar se toma en base al diagrama de equilibrio. Esto se puede observar en la Figura B.4, en donde además se contempla la posibilidad de que ocurran transformaciones sólido/líquido.

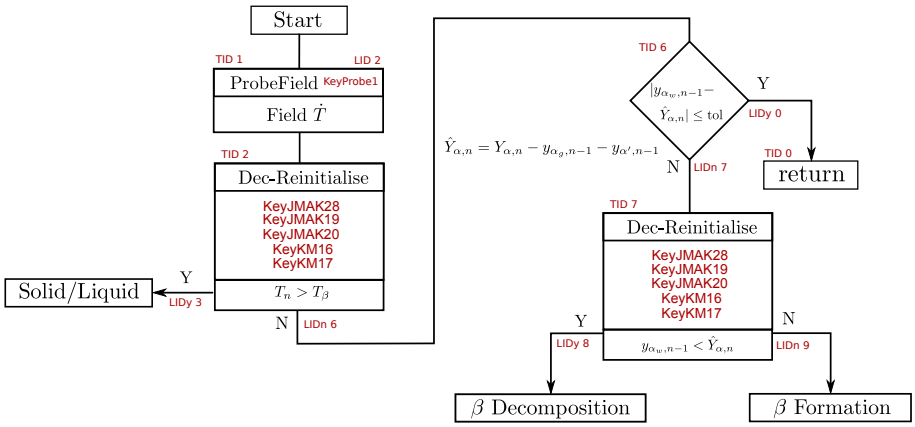


Figura B.4: Lógica propuesta para describir historias térmicas arbitrarias (Ti6Al4V).

En el caso de problemas con historias térmicas arbitrarias que se caracterizan por inducir múltiples ciclos de descomposición/formación de la fase matriz, la transición de un modelo de descomposición a uno de formación, y vice versa, introduce el requerimiento de reinicializar ciertas variables de estado. Ejemplos de estas variables de estado son los tiempos de incubación calculados por los modelos JMAK, y la fracción de la fase matriz retenida en el momento en que comienza la transformación martensítica modelada por un modelo KM.

En el caso de la aleación de Titanio Ti6Al4V, la reinicialización de estas variables de estado se controla por medio de los procedimientos Dec-Reinitialise TIDs 2 y 7 de la Figura B.4. En este tipo de procedimientos, las variables de estado a ser reinicializadas se listan en el primer espacio de las cajas que representan estos procedimientos. En el segundo espacio se especifica un condicional que controlará el proceso de reinicialización, es por esto que los procedimientos Dec-Reinitialise pueden retornar por las ramas **Y** y **N**. Las variables especificadas son reinicializadas cada vez que el resultado del condicional en el tiempo actual de la simulación difiere del resultado del condicional en el paso previo. Notar que toda la algoritmia para resolver la evolución microestructural comienza en el procedimiento **Start** de la Figura B.4.

Aleación de Titanio Ti6Al4V. A continuación se proponen modelos para la descomposición y formación de la fase β del material, adicionalmente considerando la fase líquida.

En la Figura B.5 se muestra la lógica para tratar el cambio de fase sólido/líquido. Para temperaturas mayores que T_β, sólo pueden existir β y líquido. Es por esto que, primero, el procedimiento InstantaneousDissolution, TID 3, es utilizado para hacer cero todas las fracciones de las fases excepto de la fase líquida. Luego, la fracción de la fase líquida, f_l,

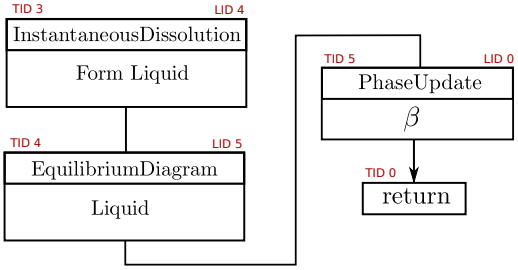


Figura B.5: Diagrama de flujo para el manejo de cambios de fase sólido/líquido.

se calcula de acuerdo con el diagrama de equilibrio, TID 4. Seguido, la cantidad de la fase β se calcula en el procedimiento PhaseUpdate, TID 5, de acuerdo con $y_\beta = 1 - f_l$.

Descomposición de β. El diagrama de flujo que caracteriza la descomposición de β se muestra en la Figura B.6. Durante las transformaciones de estado sólido no está presente la fase líquida, es por esto que por medio de un procedimiento InstantaneousDissolution, TID 8, la fase líquida se hace cero.

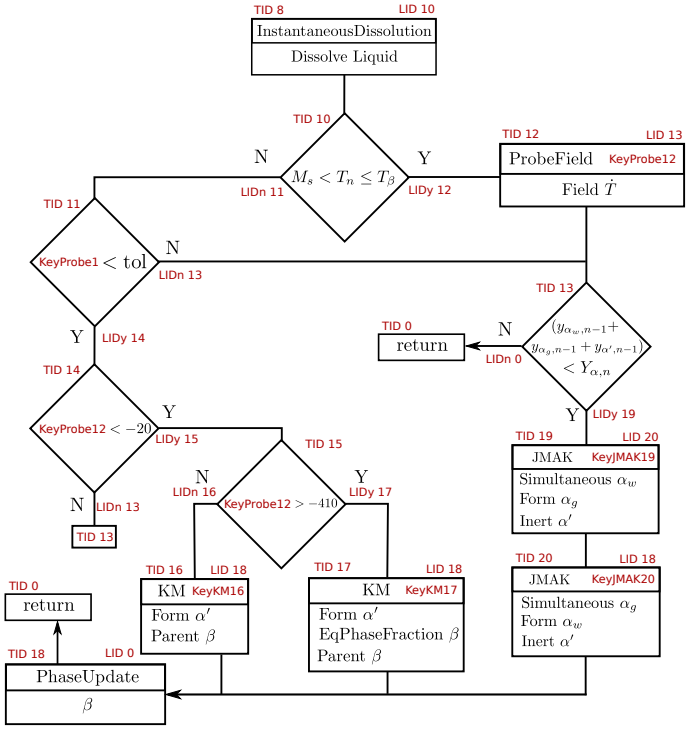


Figura B.6: Diagrama de flujo para la descripción del material Ti6Al4V para la descomposición de β.

La formación de las fases α_g y α_w se describe de acuerdo a la cinética de transformaciones difusivas utilizando modelos JMAK, donde puede tomar lugar la formación simultánea de fases. Se considera que las fases α_w y α_g comienzan a formarse en el rango de temperatura $M_s < T \leq T_\beta$ sólo si $y_{\alpha_w,n-1} + y_{\alpha_g,n-1} + y_{\alpha',n-1} < Y_{\alpha,n}$ (decisiones TID 10 y 13). Se adoptaron T_β y M_s igual a 1273K y 848K, respectivamente.

Durante la etapa de enfriamiento, la formación de martensita se modela de acuerdo con el modelo KM, tomando como exponente $r = 0.005$. Esta transformación se considera que

empieza en la temperatura crítica M_s. Se han observado transformaciones martensíticas totales para velocidades de temperatura mayores a 410Ks^{-1} [87], pero para velocidades de temperatura en el rango $-410\text{Ks}^{-1} < \dot{T} \leq -20\text{Ks}^{-1}$ debe ser considerada la posibilidad de que ocurran transformaciones difusivas. Para poder describir el primer caso se utiliza el modelo KM dado por la ecuación (B.16) con $Y_{\beta,n} = 0$ (procedimiento KM, TID 16). Para el segundo caso se utiliza el modelo KM con $Y_{\beta,n} \neq 0$ (procedimiento KM, TID 17). La velocidad de enfriamiento se calcula en el rango de temperatura $M_s < T \leq T_\beta$ [67], y se implementa en el diagrama de flujo por medio del procedimiento ProbeField, TID 12. Este procedimiento calcula y guarda el campo especificado, en este caso $\dot{T}$, en una variable identificada por un `Key`, en este caso KeyProbe12. KeyProbe12 es utilizado por las decisiones TID 14 y TID 15, para decidir si se formará martensita. Finalmente, la cantidad retenida de β se calcula por medio del procedimiento PhaseUpdate, TID 18.

La decisión identificada por TID 11 verifica que la velocidad de enfriamiento se mantenga por debajo de un umbral denotado por `tol` (en este caso tomado igual a -1Ks^{-1}). Si la velocidad de enfriamiento no se mantiene por debajo de este umbral se contempla la posibilidad de que transformaciones difusivas tengan lugar. La velocidad involucrada en la decisión TID 11, se mide en el ProbeField TID 1 y se guarda en la llave KeyProbe1 (ver Figura B.4).

Formación de β. En la Figura B.7 puede observarse el diagrama de flujo para la formación de β. Si α' está presente (decisión TID 9) y si $T_{\text{d0}} < T_n < T_{\text{dm}}$ (decisiones TIDs 22 y 27), se observa que la martensita comienza a disolverse en $\alpha_w + \beta$ por medio de un mecanismo difusivo. Para modelar la disolución $\alpha' \rightarrow \alpha_w + \beta$ se utiliza el modelo JMAK dado por las ecuaciones (B.11-B.14) (procedimiento JMAK TID 28).

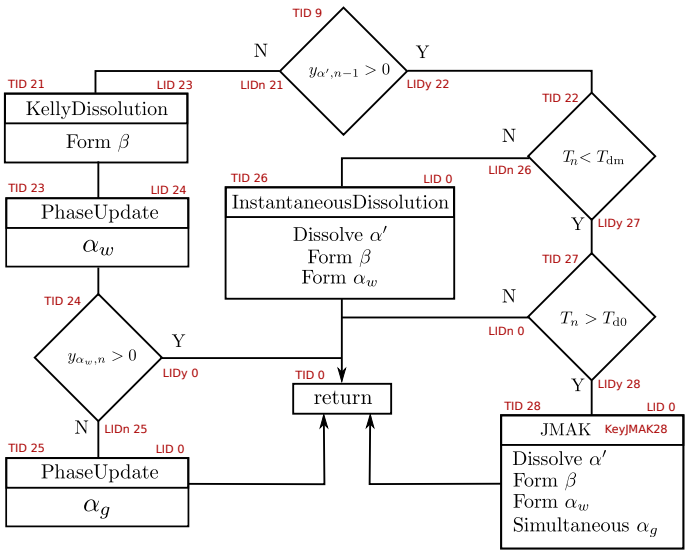


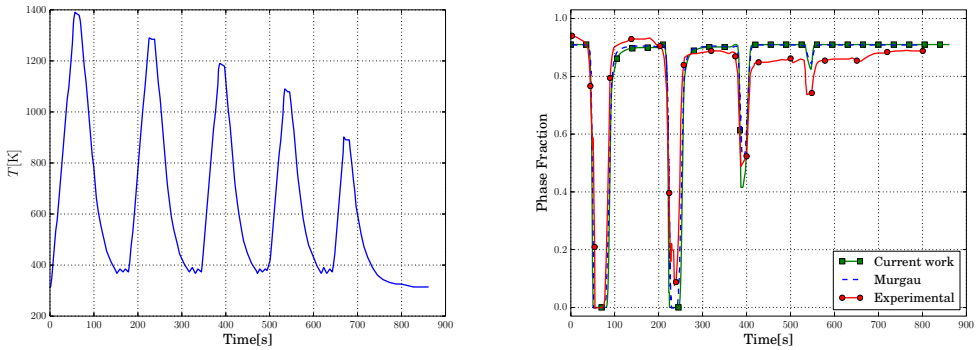
Figura B.7: Diagrama de flujo para la descripción de la formación de la fase β de la aleación de Titanio Ti6Al4V.

La temperatura crítica T_{dm} se toma como temperatura de disolución para la martensita, lo cual significa que luego de sobrepasar esta temperatura, α' se disuelve completamente, i.e. $y_{\alpha'} = 0$, contribuyendo proporcionalmente a las fases α_w y β. En el diagrama de flujo esto se refleja por medio del procedimiento InstantaneousDissolution, TID 26.

Si la fracción de la fase α' es cero, α_w y α_g forman β. El modelo de Kelly, dado por la ecuación (B.15), se utiliza para describir la reacción $\alpha_w + \alpha_g \rightarrow \beta$ (procedimiento KellyDissolution, TID 21). Se supone que la fase α_w se disuelve primero (procedimiento PhaseUpdate, TID 23), y una vez que su valor es cero (decisión TID 24), α_g comienza a disolver (procedimiento PhaseUpdate, TID 25).

B.2.3 Ejemplo de aplicación y validación

Para validar el modelo presentado para la aleación de Titanio Ti6Al4V se reproduce numéricamente el experimento realizado por Babu *et al.* [71]. La historia térmica que caracteriza a este experimento se detalla en la Figura B.8a. En la Figura B.8b se puede observar la comparación entre los resultados obtenidos con el modelo propuesto en este trabajo, los obtenidos con el modelo de Charles Murgau *et al.* [72] y los resultados experimentales. Como se puede apreciar, los resultados muestran una buena concordancia con los obtenidos por [72] y con las mediciones experimentales.



(a) Historia térmica. (b) Validación de los resultados obtenidos.

Figura B.8: Ejemplo de aplicación modelo Ti6Al4V.

B.3 Problema Mecánico

Los problemas de soldadura son resueltos haciendo uso de un esquema termo-mecánico desacoplado, donde, en cada paso de tiempo, el problema térmico se resuelve primero, y luego lo es el problema mecánico tomando como estado el campo de temperatura calculado en el paso anterior. Para modelar la respuesta mecánica del material se adopta un modelo con endurecimiento isotrópico en donde el endurecimiento se supone gobernado por la ley de Voce [100] con parámetros dependientes de la temperatura. La adopción de un modelo con endurecimiento isotrópico como ley constitutiva se apoya en el trabajo de Mullins y Gunnars [99] que concluye que las mediciones experimentales de tensiones residuales en soldaduras de tubos son mejor aproximadas por medio de este tipo de modelos. Para resolver el balance mecánico desde el punto de vista numérico se adopta una discretización FEM mixta Q1/P0 [97]. Para integrar las ecuaciones de evolución asociadas al modelo elasto-plástico con endurecimiento isotrópico se hace uso de un esquema de "return mapping" clásico [96].

Se adopta el método de los elementos inactivos para modelar la deposición de material en los problemas de soldadura. En este método, cada elemento tiene asociado un tiempo de activación que permite distinguir entre elementos activos, que deben ser considerados parte de los cálculos, y elementos inactivos, que son aquellos elementos que forman parte de cordones que no han sido depositados todavía y que no deben ser considerados como parte del cálculo. Una vez que es activado un elemento inactivo, porque su tiempo de activación así lo indica, se incorpora al problema térmico pero no al problema mecánico. En el caso del problema mecánico, para la activación de elementos se consideran también las temperaturas nodales del elemento en cuestión. Teniendo en cuenta que un entorno de una partícula material es capaz de desarrollar resistencia siempre que su temperatura T esté por debajo de la temperatura de resistencia cero (ZST) T_{zst}, se decide que un elemento mecánico se activará sólo cuando para todos sus nodos se cumpla que $T < T_{\text{zst}}$, y, adicionalmente, se verifique una de las siguientes condiciones

- En el caso bidimensional, el elemento a ser activado comparte al menos un segmento con otro elemento cuyo estado es activo.
- En el caso tridimensional, el elemento a ser activado comparte al menos una cara con otro elemento cuyo estado es activo.

En el caso de problemas termo-mecánicos donde tienen lugar cambios de fase sólido/-líquido, necesita considerarse el uso de configuraciones naturales o libres de tensiones para calcular correctamente las tensiones térmicas. Cuando es activado un elemento mecánico en particular, la tensión, la deformación y en general toda variable interna, se deben reinicializar a cero. Esto se hace por medio de adoptar una configuración natural para el entorno de todo punto material en el instante en que dicho entorno comienza a desarrollar resistencia.

Para validar la implementación del esquema numérico presentado, se resuelve el problema de Weiner-Boley [105] que consiste en la solidificación de una barra cuya respuesta mecánica se supone elasto-plástica con una tensión de fluencia dependiente de la temperatura. Los resultados obtenidos se pueden observar en la Figura B.9, de donde se puede apreciar la buena concordancia con la solución exacta.

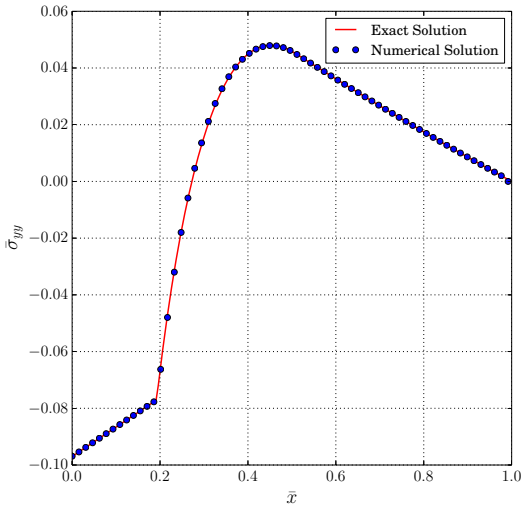


Figura B.9: Resultados obtenidos para el problema de Weiner-Boley. En este caso $\bar{\sigma}_{yy}$ y $\bar{x}$ son expresiones adimensionales para la tensión σ_{yy} y la coordenada x, respectivamente.

B.4 Implementación Paralela

Los problemas de multifísica, como los problemas de soldadura, suelen ser computacionalmente muy costosos. Generalmente se toman hipótesis simplificativas para reducir la complejidad de estos problemas y poder hacerlos más tratables. Sin embargo, muchas veces esto no alcanza, por lo cual en esta sección se considera el estudio de técnicas de Computación de Alto Rendimiento (HPC) para paliar el problema del costo computacional. Específicamente, se paraleliza el código de elementos finitos OOFELIE [39] para poder aprovechar eficientemente los recursos computacionales ofrecidos por entornos paralelos de memoria compartida distribuida o bien conocidos como entornos NUMA.

En aplicaciones FEM se puede sacar provecho del alto grado de paralelismo que caracteriza el ensamblado de las matrices y de los vectores estructurales involucrados en la discretización. Consecuentemente, se tiene como objetivo básico paralelizar el ensamblado de estas estructuras. En este contexto y siguiendo el trabajo de Farhat y Crivelli [38], si se quieren ensamblar en paralelo las contribuciones elementales directamente sobre las estructuras globales correspondientes (una matriz o un vector) los elementos que se procesan en paralelo no deben compartir grados de libertad para de esta manera evitar condiciones de carrera. Adicionalmente, previo a procesar los elementos, debe calcularse el perfil de la matriz para así pre-alocar la memoria necesaria. La estrategia computacional empleada para resolver este problema se resume en las etapas que se muestran en la Figura B.10.

Para el cálculo del perfil de la matriz, se lleva a cabo un “ensamblaje en seco”, en donde no se calculan las contribuciones elementales pero se determinan las entradas no nulas de la matriz en cuestión. Para evitar las secciones críticas, se calcula el grafo dual de la malla involucrada para cada tipo de elemento y luego se colorea. De esta manera cada color puede ser procesado concurrentemente de manera “thread-safe”.

En las Figuras B.11a y B.11b se pueden observar los factores de aceleración y las eficiencias que se obtienen con la solución implementada en cuatro máquinas distintas. La prueba consistió en medir los tiempos del ensamble de la matriz tangente de un problema 3D de cambio de fase no-isotérmico, donde se utilizaron 192000 tetraedros y 44541 grados

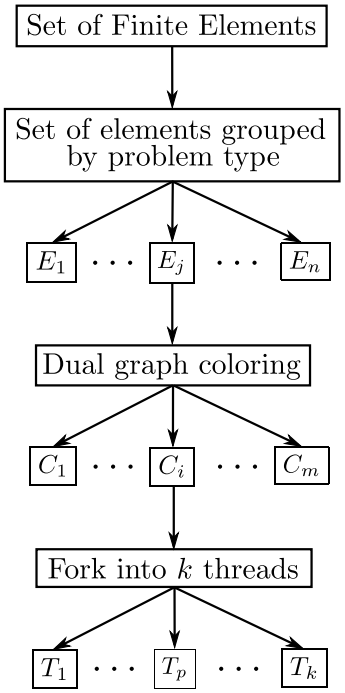


Figura B.10: Representación esquemática de la solución.

de libertad. El rendimiento obtenido con las máquinas `Intel i7` e `Intel Xeon E5620` está dentro de los resultados esperados. Esto no es así con las máquinas `Intel Xeon E5430` e `Intel Xeon X5680`. En el primer caso, los malos resultados se pueden explicar debido a que en dicha máquina el acceso a memoria es simétrico, implementado por medio de la tecnología Front Side Bus. En el segundo caso, los malos resultados son atribuidos a un mal funcionamiento de la máquina, ya que para la máquina `Intel Xeon E5620`, de similar microarquitectura, se obtuvieron buenos resutlados. De estas experiencias se puede concluir que con la solución propuesta se obtiene una buena performance en el caso de máquinas NUMA.

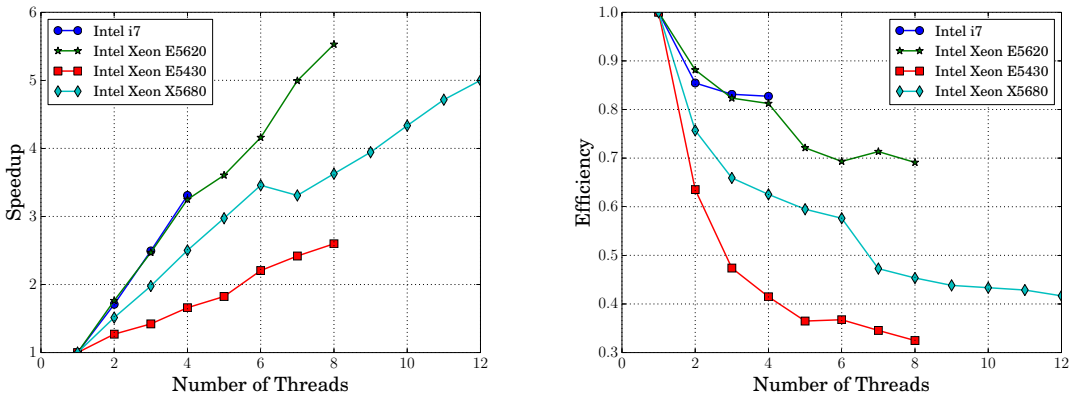


Figura B.11: Factor de aceleración y eficiencia para el ensamblado de la matriz tangente de un problema 3D de cambio de fase no-isotérmico (se utilizan elementos tetraédricos).

B.5 Formulación de Modelos de Orden Reducido

La simulación de un proceso de soldadura se caracteriza por su alta complejidad, ya que debe ser descripto como un proceso Térmico-Metalúrgico-Mecánico (TMM) [1]. A pesar del esfuerzo que se realiza para reducir el costo computacional asociado, el problema sigue siendo inabordable desde la perspectiva de aplicaciones de tiempo real.

Es en este contexto donde los Modelos de Orden Reducido (ROM) se presentan como una solución elegante y alentadora. Es por esto, que en este trabajo se investiga el diseño de ROMs para el problema de transferencia de calor no lineal, donde debe ser considerado el cambio de fase sólido/líquido [85].

Últimamente el uso de representaciones separadas para construir ROMs ha atraído la atención de la comunidad ingenieril. Básicamente hay dos maneras de construir representaciones separadas [32]. Por un lado, se tienen las técnicas de reducción *a posteriori*, donde se requiere el conocimiento de una solución a un problema de entrenamiento. La técnica *a posteriori* más prominente se basa en el método de Descomposición Ortogonal Propia (POD) [33, 34]. Por otro lado, se tienen las técnicas de reducción *a priori*, donde no se requiere conocimiento alguno de una solución a un problema de entrenamiento. En este contexto, el método con mayor difusión es el basado en la Descomposición Propia Generalizada (PGD) propuesto por Ladevèze [35, 36]. En este trabajo se adopta el esquema *a posteriori* basado en POD.

En ambos enfoques, para tratar de forma exitosa con la alta dimensionalidad del problema, se supone que es posible describir la solución en términos de un número reducido de funciones en la representación separada. En este sentido se puede hablar de la *hipótesis de la representación separada*. Una consecuencia de esta idea es hacer la suposición de que la solución del problema es *k-compresible* en *alguna* base, donde k es muy pequeño. Cuando se dice que la solución es *k-compresible*, se quiere significar que la solución tiene a lo sumo k entradas no nulas cuando se expresa en términos de aquella base *mágica* [120, 121].

En el caso de problemas no-lineales, donde el residuo y la matriz tangente deben ensamblarse en cada iteración del método empleado, se aplica una segunda reducción para eliminar operaciones cuyo orden está dado por la cantidad de elementos de la discretización. Por medio de aplicar esta segunda reducción se llega a la formulación de Modelos de Orden Hiper-Reducido.

B.5.1 Formulación de Modelos de Orden Hiper-Reducido para el problema térmico

Para describir el campo temperatura $T(\boldsymbol{x}, t)$, se utiliza la representación separada espacio-temporal de orden M dada por

$$T(\boldsymbol{x}, t) \simeq \sum_{j=1}^{M} S_j(\boldsymbol{x}) R_j(t) + T_b(\boldsymbol{x}, t), \tag{B.17}$$

donde T_b denota la función que describe las condiciones de borde esenciales no homogéneas, considerada nula por simplicidad en lo que sigue.

Un enfoque para encontrar una base (funciones $S_j(\boldsymbol{x})$ en la ecuación (B.17)) que sea capaz de capturar la dinámica del problema en un número pequeño de componentes, se basa en analizar la respuesta del sistema que viene dada por la solución a un problema de entrenamiento. Luego, para construir esta base se procede como en un Análisis de

Componentes Principales (PCA) [134]. Esto es, primero se construye un conjunto de *snapshots* o instancias temporales de la distribución espacial de la solución [135], y luego se lleva a cabo el PCA haciendo uso de la descomposición ortogonal propia, o mejor conocida en el contexto de Álgebra Lineal como Descomposición en Valores Singulares (SVD) [136].

Si los snapshots se representan por la matriz $\boldsymbol{A}$, de la SVD se tiene

$$\boldsymbol{A} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}^T, \tag{B.18}$$

donde las columnas $\boldsymbol{u}$ de $\boldsymbol{U}$ son los vectores singulares izquierdos de $\boldsymbol{A}$, las columnas $\boldsymbol{v}$ de $\boldsymbol{V}$ son los vectores singulares derechos de $\boldsymbol{A}$ y las entradas diagonales σ de $\boldsymbol{\Sigma}$ son los valores singulares de $\boldsymbol{A}$.

Para el caso en estudio, los vectores singulares izquierdos conforman la base buscada. Además, asumiendo que estos vectores capturan la dinámica del problema en los primeros k vectores, decimos que la respuesta es *k-compresible* y la representación separada dada por la ecuación (B.17) puede escribirse como

$$T(\boldsymbol{x}_i, t) \simeq \Big[\sum_{j=1}^{k} \boldsymbol{u}_j R_j(t)\Big]_i. \tag{B.19}$$

Quedan determinar los coeficientes $R_j(t)$. Cabe mencionar que los vectores $\boldsymbol{u}_j$ también se denominan modos POD de la aproximación, que una vez determinados, se los utiliza para proyectar, haciendo uso de una proyección Bubnov-Galerkin, la ecuación que caracteriza a la discretización del problema no lineal.

Expresando el campo de temperatura en términos de los modos POD se tendrá

$$T(\boldsymbol{x}, t_n) = \boldsymbol{N}^T\boldsymbol{X}\boldsymbol{a}_n = \boldsymbol{N}^T\boldsymbol{T}_n, \tag{B.20}$$

donde $\boldsymbol{X}$ denota la matriz de cambio de base cuyas columnas son los primeros k modos POD y $\boldsymbol{a}_n$ denota el vector de coeficientes $R_j(t)$ evaluado en el instante t_n. Notar, que en este caso en lugar de disponer de N grados de libertad (DOFs), como lo es en el caso del modelo de alta fidelidad, ahora se dispone de k DOFs con $k \ll N$.

El residuo del problema no lineal de cambio de fase se puede escribir como

$$\boldsymbol{\Pi} = \frac{\boldsymbol{G}_n^c - \boldsymbol{G}_{n-1}^c}{\Delta t} + \boldsymbol{G}_k + \frac{\boldsymbol{G}_n^l - \boldsymbol{G}_{n-1}^l}{\Delta t} + \boldsymbol{F} - \boldsymbol{Q} = 0, \tag{B.21}$$

donde $\boldsymbol{G}_n^c$ es el término asociado a la capacidad calorífica, $\boldsymbol{G}_n^l$ es el término asociado al calor latente, $\boldsymbol{G}_k$ es el término proveniente de la conductividad térmica, $\boldsymbol{F}$ nuclea términos frontera y $\boldsymbol{Q}$ está asociado al término fuente. Debido a que los términos no lineales necesitan ser ensamblados en cada iteración del esquema de Newton-Raphson, poco se puede ganar en términos de cantidad de operaciones [125, 126]. Para evitar esta dificultad, estos términos no lineales son calculados haciendo uso de la reconstrucción de datos gappy [130], que lleva a obtener un modelo hiper reducido. Específicamente, en esta tesis se propone hiper-reducir cada término no lineal de manera separada, distinguiéndose de otros esquemas, como el propuesto en [125], donde el residuo del problema se hiper-reduce como una unidad. En lo que sigue el proceso de hiper-reducir se expone para un término arbitrario $\boldsymbol{G}_p$.

Primero se calculan un conjunto de snapshots correspondientes a $\boldsymbol{G}_p$. Este conjunto se conforma de la contribución $\boldsymbol{G}_p$ al residuo convergido en cada iteración del esquema de

Newton-Raphson. Luego, se aplica la SVD a estos snapshots para obtener un conjunto de "modos gappy" denotados por $\boldsymbol{\Psi}$. Asumiendo significativos sólo los primeros n_g modos, una aproximación a $\boldsymbol{G}_p$ puede ser escrita como

$$\boldsymbol{G}_p \simeq \boldsymbol{\Psi}\boldsymbol{b}, \tag{B.22}$$

donde las columnas de $\boldsymbol{\Psi}$ son los primeros n_g modos gappy y $\boldsymbol{b}$ son parámetros a determinar. Supóngase que sólo se conocen n_s componentes de $\boldsymbol{G}_p$, denotadas por $\widehat{\boldsymbol{G}_p}$, y que se representan multiplicando la ecuación (B.22) por la matriz de muestreo $\boldsymbol{R}$ de dimensión $n_s \times N$, esto es

$$\widehat{\boldsymbol{G}_p} = \boldsymbol{R}\boldsymbol{G}_p \simeq \boldsymbol{R}\boldsymbol{\Psi}\boldsymbol{b} = \widehat{\boldsymbol{\Psi}}\boldsymbol{b}, \tag{B.23}$$

en donde el operador $\widehat{\cdot}$ denota el muestreo de la cantidad $\cdot$. Luego, haciendo uso de la pseudo-inversa se hallan los coeficientes $\boldsymbol{b}$ y se tiene la siguiente aproximación a $\boldsymbol{G}_p$

$$\boldsymbol{G}_p \simeq \boldsymbol{\Psi}(\widehat{\boldsymbol{\Psi}}^T\widehat{\boldsymbol{\Psi}})^{-1}\widehat{\boldsymbol{\Psi}}^T\widehat{\boldsymbol{G}_p}. \tag{B.24}$$

Esto es, con sólo conocer n_s componentes del vector $\boldsymbol{G}_p$ se puede reconstruir una aproximación del mismo. Existen muchas técnicas dedicadas a encontrar los n_s puntos donde evaluar la función $\boldsymbol{G}_p$. En este trabajo se hace uso de una extensión del Método de la Interpolación Discreta Empírica (Discrete Empirical Interpolation Method -DEIM-) [127].

Luego, para obtener el término hiper-reducido $\boldsymbol{G}_p^p$ de $\boldsymbol{G}_p$ se proyecta la aproximación dada por la ecuación (B.24) con la base POD, o modos de proyección, $\boldsymbol{X}$, resultando en

$$\boldsymbol{G}_p^p = \boldsymbol{X}\boldsymbol{\Psi}(\widehat{\boldsymbol{\Psi}}^T\widehat{\boldsymbol{\Psi}})^{-1}\widehat{\boldsymbol{\Psi}}^T\widehat{\boldsymbol{G}_p} = \boldsymbol{A}_p\widehat{\boldsymbol{G}_p}. \tag{B.25}$$

Para obtener la formulación hiper-reducida del problema cuyo residuo se especifica en la ecuación (B.21), se procede de manera similar con cada uno de los términos, obteniéndose el siguiente residuo hiper-reducido

$$\boldsymbol{\Pi}^p = \boldsymbol{A}_c\frac{\widehat{\boldsymbol{G}}_n^c - \widehat{\boldsymbol{G}}_{n-1}^c}{\Delta t} + \boldsymbol{A}_k\widehat{\boldsymbol{G}}_k + \boldsymbol{A}_l\frac{\widehat{\boldsymbol{G}}_n^l - \widehat{\boldsymbol{G}}_{n-1}^l}{\Delta t} + \boldsymbol{A}_f\widehat{\boldsymbol{F}} - \boldsymbol{A}_q\widehat{\boldsymbol{Q}}. \tag{B.26}$$

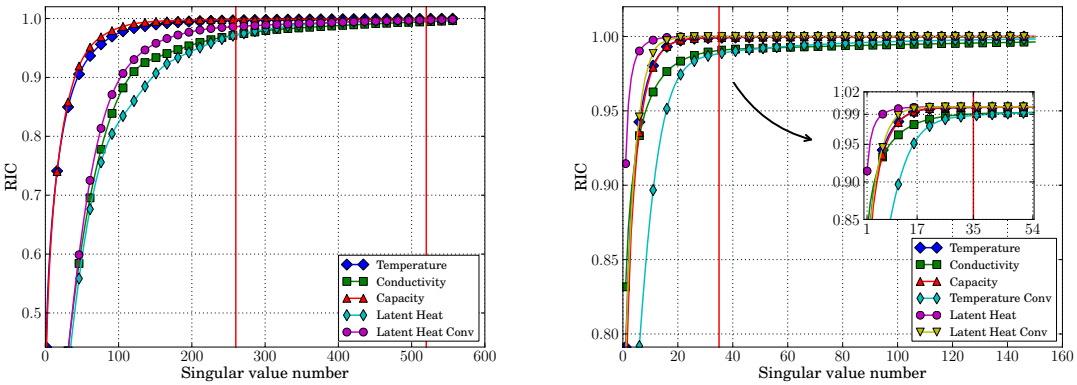
Esta formulación es válida cuando se utiliza un marco de referencia fijo con respecto al cuerpo que se está analizando. En esta tesis también se estudió la formulación de modelos de orden hiper-reducido (HROMs) considerando marcos de referencia móvil para describir fuentes de energía concentrada que varían su posición con el tiempo. La formulación de este tipo de HROMs es similar a la del caso anteriormente expuesto. A continuación se presenta un ejemplo de aplicación.

B.5.2 Ejemplo de aplicación: soldadura de un tubo

Este ejemplo consiste en la soldadura de un tubo de 0.075m de radio externo y de 0.05615m de radio interno. El modelo sin reducir, o modelo de alta fidelidad HF, consiste en 388800 elementos tetraédricos y 74100 grados de libertad. Se utiliza un incremento temporal de 0.15s para el intervalo de tiempo $[0, 100]$ s. Se comparan los resultados obtenidos considerando marcos de referencia móvil y fijo.

A continuación se hace uso del Contenido de Información Relativo, RIC, que se calcula como $\text{RIC} = \frac{\sum_{i=1}^{N_s}\lambda_i}{\sum_{i=1}^{N_r}\lambda_i}$, donde λ_i son los valores singulares de la SVD, N_s es el número de valores singulares que se conservan en la aproximación POD y N_r es el número total de valores singulares. En las Figuras B.12a y B.12b se pueden observar los espectros SVD de

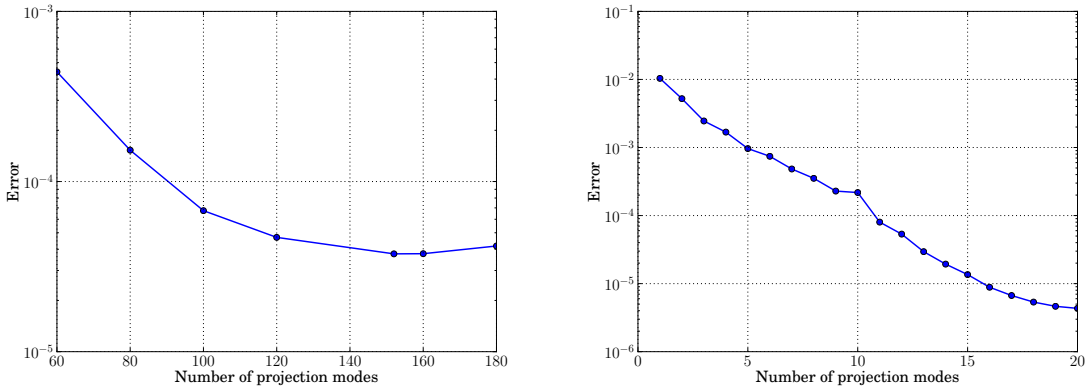
cada formulación HROM. De estos datos que se espere que el HROM asociado al marco de referencia móvil sea más eficiente, ya que las señales involucradas en este caso son más compresibles que en el caso del marco de referencia fijo. Efectivamente, esto es así, como puede apreciarse en las Figuras B.13a y B.13b el error de la aproximación que se obtiene con el HROM asociado al marco de referencia móvil es menor que el que se obtiene con el marco de referencia fijo, además del hecho de que el número de modos de proyección y gappy empleados es mucho menor. Todo esto se traduce en que los factores de aceleración obtenidos sean mayores en un caso que en el otro. Esto se refleja en las Figuras B.14a y B.14b, en donde se estudian los factores de aceleración obtenidos en función del número de modos de proyección.



(a) Marco de referencia fijo.

(b) Marco de referencia móvil.

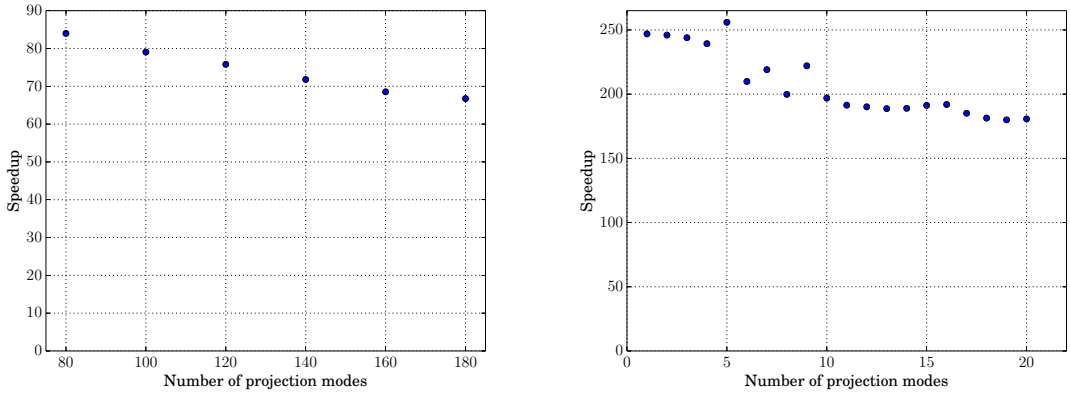
Figura B.12: Espectros SVD de cada uno de los términos involucrados.



(a) Marco de referencia fijo. Error relativo tomando $n_g = 200$ y $n_s = 400$.

(b) Marco de referencia móvil. Error relativo tomando $n_g = n_s = 35$.

Figura B.13: Error obtenido variando el número de modos de proyección.



(a) Marco de referencia fijo. Speedup tomando $n_g = 260$ y $n_s = 520$.

(b) Marco de referencia móvil. Speedup tomando $n_g = n_s = 35$.

Figura B.14: Speedup variando el número de modos de proyección.

B.6 Aplicación: unión soldada de una central nuclear

Esta tesis tiene el objetivo particular de emplear parte de las herramientas desarrolladas en la predicción de las tensiones residuales de soldadura (WRS) en una unión soldada de una central nuclear tipo. Este tipo de aplicación permite validar la robustez del marco computacional introducido, y consecuentemente mejorar las técnicas existentes por medio de dar respuesta a las preguntas que emergen del análisis de los resultados obtenidos.

El agrietamiento por esfuerzos y corrosión (SCC) es uno de los mecanismos de degradación más peligrosos que se asocia al crecimiento de fisuras circunferenciales de longitud considerable. El SCC está impulsado principalmente por las tensiones residuales de soldadura. Es por esto que para poder predecir el crecimiento de fisuras sea primordial conocer las tensiones residuales presentes. En este trabajo se aplica el marco computacional desarrollado al estudio de una de las soldaduras presentes en la tubería principal de una central nuclear tipo, específicamente a la soldadura de las boquillas del recipiente de presión del reactor con la rama fría.

La sección longitudinal de la tubería involucrada puede observarse en la Figura B.15, donde el diámetro interno es igual a 750mm. Debido a las grandes dimensiones de la geometría, sería prohibitivo un análisis 3D del problema, optándose por un análisis 2D axisimétrico. Las condiciones de borde del problema térmico están dadas por condiciones de borde Robin en las caras internas y externas, incluidos los bordes de los cordones que se van depositando. Ambos extremos del tubo son modelados con condiciones de borde adiabáticas. En el caso del problema mecánico, se imponen condiciones de borde de tracción libre en todo el borde, y el sistema se sujeta por medio de los resortes que se muestran en la Figura B.15.

A continuación se puede encontrar un detalle de los resultados obtenidos. El campo de temperatura en la mitad del proceso de soldadura se puede observar en la Figura B.16. En la Figura B.17 se puede apreciar la tensión residual circunferencial calculada al final del proceso. En las Figuras B.18a y B.18b se pueden observar las componentes de las tensiones residuales a lo largo de una línea que pasa por el centro de la soldadura sigu-

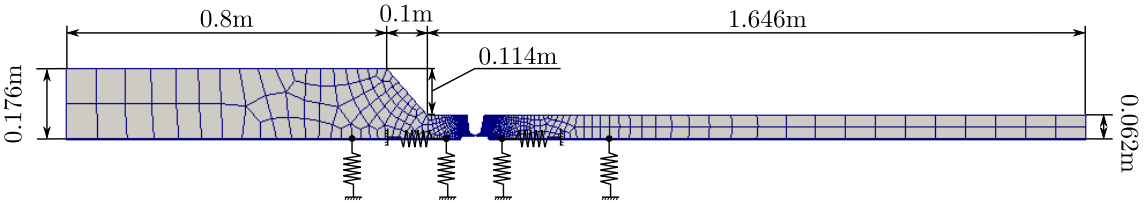


Figura B.15: Sección longitudinal de la tubería implicada. Especificación del dominio y de la malla que definen el análisis 2D axisimétrico del problema.

iendo la dirección radial, denominada línea **C**. Adicionalmente, se comparan los resultados obtenidos considerando y no considerando la contribución del calor latente del cambio de fase sólido/líquido. De esta comparación se puede concluir que para este ejemplo, la contribución del calor latente del cambio de fase sólido/líquido sobre las tensiones residuales no es significativa.

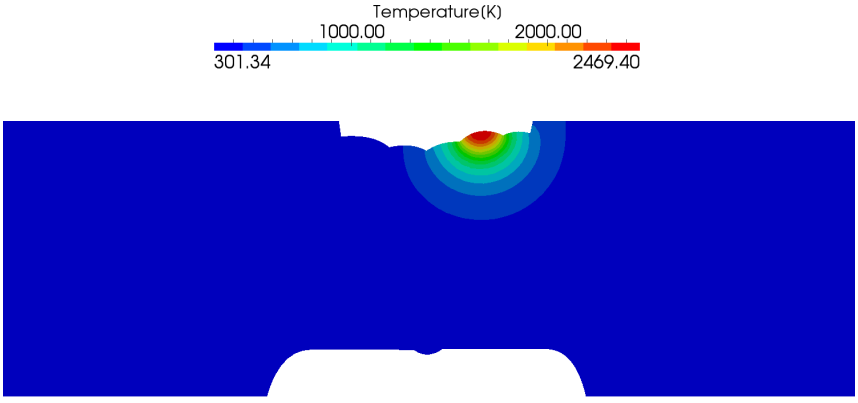


Figura B.16: Temperatura en la mitad del proceso de soldadura.

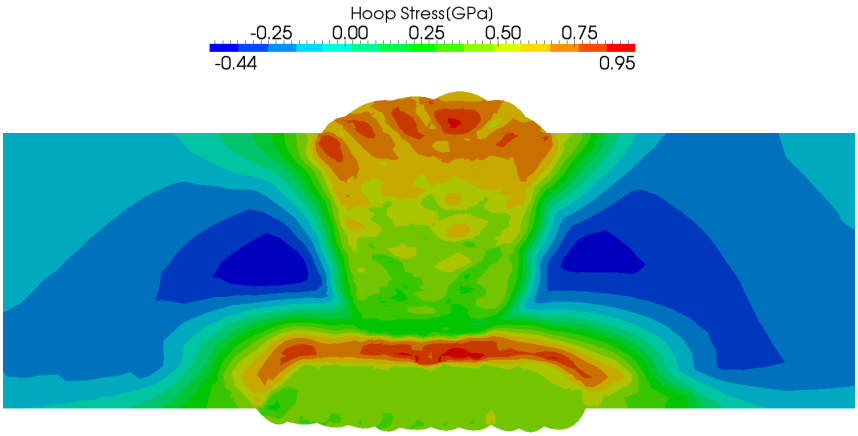
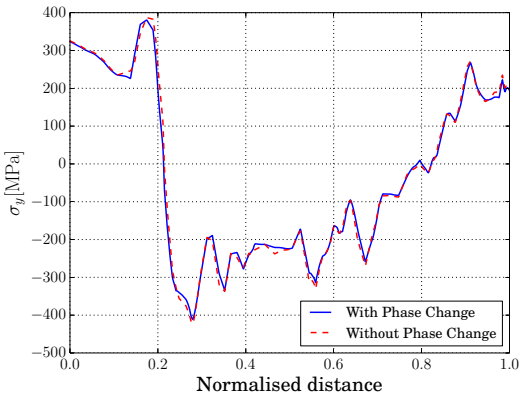
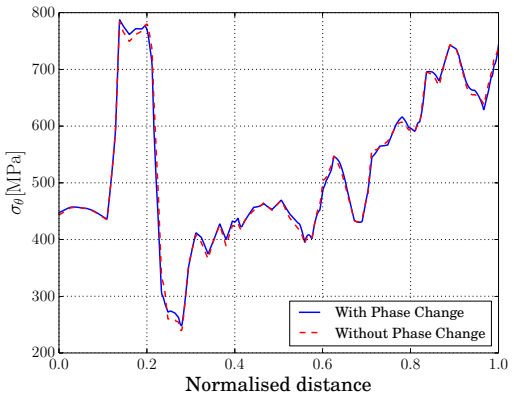


Figura B.17: Tensión residual circunferencial.



(a) Esfuerzo axial a lo largo de la línea **C**.



(b) Esfuerzo circunferencial a lo largo de la línea **C**.

Figura B.18: Componentes de las tensiones residuales a lo largo de la línea **C**. Comparación de los resultados obtenidos considerando y no considerando la contribución del calor latente del cambio de fase sólido/líquido.

B.7 Conclusiones

Principales contribuciones de este trabajo. En esta tesis se consideraron distintos aspectos del modelado multifísico del proceso de soldadura. Las principales contribuciones originales fueron hechas en los siguientes temas listados a continuación:

- Problema térmico: se presentó un nuevo elemento finito enriquecido para resolver problemas de cambio de fase isotérmicos. Es importante remarcar que el método propuesto evita el uso de una ecuación auxiliar para determinar la posición del enriquecimiento, la cual es usual en formulaciones "level set". Este punto establece un enfoque moderno en lo que respecta a técnicas de enriquecimiento.

- Evolución microestructural: se propuso un modelo microestructural para la aleación de Titanio Ti6Al4V capaz de ser aplicado a procesos caracterizados por historias térmicas arbitrarias, que a pesar de diferir de los modelos existentes en pequeños detalles, permite analizar en profundidad la cinética de las transformaciones microestructurales del Titanio por medio de considerar diagramas de flujo. Así, el uso de diagramas de flujo se adoptó como herramienta fundamental para definir la cinética de las transformaciones microestructurales. Esto permitió desarrollar un marco computacional lo bastante flexible como para modelar el comportamiento microestructural de un amplio rango de materiales.

- Modelos de Orden Reducido: se analizaron distintas estrategias para construir Modelos de Orden Hiper-Reducido (HROMs) para el problema no lineal de cambio de fase sólido/líquido. Se introdujo una nueva propuesta basada en hiper-reducir separadamente cada uno de los términos que contribuyen al residuo del problema en cuestión. Adicionalmente, se consideró el uso de marcos de referencia móvil para formular HROMs para el problema de soldadura.

Se revisaron los conceptos que involucra la descripción termo-mecánica de cuerpos que experimentan cambios de fase sólido/líquido, y fueron implementados en el software

de elementos finitos OOFELIE. Para poder lidiar con el costo computacional asociado a las simulaciones de procesos de soldadura, se llevó a cabo la paralelización de OOFELIE, aprovechando de esta manera los recursos ofrecidos por arquitecturas de memoria compartida distribuida.

Se resolvió un problema de importancia tecnológica. Específicamente, se aplicó el marco computacional desarrollado a la simulación de la soldadura de una de las boquillas del recipiente de presión del reactor con la rama fría de una central nuclear tipo. Esta aplicación permitió predecir las tensiones residuales de soldadura presentes en la unión soldada, un factor que influencia en gran medida a la formación de fisuras. Adicionalmente, se pudo establecer que para este problema la contribución del calor latente del cambio de fase sólido/líquido a las tensiones residuales puede ser despreciada.

Trabajo futuro. Los siguientes temas son objeto de investigación en el futuro:

- Comportamiento mecánico de cuerpos que experimentan cambios de fase sólido/líquido: se estudiará utilizar una formulación Lagrangiana actualizada y un algoritmo para posicionar correctamente los elementos a medida que se activan. Adicionalmente, se considerará el modelado del tratamiento térmico post soldadura.

- Modelos de orden reducido: se estudiará en mayor detalle el mal condicionamiento expuesto por las formulaciones HROM, buscando métodos que eviten este tipo de problemas. Además, se estudiará la hiper-reducción de problemas de soldadura con deposición de material.

- Evolución microestructural: se tiene como trabajo futuro estudiar modelos microestructurales formulados en la meso-escala. También, gran parte del trabajo estará dirigido a obtener formulaciones meso-escala que sean tratables desde el punto de vista del costo computacional.

- Modelado de problemas de cambio de fase sólido/líquido: en un trabajo futuro se tiene previsto extender el elemento finito enriquecido presentado en esta tesis a problemas 2D y 3D.

Bibliography

[1] Ronda, J., and Oliver, G., 2000. "Consistent thermo-mechano-metallurgical model of welded steel with unified approach to derivation of phase evolution laws and transformation-induced plasticity". *Computer Methods in Applied Mechanics and Engineering,* **189**(2), pp. 361 – 418.

[2] Felippa, C., Park, K., and Farhat, C., 2001. "Partitioned analysis of coupled mechanical systems". *Computer Methods in Applied Mechanics and Engineering,* **190**(24-25), pp. 3247–3270.

[3] Lindgren, L.-E., 2007. *Computational Welding Mechanics: thermomechanical and microstructural simulations*. CRC Press.

[4] Koistinen, D., and Marburger, R., 1959. "A general equation prescribing the extent of the austenite-martensite transformation in pure iron-carbon alloys and plain carbon steels". *Acta Metallurgica,* **7**(1), pp. 59–60.

[5] Oddy, A., Goldak, J., and McDill, J., 1989. "Transformation effects in the 3D finite element analysis of welds". *Proc. of the 2nd International Conf. on trends in Welding research, Gatlinburg Tennessee USA*.

[6] Xiao, N., Chen, Y., Li, D., and Li, Y., 2012. "Progress in mesoscopic modeling of microstructure evolution in steels". *Science China Technological Sciences,* **55**(2), pp. 341–356.

[7] Kirkaldy, J., and Venugopalan, D., 1984. "Phase transformations in ferrous alloys". pp. 125–148.

[8] Oddy, A., McDill, J., and Karlsson, L., 1996. "Microstructural predictions including arbitrary thermal histories, reaustenization and carbon segregation effects". *Canadian Metallurgical Quarterly,* **35**(3), pp. 275 – 283.

[9] Kim, D., 2012. "Prediction of microstructure evolution of heat-affected zone in gas metal arc welding of steels". PhD Thesis, Faculty of the Graduate School of The University of Texas at Austin.

[10] Watt, D., Coon, L., Bibby, M., Goldak, J., and Henwood, C., 1988. "An algorithm for modelling microstructural development in weld heat-affected zones (part a) reaction kinetics". *Acta Metallurgica,* **36**(11), pp. 3029 – 3035.

[11] Johnson, W. A., and Mehl, R. F., 1939. "Reaction kinetics in processes of nucleation and growth". *Trans. Am. Inst. Min. Metall. Eng.,* **135**, pp. 416–442.

[12] Avrami, M., 1941. "Granulation, phase change, and microstructure kinetics of phase change. iii". *The Journal of Chemical Physics,* **9**(2), pp. 177–184.

[13] Kolmogorov, A., 1937. "On the Statistical Theory of Metal Crystallization". *Izv. Akad. Nauk SSSR, Ser. Math,* **1**, pp. 335–360.

[14] Perlade, A., Bouaziz, O., and Furnémont, Q., 2003. "A physically based model for trip-aided carbon steels behaviour". *Materials Science and Engineering: A,* **356**(1-2), pp. 145–152.

[15] Bouquerel, J., Verbeken, K., and De Cooman, B., 2006. "Microstructure-based model for the static mechanical behaviour of multiphase steels". *Acta Materialia,* **54**(6), pp. 1443–1456.

[16] Oddy, A. S., Goldak, J. A., and McDill, J. M. J., 1992. "Transformation plasticity and residual stresses in single-pass repair welds". *Journal of Pressure Vessel Technology,* **114**(1), pp. 33–38.

[17] Goldak, J., and Akhalaghi, M., 2005. *Computational Welding Mechanics*. Springer.

[18] Anca, A., Fachinotti, V., Escobar-Palafox, G., and Cardona, A., 2011. "Computational modelling of shaped metal deposition". *International Journal for Numerical Methods in Engineering,* **85**(1), pp. 84–106.

[19] Anca, A., Cardona, A., Risso, J., and Fachinotti, V., 2011. "Finite element modeling of welding processes". *Applied Mathematical Modelling,* **35**(2), pp. 688–707.

[20] Idelsohn, S., Storti, M., and Crivelli, L., 1994. "Numerical methods in phase-change problems". *Archives of Computational Methods in Engineering,* **1**, pp. 49–74.

[21] Lynch, D. R., and O'Neill, K., 1981. "Continuously deforming finite elements for the solution of parabolic problems, with and without phase change". *International Journal for Numerical Methods in Engineering,* **17**(1), pp. 81–96.

[22] Beckett, G., MacKenzie, J. A., and Robertson, M. L., 2001. "A moving mesh finite element method for the solution of two-dimensional stefan problems". *Journal of Computational Physics,* **168**(2), pp. 500–518.

[23] Baines, M. J., Hubbard, M. E., and Jimack, P. K., 2005. "A moving mesh finite element algorithm for the adaptive solution of time-dependent partial differential equations with moving boundaries". *Applied Numerical Mathematics,* **54**(3-4), pp. 450–469.

[24] Voller, V., and Cross, M., 1981. "Accurate solutions of moving boundary problems using the enthalpy method". *International Journal of Heat and Mass Transfer,* **24**(3), pp. 545–556.

[25] Tamma, K. K., and Namburu, R. R., 1990. "Recent advances, trends and new perspectives via enthalpy-based finite element formulations for applications to solidification problems". *International Journal for Numerical Methods in Engineering,* **30**(4), pp. 803–820.

[26] Runnels, S. R., and Carey, G. F., 1991. "Finite element simulation of phase change using capacitance methods". *Numerical Heat Transfer, Part B: Fundamentals,* **19**(1), pp. 13–30.

[27] Celentano, D., Oñate, E., and Oller, S., 1994. "A temperature-based formulation for finite element analysis of generalized phase-change problems". *International Journal for Numerical Methods in Engineering,* **37**(20), pp. 3441–3465.

[28] Fachinotti, V. D., Cardona, A., and Huespe, A. E., 1999. "A fast convergent and accurate temperature model for phase-change heat conduction". *International Journal for Numerical Methods in Engineering,* **44**(12), pp. 1863–1884.

[29] Crivelli, L. A., and Idelsohn, S. R., 1986. "A temperature-based finite element solution for phase-change problems". *International Journal for Numerical Methods in Engineering,* **23**(1), pp. 99–119.

[30] Storti, M., Crivelli, L. A., and Idelsohn, S. R., 1987. "Making curved interfaces straight in phase-change problems". *International Journal for Numerical Methods in Engineering,* **24**(2), pp. 375–392.

[31] Fachinotti, V. D., 2001. "Modelado numérico de fenómenos termomecánicos en la solidificación y enfriamiento de aceros obtenidos por colada continua". PhD Thesis, Facultad de Ingeniería y Ciencias Hídricas, Universidad Nacional del Litoral.

[32] Nouy, A., 2010. "A priori model reduction through proper generalized decomposition for solving time-dependent partial differential equations". *Computer Methods in Applied Mechanics and Engineering,* **199**(23-24), pp. 1603 – 1626.

[33] Kunisch, K., and Volkwein, S., 2002. "Galerkin proper orthogonal decomposition methods for a general equation in fluid dynamics". *SIAM Journal on Numerical Analysis,* **40**(2), pp. 492–515.

[34] Bergmann, M., Bruneau, C.-H., and Iollo, A., 2009. "Enablers for robust POD models". *Journal of Computational Physics,* **228**(2), pp. 516 – 538.

[35] Néron, D., and Ladevèze, P., 2010. "Proper generalized decomposition for multiscale and multiphysics problems". *Archives of Computational Methods in Engineering,* **17**(4), pp. 351–372.

[36] Chinesta, F., Ammar, A., and Cueto, E., 2010. "Recent advances and new challenges in the use of the proper generalized decomposition for solving multidimensional models". *Archives of Computational Methods in Engineering,* **17**(4), pp. 327–350.

[37] Patterson, D. A., and Hennessy, J. L., 2011. *Computer Architecture: A Quantitative Approach, 5th Edition.* Morgan Kaufmann.

[38] Farhat, C., and Crivelli, L., 1989. "A general approach to nonlinear {FE} computations on shared-memory multiprocessors". *Computer Methods in Applied Mechanics and Engineering,* **72**(2), pp. 153 – 171.

[39] Cardona, A., Klapka, I., and Geradin, M., 1994. "Design of a new finite element programming environment". *Engineering Computations,* **11**, pp. 365–381.

[40] Lindgren, L., 2007. *Computational Welding Mechanics: Thermomechanical and Microstructural Simulations*. Woodhead Publishing.

[41] Anca, A., Fachinotti, V. D., Escobar-Palafox, G., and Cardona, A., 2011. "Computational modelling of shaped metal deposition". *International Journal for Numerical Methods in Engineering,* **85**(1), pp. 84–106.

[42] Fachinotti, V., Cardona, A., Cosimo, A., Baufeld, B., and Van der Biest, O., 2010. "Evolution of temperature during shaped metal deposition: Finite element predictions vs. observations". In Mecánica Computacional, E. Dvorkin, M. Goldschmit, and M. Storti, eds., Vol. XXIX, pp. 4915–4926 (complete article).

[43] Basombrío, F., 1997. "El problema de dos fases en materiales heterogéneos. aplicaciones". *Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería,* **13**(3), pp. 351–366.

[44] Koric, S., and Thomas, B. G., 2006. "Efficient thermo-mechanical model for solidification processes". *International Journal for Numerical Methods in Engineering,* **66**(12), pp. 1955–1989.

[45] Salcudean, M., and Abdullah, Z., 1988. "On the numerical modelling of heat transfer during solidification processes". *International Journal for Numerical Methods in Engineering,* **25**(2), pp. 445–473.

[46] Nallathambi, A. K., Specht, E., and Bertram, A., 2009. "Computational aspects of temperature-based finite element technique for the phase-change heat conduction problem". *Computational Materials Science,* **47**(2), p. 332.

[47] Davey, K., and Mondragon, R., 2010. "A non-physical enthalpy method for the numerical solution of isothermal solidification". *International Journal for Numerical Methods in Engineering,* **84**(2), pp. 214–252.

[48] Fries, T.-P., and Belytschko, T., 2010. "The extended/generalized finite element method: An overview of the method and its applications". *International Journal for Numerical Methods in Engineering,* **84**(3), pp. 253–304.

[49] Simone, A., 2007. "Partition of unity-based discontinuous finite elements: GFEM, PUFEM, XFEM". *Revue Européenne de Génie Civil,* **11**(7-8), p. 1045.

[50] Ausas, R. F., Buscaglia, G. C., and Idelsohn, S. R., 2012. "A new enrichment space for the treatment of discontinuous pressures in multi-fluid flows". *International Journal for Numerical Methods in Fluids,* **70**(7), pp. 829–850.

[51] Fries, T., 2008. "A corrected XFEM approximation without problems in blending elements". *International Journal for Numerical Methods in Engineering,* **75**(5), p. 503.

[52] Chessa, J., Smolinski, P., and Belytschko, T., 2002. "The extended finite element method (XFEM) for solidification problems". *International Journal for Numerical Methods in Engineering,* **53**(8), p. 1959.

[53] Bernauer, M., and Herzog, R., 2011. "Implementation of an X-FEM solver for the classical two-phase Stefan problem". *Journal of Scientific Computing,* **52**(2), pp. 1–23.

[54] Sethian, J. A., 1996. *Level Set Methods and Fast Marching Methods.* Cambridge University Press.

[55] Ji, H., Chopp, D., and Dolbow, J. E., 2002. "A hybrid extended finite element/level set method for modeling phase transformations". *International Journal for Numerical Methods in Engineering,* **54**(8), pp. 1209–1233.

[56] Merle, R., and Dolbow, J., 2002. "Solving thermal and phase change problems with the extended finite element method". *Computational Mechanics,* **28**(5), p. 339.

[57] Ladevèze, P., and Simmonds, J. G., 1999. *Nonlinear computational structural mechanics: new approaches and non-incremental methods of calculation.* Springer.

[58] Fries, T.-P., and Zilian, A., 2009. "On time integration in the XFEM". *International Journal for Numerical Methods in Engineering,* **79**(1), pp. 69–93.

[59] Coppola-Owen, A. H., and Codina, R., 2005. "Improving eulerian two-phase flow finite element approximation with discontinuous gradient pressure shape functions". *International Journal for Numerical Methods in Fluids,* **49**(12), pp. 1287–1304.

[60] Wriggers, P., 2008. *Nonlinear Finite Element Methods.* Springer, Berlin.

[61] Kelley, C. T., 1999. *Iterative Methods for Optimization.* Society for Industrial and Applied Mathematics, Philadelphia.

[62] Nigro, N., Huespe, A., and Fachinotti, V., 2000. "Phasewise numerical integration of finite element method applied to solidification processes". *International Journal of Heat and Mass Transfer,* **43**(7), pp. 1053–1066.

[63] Soghrati, S., Aragón, A. M., Armando Duarte, C., and Geubelle, P. H., 2012. "An interface-enriched generalized FEM for problems with discontinuous gradient fields". *International Journal for Numerical Methods in Engineering,* **89**(8), pp. 991–1008.

[64] Tarzia, D. A., 2011. *Advanced Topics in Mass Transfer.* InTech, ch. 20, pp. 439–484.

[65] Alexiades, V., and Solomon, A. D., 1993. *Mathematical Modeling of Melting and Freezing Processes.* Hemisphere Publishing Corporation, Taylor and Francis Group.

[66] Goldak, J., Chakravarti, A., and Bibby, M., 1984. "A new finite element model for welding heat sources". *Metallurgical and Materials Transactions B,* **15**, pp. 299–305.

[67] Kelly, S. M., 2004. "Thermal and microstructure modeling of metal deposition processes with application to Ti-6Al-4V". PhD Thesis, Faculty of the Virginia Polytechnic Institute and State University.

[68] Moelans, N., Blanpain, B., and Wollants, P., 2008. "An introduction to phase-field modeling of microstructure evolution". *Calphad,* **32**(2), pp. 268 – 294.

[69] Bohl, M., and Rynn, M., 2007. *Tools for Structured and Object-Oriented Design.* Pearson Prentice Hall.

[70] Lee, M.-G., Kim, S.-J., Han, H. N., and Jeong, W. C., 2009. "Implicit finite element formulations for multi-phase transformation in high carbon steel". *International Journal of Plasticity,* **25**(9), pp. 1726 – 1758.

[71] Babu, S., Kelly, S., Specht, E., Palmer, T., and Elmer, J., 2005. "Measurement of phase transformation kinetics during repeated thermal cycling of Ti-6Al-4V using time-resolved X-ray diffraction". Vol. 2, pp. 503–508.

[72] Murgau, C., Pederson, R., and Lindgren, L., 2012. "A model for Ti6Al4V microstructure evolution for arbitrary temperature changes". *Modelling and Simulation in Materials Science and Engineering,* **20**(5).

[73] Rios, P. R., and Villa, E., 2011. "Simultaneous and sequential transformations". *Acta Materialia,* **59**(4), pp. 1632 – 1643.

[74] Jones, S., and Bhadeshia, H., 1997. "Kinetics of the simultaneous decomposition of austenite into several transformation pproducts". *Acta Materialia,* **45**(7), pp. 2911 – 2920.

[75] Crespo, A., 2011. "Modelling of heat transfer and phase transformations in the rapid manufacturing of titanium components". In Convection and Conduction Heat Transfer, A. Amimul, ed., InTech.

[76] Mittemeijer, E. J., 2011. *Fundamentals of Materials Science.* Springer.

[77] Geijsalers, H., 2003. "Numerical simulation of stresses due to solid state transformations: The simulation of laser hardening". PhD Thesis, University of Twente, Enschede.

[78] Scheil, E., 1935. "Anlaufzeit der austenitumwandlung". *Archiv für Eisenhüttenwesen,* **8**, p. 565.

[79] Cahn, J. W., 1956. "Transformation kinetics during continuous cooling". *Acta Metallurgica,* **4**(6), pp. 572–575.

[80] Hildenwall, B., and Ericsson, T., 1977. "Prediction of residual stresses in case-hardening steels". *Hardenability Concepts with Applications to Steel*, pp. 579–606.

[81] Gil Mur, F., Rodriguez, D., and Planell, J., 1996. "Influence of tempering temperature and time on the α'-Ti-6Al-4V martensite". *Journal of Alloys and Compounds,* **234**(2), pp. 287–289.

[82] Otsuka, K., and Wayman, C. M., 1999. *Shape memory materials.* Cambridge University Press.

[83] Fan, Y., Cheng, P., Yao, Y. L., Yang, Z., and Egland, K., 2005. "Effect of phase transformations on laser forming of Ti-6Al-4V alloy". *Journal of Applied Physics,* **98**(1), p. 013518.

[84] Charles, C., 2008. "Modelling microstructure evolution of weld deposited Ti-6Al-4V". PhD Thesis, Luleå University of Technology.

[85] Cosimo, A., Fachinotti, V., and Cardona, A., 2013. "An enrichment scheme for solidification problems". *Computational Mechanics,* **52**(1), pp. 17–35.

[86] Şimşir, C., 2008. "3D FEM Simulation of Quenching for Prediction of the Microstructure, Residual Stresses and Distortion". PhD Thesis, Middle East Technical University.

[87] Ahmed, T., and Rack, H., 1998. "Phase transformations during cooling in $\alpha+\beta$ titanium alloys". *Materials Science and Engineering: A,* **243**(1-2), pp. 206–211.

[88] Sourmail, T., 2001. "Precipitation in creep resistant austenitic stainless steels". *Materials science and technology,* **17**(1), pp. 1–14.

[89] Dalcin, L. D., Paz, R. R., Kler, P. A., and Cosimo, A., 2011. "Parallel distributed computing using Python". *Advances in Water Resources,* **34**(9), pp. 1124 – 1139.

[90] Behnel, S., Bradshaw, R., Citro, C., Dalcin, L., Seljebotn, D., and Smith, K., 2011. "Cython: The best of both worlds". *Computing in Science Engineering,* **13**(2), pp. 31 –39.

[91] Beazley, D. M., 1996. "SWIG: An Easy to Use Tool for Integrating Scripting Languages with C and C++". In Proceedings of the 4th Conference on USENIX Tcl/Tk Workshop, 1996 - Volume 4, TCLTK'96, USENIX Association, pp. 15–15.

[92] Şimşir, C., and Gür, C. H., 2008. "A FEM based framework for simulation of thermal treatments: Application to steel quenching". *Computational Materials Science,* **44**(2), pp. 588 – 600.

[93] Wilson, B., and Weins, W., 2000. "Retained austenite and tooling failure case studies". In 20 th ASM Heat Treating Society Conference, pp. 566–573.

[94] Ogden, R., 1997. *Non-linear Elastic Deformations.* Dover Civil and Mechanical Engineering Series. Dover Publications.

[95] Lubliner, J., 2013. *Plasticity Theory.* Courier Dover Publications.

[96] Simo, J. C., and Hughes, T. J. R., 1998. *Computational Inelasticity.* Springer.

[97] Hughes, T., 2000. *The finite element method: linear static and dynamic finite element analysis.* Dover Publications.

[98] Lindgren, L.-E., 2001. "Finite element modeling and simulation of welding. Part 2: Improved material modeling". *Journal of Thermal Stresses,* **24**(3), pp. 195–231.

[99] Mullins, J., and Gunnars, J., 2009. Influence of hardening model on weld residual stress distribution. Tech. Rep. 2009:16, Inspecta Technology AB, Stockholm, Sweden, June.

[100] Voce, E., 1955. "A practical strain-hardening function". *Metallurgica,* **51**, pp. 219–226.

[101] Donea, J., and Huerta, A., 2003. *Finite Element Methods for Flow Problems.* Finite Element Methods for Flow Problems. John Wiley & Sons.

[102] Risso, J., Huespe, A., and Cardona, A., 2003. "A generalized plane strain formulation for continuous casting simulation". In Mecánica Computacional Vol. XXII, M. B. Rosales, V. H. Cortínez y D. V. Bambill, eds., pp. 2513–2527.

[103] Lundbäck, A., and Lindgren, L.-E., 2011. "Modelling of metal deposition". *Finite Elements in Analysis and Design,* **47**(10), pp. 1169 – 1177.

[104] Lindgren, L.-E., and Hedblom, E., 2001. "Modelling of addition of filler material in large deformation analysis of multipass welding". *Communications in Numerical Methods in Engineering,* **17**(9), pp. 647–657.

[105] Weiner, J., and Boley, B., 1963. "Elasto-plastic thermal stresses in a solidifying body". *Journal of the Mechanics and Physics of Solids,* **11**(3), pp. 145 – 154.

[106] Carslaw, H., and Jaeger, J., 1986. *Conduction of heat in solids. 2nd edition, Edition anglaise*. Oxford Science Publications. Clarendo Press.

[107] Commend, S., Truty, A., and Zimmermann, T., 2004. "Stabilized finite elements applied to elastoplasticity: I. mixed displacement-pressure formulation". *Computer Methods in Applied Mechanics and Engineering,* **193**(33-35), pp. 3559 – 3586.

[108] Lee, M., Malaya, N., and Moser, R. D., 2013. "Petascale direct numerical simulation of turbulent channel flow on up to 786k cores". In Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis, SC '13, ACM, pp. 61:1–61:11.

[109] Hennessy, J., Patterson, D., and Asanović, K., 2012. *Computer Architecture: A Quantitative Approach*. Computer Architecture: A Quantitative Approach. Morgan Kaufmann/Elsevier.

[110] Intel Math Kernel Library. http://software.intel.com/en-us/articles/intel-mkl/.

[111] An Introduction to the Intel QuickPath Interconnect. http://www.intel.com.

[112] OpenMP. http://openmp.org/wp/.

[113] Buntinas, D., Mercier, G., and Gropp, W., 2006. "Design and Evaluation of Nemesis, a Scalable, Low-Latency, Message-Passing Communication Subsystem". In Proceedings of the Sixth IEEE International Symposium on Cluster Computing and the Grid, CCGRID '06, IEEE Computer Society, pp. 521–530.

[114] Buntinas, D., Mercier, G., and Gropp, W., 2006. "Implementation and Shared-Memory Evaluation of MPICH2 over the Nemesis Communication Subsystem". In *Recent Advances in Parallel Virtual Machine and Message Passing Interface*, B. Mohr, J. Träff, J. Worringen, and J. Dongarra, eds., Vol. 4192 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, pp. 86–95.

[115] Kerrisk, M., 2010. *The Linux Programming Interface*. No Starch Press.

[116] Rosen, K. H., 1991. *Discrete Mathematics and Its Applications (2Nd Ed.)*. McGraw-Hill, Inc., New York, NY, USA.

[117] Karypis, G., and Kumar, V., 1998. A software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices. Tech. rep., Univ. of Minnesota.

[118] Rojas Fredini, E., Benitez, F., Cosimo, A., and Cardona, A., 2012. "Paralelización de un código de elementos finitos en multiprocesadores de memoria compartida". In Mecánica Computacional, A. Cardona, P. H. Kohan, R. Quinteros, and M. Storti, eds., Vol. XXXI, pp. 3153–3164 (complete article).

[119] Ryckelynck, D., 2002. "An a priori model reduction method for thermomechanical problems [Réduction a priori de modèles thermomécaniques]". *Comptes Rendus - Mecanique,* **330**(7), pp. 499–505.

[120] Berinde, R., and Indyk, P., 2008. *Sparse recovery using sparse random matrices.*

[121] Eldar, Y., and Kutyniok, G., 2012. *Compressed Sensing: Theory and Applications.* Compressed Sensing: Theory and Applications. Cambridge University Press.

[122] Carlberg, K., Bou-Mosleh, C., and Farhat, C., 2011. "Efficient non-linear model reduction via a least-squares Petrov-Galerkin projection and compressive tensor approximations". *International Journal for Numerical Methods in Engineering,* **86**(2), pp. 155–181.

[123] Cardona, A., and Idelsohn, S., 1986. "Solution of non-linear thermal transient problems by a reduction method". *International Journal for Numerical Methods in Engineering,* **23**(6), pp. 1023–1042.

[124] Hernández, J., Oliver, J., Huespe, A., and Caicedo, M., 2012. *High-performance Model Reduction Procedures in Multiscale Simulations.* Monograph CIMNE. International Center for Numerical Methods in Engineering.

[125] Carlberg, K., Farhat, C., Cortial, J., and Amsallem, D., 2013. "The {GNAT} method for nonlinear model reduction: Effective implementation and application to computational fluid dynamics and turbulent flows". *Journal of Computational Physics,* **242**(0), pp. 623 – 647.

[126] Baiges, J., Codina, R., and Idelsohn, S., 2013. "Explicit reduced-order models for the stabilized finite element approximation of the incompressible navier–stokes equations". *International Journal for Numerical Methods in Fluids,* **72**(12), pp. 1219–1243.

[127] Chaturantabut, S., and Sorensen, D., 2009. "Discrete empirical interpolation for nonlinear model reduction". In Decision and Control, 2009 held jointly with the 2009 28th Chinese Control Conference. CDC/CCC 2009. Proceedings of the 48th IEEE Conference on, pp. 4316–4321.

[128] Ryckelynck, D., 2005. "A priori hyperreduction method: an adaptive approach". *Journal of Computational Physics,* **202**(1), pp. 346 – 366.

[129] Sarbandi, B., Cartel, S., Besson, J., and Ryckelynck, D., 2010. "Truncated integration for simultaneous simulation of sintering using a separated representation". *Archives of Computational Methods in Engineering,* **17**(4), pp. 455–463.

[130] Everson, R., and Sirovich, L., 1995. "Karhunen-Loeve procedure for gappy data". *J. Opt. Soc. Am. A,* **12**, pp. 1657–1664.

[131] Astrid, P., 2004. "Reduction of process simulation models: a proper orthogonal decomposition approach". PhD Thesis, Technische Universiteit Eindhoven.

[132] Ryckelynck, D., 2009. "Hyper-reduction of mechanical models involving internal variables". *International Journal for Numerical Methods in Engineering,* **77**(1), pp. 75–89.

[133] González, D., Ammar, A., Chinesta, F., and Cueto, E., 2010. "Recent advances on the use of separated representations". *International Journal for Numerical Methods in Engineering,* **81**(5), pp. 637–659.

[134] Jolliffe, I., 2002. *Principal Component Analysis.* Springer Series in Statistics. Springer.

[135] Sirovich, L., 1987. "Turbulence and the dynamics of coherent structures. I - Coherent structures. II - Symmetries and transformations. III - Dynamics and scaling". *Quarterly of Applied Mathematics,* **45**, Oct., pp. 561–571.

[136] Strang, G., 1993. "The fundamental theorem of linear algebra". *The American Mathematical Monthly,* **100**(9), pp. pp. 848–855.

[137] Galbally, D., Fidkowski, K., Willcox, K., and Ghattas, O., 2010. "Non-linear model reduction for uncertainty quantification in large-scale inverse problems". *International Journal for Numerical Methods in Engineering,* **81**(12), pp. 1581–1608.

[138] Nguyen, N. C., and Peraire, J., 2008. "An efficient reduced-order modeling approach for non-linear parametrized partial differential equations". *International Journal for Numerical Methods in Engineering,* **76**, Oct., pp. 27–55.

[139] Golub, G., and Van Loan, C., 1996. *Matrix Computations.* Johns Hopkins Studies in the Mathematical Sciences. Johns Hopkins University Press.

[140] Nguyen, N. C., Patera, A. T., and Peraire, J., 2008. "A 'best points' interpolation method for efficient approximation of parametrized functions". *International Journal for Numerical Methods in Engineering,* **73**(4), pp. 521–543.

[141] Barrault, M., Maday, Y., Nguyen, N. C., and Patera, A. T., 2004. "An 'empirical interpolation' method: application to efficient reduced-basis discretization of partial differential equations". *Comptes Rendus Mathematique,* **339**(9), pp. 667 – 672.

[142] Fachinotti, V. D., Cardona, A., Baufeld, B., and der Biest, O. V., 2012. "Finite-element modelling of heat transfer in shaped metal deposition and experimental validation". *Acta Materialia,* **60**(19), pp. 6621 – 6630.

[143] Goldak, J., Bibby, M., Moore, J., House, R., and Patel, B., 1986. "Computer modeling of heat flow in welds". *Metallurgical Transactions B,* **17**(3), pp. 587–600.

[144] Material property database. http://www.jahm.com.

[145] Sun, Z., and Moisio, T., 1994. "Melting ratio in laser welding of dissimilar metals". *Journal of Materials Science Letters,* **13**(13), pp. 980–982.

[146] Jones, E., Oliphant, T., Peterson, P., et al., 2001–. SciPy: Open source scientific tools for Python.

[147] Henderson, A., 2007. ParaView Guide, A Parallel Visualization Application. Kitware Inc.

[148] Cosimo, A., Cardona, A., and Idelsohn, S., 2014. "Improving the k-compressibility of hyper reduced order models with moving sources: Applications to welding and phase change problems". *Computer Methods in Applied Mechanics and Engineering,* **274**(0), pp. 237 – 263.

[149] Cosimo, A., and Cardona, A. "Aspectos de implementación informática para el modelado computacional de la evolución microestructural de materiales". *Complete article accepted - XXI Congress on Numerical Methods and their Applications, 23th-26th September 2014, Bariloche, Patagonia, Argentina.*

[150] Cosimo, A., Cardona, A., Novara, P., and Calvo, N. "Weld residual stresses modelling. Application to a nuclear power plant welded joint.". *Complete article accepted - XXI Congress on Numerical Methods and their Applications, 23th-26th September 2014, Bariloche, Patagonia, Argentina.*

[151] Cavalieri, F. J., Cosimo, A., and Cardona, A. "Estudio de desgaste en válvulas de motores de combustión interna mediante el método de los elementos finitos y ensayos experimentales". *Complete article accepted - XXI Congress on Numerical Methods and their Applications, 23th-26th September 2014, Bariloche, Patagonia, Argentina.*

[152] Cosimo, A., and Cardona, A. "Modelación microestructural computacional de procesos con historias térmicas arbitrarias. Aplicación: aleación de Titanio Ti6Al4V". *Complete article accepted - International Congress of Metallurgy and Materials SAM-CONAMET / IBEROMAT 2014, 21th-24th October 2014, Santa Fe, Argentina.*

[153] Cosimo, A., Cardona, A., and Idelsohn, S., 2013. "Modelos de orden reducido para el problema térmico de soldadura". In Mecánica Computacional, C. García Garino, A. Mirasso, M. Storti, and M. Tornello, eds., Vol. XXXII, pp. 3151–3163 (complete article).

[154] Cosimo, A., and Cardona, A., 2012. "Problemas de choque térmico en el método de los elementos finitos". In Mecánica Computacional, A. Cardona, P. H. Kohan, R. Quinteros, and M. Storti, eds., Vol. XXXI, pp. 1937–1950 (complete article).

[155] Cosimo, A., Fachinotti, V., and Cardona, A., 2012. "Temperature gradient discontinuity aware numerical scheme for solidification problems (complete article)". In Proceedings of the 10th World Congress on Computational Mechanics.

[156] Cosimo, A., Rojas Fredini, E., Benitez, F., and Cardona, A., 23-24 July 2012. Parallelization of a finite element code on shared memory multiprocessors. RISC High-Performance Computing Latin America Symposium (HPCLATAM). Buenos Aires, Argentina.

[157] Fachinotti, V., Cardona, A., Cosimo, A., Baufeld, B., and Van der Biest, O., 2010. "Evolution of temperature during shaped metal deposition: Finite element predictions vs. observations". In Mecánica Computacional, E. Dvorkin, M. Goldschmit, and M. Storti, eds., Vol. XXIX, pp. 4915–4926 (complete article).

Doctorado en Ingeniería
Mención Mecánica Computacional

Título de la obra:

Thermo-Mecano-Metallurgical modelling of welding: Application to welded joints in nuclear power plants

Autor: Alejandro Cosimo

Lugar: Santa Fe, Argentina

Palabras Claves:

Computational Welding Mechanics, Enrichment Scheme,
Microstructure Evolution, Titanium alloy Ti6Al4V,
Parallel FEM, Reduced Order Models, HROM